CORA Cork Open Research Archive
Cartlann Taighde Oscailte Chorcaí

Title	Baran: a service-oriented cloud-based user monitoring and data analysis framework
Authors	Hashemi, Mohammad
Publication date	2019
Original Citation	Hashemi, M. 2019. Baran: a service-oriented cloud-based user monitoring and data analysis framework. PhD Thesis, University College Cork.
Type of publication	Doctoral thesis
Rights	© 2019, Mohammad Hashemi. - http://creativecommons.org/licenses/by-nc-nd/3.0/
Download date	2024-11-01 01:18:06
Item downloaded from	https://hdl.handle.net/10468/9462

UCC
University College Cork, Ireland
Coláiste na hOllscoile Corcaigh

Baran

A Service-Oriented Cloud-Based User Monitoring and Data Analysis Framework

Mohammad Hashemi
MSC
112220161

Thesis submitted for the degree of Doctor of Philosophy

NATIONAL UNIVERSITY OF IRELAND, CORK

FACULTY OF SCIENCE

DEPARTMENT OF COMPUTER SCIENCE

September 2019

Head of Department: Prof Cormac J. Sreenan

Supervisors: Dr John Herbert
Prof Cormac J. Sreenan

Research supported by Telecommunications Graduate Initiative (TGI)

Contents

List of Figures

This is to certify that the work I am submitting is my own and has not been submitted for another degree, either at University College Cork or elsewhere. All external references and sources are clearly acknowledged and identified within the contents. I have read and understood the regulations of University College Cork concerning plagiarism.

Mohammad Hashemi

This work is dedicated to all my family and friend, who offered me unconditional love and support and without them, I would not be the person who I am now. Most especially to my mother, father, mother-in-law, and father-in-law. Also to my brothers and sisters who have encouraged me to keep challenging myself.

I would like to give a special appreciation to my beautiful wife, Rezvan Pakdel, who not only has always supported me but also inspired me to try and achieve what I dream.

Finally, I am giving my most profound feeling of love to my joyful son, Sherwin, who made my world a more beautiful place to live.

Thank you all so much. You are all precious to me.

Abstract

Most humans now live in an age of connected digital devices where their interactions with these devices are recorded in a number of ways, by different organisations, under various levels of user control. A user is often unaware of how much data they create, of where the data resides, and of how their data is used. It is challenging for a user to inspect all this personal data, to control the storage and use of the data, and to exploit the data for their benefit in a safe way. These are the challenges that this thesis addresses. The focus is on the user's digital trail of information – i.e. the record of user interactions and associated context information from the digital devices associated with the user. The digital devices associated with a user, and with which they interact, might include smartphones, laptops, smart watches, fitness bands, and smart household appliances. The thesis develops a conceptual model and extensible data structure, called the UDI (User Digital Imprint), which accommodates a variety of digital data from digital devices along with information derived from that data. A software framework, Baran, is developed that implements the UDI and provides services that support the gathering, management and analysis of the user data. The Baran framework allows the user to inspect and analyse their data and supports (under user control) the sharing of the data in order that assistive services for the user can be provided by 3rd parties. The framework is cloud-based and service-oriented, enabling the framework to make use of external services (e.g. machine learning services) and to provide services for external entities (e.g. supplying some subset of user data to a smart coffee maker). Thus, Baran can be extended with both user services and external services, where the sharing of user data with external entities is directly under user control on a per-use basis. By gathering, analysing and making available comprehensive information about a user's digital interactions, Baran also enables study of aspects of User Experience (UX), as might be of interest to UX researchers or product designers. Three case studies are presented to demonstrate aspects of Baran and how it can be used in different scenarios. Rather than a limited-scope, specific solution (such as recent functionality introduced by Google and Apple) Baran provides a general extensible framework that covers a range of user digital interactions on a variety of devices, and enables a wide range of applications, where all data sharing is under explicit user control.

Acknowledgements

Firstly, I would like to express my sincere gratitude to my great supervisor Dr John Herbert for the continuous support of my PhD study, insightful comments, and encouragement. His guidance helped me throughout the research and writing of this thesis. I could not have imagined having a better mentor for my PhD study.

Besides my supervisor, I would like to thank Professor Cormac J. Sreenan, for his kind support. My sincere thanks also go to my MISL colleagues, Mary Noonan, Dapeng Dong, Thuy Truong, Jason Quinlan, Paul Sutton, Ilias Tsompanidis, and Bingchuan Yuan, who provided me with such a friendly, supportive environment. I have also received incredible support from my MaREI colleagues, Jeremy Gault, Barry O'Dwyer, and Declan Dunne. Without their valuable support, it would not have been possible to complete this work. I appreciate the thorough review and feedback from my examiners, Professor Mike Hinchey and Dr Ian Pitt.

This research was supported by the Telecommunications Graduate Initiative (TGI), funded by the Higher Education Authority (HEA) under the Programme for Research in Third-Level Institutions (PTRLI) Cycle 5 and co-funded by the European Regional Development Fund (ERDF). The completion of the PhD was also supported by the MaREI Centre, Environmental Research Institute, UCC.

Chapter 1

Introduction

The digital revolution began in the 1950s with the introduction of digital computers and digital records [Zho13]. For several decades, information systems existed for the most part in parallel but separate from daily life. In recent decades, this has changed; for most humans, digital systems are now ever present due to Internet connectivity and modern devices such as smartphones, embedded digital systems in consumer products and digital sensors. Ofcom (the UK regulatory and competition authority for the broadcasting, telecommunications and post) [Ofc18] reports that in 2018, for the UK, 78% of the population has a smartphone, and adults check the smartphone on average every 12 minutes (those under 24, every 8.6 minutes). Ofcom also reports that for households, 42% have a smart TV, 44% a gaming console, 56% a digital video recorder, 20% smart watches and fitness trackers, 13% smart speakers. While less common, it is already possible to control central heating and lighting using smartphone apps. The term Internet of Things (IoT) describes the embedding of computation and communication capability in everyday devices (Figure 1.1) [Möl16]. The increased adoption of IoT by industry is expanding the number of network-connected user devices [AAS18].

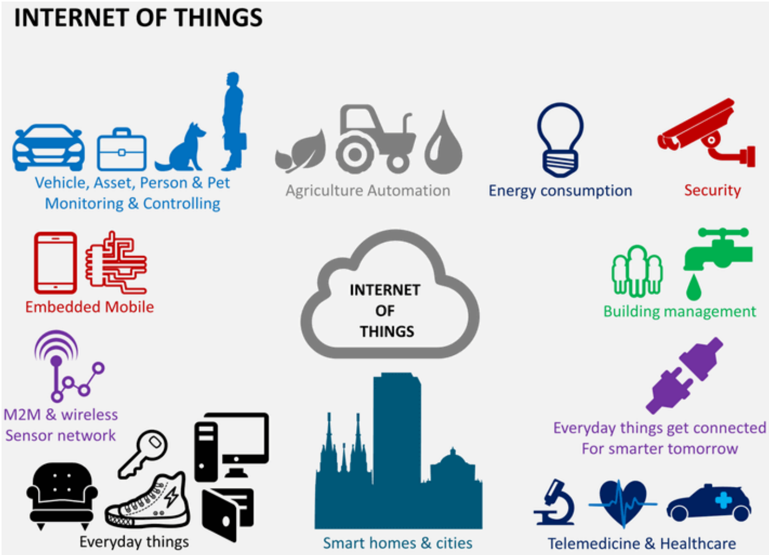

Figure 1.1: Internet of Things (IoT)

These technological changes, and widespread use of social media, mean that life is now experienced both digitally and physically. The positive and negative aspects of these new means of communicating and accessing information are subject to ongoing debate and investigation. Important concerns for users include lack of user control of their data and lack of transparency over the use of their data. Misuse of user data, such as occurred in the Cambridge Analytica scandal, and the need for legislation such as the EU General Data Protection Regulation (GDPR) [PU16], implemented in May 2018, are evidence of the lack of user control over their digital data. Despite the GDPR, a non-profit EU NGO called noyb (none of your business) [noy19], claims that "these [legal] rights are not respected by large parts of the tech industry" and a complaint by noyb has been part of a case that resulted in a €50 million for Google in January 2019. noyb further claims that online streaming services from companies like Amazon, Apple, Spotify and Netflix are violating Article 15 of GDPR, and has launched complaints against eight streaming companies in January 2019.

The confusion over the use of data is further exaggerated by the large number of different companies that access the user's data from a range of devices. Just considering smartphones, the average user has more than 80 apps on their smartphone [Ann18].

Beyond smartphones, users are often unaware of the collection of their data in smart devices. In 2017, smart TV manufacturer, Vizio, was fined $2.2 million for selling customers viewing habits without their consent [Ste17]. A January 2019 article in Business Insider states "There is a simple reason your new smart

TV was so affordable: It is collecting and selling your data" [Gil19].

The expanding experience of digital life is enriching people's lives but is, therefore, also bringing significant challenges related to the control and use of their digital data. The term "digital imprint" is used to convey the digital record of the user's digital activities across all their digital devices. The challenge is to bring some control and transparency over this digital imprint, and, furthermore, to enable the exploitation, under user control, of the comprehensive information in the digital imprint to provide beneficial and novel services for the user. One of the keys to providing useful services that enhance user life and improve the user experience (UX) when using technology is to augment the record of user digital activities with as much context as possible. The context for a user's digital activities include the data from the smartphone sensors, fitness bands, any health monitoring sensors, smart household devices, and any other connected digital devices associated with the user.

1.1 Monitoring a user's digital life

The starting point for recording the digital imprint is the monitoring of user activities across all the digital devices associated with that user. Monitoring user activity has been widely studied and can be useful in several applications such as tracking a user's physical activity using a smartphone and wireless sensors [RR18] or through wearable devices (e.g. ECG sensor) [LCJ$^+$18]. It is also advantageous to monitor a specific group of users such as children [SGD18, NG17, Mar17]. Several monitoring frameworks have been proposed [BG07, RK08, DAS09, GSD14, BCKO15] that make use of user data.

In a user monitoring system, the main tasks are: ***collecting data*** (e.g. user activity, context, etc.) from various data sources (e.g. devices, sensors, etc.); ***storing data*** in a standard format, in a safe place, and accessible so that the data can be exploited later; ***analysing data*** that supports enriching data quality by finding the patterns and extracting insights from the data; ***sharing data*** so that external data consumers can exploit the data. These tasks introduce several challenges such as security, data accessibility, data privacy, and communication costs.

In a monitoring system, which monitors a user's digital life, one of the fundamental purposes is to make use of data and assist the user with a better experience in their digital world.

1.2 Examples of target applications

Monitoring and analysing user activity can contribute to many aspects of life. The following is a selection of some important applications.

1.2.1 User self-monitoring and improving awareness of digital life

Users interact with a variety of digital devices. They use a camera, smartphone, coffee-maker, smart-home systems, etc. An interaction with a digital device that can be digitally recorded is called a digital interaction, a set of which is called digital imprint. Users are usually unaware of their digital imprints. Self-monitoring can help users to regulate the effect of technology through effective awareness management [IHSH06]. Self-monitoring and the insights from data analytics regarding users digital imprint can assist users in improving awareness of their digital lives and having a better understanding of digital activities and their influences.

Use of social networks has become one of the leading daily human activities [AKS14]. Virtual interpersonal communication and interaction in social networks trigger user emotional reactions [KR18]. A concern is that an individual's behaviour may be affected by digital social interactions they are not aware of [ZWT$^+$17]. Capturing and understanding web user behaviour [FČ18] can increase user awareness of these influences. A social interaction monitoring system can make users aware of digital social influences. The solution is a monitoring system that aggregates the individual social interactions and its context to obtain a high-quality awareness for the user.

1.2.2 Assistance for dependent individuals

Dependent individuals (e.g. Children or Special Needs Individuals) require more attention regarding use of technologies. Monitoring their digital activities can be beneficial for them and their guardians/carers. Children raised in a digital media-saturated world have access to a tremendous amount of information and media. Due to this fact, the influence of digital media on all age groups, especially children and young people, becomes a significant concern requiring more careful study [ZK18]. Children need to receive media-rich learning information rather than accessing inappropriate information. Children may suffer from technology addiction. It is of concern to parents how children's behaviour is affected by technologies and the content made available to them by technology. Monitoring is a proposed solution [WSEK17, LEM$^+$17] for addressing theses concerns and enabling parents to become aware of children's activities and the content they visit. This solution helps parents to support their children to stay safe from negative aspects of digital technologies.

Individuals with conditions like autism, ADHD, Asperger's or Rett Syndrome benefit greatly from appropriate technology. Designing software and applications for them, however, is not an easy task when considering that interviewing them may not be easy. There are several activities in this area for Special Needs Children (SNC) with autism such as the game design [MBVS16], the UI design of scheduling activity application [AKE18], the teaching supportive application [AMV$^+$17], and the design of a mobile social application [AAJH16]. A comprehensive solution to help and support design for special needs users is based on monitoring their activities in a natural environment transparently without interrupting them. Such a monitoring system can provide information about their needs and the way they use technologies, and consequently can contribute to better design for them.

1.2.3 Innovative user services and improved UX

Assistive applications and innovative services can help users experience a better quality of life. Learning about users, their habits, and their preferences with regards to the use of technologies can help assistive applications to provide relevant information and services to users. Feeding a historical user digital imprint into a system that uses machine learning algorithms to predict outcomes can make predictions possible and support improving UX. Therefore, it is essential to record what is occurring in the present to build the right future. A system can improve UX by presenting relevant information to users based on the history of their digital activities, interests and current situations.

> "Prediction definitely has the power to drive the future of interaction" [Bal17].

Predicting the user's next possible actions and supporting the user to perform them more efficiently is another way of improving UX. There are many beneficial ways to use user data and assist users [DGP14, NM16] in areas such as recommender systems, adaptive services, and context-aware applications [LL14].

UX-focused design can provide users with a more usable product and better design. In UX research, the user's use of the product and their experiences are the basis for improving UX.

> "Applying a predictive-analytics algorithm to UX design presents users with relevant information" [Bal17].

Analysing the usability of an interactive system/product requires a clear understanding of how users use and interact with it and its UI [CDH14]. A recent challenge for content providers is to provide personalised services for users by monitoring their activities and discovering their interests and preferences. Building rich user profiles is the key to providing personalised recommendation [NSD13, GPM17], news [NM16], advertisements [PRMF16], media [TYB16], and profile-based web search [RJ16].

1.3 Research aims

Users interact with several personal digital devices producing a digital imprint. This work aims to enable users to engage with their digital imprint and to provide them with control over how the data is collected, stored, analysed, and shared.

User interactions and associated context data can be aggregated and collected in a central location. The user activity data for all user devices can provide a more comprehensive view of the user. This work aims to provide data collection services for various software platforms and support various technologies. Data analysis is then the process of extracting meaningful information and insights from the raw data, enriching the data quality, discovering frequent patterns, and identifying the correlations between data. This work also aims to share the user data with data consumers so the data can be exploited mainly for the benefit of the user. This part must be carefully designed so the user is explicitly aware of, and can control, the sharing process.

We propose a comprehensive software solution, Baran [HH14, HH15, HH16c, HH16e], that is engineered as a cloud-based, service-oriented, user monitoring, and data analysis framework (Figure 1.2). Baran offers data collection services that transparently, efficiently, and implicitly record user digital activities and associated context data.

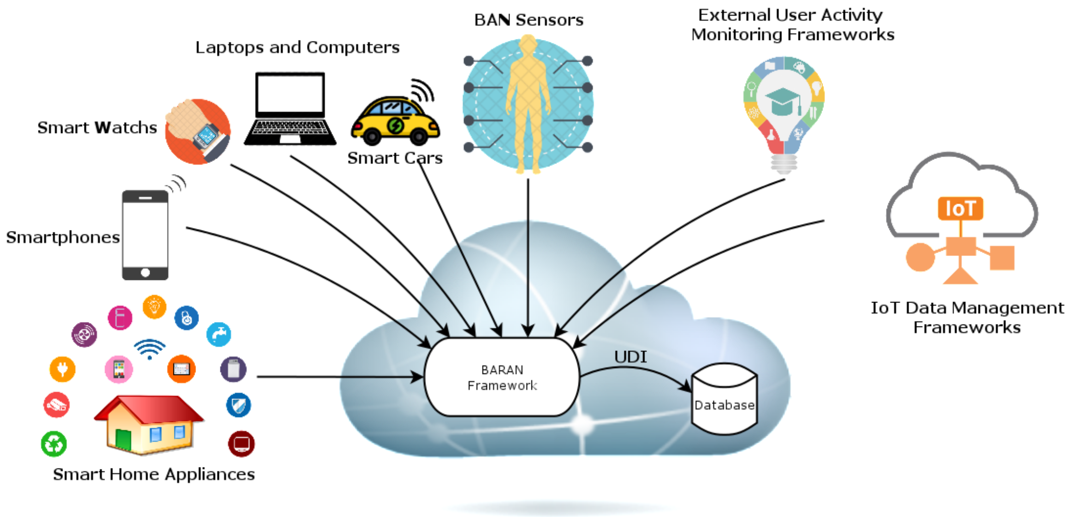

Figure 1.2: Baran overview

The proposed framework enables users to aggregate their digital activities and related context data into a standard data model. Users will have a history of their use of digital devices. This data, later, can be used by various services to support users' needs. Baran also offers data analysis services that extract information and insights from the raw data, and Baran enables other IT systems to use the information and provide better (e.g. personalised) services to users. Baran is constructed as a service-oriented framework, and 3rd party services can be built on the top of it. Baran proposes a comprehensive data model, the User Digital Imprint (UDI), that holds user digital activities and relevant contextual data in a standard data structure. Baran provides full control of data collection and data sharing to the users. The user can control what data and for how long the data is shared with a data consumer (e.g. a 3rd party service). The framework is implemented in the Amazon Web Services (AWS) cloud and also supports IBM BlueMix, OpenStack, Azure, and is not limited to these cloud software providers. Baran provides data collection services for Android-based devices and Windows-based computers. The data collection service allows smartphone sensors to supply context information to the user model. The collected data, then, is securely sent to the Baran framework.

Baran is useful for many data consumers. The user whose data is collected by Baran is one of the primary consumers of the framework. Users can gain insights into their use of technologies and better understand how they are using interactive devices. Application developers are other data consumers of the framework. They can seek to understand how users use their applications and, based on that, improve their products. To reduce the burden on developers, Baran provides essential libraries so that developers can efficiently develop their software on top of the Baran framework. Baran also aims to support researchers who usually perform laboratory-based experiments to monitor users digital activities. Using the proposed framework they can conduct their experiments in a natural environment instead of the laboratory. It also allows them to experiment on a larger scale and with a broader scope. The other group, who can benefit from Baran, is the 3rd party services. They can access a massive amount of user data and expand and improve their services for their users by identifying users' needs and, based on that, provide better services.

1.4 Objectives

This work revolves around four key objectives a) to study the related works, discover their strengths and weaknesses in the area of monitoring a user, and identify their challenges and future opportunities; b) to design and implement a comprehensive user monitoring framework which, considering the identified challenges (e.g. data privacy, security, etc.), offers data collection, storage, analysis, and provision services; c) to propose a standard and comprehensive data model for the collected data (user interaction and associated context data) and to also support future unknown data; d) to evaluate the performance of the proposed framework and conduct case studies to demonstrate use of the framework in different scenarios.

1.5 Research challenges

Over the years, how people perform digital interactions using various interactive technologies has dramatically changed. Users now have their smartphones with them most of the time and use them for over 2 hours and 50 minutes a day[1]. They interact with their personal and work laptops, tablets, etc. They watch smart TVs, listen to smart speakers, cook with smart kitchen appliances, and generally, they interact with several other smart technologies. We take this as the basis of this work and define the following research questions.

> **Research Challenge I:** How can user interactions with their various digital devices be monitored?
>
> **Solution:** Users use many interactive digital devices including smartphone, smart home appliances, and health and fitness trackers; users are surrounded by many IoT sensors. These are the data sources that can be monitored and provide user-related measurements and information. Baran provides data collection services for devices running Android, Windows, and Apple iOS operating systems. Most of these operating systems (OSs) support and share sensor data to 3rd party applications and, except Apple iOS, allow collection of detailed user interaction data. Performance evaluation experiments are conducted to demonstrate that the data collection services work efficiently on devices. Baran also provides services that support the integration of other personal digital devices (e.g. health and fitness bands), IoT devices and other smart household devices.

Considering the user data (digital and associated context data) is collected from various sources (e.g. smartphones, smart watches, TVs, etc.), then it needs to be aggregated and stored. Data analysis components and external data consumers (e.g. 3rd parties) can exploit the data. In order to support them, the framework must meet the requirement of putting the data in a standard data structure that is suitable for organising large and complex data.

> **Research Challenge II:** How can a user model be designed that provides a standard and comprehensive data structure and supports different data levels (data, information, and insights) and various data properties including unknown future data?
>
> **Solution:** We propose a uniform representation scheme for the data in this work. It is an easy to manage, flexible, and scalable data structure. It is designed to support various types of data, including unknown future types of data, from different data sources. The data structure is based on a timeline so the data is timestamped. Multifaceted data can be assigned to any time fraction. The data is served in a multi-dimensional structure

[1] Mobile Matures as the Cross-Platform Era Emerges article: https://goo.gl/Z6j96E

where each data layer can have different levels of data, information, and insights. The proposed data scheme is called the User Digital Imprint (UDI) and is presented in section 3.3.5.

At this stage, the collected data stored in the UDI needs to be analysed. This process enriches the quality of the raw data by extracting information and insights from it. Data analysis produce useful shareable information. The process of sharing is also concerned with data privacy, security, and clarity.

Research Challenge III: What system architecture can support the discussed data analysis, data sharing and provision, and data collection services so that they offer the users full control of data? How can the proposed framework let external parties contribute to its functionalities?

Solution: We introduce a cloud-based framework that offers a service-oriented architecture so several services can independently run on top of it. The framework offers essential services and utilities required for data collection and storage. In addition to those, the framework also provides data analysis services that can enrich the data quality by extracting insights from raw data. The data analysis services leverage the modelling technologies and methods to learn patterns from the data and use machine learning algorithms to produce prediction models. The proposed framework addresses the identified challenges of existing frameworks such as data security, user privacy and the clarity of the sharing process. The framework gives complete control of user data, and any data processing and sharing to the users.

1.6 Contribution

A framework, Baran [HH14, HH15, HH16a, HH16b, HH16c, HH16d, HH16e], is implemented and evaluated. Associated with the proposed framework, seven conference papers (IEEE and ACM Press) and one conference poster have been presented and are listed here.

- UIXSim: A User Interface Experience Analysis Framework in Fifth International Conference on Intelligent Systems, Modelling and Simulation (***IEEE***), Langkawi, Malaysia, January 2014 [HH14].
- Baran: An Interaction-Centred User Monitoring Framework in Interna-

tional Conference on Physiological Computing Systems 2015, Angers, France, February 2015 [HH15].

- A Next Application Prediction Service Using the BaranC Framework in 40th Annual Computer Software and Applications Conference COMPSAC (***IEEE***), Atlanta, Georgia, USA, June 2016 [HH16a].
- A Service-Oriented User Interaction Analysis Framework Supporting Adaptive Applications in 40th Annual Computer Software and Applications Conference COMPSAC (***IEEE***), Atlanta, Georgia, USA, June 2016 [HH16c].
- Child-centred design supported by comprehensive child application use analysis in SIGCHI Conference on Interaction Design and Children IDC (***ACM***), Manchester, United Kingdom, June 2016 [HH16d].
- User interaction monitoring and analysis framework in International Conference on Mobile Software Engineering and Systems (MOBILESoft) (***IEEE/ACM***), Austin, Texas, USA, 2016 [HH16e].
- A pro-active and dynamic prediction assistance using BaranC framework in International Conference on Mobile Software Engineering and Systems (MOBILESoft) (***IEEE/ACM***), Austin, Texas, USA, 2016 [HH16b].

1.7 Thesis structure and outline

In this chapter, an overview of the subject area of the work is given. The background of the study, aims, and objectives along with the research challenges and the proposed solutions, followed by the contribution of the study are presented.

Chapter 2 provides a comprehensive literature review of existing user monitoring and data analysis frameworks. It also presents the terminology and defines the concepts and terms that are used in this thesis.

Chapter 3 describes the system design and conceptual architecture of the Baran framework. This chapter also presents the methodology of the study and discusses how the challenges are approached and what methods are used to fulfil the identified requirements.

Chapter 4 presents the implementation of the framework. In this chapter, how the conceptual ideas from the previous chapter are developed will be explained.

Chapter 5 proposes scenario-based evaluations of Baran and presents the results of these experiments. The experiments are designed to support a user self-monitoring service, user assistance for dependent individuals, and an innovative user service (e.g. adaptive predictive assistant).

Chapter 6 is the discussion and conclusion chapter. In this chapter, a summary of the thesis and the Baran framework, the challenges, and future applications of the work are presented.

Chapter 2

Background

This chapter introduces terminology and definitions, and also provides an overview of related work. A variety of studies that characterise user monitoring and modelling are reviewed.

2.1 Introduction

> *Today, users are the quality meters and their perceptions, expectations, emotions, and needs are critical factors in evaluating their satisfaction.*
>
> [RCC12]

Users live in a digital era, surrounded by digital devices that mostly are connected to the internet. Quality of Experience (QoE) is a multidisciplinary field focusing on overall human requirements. It involves social psychology, cognitive science, economics, and engineering science [RCC12]. Related to QoE, User Experience (UX) is how a user feels while interacting with a product (software and hardware) [KRH$^+$15]. Effective UX starts with a good understanding of users [Tok13]. A deep understanding of users helps technologies to develop successful solutions that satisfy users (see quotation above). Monitoring and gathering information about a user plays a crucial role in understanding a user. Encapsulating this understanding of the user is the basis of user modelling. Data analysis contributes to understanding the user by extracting information and insights from the collected data.

In this section we provide a review of the methods and frameworks that monitor users, collect, store, analyse, and provide the data related to the user's digital activities.

2.2 Digital life

Humans live in an age of connected digital devices. They have digital experiences in their digital lives. Digital devices include smartphones, laptops, smart watches, fitness bands, smart heating controls, smart TVs, smart fridges and smaller smart household appliances such as coffee makers. User interactions with these devices can be recorded in various ways and contribute to better understanding the user. Understanding a user is the key to several challenges. Furthermore, the key to understanding a user is what they do.

Users perform actions. An action is an intentional, purposeful, conscious and subjectively meaningful activity causing wilful human bodily movements of a more or less complex kind [Blo05]. There is a distinction between physical and digital activities. Physical activity is caused by physical action, which has a physical effect in the real world, e.g. eating, walking, and watching a movie. In contrast, digital activity is initialised by physical action, but its primary purpose is intended to have a digital effect, e.g. composing an email, turning a TV on, and playing a game on a smartphone. This study focuses on user digital activities while it can also be used to monitor user physical activities.

2.3 User modelling

User modelling is the process of integrating a user's information and constructing a user model[2]. It can be used in reasoning about user needs, preferences or future behaviour. A consequence of any "smart" device behaviour is that the "smart" agents can guess incorrectly and perform changes that a user does not like. More comprehensive user modelling can better reflect the user's needs, preferences, and situation, so the "smart" behaviour will be more accurate in solving a problem [Fis01].

[2]User modelling on Science Direct: https://goo.gl/sdm3jB

2.3.1 User model

A user model, also known as user profile, persona or archetype, is an explicit representation of an individual's properties including preferences, needs, and characteristics (e.g. behavioural, cognitive, and physical). The user model is the basis for applications such as adaptive, expert, recommender, and user-simulation systems [OFL10]. A user model may contain personal information, interests, skills and knowledge, goals and plans, preferences, dislikes, and data about behaviours and interactions with a system [BR10].

2.3.1.1 Static user model

A static user model is a basic form of user modelling. It holds the original gathered data without updating with more user data.

2.3.1.2 Dynamic user model

A dynamic user model can dynamically change over time. It supports updating the model and provides an up-to-date representation of the user including changes in their interests, their learning progress or interactions with the system. The goal of this model is to take the user's current situation and needs into account.

2.3.1.3 Stereotype user model

A stereotype user model takes advantage of classifying users into common stereotypes. Each stereotype model contains information that is true for users to whom the stereotype applies. It usually relies on statistical data.

2.3.1.4 Highly adaptive user model

A highly adaptive model needs much information (including dynamic data) about a user in order to provide an adaptive solution. It represents one particular user and allows a high adaptivity of the system for that user. [KKKP08] proposed an adaptive user model that adapts to an individual's circumstances based on data from wearable sensors.

2.3.1.5 Predictive model

A predictive model is formed of a number of predictors, which are variable factors that are likely to influence future behaviour. In predictive modelling, data is collected for the relevant predictors; a statistical model is formulated, and predictions are delivered; the model will be then validated (or revised) as additional data becomes available.

2.3.1.6 Task model

A task model describes the tasks an end user performs. It is helpful in prescribing what interaction capabilities should be designed. This model is built by analysing user interactions with a system.

2.3.1.7 Interaction model

An interaction model represents the interactions between a user and a system. It also describes how objects within an interactive environment interact. It helps to understand and explain how users move from object to action within a system. Users internalise an interaction model but do not articulate it. They internalise well-constructed interaction models, which make most actions within a system more predictable. Defining an interaction model is a foundational requirement for a digital system [Nie12].

2.3.1.8 Context model

A context model represents the contextual information of a ubiquitous environment. It is an element of a system that provides the behavioural description of the surrounding environment. An ontology-based context model has been proposed to address critical issues including formal context representation, knowledge sharing and logic based context reasoning [WDTP04].

2.3.2 User modelling history

One of the earliest successful examples of user modelling was the WEST system [BB79]. It is a coaching system for a game called, *How the West was Won*.

The system uses the user's moves in the game (*Chutes and Ladders*) and learns what strategy they use most. The system then uses the learned patterns and provides users with optimal moves. The authors introduced the concepts of *Shared Context*, *Initiative and Intrusiveness*, and *Relevance* in user modelling [BB79].

Olmedilla et al. [OFL10] analysed navigation logs of a large set of mobile users in a developed country. They categorised the navigation logs and modelled a user's internet surfing behaviour on mobile devices. An ontology-based unified user context model was proposed in order to support cross-system personalisation. They addressed an essential problem in user modelling, which is the restriction on using a user model in a single system. They explored how to reuse a user model in other systems in a unified and extensible way [CNS+04].

We are also interested in addressing the challenges identified by previous studies in user modelling. Shared context, relevance, and extensibility in user modelling are essential elements of our user modelling.

2.4 Data collection and storage

People use several digital products every day. They interact with a digital product using some kind of user interface (UI). Nowadays, these UIs are often highly interactive and complex. Designing a sophisticated and interactive UI is a complicated process for designers. Monitoring how users interact with the UI and analysing interactions can help to measure the effects the design has on users and can also assist the process of designing a better user interface. User monitoring is a general research area about how to monitor users and their activities (e.g. physical and mental). For instance, Real User Monitoring (RUM) is a passive technique of recording user interactions with a product in a real environment. This method can be used for testing a new design or an updated version of a product before and after deployment. Monitoring the end-user provides extra information that helps in the design of a product. Collecting data is the main element in monitoring a user's activity. Data can be collected from different sources. There are several works providing methods and techniques for user data collection and we discuss some of them in this section.

While monitoring a user, there is a considerable amount of generated data that needs to be collected and purposefully exploited. Storing data is a critical part of a monitoring framework. Frameworks need to use a public open standard data storage method, so the stored data can be used by data consumers (e.g. users, developers, and 3rd parties). The data storage must be secure as it holds sensitive information. Frameworks also need to provide APIs to securely transfer data using techniques such as data encryption. We will discuss how current frameworks store data and what challenges they have faced.

2.4.1 Existing frameworks of data collection and storage

Regardless of the type of user activity, services are required to sense and collect user data. In this section, we discuss the frameworks and tools that provide data collection and storage.

2.4.1.1 MyLifeBits

The authors in [BG07] recorded a digital version of physical activities in a system called MyLifeBits. It is designed to store and manage a lifetime's worth of data. Human memory is fallible, and because of that, researchers have been looking for a system that automatically records documents, images, communications, and video, storing them in a reusable archive. It is becoming easier to record activities digitally as cheap, and reliable, sensors and data storage become available. The authors challenge software designers to try to organise the information to be accessible. They list the benefits of having a digital memory in medical care, job productivity, and other areas, but challenge developers to ensure that the archives are secure. The authors believe that a digital memory of user activities is beneficial, so they presented an example to show it:

> A stockbroker, Dave Digital, archives all documents, emails, phone calls, and web site visits during the workday. As Dave composes an email, his time management program warns that he is spending too much time on communicating with an unimportant customer. He forgoes polishing the message and sends it off as is, then goes to work on a higher-priority account. Dave also reviews recent records of his weight, heart rate, and caloric input to determine the adverse effects of two days of stressful corporate meetings [BG07].

2.4.1.2 Hermes

Hermes [DAS09, BLG$^+$12] is a context-aware application development framework and toolkit for the mobile environment. It is a middleware that collects and provides context data to applications. Hermes provides two parts at its highest level: the framework and the toolkit that provides a collection of classes for developers to utilise in their application for collecting context data. The Hermes framework provides applications with resource discovery, communication, storage, security, and power management services in order to share data in the mobile environment. The current version of its implementation is for Android and Linux/Windows-based devices. Hermes data collection does not include user interaction. It also does not seem to provide a user with access to the collected data and to consider the privacy of the users. A Hermes software application needs to have the toolkit library included in its code, and this makes the software application more heavyweight. This framework suggests a cloud infrastructure in order to share data from device-to-device and user-to-user; however, it does not seem to be implemented.

2.4.1.3 Lifelogging

Lifelogging [GSD14] is a user monitoring framework, in which user activities (mostly physical activities) are tracked using wearable sensors and devices. This framework attempts to capture the life details of moments, events, and activities. The system provides the basis for monitoring and storing the data. The framework requires a user to wear sensor devices. One of them is a device called SenseCam, a small wearable device that has a digital camera and multiple sensors including accelerometer, thermometer, infrared, and light sensor. This device can generate 5000 images per day. The image data is unstructured data and needs to be processed by feature detection algorithms. Video indexing, object detection, and event classification algorithms can be used for processing the collected images. The framework digitally records the individual's daily life and activities in order to better understand the user. Using this framework, researchers proposed some ideas such as identifying people with dementia [PS14] by recording and analysing user activities. The privacy issue of Lifelogging is studied in [GAJI14], and also a survey on data collection methods is presented in [ZG12].

2.4.1.4 Contory

The CONTextfactORY (Contory) is middleware for context provisioning on smartphones [Riv06, RK08]. It provides three data collection strategies for context provisioning of data that are internal (local to the device), external (gathered from a server), and distributed (gathered from ad-hoc network elements connected via Bluetooth or WiFi) [WW10]. It defines the collected elements of the context to include source, value, time-stamp, type, and lifetime plus an extra field for other information [FL13]. It offers the ability to send a context query from another application to itself using a Structured Query Language like (SQL-like) interface, which supports filtering the contextual elements.

2.4.1.5 SCM

Service Context Manager (SCM) [DM09] is a framework that provides context gathering, processing, inferring, and reasoning. Context Broker Architecture (CoBrA) [CFJ03] is proposed as a context-aware agent-based system for smart environments. It provides a context broker as an intelligent agent to maintain models and share them.

2.4.1.6 Frappe

Frappe [BCKO15] is proposed as a context-aware mobile application recommender system. It records how frequently a user uses an application. It collects context data such as location, time, and the application category. It processes the collected data and recommends to a user what application they may likely want to use based on the current context. It seems that the work is designed specifically for mobile applications.

2.4.1.7 MoodScope

MoodScope [LLLZ13] is proposed to collect and analyse users data in order to infer the user's mood based on smartphone usage and to react based on the current situation.

2.4.1.8 MobileMiner

MobileMiner is a general-purpose middleware that mines user behavioural patterns and discovers co-occurrence patterns on the phone while a processor is not busy [SMM+14]. One of the main contributions of this work is that their method outperforms the majority of predictors which use frequency counts. They have introduced a much more comprehensive prediction model by considering more factors as inputs. One interesting future challenge mentioned in this study is to find common patterns across multiple users. It helps to improve user experience in a generic style and also find common behaviour patterns for a specific group of people doing the same tasks or having similar characteristics (e.g. age and gender).

2.4.1.9 MSM

The Mobile Sequence Mining (MSM) engine aims to learn smartphone usage, discover sequential patterns, and use the learned patterns to enable a variety of applications including proactive assistance for a variety of use cases [MSW14]. It needs no internet connection, as it mines and stores data on-device, and uses this technique to preserve privacy and security. The engine is designed to provide sequential patterns and predictions over multiple data streams, and also allow individual mobile applications to stream their private data to mine sequential patterns. The authors tested MSM (Android version) with some users and mined 7 to 53 days of user context data (location, app usage, and call logs) and provided some useful results and challenges. Future work was to include multi-modal context sequence, sequence-based prediction, and automatic parameter selection.

2.4.2 What data is collected?

We have reviewed several existing data collection frameworks and toolkits. Their functionalities and challenges are discussed. They are designed to collect user physical activities as the primary data collection element along with relevant user context data. In this section, we define what the context is and review what context data they collect from a user.

A context-aware system is a piece of software that senses and reacts to the context. It requires having context information in order to provide a better service to a user. "*Context is a vast concept that encompasses all possible parameters identifying a situation*" *[MMJ07]*. In other words, context is the surroundings, setting, situation, and state [Pas98, RCDD98, HNBR97]. One of the most widely adopted definitions of context is proposed by Dey and Abowd [DFAB04] as follows:

> *Context is any information that can be used to characterise the situation of entities (i.e., whether a person, place, or object) that are considered relevant to the interaction between a user and an application, including the user and the application themselves.* [DFAB04]

[DFAB04, WZL05] categorise the context into location, time, activity, and identity (personal). Context data, application usage, phone logs, and location were collected [MSW14] in order to predict future activities. Call and SMS events, inferred place, phone charging status, battery level, and WiFi or cell connectivity information are collected in [SMM$^+$14] as the input of a predictive model. Based on the classification of [Dou04], the context has two main types, representational (e.g. time and location) and interactional (e.g. clicks and usage).

In summary, action, interaction, response, and event (interactional data) are the primary elements in data collection and time, location, behaviour, and environment status (representational data) are examples of the relevant context data.

2.4.2.1 Identity or personal context

This refers to any personal information of a user such as name, age, gender, height, relationship or contact with colleagues or friends. Personal information can vary depending on the scope of the context.

2.4.2.2 Location context

Location context refers to the position and proximity about users, devices, and the environment. GPS[3] is a space-based system providing location information. The three dimensions in space are latitude, longitude, and altitude. Cellular network infrastructures such as GSM[4] can provide this information. As GPS does not work well indoors, a new area of indoors localisation has been introduced which uses Wifi signals [PW08] or properly placed, embedded motion sensors [NAY18, ANS15] in order to locate a user or a device. Predicting a user's future location is one of the examples of location-awareness applications [NCW+12].

2.4.2.3 Time context

Time context is essential in solving many business use cases[5]. The time context should tell us different types of time information, such as the time zone, actual time, virtual time, etc. [WZL05].

2.4.2.4 Activity context

Activity refers to what a user performs to achieve a possible result. In general, an activity can be classified into physical and digital activities.

2.4.2.4.1 Physical activity Physical activity is caused by any bodily movement performed by skeletal muscles that require energy expenditure. Physical activity has a physical effect in the real world, e.g. eating, walking, and watching a movie.

2.4.2.4.2 Digital activity Digital activity is also caused by physical action, but its primary purpose is intended to have a digital effect, e.g. composing an

[3] GPS stands for Global Positioning System
[4] GSM stands for Global System for Mobile Communications
[5] Outline in solving business cases article: https://goo.gl/JzTQHL

email, turning a TV on, and playing a game on a smartphone. UI interaction is an excellent example of digital activity.

2.4.3 What are the sources of data?

The discussed context data is sensed through a sensor, which is an object whose purpose is to detect and report changes in its environment. The sensed information can be used in order to understand the context of an event and accordingly act based on the recognised context. There are various kinds of sensor and a number of them are reviewed in this section.

2.4.3.1 Sensors in digital devices

Nowadays, many digital devices (e.g. smartwatch, smartphone, and car) are equipped with several different sensors. Sensors can be classified into two groups, either passive or active sensors. Active sensors require an external power supply to operate and produce the output signal. Passive sensors are self-generating devices because their properties change in response to an external effect. They can produce output voltage and signal amplification. There are various kinds of sensors. In this section, the sensors used in the reviewed studies are discussed.

2.4.3.1.1 Accelerometer sensor An accelerometer is a device that measures proper acceleration. The proper acceleration is a relative measurement to free-fall. The three axes for acceleration are "x", "y", and "z" (Figure 2.1).

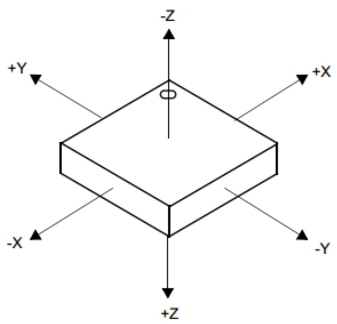

Figure 2.1: Accelerometer sensor axis

2.4.3.1.2 Gyroscope sensor A Gyroscope sensor is a device primarily used for navigation and measurement of angular velocity. Gyroscopes have evolved from mechanical-inertial spinning devices consisting of rotors, axles, and gimbals to various incarnations of electronic and optical devices, each of which exploits some physical property of the system allowing it to detect rotational velocity about some axis (Figure 2.2). Three-axis gyroscopes are often implemented along with a 3-axis accelerometer to provide a full six degree-of-freedom (DoF) motion tracking system.

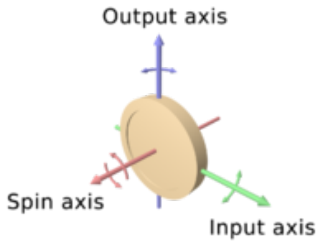

Figure 2.2: Gyroscope sensor axis

2.4.3.1.3 Orientation sensor In geometry, the orientation, angular position, or attitude of an object is part of the description of how it is placed in the space. The orientation sensor measures three-axis orientation (Figure 2.3) of an object in space.

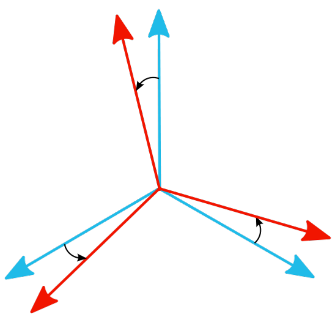

Figure 2.3: Orientation sensor axis

2.4.3.1.4 Gravity sensor A gravity sensor measures the acceleration effect of the earth's gravity on the device enclosing the sensor along three axes (Figure 2.4). It is typically derived from the accelerometer, where other sensors (e.g. the magnetometer and the gyroscope) help to remove linear acceleration from the data.

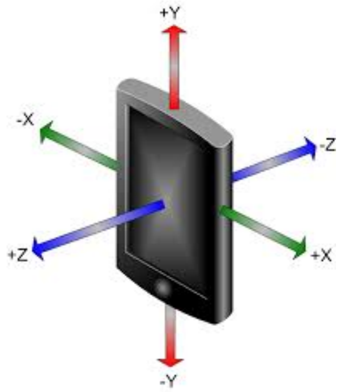

Figure 2.4: Gravity sensor axis

2.4.3.1.5 Pressure sensor A pressure sensor measures the barometric air pressure. Atmospheric pressure, sometimes also called barometric pressure, is the pressure exerted by the weight of air in the atmosphere.

2.4.3.1.6 Humidity sensor Humidity is the amount of water vapour in the air. The humidity sensor measures it in three ways: absolute, relative, and specific.

2.4.3.1.7 Magnetic-field sensor A magnetometer is a measurement instrument used for two general purposes: measuring the magnetisation of a magnetic material like a ferromagnet, and measuring the strength and, in some cases, the direction, of the magnetic field at a point in space.

2.4.3.1.8 Light sensor The light sensor measures the brightness of the ambient light level. It is useful for reducing power consumption using display brightness adjustment in digital devices.

2.4.3.2 Body Area Network (BAN)

A body area network (BAN) is a network of wearable sensor devices placed on the human body that produce, store, and transmit the data through the network. It is also known as a wireless body area network (WBAN) or body sensor network (BSN). A BAN represents the natural union between miniaturisation and connectivity and enables more comprehensive monitoring of a user.

One of the most critical BAN applications is healthcare. A health carer can monitor and better understand a patient. The sensors provide a user's physiological and even biological information that can be processed and used to diagnose precisely [FCFRP12]. For instance, [SKC$^+$12] presents an activity-aware mental stress study, that used electrocardiogram (ECG), galvanic skin response (GSR), and accelerometer data, to detect stress while users sit, stand, and walk, claiming 92.4% accuracy for mental stress classification. Miller et al. constructed a framework for optimally sensing a patient's health state with a wireless body area network (WBAN) [MZBBG18].

In this section, we discuss some of the measurements used in body area networks.

2.4.3.2.1 Electrocardiogram (ECG) Electrocardiogram (ECG) measures the electrical activity and condition of the heart. An ECG can be used to measure the rate and rhythm of heartbeats, the size, and position of the heart chambers, the presence of any damage to the heart's muscle cells or conduction system, and the effects of cardiac drugs. Several devices in the market provide the ability to record ECG or heart rate. New smartwatches are aften equipped with sensors to detect heart rate. [FOFH15] attempted to use ECG as the input of a video game.

2.4.3.2.2 Electroencephalogram (EEG) Electroencephalogram (EEG) monitors the electrical activity in the brain. The EEG signal is composed of multiple individual oscillations at five major frequency bands (delta, theta, alpha, beta, and gamma) [MGWM13]. The EEG can be used to detect a user's emotion such as anger, happiness, frustration, etc. from the user's facial expressions as emotion-specific muscle movement can be detected from brain waves [APSV08].

2.4.3.2.3 Electromyograph (EMG) Electromyography (EMG) measures the electrical activity in the muscles, where the motor neurons transmit electrical signals that cause muscles to contract. An EMG translates these signals into graphs, sounds or numerical values that a specialist interprets.

2.4.3.2.4 Electrodermal Response (EDR) Electrodermal Response (EDR), also known as Galvanic Skin Response (GSR), measures variations in the electrical characteristics of the skin. The skin resistance varies with the state of sweat glands in the skin.

2.4.4 Where is the data stored?

Data collection services in existing frameworks are reviewed below. With over 175 zettabytes of data expected by 2025[6] in digital health, AI, Fintech, etc., storing data is becoming a challenging task. Generally, the data can be stored in two different ways: on the device that generates it, or outside of the device, e.g. network-attached storage (NAS) or cloud-based storage. These both introduce their own challenges and complications.

2.4.5 Data collection and storage challenges

How, when, and where to store the data are challenges that need to be tackled. When the data is big, then the local storage of the device cannot handle it. However, it provides better privacy guarantees and reduces dependency on network connectivity [SMM+14]. Storing data is not the only challenge, as the collected data needs to be processed. So, if the data is in the local storage of the device, the processing has to be performed on the device. Furthermore, the hardware constraints of current devices introduces another challenge of not exceeding the device resources. The MobileMiner framework [SMM+14] uses the device local storage for storing the collected data. [GBL06, GLB03] use an external hard drive to store the collected data. The Lifelogging framework [GSD14, DCC+11] uses the local storage of the device for data collection. The collected data is extracted from the device for processing. Mobile Sequence Mining (MSM) [MSW14] provides on-device mining and stores data on the device.

Cloud-based and network-attached storage are good choices for storing big data, as there is no limitation on the amount of storage. They are also good options when data accessibility is a concern. The Frappé framework [BCKO15] sends the collected data to a cloud-based server.

[6] The future of Data Centres on CB Insights: https://goo.gl/JzJt2A

2.4.5.1 Security challenge

Security is a challenge in any system that collects digital data. Data can be stored in an encrypted form. Encryption is a widely used technique to secure a system. It is a process of encoding information using a key. The primary challenge is to keep the key safe; if it is not then the encryption is compromised [CSR+16]. There are two ways to encrypt data using a key. There is a public key that everybody can use to encrypt the data; however, only one party (e.g. receiver) has the key to decrypt the encrypted data. Secondly, there is a symmetric key that both sender and receiver must have in order to encrypt and decrypt the data.

The collected data may include sensitive information such as personal, medical, or personally identifiable information [CSR+16]. Personally identifiable information (PII) or sensitive personal information (SPI) is information that can be used to identify and locate a person. In the literature, confidentiality, integrity, and availability (CIA) are the three fundamental properties of security that need to be addressed in order to have a secure system [CSR+16].

2.4.5.1.1 Confidentiality No adversary should be able to access information about an individual without permission. It is also essential to learn how, when, and who accesses an individual's information. Having different levels of confidentiality is possible. For instance, basic information which cannot be used to identify an individual is less confidential than the information that can be used to identify an individual.

2.4.5.1.2 Integrity No adversary should be able to compromise a system and falsify information. The system must make sure that the data is genuine and guarantee that no false information is collected. The system also must be able to detect and mitigate discrepancies due to false information.

2.4.5.1.3 Availability Data, and the ability to collect data, should not be unavailable for a reason such as Internet disconnectivity. Availability is one of the concerns of data collection [CSR+16]. It refers to a requirement of having data available and accessible where and when it is required. For instance, data should be accessible when connectivity is not possible or the device is lost/stolen. Denial-of-service attackers may be able to prevent the availability of a service or the collected data. The system must be robust enough to secure the system against them.

2.4.5.2 Data size challenge

The collected data is large and requires sufficient memory, and memory that also provides fast retrieval mechanisms [MMJ07]. The size of data depends on the data collection frequency, the number of devices per user, and the type of data (e.g. String or Integer). Storing a massive amount of data in a local device is not possible due to the memory limitation. It is also costly to transfer data to a server. Data compression is a well known and widely used technique to reduce the cost of data storage and transmission. When the collected data is large, it may contain redundant data. Data compression is a helpful technique that reduces data redundancy and lessens the overall overhead. Data can also be downscaled (e.g. summary / analysed data is used instead of full resolution raw data) [HWCL14].

2.4.5.3 Outdated, unused, and rarely-used data

There can be some old data. That does not need to be stored, and that must be purged. One of the challenges is to discover the frequency of using the data and, based on that, to decide how to reduce the cost of storing data. For instance, in cloud-based data storage services, there are several kinds of storage provided by cloud service providers. They can differ as regards cost, bandwidth I/O, and hardware technology (this affects data retrieval speed). If the outdated, unused, or rarely-used data can be stored in less expensive data storage, this could help to reduce the costs for both users and the services [MMJ07]. Some other techniques could be used such as storing summaries and analysed data in data storage with a fast retrieval mechanism, and storing the raw data in slower, lower cost storage.

2.4.5.4 Data extensibility

Data collection is an incremental process. The user data being accumulated is connected and should not be split. So, the data structure should be extensible (supporting more data fields) and scalable (supporting more data items). Uniform data representation is required to support aggregating raw data, analysed data, and other high-level data and to support unknown future data fields.

2.4.5.5 Communication costs

Sending data to external storage introduces a communication cost issue. The challenge is how to reduce the communication cost using techniques and technologies to save energy and resources. For instance, data compression is a technique to reduce the size of the data, and that also reduces the cost of CPU, memory, network bandwidth, and communication.

2.4.6 Discussion

Several frameworks and tools, providing data collection and storage, are reviewed. The data collected and its source (e.g. sensors), and where and how it is stored, in the existing frameworks, have been studied. The challenges that the existing frameworks have faced are also explained.

Unlike existing cloud-based intelligent services (e.g. GoogleNow) that rely on Internet access and may compromise privacy, MSM provides device intelligence by leveraging mined patterns while preserving privacy via on-device mining [MSW14]. However, storing the data in local storage introduces more challenges. Confidentiality is one of the challenges that is often not addressed with care. Monitoring a user is important, but it has to comply with user privacy. A user has the right to know how, when, and who accesses their data. Confidentiality is required against malicious attacks when sensitive information is being transferred [MMJ07].

A user also has the right to control how their data is used, so when they want to, they should be able to prevent or limit the sharing process. There must be some techniques to secure the data shareability [MMJ07]. The reviewed frameworks do not seem to enable users to control the data collection and sharing processes. They also do not seem to inform the users about how their data is used.

Data availability is an identified challenge (e.g. [DAS09]) and in some of the discussed frameworks data availability is forfeited. Data extensibility is an issue that is partially supported [Riv06, BLG+12, DAS09]; this allows the data structure to grow and lets external parties contribute to enriching data.

Most of the works discussed do not seem to consider security as essential in data collection and provision. Some state that by storing data in local storage the security is guaranteed. Collecting precise data about user interaction with interactive devices can be beneficial in understanding user activities. The current frameworks do not seem to include user interaction as a part of their data collection methods. They also do not seem to address data confidentiality, integrity, availability, and extensibility challenges.

2.5 Data analysis

An intelligent environment can support users by learning their habits. Data analysis is the key to learning about users and their environment [AI09]. Users are engaged in more digital environments every day and generate more data. Raw data is just a set of numbers, and they mean nothing unless they are processed. Analysing raw data along with contextual data is vital, but time-consuming. Gathering context data is the first step, the next step is to look for contextual insights and, as a result, better understand a user and their behaviour [Swe14]. The next revolution in user experience is the use of smart devices that can adapt to their user's lifestyles and surroundings to become a proactive personal assistant [MSW14]. Data analysis of user information and context provide the basis for these adaptive services.

The foundation of a user-centred services is to know how a user uses a product, which is a difficult task. It requires a monitoring system to collect information about the user and their activities. The available context data at the time of using a product can contribute to the information about the user. Recording a user's behaviour is the key to constructing a comprehensive user model. While the raw data is the foundation of a model, recording it alone is not sufficient for building a rich model [KHW06]. The raw data needs to be processed and transformed into a higher level of information, knowledge, and insights. Systems need to have a rich user behavioural model in order to provide an adaptive and personalised service to them.

2.5.1 Existing data analysis frameworks

In this section, we briefly discuss some of the existing frameworks and explain how they analyse data.

2.5.1.1 Sequential Patterns of User Behaviour (SPUBS)

Sequential Patterns of User Behaviour (SPUBS) is a system that discovers a user's habits and common behaviour [AIB+10]. The authors divided their algorithm into four steps in order to mine the patterns. A set of actions that frequently occur will be identified first. The second step is to discover a topology for the identified actions. In this step, they give the actions a sequential structure that clarifies the order of actions. Thirdly, they try to relate an action to the actual time that it occurred and its duration. In the fourth step, the conditions will be discovered to contextualise the sequence of actions. Figure 2.5 provides an overview of SPUBS architecture.

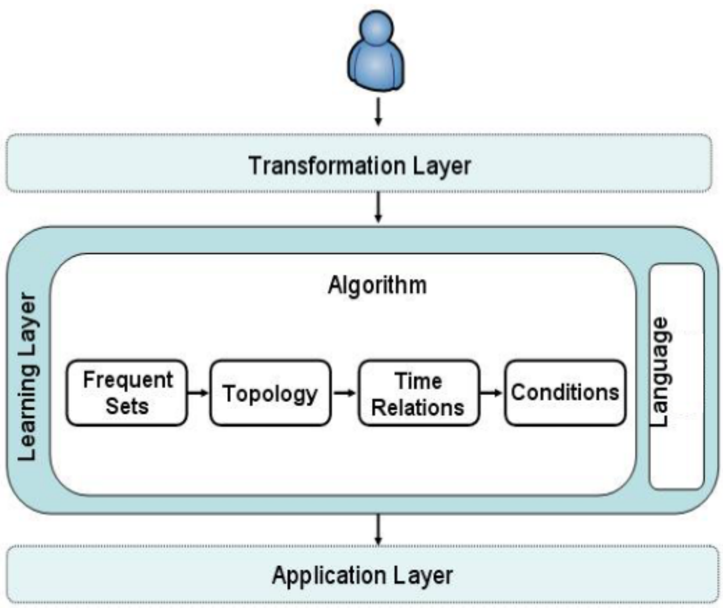

Figure 2.5: Global architecture of SPUBS

Using a real-world scenario, the authors discuss how their system can analyse user data in order to ease their life. They declare that the decision tree and rule induction could be used to identify abnormal situations. They also discussed the use of fuzzy rules in order to find a solution for an identified problem that an intelligent system can handle on behalf of a user. Case-Based Reasoning (CBR) is the method they used to learn user's preferences [AI09].

2.5.1.2 Sporting activities identification

The Support Vector Machine (SVM-based) approach, K-Nearest Neighbour (K-NN), and Bayesian Network (BN) were used to identify sports activities [MMO13]. The authors used the extracted data from the available sensors of the user smartphone. The average maximum F-measure accuracy of 87% was achieved using a fusion of classifiers, which was 6% better than a single classifier model and 23% better than a standard SVM approach [MMO13].

2.5.1.3 Mental stress and mood identification

A continuous monitoring system is used to identify user activities and stress [MZBBG18]. This study collects Electrocardiogram (ECG), Galvanic Skin Response (GSR), and accelerometer data from 20 participants. The participants were asked to do three activities: sitting, standing, and walking. The authors classify the stress of the user using some machine learning algorithms: Decision Tree, Bayesian Network, and Support Vector Machine. They used a framework, called Weka, that provides several machine learning algorithms. They divided the training data into two sets. One set includes ECG and GSR data, and another set includes ECG, GSR, and accelerometer information. Having these two training sets can influence the results of inference. Their results show that having accelerometer data is necessary to improve mental stress detection in a mobile environment. They also conclude that GSR features are relatively dependent on three physical activities (sitting, standing, and walking). They declare that physiological signals tend to be user-dependent [SKC$^+$12].

A software system, MoodScope [LLLZ13], is proposed to infer the user's mood based on smartphone usage. They collect the user data, analyse it, and react accordingly based on the current situation.

2.5.1.4 User behaviour and interaction analysis frameworks

The authors of [NSF10] proposed a statistical model in the area of adaptive user interfaces. It models user behaviour and infers information from the user's interaction. They improved the Karatsuba algorithm (KO*/19) [NSF10], and used it to recognise user activities from behavioural patterns. This algorithm finds reoccurring sequences of actions and defines an activity as a set of actions.

It then calculates the probabilities and predicts the next action based on the current actions. They evaluated their algorithm by recording five different user activities using software called PKITool. They asked 26 skilled students to do 12 different tasks. They compared their algorithm prediction accuracy with a Markov Chain algorithm and discovered a mean prediction probability (MPP) of 45.23% that is 12.12% higher than Markov Chain MPP.

2.5.1.5 Contextual-based recommender systems

A recommender system (RS) typically produces a list of recommendations. It presents the recommendations either based on the contents that the user has been interested in or based on a user personality profile. For instance, a rich user profile is used in recommending personalised news [NM16] and personalised movie recommendations [TYB16] to the user. There are several works proposed in this area, but most approaches do not consider the contextual information about the user [AT08].

2.5.1.5.1 Context-aware music recommender systems Users listen to music every day. Choosing what song to listen to can be difficult when the number of choices is high. The authors proposed a framework that considers the context of the user while also considering the music genre, artist, etc. [HMB12]. They considered the sequence of the music a user chooses to listen to, as well as the tags they search for. One of the main challenges of these works is to collect contextual information about their users to enable them to personalise the predictions and recommendations. For instance, [PY06] uses the time that a user listens to particular music or genre, the weather, and the temperature. Other work uses the location to provide a suitable recommendation list to a user [KR11].

2.5.1.5.2 Context-aware application recommender systems The exponential growth in the number of mobile applications drives users to install many applications on their smart devices. Apple reported having about one million apps released[7] in 2013 and it has reached two million in 2018[8]. The number of

[7] Apple announces 1 million apps in the App Store: http://goo.gl/yMySLB

[8] Number of apps available in leading app stores as of 3rd quarter 2018 article on Statista website: https://goo.gl/n9eqJT

app downloads had reached 100 billion in 2013[9]. Google play has 3.8 million apps in the market in 3rd quarter 2018[8].

Context-aware recommender services [KBC+12, BCKO15] have been proposed to make it easier to find an application based on a specific location or time. These frameworks do not seem to focus on recommendations based on user habits. For instance, a recommender system (RS) [BCKO15] is proposed to recommend an application to a user based on the context for buying/installing. The recommendation seems to be independent of the user profile and habits.

Context-aware recommender systems (CARS) [KBC+12] consider context factors (e.g. time and location) in recommending an application to a user. It has been shown that contextual factors strongly influence the recommendations [BLPR11]. The aim of a hybrid RS [WSW07] is to recommend what application to install based on what other users installed for the same context. Nagarajan et al. [NSD13] present an algorithm, iConRank, that ranks applications based on the recent app sequences. They believe that the sequence of recent applications is related to the next application. For instance, if a user uses a "Contact" application, he/she is likely to use a "Message" or "Email" application next. The authors consider the interactional context as the foundation of their algorithm [NSD13]. Most of the current works [DM11, GM10] seem to focus on recommending an application to install based on the context and only consider representational context such as location, time, etc.

2.5.2 What are the challenges?

Several works have been reported that provide different data analysis techniques and mechanisms. Once data is collected, namely raw data, it contains a considerable amount of meaningful information. Data analysis is a process that serves two purposes. The first purpose is to discover patterns within the data that might not be at all obvious from looking at the raw data. The second purpose is to summarise it in a way that it is more understandable and meaningful. Statistical analysis is one of the methods widely used in this area. It applies statistical and mathematical algorithms and functions to data. This type of data analysis is basic and usually not used for comprehensive analysis. On the other hand, qualitative analysis is a more complicated process with a broader scope of

[9]Cumulative number of apps downloaded from the Apple App Store from July 2008 to June 2017 article on Statista website: http://goo.gl/rTNE80

input data and provides a deeper and more comprehensive analysis of raw data. It, however, is a resource-heavy and often time-consuming process [Jou12].

There are various challenges in the area of data analysis. Data analysis is a resource-heavy and time-consuming process especially when the size of the data is significant. One challenge is how to analyse the data efficiently. Data analysis services generate data derived from raw data. Another challenge is the accuracy of the data generated by data analysis services. There should be a quality check on the accuracy and reliability of the analysed data.

2.6 Data provision

Data provision involves the processes of preparing and providing the data to data consumers. Data consumers include the users who are generating data, any 3rd parties that use the user data for some purpose (e.g. providing personalised services to the user), and researchers who need to study the users. In a monitoring system, the data collection provides the raw data. The data analysis will be used to enrich the raw data with information, knowledge, and insights. The mission is not complete if the analysed data is not used. Data provision is the process of providing and sharing data with parties that need user data. In this section, we report on the techniques and challenges reported in related works.

2.6.1 Existing frameworks of data provision

An overview is given of several works that provide data collection and analysis. The collected and analysed data will be useful for some other parties such as researchers, designers, developers, and the users themselves. One of the challenges that the reported frameworks have faced is how to efficiently and securely share the data with other parties.

2.6.1.1 Hermes

The authors of the Hermes framework provide a loosely coupled component-based architecture that facilitates the decomposition of context-aware applications into multiple smaller components [BLG+12]. It is a widget-based framework. Each widget is responsible for a piece of context data (e.g. location, time, and sensor data) or for doing a specific task (e.g. connecting an application to another widget). This architecture allows context data to be transmitted securely between widgets. A developer needs to develop software using the toolkit provided by Hermes [BLG+12] and configure widgets for an appropriate purpose. A part of their work is a resource discovery mechanism that provides the ability to share widget functionality efficiently. For instance, if an application is using a GPS widget, and there is another application requiring the same piece of context information, it can request the first application to share the GPS widget instead of bringing up a second GPS widget.

2.6.1.2 Contory

Contory [RK08] is middleware that introduces multiple provisioning strategies in order to provide efficient context provisioning on mobile devices. Internal sensors-based, external infrastructure-based and distributed provisioning are the strategies they have developed. One functionality they achieve is to provide application developers with uniform context abstractions. They also provide a standard query interface to 3rd parties so they can request context data using a SQL-like query language. Third parties can push or pull data to/from Contory. They can also run an on-demand or long-running query.

2.6.2 What are the challenges?

The discussed frameworks collect and store data from their users. User data can be shared with 3rd parties. There are challenges when sharing user data and we list some of them here.

2.6.2.1 Data privacy

Sharing user data is very sensitive as the data may contain personal information. User privacy has always been a concern. It is important for both the user and the framework to have a clear understanding of the shared data [MNJ16].

- Does the user know what data is shared?
- Does the user know with whom the data is shared?

- Does the user have control of the shared data?
- Does the user trust the framework that wants to share data?
- Is the framework transparent enough in terms of privacy?

Positive answers to these questions can help users to trust a system that collects their information and shares it with 3rd parties.

2.6.2.2 Data security

Once the data is ready to be shared with an authorised 3rd party, the second step is to secure the data sharing using techniques such as cryptography [GK16]. Choosing an efficient and secure cryptographic algorithm is challenging.

2.6.3 Data consumers

In the area of human-computer interaction (HCI), the collected data from the users will be used by researchers for usability studies. Another group, who is interested in using the collected data, is designers who seek to achieve new designs and products in order to address problems (e.g. designing graphical user interfaces and web interfaces) in the real world. We also believe one of the consumers of the data is the user themselves. Currently, users live in an era of technologies, interacting with various interactive devices in their everyday life, and they seek insights into their use of digital technology.

2.7 Commercial personal user device monitoring

Many commercial device producers now offer user monitoring services for devices, e.g. smart TV, smart speaker, smartphone. Apple and Google have recently introduced user monitoring services embedded in the operating systems (iOS and Android) for their devices.

2.7.1 Apple Screen Time

Apple Inc. is a technology company that designs, develops, and sells consumer electronics, computer software, and online services. Apple is the world's largest company with a market value of one trillion US dollars [Sal18]. Apple's software includes the Mac OS and iOS operating systems. iOS is the operating system of Apple smartphone and tablet devices. It is the second most popular mobile operating system globally after Android. Apple recently announced iOS 12 and released it in Autumn 2018. One of its new features is *Screen Time*. This allows customers to understand and take control of the time they spend interacting with their iOS devices [App18, Per18]. Screen time (Figure 2.6) provides a detailed activity report of how a user spends time with their iOS device.

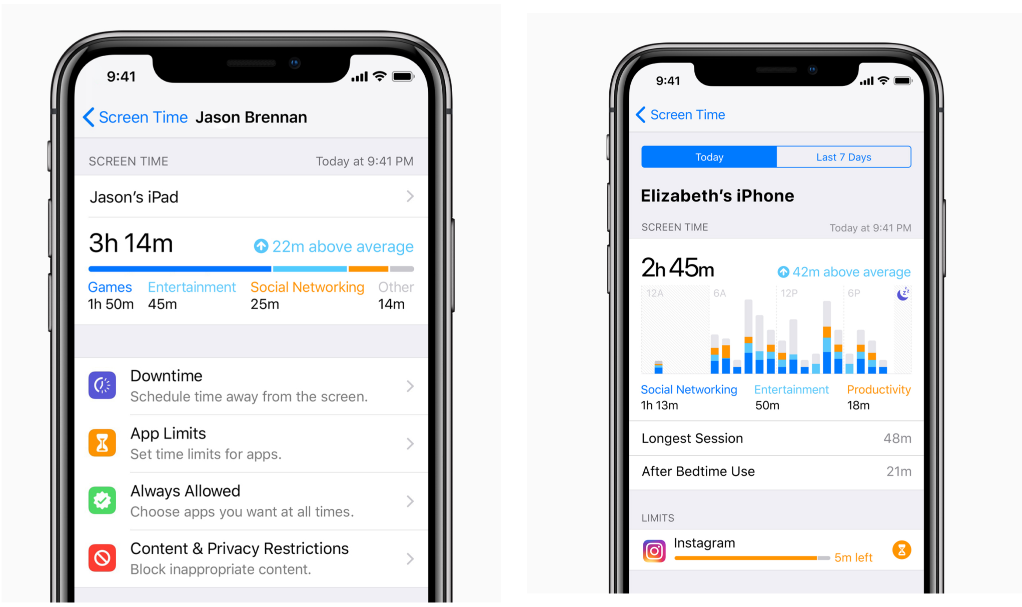

Figure 2.6: Apple Screen Time - time view

Empowering customers with insight into how they are spending time with apps and websites, Screen Time creates detailed daily and weekly activity reports that show the total time a person spends in each app they use, their usage across categories of apps, how many notifications they receive, and how often they pick up their iPhone or iPad. By understanding how they are interacting with their iOS devices, people can take control of how much time they spend in a particular app, website or category of apps. The "App Limits" feature allows people to set a specific amount of time to use an app, and a notification prompts when a time limit is about to expire.

Another functionality of Screen Time is to show the most used apps or most used category of applications (Figure 2.7).

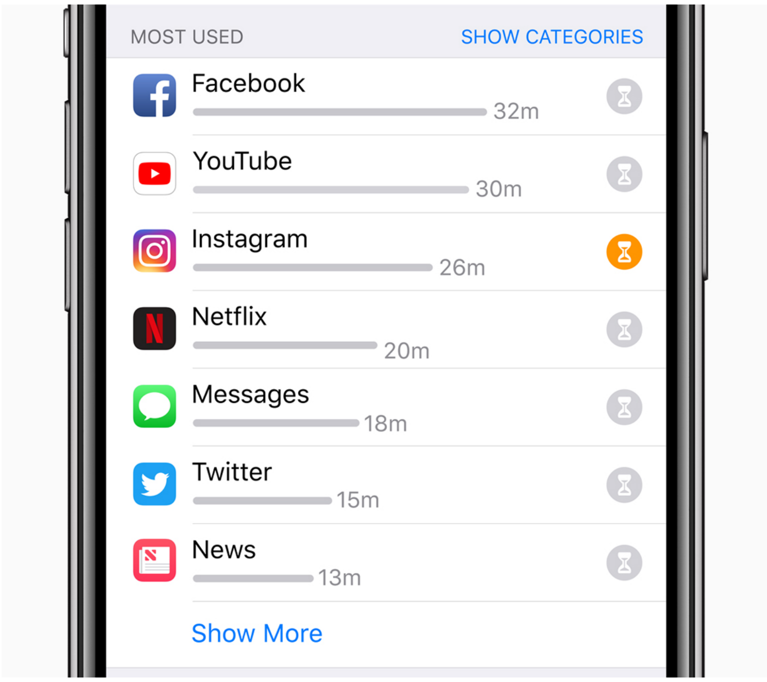

Figure 2.7: Apple Screen Time - app view

2.7.1.1 Apple parental controls

Screen Time is great for everyone to better understand and manage their device usage. It is extraordinarily helpful for children and their guardians. Parents can access their child's activity report right from their own iOS devices to understand where their child spends their time, and they can manage and set "App Limits" for them. Screen Time also gives parents the ability to schedule a block of time to limit when their child's iOS device cannot be used, e.g. bedtime. Parents can choose specific apps like Phone or Books that will always be available, even during downtime or after a usage limit is spent. Screen Time is account-based and works across all of a child's iOS devices, so settings, reports, and allowances are based on their total usage. Screen Time works with "Family Sharing" and is quick and easy to set up. Parents can configure Screen Time settings remotely for their child within the same "Family Sharing" group or locally on a child's device.

2.7.2 Google Digital Wellbeing

Google[10] is a technology company that specialises in Internet-related services and products, which include online advertising technologies, search engine, cloud computing, software, and hardware. Alphabet Inc. was set up in 2015 as the parent company of Google and now is the second largest company listed by Standard & Poor's 500 (S&P 500)[11]. Google developed the *Android* mobile operating system in 2007 [Sch07]. The latest version of the Android OS is Android 9 Pie released in August 2018. One of the features of this version is *Digital Wellbeing* [Goo18]. It is released in beta version and only available to those with a Google Pixel smartphone that runs Android 9 Pie. Pixel is an Android smartphone designed, developed, and marketed by Google.

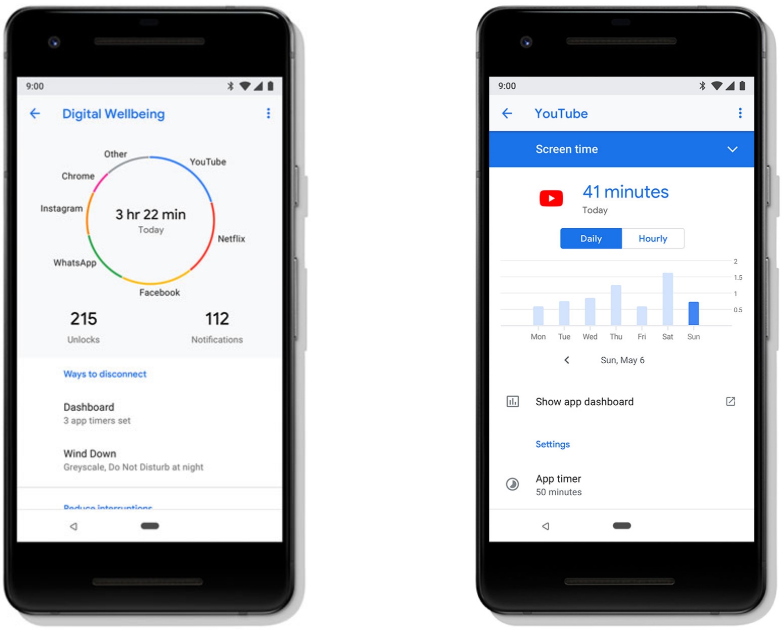

Figure 2.8: Google Family Link

Google believes that the first step toward Digital Wellbeing is often understanding more about how users interact with technology in the first place. Google is introducing new ways to keep users more informed and proactive. Digital Wellbeing is a tool that helps people better understand their technology usage, focus on what matters most, disconnect when needed, and create healthy habits for the whole family. The dashboard of Digital Wellbeing gives the users a complete picture of how they use their phones. Users can get a daily view of the

[10] About Google: https://about.google/intl/en/
[11] Amazon and the Race to Be the First $1 Trillion Company article: https://goo.gl/Qoiy19

time they spend on their phones, how frequently they use different apps, and how many notifications they get. Like the Apple Screen Time service, Google Digital Wellbeing gives the ability to monitor and limit the application usage as shown in Figure 2.8.

2.7.2.1 Google Family Link

Google announced a parental service called *Family Link* in its beta version for the U.S. market in March 2016. Google released Family Link in more countries in late 2017.

Using this service, a child can have an account while they are under a certain age. The parents can monitor children until they reach the age of 16 in Europe, except for Austria (14 years old), and Denmark and Sweden (13 years old)[12]. The Family Link service lets parents set specific ground rules, keep an eye on screen time, and remotely lock the child's device. This service provides the management of the apps the child can use, the reminder of the on-screen time limit, notification to take a break or stop using the application or device, and ability to lock the device remotely (Figure 2.9).

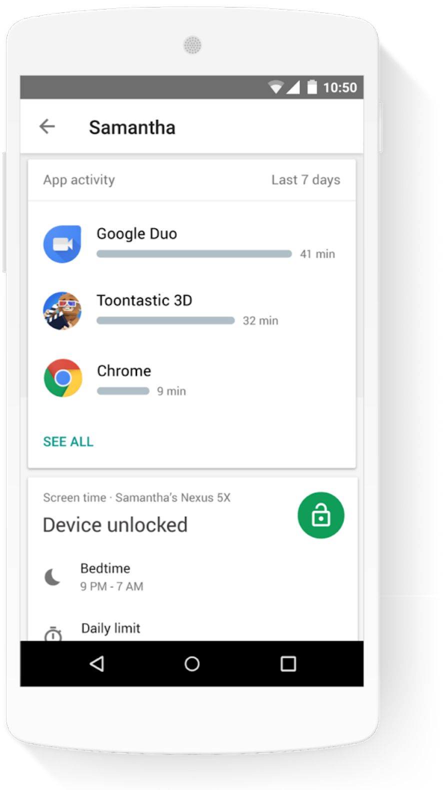

Figure 2.9: Google Digital Wellbeing - app view

[12]Family Link article: https://goo.gl/djo4Va

Children can run Family Link on Android devices running version 7.0 (Nougat) and higher. Some devices running Android versions 5.0 and 6.0 (Lollipop and Marshmallow) may also be able to run Family Link. Parents can run Family Link on Android devices running OS versions 4.4 (Kit Kat) and higher, and on Apple devices (e.g. iPhone and iPad) running iOS versions 9 and higher.

2.8 Technologies

Data collection, storage, analysis, and provision needs to be performed securely and efficiently. To achieve these requirements researchers and developers need to use cutting-edge technologies. We discuss some of thee technologies that are used by the studies discussed in this chapter and the technologies we use in our work.

2.8.1 Cloud computing

Although the concept of the cloud has been widely studied for several years, the word "Cloud" for this concept was first used by Eric Schmidt, Google's CEO, in 2006. He used "Cloud" for describing a business model of services across the internet [NJP15]. U.S. NIST (National Institute of Standards and Technology) provides a widely accepted definition of cloud computing:

> Cloud computing is a model for enabling convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction [DWC10].

Cloud providers offer various services and technologies. We discuss some of these cloud providers and their offerings in this section.

2.8.1.1 OpenStack

OpenStack (Figure 2.10) is a project originally started by NASA[13] and Rackspace[14]. It is an open source software platform that delivers a free cloud

[13]NASA website; https://www.nasa.gov
[14]Rackspace website: https://www.rackspace.com

Figure 2.10: OpenStack project

computing Infrastructure as a Service (IaaS), mainly consisting of computing, object storage, and image service components [Kon16]. IaaS provides the ability to virtualise computing resources over the Internet and offers flexible and scalable resources on-demand [MKD+15].

2.8.1.2 Amazon Web Service (AWS)

Amazon Web Service (AWS)[15] (Figure 2.11) is a cloud service provider [NJP15]. Regarding security, it provides all three keys to a secure system proposed by [CSR+16]: confidentiality, integrity, and availability (CIA). AWS is an on-demand cloud service, and customers can scale up/down resources based on their needs and pay for what they consume [NJP15]. AWS provides a wide range of cloud computing services and training to developers in order to develop and deploy a service. Some of the services AWS provides are network security, disaster recovery, auto-scaling [LKL15], securing data with encryption, and backup/recovery service [NJP15].

Figure 2.11: Amazon Web Service (AWS)

[15] Amazon Web Service website: https://aws.amazon.com

2.8.1.3 IBM BlueMix

IBM BlueMix[16][16] (Figure 2.12) is a cloud-based Platform as a Service (PaaS) designed by IBM[17][17] for developers to build and manage web and mobile applications [HHW$^+$16]. It offers three open computing technologies: OpenStack[18], Docker, and Cloud Foundry[19]. It supports several programming languages such as Java, Node.js, PHP, and Swift.

Figure 2.12: IBM BlueMix

2.8.1.4 Windows Azure

Windows Azure[20] (Figure 2.13) is a powerful cloud service created by Microsoft[21]. Windows Azure provides Software as a Service (SaaS), Platform as a Service (PaaS) and Infrastructure as a Service (IaaS). It supports several programming languages, tools, and frameworks. It has several cloud computing services including compute, mobile, storage, data management, messaging, and machine learning services [CBBP16].

Figure 2.13: Windows Azure

[16] IBM BlueMix website: https://www.ibm.com/cloud-computing/bluemix/
[17] IBM website: https://www.ibm.com/
[18] OpenStack website: https://www.openstack.org
[19] Cloud Foundry website: https://www.cloudfoundry.org
[20] Windows Azure website: https://azure.microsoft.com
[21] Microsoft website: https://www.microsoft.com/

2.9 Summary and conclusion

In this chapter, we have discussed the existing technologies, terminologies, related work and frameworks. We reported the techniques and methodologies that the existing works used. We also listed the challenges they have faced and how they have tackled them. In summary, we have shown how important it is to monitor users and their use of digital technologies, and to make use of this data in order to provide users with better services.

In data collection, contextual data (e.g. personal data and identity, location, and time) must be included. Most of the existing frameworks try to collect mostly representational context data (e.g. location and light level). However, we believe that as well as the representational context data, what users are doing (interactional data) is essential. The activity which a user is engaged in can play an important role in understanding the user. Context data is more useful if merged with user activity.

Once the data is collected, it needs to be analysed in order to extract information and insights from the raw data. The data can be analysed in various ways. Data analysis services can process user data to generate new information about the user ranging from user sports activities to user emotional state and mood. It is difficult for any framework to provide data analysis methods that cover everything. This challenge can be addressed by allowing external 3rd party services to contribute to data analysis, so that a framework's capabilities can be expanded.

When the data is collected and analysed, it is time to make use of the data. Data consumers, including the users themselves, researchers, service providers, and designers, can exploit the data to achieve their goals. Thus, sharing data with 3rd parties is necessary to take full advantage of a monitoring framework. The sharing process should be under user control to address the identified challenges of data privacy and data security.

In this work, we try to undertake all these challenges and propose a comprehensive software solution that provides the required services to address identified challenges, and give users a novel solution for managing their digital imprint.

Chapter 3

Design

In this chapter, the proposed conceptual models, system architecture, and how we approach the challenges are discussed. Baran provides a comprehensive architecture to cover the required functionalities for monitoring user digital activities, and for managing, analysing and sharing the user data.

3.1 User digital life

A user can engage in various activities in the physical world and in activities in the digital world. Our concern, however, is with the activities that the user engages in the digital world, and in particular activities in the digital world where the primary purpose is a digital effect in the digital world (e.g. adding a comment to a Facebook post or playing a smartphone game). In other words, many devices associated with a user may have a digital UI and digital controller but the user's interaction with these devices is primarily to have a physical effect, e.g. controlling the network-connected heating system via a UI or smartphone app. The digital imprint that we are recording can also include data from these other digital devices associated with an individual user, including the digital heating system, smart household appliances, and fitness band. The information from these other (physical world) digital devices provides context for the main concern, the activities of the user in the digital world.

3.2 Overview of design requirements

Users generate a tremendous amount of digital data. This large amount of digital data may contain private and personal information. Users are often unaware of how much data they create and where the data resides. More importantly, users are unaware of who, and for what purpose, uses their data. User data is distributed across many external services that provide different data protection and privacy policies. These services give users various levels of control over their data. In order to provide full control of the data and explicit awareness of how the data is used, a system should provide essential features that resolve the discussed issues. Security is the primary key feature in a trustworthy system, in which user privacy matters. Such a system should include a clear workflow of the user data sharing process so that the user can easily review how their data is used.

The number of interactive devices grows. For a user interacting through a User Interface (UI), precise information on how they are using the UI, which is achieved by a ***user interaction monitoring*** service, can contribute to understanding user activities. Understanding user activities with interactive digital devices can help in understanding the users, their preferences, and how they experience these devices. As well as a user's use of the UI, contextual information related to the user can help to understand the user. Context includes the readings from the various general sensors (e.g. accelerometer, gyroscope, magnetometer, and light level), and physiological health monitoring sensors (e.g. heart rate, GSR, ECG, EEG, and EMG) [Yua14]. User context is dynamic, and applications and services need to track the changes in order to adapt to them. Reacting to this real-time external digital information makes a system ***context-aware***.

Providing a good User Experience (UX) is a concern for device and service producers. Understanding UX is complicated, and it is even more complicated when UX is for interactive products and systems [FB04]. There are several approaches to understanding a user; the user-centred perspective that focuses on the user, the product-centred perspective that focuses on the product, and the ***interaction-centred*** perspective that focuses on the interaction between the users and the products. Interaction-centred is a widely accepted approach for understanding users and their experiences related to a product [FB04].

In a monitoring system, one of the fundamental purposes is to make use of data. It matters who benefits from the data. This study aims to make the users themselves the leading data consumers. In contrast to similar work (discussed in Chapter 2), this work focuses on giving control of the data to users (data owners) and enables them to primarily avail of the benefits of their data. In order to do this, ***data analysis*** services are needed to enrich this data, and provide the basis for useful services (e.g. personalised assistive services) for the user. As well as the existing services that the framework provides, it should be designed in a way that other services can be incorporated. For that purpose, a ***service-oriented*** design is targeted that allows the system to be extended by external 3rd party or new internal user services.

As the collected data is large, it needs to be efficiently stored and processed. There will also be a heavy processing load in order to analyse the big data, requiring sufficient and scalable hardware and software resources. For that purpose, as discussed in Chapter 2, the most appropriate solution is a ***cloud-based*** system.

Thus, a ***cloud-based***, ***service-oriented***, ***interaction-centred***, ***context-aware***, ***data analysis*** and ***user monitoring*** system design is chosen to address the identified challenges in this study. The proposed system should aggregate data from various user-related devices and services (Figure 3.1) and act as a central controller for user digital imprints.

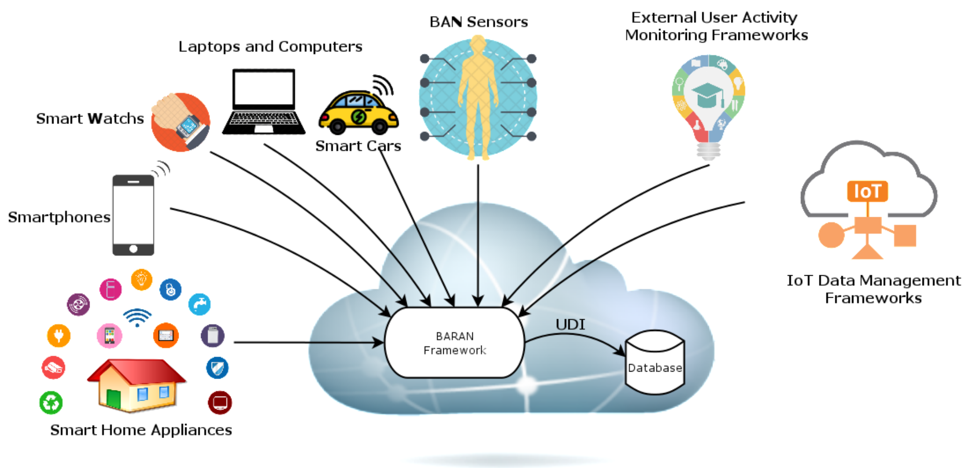

Figure 3.1: High level system design of the Baran framework

3.3 Baran framework

Baran [HH16c] is engineered as a cloud-based, service-oriented, user monitoring and data analysis framework. It enables users to monitor their digital activities. Moreover, it enables data consumers to access user data (under user control) in order to improve their services for the user. It is cloud-based because a large amount of data needs to be stored, processed, and analysed. Heavy processing for data analysis needs scalable and robust resources.

Figure 3.2 provides an overview of how external entities interact with the Baran framework. It shows a user interactively working with a digitally-controlled internet-enabled device, which supplies data to the Baran framework. Baran stores and processes the collected data. Figure 3.2 shows how Baran provides anonymised data to a 3rd party data analysis service that contributes to Baran data analytics. The figure also shows how a 3rd party service can use the analysed data (e.g. enriched user model) in order to provide personalised services to the user.

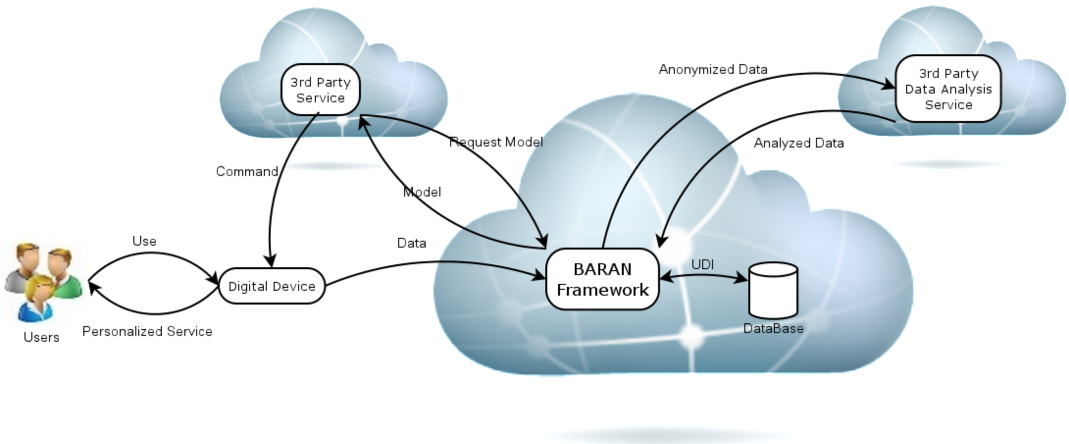

Figure 3.2: Overview of the Baran framework

In the following sections, the Baran components (i.e. data collection, data storage, data analysis, and data provision) are described.

3.3.1 Data collection

Users engage with several digital devices everyday and everywhere. They use digital devices and experience a digital life, which consists of a series of digital interactions. A digital interaction is an action a user performs with a digital device [RT03]. Digital devices are the sources of the data from which the user model is built.

Data collection involves collecting user digital interaction and associated context data. After a user agrees to use the Baran framework, the Baran data collection service transparently, efficiently, and implicitly records user activities.

3.3.1.1 User privacy and data protection

User privacy and data protection are the keys to a trustworthy user monitoring framework. In Baran, the collected data may contain personal and sensitive information. Baran should protect user data. Baran should also provide full and explicit control of the data, data collection, and data sharing.

3.3.2 Data storage

Baran should provide data storage and ensure that user data is securely stored. The data should be stored in a standard format that can be efficiently exploited. The data should also be anonymised so that no personal information leaks from the Baran framework.

3.3.2.1 Data accessibility

Data should be always accessible and available because it can be used by several services at any time. Cloud computing provides technologies to improve data accessibility and availability [PH16].

3.3.3 Data analysis

Data analysis is a process of inspecting data, forming it into information, and discovering insights (Figure 3.3). It supports decision making and recommendations. Data analysis is a complicated task. It requires data as input. The data is collected in a raw format. The data must be processed and converted into a format that can be analysed. It may also need to be repaired if there are incomplete, redundant, or corrupted sections.

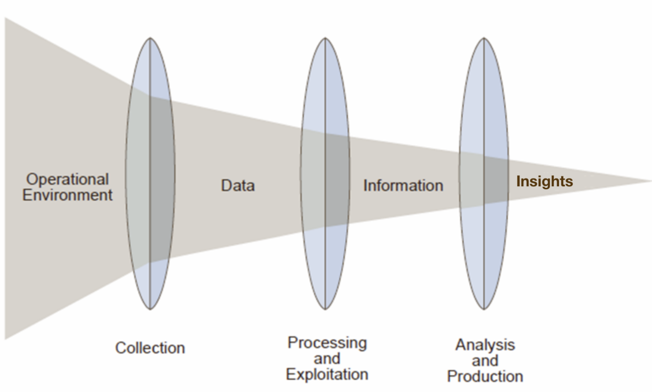

Figure 3.3: How raw data travels

Analysing user interaction enables a system to better understand a user [KHW06] and provide personalised services [LL14, LLLZ13]. Baran analyses the collected data and extracts information and insights from it. Baran should enable data consumers to make use of user data to provide better (e.g. personalised) services.

3.3.4 Data provision

Baran provides services that collect, process, and analyse data. Baran should provide an open system that supports data consumers (e.g. 3rd party services) to make use of user data, mainly to benefit the user. Firstly, 3rd party data analysis services can enrich user data. Secondly, 3rd party services can improve their services for users using user data. So, Baran should provide a data provision service that shares user data (anonymised if required) with 3rd party services under user control.

3.3.4.1 Security

In Baran, the collected data can be shared. Baran should provide essential security services that ensure the data is shared securely. This service provides decryption and encryption functionality.

3.3.5 User Digital Imprint (UDI)

Recording user data requires a standard, usable, and scalable representation scheme, which should provide efficient retrieval and structuring so that it assists data analysis and data sharing. It should have a hyper temporal space of data, in which a single data point refers to a moment in time and contains all attributes of data.

The UDI is a representation scheme with a manageable, flexible, and scalable data structure that holds various types of data and information. The UDI is a comprehensive digital record of the user digital imprint. The digital imprint is any activity, event, or interaction with an interactive device, or generally any kind of activity associated with the user that can be recorded and digitised.

Data is a series of disconnected facts and observations. Adding meaning, summarising, analysing, cross-referring, and selecting contribute to converting data to information. Patterns of information, in turn, can be worked into a coherent body of knowledge [Sto93]. Data is the currency of the digital economy. The value of the different levels of data, information, and knowledge is important for data consumers [Sat18]. Data and its subsequent phases are modelled in the well-known Data, Information, Knowledge, and Wisdom (DIKW) pyramid [Row07].

The UDI model hierarchy derives from DIKW hierarchy and simplifies it to three levels: *Data*, *Information*, and *Insights* (Figure3.4). Data are facts that are the result of observation or measurement. The measurements from personal devices and sensors (e.g. smartphone UI and smartphone accelerometer and gyroscope) provide the lowest level of the hierarchy. Information is meaningful data. Adding semantics to basic data measurements to provide meaning is the main process in converting data to information [Zin07]. The term *insight* is used for all higher level derived information, as provided, for example, by machine learning algorithms.

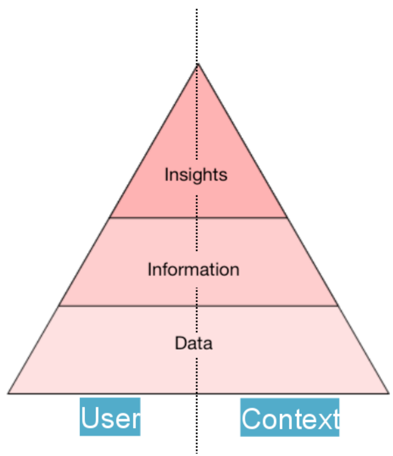

Figure 3.4: The UDI hierarchy

The UDI is valuable to many applications such as recommender systems [PP08, NM16], games, adaptable services [KKKP08]. These can change their services and system behaviours based on the UDI.

3.3.6 Conceptual model of digital activities and actions

Our main concern in this thesis is human interaction with the digital world. This can be viewed under a general conceptual framework related to the idea of activities. In the (non-quantum) world where human activities occur, all activities take time. This is also true of the conventional digital world where information cannot be transmitted quicker than the speed of light in vacuum. The basic conceptual model is one where an activity is represented with a semantic label identifying the activity, a start time and end time, and a set of activity predicates providing meta-information about the activity (e.g. identifier indicating the user who is engaged in the activity, a machine identifier representing the machine context for the activity, a geolocation predicate giving the geographical location of the activity). For the purposes of this thesis, the meta-information of interest (and one that is required) is the user identifier. So the basic activity information consists of: user identifier; activity label; start time; end time. The duration is a half-open interval i.e. it includes the lower limit (start time) and excludes the upper limit (end time).

Activities can be decomposed into constituent sub-activities. An activity of a session of using a smartphone might have sub-activities of using Facebook, using Gmail, and using YouTube. The using Gmail sub-activity might have its own sub-activities of reading emails, composing emails, and deleting emails. Some activities are instantaneous with respect to the reference time scale. That means that they have a duration that is less than the minimum interval in the reference time scale being used and so their duration interval is start-time, start-time + delta (where delta is the smallest time increment in the reference time scale). These instantaneous activities are called actions (or events) and can be represented with just a start-time rather than a duration interval. Thus, within the sub-activity of reading an email, constituent sub-activities of touches to the screen to select or swipes to scroll may be represented as actions since they are (with respect to the time model) instantaneous sub-activities of reading an email. Gathering the data about a user's activities will usually involve gathering low-level events (or actions) corresponding to the start and finish of the activity, and deriving the higher-level activity from these lower-level events.

3.3.6.1 Interaction Profile

It is often difficult to read and understand statistics, more especially where the data is large. Visualising data makes it easier to read and understand statistics. We conceptualise a graphical visualising of user data, called the *Interaction Profile*, in a way that is easy to read, understand, and interpret. As well as visualising user data, the Interaction Profile also provides an innovative comparison scheme. We define three primary metrics (Frequency, Duration, and Interaction) that can be used to characterise an activity, sub-activity, event, and interaction. We also propose two metrics derived from the primary metrics (Engagement and Intensity) that emphasise other aspects of the activity profile.

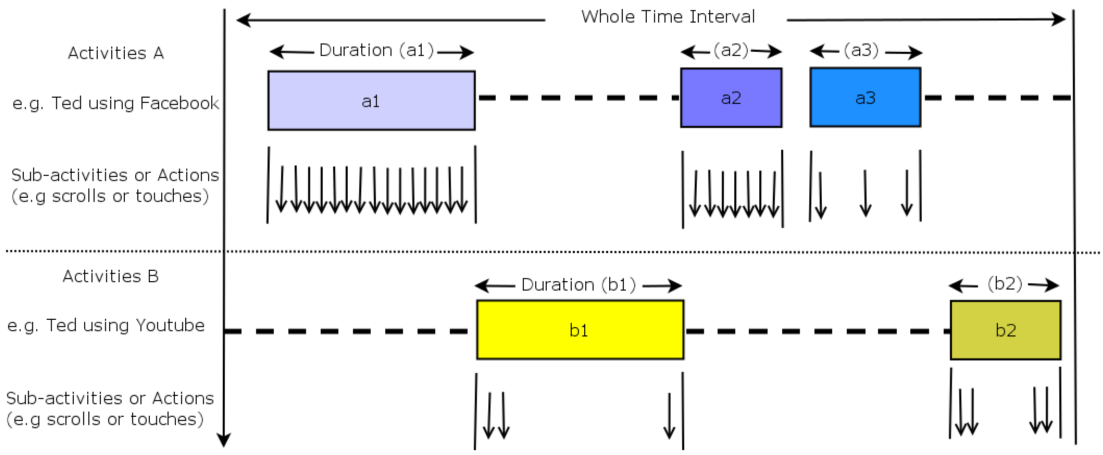

Figure 3.5: Conceptual model of Interaction Profile, illustrated by overview of two activities for one user

Figure 3.5 helps to understand how digital activities and actions can be characterised in a conceptual model. The Interaction Profile is based on a fixed-length time interval to standardise the calculation of primary metrics. The Interaction Profile can model high-level activities (e.g. interacting with a smartphone or using the Facebook app) to low-level sub-activities (e.g. posting on the Facebook app) to the lowest level of events and interactions (e.g. touch a button) that have a duration less than the minimum interval in the reference time scale. High-level activities (Activities A and B) are shown in Figure 3.5 with different colours. Each activity has a start time and an end time, and from these, the duration can be calculated. At the lower level, for each activity, there can be several sub-activities with different durations.

- **Frequency**:

  The *Frequency* is the number of times something occurs within the chosen time interval. For example, in Figure 3.5, from a general point of view, there are two activities. The frequency for the *Activity A* is three and the frequency for *Activity B* is two. The frequency of sub-activities of the first session of *Activity A* is 15.

- **Duration**:

  The total amount of time someone spends in an activity within the time interval is the *Duration* and it is measured in seconds. For instance, if a user spends a total of one hour and five minutes on an activity (e.g. using Facebook) over a period of time (e.g. day), then the duration for that activity is 3900 seconds per day. In Figure 3.5, the duration of *Activity A* is the total duration of the *Sessions* "*a1*", "*a2*", and "*a3*."

- **Interaction**:

  The *Interaction* is defined as the total number of intentional lower-level interactions someone performs during an activity or interval. The interaction metric can be calculated by accumulating the number of low-level user interactions (e.g. touch and click) within an activity or a sub-activity. For instance, in Figure 3.5, for *Activity A*, *Session* "*a1*", there are 15 interactions performed, so the number of interactions for *Activity A*, *Session* "*a1*" is 15. The interaction for *Activity A* is the total number of interactions of the *Sessions* "*a1*", "*a2*", and "*a3*." and that equals to 25.

- **Intensity**:

  The *Intensity* is derived from the *Interaction* and the *Duration* metrics, and it shows the interactivity of a user while engaging with an application. For example, if a user uses *Facebook* for a short time (Figure 3.5, *Activity A*, *Session* "*a3*") and performs a number of interactions, the intensity of that activity is higher than a user who uses *YouTube* (Figure 3.5, *Activity B*, *Session* "*b1*") for a longer period of time and performs the same number of interactions.

  The calculation: $Intensity = Interaction \div Duration$

  In Figure 3.5, the overall intensity of *Activity A* is higher than the overall intensity of *Activity B* because of the higher number of sub-activities (e.g. interactions) within *Activity A*.

- **Engagement**:

  The amount of time someone is engaged in doing something is called the *Engagement*. This metric is derived from the *Frequency* and the *Duration* metrics. It tells us how much a user is engaged with an activity or sub-activity. For instance, the engagement of a user using *Facebook* six times a day for a total duration of one hour is lower than a user who uses *YouTube* two times a day for a duration of one hour).

  The calculation: $Engagement = Duration \div Frequency$

3.3.7 Control of the communication between Baran and external entities

Baran is an extensible framework; as well as its internal data monitoring, management, security services etc., it also supports external services to communicate with Baran. External services can contribute to data collection, analysis, and provision. Baran should allow users to control who, how, and what information is collected. The framework should explicitly inform the user about external services accessing to their data. A user can then choose to grant or refuse an access request.

3.3.7.1 Service node communication preferences

Various services and devices can internally and externally communicate with Baran. Each point of communication is called a service node. For security purposes, these service nodes should register as a trusted entity in the framework so that hackers and bots cannot compromise the system. In order to do that, Baran provides Service Node Communication Preferences (CP) which is a data bundle containing an identifier, the method of communication, and security information. Baran should provide a globally available CP, which a service can use to start communicating with Baran. Baran should also provide a registration mechanism that enables service nodes to register themselves as trusted entities. The node should create a CP and share it with Baran during the registration process so that Baran can use it to communicate with the node.

3.4 Summary

In this chapter, the conceptual system architecture of the Baran framework has been presented to address the identified key features for Baran, e.g. interaction-centred, service-oriented, and extensibility. Baran essential services (data collection, data storage, data analysis, and data provision) and their requirements have been discussed. Required functionalities have been proposed in order to address the identified challenges in Chapter 2. A unique, manageable, and scalable model, the User Digital Imprint (UDI), has been designed to accommodate a variety of digital data from different digital devices (associated with user) along with information derived from that data. An innovative graphical user data visualisation (Interaction Profile) has also been conceptualised to help users better understand digital activity statistics. The following chapter presents how the current design is implemented using state-of-the-art technologies to support the conceptual idea of this work.

Chapter 4

Implementation

The previous chapter identified essential features and requirements that the implementation of Baran must meet. This chapter describes the Baran framework components and services. In this chapter, the implementation of the system architecture, User Digital Imprint (UDI) user model, Baran predictive model, and Baran Interaction Profile are presented.

4.1 Baran system overview

Baran is a user monitoring framework. On the client-side, Baran provides various data collection services for various interactive devices. Baran user monitoring services primarily collect user interaction data through the user's digital devices. These services are intended to be context-aware in that they not only gather data from the sensors in user devices but also aggregate data regarding the user from other data sources such as user-associated BAN sensors (e.g. heart rate sensor) and user environmental sensors (e.g. user office temperature sensor). In order to achieve this, Baran offers a comprehensive and dynamic data structure (UDI) that can accommodate this variety of data from different data sources.

Baran is also a data analysis framework. It deals with a large amount of data that requires storage and computing resources. Cloud computing offers a resourceful, cost-efficient, and easy-to-manage data storage and computing environment. In the Baran framework, several computing components and services work together to overcome the identified challenges (in Chapter 3) and offer the required functionalities. On the cloud-side, Baran offers a cloud-based service-oriented software architecture. Figure 4.1 is an overview showing how internal and external services work together.

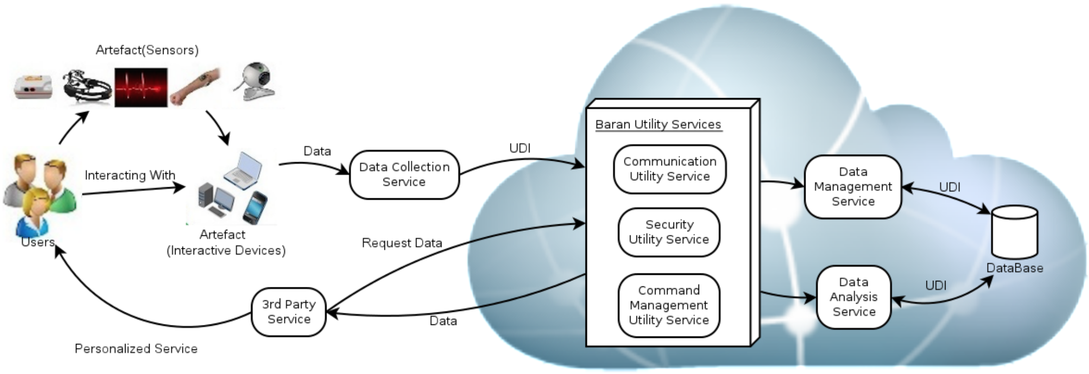

Figure 4.1: Overview of the Baran framework

4.2 Cloud-side: Baran architecture

As shown in Figure 4.1, the Baran cloud-side framework acts as a data management and computing system. It receives requests and data packages from data collection services, stores the data, and responds to the requests. It also manages 3rd party services, and their data access requests; it provides users with the details of 3rd party requests (what data, for how long, and for who) and gives the user full control of data sharing; in other words, it reduces the complexity of communication between users and 3rd parties. It aims to not only keep the user data, but also to make it usable, shareable, and exploitable mainly for the user's benefit. For this purpose, it offers data analysis services (e.g. machine learning) to enrich data quality. It proposes an innovative predictive model based on a data mining, machine learning algorithm (Association Rule (AR)). It also develops an innovative way of graphically visualising and presenting user data so that users can better understand the statistics and gain insight into their digital activities.

In this section, the architecture of the framework is described. Figure 4.2 shows the architecture of the Baran cloud system. It shows how the components and services cooperate internally.

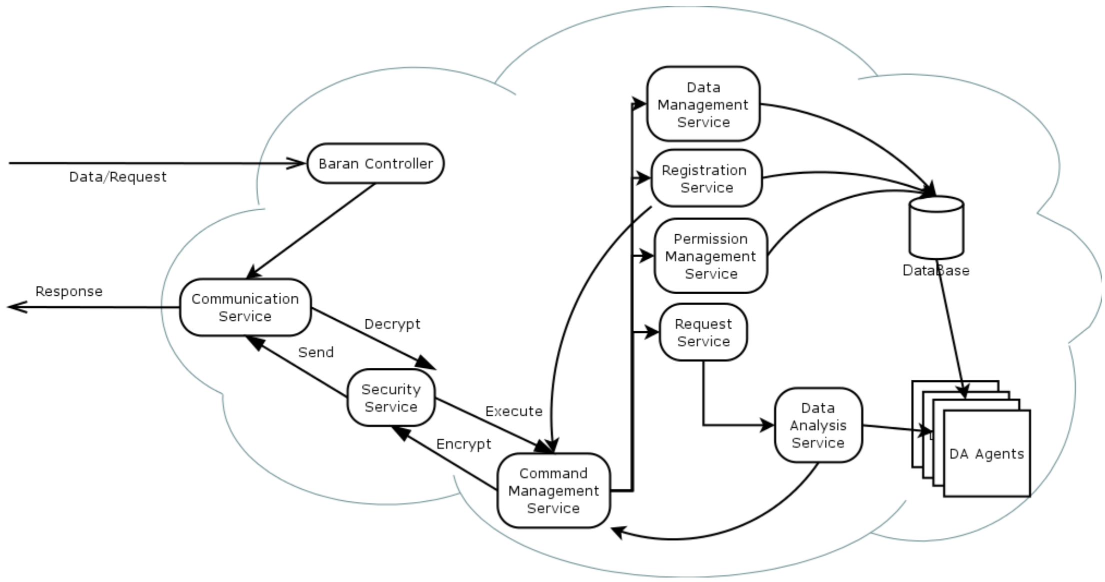

Figure 4.2: Baran cloud architecture

4.2.1 Technologies overview

This section describes what technologies are involved in implementing Baran and its services.

Baran is mainly implemented using the Amazon Web Services (AWS) cloud framework. Baran is an extensible framework and supports other cloud software frameworks (e.g. IBM BlueMix, OpenStack, and Azure). The Baran design allows services to be implemented in different cloud software frameworks and work together as a unified software solution.

Various services act as compute nodes. These nodes run in a single instance of Amazon Elastic Compute Cloud (Amazon EC2). EC2 nodes can communicate to other nodes using synchronous peer-to-peer (P2P) techniques that keep the node busy while receiving the data. Baran uses asynchronous techniques to allow nodes to operate independently from data requests; in order to do that, each node has a dedicated queue for incoming requests that is served by Amazon Simple Queue Service (SQS). Compute nodes also have a dedicated command queue to communicate with each other and the main framework controller node. These communications should not be confused with the regular requests and must be served at the highest priority. Using this queue, the controller can start, stop, and restart the nodes.

For the communication between a user digital device and the cloud-based framework, each side should provide a unique Internet Protocol (IP) address.

Smartphones and tablets connected to the Internet usually do not possess a unique IP. In order to resolve this, a service called Amazon Simple Notification Service (SNS) is employed, using it, a digital device can register and receive an access token. Using this token, the device creates a communication preferences (CP) data bundle that contains necessary information (about the device and the user) and registers itself with the Baran framework. The digital device, then, can use its CP to communicate with and send/receive notifications (a message in text format) to/from Baran (via the Baran controller and Baran communication services).

There are many techniques to optimise cloud-based software. Cloud-computing enables Baran to scale up and down its computing resources based on the business model, cost model, and the predefined configuration of each computing node. Another optimisation technique is replicating the nodes to increase the number of instances of the same process in order to improve the performance of the framework. The Baran controller is a node that is responsible for checking the number of requests and regulating the computing resources based on the data traffic and available resources, in order to serve requests efficiently.

4.2.1.1 Baran controller service

Baran provides several data storage and analysis services. Usually, they work internally together; however, they may need to be accessible by user devices and external 3rd party services. In order to protect these services, Baran encapsulates them and makes them externally inaccessible. Then, Baran offers a publicly accessible controller service with which external entities can communicate.

The Baran controller service receives a request, performs a security-check on its sender, and examines its content. If the content is not exposed and the sender is trusted, the controller passes the request through to the framework. If the sender is untrusted, it refuses the request. The only exception is when it is the first time that an untrusted entity wants to register with the framework, where, if the request is created using Baran's provided libraries, the controller passes it through.

4.2.1.2 Baran utility services

Baran has three essential utility services that are responsible for sending/receiving, encryption/decryption, and executing a request. Figure 4.3 is an overview of how they interact.

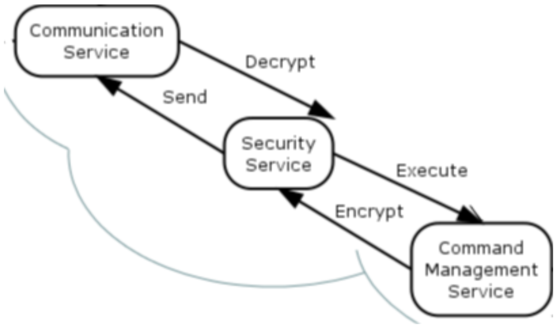

Figure 4.3: Baran utility services

4.2.1.2.1 Communication utility service

The communication utility service is the lowest layer in the utility service hierarchy. It offers two main functions: a) sending a message to an external entity; b) receiving a request from the Baran controller service (that is a verified request from an external entity). The Baran data collection service is already able to produce a trusted request. Baran also provides a library of APIs for 3rd party service developers so that they can create a trusted request.

(a) Sending a message to an external entity requires the entity's CP, which contains information such as the entity's receiving technology and communication token. For instance, the receiving technology used for smartphones and tablets is a notification service (e.g. Amazon SNS).

(b) Receiving a request requires the Baran controller to verify the request. Once the request is received, this service checks and ensures it is not corrupted Then, it passes the request to the security service.

4.2.1.2.2 Security utility service

The security utility service has two main functions: a) data encryption; b) data decryption. This service fulfils the security requirements of the framework. It ensures that the data going out of the framework is encrypted, and the data coming into the framework is decrypted. By default, Baran uses a general encryption or decryption algorithm unless a customised algorithm is supplied to

Baran, and where Baran uses this new algorithm for encryption/decryption. The information about this customised algorithm is placed in the entity's CP.

(a) The encryption function encrypts the readable data into encrypted data by using the default or customised encryption algorithm.

(b) The decryption function decrypts the encrypted data into readable data by using the default or customised decryption algorithm.

4.2.1.2.3 Command management utility service

The command management utility service receives a piece of data containing a command and a data section. It is responsible for executing commands such as sending, receiving, storing, analysing data, and registering an entity to the framework. It routes the request to the relevant service responsible for its command.

4.2.1.3 Data management service

The data management service stores the data and retries the data to/from data storage. It stores and retrieves large amounts of data. This service plays a vital role in providing efficient and reliable services to the user and 3rd party requests. In the current implementation of Baran in Amazon AWS, Amazon Simple Storage Service (S3) is used for data storage. S3 is blob storage and the data management service stores the data as is. In order to optimise the data retrieval, this service also stores a summary of the data in a fast retrieval database provided by Amazon, DynamoDB, so that data analysis services can benefit from fast retrieval.

4.2.1.4 Registration service

As discussed, a critical challenge, in order to provide a protected system, is to establish trust between entities and the Baran framework. In order to tackle this challenge, Baran provides a registration service that allows user devices and 3rd party services to register themselves with Baran and provide the necessary information (CP) so that the Baran services can communicate with the device.

4.2.1.4.1 Device registration

Once a user installs the Baran data collection service on a device, this service asks for the user's identifier (userID). If the user has not yet created a unique userID, then they can create one using a valid email address through this service. The data collection service then creates a registration request, enclosed with the necessary information about the device (e.g. International Mobile Equipment Identity (IMEI)), and sends it to the Baran controller.

On the cloud-side, an access token is generated for the device. The device is recorded in the user's list of devices along with the provided information. At this stage, the Baran framework has enough information to send a notification to the device using Amazon SNS service (or other similar services). Registration confirmation is sent to the device, and then the device can start communicating with Baran.

4.2.1.4.2 3rd party registration

External 3rd party services and assistive applications may also need to communicate with the Baran framework. The framework provides an independent library to developers and, using that, they can register their services. Once they are registered, they can start communicating with the framework, but not yet access an individual user's data. APIs are provided to the developers so that they can request access to a user's data. Requesting access to the data involves the user.

4.2.1.5 Request service

The request service manages the data access requests from 3rd party services. This service receives a request from a 3rd party service who wants to access user data. It then builds a user-friendly request for the user containing essential information about the 3rd party service and their requests. The user-friendly request transparently and comprehensively informs the user concerning who wants to access the data, what level of data (what scope), how and for how long it is going to be accessed. If the user approves the external 3rd party service then it can access the user data.

Figure 4.4 elaborates a scenario where a user starts using a 3rd party service, the 3rd party service requests access to the user data (e.g. user model), and the Baran framework explicitly communicates with the user to ensure the data sharing process is under full user control [HH16e]. The user can choose to grant or deny the data access permission. The user can also control the permission parameters (e.g. data scope, data level, and sharing duration). If the user grants the permission, Baran generates an access token based on the 3rd party request and the permitted criteria, and sends it to the 3rd party service. Now the 3rd party service can start requesting and accessing the user data under the terms and conditions of the permission.

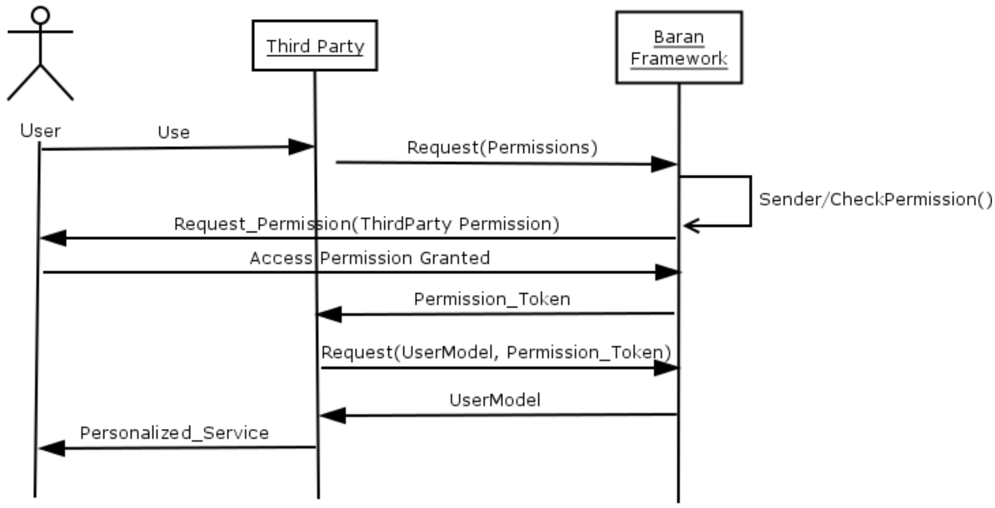

Figure 4.4: Sequence diagram of user permission being granted for a 3rd party service to access user data

4.2.1.6 Permission management service

The permission management service manages the permission certificates for 3rd party services. Once a user approves a 3rd party service, this service records the permission parameters of how, what data, and for how long it can be accessed. This service later examines a 3rd party request and ensures that the request meets the permission parameters specified by the user and the framework terms and conditions.

4.2.1.7 Data analysis service

Baran provides a wide-ranging solution that allows digital devices to monitor user digital interactions and relevant context data, and using them, a user model (UDI) is constructed. The UDI is a general hierarchical model containing raw data and derived information and insights. While the raw data is the foundation of the user model, recording it alone is not sufficient for building a rich model [KHW06]. The raw data needs to be processed and transformed into higher level information. Thus, while the raw data is the foundation for building the UDI, the data analysis services of Baran are the foundation for transforming the data to higher level information which then gets added to the UDI model. These data analysis services are the heart of the framework.

The data analysis service contains several software agents that process the UDI, mining patterns, extracting information and deriving insights. For example, the numbers recorded from an accelerometer sensor themselves have no meaning, but a software agent could interpret them to generate meaningful information regarding user movement (e.g. standing still, walking, and running).

Many methods and algorithms have been proposed to derive user behaviour patterns from user activities [AIB+10, CLWA13]. The data analysis is not limited to the built-in data analysis services provided by Baran. Baran provides an extensible and service-oriented architecture. It can allow a 3rd party data analysis service (e.g. sophisticated machine learning service) to access user anonymised data (under user control) and to contribute to enriching the quality of the data model.

4.3 Client-side: data collection service

In Baran, several services work together on the cloud-side and client-side. On the client-side, Baran offers data collection services for digital devices (e.g. smartphone or tablet). These services can monitor user interactions and associated context data (e.g. sensor data).

The Baran data collection service runs on user digital devices such as smartphones, laptops, smart watches, fitness bands, smart heating controls, smart cars, smart household appliances such as smart TVs and smart coffee makers. In this work, we implemented two data collection services: one for Android-based devices (e.g. smartphone and tablet) and one for Windows-based computers.

The Android-based service is implemented using the Android Studio IDE. It is designed as an application that initiates a background service. It provides a UI so that the user can manage the service. Using the UI, the user can start/stop the service, connect the service with their Baran account so that Baran links their data to their account, and also control 3rd party services and respond to their requests. While the UI provides the user with a management dashboard, the background service records user interactions and collects associated context data into UDI packages (under user control). It compresses the packages, creates a "sending" command, and sends the data to the Baran controller service (discussed in Section 4.4.1.1).

Customised data collection services can also be developed by 3rd parties (e.g. external developers) in order to collect user data and contribute to building the UDI model. To reduce the burden on developers, Baran [HH16a, HH16b] provides application programming interface (API) libraries so they can develop their own services that will work correctly with the framework.

4.4 Baran model and graphical presence

Baran proposes a standard data structure, the User Digital Imprint (UDI), that is used to model user digital life. It is designed to standardise data interpretation. Baran also presents an innovative graphical data representation, the Interaction Profile. It is an informative tool that helps the user to quickly view user activity reports and more easily understand statistics. Furthermore, Baran uses a machine learning algorithm and builds a useful predictive model. This predictive model uses the history of user activities and context data, learns common patterns and predicts the user's future activity based on their current situation.

In this part, the Baran data structure and model (UDI), the graphical representation (Interaction Profile), and Baran predictive model are presented.

4.4.1 User Digital Imprint (UDI)

Creating a comprehensive record of user activities is the basis for learning about users and their habits [KHW06]. As users increasingly live a 'digital life', involving physical and digital activities that create digital effects, recording user digital activities together with associated context data provides the basis to better understand users and build richer user models. These models can support the design of adaptive user assistance applications.

The UDI (Figure 4.5) is the user model that underlies the Baran framework. It is a model with a manageable, flexible, and scalable data structure that holds various types of data and information. The main focus of the UDI is to record the user digital imprint, and by that, we mean to record dynamic user interaction with digital devices. Thus, when a user touches the screen while using an application, all the information about this event is recorded in the UDI. Other events such as changes in values of sensors that are being monitored or start/stop events from a smart appliance would also be recorded in the UDI.

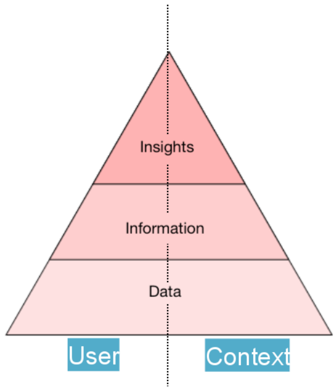

Figure 4.5: The UDI hierarchy

Beyond the basic collected data, interpretation and analysis are also applied to the data. This additional information enriches the UDI. The UDI is developed as a multi-layer data structure. It presents a timeline of a series of events. Each event possesses a start time and a duration and associated context data. The UDI model hierarchy, therefore, contains three levels corresponding to *Data*, *Information*, and *Insights*.

In order to better explain the UDI model hierarchy, consider a scenario where Bob likes to drink a cup of coffee at home on Saturday mornings. In this scenario, the raw sensor values (e.g. the time the coffee maker is switched on) lie at the data level, the fact that this corresponds to making coffee lies at the information level, and the pattern that Bob makes coffee at home on Saturday mornings lies at the insights level. A company offers a smart coffee maker that can be remotely commanded to make coffee. A data analysis service can access a user's data (under user control), learn their "making coffee" habits, and proactively and remotely send the "making coffee" instruction to the coffee maker. For instance, if today is Saturday, it is morning, and Bob is at home, then the coffee maker can be commanded to make coffee for Bob.

Baran uses JavaScript Object Notation (JSON) format for storing and transmitting the UDI. JSON is an open-standard and language-independent data format. It is a human-readable data format consisting of serialisable attribute-value pairs that can be read with no complicated parsing and translation. Google

GSON is a commonly used library for converting a JSON file to a JSON object and vice versa. It is not efficient to transfer the JSON file as is because it is large and contains repetitions and redundancies; however, when it is compressed, the size of the file is significantly reduced.

4.4.1.1 UDI life-cycle

Figure 4.6 shows the UDI life-cycle where the UDI travels through the utility services to be encrypted, dispatched to the other party, and again travels through the utility service to be decrypted, so the UDI is transmitted securely between two entities.

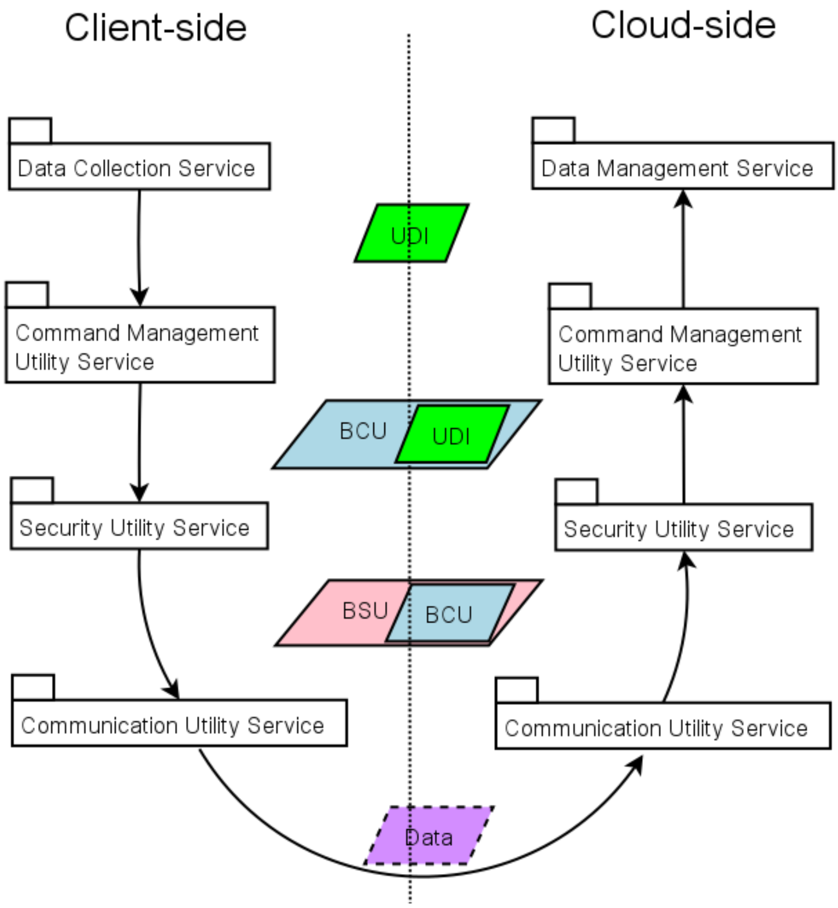

Figure 4.6: The UDI life-cycle from a device to the Baran framework

In this scenario, the Baran data collection service (client-side) collects user data, creates UDI packages, and transfers them to the Baran framework (cloud-side). The UDI packages other data structures using utility services. Here we describe the UDI life-cycle (Figure 4.6) through the utility services.

The data collection service prepares a UDI package, sets its command to "storing" and the destination to "Baran controller", and, finally, invokes the "sending" function of the command management utility service with the UDI package attached.

The command management utility service, next, creates a data structure, called Baran Command Unit (BCU), that provides two sections (header and body). The header of BCU contains information about the destination and the initial command (in this scenario "sending"). The commands of this layer (sending and receiving) should not be confused with the actual data-related command (in this scenario "storing") in the data collection service. The body of the BCU contains the UDI package. As the data is going to be transmitted through a communication channel, this service must ensure the data is secured; in order to do that, this service invokes the "encryption" function of the security utility service with the BCU data attached.

The security utility service creates a Baran Security Unit (BSU) data structure, then uses the default (or customised) encryption algorithm to encrypt the body part of the BCU, and puts the prepared data into the BSU. The BSU also contains a header section that provides information about the encryption algorithm and some error checks. Then, the security utility service invokes the function "transmitting" of the communication utility service with the BSU attached.

The communication utility service converts the BSU data into binary data and sends it over the network (e.g. Internet) to the provided destination (in this scenario "Baran controller").

On the other side, in the Baran framework, the Baran controller receives the binary data, examines the trustworthiness of the sender, and, finally, passes the data package to the communication utility service. The communication utility service gets the BSU data and passes it to the security utility service, which decrypts the content of the BSU using the provided decryption algorithm, then gets the BCU data, and passes it to the command management utility service. This service, finally, gets the UDI package, identifies the data-related command (in this scenario "storing" which should be processed by a data management service), and passes the UDI to the relevant service (in this scenario "data management service") for further processing.

4.4.2 Baran predictive model

Baran can make use of any appropriate data analysis algorithm, either as a built-in or 3rd party service. A machine learning model based on the Association Rule algorithm has been implemented and included as a default built-in service.

4.4.2.1 Association Rule (AR)

Association rule learning is a rule-based machine learning method for discovering interesting relations between variables in large databases. It is designed to identify strong rules in databases using some measure of interest. Association rules analysis is a technique to uncover how items are associated with each other in three ways.

- **Support**: determining how popular an item is by appearing in the data. For instance in Figure 4.7, the *Apple* appears in the listed transactions four times out of eight total appearances. So, the **Support** of *Apple* is $\frac{4}{8}$ that equals $\frac{1}{2}$.

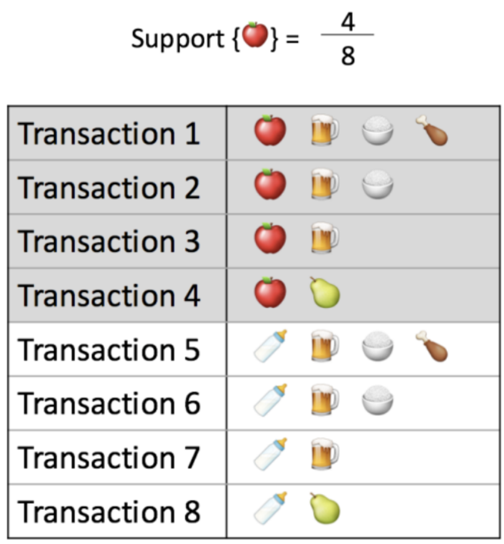

Figure 4.7: Association Rules: Support

- **Confidence**: estimating how likely item **Y** appears given the appearance of item **X**, **X** $\rightarrow$ **Y**. For instance, as shown in Figure 4.8, in order to find the **Confidence** of *Apple* $\rightarrow$ *Beer*, we need the **Supports** of *Apple with Beer* and *Apple*. The *Apple* support of $\frac{1}{2}$, and the support of the *Apple with Beer* is $\frac{3}{8}$, so the **Confidence** of *Apple* $\rightarrow$ *Beer* equals $\frac{3}{8} \div \frac{1}{2} = \frac{6}{8}$.

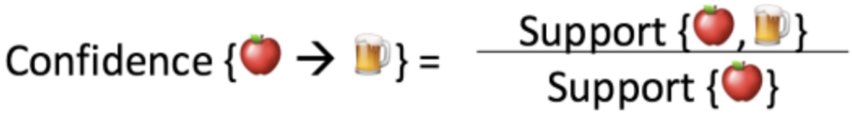

Figure 4.8: Association Rules: Confidence

- **Lift**: measuring how popular the item **Y** is along with the item **X**. In order to find the lift, the appearances of both items together are divided by multiplying the appearance of each of them separately. For example, the **Lift** of *Apple* $\rightarrow$ *Beer* equals the **Support** of *Apple* $\rightarrow$ *Beer* divided by the **Support** of *Apple* multiplied by the **Support** of *Beer*; giving the equation, $\frac{3}{8} \div (\frac{1}{2} * \frac{6}{8}) = \frac{48}{48} = 1$. A **Lift** value greater than one means it is likely to happen and equal or less than one means it is less likely to happen.

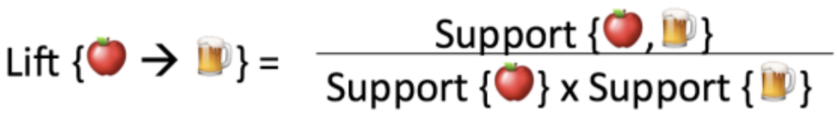

Figure 4.9: Association Rules: Lift

Apriori is a well-accepted algorithm in the Association Rules analysis community. It analyses the data and produces a list of all available subsets of the data based on the configuration parameters that are discussed shortly. The output contains a set of attributes and values along with the probability of each set. In general, if you have r variables $(var_1, var_2, var_3, \ldots, var_r)$, each of which can take n values $(val_1, val_2, val_3, \ldots, val_n)$, then there are in total $n * n * n \ldots * n = (n)^r$ combinations possibilities. As the output is a long list of possible combinations, it is a challenging task to search for a result. In order to tackle this, a tree-style model is proposed. It provides a fast query interface to a set of rules. In this model, there is a root (F). Each level of the tree corresponds to an attribute, and at each level, the number of leaves equals the number of unique values of that attribute. So, the depth of the tree is equivalent to the number of attributes. There is a challenge when a search query does not include all attributes. The proposed model defines a fixed leaf, value *NULL*, for each level of the tree, and this fixes the problem. The graph of this model is shown in Figure 4.10.

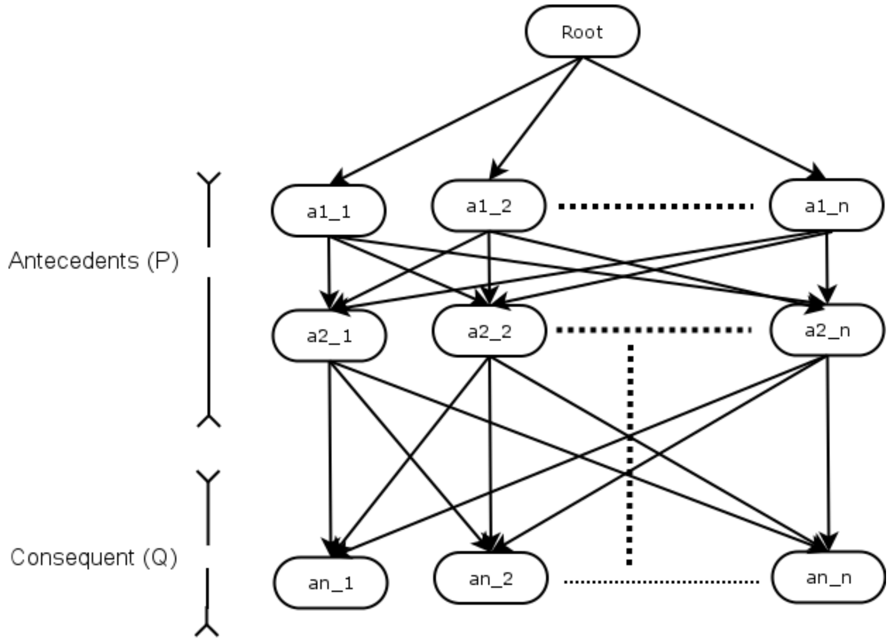

Figure 4.10: Baran predictive model tree-style graph

Let F be the Baran predictive model. Let A={a_1,a_2,...,a_n} be a set of attributes. Let P={p_1,p_2,...,p_n} be a set of antecedents, each of which can include a subset of A. Let Q={q_1,q_2,...,q_n} be a set of consequents, each of which is a member of A. The depth of the tree is the maximum size of P + 1 for a leaf leading to a Q.

$\forall p \in P, \quad \exists a \subseteq A.$
$\forall q \in Q, \quad \exists a \subseteq A.$
$\forall x \in P, \quad \exists y \subseteq Q, \quad f(x)- > y.$

The Association Rules analysis algorithm, Apriori, introduces several ways of producing the result set. It has two main parameters: the support and confidence. For example, having about 900 instances of user data, a support of 0.05, and a confidence of 10%, will provide about 75000 rules. Each rule has two parts, Antecedent P and Consequent Q, and also support and confidence. Running a search algorithm over 75000 records does not provide adequate performance for a real-time or even a semi real-time prediction. On the other hand, in the Baran framework, the goal of this predictive model is to run a query for the current context and find all the matches, not only one match. In order to do that, $2^n - 1$ runs are required to get a set of predictions. In the standard normal search algorithm, it takes 2ms to run a query (on a machine/instance). For example, if there are 11 attributes for P and one for Q, there will be 2047 queries. It takes about 4.5s to get the Q of a P. It will be exponentially worse for additional attributes in P. The idea of the proposed model for Association Rule outputs is to make the search process quicker.

If P, then Q.
Query counts: $2^n - 1$.
n is the number of attributes (A) in P.

Training the Baran predictive AR model is a time-consuming process as it involves processing, rule extraction, rule normalisation, and tree model creation. Apriori performs the rule extraction and its processing time depends on the size of the dataset, the support, and the confidence parameters. Rule normalisation processing time depends on the number of rules extracted from the previous part; this part is quicker than Apriori rule extraction. Finally, the model creation processing time depends on tree size. Table 4.1 shows a summary of how the proposed model compares with a normal search algorithm.

Table 4.1: Normal search vs. Baran model comparison

Model	Apriori Support	Apriori Confidence	Training Size	Rules Extracted	Apriori Processing Time	Normalization	Model Creation	Attribute Counts	Query Counts	Prediction Search
Normal Search	0.05	0.1	858	35k	5s	-	17s	11	2047	2455ms
	0.005	0.1	858	73k	13s	-	102s	11	2047	4691ms
	0.005	0.1	260k	138k	52m	-	3m	11	2047	>10mins
Baran Model	0.05	0.1	858	17k	17s	1s	2.5s	11	2047	79ms
	0.0005	0.1	858	56k	27s	3s	4.5s	11	2047	100ms
	0.005	0.1	260k	138k	52m	2s	4s	11	2047	100ms
	0.005	0.1	260k	138k	52m	2s	4s	12	4095	285ms

k = thousand s = second ms = millisecond

There are two steps to be compared: one is the training and the second is the testing process. In the training process, the part that most delays the process is Apriori processing time. For creating a model, the Baran model is faster than the normal model as shown in Figure 4.11. As the Baran model is a tree-style model, its execution time does not depend on the size of the tree, and, by increasing the number of attributes, this causes an increase in the number of queries. It has a linear performance in comparison to the normal search algorithm that has an exponential performance. Figure 4.12 shows the performance of the two algorithms, each of which has a different number of rules, for finding a match.

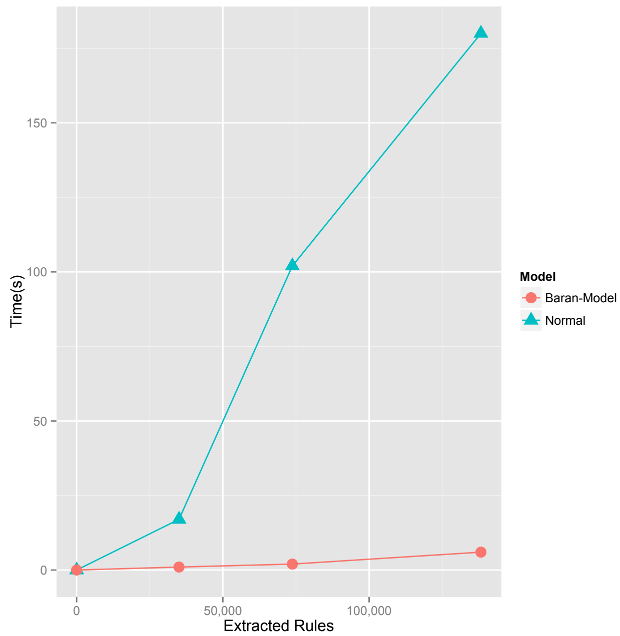

Figure 4.11: Baran predictive model tree graph - model creation time

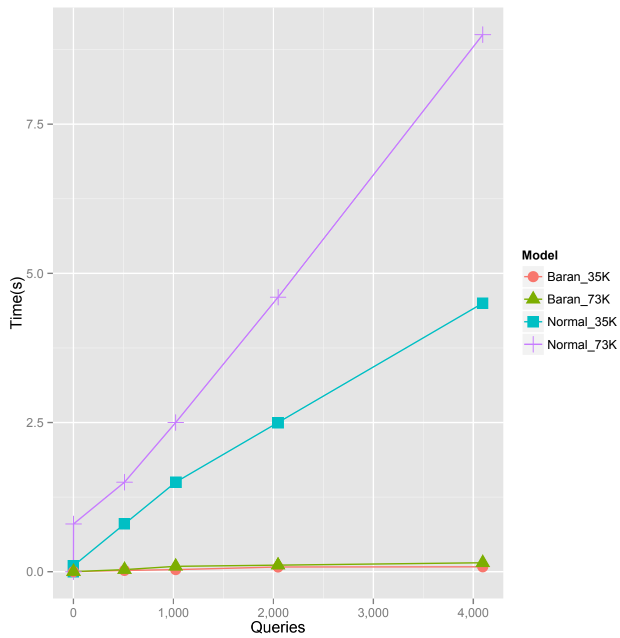

Figure 4.12: Baran predictive model tree - query execution performance

As it is shown, by increasing the number of rules, the normal search algorithm performs poorly as its model with 73K rules requires 900ms to execute a single request containing 4095 queries. In contrast, our proposed model outperforms the normal search algorithm and executes the same query with the same configuration in less than 150ms. Moreover, another advantage of the proposed model is that an increased number of attributes does not make a significant difference as seen in Figure 4.12.

4.4.3 Interaction Profile

The Interaction Profile (described in detail in Chapter 3, Section 3.3.6.1) provides a concise, easy to understand information graphic to present the results of user digital activity monitoring.

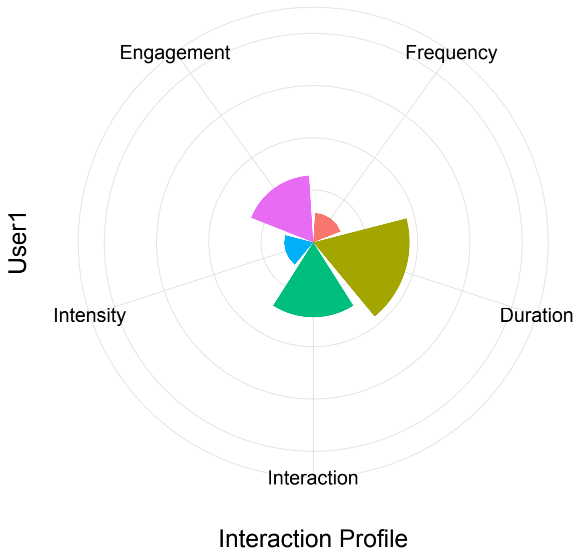

Figure 4.13: Interaction profile

A graphical symbol is designed in a polar graph style (shown in Figure 4.13). This graphical symbol can be used in various scenarios to visualise data and support side-by-side comparison of the results of digital activities monitoring. An important use of the Interaction Profile is to provide comparisons with respect to the three main parameters: user, activity, and time. By varying one of these parameters while keeping the other two fixed, we can offer clear side-by-side comparisons.

4.4.3.1 Time-based comparison

When a user performs the same activity, e.g. using the same application and it needs to be compared at different times (e.g. comparing a working day to a leisure day), two interaction profiles can clearly show the differences when they are side-by-side (Figure 4.14). In this comparison, the time is the basis variable, and the user and the activity are fixed parameters. The starting point of the time can be different, but it must have the same duration (e.g. one hour or three days).

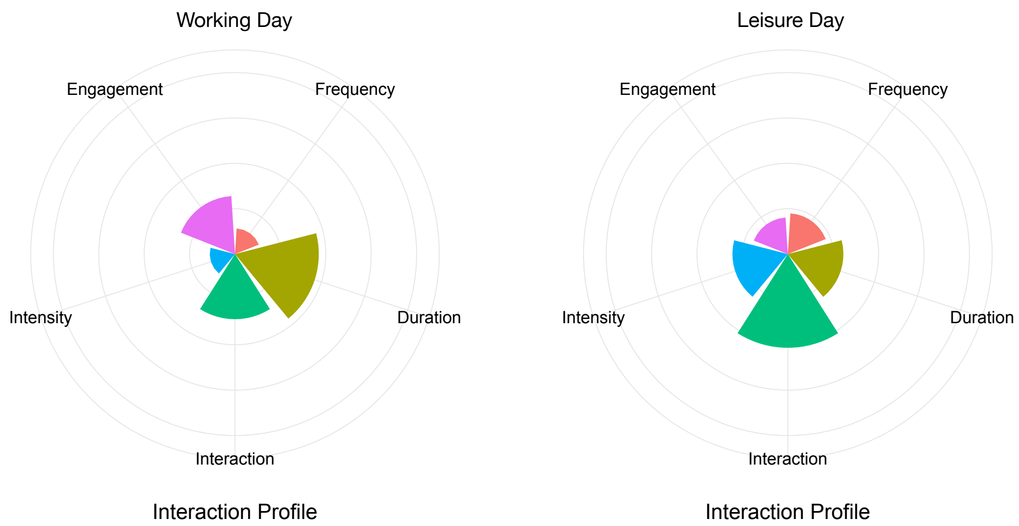

Figure 4.14: Time-based Interaction Profile

4.4.3.2 Activity-based comparison

It is also possible to compare a user performing different activities, e.g. using different applications (Figure 4.15), over the same period. In this comparison, the time and the user are the fixed parameters, and the activity is the basis variable.

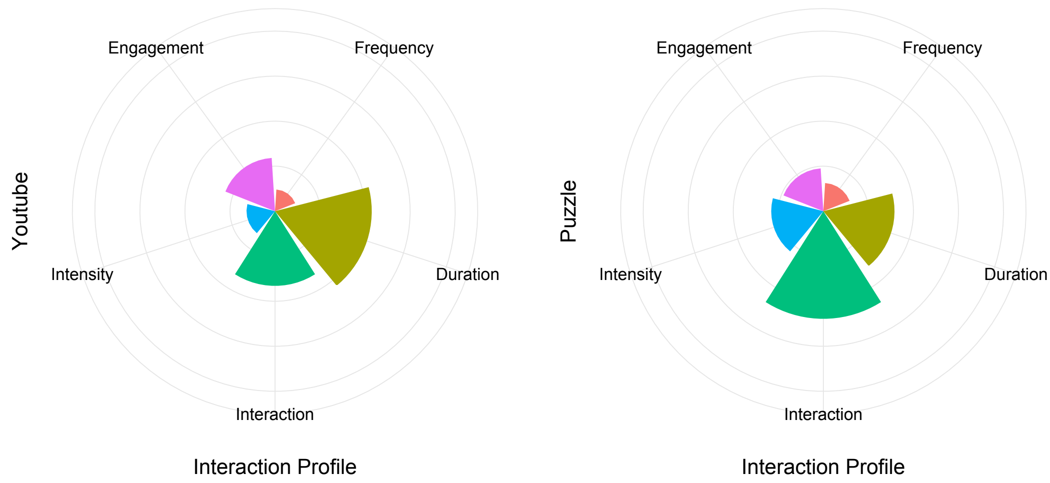

Figure 4.15: Activity-based Interaction Profile

4.4.3.3 User-based comparison

The Interaction Profile can provide information about the same activity over the same period (fixed parameters), but for different users (basis variable) (Figure 4.16).

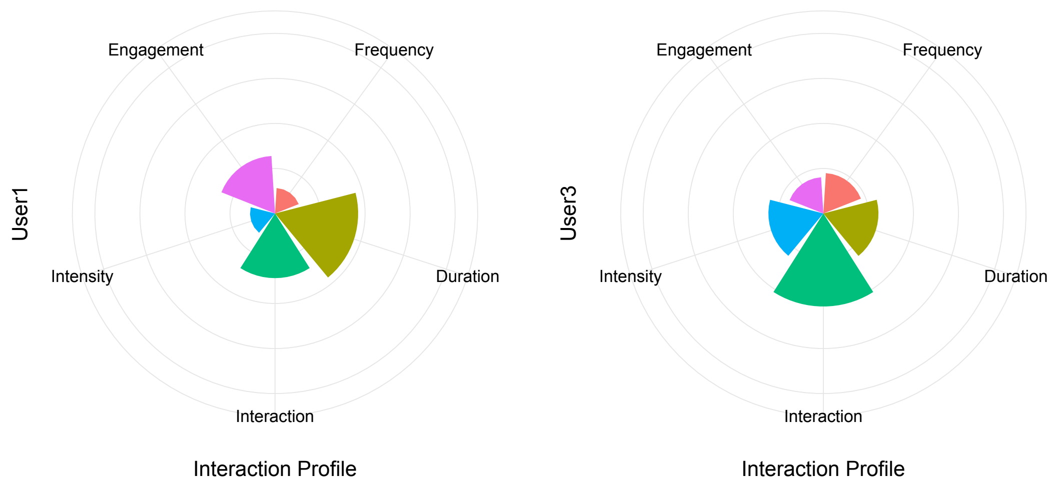

Figure 4.16: User-based Interaction Profile

Technically, bringing together different metrics with different units is complicated. In order to simplify the process, a maximum number of 10 billion is used in calculation formulas. Also, in order to normalise the metrics the Logarithm to the base 10 (Log_{10}) of values is used to provide a number less than 10. A final result is a number within the range zero to a maximum of ten ([0:10]). Finally, the metric will have a value of zero to one ([0:1]) by dividing by 10.

4.5 Summary

This chapter presents the implementation of the Baran system architecture and the required components and services are identified. Baran has two sides: the cloud-side framework and the client-side services.

On the cloud-side, we implemented a cloud-based service-oriented software framework. An overview of the technologies used in the implementation of Baran is given. The Baran framework consists of several computing services, each of which is described in detail. On the client-side, we implemented data collection services that run on Android-based devices and Windows-based computers. Baran also offers a user model (UDI), a machine learning predictive model, and an innovative graphical user data presentation (Interaction Profile). This chapter provided comprehensive definitions for them along with an evaluation for the implemented predictive model.

The next chapter presents several scenario-based experiments where Baran and its services work together to show the variety of use cases Baran can support.

Chapter 5

Experiments and Results

Baran is a framework that provides a wide range of functionalities and services to users and 3rd party services. In order to show the capabilities of the framework and how the designed system works, several scenario-based experiments are performed, and the results are presented. The scenarios are chosen to be examples of different use cases in which Baran can be beneficial. The scenarios also demonstrate how Baran system components work together to accomplish the goal of assisting the user. In this chapter, the designed scenarios and their results demonstrate the capability and the usefulness of the Baran system in monitoring user interaction, leveraging the user digital imprint, and letting 3rd party services use user data (under user control) to provide smart services for the users. This section introduces and discusses the scenario-based experiments in detail.

5.1 Scenario A: self-monitoring use of digital devices

This is the simplest scenario of a user using Baran to monitor their own use of digital devices and provide them with a summary of patterns of use (insights into how they use their devices). These analyses can be performed on the user's device itself, or the user can make use of a more sophisticated external analysis service provided by a 3rd party.

Users spend much of the day digitally connected. They increasingly experience a virtual digital life. According to comScore's 2017 report, the average adult

(18+) spends about 4 hours a day on their smartphones. Besides all the positives of a virtual life, there are also downsides. A user's personality is affected by the positives and negatives of being in a virtual life. The user usually does not distinguish the source of the influences. One of the first steps to understanding how virtual life affects a user's life is to know how they interact with their digital devices. Analysing the user digital imprint can contribute to finding out how a series of digital actions are affecting a user.

This scenario is designed to monitor a user's activities and report how they interact with their digital devices. The user can obtain basic descriptive statistics (e.g. frequency and duration) or more complete insights. In this scenario, it is assumed that a user has a smartphone, has the Baran data collection service installed on the smartphone, and agrees with the Baran framework terms and condition with regards to collecting and analysing his/her data.

5.1.1 Experiment

A pattern recognition service is proposed to access a user's data model (UDI), under user control, analyse the data, and extract common patterns of user interactions. This service demonstrates how raw data can be transformed into insights (e.g. patterns). This service can be an external service, and as Baran provides service extensibility, the external service can contribute to the Baran framework. The user later can access the output of the external service (e.g. patterns) and can also share them with other 3rd party services. Baran is a secure and trustworthy framework providing the user with the option of not sharing personal information with internal and external services. In this experiment, the assumption is that the user chooses to be unidentifiable in the data. Baran removes the user's personal data and anonymises the data in the sharing process to ensure the data is not traceable and identifiable by the analysis service. In this experiment, two months of user data are collected from the user's smartphone. The data includes digital activities and excludes personal information and private data. The user reported no noticeable extra performance cost or extra battery usage because of running the Baran data collector service. We also have examined the performance of the service on various Android devices and measured the cost as ranging from a minimum of 0.3% to a maximum of 2.9% CPU usage on average per day.

Here is the Baran report of some essential statistics and graphs to demonstrate how the collected data can be presented to a user.

5.1.2 Analysis and discussion

Figure 5.1 provides the frequency and duration usage of the user interaction with his/her device over a week and also compares the weekdays and weekend data.

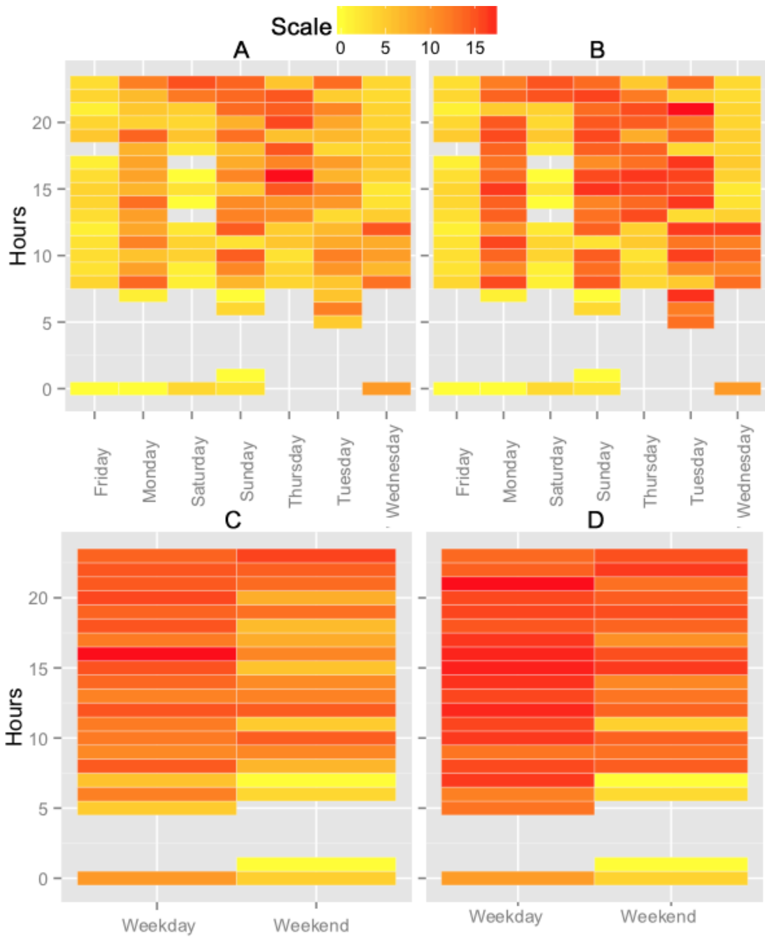

Figure 5.1: Device usage frequency (A and C) and duration (B and D)

Users interact differently with their digital devices over time, so analysing the user interactions provides the pattern of device usage. The heat map in Figure 5.1 shows the pattern of the user device usage and includes the most frequent days and times of day he/she interacts with his/her device. The device usage pattern is useful in many ways. One of them is to help services prepare in advance to assist users by predicting when the user wants to use their services. Another way is to allow services to do their complex and heavy computing tasks on the device when users are not usually working with their devices.

For instance, let us compare the user's activities on Saturday and Sunday. In Figure 5.1 graph **A** *(the frequency of device usage)* shows that on Saturday the user uses his/her device less frequently than on Sunday. In contrast, graph **B** *(the duration of device usage)* indicates that on Sunday the user spends more time on the device than on Saturday.

Several insights from analysing the frequency and duration patterns of device usage and user interactions are listed below:

- The user is less active on Friday and Saturday than the other days.
- The user spends more time interacting with the device on Monday and Sunday than the other days.
- The user spends less time interacting with the device on Wednesday afternoon than morning.
- The user never uses the device from 1 am to 5 am.
- The user uses the device more frequently between 8 am to 4 pm over the weekdays and more in the evening over the weekend.

Figure 5.2 gives a detailed summary of how individual smartphone applications are used on average. Google Chrome, Viber messaging, and email are the most used applications. The user used Viber more on Sunday and Thursday than other days as shown in the "Day" section of the Figure 5.2. It also can be concluded that the user used the Viber application in the evening more than the other parts of the day on average. Further analysis can provide more profound insights. Figure 5.3 shows the duration of the application usage by the user over different time-scales (e.g. hours, days).

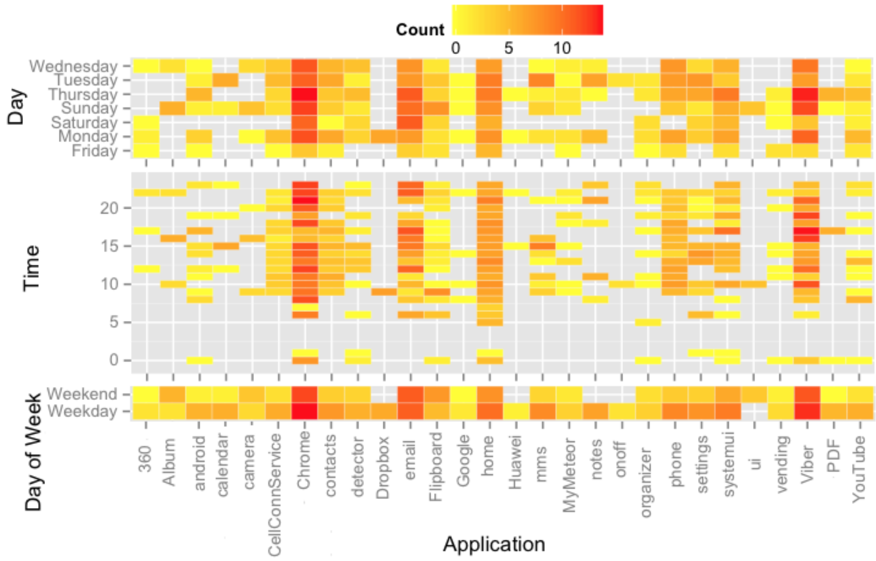

Figure 5.2: Application frequency of usage

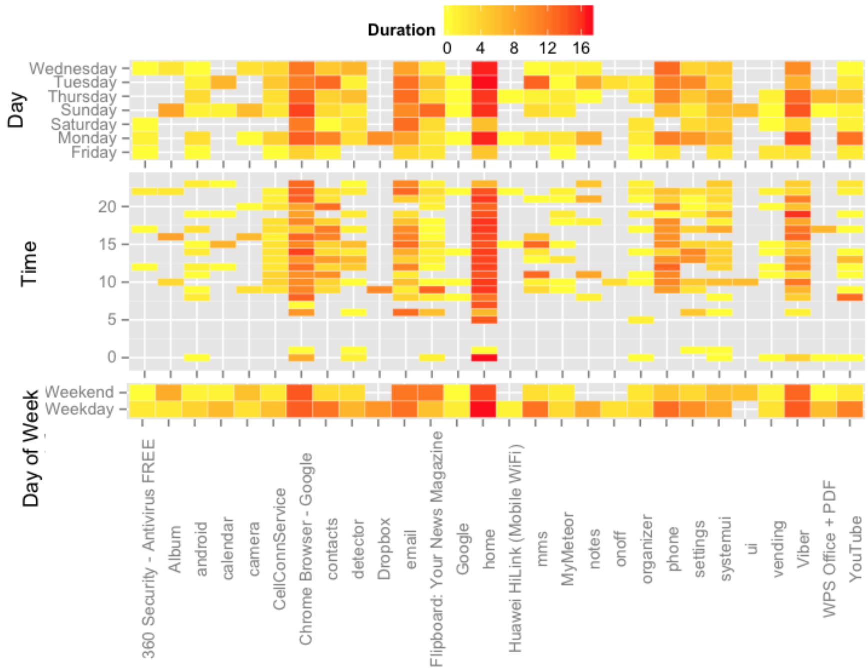

Figure 5.3: Application duration of usage

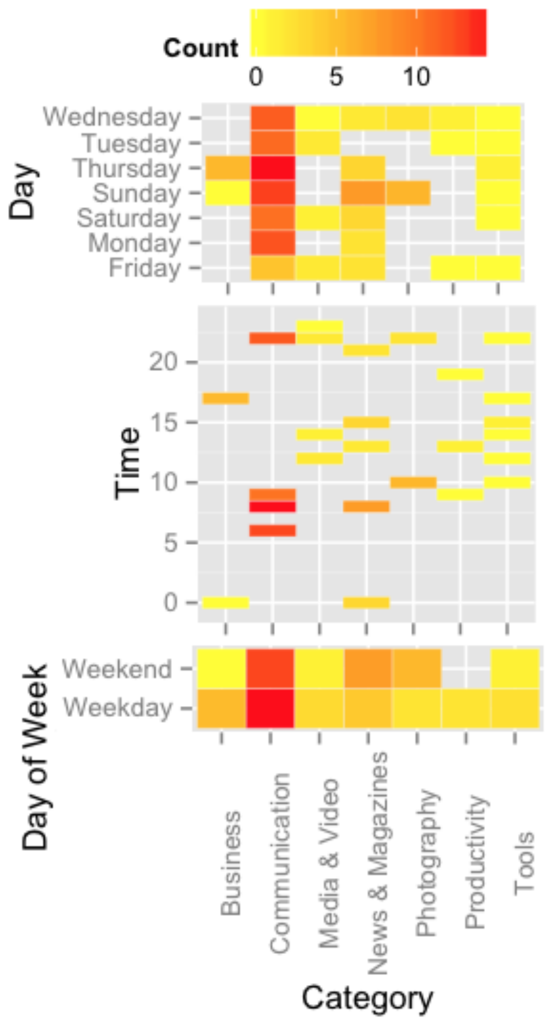

Figure 5.4: Application categories frequency of usage

Putting applications into categories and performing the analysis based on these provides interesting patterns, which represent how the user uses different types of application. These patterns are useful in classifying users based on their interests. Figure 5.4 is a diagram showing how frequently the user uses the applications in various categories. Figure 5.5 shows the amount of time the user spends on different application categories. The result shows the applications in the Communication and News&Magazines categories are used more than other categories, particularly in the morning. The Tools category is used on the weekdays more than the weekend.

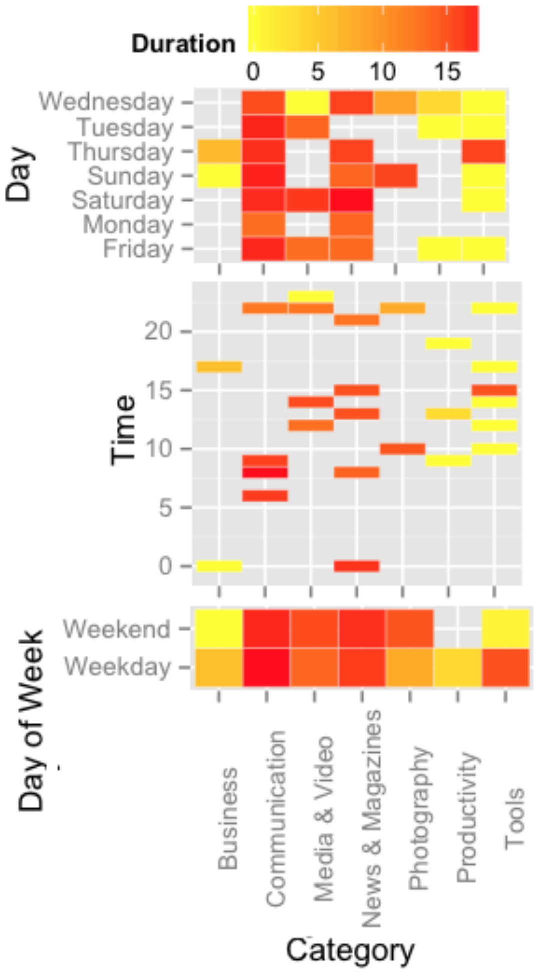

Figure 5.5: Application categories duration of usage

5.1.3 Summary

This scenario shows how a user's data can be analysed, and the extracted insights can be interpreted. In this experiment, the Baran framework shares anonymised data with a service, under user control, demonstrating that the user controls the whole process of data sharing. It eliminates data privacy and protection concerns. Finally, there are some useful patterns showing use of the smartphone and applications, and these patterns could be beneficial for the users or for service producers to take the user's habits into account in providing their services. This experiment also demonstrates that internal and external services can be incorporated to enrich the data quality and transform data into information and insights.

5.2 Scenario B: monitoring child use of technology

As the influence of digital technology and especially the Internet has increased, the debate about its impact has grown. Children born in the digital age use digital technologies everywhere (e.g. home and school). As the use of digital technologies, such as the Internet, is extending to younger children in critical activities like their education and learning, there is an increasing need for empirical research to understand children's use and experience with these new technologies in their daily lives [LEM$^+$17].

While our user monitoring and data analysis framework, Baran, enables children to self-monitor themselves, it also enables their parents and carers to play a role in their use of technologies. A parent can check which applications a child uses, for how long, and how they interact with them. Studying UX for children is usually done in a laboratory setting. Baran can support UX research for children by, unobtrusively, in the background, monitoring the children as they interact in their natural environment without artificial constraints. Thus, we can discover to what extent a child of a particular age engages with, and how they interact with, existing applications. This monitoring process provides an understanding of how children naturally use the technologies in real life and that is the key to redesigning and improving technology for children.

In a Baran-based research study [Mit16], a web service for parents was designed to make use of low-level information collected by Baran and to monitor children and analyse their use of smartphones (some results are presented in Appendix G). It monitored the child's smartphone usage and behaviour patterns. It anonymously presents the child's data to protect the child's privacy. If the parent wants to drill down to see the data, then the child must be notified. It is a proof of concept scenario of how Baran can support the monitoring of children.

5.2.1 Experiment

In this experiment, Baran records a child's interaction and associated context data, analyses the collected data, extracts information and insights from the raw data, and presents patterns such as diversity of technology use according to age and gender. It shows the value of this approach and presents exciting results of precise monitoring of children's smartphone usage. In order to address user privacy, Baran initially provides anonymised data within the extracted patterns. Carers can request access to the de-anonymised user data if they want to get more information. The user will be notified and can permit or deny the request.

5.2.2 Selection and participation of children

In this experiment, a number of children participated. The participants were invited, the purpose of the research study was explained, and the permission of the parents was obtained to allow Baran to collect children's digital interactions with the devices. Children interact with smart devices (i.e. Android smartphone or tablet) freely without instructions. A range of applications was installed and used. The participants were aged 5 to 11 years old (five children of 5 to 8 years old and three children of 8 to 11 years old). Among the participants, there were four boys and four girls, located in Europe and Asia.

5.2.3 Results

Baran collected about 36 hours of smartphone usage with 74954 interactions and 20 applications in total. The data analysis reveals that the children performed 49 touches, 85 scrolls, and with an average duration of 3 minutes and 52 seconds per application session. An application session is a period from initialising an application to when it is closed.

Figure 5.6 provides an overview of all applications (mostly in the gaming category) used by all participants and Figure 5.7 presents the amount of time spent on different applications by all participants. Ignoring the home screen application (*Home*), the figures show the popularity of *YouTube* among the children. It reports a significant amount of time spent on this application.

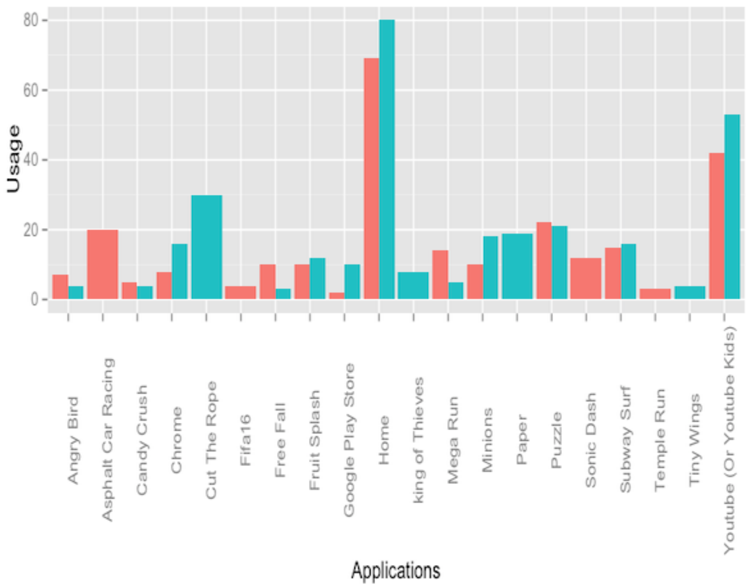

Figure 5.6: Application usage for all participants

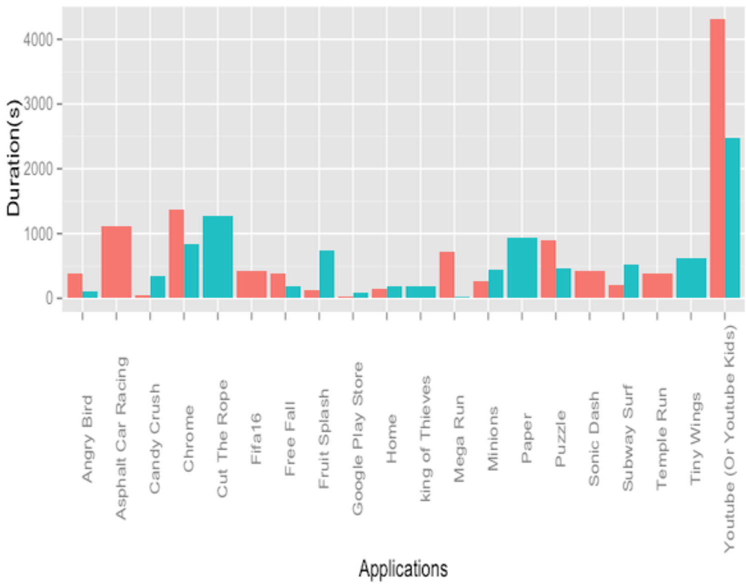

Figure 5.7: Total duration of using applications for all participants

Data analysis can show the differences and interests of children in different age groups. Figure 5.8 presents the duration of application usage for children in 5-8 years and 8-11 years old groups. It is beneficial to understand the child and their interests at a certain age. Figure 5.8 shows that younger children spend more time than older children on their devices. It also shows different interests among them. For instance, younger children are more interested in watching *YouTube* and using some particular applications such as *Tiny Wings* and *Angry Birds*. On the other hand, it shows that older children tend to use

Google Chrome, apparently because the younger group does not have enough ability to use it.

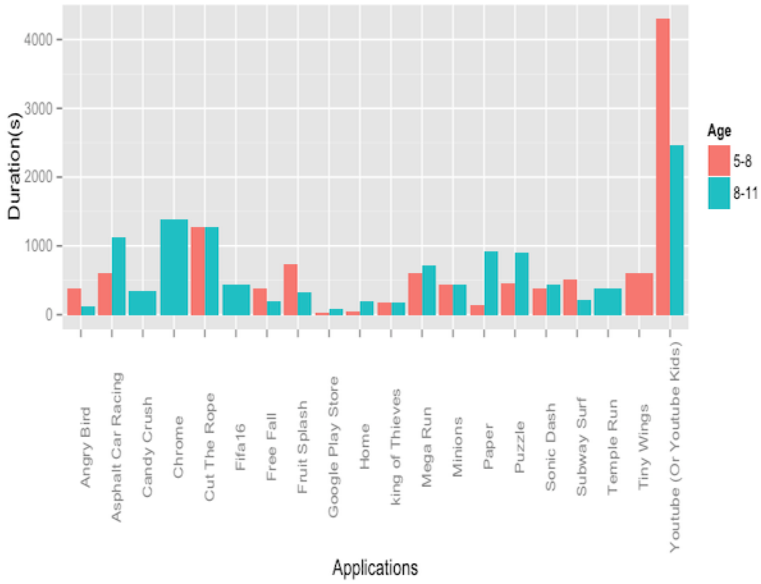

Figure 5.8: Total duration of using applications based on age

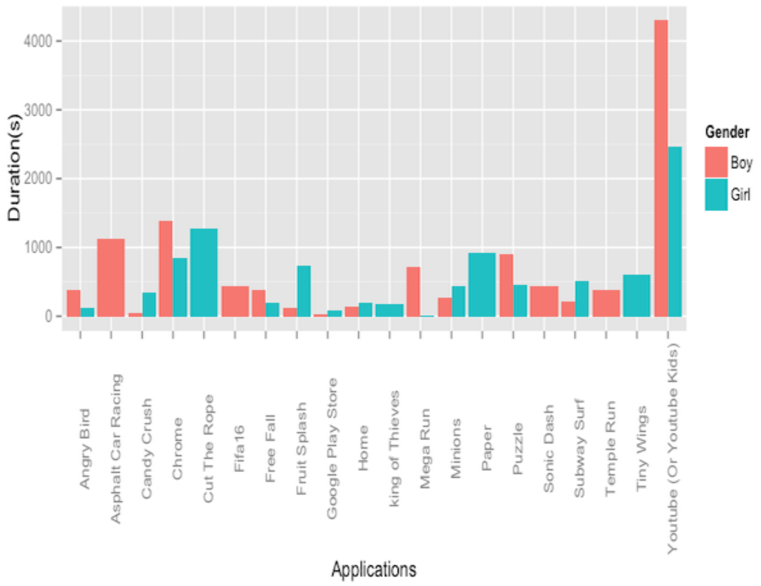

Figure 5.9: Total duration of using applications based on gender

Subway Surf is a game played more by the children aged 5-8 years old. When we investigated the information about the app, we realised that it is rated for the children aged 9+[22]. Here is the definition of the 9+ group: "9+ Applications in this category may contain mild or infrequent occurrences of cartoon,

[22]Subway Surfers Information Page on Apple App Store: https://goo.gl/9f0VEE

fantasy or realistic violence, and infrequent or mild mature, suggestive, or horror-themed content which may not be suitable for children under the age of 9"[23]. Our study shows children may tend to play applications not designed and rated for their age group. Using such a framework (e.g. Baran) can benefit application developers to know their users and design for them, and also help parents to ensure their children use appropriate application for their age.

It is also interesting to reflect on the gender of the children in the analysis; Figure 5.9 represents interesting statistics about the duration of the range of applications used by boys versus girls. For instance, boys tend to use *Chrome* and *YouTube* more than girls. The figure also reveals the popularity of racing games (e.g. *Asphalt*) among boys in contrast to the use of *Subway Surf*, *Fruit Splash*, and *Minions* among the girls. By considering different parameters (e.g. age, gender, and location) the data analysis can produce many different insights that help to understand the users. In this study, we do not focus on in-depth data analysis, as it is out of the scope of this work.

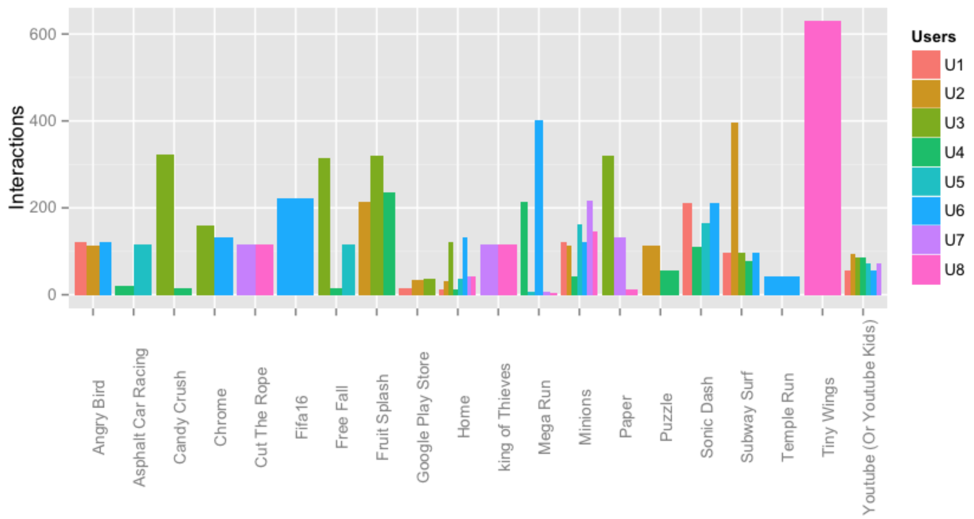

Figure 5.10: Children's interaction summary

Besides the duration and the usage of the applications in the data analysis, the interaction (e.g. Click, Touch, Scrolling) can also contribute to learning how a user experiences an application. Figure 5.10 presents the interaction summary of all eight children for different applications. *YouTube* is the most common app used by children. However, it is not very interactive due to the nature of watching video streams. *Tiny Wings* is used by one of the participants and is

[23] Apple Store Connect Help - App Ratings: https://goo.gl/FdeoZb

very interactive. A gender-based data analysis of the data can provide a high-level view of the interests of boys and girls, and how they experience different applications.

Figure 5.11 provides a detailed analysis of the application usage, duration, and interaction summary based on the participant's gender. For instance, *Tiny Wings*, *Mega Runs*, and *Fruit Splash* are the most interactive applications. *Tiny Wings* is used by girls only. *Mega Runs* is used by boys more than girls while, in contrast, *Fruit Splash* is used by girls more than boys. *Fruit Splash* is not a frequently-used application but is highly interactive as opposed to *YouTube*. Knowing that, helps the application developer to improve their applications. These kinds of analytics could be better accomplished in a broader experiment with more number of users and based on different parameters (e.g., gender, age, location).

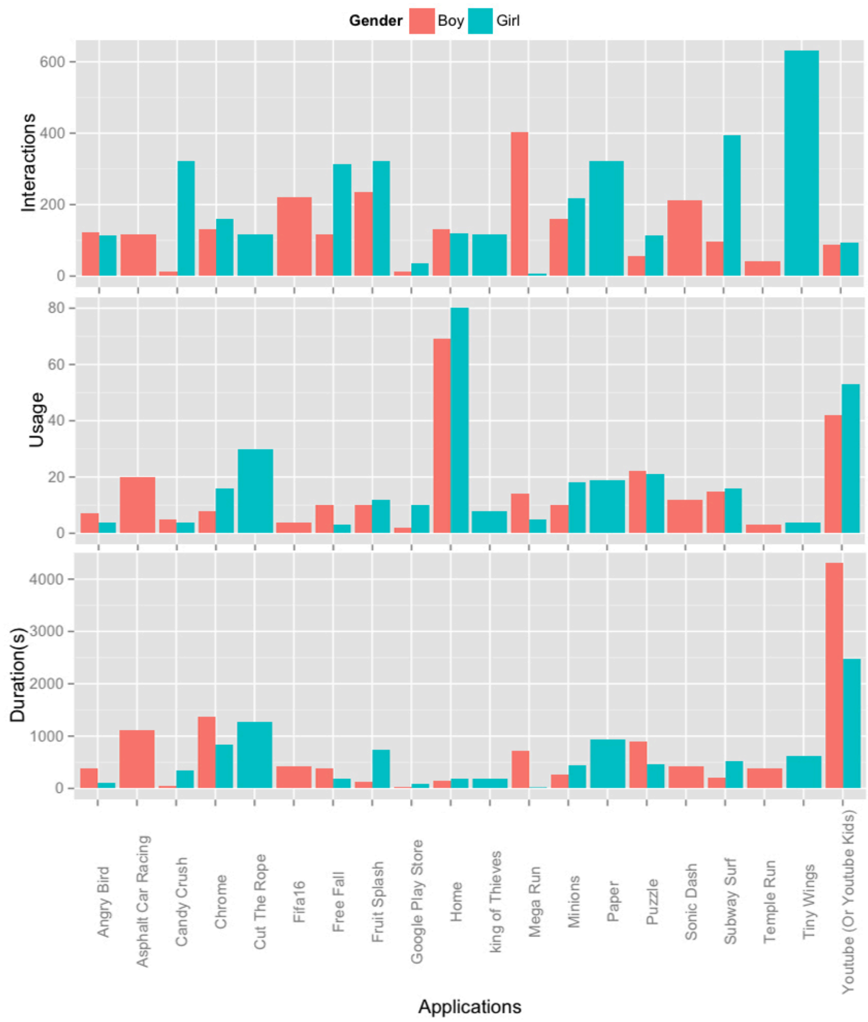

Figure 5.11: Application usage, duration, and interaction summary based on gender

5.2.3.1 Viewing results using the Interaction Profile

The Interaction Profile is an innovative graphical representation of user data (described previously in section 4.4.3). It provides several useful statistics in a concise graphical form. Figure 5.12 shows user-based interaction profiles of two of the participants (User1 and User3) regarding their smartphone usage.

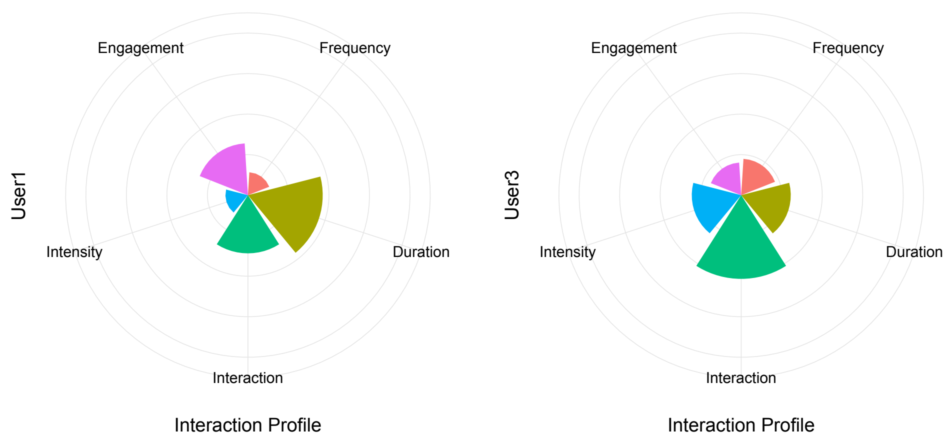

Figure 5.12: User-based interaction profile for User 1 vs. User 3

Looking at the Frequency property, User3 uses his/her smartphone more frequently than User1. On the other hand, the Duration property shows that User1 uses the smartphone for longer than User3. Bringing the Interaction property in, even though User1 spends more time on his/her smartphone, he/she interacts less than User3 who spends less time on his/her smartphone and interacts more. It can also be concluded by looking at the Intensity property, which shows that the activities of User3 are more intense than User1. Finally, the Engagement property shows that User1 is more engaged in using his/her smartphone than User3. All this information can be extracted from Figure 5.11, but it is easier to use the Interaction Profile as a comparison tool.

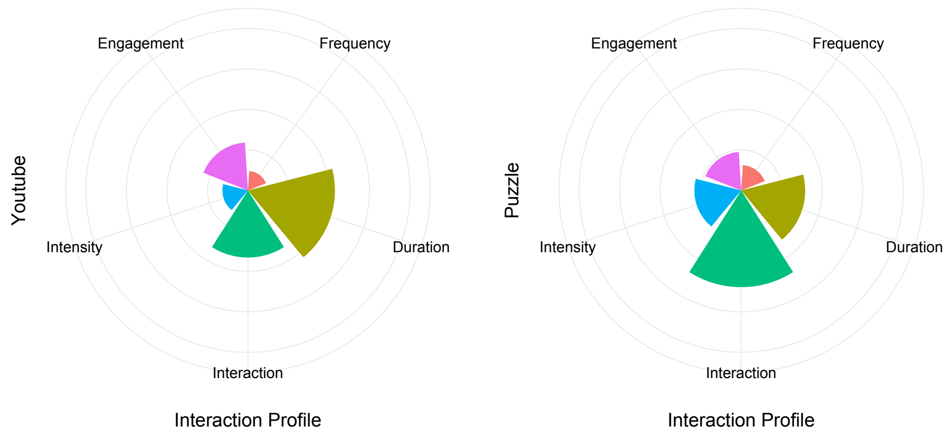

Figure 5.13: Activity-based interaction profile for YouTube vs. Puzzle

The Interaction Profile can be presented in many ways with various bases (time-based, user-based, and activity-based). Two activity-based Interaction Profiles for all of the participants using the *YouTube* application and the *Puzzle* game are presented in Figure 5.13. It shows that *Puzzle* is used for a slightly shorter time, more frequently, and is more interactive. *YouTube* is less interactive but used longer due to its nature of being a video-watching service, so it is more engaging and less intense.

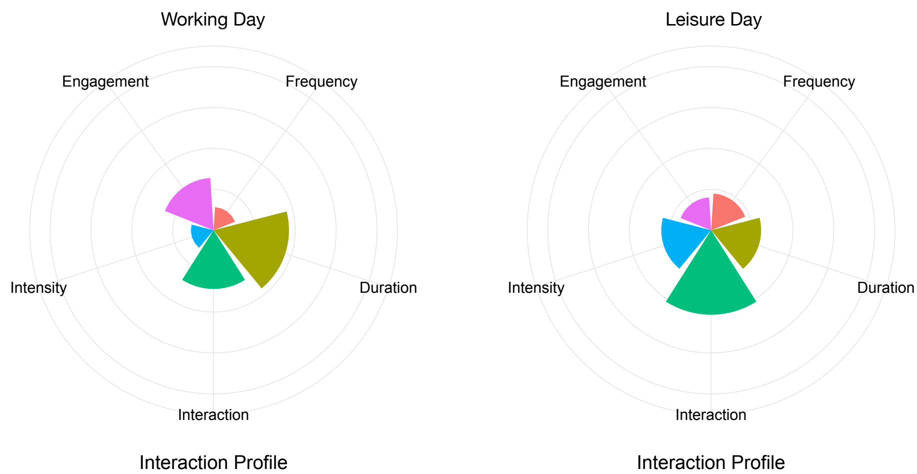

Figure 5.14: Time-based interaction profile for working days vs. leisure days

One other way of using the Interaction Profile as a comparison tool is to generate a time-based Interaction Profile for the same user, the same activity, and with the same period, but at a different time. As shown in Figure 5.14, two Interaction Profiles are presented for smartphone usage for the same user during working days and leisure days (weekend). It shows the user spends a more extended period on his/her smartphone, while he/she uses the smartphone less frequently during working days rather than leisure days. He/she performs more interactions during leisure days. The intensity metric shows his/her activities are more intense during leisure days, but the engagement metric reveals that he/she pays more attention to his/her smartphone during working days.

5.2.4 Summary

Children want to engage media-rich application on the Internet. Unfortunately, inappropriate information is also accessible. How children's behaviour is affected by technologies is unclear, and technology addiction is a concern for parents. In design, for children, the approach taken in child-centred design is to find out precisely (in an unobtrusive way) how a child uses technology and existing applications, and then use this as the starting point and reference point for subsequent design activities.

Baran enables parents, guardians, and researchers to analyse child use of technologies and discover what they are interested in and precisely show how they interact with them. Baran is significantly useful as it provides a monitoring service that works transparently for users so that they can interact with their digital devices in a natural environment without artificial constraints. It is demonstrated that a preliminary experiment can produce interesting elementary insights and initial results, which can support understanding children's digital activities. The Interaction Profile is an innovative way of comparing two entities. It provides an informative summary of interaction based analysis with its five useful metrics: Frequency, Duration, Interaction, Intensity, and Engagement.

5.3 Scenario C: an adaptive predictive assistant

One of the key features of the Baran framework is to support 3rd party services to contribute to providing services to the users. Baran is designed to protect user privacy and data. This scenario is designed as a proof-of-concept to demonstrate how the Baran system enables a 3rd party service to obtain a user's data (under user control) and assist him/her. Predictive assistive applications are studied [DGP14, NM16] and are beneficial in several areas such as recommender systems, adaptive services, and context-aware applications [LL14].

5.3.1 Experiment

In this scenario, an adaptive predictive assistant is designed to analyse a user's data (application usage) and assist the user by providing a list of possible application choices he/she likely wants to use next. A 3rd party service, Next-App, is developed to work with the Baran framework using the provided APIs.

The assistant application, Next-App, is predictive as it obtains a list of possible choices of the next applications using a predictive model, which is built from the history of application usage and the available context information. The assistant application is adaptive. It adapts itself to the current context and proactively predicts what applications the user is likely to want. In this experiment, the user agrees to use a 3rd party app (Next-App) on his/her smartphone and permits the 3rd party service to access some parts of his/her data (i.e. application usage) that is previously collected by the Baran framework. The user has control over choosing what part of the data is shared with who and for how long.

5.3.2 Architecture and implementation

Next-App is an Android application, implemented using Android Studio. The application is designed to provide a list of applications as a notification to a user. The application works transparently in the background and uses an adaptive predictive service, so it does not need to be open all the time, and the user can work with his/her smartphone.

The service architecture has four components: Data Handler, Pattern Recognizer, Rule Learner, and Recommender components (Figure 5.15). Once a user starts using Next-App, they have to enter their identification used in Baran so that the 3rd party service can request user data. Once Baran receives the request from the 3rd party service, the Baran permission management service ensures the required permission is in place. If not, the user receives a request from the Baran framework carrying the information about who requests to access his/her data, for how long they want to access it, and which level of access they are requesting. When the user provides Baran with the required permissions, then the Baran framework starts providing the 3rd party application with the user data under the permission criteria.

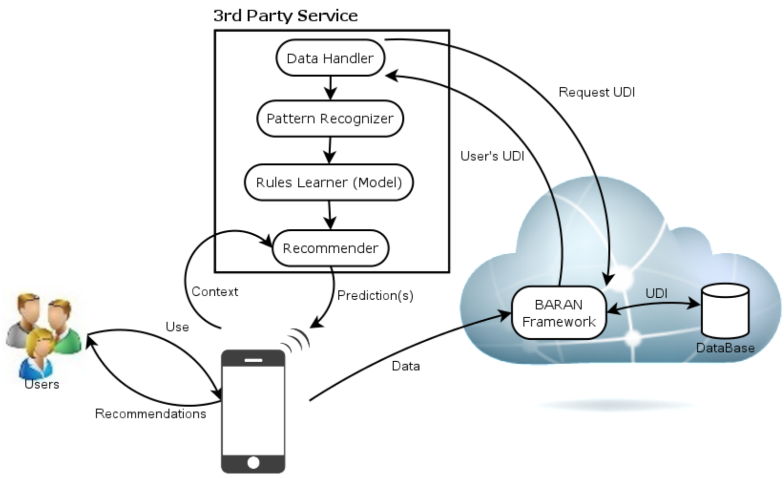

Figure 5.15: Next-App service cooperation with the Baran framework

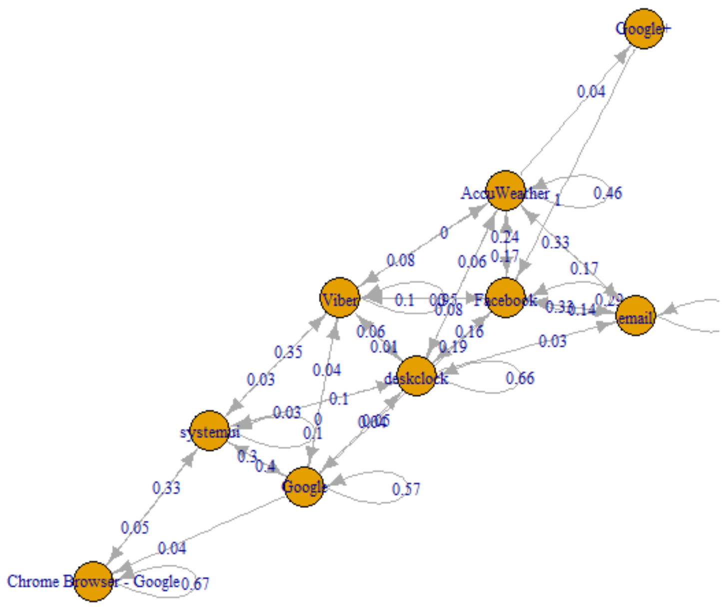

Figure 5.16: Markov model of application correlations

The Next-App application begins to request a user model (UDI) from the Baran framework. The Data Handler component periodically requests an up-to-date user model. The Pattern Recognizer component extracts patterns of how frequently the user uses the smartphone applications. Generally, this component analyses the user data and extracts the user habits of application usage based on different parameters (e.g. time, location, etc.). The Rule Learner component extracts rules from the history of application usage by considering the context

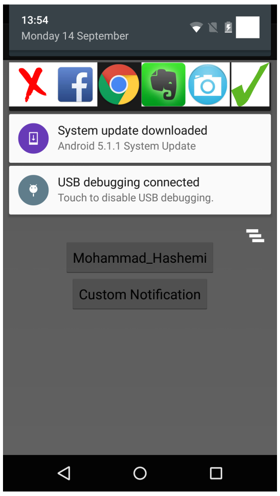

Figure 5.17: Next-App notification user interface showing a list of predictions

data as the observers of the classification. This component uses the Association Rule (AR) technique [Zha03, SRA+06] in order to find the correlations between different applications and the context data. The Rule Learner component then builds a predictive model accordingly. Figure 5.16 presents a Markov chain model of correlations between applications.

The Recommender component is a pro-active predictive service that uses the predictive model and predicts a list of possible choices of the next N applications the user is likely to use based on the context data (e.g. location, time). As the Next-App application is a notification-based service, it intelligently notifies the user with a list of recommendations, and by that we mean, it increases/decreases the frequency of generating predictions based on the device usage pattern in order to save battery life. It avoids unnecessary generation of predictions when the device is unlikely to be in use. The list (as shown in Figure 5.17) shows six icons of the possible choices. Four applications are predicted using the model, and two icons for evaluation purposes.

5.3.3 Evaluation

Six users participated in this experiment and used the Next-App service. Baran had previously collected two months of user data. Users are presented with the application user interface (Figure 5.17) where they can access the list of application icons and use the in-app rating service that provides a like and a dislike button in order to evaluate the accuracy of the predicted list.

Figure 5.18 presents the acceptance rate of the predictions, the positive (num-

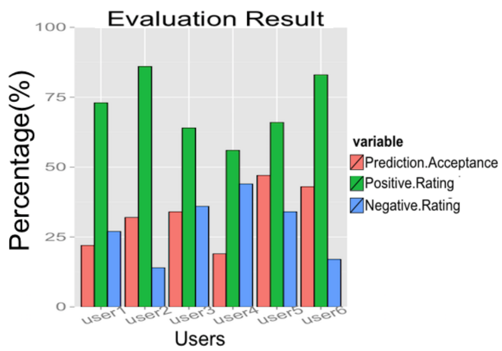

Figure 5.18: Service evaluation result

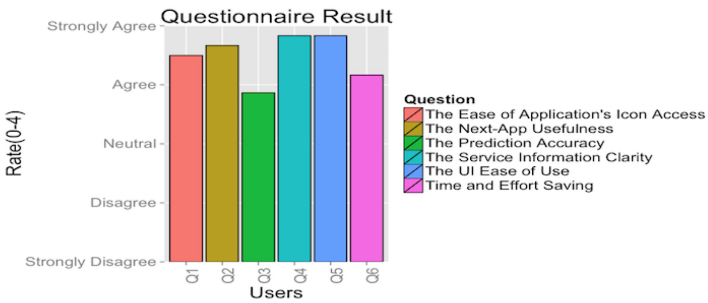

Figure 5.19: Service questionnaire result

ber of likes) and negative (number of dislikes) ratings that are recorded by the in-app rating service. It shows that, on average, the users take approximately 30% of the predictions. It also shows that the service obtains an excellent positive rating versus a negative rating. Besides, we provided a questionnaire to each user in order to obtain their opinions regarding the service presentation and its usefulness. Figure 5.19 shows a summary of the questionnaire feedback. It indicates positive feedback for the service presentation, usefulness, and prediction accuracy.

5.3.4 Summary

Next-App is a 3rd party service that predicts the next application(s) a user likely wants to use based on the current context. The Next-App service demonstrates the ease of implementing a 3rd party service based on the Baran framework and shows how analysing a user's interaction data can produce a useful user model that is the basis for a personalised and dynamic prediction service. In this scenario, we used the patterns of the pattern recognition service (Scenario A) to pro-actively predict the next application a user likely wants to use. At the time that a user is deemed to turn the device on and start using it, the algorithm predicts in a more frequent cycle, and vice versa, it predicts less frequently when the device is not supposed to be used to save battery life. The Next-App service has been implemented, evaluated by the users and found to be a useful service.

5.4 Conclusion

Baran is a fully working research-based framework and is useful in a wide range of case studies, three of which have been discussed. The framework provides the monitoring and data collection services, the communication mechanism for transferring data to the Baran cloud-based server, and the security component to ensure the user data is safely transferred, stored, and shared with 3rd parties. Baran provides data analysis services to enrich the quality of the data and extract high-level information from raw data. Baran securely shares data while complying with user privacy and data protection policies, under user control. In this chapter, three scenario-based case studies are presented that demonstrate the usefulness of the Baran framework.

The first case study was a self-monitoring use of technology. It is designed for an individual interested in monitoring his/her digital activities. A data collection service is installed on the user's smartphone and collects the user's interaction into a data model (UDI). The results of the data analysis services show the pattern of usage and activity classification. They assist a user in understanding their digital activities. This case study could be extended to a more comprehensive experiment that, in addition to reporting the statistics to the user, could also use the data in order to perform customised actions according to the statistics and reports. This scenario is applicable for setting usage limits, activity suggestion, or abnormal behaviour detection applications.

The second case study is designed for parents and guardians to monitor children. It can help them to increase their awareness of their child's use of technology. In this case study, eight children of different ages and genders have taken part in the experiment. We have collected about 36 hours of data from the children. The pattern of smartphone usage and the diversity of the children's activities, based on gender and age, have been reported. The result of this experiment is not only useful for guardians but also is relevant for researchers, child psychologists and application designers.

The third case study is designed to demonstrate how a 3rd party service can access a user's data under his/her control and how it provides the user with a better service that is personalised using the user data. This scenario includes all the necessary steps to be taken by a 3rd party service, a user, and the Baran framework, from sharing the user's data with a 3rd party to serving the user with the personalised service by the 3rd party. This proof of concept demonstrates that Baran can be safely extended with an external service, thus expanding the capability of the framework.

In conclusion, the Baran framework can be used to collect user data and digital activities. The diversity of how serviceable Baran is for individuals, parents and guardians, and possibly the scientific community are discussed and presented.

Chapter 6

Discussion and Conclusion

This chapter presents a summary of the thesis and important aspects of the work, along with a discussion of current challenges and future applications.

6.1 Thesis summary

In this age of connected digital devices, user digital activities are recorded by many devices, and many enterprises, under various levels of user control. Users are often unaware of how much digital data is created by their actions, of where the data resides, and how the data is used. The focus of this thesis is on challenging the current situation and providing a solution that reimagines the relationship between the user and their digital trail of information (i.e. the digital information from digital devices associated with the user). The Baran framework provides the user with full control over the collection, aggregation, storage, analysis and sharing of their data. Baran supports the integration of 3rd parties as providers of services, and also as consumers of user data, under user control, in order to provide helpful 3rd party services to the user.

In Chapter 2, existing user monitoring and modelling frameworks and related works were reviewed, and the challenges the researchers uncovered were identified. Chapter 3, firstly, introduces a conceptual model and extensible data structure, called the UDI (User Digital Imprint), which accommodates a variety of digital data from different digital devices (associated with a user) along with information derived from that data. This chapter also presents the design of Baran, a software framework that aims to address the deficiencies identified in existing work, and provide a novel solution (based on the UDI) to support the gathering, management, analysis, and sharing of this user data. The Baran framework allows the user to inspect and analyse their data and supports (under user control) the sharing of the data in order that 3rd parties can provide assistive services for the user. The framework is cloud-based and service-oriented, enabling the framework to make use of external services (e.g. a new machine learning service) and provide services for external entities (e.g. supplying some subset of user data to a smart home appliance). Thus, Baran is designed to be extensible with both user services and external services, where the sharing of user data with external entities is directly under user control on a per-use basis. The proposed conceptual models and software framework are implemented as a fully working research prototype software solution. Chapter 4 presents the implementation of Baran's components, services, the data structure (UDI), predictive model, and Interaction Profile. Three proof-of-concept case studies are presented in Chapter 5 in order to demonstrate aspects of Baran and how it can support different scenarios of use. The first scenario is one where Baran allows the user to gain insight into their own use of digital devices, through viewing summaries and analysis of their digital activities. The second scenario is one where a responsible carer (e.g. parent) assists a dependent individual (e.g. child) by partaking in the monitoring and reviewing of the individual's activities on their digital devices. The third scenario is one where Baran interacts with an external 3rd party service that provides smart assistance to the user. Baran shares a subset of user data with the 3rd party, and this 3rd party then leverages that data to provide a smart service, personalised for that user, at that time. The user controls which data, for how long, and with whom it is shared. A 3rd party service can obtain data dynamically over time, and dynamically adjust the personalised service to the user to reflect changing user digital behaviour. By gathering, analysing and making available comprehensive information about a user's digital interactions, Baran also enables the study of aspects of User Experience (UX), as might be of interest to product designers or UX researchers. Finally, this chapter (Chapter 6), presents a summary of the thesis and the framework along with a discussion of challenges and future applications of Baran.

6.2 Baran framework features

In this work, several research challenges were targeted to achieve the objectives of the study. In this section, the solutions that the Baran framework offers and how Baran addresses the research challenges are presented.

> **Research Challenge I:** How can user interactions with their various digital devices be monitored?

Users use many interactive digital devices including smartphone, smart home appliances, and health and fitness trackers; users are surrounded by many IoT sensors. These are the data sources that Baran can monitor and provide user-related measurements and information. The importance of monitoring user digital interactions has become apparent with recent functionality introduced by Google (offering Digital Wellbeing and Family Link) and Apple (offering Screen Time). Rather than a limited-scope, specific solution, Baran provides an extensible generic framework that covers a range of user digital interactions on a variety of devices and supports a wide range of applications, including 3rd party applications, where all data sharing is under explicit user control. Baran offers various data collection services for devices running Android, Windows, and Apple iOS operating systems.

> **Research Challenge II:** How can a user model be designed that provides a standard and comprehensive data structure and supports different data levels (data, information, and insights) and various data properties including unknown future data?

In Baran, data extensibility is addressed by the design of the UDI model and corresponding UDI data structure. The UDI supports data from a range of sources, and data at different levels (e.g. data derived from collected data by machine learning). Baran provides an extensible, service-oriented framework. New data collection services can be added for new devices. Baran can extend its existing data analysis component with new data analysis services to enrich the collected raw data. Moreover, in order to further enrich the user model, Baran enables 3rd party services to access user data (under explicit user control), and perform specialised analysis and return the results for integration into the UDI.

> **Research Challenge III:** What system architecture can support the discussed data analysis, data sharing and provision, and data collection services so that it offers the users full control on data? How can the proposed

framework let external parties contribute to its functionalities?

The Baran architecture is also designed to be generic. The cloud-side of the framework can be implemented using different cloud-based software providers (e.g. Amazon AWS and IBM BlueMix). The client-side of the framework can be implemented using various programming languages for various platforms (e.g. Android, Windows and iOS). The framework offers essential services and utilities required for data collection and storage. In addition to those, the framework also provides data analysis services that can enrich the data quality by extracting insights from raw data. The data analysis services leverage the modelling technologies and methods to learn patterns from the data and use machine learning algorithms to produce prediction models.

Baran also enables external 3rd party services to make use of user data in order to provide personalised 3rd party services for the user. (For example, a smart home appliance manufacturer could offer a better personalised service to the user when it has access to the Baran user model.) All sharing of data with external parties is under explicit user control and fully complies with user privacy and data protection policies such as EU General Data Protection Regulation "GDPR" [PU16]. Included in the Baran framework are a number of general utility services that support many scenarios. These include registration, communication and security services.

6.3 Challenges

In this section, some challenges encountered in this work are discussed.

6.3.1 Additional data collection services

To support the current scenarios, involving detailed monitoring of user digital activities, Baran provides just Android and Windows data collection services. This has confined the scope of the experiments to these devices. While the Android operating system supports comprehensive data collection service, the Apple operating system introduces limitations for data collection services. This is because the Apple OS sandboxes each application's data and does not allow an application to access another's data. This prevents the collection service capturing user interactions within user applications on Apple devices (e.g. iPhone and iPad). Thus, while high-level use (e.g. starting time and ending time) of an application on an Apple device can be monitored, it is not currently possible to gather low-level user interactions (e.g. where the user touches, what the user types, and what information the user sees). This constrains Baran data collection until Apple allows more low-level access.

Further challenges exist in providing data collection services for IoT devices and commercial smart appliances. IoT devices are typically resource constrained and do not allow the uploading and running of external code (such as a data collection service) on these devices. Similarly, commercial smart appliances will have their own data collection service but are very unlikely to allow external code to run on their appliances. For both IoT and smart appliances, integration depends on the existence of a public API where the device manufacturer provides data for external users. For Baran, therefore, the existence of the API and the availability of suitable data through the API may provide challenges.

6.3.2 Trust

Trust is an essential factor for users when considering using a software service. Users are concerned about breaches of data privacy and other security issues [YWY+17]. A high level of trust is of particular importance when a new digital service gathers user data.

A 2018 international survey of smartphone users reports that 57% say that personal data collection is a risk to them and that 68% think it is important to know how their data is being used [MEF18]. In the case studies presented in Chapter 5, participants were using their own personal devices. Despite the explanations to users of the functionality of Baran and its security mechanisms, some still expressed concern about possible access to their personal information on the devices. Thus, gaining user trust will remain an ongoing challenge for Baran. A 2017 survey [MEF17] reports that the top factors that contribute to a user trusting an app are:

- A clear and simple privacy statement (33%);
- Brand recognition (32%);
- Positive media coverage/reviews (29%); and
- Recommendation from friends or family (27%).

6.3.3 Integration of 3rd party services

There is a challenge in integrating commercial 3rd party services as commercial enterprises at present often do their own data collection when users use their products (e.g. smart TVs monitor the viewing habits of viewers). Existing developers of software apps or smart appliances, therefore, may be reluctant to stop doing their own data collection, and, instead, use a Baran data collection service despite the fact that Baran offers additional value in that Baran can also share with the 3rd party, if the user allows, a much richer user model so that the 3rd party can offer the user better personalised services. Developers of new products may be less reluctant as, for minimal effort, they would gain access to a robust framework that could support their user services.

6.3.4 Additional security techniques

Baran already provides security mechanisms such as symmetric and asymmetric encryption, and cryptographic hashing. Many other security techniques, however, can be implemented to improve security and further protect user data. These additional mechanisms include digital signatures, digital certificates, and key wrapping. More sophisticated mechanisms can provide greater reassurance for the user. For instance, Apple uses a highly-secure mechanism to encrypt the user data, where even Apple can not see the data. Secure enclave is a co-processor chip that can store ephemeral keys on a user device. The Apple OS can request the enclave to decrypt some data, but it never sees the actual key. A similar mechanism for Baran, as well as improving security, would also assist by enhancing trust in the users of Baran.

6.4 Future applications of Baran

Baran, can already support more extensive solutions that just follow the same pattern as the scenarios described in Chapter 5. In this section, other future applications of the Baran framework are discussed.

6.4.1 A family e-life application

As well as the basic monitoring an individual, the Baran framework can be the basis for a family activity monitoring application. Baran can be extended with a service that integrates the personal devices of all family members, and, with some control over levels of privacy, it can provide a family e-life monitoring and analysis service. This service can analyse everyday family activities, provide insights to the family regarding their use of digital technologies, and allow the family to introduce family practices that support better use of digital devices and allow space for family time. Encouraging family communication and cooperation, and reclaiming family time by analysing content and usage across all family devices can be an innovative way of using Baran.

6.4.2 Commercial product evaluation

At present the usual way an enterprise evaluates how users use their product, and how satisfied their users are, is through either very brief on-line surveys or very detailed in-person surveys. One particular concern is to evaluate user response when a new feature is added to a product. Baran can provide a much more efficient and comprehensive solution. If the company uses a Baran data collection service or provides an API for Baran, it will enable the company to exploit all of Baran services, such as for machine learning, visualization, security, communication, as well as the possibility of, with the user's permission, integrating their data with the UDI for that user.

For example, a smart toothbrush manufacturer could use Baran to analyse how and when a user uses their toothbrush, and would gain an even better analysis if information from Baran's UDI user model is also used to provide further context for the use of the toothbrush. When a new feature is added to the smart toothbrush, the pattern of use can be immediately analysed to determine how the user has responded to the new version of the product.

6.4.3 Supporting scientific experiments

Laboratory experiments are often used by scientists (and product designers) to evaluate human response and behaviour with respect to technology, in particular with regard to user interfaces, and often as a part of investigating UX. As mentioned with regard to designing technology for children, Baran provides a solution where the evaluation can take place during the normal use of the technology in everyday life. Provided the user grants permission, Baran can share an anonymous version of whatever part of the UDI is relevant to the scientific experiments. Baran supports a scientist to conduct a much more comprehensive experiment in a more natural environment, and with many more users than a similar laboratory experiment.

6.4.4 Supporting an integrated smart environment

The expansion in smart appliances, personal health sensors and the internet of things (IoT) means that, in the future, there will be more connected devices in the environment acting as data sources, supplying more extensive contextual data. While the current situation is that most of these devices are separate entities, often with proprietary interfaces, that cannot easily co-operate, future users will demand more integration to support more efficient and effective applications. Baran can be at the heart of an integrated smart environment by combining the comprehensive monitoring of user activities across a range of personal devices with the rich contextual data available from other devices such as smart appliances, health monitors and IoT devices.

The following describe some specific future scenarios where Baran can support useful user applications.

6.4.4.1 Home appliance assistive services

Baran allows smart home appliances, e.g. Samsung smart appliances[24] and Xiaomi smart home appliances[25], to contribute their data to the user model (UDI), following the approach outlined in Section 4.2.1.4.2. There are various smart home automation frameworks, e.g. Samsung Smart Home Cloud[26] and Smappe[27], that allow users to monitor their home appliances and usage of the electricity, water, air conditioning, heat pump and boiler, solar energy, home battery, smart devices, and home theatre. These frameworks enable users to

[24]Samsung Smart Home Appliances: https://goo.gl/4U4pqA
[25]Xiaomi smart home: https://goo.gl/2vXCuy
[26]Samsung Smart Home Cloud: https://goo.gl/ej48Rn
[27]Smappee (Analyse, Control, and Save Energy): https://goo.gl/f2GFX1

not only gain insights and better understand their energy consumption but also to control it.

6.4.4.2 Health and fitness monitoring

Wearable sensors enable users to continuously monitor, collect, and wirelessly broadcast their data to a server at a distance [CL18].Even though the most impact of wearable technology is likely to be in health and fitness, the technology can also be important in sports and entertainment, disabilities, education, finance, transportation, and enterprise [WP17]. Baran implements an extensible framework that allows users to connect their monitoring devices (e.g. fitness and health monitoring device) to their Baran account. Baran can aggregate user data from the user monitoring services and, using their provided APIs, integrate user data to their user model (UDI). Other than these monitoring devices, there are various data recording applications and frameworks, such as Sony Lifelog[28], Google Fit SDK[29], and Fitbit SDK[30], which allow users to aggregate their data from associated company products. Baran services can also be extended to connect to these services and add the device data to make a more comprehensive and useful user model.

6.4.4.3 Smart appliance predictive assistant

Baran already supports 3rd party assistive services (Chapter 5, Section 5.3) that cooperate with Baran to provide personalised services based on the user data. Another example of how the UDI model could be used to ease a user's life is presented here.

Bob has two smart coffee makers (one at home and one at work) and drinks coffee at various times during the day. The coffee maker contributes to building Bob's UDI with the information such as the dates and times and the type of coffee Bob makes. A predictive assistant service can be designed to discover the pattern of days, times and types of coffee that Bob usually drinks, along with any other factors in the UDI that may affect the pattern of coffee drinking.

[28]Sony Lifeloghttps Program: https://goo.gl/1yF5Mh
[29]Google Fit SDK: https://goo.gl/osrdxu
[30]Fitbit SDK: https://goo.gl/ucM6Kz

Using Bob's drinking coffee patterns, a smart assistant can be scheduled to send the correct command to the correct coffee maker in the correct location at the correct time: a "Make Medium Roast Americano" command when at work, mid-morning; a "Make Decaf Latte" when at home at night. The assistive service can require confirmation of the "Make Coffee" command, so that Bob can cancel the usual night-time option when he intends working late at home to meet a deadline.

6.5 Conclusion

One of the challenges of modern life for individuals and society is the transition to a lifestyle dominated by the digital world. The gathering and use of personal digital data is a concern to users, and the effect of unhealthy over-engagement in the digital world is a concern for society. Baran offers comprehensive monitoring of a user's digital activities, and the storing and exploitation of this data, under strict user-control. The generic, service-oriented architecture, UDI model and data structure, Interaction Profile infographic, and 3rd party service integration are all notable aspects of the Baran framework.

Baran implements a solution that is not allied to current commercial practice where user data is gathered and exploited by enterprises, often with little user knowledge or control. This thesis aims to demonstrate a coherent user-centred alternative where the user regains control of their data and where helpful user services can be built on top of a rich user model, under user control.

Appendix A

References

References

[AAJH16] Iman Nur Nabila Ahmad Azahari, Wan Fatimah Wan Ahmad, Zulikha Jamaludin, and Ahmad Sobri Hashim. The design of mobile social application for children with autism. In *2016 3rd International Conference on Computer and Information Sciences, ICCOINS 2016 - Proceedings*, pages 547–552. IEEE, aug 2016.

[AAS18] Faisal Alsubaei, Abdullah Abuhussein, and Sajjan Shiva. An Overview of Enabling Technologies for the Internet of Things. In *Internet of Things A to Z*, pages 77–112. John Wiley & Sons, Inc., Hoboken, NJ, USA, may 2018.

[AI09] Asier Aztiria and Alberto Izaguirre. Learning about preferences and common behaviours of the user in an intelligent environment. *Volume 3: Behaviour Monitoring and Interpretation - BMI*, pages 289 – 315, 2009.

[AIB+10] Asier Aztiria, Alberto Izaguirre, Rosa Basagoiti, Juan Carlos Augusto, and Diane J. Cook. Automatic modeling of frequent user behaviours in intelligent environments. *Proceedings - 2010 6th International Conference on Intelligent Environments, IE 2010*, pages 7–12, 2010.

[AKE18] Gema Syahidan Akbar, Emil R. Kaburuan, and Veronikha Effendy. User interface (UI) design of scheduling activity apps for autistic children. In *Proceedings of the 2017 International Conference on Orange Technologies, ICOT 2017*, volume 2018-January, pages 129–133. IEEE, dec 2018.

[AKS14] Suha AlAwadhi, Jay P. Kesan, and Marko M. Skoric. Introduction to social media & social networking and government minitrack.

In *Proceedings of the Annual Hawaii International Conference on System Sciences*, page 2220. IEEE, jan 2014.

[AMV+17] Camila Alvarado, Roberto Munoz, Rodolfo Villarroel, Oscar Acunã, Thiago S. Barcelos, and Carlos Becerra. ValpoDijo: Developing a software that supports the teaching of chilean idioms to children with autism spectrum disorders. In *12th Latin American Conference on Learning Objects and Technologies, LACLO 2017*, volume 2017-January, pages 1–4. IEEE, oct 2017.

[Ann18] App Annie. The average smartphone user accessed close to 40 apps per month in 2017, 2018. `https://www.appannie.com/en/insights/market-data/apps-used-2017/`, Accessed: 2019.02.29.

[ANS15] M.V. Azghandi, Ioanis Nikolaidis, and Eleni Stroulia. Indoor sensor placement for diverse sensor-coverage footprints. *SENSORNETS 2015 - 4th International Conference on Sensor Networks, Proceedings*, pages 25–35, 01 2015.

[App18] Apple. iOS 12 introduces new features to reduce interruptions and manage Screen Time, 2018. `https://www.apple.com/uk/newsroom/2018/06/ios-12-introduces-new-features-to-reduce-interruptions-and-manage-screen-time/`, Accessed: 2019.02.27.

[APSV08] Amal Achaibou, Gilles Pourtois, Sophie Schwartz, and Patrik Vuilleumier. Simultaneous recording of EEG and facial muscle reactions during spontaneous emotional mimicry. *Neuropsychologia*, 46(4):1104–13, mar 2008.

[AT08] Gediminas Adomavicius and Alexander Tuzhilin. Context-aware recommender systems. *Proceedings of the 2008 ACM conference on Recommender systems - RecSys '08*, page 335, 2008.

[Bal17] Alipta Ballav. Predictive user experience, 2017. `https://www.uxmatters.com/mt/archives/2017/06/predictive-user-experience.php`, Accessed: 2015.09.04.

[BB79] Richard R. Burton and John Seely Brown. An investigation of computer coaching for informal learning activities. *International Journal of Man-Machine Studies*, 11(1):5–24, jan 1979.

[BCKO15] Linas Baltrunas, Karen Church, Alexandros Karatzoglou, and Nuria Oliver. Frappe: Understanding the Usage and Perception of Mobile App Recommendations In-The-Wild. *CoRR*, abs/1505.0, 2015.

[BG07] Gordon Bell and Jim Gemmell. A Digital Life. *Scientific American*, 296:58–65, 2007.

[BLG+12] Senaka Buthpitiya, Faisal Luqman, Martin Griss, Bo Xing, and Anind K Dey. Hermes – A Context-Aware Application Development Framework and Toolkit for the Mobile Environment. In *Advanced Information Networking and Applications*, 2012.

[Blo05] Ned Block. Action in Perception by Alva Noë. *The Journal of Philosophy*, 102(5):259–272, 2005.

[BLPR11] Linas Baltrunas, Bernd Ludwig, Stefan Peer, and Francesco Ricci. Context relevance assessment and exploitation in mobile recommender systems. *Personal and Ubiquitous Computing*, 16(5):507–526, jun 2011.

[BR10] Pradipta Biswas and Peter Robinson. A brief survey on user modelling in hci. In *Proc. of the International Conference on Intelligent Human Computer Interaction (IHCI) 2010*, 2010.

[CBBP16] G. Carutasu, M. A. Botezatu, C. Botezatu, and M. Pirnau. Cloud computing and windows azure. In *2016 8th International Conference on Electronics, Computers and Artificial Intelligence (ECAI)*, pages 1–6. IEEE, jun 2016.

[CDH14] José Creissac Campos, Gavin Doherty, and Michael D. Harrison. Analysing interactive devices based on information resource constraints. *International Journal of Human Computer Studies*, 72(3):284–297, mar 2014.

[CFJ03] Harry Chen, Timothy W Finin, and Anupam Joshi. Using OWL in a Pervasive Computing Broker. *Proceedings of the Workshop on Ontologies in Agent Systems (OAS 2003)*, pages 9–16, 2003.

[CL18] Armando Cruz and Jose P. Lousado. A survey on wearable health monitoring systems. In *Iberian Conference on Information Systems and Technologies, CISTI*, volume 2018-June, pages 1–6. IEEE, jun 2018.

[CLWA13] Shuwei Chen, Jun Liu, Hui Wang, and Juan Carlos Augusto. A Hierarchical Human Activity Recognition Framework Based on Automated Reasoning. *2013 IEEE International Conference on Systems, Man, and Cybernetics*, pages 3495–3499, 2013.

[CNS+04] Bhaskar Mehta Claudia Niederée, Avaré Stewart, Claudia Niederée, Avaré Stewart, Bhaskar Mehta, and Matthias Hemmje. A multi-dimensional, unified user model for cross-system personalization. *Proceedings of the AVI . . .* , 2004.

[CSR+16] Camille Cobb, Samuel Sudar, Nicholas Reiter, Richard Anderson, Franziska Roesner, and Tadayoshi Kohno. Computer Security for Data Collection Technologies. *Ictd*, 2016.

[DAS09] Anind K Dey, Gregory D Abowd, and Daniel Salber. A Conceptual Framework and a Toolkit for Supporting the Rapid Prototyping of Context- Aware Applications. *Human Computer Interaction*, 16(September 2014):97–166, 2009.

[DCC+11] Aiden R. Doherty, Niamh Caprani, Ciarán Ó Conaire, Vaiva Kalnikaite, Cathal Gurrin, Alan F. Smeaton, Noel E. O'Connor, and Noel E. O'Connor. Passively recognising human activities through lifelogging. *Computers in Human Behavior*, 27(5):1948–1958, 2011.

[DFAB04] Alan Dix, Janet Finlay, Gregory Abowd, and Russell Beale. *Human-computer interaction*. Prentice hall, 2004.

[DGP14] Trinh Minh Tri Do and Daniel Gatica-Perez. Where and what: Using smartphones to predict next locations and applications in daily life. *Pervasive and Mobile Computing*, 12:79–91, June 2014.

[DM09] Suparna De and Klaus Moessner. A framework for mobile, context-aware applications. *2009 International Conference on Telecommunications*, pages 232–237, 2009.

[DM11] Christoffer Davidsson and Simon Moritz. Utilizing implicit feedback and context to recommend mobile applications from first use. In *Proceedings of the 2011 Workshop on Context-awareness in Retrieval and Recommendation - CaRR '11*, pages 19–22, New York, New York, USA, feb 2011. ACM Press.

[Dou04] Paul Dourish. What we talk about when we talk about context. *Personal and Ubiquitous Computing*, 8(1):19–30, feb 2004.

[DWC10] Tharam Dillon, Chen Wu, and Elizabeth Chang. Cloud Computing: Issues and Challenges. *2010 24th IEEE International Conference on Advanced Information Networking and Applications*, pages 27–33, 2010.

[FB04] Jodi Forlizzi and Katja Battarbee. Understanding experience in interactive systems. *Proceedings of the 2004 conference on Designing interactive systems processes, practices, methods, and techniques - DIS '04*, page 261, 2004.

[FČ18] Petr Filip and Lukáš Čegan. Webalyt: Implemetation of architecture for capturing web user behaviours with feedback propagation. In *2018 28th International Conference Radioelektronika, RADIOELEKTRONIKA 2018*, pages 1–5. IEEE, apr 2018.

[FCFRP12] Filipe Felisberto, Nuno Costa, Florentino Fdez-Riverola, and António Pereira. Unobstructive Body Area Networks (BAN) for efficient movement monitoring. *Sensors (Basel, Switzerland)*, 12(9):12473–88, 2012.

[Fis01] Gerhard Fischer. User modeling in human-computer interaction. *User Modeling and User-Adapted Interaction*, 11(1-2):65–86, 2001.

[FL13] L. B. Ruiz F. A. Silva, T. R. M. B. Silva and A. A. F. Loureiro. ConProVA: A smart context provisioning middleware for VANET applications. In *IEEE Vehicular Technology Conference*, pages 1–5. IEEE, jun 2013.

[FOFH15] Adam Chęć Ferreira, Dominika Olczak, Tiago Fernandes, and Hugo. Physiological Computing Gaming - Use of Electrocardiogram as an Input for Video Gaming. In *Proceedings of the 2nd International Conference on Physiological Computing Systems*, pages 157–163. SCITEPRESS - Science and and Technology Publications, 2015.

[GAJI14] Cathal Gurrin, Rami Albatal, Hideo Joho, and Kaori Ishii. A privacy by design approach to lifelogging. *Digital Enlightenment Yearbook 2014*, pages 49–73, 2014.

[GBL06] Jim Gemmell, Gordon Bell, and Roger Lueder. MyLifeBits: a personal database for everything. *Communications of the ACM*, 49:88–95, 2006.

[Gil19] Ben Gilbert. There's a simple reason your new smart tv was so affordable: It's collecting and selling your data, 2019. `https://www.businessinsider.com/smart-tv-data-collection-advertising-2019-1?r=US&IR=T`, Accessed: 2019.02.29.

[GK16] Ankit Grover and Banpreet Kaur. A framework for cloud data security. In *2016 International Conference on Computing, Communication and Automation (ICCCA)*, pages 1199–1203. IEEE, apr 2016.

[GLB03] Jim Gemmell, Roger Lueder, and Gordon Bell. The MyLifeBits lifetime store. *Proceedings of the 2003 ACM SIGMM workshop on Experiential telepresence ETP 03*, pages 80–83, 2003.

[GM10] Andrea Girardello and Florian Michahelles. AppAware: Which Mobile Applications Are Hot? In *Proceedings of the 12th international conference on Human computer interaction with mobile devices and services - MobileHCI '10*, page 431, New York, New York, USA, sep 2010. ACM Press.

[Goo18] Google. Digital Wellbeing by Google, 2018. `https://wellbeing.google`, Accessed: 2019.02.27.

[GPM17] Sheetal Girase, Varsha Powar, and Debajyoti Mukhopadhyay. A user-friendly college recommending system using user-profiling and matrix factorization technique. In *Proceeding - IEEE International Conference on Computing, Communication and Automation, ICCCA 2017*, volume 2017-Janua, pages 1–5. IEEE, may 2017.

[GSD14] Cathal Gurrin, Alan F. Smeaton, and Aiden R. Doherty. LifeLogging: Personal Big Data. *Foundations and Trends in Information Retrieval*, 8(1):1–125, 2014.

[HH14] Mohammad Hashemi and John Herbert. UIXSim: A User Interface Experience Analysis Framework. In *2014 Fifth International Conference on Intelligent Systems, Modelling and Simulation*, pages 29–34, Langkawi, Malaysia, January 2014. IJSSST.

[HH15] Mohammad Hashemi and John Herbert. Baran: An Interaction-Centred User Monitoring Framework. In *International Conference on Physiological Computing Systems 2015*, pages 52–60, feb 2015.

[HH16a] Mohammad Hashemi and John Herbert. A Next Application Prediction Service Using the BaranC Framework. *2016 IEEE 40th Annual Computer Software and Applications Conference (COMPSAC)*, pages 519–523, 2016.

[HH16b] Mohammad Hashemi and John Herbert. A pro-active and dynamic prediction assistance using BaranC framework. In *IEEE/ACM International Conference on Mobile Software Engineering and Systems (MOBILESoft)*, pages 269–270, Austin, TX, USA, 2016. ACM Press.

[HH16c] Mohammad Hashemi and John Herbert. A Service-Oriented User Interaction Analysis Framework Supporting Adaptive Applications. *2016 IEEE 40th Annual Computer Software and Applications Conference (COMPSAC)*, pages 546–551, 2016.

[HH16d] Mohammad Hashemi and John Herbert. Child-centred design supported by comprehensive child application use analysis. *ACM SIGCHI Conference on Interaction Design and Children 2016*, pages 707–712, 2016.

[HH16e] Mohammad Hashemi and John Herbert. User interaction monitoring and analysis framework. In *2016 IEEE/ACM International Conference on Mobile Software Engineering and Systems (MOBILESoft)*, pages 7–8, Austin, TX, USA, 2016. ACM Press.

[HHW+16] Emir Husni, Galuh Boy Hertantyo, Daniel Wahyu Wicaksono, Faisal Candrasyah Hasibuan, Andri Ulus Rahayu, and Muhamad Agus Triawan. Applied Internet of Things (IoT): Car monitoring system using IBM BlueMix. In *2016 International Seminar on Intelligent Technology and Its Applications (ISITIA)*, pages 417–422. IEEE, jul 2016.

[HMB12] Negar Hariri, Bamshad Mobasher, and Robin Burke. Context-aware music recommendation based on latenttopic sequential patterns. In *Proceedings of the sixth ACM conference on Recommender systems - RecSys '12*, page 131, New York, New York, USA, sep 2012. ACM Press.

[HNBR97] Richard Hull, Philip Neaves, and James Bedford-Roberts. Towards situated computing. In *Proceedings of the 1st IEEE International Symposium on Wearable Computers*, ISWC '97, pages 146–, Washington, DC, USA, 1997. IEEE Computer Society.

[HWCL14] Han Hu, Yonggang Wen, Tat-Seng Chua, and Xuelong Li. Toward Scalable Systems for Big Data Analytics: A Technology Tutorial. *IEEE Access*, 2:652–687, 2014.

[IHSH06] William Ickes, Renee Holloway, Linda L. Stinson, and Tiffany Graham Hoodenpyle. Self-monitoring in social interaction: The centrality of self-affect, jun 2006.

[Jou12] Moyra Journeaux. Introducing Research and Evidence-Based Practice for Nurses. *Journal of Perioperative Practice*, 22(5):145–145, 2012.

[KBC+12] Alexandros Karatzoglou, Linas Baltrunas, Karen Church, B Matthias, #246, and Hmer. Climbing the app wall: enabling mobile app discovery through context-aware recommendations. *Proceedings of the 21st ACM international conference on Information and knowledge management*, pages 2527–2530, 2012.

[KHW06] Alexander Kr, Dominik Heckmann, and Wolfgang Wahlster. SPECTER : Building , Exploiting , and Sharing Augmented Memories. *Proc. of KSEL06, Knowledge Sharing for Everyday Life*, 2006.

[KKKP08] Hyoungnyoun Kim, Ig Jae Kim, Hyoung Gon Kim, and Ji Hyung Park. Adaptive modeling of a user's daily life with a wearable sensor network. *Proceedings - 10th IEEE International Symposium on Multimedia, ISM 2008*, pages 527–532, 2008.

[Kon16] Divya K. Konoor. Auditing in Cloud Computing Solutions with OpenStack. In *2016 IEEE International Conference on Cloud Computing in Emerging Markets (CCEM)*, pages 176–176. IEEE, oct 2016.

[KR11] Marius Kaminskas and Francesco Ricci. Location-adapted music recommendation using tags. In *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, volume 6787 LNCS, pages 183–194. Springer Berlin Heidelberg, 2011.

[KR18] Maria E. Kudryavtseva and Mira B. Rotanova. The problem of sense creation in interpersonal communication of youth in the digital age. In *Proceedings of the 2018 IEEE Communication Strategies in Digital Society Workshop, ComSDS 2018*, volume 2018-Janua, pages 33–35. IEEE, apr 2018.

[KRH+15] Eija Kaasinen, Virpi Roto, Jaakko Hakulinen, Tomi Heimonen, Jussi P. P. Jokinen, Hannu Karvonen, Tuuli Keskinen, Hanna Koskinen, Yichen Lu, Pertti Saariluoma, Helena Tokkonen, and Markku Turunen. Defining user experience goals to guide the design of industrial systems. *Behaviour & Information Technology*, 34(10):976–991, oct 2015.

[LCJ+18] Juzheng Liu, Jing Chen, Hanjun Jiang, Wen Jia, Qingliang Lin, and Zhihua Wang. Activity Recognition in Wearable ECG Monitoring Aided by Data ˜ a ˜ r. In *2018 IEEE International Symposium on Circuits and Systems (ISCAS)*, pages 3–6. IEEE, may 2018.

[LEM+17] M. Lips, E. Eppel, H. McRae, L. Starkey, A. Sylvester, P. Parore, and L. Barlow. Understanding children's use and experienec with digital technologies. Technical Report June, Victoria University of Wellington, 2017.

[LKL15] Wen-Hwa Liao, Ssu-Chi Kuai, and Yu-Ren Leau. Auto-scaling Strategy for Amazon Web Services in Cloud Computing. In *2015 IEEE International Conference on Smart City/SocialCom/SustainCom (SmartCity)*, pages 1059–1064. IEEE, dec 2015.

[LL14] Wei-Po Lee and Ke-Han Lee. Making smartphone service recommendations by predicting users' intentions: A context-aware approach. *Information Sciences*, 277:21–35, September 2014.

[LLLZ13] Robert LiKamWa, Yunxin Liu, Nicholas D. Lane, and Lin Zhong. Moodscope: Building a mood sensor from smartphone usage patterns. In *Proceeding of the 11th Annual International Conference on Mobile Systems, Applications, and Services*, MobiSys '13, pages 389–402, New York, NY, USA, 2013. ACM.

[Mar17] Massimo Marchiori. The secure mobile teen: Looking at the secret world of children. In *International Conference on Wireless and Mo-*

bile Computing, Networking and Communications, volume 2017-October, pages 341–348. IEEE, oct 2017.

[MBVS16] Roberto Munoz, Thiago S. Barcelos, Rodolfo Villarroel, and Ismar Frango Silveira. Game design workshop to develop computational thinking skills in teenagers with Autism Spectrum Disorders. In *Iberian Conference on Information Systems and Technologies, CISTI*, volume 2016-July, pages 1–4. IEEE, jun 2016.

[MEF17] MEF. Global Consumer Trust Survey 2017 - MEF. Technical report, MEF, 2017.

[MEF18] MEF. Global Consumer Trust Report 2018 - MEF. Technical report, MEF, 2018.

[MGWM13] A.-N. Arghir-Nicolae Moldovan, Ioana Ghergulescu, Stephan Weibelzahl, and Cristina Hava Muntean. User-centered EEG-based multimedia quality assessment. In *Broadband Multimedia Systems and Broadcasting (BMSB), 2013 IEEE International Symposium on*, pages 1–8. Ieee, jun 2013.

[Mit16] Siobhan Mitchell. *The Appropriate Use of Digital Devices Application*. PhD thesis, University College Cork, 2016.

[MKD+15] Geetanshu Mangal, Payal Kasliwal, Umesh Deshpande, Manish Kurhekar, and Girish Chafle. Flexible Cloud Computing by Integrating Public-Private Clouds Using OpenStack. In *2015 IEEE International Conference on Cloud Computing in Emerging Markets (CCEM)*, pages 146–152. IEEE, nov 2015.

[MMJ07] N Malik, Umar Mahmud, and Y Javed. Future challenges in context-aware computing. *proceedings of the IADIS . . .*, pages 306–310, 2007.

[MMO13] Edmond Mitchell, David Monaghan, and Noel E O'Connor. Classification of sporting activities using smartphone accelerometers. *Sensors (Basel, Switzerland)*, 13(4):5317–37, jan 2013.

[MNJ16] Seraj Al Mahmud Mostafa, Sheak Rashed Haider Noori, and Saujanna Jafreen. Transparency - A key feature integration in existing privacy frameworks for online user. In *2016 International Workshop on Computational Intelligence (IWCI)*, pages 74–78. IEEE, dec 2016.

[Möl16] Dietmar P. F. Möller. Introduction to the Internet of Things. In *Internet of Things A to Z*, pages 141–184. John Wiley & Sons, Inc., Hoboken, NJ, USA, may 2016.

[MSW14] Abhishek Mukherji, Vijay Srinivasan, and Evan Welbourne. Adding intelligence to your mobile device via on-device sequential pattern mining. In *Proceedings of the 2014 ACM International Joint Conference on Pervasive and Ubiquitous Computing Adjunct Publication - UbiComp '14 Adjunct*, pages 1005–1014, New York, New York, USA, 2014. ACM Press.

[MZBBG18] Daniel Miller, Zhengyuan Zhou, Nicholas Bambos, and Irad Ben-Gal. Optimal Sensing for Patient Health Monitoring. *IEEE International Conference on Communications*, 2018-May:1–7, may 2018.

[NAY18] Ayse Vildan Nurdag, Bert Arnrich, and Arda Yurdakul. WIP: Daily Life Oriented Indoor Localization by Fusion of Smartphone Sensors and Wi-Fi. *2018 IEEE International Conference on Smart Computing (SMARTCOMP)*, pages 258–260, jun 2018.

[NCW+12] Le T Nguyen, Heng-tze Cheng, Pang Wu, Senaka Buthpitiya, and Ying Zhang. PnLUM : System for Prediction of Next Location for Users with Mobility. *Mobile Data Challenge by Nokia Workshop at Pervasive*, pages 1–5, 2012.

[NG17] Alexandra Nikolova and Vladimir Georgiev. A framework for evaluating and improving skills and knowledge of children up to 6 years of age. In *South-East Europe Design Automation, Computer Engineering, Computer Networks and Social Media Conference, SEEDA-CECNSM 2017*, pages 1–5. IEEE, sep 2017.

[Nie12] Jim Nieters. Defining an Interaction Model : The Cornerstone of Application Design, 2012. `http://www.uxmatters.com/mt/archives/2012/01/defining-an-interaction-model-the-cornerstone-of-application-design.php`, Accessed: 2019.02.01.

[NJP15] Saakshi Narula, Arushi Jain, and Prachi. Cloud Computing Security: Amazon Web Service. *2015 Fifth International Conference on Advanced Computing & Communication Technologies*, pages 501–505, feb 2015.

[NM16] Suraj Natarajan and Melody Moh. Recommending news based on hybrid user profile, popularity, trends, and location. In *Proceedings - 2016 International Conference on Collaboration Technologies and Systems, CTS 2016*, pages 204–211. IEEE, oct 2016.

[noy19] noyb. Netflix, spotify & youtube: Eight strategic complaints filed on "right to access", 2019. `https://noyb.eu/access_streaming/`, Accessed: 2019.02.29.

[NSD13] Nagarajan Natarajan, Donghyuk Shin, and Inderjit S. Dhillon. Which app will you use next? In *Proceedings of the 7th ACM conference on Recommender systems - RecSys '13*, pages 201–208, New York, New York, USA, 2013. ACM Press.

[NSF10] Kawa Nazemi, Christian Stab, and Dieter W. Fellner. Interaction analysis: An algorithm for interaction prediction and activity recognition in adaptive systems. In *Intelligent Computing and Intelligent Systems (ICIS), 2010 IEEE International Conference on*, volume 1, pages 607–612. IEEE, oct 2010.

[Ofc18] Ofcom. The communications market report 2018, 2018. `https://www.ofcom.org.uk/__data/assets/pdf_file/0022/117256/CMR-2018-narrative-report.pdf`, Accessed: 2019.02.29.

[OFL10] Daniel Olmedilla, Enrique Frías-Martínez, and Rubén Lara. Mobile web profiling: A study of off-portal surfing habits of mobile users. In *User Modeling, Adaptation, and Personalization, 18th International Conference, UMAP 2010, Big Island, HI, USA, June 20-24, 2010. Proceedings*, pages 339–350, 2010.

[Pas98] Jason Pascoe. Adding generic contextual capabilities to wearable computers. *International Symposium on Wearable Computers, Digest of Papers*, 1998-October:92–99, oct 1998.

[Per18] Perry Tekla S. Apple Announces Tools to Help You Put Your Phone Down - IEEE Spectrum, 2018. `https://spectrum.ieee.org/view-from-the-valley/consumer-electronics/portable-devices/apple-announces-tools-to-help-you-put-your-phone-down`, Accessed: 2019.02.27.

[PH16] Rezvan Pakdel and John Herbert. Efficient Cloud-Based Framework for Big Data Classification. In *Proceedings - 2016 IEEE 2nd International Conference on Big Data Computing Service and Applications, BigDataService 2016*, pages 195–201. IEEE, mar 2016.

[PP08] Jong-Hyun Park and Won-Ik Park. Personalized recommender system for resource sharing based on context-aware in ubiquitous environments. In *2008 International Symposium on Information Technology*, volume 4, pages 1–5. IEEE, 2008.

[PRMF16] Silvia Puglisi, David Rebollo-Monedero, and Jordi Forné. On Web User Tracking: How Third-Party Http Requests Track Users' Browsing Patterns for Personalised Advertising. In *2016 Mediterranean Ad Hoc Networking Workshop (Med-Hoc-Net)*, pages 1–6. IEEE, jun 2016.

[PS14] P Piasek and AF Smeaton. Using lifelogging to help construct the identity of people with dementia. *The international conference on Intelligent Human Computer Interaction (IHCI)*, 2014.

[PU16] THE EUROPEAN PARLIAMENT and THE COUNCIL OF THE EUROPEAN UNION. General data protection regulation (gdpr), 2016. `https://gdpr-info.eu/`, Accessed: 2019.02.18.

[PW08] Anindya S. Paul and Eric A. Wan. Wi-Fi based indoor localization and tracking using sigma-point Kalman filtering methods. In *2008 IEEE/ION Position, Location and Navigation Symposium*, pages 646–659. IEEE, 2008.

[PY06] H S Park and J O Yoo. A context-aware music recommendation system using fuzzy bayesian networks with utility theory. *Fuzzy Systems and Knowledge Discovery*, 4223 LNAI:970–979, 2006.

[RCC12] Khalil Ur Rehman Laghari, Kay Connelly, and Noel Crespi. Toward total quality of experience: A QoE model in a communication ecosystem. *IEEE Communications Magazine*, 50(4):58–65, 2012.

[RCDD98] Tom Rodden, Keith Chervest, Nigel Davies, and Alan Dix. Exploiting context in hci design for mobile systems. In *in Workshop on Human Computer Interaction with Mobile Devices*, 1998.

[Riv06] Oriana Riva. Contory: a middleware for the provisioning of context information on smart phones. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 4290:219–239, nov 2006.

[RJ16] A. T. Ramitha and J. S. Jayasudha. Personalization and privacy in profile-based web search. In *International Conference on Research Advances in Integrated Navigation Systems, RAINS 2016*, pages 1–4. IEEE, may 2016.

[RK08] Oriana Riva and Jaakko Kangasharju. Challenges and lessons in developing middleware on smart phones. *Computer*, 41(10):23–31, oct 2008.

[Row07] Jennifer Rowley. The wisdom hierarchy: Representations of the DIKW hierarchy. *Journal of Information Science*, 33(2):163–180, apr 2007.

[RR18] S. Bhagya Rekha and M. Venkateswara Rao. Methodical activity recognition and monitoring of a person through smart phone and wireless sensors. In *IEEE International Conference on Power, Control, Signals and Instrumentation Engineering, ICPCSI 2017*, pages 1456–1459. IEEE, sep 2018.

[RT03] Claudia Roda and Julie Thomas. Digital interaction: introduction to the first international workshop. In *Proceedings of the 1st Intenational Symposium on Information and Communication Technologies, Dublin, Ireland*, pages 399–402, 01 2003.

[Sal18] Sara Salinas. Apple hangs onto its historic $1 trillion market cap, 2018. `https://www.cnbc.com/2018/08/02/apple-hits-1-trillion-in-market-value.html`, Accessed: 2019.02.29.

[Sat18] Suthamathy Sathananthan. Data valuation considering knowledge transformation, process models and data models. In *2018 12th International Conference on Research Challenges in Information Science (RCIS)*, volume 2018-May, pages 1–5. IEEE, may 2018.

[Sch07] Erick Schonfeld. Breaking: Google Announces Android and Open Handset Alliance, 2007. `https://techcrunch.com/2007/11/05/breaking-google-announces-android-`

and-open-handset-alliance/?guccounter=1 https://techcrunch.com/2007/11/05/breaking-google-announces-android-and-open-handset-alliance/, Accessed: 2019.02.27.

[SGD18] Surbhi Saraswat, Hari Prabhat Gupta, and Tanima Dutta. An interactive writing monitoring system for kids. In *2018 10th International Conference on Communication Systems and Networks, COMSNETS 2018*, volume 2018-January, pages 534–536. IEEE, jan 2018.

[SKC+12] Feng-tso Sun, Cynthia Kuo, Heng-tze Cheng, Senaka Buthpitiya, Patricia Collins, and Martin Griss. Activity-Aware Mental Stress Detection Using Physiological Sensors. *Mobile Computing, Applications, and Services*, pages 211–230, 2012.

[SMM+14] Vijay Srinivasan, Saeed Moghaddam, Abhishek Mukherji, Kiran K. Rachuri, Chenren Xu, and Emmanuel Munguia Tapia. MobileMiner: Mining Your Frequent Patterns on Your Phone. *Proceedings of the Joint Conference on Pervasive and Ubiquitous Computing (UbiComp'14)*, pages 389–400, 2014.

[SRA+06] Computer Science, Carolina Ruiz, Thesis Advisor, Matthew Ward, Thesis Reader, and Michael Gennert. Association Rule Based Classification by. Master's thesis, Worcester Polytechnic Institute, 2006.

[Ste17] Billy Steele. Vizio tracked and sold your tv viewing habits without consent (updated), 2017. https://www.engadget.com/2017/02/06/vizio-smart-tv-viewing-history-settlement-ftc/?guccounter=1, Accessed: 2019.02.29.

[Sto93] Tom Stonier. *The wealth of information, A profile of the post industrial economy*. Thames Methuen, 1993.

[Swe14] Doreen Swetkis. Putting Big Data in Context. *Journal of Public Administration Research & Theory*, 24(1):264–266, 2014.

[Tok13] Helena Tokkonen. How User Experience is Understood ? *Science and Information Conference 2013*, pages 791–795, 2013.

[TYB16] Anand Shanker Tewari, Naina Yadav, and Asim Gopal Barman. Efficient tag based personalised collaborative movie reccommendation system. In *Proceedings of the 2016 2nd International Con-*

ference on Contemporary Computing and Informatics, IC3I 2016, pages 95–98. IEEE, dec 2016.

[WDTP04] X.H. Wang, Da Qing Zhang, Tao Gu, and H.K. Pung. Ontology based context modeling and reasoning using OWL. *IEEE Annual Conference on Pervasive Computing and Communications Workshops, 2004. Proceedings of the Second*, pages 18–22, 2004.

[WP17] Nisha D. Wanjari and Shailaja. C. Patil. Wearable devices. In *2016 IEEE International Conference on Advances in Electronics, Communication and Computer Technology, ICAECCT 2016*, pages 287–290. IEEE, dec 2017.

[WSEK17] Syafrizal Wardhana, Mira Kania Sabariah, Veronikha Effendy, and Dana S. Kusumo. User interface design model for parental control application on mobile smartphone using user centered design method. In *2017 5th International Conference on Information and Communication Technology, ICoIC7 2017*, pages 1–6. IEEE, may 2017.

[WSW07] Wolfgang Woerndl, Christian Schueller, and Rolf Wojtech. A Hybrid Recommender System for Context-aware Recommendations of Mobile Applications. In *2007 IEEE 23rd International Conference on Data Engineering Workshop*, pages 871–878. IEEE, apr 2007.

[WW10] Thiranjith Weerasinghe and Ian Warren. Odin: Context-aware middleware for mobile services. In *Proceedings - 2010 6th World Congress on Services, Services-1 2010*, pages 661–666. IEEE, jul 2010.

[WZL05] J. Wolfgang Kaltz, Jürgen Ziegler, and Steffen Lohmann. Context-aware Web Engineering: Modeling and Applications. *Revue d'intelligence artificielle*, 19(3):439–458, jun 2005.

[Yua14] Bingchuan Yuan. *Context-aware Real-time Assistant Architecture For Pervasive Healthcare*. PhD thesis, University College Cork, 2014.

[YWY$^+$17] Y. Yang, L. Wu, G. Yin, L. Li, and H. Zhao. A survey on security and privacy issues in internet-of-things. *IEEE Internet of Things Journal*, 4(5):1250–1258, Oct 2017.

[ZG12] LM Zhou and Cathal Gurrin. A survey on life logging data capturing. *SenseCam 2012*, 2012.

[Zha03] Qiankun Zhao. Association Rule Mining : A Survey. *Survey Paper*, pages 1–20, 2003.

[Zho13] MengChu Zhou. Internet of Things: Recent advances and applications. *Computer Supported Cooperative Work in Design (CSCWD), 2013 IEEE 17th International Conference on*, page 1, jun 2013.

[Zin07] Chaim Zins. Conceptual approaches for defining data, information, and knowledge. *Journal of the American Society for Information Science and Technology*, 58(4):479–493, 2007.

[ZK18] Marijana Raznjevic Zdrilic and Vesna Kalajzic. Apprehension of media literacy by the student population. In *2018 41st International Convention on Information and Communication Technology, Electronics and Microelectronics, MIPRO 2018 - Proceedings*, pages 726–732. IEEE, may 2018.

[ZWT$^+$17] Jun Zhou, Guiping Wu, Manshu Tu, Bing Wang, Yan Zhang, and Yonghong Yan. Predicting user influence under the environment of big data. In *2017 2nd IEEE International Conference on Cloud Computing and Big Data Analysis, ICCCBDA 2017*, pages 133–138. IEEE, apr 2017.

Appendix B

Footnotes

1. Mobile Matures as the Cross-Platform Era Emerges article: https://goo.gl/Z6j96E

2. User modelling on Science Direct: https://goo.gl/sdm3jB

3. GPS stands for Global Positioning System

4. GSM stands for Global System for Mobile Communications

5. Outline in solving business cases article: https://goo.gl/JzTQHL

6. The future of Data Centres on CB Insights: https://goo.gl/JzJt2A

7. Apple announces 1 million apps in the App Store: http://goo.gl/yMySLB

8. Number of apps available in leading app stores as of 3rd quarter 2018 article on Statista website: https://goo.gl/n9eqJT

9. Cumulative number of apps downloaded from the Apple App Store from July 2008 to June 2017 article on Statista website: http://goo.gl/rTNE80

10. About Google: https://about.google/intl/en/

11. Amazon and the Race to Be the First $1 Trillion Company article: https://goo.gl/Qoiy19

12. Family Link article: https://goo.gl/djo4Va

B. FOOTNOTES

13. NASA website; https://www.nasa.gov

14. Rackspace website: https://www.rackspace.com

15. Amazon Web Service website: https://aws.amazon.com

16. IBM BlueMix website: https://www.ibm.com/cloud-computing/bluemix/

17. IBM website: https://www.ibm.com/

18. OpenStack website: https://www.openstack.org

19. Cloud Foundry website: https://www.cloudfoundry.org

20. Windows Azure website: https://azure.microsoft.com

21. Microsoft website: https://www.microsoft.com/

22. Subway Surfers Information Page on Apple App Store: https://goo.gl/9f0VEE

23. Apple Store Connect Help - App Ratings: https://goo.gl/FdeoZb

24. Samsung Smart Home Appliances: https://goo.gl/4U4pqA

25. Xiaomi smart home: https://goo.gl/2vXCuy

26. Samsung Smart Home Cloud: https://goo.gl/ej48Rn

27. Smappee (Analyse, Control, and Save Energy): https://goo.gl/f2GFX1

28. Sony Lifeloghttps Program: https://goo.gl/1yF5Mh

29. Google Fit SDK: https://goo.gl/osrdxu

30. Fitbit SDK: https://goo.gl/ucM6Kz

Appendix C

List of Publications

Mohammad Hashemi and John Herbert. User interaction monitoring and analysis framework. In The 38th International Conference on Software Engineering IEEE/ACM Press, 2016.

Mohammad Hashemi, and John Herbert. A pro-active and dynamic prediction assistance using BaranC framework. In Proceedings of the International Workshop on Mobile Software Engineering and Systems (MOBILESoft) ACM Press, 2016.

Mohammad Hashemi and John Herbert. Child-centred design supported by comprehensive child application use analysis. SIGCHI Conference on Interaction Design and Children ACM Press, 2016.

Mohammad Hashemi and John Herbert. A Service-Oriented User Interaction Analysis Framework Supporting Adaptive Applications. IEEE 40th Annual Computer Software and Applications Conference (COMPSAC), 2016.

Mohammad Hashemi and John Herbert. A Next Application Prediction Service Using the BaranC Framework. IEEE 40th Annual Computer Software and Applications Conference (COMPSAC) IEEE Press, 2016.

Mohammad Hashemi and John Herbert. Baran: An Interaction-Centred User Monitoring Framework. In International Conference on Physiological Computing Systems 2015, 2015.

Mohammad Hashemi and John Herbert. UIXSim: A User Interface Experience Analysis Framework. In 2014 Fifth International Conference on Intelligent Systems, Modelling and Simulation (IJSSST), 2014.

Appendix D

Demonstration of the Baran Framework in Use.

This section presents the screenshots of various parts of the Baran framework on the cloud-side and client-side (Figure D.1), including the data collection service (running on an Android device) and the Baran cloud services (running on Amazon AWS).

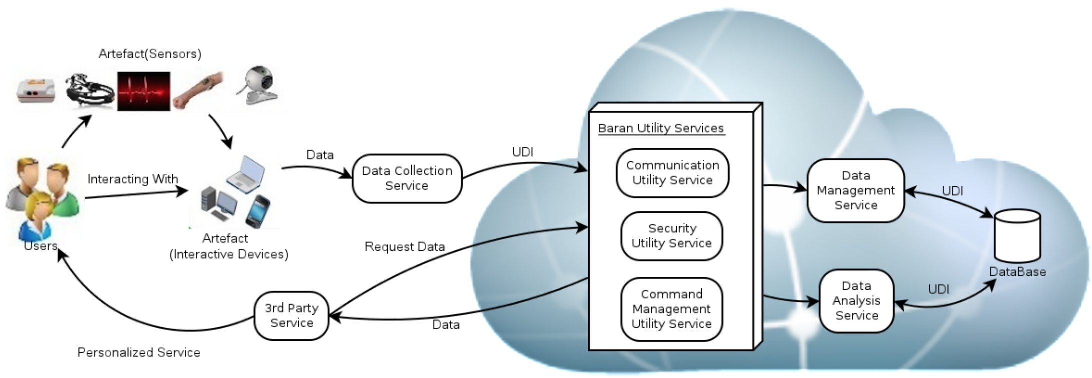

Figure D.1: Baran framework services on the cloud-side and client-side

D.1 Baran on the client-side

The Android-based Baran data collection service is implemented using Android Studio. It needs to access capabilities and information (known as permissions) on the device. The user is informed about what permission the app requires in the process of installation (Figure D.2). The data collection service is disabled by default and does not monitor the user's activities unless the user activates it. Baran data collection service provides a UI (Figure D.3) so that the user can start/stop the service, connect the service with their Baran account so that Baran links their data to their account, and also control 3rd party services and respond to their requests. Figure D.4 shows the CPU, memory, and network usage of the device while the data collection service is running.

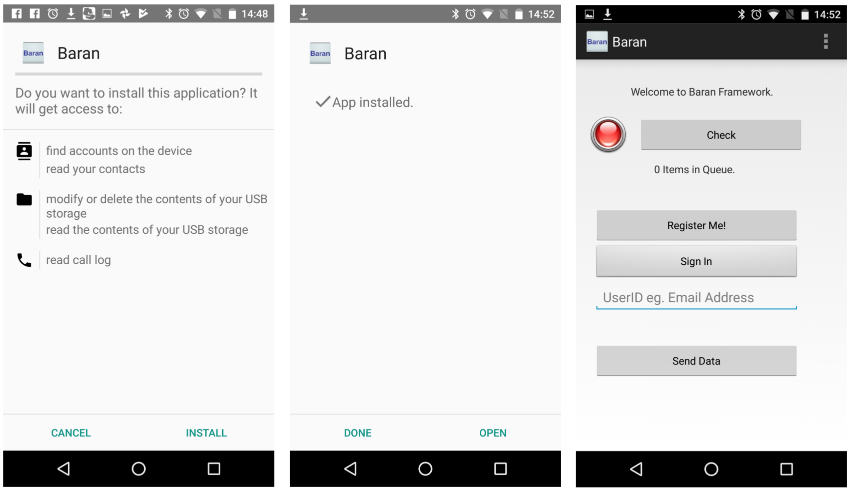

Figure D.2: Baran data collection service installation

The screenshot on the left shows the permissions required by the service. The screenshot on the right shows the service's UI after installation.

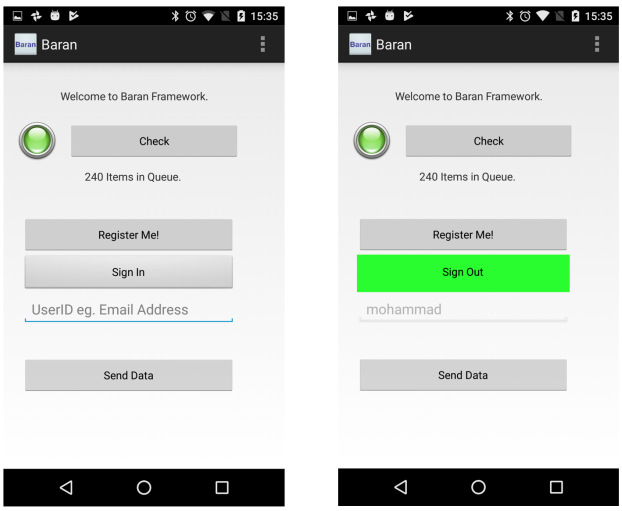

Figure D.3: Baran data collection service UI

Screenshot of Baran data collection service UI (described in Chapter 4, Section 4.3) running on an Android device. On the screenshot on the right side, the user "Mohammad" registered the device under his name, so the collected data will be associated with his account.

Figure D.4: The hardware usage of the device when Baran data collection service is running

D.2 Baran on the cloud-side

The cloud-side of the Baran framework performs data management and processing. Various services (Figure D.5) work together to manage the data received from the data collection services on the client-side. Baran is implemented in a generic way so it can be deployed on various cloud software platforms. In this work, Baran is deployed on Amazon AWS cloud platform. Figure D.6 shows the list of queues used by Baran services. Figure D.7 shows the list of topics used by Baran in order to send notifications to client devices.

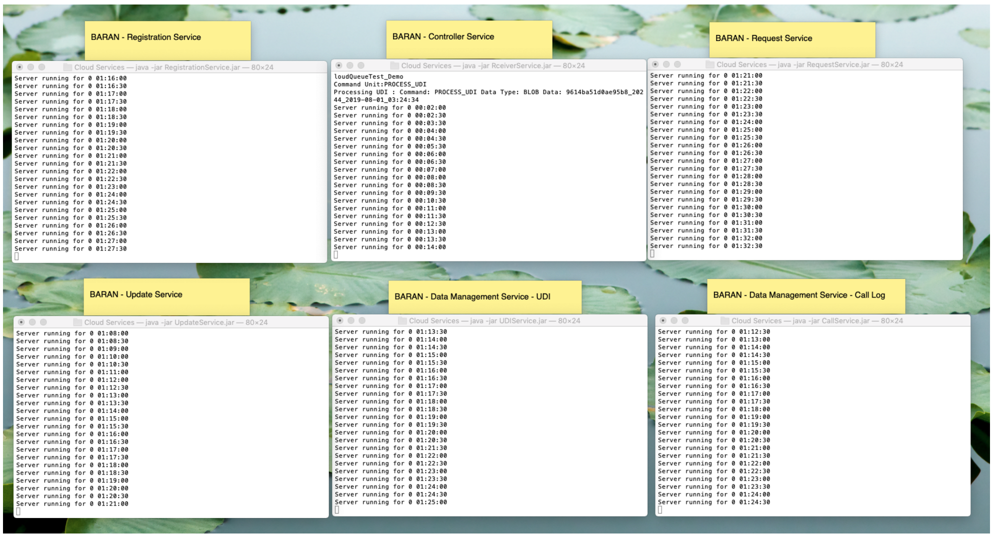

Figure D.5: Baran cloud services

Various cloud services (described in Chapter 4, Section 4.2) working together in the Baran framework.

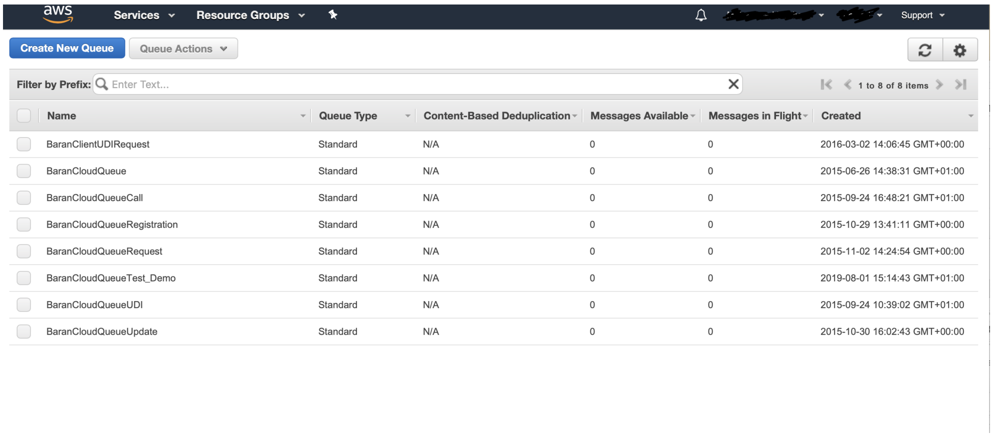

Figure D.6: Baran data queues

List of data queues on the Amazon AWS - Simple Queue Service (SQS) system. The queues are used by Baran services for communication purpose.

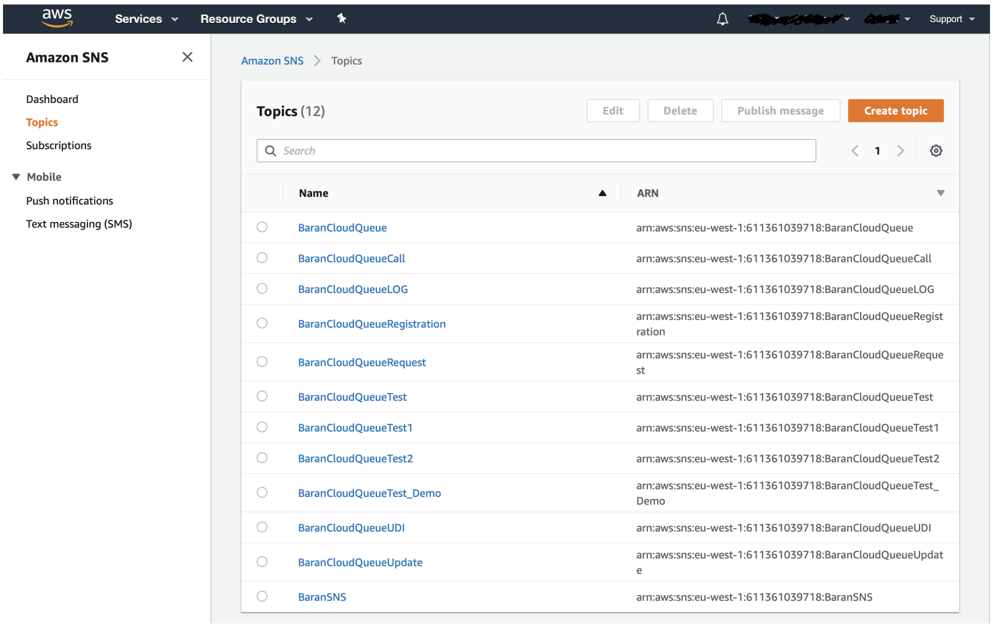

Figure D.7: Baran data topics

List of data topics on the Amazon AWS - Simple Notification Service (SNS) system. The topics are used by Baran services in order to push a notification (message) to client devices (e.g. smartphone and tablet).

Appendix E

Questionnaire

This section presents the questionnaire that was used in the conducted experiment (Chapter 5, Section 5.3.3) of this study.

Questions	Strongly Disagree	Disagree	Neutral	Agree	Strongly Agree
1. Accessing the predicted application's icon is easy.					
2. Next-app (predictive assistant service) is useful.					
3. Next-app provides accurate predictions.					
4. Next-app presents the information with clarity.					
5. Next-app user interface (UI) is easy to use.					
6. Next-app helps to save time and effort.					

Appendix F

Baran Framework Privacy Policy Statement

This is the policy statement used in the experiments involving Baran.

BARAN FRAMEWORK PRIVACY POLICY STATEMENT

WELCOME TO THE BARAN FRAMEWORK PRIVACY POLICY

This Privacy Policy applies only to Baran research and any downloadable file under its name owned by Mohammad Hashemi. This Privacy Policy only covers information collected by or via the Baran Android application and does not cover any information collected by any other resources. This Privacy Policy addresses how Baran collects information by or via the application, what information it collects, how it may use this information and other related issues. Please review this Privacy Policy carefully.

HOW INFORMATION IS COLLECTED AND USED

Information You Provide To The Baran Framework

We do not collect any personally identifiable information about you, such as your name and e-mail address ("*Personal Information*") and other information that, on its own, can be used to identify you personally, such as your age ("*Demographic Information*"). We also do not collect any information about what you type anywhere in your smartphone or any number you call. We may collect the information such as the name of an application, number of clicks, scrolling, and touches within application, sensor information e.g. accelerometer, gyroscope, etc., and the time and the date of your device interactions.

How We Use Your Information

We use your information anonymously for research purposes only. We do not share your information with third parties. We will delete all data after the research is finished.

The Effective Date of this Privacy Policy is January 1st, 2015.
© 2015 Baran Framework All Rights Reserved.

Your Permission to Baran to Collect Information
In order to collect information from your smartphone or tablet, you will need to perform some steps, and whenever you do not want to continue you can stop the Baran application collecting information. PLEASE do not uninstall the application until you send us the current collected data, because by uninstalling the application, all collected data will be erased.

Step to Start:

1- Go to "Setting" -> "Security", Find "Unknown Sources", and Check it.
2- Install the Baran Framework Android Application.
3- Go to "Setting" -> "Security", Find "Unknown Sources", and Uncheck it. for your privacy, Please make sure it is UN-CKECKED.
4- Find the installed Baran Application, Open it, Click on "Refresh" Button once.
5- Go to "Setting" -> "Accessibility" -> "Baran", Set it "ON"
6- To make sure everything is working properly, Open the installed Baran Application, Click on "Refresh" Button. If the label below the "Refresh" button changes, that means the Baran application works properly. If not, please contact us via Email: mhashemi_shz@yahoo.com

Step to Stop:

1- Find the installed Baran Application, Open it, Click on "Send Data" Button.

2- Go to "Setting" -> "Accessibility" -> "Baran", Set it "OFF"

Note: You also can completely uninstall the app from your device.

The Effective Date of this Privacy Policy is January 1st, 2015.
© 2015 Baran Framework All Rights Reserved.

Appendix G

The Appropriate Use of Digital Devices Application

This section presents four figures extracted from a research study [Mit16] that uses Baran to monitor children, discovers their communication patterns, and supports parents to supervise their children.

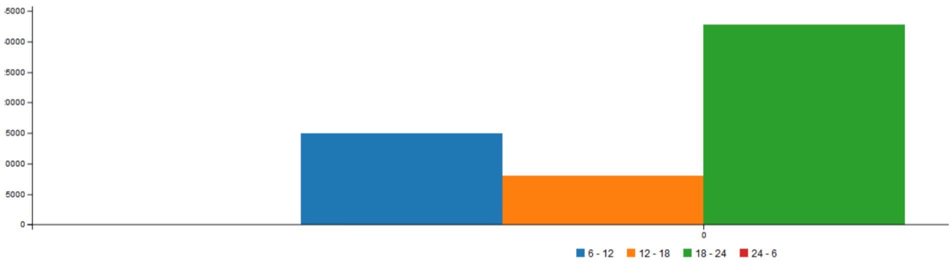

Figure G.1: Showing Length of time on phone at different times of the day

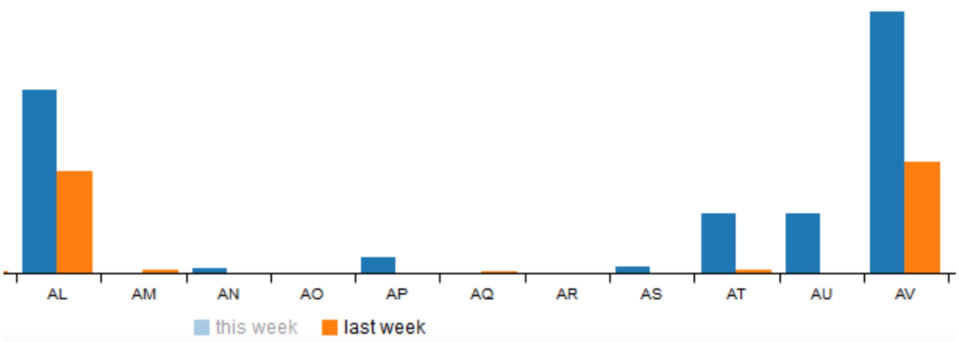

Figure G.2: Comparing phone call length over two successive weeks

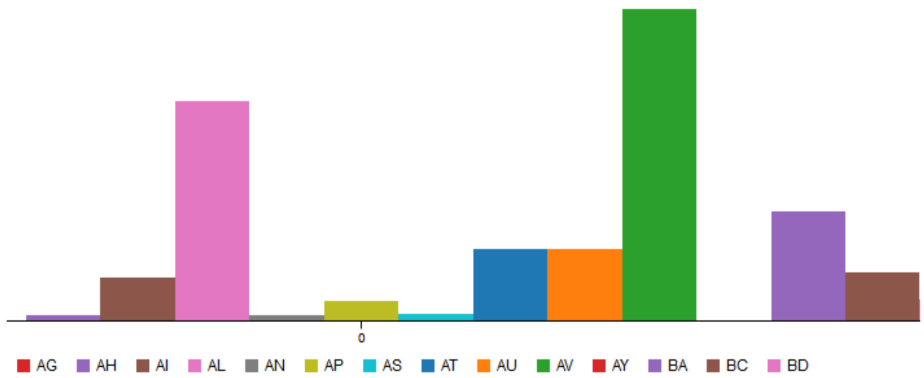

Figure G.3: Showing phone call length to individuals (anonymised)

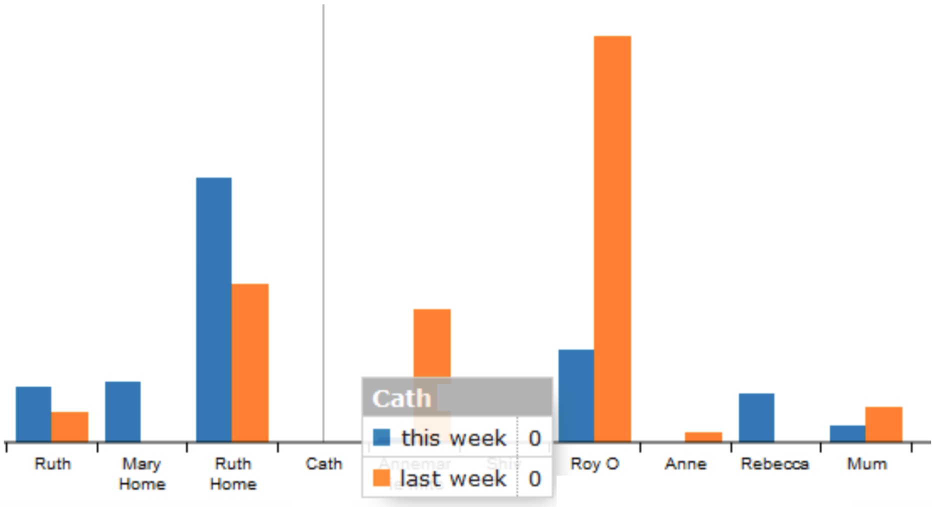

Figure G.4: The de-anonymised user communication summary