Mobile App Driven by Linked Data for Power Supply Information

by

Ying Wang, B.Sc.

Dissertation

Presented to the

University of Dublin, Trinity College

in fulfillment

of the requirements

for the Degree of

Master of Science in Computer Science

University of Dublin, Trinity College

September 2017

Declaration

I, the undersigned, declare that this work has not previously been submitted as an exercise for a degree at this, or any other University, and that unless otherwise stated, is my own work.

Ying Wang

August 23, 2017

Permission to Lend and/or Copy

I, the undersigned, agree that Trinity College Library may lend or copy this thesis upon request.

Ying Wang

August 23, 2017

Acknowledgments

I would first like to express my deepest appreciatio my supervisor Dr. Rob Brennan, without his guidance and persistent help and feedback this dissertation would not have been possible.

I would like to thank Sahil Mathur, a PhD candidate from the ADAPT Centre, who provided me the infrastructure of this project.

I would also like to thank to all those who participated in the experiments.

Finally, I must express my very profound gratitude to my parents and to all of my friends and classmates for providing me with unfailing support and continuous encouragement throughout my year of study.

Ying Wang

University of Dublin, Trinity College

September 2017

Mobile App Driven by Linked Data for Power Supply Information

Ying Wang, M.Sc.
University of Dublin, Trinity College, 2017

Supervisor: Dr. Rob Brennan

This study explores the approaches to building high usability of consumer-oriented product matching Linked Data mobile applications. A commercial product matching mobile application use case was designed, implemented and evaluated through an iterative process in support of this study. This dissertation also presents the potential challenges in consumption of Linked Data and user interface design on mobile devices, and proposed a possible solution to overcome these challenges. It also addressed the lack of research on Linked Data mobile application usability.

The evaluation consisted of a combined experimental investigations and comparative investigations. The usability of the mobile app was measured by applying a set of standard usability metrics such as effectiveness, efficiency, user satisfaction and so on. The results of experiments indicated that it is possible for a consumer-oriented and product matching Linked Data mobile application to achieve high usability. The analysis of the results also showed the importance of the user feedback to improve the usability during the mobile application design.

The main contribution of this study is to provide a proposed approach on the design of a high usability Linked Data mobile application, which could support the decision making for future applications. This was done by comparing the technical approach between multiple Linked Data mobile applications and identifying which design patterns can improve the usability of the product matching Linked Data mobile application. The experimental results from a series of usability experiments have shown that the proposed approach can enable

product matching Linked Data mobile applications to achieve high usability. The future work of this study will expend to the experiments to a large amount of participants.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

1.1 Motivation

In recent years, mobile applications dominate consumer use of computers. According to the study from Statista[1], there are millions of mobile applications available for download in leading app stores in 2017 (Android: 2.8 million mobile applications, iPhone: 2.2 million mobile applications). Product matching is a popular type of mobile application. People are using this kind of mobile application from Amazon, eBay, and BuyVia to find products related to other products, e.g. product accessories like power supplies. However, collecting the relevant data to support these product matching applications is always challenging.

The leading web-based, standards-based data publishing and management technology is Linked Data based on the Semantic Web. *"The Semantic Web is an extension of the current web in which information is given well-defined meaning, better enabling computers and people to work in cooperation."* [1] Linked Data is a step towards the Semantic Web. Linked Data can be defined as *a set of best practices for publishing and connecting structured data on the Web* [2]. Linked Data is about applying the standard principles to make data available for sharing

---

[1] https://www.statista.com/statistics/276623/number-of-apps-available-in-leading-app-stores/

based on how to publish and interlink data in both human and machine readable way, so that people can navigate to another related resources simply while browsing one resource on the web. However, the main challenge regarding this research area is although Linked Data provides a large source of structured knowledge on the web, but it is still largely impossible for non-specialists to consume [7].

There are currently many existing successful web applications using Linked Data approach to consume data [3]. However, with the rapid growth of mobile technology and the continuous increase in mobile users, many developers are shifting their attention to the mobile world [4]. The Linked Data approach can be applied in mobile application development, but it also implies unique challenges for how to design a well-integrated mobile system with a user-friendly interface. Unlike Linked Data web applications, there is not much previous research to support in the area of usable mobile linked data applications. According to the survey from Yus and Pappachan, there were only 36 Linked Data mobile applications available until 2015 and there was none related to product matching [6]. Therefore, it is timely to investigate how to make it easy for people to consume and discover useful information when applying Linked Data to the real mobile world.

1.2 Research Question

To what extent can consumer-oriented, product matching mobile applications based on Linked Data achieve high usability?

The following terms used in the research question are defined below to further clarify the scope of the planned work.

Consumer-oriented means this project is focusing on general web users in the mass market.

Product matching is defined as searching and browsing for related products.

In addition, this study uses ISO (International Organization for Standardization) 9241-11 to measure the usability of the mobile application.

Therefore, **usability** here is defined in ISO 9241-11 as *"The extent to which a product can be used by specified users to achieve specified goals with effectiveness, efficiency and satisfaction in a specified context of use."*

1.3 Research Objectives

The following research objectives are necessary for answering research question.

1. To review the state of the art in mobile technologies, usability design and mobile Linked Data applications, to establish appropriate design methodologies, potential challenges and evaluation techniques.
2. To establish, in conjunction with MyVolts Ltd., a rich use case where end-users explore Linked Data to solve a practical product matching problem with a mobile app.
3. To design a high usability mobile interface to explore and query the MyVolts Linked Data to satisfy the product matching use case.
4. To implement a prototype mobile app based on the design.
5. To evaluate and improve the usability of the prototype iterations by running a series of structured usability experiments, user interviews and consumer surveys.

1.4 Technical Approach

MyVolts[2] is an online power supply retailer which provides a wide range of products and solutions to allow people to buy replacement power supplies and cables for their devices. However, finding the correct power supply for a customer's device is complex due to the large number of devices and the number of variations in power supplies. As the underlying datasets are very large and complex, theoretically a linked data approach can provide ability to improve efficiency and usability to develop a mobile app to address the issue. However, without the full stages of design, implementation and evaluation, such an app is unlikely to have high usability.

[2] http://myvolts.com/

The MyVolts app has been decided to be built on a native android mobile platform. This application builds on the infrastructure created by PhD candidate Sahil Mathur from the ADAPT Centre, which uses the OnTop platform [6] to dynamically create a linked data dataset from the commercial MyVolts relational database of power supply and device technical specifications. Then this provides SPARQL remote server endpoints to allow the MyVolts mobile client to explore and query datasets by sending a HTTP request.

There are currently many mobile application design guidelines available such as Norman's human-computer interaction (HCI) models and Constantine & Lockwood's 6 user interface design principles, which are foundations of designing a standard mobile app. There are also many challenges related to the consumption of Linked Data and the implementation of a high usability mobile application that need to be addressed during the app design. In order to decide the best technical solutions to implement the MyVolts mobile application, the review of the state of the art Linked Data mobile applications are necessary. In addition, the analysis of modern search design patterns is significant to help to identify which are best to improve the usability of the mobile applications.

When designing the user interface of MyVolts mobile application, paper prototyping is helpful to visualize the design ideas and identify which design is able to enable the best results. The use cases are established to identity the requirements of MyVolts mobile application in conjunction with MyVolts Ltd. Then the technical and the search design pattern decisions are applied to design and implement the MyVolts mobile application.

1.5 Evaluation Methodology

This study aims to investigate the usability of applying linked data approach to the real mobile world. The evaluation methods focus on the combined lab-based user task performance experiments and an online questionnaire.

- **Lab-based user task performance experiments**

  These will be controlled trials with users being provided some training in the app, then being given a set of tasks to complete and finally a post-task questionnaire and structured interview.

- **Web-based questionnaires**

  These will be used to survey the wider community of potential users for feedback on suggested app features and early screen mockups.

The evaluation stage includes four iterative experiments, volunteers are selected ranging from MyVolts experts to people with CS background and non CS background. The initial experiment aims to gather the early feedbacks from expert participants who come from MyVolts based on discussing and examining functionalities and usability of the early prototype. These feedbacks are helpful to maximize users' satisfaction and keep the development in the right stage.

The second experiment is done after a full design and implementation of the app by investigating both experts and potential users. Volunteers are requested to complete full related usability tasks, and then fill the System Usability Scale (SUS) survey in order to help measure the usability of mobile client. The experimental results are analyzed from the users' feedback and their performance when interacting with the app, focusing on how well the participant understand the whole system, how they interact with the system functionalities, and what preference they have for completing a specific task.

The third experiment aims to investigate the difference of usability between MyVolts mobile application and web application. Volunteers are requested to complete same usability tasks from the second experiment and fill the SUS survey, which supports the comparison between two experimental results.

The fourth experiment focuses on investigating potential users from MyVolts mailing list. An online questionnaire is used to examine people by providing prototype screenshots and several interesting questions regarding the app usability design, which allows to gather more feedbacks of what people are thinking about and expect.

1.6 Structure of the Rest of this Dissertation

Chapter 2 – Background introduces the fundamental knowledge to support the design and evaluation of this project including the Linked Data technologies and challenges for app-based Linked Data consumption, human-computer interaction, mobile usability design, and standard usability evaluation methodologies.

Chapter 3 – State of the Art reviews the state of the arts knowledges which is relevant to this project. This chapters will discuss the most well-known Linked Data mobile applications and usability evaluation tools.

Chapter 4 – Design presents how the background research can be applied in real mobile application usability design and how potential challenges can be resolved. This chapter illustrates a detailed design for a real Linked Data mobile application which provides the ability for high usability consumption and discovery of the information from a large scale dataset.

Chapter 5 – Evaluation evaluates how well the design of this project meets the research objectives by running a series of usability experiments.

Chapter 2

Background

The aim of the background chapter is to review the fundamental concepts and technologies which support the completion of this project. Topics include: an introduction to Linked Data technologies, challenges for consumption of Linked Data datasets, the foundation of usability design including human-computer interaction, the standard usability model, and mobile usability. Mobile user app and usability evaluation techniques are covered since they are important to identify how successful of the research outcomes are and which aspects can be improved in future, it is helpful to leverage existing usability evaluation standard and metrics to increase the comparability of the results to similar studies and ensure evaluation is carried out on a sound basis.

2.1 Linked Data

This section focuses on introducing the basic concepts of Linked Data and how potentially to apply Linked Data technologies to this research topic. In addition, the main challenges for applications consuming Linked Data are discussed. Figure 2.1 presents the general architecture of Linked Data applications.

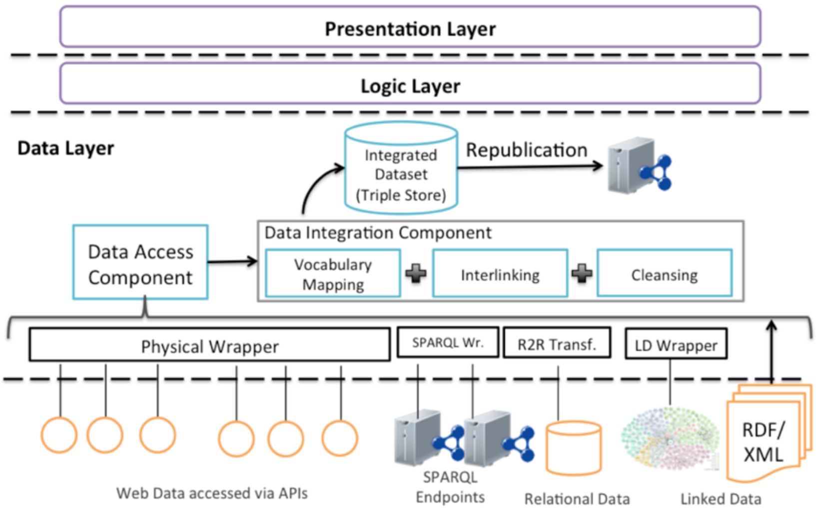

Figure 2.1: General Architecture of Linked Data Applications.

2.1.1 Linked Data Technologies

Linked Data provides a set of best practices in the semantic web to make interlinked data sources on the web not only human readable, but also machine readable. Linked Data can be defined as *"a term used to describe best practice for exposing, sharing, and connecting pieces of data, information, and knowledge on the Semantic Web using URIs and RDF."* [26]

The inventor of the World Wide Web and the creator of the Semantic Web and Linked Data, Tim Berners-Lee, has introduced four design principles to support publishing data on the web as Linked Data [32]. These are shown below.

- Use URIs as name for things;
- Use HTTP URIs so that people can look up those names;
- When someone looks up a URI, provide useful information, using the standards (RDF*, SPARQL);
- Include links to other URIs, so that they can discover more things.

1) URIs

URIs are used to provide the unique identification for any data item, which allows them to be distinguished without confusion. The most popular approach to access the Linked Data is to use HTTP URIs which allows web clients to fetch useful information and discover related resources.

2) RDF

Linked Data recommends RDF as a standard to encode interlink data for supporting or related useful information. Information in RDF is represented as a series of statements (triples). Each statement consists of the subject, predicate (property), and object. The subject of the statement is always identified by a URI and represents a single concept or instance. The object of the statement can be defined as the resources behind the URI which can be either a single URI referring to another concept/instance or a data value (literal). The predicate is then used to describe the relationship between the subject and the object. Figure 2.1 presents an example of RDF triple.

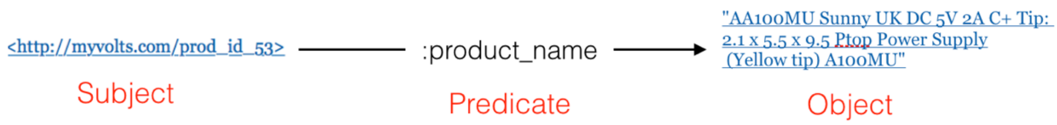

Figure 2.2: A sample RDF triple.

3) Vocabulary

To publish data in the relational database, it requires understanding of the whole data structure (schema). In contrast, there is no requirements to make data available within a global in Linked Data. People are free to publish their own data statements with their own local schema, which gives more flexibility. However, to maximize data interoperability, it is still useful to agree common schema designs in some way. These are called vocabularies.

On the Semantic Web, vocabularies define the concepts and relationships used to describe and represent an area of concern [28]. The examples of vocabularies offered by W3C including RDF and RDF Schemas, Simple Knowledge Organization System (SKOS), Web Ontology Language (OWL), and the Rule Interchange Format (RIF).

4) SPARQL

SPARQL is a query language which supports the access to the Web of Data [29]. A SPARQL query consists of three parts [30]. First, the *pattern matching part* provides several useful features of pattern matching such as union of patterns, values filter, nesting, and so on. Second, the *solution modifiers* allow modification of the output after computation of the query by using operators like distinct, limit, offset, order by, group by, and etc. The third one is the *output* of a SPARQL can be presented in different types including yes/no queries, the descriptions of resources, selections of variables, output tables, and so on.

2.1.2 Challenges in Consumption of Linked Data

Undoubtedly, Linked Data has supported a large amount of data to be published on the web and made them machine understandable and meaningful. However, there are also many challenges to consume data on the web due to its open, dynamic and distributed nature. The following challenges are addressed in detail.

- The change frequency and volume of data can significantly affect all related datasets [32].
- As there is lack of interoperability between the user interface and the underlying data sources, it requires to remodel the data before presenting them in the user interface.
- As the remodeling could be very complicated and may require frequently query the multiple datasets, which potentially lead the less efficiency and time consuming [33]. But normally the efficiency to interaction with the data stores are very important for user interfaces, thus it could be the main challenge in Linked Data consumption.
- There are also challenges to consume multiple sources of Linked data within one homogeneous search space as there is lack of mature method for use of the Web of Data to access a generic homogenous resource. The unresolved issues include co-reference, ontology mapping, aggregation from distributed sources, and resource discovery [34].
- Linked Data publisher and consumer can be different people, and this may cause a difficulty for consumer to understand the whole model if without explanation as consumer may only have partial knowledge [34].

2.1.3 Summary

This section introduced the foundation of the Linked Data including how data are stored and accessed, this gave the basic ideas of how the MyVolts app could retrieve information from the remote server endpoints. In addition, the challenges of consuming Linked Data were also addressed.

2.2 Human-Computer Interaction

HCI provides the ability to bring and increase the power of digital computer systems and it concerns on how people are potentially affected by computer systems. This purpose of this section is to provide a foundation of HCI in design. As the effective design of HCI could be very difficult, this section also introduced the user-centered HCI model which will support the design of the MyVolts app.

2.2.1 Introduction to HCI

Human-Computer Interaction (HCI) is the study of interaction between people (users) and computer. The Association for Computing Machinery [26] defined HCI as *"a discipline concerned with the design, evaluation and implementation of interactive computing systems for human use and with the study of major phenomena surrounding them."* It is evolved by computer science, cognitive psychology, design and several other related fields. The increasing development of communication technologies and networks has supported a wide range of digital communications. As there is a diversity of user populations, it is necessary to make the interactive devices and systems available for everyone anytime and anywhere. The research of HCI provides the ability to response the rapid development of technologies.

Each term in the HCI acronym can then be defined as following:

1) **Human**

The term of *human* typically refers to the users of the system. It can be difficult to identify who is the user as there are many variables to describe a person such as the gender, age, and nationality. The system should allow the flexibility and possibility of different types of people to use the system. As different people could have different way to interact with the system, how to meet different users' needs is a main challenge in HCI.

2) **Computer**

The term of *computer* refers to the electronic devices which provide the ability to receive and manipulate data. It can be desktops, laptops, mobile devices, and etc. Different types of devices have different supports for interactions, which also requires different design approaches.

3) **Interaction**

The term of *interaction* can be defined as the exchange of data between the human and computer.

2.2.2 Interaction Model

With the continuous increasing availability of digital information, computers are ubiquitous and highly effective designs are playing a significant role in our daily activities. However, due to the complex human cognitive system, it is difficult to design a system which allows users to interact with the system effectively. De Bono emphasizes that it is better to simplify a process rather than to instruct people to deal with the complexity [35]. There are currently many HCI process models including ACM SIGCHI model [39], Suchman's model [40], and Norman's model [41], and so on to assist design of interactions at an appropriate level by defining general processes.

Norman's model is one of the most well-known process model to describe interaction as shown in figure 2.1. Norman's interaction model focuses on human thought and how they accomplish actions. It consists of seven stages from establishing a goal to evaluating the results. The biggest advantage to apply user-centered interaction design is that it is useful to allow designers to understand the reasons of user interface failures from the users' perspective.

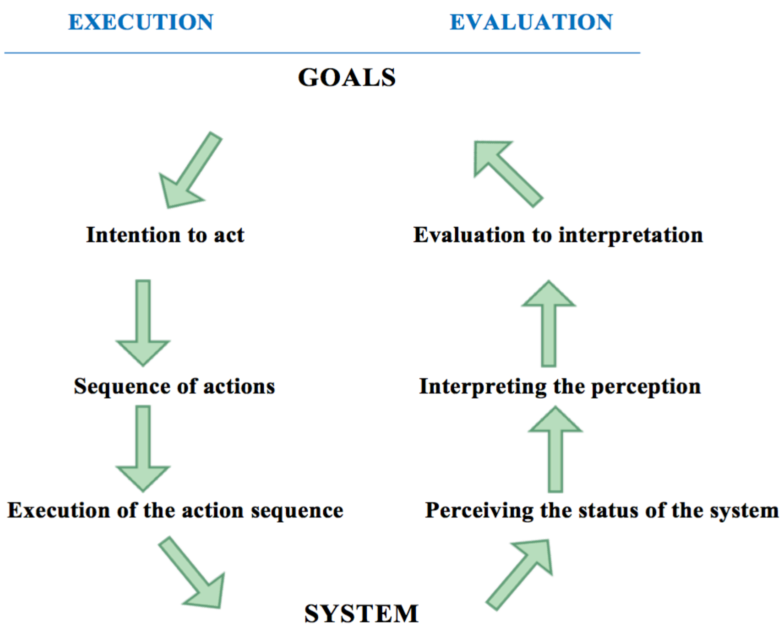

Figure 2.3: Norman's Interaction Model.

It is also worth to mention that the Norman's model supports a feedback loop. In HCI, most researchers are agreed that the feedback process provides the ability to allow users to communicate with system closely [36,37,38]. Feedback is able to give the information about how close they are to achieving their goal through appropriate mechanisms. There mechanisms include: dialog information, progress bars, or error messages. Table 2.1 illustrates each stage's content and description.

TABLE 2.1 Seven Stages of Norman's Model

Stages	Stage Content	Description
1	Goals	What do you want?
2	Intention to act	What would satisfy the goal?
3	Sequence of actions	What do you have to do to achieve the goal?
4	Execution of the action sequence	Steps of the action.
5	Perceiving the status of the system	Guessing the system status by using your sense (e.g. progress bar status).
6	Interpreting the perception	What has been changed?
7	Evaluation the interpretation	Have you achieved the goal?

Norman's interaction model includes three different types including user's mental model, the system image model, and the conceptual model.

- **User's Mental Models**

  The user's mental model represents people's understanding of how things work [41], different people have different mental models so that they may have different thoughts on the same thing. User's mental models are always changing and evolving based on the adoption of the new knowledges and the changes of their surroundings.

- **System Image Model**

  The system image model refers to the visibility of the physical devices. Users are not designers of the devices which they are interacting with, so they rely on the previous experience from similar apps or any other available information. The combined information gives the basic system image.

- **The Conceptual Model**

  The conceptual model is a technical model of the system created by designers for the internal technical use. A conceptual model describes and often simply how the thing works. Everything people see on the device's screen helps people to create a conceptual model of these things inside the computer.

A good interaction design should always start from the user's mental model. The system image model and the conceptual model are related to the system design, so the design should be close to the user's mental model in order to allow users to use the system effectively.

2.2.3 Summary

This section introduced the basic knowledge of the HCI and discussed the well-known user-centered HCI model which could support the interaction design of the application created in this study. As discussed above, it is concluded that the application will focus on the user thought to make sure the app is acceptable to users.

2.3 Mobile User Interface Design

Good usability is a crucial factor to define how successful a mobile application will be in competition with other existing solutions. In other words, no matter how much time and effort are spent on designing, a lack of considering user issues in the mobile context will still bring failures. In fact, mobile devices are basically computers with full features, functions, and small screens. However, it still requires a different approach on user interface design. This section introduces the basic user interface design principles and most commonly mobile user interface design challenges.

2.3.1 Basic User Interface Design Principles

The design process of the user interface can be very complicated. Most user interface design specialists champion that the *user* is the most important factor need to be considered from initial design [25]. There are currently many user interface design guidelines and patterns which support designers to follow user-centered approach and quality their design. One of the well-known design principles is *Constantine & Lockwood's 6 user interface design principles* [24].

1) The Structure Principle

A structured design is referred to the overall architecture of user interface is clear and well organized. Consistency should be implemented throughout models in order to allow certain action to trigger the correct response. A consistent user interface design does not confuse users and make a sense for users on how to control the system.

2) The Simplicity Principle

The balance between system functionalities and simplicity has to be critically taken into account when designing any user interface. A simple mobile application does not mean a simple style user interface design, it can be defined as a learnable system, which means it should provide the ability to allow users to interact with system without taking long time to identify the detailed instructions. The design should consider the level of ease for completing the common tasks, the consistency of the system, and the least steps to achieve the goals.

3) The Visibility Principle

A good user interface design should make all related information visible for leading users to complete the tasks, giving them guidelines, and indicating the possible actions they could take. The interface shouldn't display useless or redundant information which could potentially confuse the users.

4) The Feedback Principle

Users should always be informed about the status of actions when interacting with the system through appropriate feedback. Feedback implies how close the users to complete a task. If errors have occurred, feedback also provides the ability to tell users they are taking the wrong action. Feedbacks can be presented in different ways such as dialogs, hints, or color changes. These are significant to allow users to know the outcome of their actions is achievable and meaningful.

5) The Tolerance Principle

The design should be tolerant by implementing appropriate validation to prevent potential user issues and allow users to recover their mistakes and misuse. A highly tolerate system is helpful to achieve both cost and time effective.

6) The Reuse Principle

The reuse principle is actually referred to a learnable system. A good user interface design should allow users to complete the tasks without thinking or remembering every time.

2.3.2 Mobile Usability Challenges

Mobile devices have many constraints and limitations from its tiny screen nature and new interaction patterns, which may bring many different kinds of challenges to design a user-friendly interface. However, it is also worth mentioning that there are many positive sides for mobile platforms such as accelerometers, GPS, high-resolution screens, and so on. Some findings [42] have shown that instead of thinking about limitations of mobile platform, alternatively focusing on the user goals and turning limitation into opportunities might redefine how people interact with new generation smartphones.

One of the challenges for mobile applications design is to understand user issues. There are many issues making mobile applications difficult to use, the following are some most standard issues identified by Zhang and Adipat [9].

1) Mobile Context

Unlike fixed indoor context, mobile contexts are easy to change. Mobile technologies can be considered as the tools to help people complete various tasks and goals for their daily activities on the move. As mobile environments are unpredictable and dynamic, it requires heightened awareness to tolerate any possible situation and provide alternative solutions for different user actions. Context-aware devices are supposed to monitor the changing contexts of the user and adapt appropriately through interpreters, aggregators, and services [8]. Generally, people are easier to focus on the specific activity in fixed contexts such as home, offices, and classrooms rather than in mobile contexts, which means mobile technologies design should be user-cantered. Understanding mobile context requires rich knowledge from end-users' perspective, including how mobile contexts are created by human interactions, surroundings, and technology capabilities and availabilities.

2) Limited Connectivity

According to a new study from Google in 2016, *"The Need for Mobile Speed"*[3], 53% of mobile site visits are abandoned if pages take longer than 3 seconds to load. The slow and unreliable network connections on mobile devices are common barriers of mobile applications. Although with the development of the wireless technologies, the connectivity has been improved [10], however, it still can be an issue that may impact the performance of the mobile applications.

3) Small Screen Size

As the natural limitation, it is very difficult to display a large amount of information on the small screen, which can significantly impact the overall usability of the mobile applications.

4) Limited Processing Capability and Power

Although the ability of the mobile devices has been increased rapidly, however, it can still not be equivalent to the desktops and laptops. As the mobile devices have the constraints on memory and power, they cannot provide a high level processing performance compared to desktops and laptops. For example, some desktop applications may require a large memory for processing graphics or fasting

[3] https://www.doubleclickbygoogle.com/articles/mobile-speed-matters/

processing speed [44], but it may not be applied to mobile devices because of the limited processing capability.

5) Data Entry Method

With the physical limitations of mobile devices, it is unable to use the full keyboards as the data input method. In addtion, typing on the screen can be hazardous and cause many usability problems like “fat finger problem” [4]. Small input boxes and buttons limit users’ ability on data input, which may increase the potential user issues. Also when using a mobile device, different user activities such as walking, sitting, and talking may significantly impact the efficiency and effectiveness of mobile applications, it is worth to implement different data entry methods.

2.3.3 Summary

In this section, the mobile user interface design principle was introduced to give a basic instruction of how to design a quality user interface. The most well-known mobile design challenges were also analyzed which gave the background in order to provide a solution to mitigate the impact of mobile limitations when designing the MyVolts app.

2.4 Usability Evaluation Techniques

Usability evaluation provides the ability to help designers and developers to identify potential issues of the product. During the evaluation, it is able to learn for example if the users can complete the tasks efficiently and effectively without any unintended behaviors. In this section, the usability model used in this project is introduced. The metrics which support to evaluate the usability of the MyVolts app are also discussed.

2.4.1 Usability Model

When evaluating the usability of the system, it is very difficult to design a correct approach. The International Organization for Standardization (ISO) has developed various of, which are widely used to measure the usability of the system.

ISO 9241-11 provides many advantages comparing with other models such as it considered context of use [21]. Usability is defined in ISO 9241-11[19] as "*the extent to which a product can be used by specified users to achieve specified goals with effectiveness, efficiency and satisfaction in a specified context of use.*"

Usability attributes are commonly used to measure how usable a system is. In ISO 9241-11, Each usability attribute is then defined as follows:

- **Effectiveness**: *the accuracy and completeness with which users achieve specified goals.*
- **Efficiency**: *the resources expended in relation to the accuracy and completeness with which users achieve goals.*
- **Satisfaction**: *freedom from discomfort, and positive attitude to the use of the product.*

There are three factors [20] should be taken into account in this definition, which can significantly impact the overall usability of the system.

- **User**: *Person who interacts with the product.*
- **Goal**: *Intended outcome.*
- **Context of use**: *Users, tasks, equipment (hardware, software and materials), and the physical and social environments in which a product is used.*

From the definition, the usability in ISO 9241-11 is user-centered, particularly focusing on achieving the outcomes in the specific context. This can be extended to how successful users achieved the goals, how many resources are spent on achieving them, and how satisfied users think about the overall systems.

2.4.2 Usability Metrics

The usability metrics is a way to evaluate the user experience. Metrics provide the ability to quantify the usability when evaluating the system, which are extremely useful to allow the system to compare against benchmarks and identify which aspects needs to be improved. All desired Metrics must be observable and quantifiable [13].

Tullis and Albert [13] said that *"Confidence intervals are extremely* *valuable for any usability professional. A confidence interval is a range that estimates the true population value for a statistic."* Confidence interval are highly recommended to be used to qualify the results from small samples as it is able to give a more likely range of the unknown population. 95% confidence interval is widely used as when the treatment effect is significant at the 5%, the risk will be absolutely reduced [14]. In this study, 95% confidence interval will be applied in calculation of usability metrics from small samples.

Based on the Sauro's study, the most commonly used formula for calculating confidence interval is the Wald method as shown below [21].

$$95\%\ Confidence\ Interval = \hat{P} \pm Z_{\alpha/2}\sqrt{\hat{P}(1-\hat{P})/n}$$

(n is the sample size, z in 95% confidence interval is 1.96)

The following usability metrics that will be used to evaluate the MyVolts mobile application are discussed.

1) Effectiveness

According to the study from Tullis and Albert, the calculation of effectiveness can be done by measuring the task completion rate [13]. Completion Rate is typically recorded as Binary Success. It is the simplest and most appropriate approach to measure task success, users can either to complete a task successfully or not. A numeric number 0 or 1 in the form of 1's (Success) and 0's (Failure), which is given to record whether the users accomplished the task or not after the tasks being performed. The advantage of using a numeric score is that allows to calculate the percent correctly as well as other statistics that might be needed [13].

Wald method produces intervals that are too narrow to calculate completion rate when the sample sizes are small (n < 150) because small samples are easy to get the full or zero completion rate from the testers. Therefore, Sauro recommends a proposed method named "Adjusted Wald Method", which simply add two successes or two failures to the observed completion rate [15].

2) Efficiency

Efficiency is measured in terms of task completion time [9]. In usability evaluation, task times are commonly reported as a usability metric [10]. The mean for each task is widely picked as the value of the task times in usability. Based on the study from Sauro, either the mean or median can fairly serve as a statistic of central tendency when data are roughly symmetric. However, the arithmetic mean has a poor performance if there are unusually long task times [11]. He also recommends geometric mean as a standard method to calculate the central tendency for the small or average sample size (n <= 25) as it has a lower error rate through research and estimation [11].

For a set of numbers $\{x_i\}_{i=1}^{N}$, the geometric mean can be calculate using the following formula[4].

$$\left(\prod_{i=1}^{N} x_i\right)^{I/N} = \sqrt[n]{a_1 a_2 \dots a_n}$$

3) Satisfaction

SUS (System Usability Scale) provides the ability to measure usability generally across a broad range of contexts by investigating the users' subjective impression using questionnaire [12]. SUS supports a variety of aspects of system usability presented by ten items with alternate five positive and negatives ones. Each item is ranked from 0 to 4. Participants were asked to give responses through rating each item. SUS scores can be represented as percentage, this can be done by counting total scores from all questions and multiply total scores by 2.5 [37].

$$Total\ SUS\ Percentage = Total\ Score \times 2.5$$

4) User Error Occurrence Rate

Identifying how many errors made by users while interacting with the app is very useful to understand user performance and usability issues. Good usability implies a low error rate [15]. Error occurrence rate can be calculated in several ways based on the number of error opportunities in each task [16]. On the other word, the accuracy of the number of error opportunities can significantly impact the accuracy of the results. To calculate the error occurrence rate, the following

[4] http://www.statisticshowto.com/geometric-mean-2/

formula can be used [16].

$$Error\ Occurrence\ Rate = \frac{Total\ number\ of\ occurred\ errors\ for\ all\ users}{Total\ number\ of\ error\ opportunities\ for\ all\ users}$$

2.4.3 Qualitative Method

This section introduces the qualitative methods including lab-based experiments, web-based survey, usability problems, and user feedback that used to support the evaluation of the usability in this study.

1) Lab-based Experiments

Lab-based experiments apply task-based context approach which provides the advantage to allow different elements including user experience, user's behavior, user's subjective impression and so on to be integrated. Lab-based experiments are able to avoid problems that arise in field experiments [63] as the experiment is run in a controlled environment. And it is possible to employ facilities to collect high-quality data.

2) Web-based Survey

Nowadays, the internet has become a medium to conduct the research. Web-based survey consists of a set of customized questions. It allows a quick response to gather use feedback. Web-based survey can be able to potentially increase the sample size and reduce the cost.

3) Usability Problems

The usability problems bring negative impact to the overall usability of the system as it reduces the efficiency. Understanding the usability problems that users encountered when interacting with system plays an important role among different development stages. Each specific usability problem provides many useful information such as the frequency of this usability problem occurred, which users encountered it, and did they solve it. It can significantly help development team to identify which aspects can be improved to achieve higher usability.

4) User Feedback

User feedback allows quick responses from users on how they directly think about the system. As the potential users can be from different background with different level of skills, they could have different expectations of the system. Good user feedback is helpful to identify potential usability issues and improve the system usability in future. The way to gather user feedback can either ask users directly what they think about the systems after they completing the usability tasks or provide a list of meaningful questions.

2.4.4 Summary

This section introduced the usability model which will be applied across the whole design stage in order to improve the quality of the design. In addition, the appropriate usability metrics were also discussed to qualify the evaluation of the app usability.

2.4 Existing MyVolts Web Application

MyVolts web application offers users the ability to find the correct power supplies/cables. It provides the functionality to allow users to find the matched power supply/cable based on their device information by entering queries in the search box. In order to enable the different ways to search, MyVolts web application also implemented power supply/cable finder which allows user to search by manufacturer, device, and model reference by simply selecting from drop down boxes. The drop down boxes include all related products they are offering. Figure 2.4 presents the screenshot of MyVolts web application *search* page.

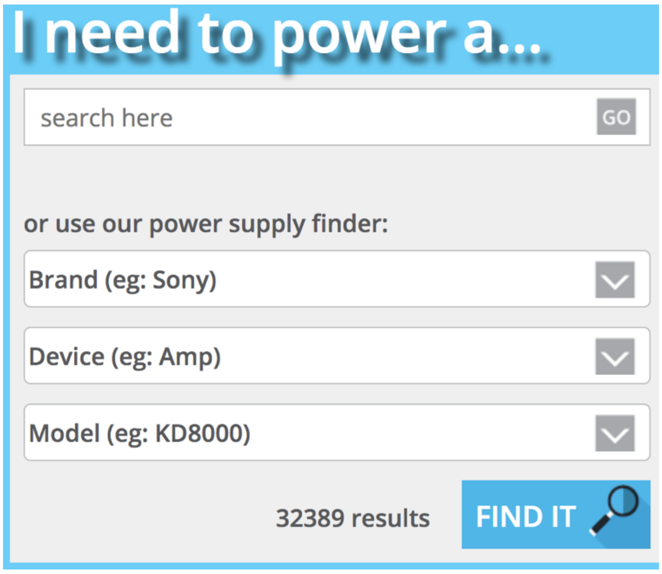

Figure 2.4: MyVolts Web Application Search Page.

The *search results* page displays all related results. It provides the functionality to allows users to refine the search. In addition, users are able to sort by categories, powering solution, devices, or models. Figure 2.5 presents the screenshot of MyVolts web application *search results* page.

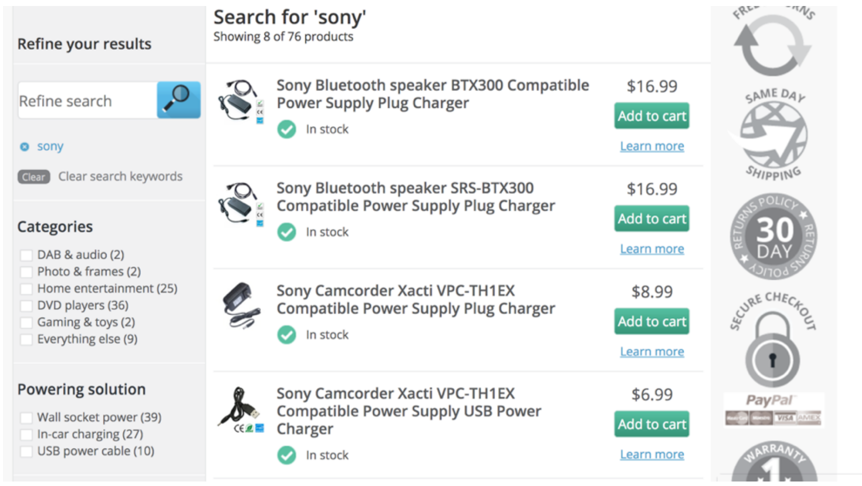

Figure 2.5: MyVolts Web Application Search Results Page.

The *product description* page displays the details of each specific product including technique specification, requirements, warranty and so on. Figure 2.6 presents the screenshot of MyVolts web application *product description* page.

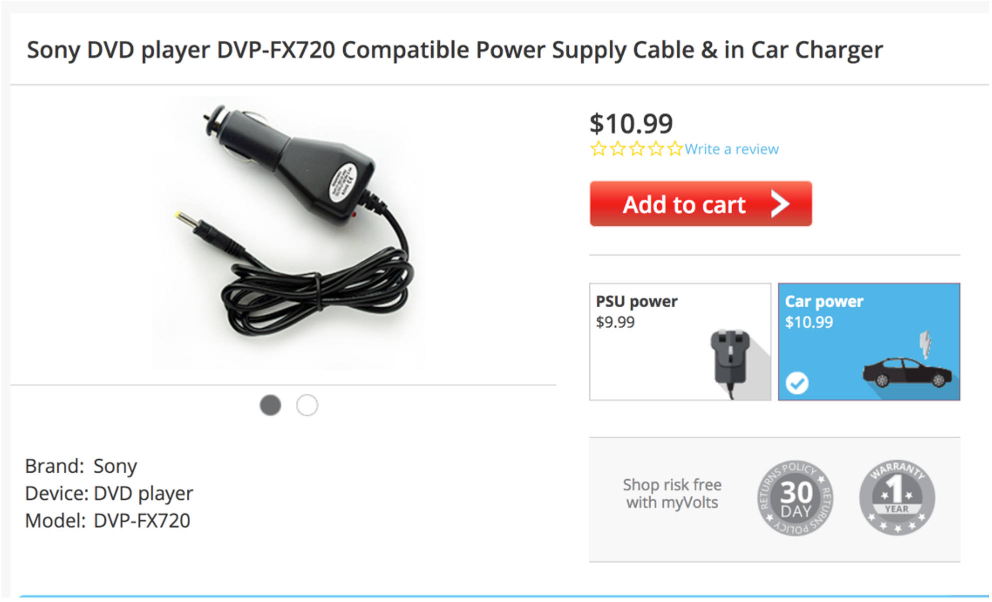

Figure 2.6: MyVolts Web Application Product Description Page.

2.6 Chapter Summary

This chapter reviewed the basic Linked Data technologies, HCI, mobile user interface design principles, and evaluation metrics, which provides the fundamental knowledge of the design of the mobile application. In addition, the Linked Data challenges and mobile usability challenges were also address, which requires to present the solution to resolve in design chapter (Chapter 4). The next chapter will review the state of the art Linked Data mobile applications and search user interface design patterns.

Chapter 3

State of the Art

This chapter focuses on analyzing the research of existing Linked Data mobile applications and user interface search design pattern guidelines. This will be used to support the design of MyVolts mobile application technical architecture and user interface by identifying search user interface design patterns and requirements.

3.1 Linked Data Mobile Application

To date, here are few examples of Linked Data mobile applications. In this section, the current technical approaches in relation to the design and implementation of Linked Data mobile applications are discussed and compared.

3.1.1 DBpedia Model

DBpedia Mobile [45] is a location-centric client application which extracts information from DBpedia [52] and presents them on the map. DBpedia Mobile allows the user to explore background information of nearby locations by interacting with a map using a mobile device. The data in DBpedia about these locations are interlinked with other related datasets such as GeoNames[5].

The initial view of DBpedia is presented with a map view which contains information of nearby locations with corresponding labels and icons based on the user's position. A Fresnel [46] based Linked Data browser is provided to display a summary view of the selected resource when user clicking an item on the map. There are also links at the bottom of the abstract to allow the user to switch to the alternative photo view and view all available properties for the resource, which include information from related resources.

DBpedia Mobile is built on a client-server architecture, which provides the functionality for the client application to search, retrieve, and store data. This can be done by the client sending a SPARQL query with the user's current location, language, and any other filter settings to the server. And then the server returns data with RDF triples.

3.1.2 mSpace Model

mSpace Mobile [47] is a Semantic Web application that provides relevant information of in a selected knowledge domain to mobile users based on their location. It builds on a three-layer architecture including the mSpace Mobile Application (MA), the mSpace Query Service (MQ), and RDF triplestore knowledge interfaces (MK). The responsibility of the MK is to manage data from multiple service providers easily. The MQ can be referred to as the repository, which provides the ability to connect the MA and the MKs. The MA accesses the knowledge of domains and relevant information by querying the MQ using Simple Object Access Protocol (SOAP) and HTTP. The MK is therefore interacting with a specified triplestore using RDF Data Query Language (RDQL) [48].

The limitations of mSpace Mobile application can be lack of data sources, trustability of data sources, and issues of real mobility.

3.1.3 OntoWiki Model

OntoWiki Mobile [5] is an open source HTML5 application which supports a wide range of usages including knowledge acquisition in a collaborative way and

[5] http://www.geonames.org/

browsing of semantic data. OntoWiki follows the standard MVC architecture (Model-View-Controller) which consists of persistence layer, application layer and interface layer. The OntoWiki framework uses the Erfurt API[6] which provides the functionality to allow access on the different RDF stores. OntoWiki provides offline usage by implementing a persistent local storage through caching data at the client side, which aims to improve app performance by reducing page loading time. OntoWiki also uses RDFautor to support resource editing. Data replication and conflict resolution mechanism is also implemented to support concurrent edits of the same resource.

3.1.4 Who's Who

Who's Who [50] provides the functionality to allow users to access the Linked Data cloud by exploring contextual information and visualize Linked Data through linking the physical world with the virtual. Who's Who improves the performance to resolve low bandwidth and network connection issues by enhancing the processing of sematic and Linked Data. It also provides the solution to resolve the latency problem by including a mobile browser's light-weight triple store within the application. Who's Who uses novel visualization strategies to overcome the usability challenges, which allows non-expert access to Linked Data.

Who's Who uses a client-server architecture. The data communication can be established by the client-side creating the SPARQL query and send to the server-side. The server-side executes the query and parsed in JSON format, and then returns the result with a JavaScript callback to the client. The client retrieves and stores the result triples into a local RDF triple store. Then the interaction can be done by executing local SPARQL queries for the further filter from the available cached information.

The advantages of this approach are: (1) users don't need to download the application; (2) users don't need to know the URI of any physical entity; (3) there is a balance of the triple load between the server and mobile client sides; (4) the mobile client allows users to filter the data locally, which reduces latency in order to improve the usability of the interface.

---

[6] http://aksw.org/Projects/Erfurt.html

3.1.5 GetThere

GetThere [51] provides a real-time information about public transport for rural areas, which allows users to plan their journeys. GetThere followed client-server architecture. The client side invokes web services, which execute SPARQL queries against the dataset. The back-end server then sends a response to the client with a list of available routes.

3.1.6 Feature and Architecture Comparison between Linked Data Mobile Applications

Table 3.1 illustrates the comparison of different features between Linked Data mobile applications. Through comparing technical approaches between different Linked Data mobile applications, it has been shown that all presented Linked Data mobile applications adopted the client-server approach. The client side is responsible for visualizing data into the user interface by querying the server using SPARQL over HTTP. This approach will also be adopted by MyVolts mobile application as the client-server approach is most widely used.

Among the reviewed applications, OntoWiki mobile and Who's Who supports the local storage and filters in order to improve the performance of the mobile applications by resolving the latency and network unreachable problems. The advantage of these designs is to allow users to interact with the mobile applications under the unreliable network conditions, which aims to improve the usability of the mobile applications. OntoWiki mobile also supports offline usage, which enables users to use the application without network coverage.

However, considered the mobile usability constraints as discussed in section 2.3.2, the mobile memory is limited, so a large amount of local storage and filters should be avoided when designing the MyVolts mobile application. Offline mobile applications could take several minutes for initialization depending on the amount of data. In addition, an offline search may not be able to locate the expected record even if it exists, which could decrease the effectiveness of the mobile applications. Therefore, this approach may not be appropriate to the MyVolts project.

Table 3.1: Comparison of the Different Linked Data Mobile Applications

Features	DBpedia Mobile	mSpace Mobile	OntoWiki Mobile	Who's Who	GetThere	MyVolts Mobile
Client-Server	Yes	Yes	Yes	Yes	Yes	Yes
SPARQL	Yes	Yes	Yes	Yes	Yes	Yes
Local Storage & Filters	No	No	Yes	Yes	No	Partial
Offline Usage	No	No	Yes	No	No	No

3.1.7 Summary

This section compared the different technical approaches between Linked Data mobile applications. As discussed in section 3.1.6, It has been decided that MyVolts mobile application will adopt the client-server architecture. Considering mobile usability constraints, the MyVolts mobile application will not designed for offline usage, and will only consider a small amount of data to be stored in the local database if there is a necessity.

3.2 Search User Interface Design Patterns

Search is the fundamental activity of the mobile application, good search patterns can allow users to search efficiently and effectively [61]. So there is no doubt that it is beneficial to review some best practices regarding to the mobile search design patterns. Through browsing from Google Scholar[7], there are three top search design guidelines regarding mobile user interface design. This section discusses and compares search design patterns in relation to mobile interface design from these three guidelines including Nudelman's android search design patterns [54], Neil's search design patterns [55], and Morvillie & Callender's search design patterns [56]. First the search patterns from each source are summarized and then an analysis is presented.

3.2.1 Nudelman's Search Design Patterns

As the MyVolts mobile app was designed to build on the Android platform, reviewing design patterns in relation to the Android development can significantly benefit the future development from design and technical perspective. Nudelman

[7] https://scholar.google.com/

introduced design patterns based on the official Google Android design guidelines[8] and took advantages from a set of best practices. Table 3.2 illustrated the most valuable patterns when designing the search user interface introduced by Nudelman.

Table 3.2: Description of Nudelman's Android Search Design Patterns

Design Patterns	Description
Voice Search	Audio query inputted via an on-board microphone is used as input for searching instead of a keyword query.
Auto-Completion	Auto-Complete and Auto-Suggest, are broad classifications of keyword-entry helper patterns.
Tap-Ahead	Tap-Ahead implements auto-suggest one word at a time, through step-wise refinement, creating a kind of *keyword browsing*.
Pull to Refresh	Search results are refreshed when the customer swipes down (pulls down) on the results.
Search from Menu	Search is an option that can be accessed from the navigation bar menu.
Search from Action Bar	The customer can access search via a dedicated button on the app's action bar.
Dedicated Search	The search box is placed on top of the search results and does not scroll with them.
Search in the Content Page	The search box is on top of the search results and part of the content page, so it scrolls with the rest of the content.

1) Voice Search

Audio query input is an alternative option to search. The user speaks via an on-board microphone, then the audio query is captured and transcribed into a text-based keyword query. Many mobile search is done when users are walking or in the multitasking context such as driving, so using text input at that time can be awful and increase error rate. In addition, voice search can also solve many usability problems such as fat finger problem.

[8] http://developer.android.com/design/get-started/ui-overview.html

2) Auto-Complete and Auto-Suggest

Auto-Complete and Auto-Suggest provide the functionality to produce a set of suggestions when the user enters one or more characters into the input field, which reduces the effort of user typing and the number of entry errors. The suggestions filed contains a set of queries related to the user input. The difference between Auto-Complete and Auto-Suggest is that Auto-Complete produces queries which contains the original keyword from the user. Considering the potential entry errors made by users, Auto-Suggest provides additional functionalities including spelling corrections, keyword substitutions, query expansions, and so on.

3) Tap-Ahead

The Tap-Ahead pattern is actually related to the Auto-Complete and Auto-Suggest pattern. Tap-Ahead supports the suggestions filed by providing a guessing process, while Auto-Complete and Auto-Suggest aims only to guess the entire query the user is going to type. When the user enters one or more characters, the suggestions field offers two options: (1) Select the query if it sufficiently matches what the user is looking for. (2) Tap the diagonal arrow icon on the right side of the query to populate that query to the search box, and continue the Auto-Complete/Auto-Suggest function. This process offers much more flexible and efficient approach to improve the user experience on mobile devices by significantly solve usability issues. Users can quickly access a lot of keywords combination by inputting a few characters.

4) Pull to Refresh

The Pull to Refresh pattern provides a lazy approach to refresh new updates. As considering the list of resources can be typically represented order by time, the most recent posts are displayed first, it is important to provide an efficient way to allow users to view updates. When users want to view the most recent updates, they will naturally scroll up the page to the top. Therefore, using a scroll-up gesture instead of button clicking is able to get a quick response of updating.

5) Search Options

Searching from the navigation bar is now deprecated and recommended for the older (older than Android 4.0) Android OS implementations as the user has to tap the menu button to select the search option. For the newer Android OS implementations, the user can search by clicking the dedicated button on the mobile app's action bar. The search button normally formed as a standard magnifying glass icon. After the user clicks the search button, they will be forwarded to the search page with several optional valuable features such as popular searches.

6) Dedicated Search

Users generally like to refine the search rapidly. When users scroll down too much, they may feel very inconvenient to refine the search if there are no quick options. Dedicated search provides the functionality to allows users to refine the search easily by placing the search button or box on the top of the search results and scrolling without them.

7) Search in the Content Page

This pattern is an alternative option of the Dedicated Search pattern. In this approach, the search box is a part of the content page. So the user scrolls the content page down with the search box, and must scroll up to the top of the page to refine the search. This pattern is very popular in iOS design but very seldom in the Android design.

3.2.2 Neil's Search Design Patterns

Neil introduced mobile search design patterns in his book *Mobile Design Pattern Gallery: UI Patterns for Smartphone Apps.* Neil thought as more awareness of people's individual needs, mobile applications can reduce instances where people have to explicitly search for what's relevant to themselves. Therefore, he introduced a set of search design patterns which are more customized and able to improve the usability of search functions. Neil also agreed the auto-complete pattern is the most useful search pattern. The rest search patterns as shown following.

1) Dynamic Search

Dynamic search pattern also is referred to as dynamic filtering. The mobile application will filter the data dynamically when the user entering the text in the input filed. The dynamic search pattern is generally used to refine a set of constraint data set such as a contact library, it is not appropriate to be used to filter a large scale data sets.

2) **Scoped Search**

Scoped search pattern allows users to get the desired results efficiently by providing a set of the search criteria to scope before displaying the results. The scoping options should be reasonable considered based on the data set.

3) **User's Effort**

Good mobile interfaces should respect the users' effort. The saved and recent search patterns allow users to select from their search records, instead of repeating the same typing. Other options including location based searching, bar code searching, and voice searching can also significantly reduce the users' search effort.

4) **Search Form**

This pattern is presented as a separate form allowing users to enter search criteria based on the data set in order to filter search results. This pattern should consider to minimize the number of input fields to make users type or select quickly.

5) **Lazy Loading**

This pattern is very similar as the pull to refresh pattern. The difference is that the lazy loading pattern is widely used to display more results on the screen by either providing a button to click or automatically load more results. Lazy loading should be recommended to be applied instead of paging on a mobile device.

3.2.3 Morvillie & Callender's Search Design Patterns

Morvillie and Callender recommended a set of search design patterns which are intelligent and personalized to improve the quality of users' searching. They were not only simply identifying and describing the value of each patter, but also investigated how each pattern can be related to other patterns.

Morvillie and Callender also highly recommended auto-complete pattern, they also motioned that it is worth to tune suggestions to a specific category. In addition, they agreed that lazy loading pattern plays an important role for delivering results, and applying flexible sort and filter options can be able to avoid infinite loading. The following list displays the rest of search patterns recommended by Morvillie and Callender.

1) Best First

Best First is the most important design pattern in search. The first few results will draw most of the attention, Users typically expect results that are up to date, fashionable, and relevant. Therefore, displaying the best results is significant to make search simply and fast. Users just need to enter queries, look the first few results, and click the desired result.

2) Personalization

It is often useful to provide query suggestions using search history as repeating queries has become very common in many search contexts. However, simple solutions are already well presented. Personalization is to provide suggestions based on the user's past search queries and behaviors, which aims to predict users' interests and expectations. Often, personalization can work well with other search patters, for example, it can inform the algorithms of best first.

3) Faceted Navigation

Faceted navigation is a modern pattern. The implementation of faceted navigation brings impact to all search patterns and the whole information architecture. The faceted navigation model provides visible options to users for clarifying and refining search queries. It offers an integrated search experience which allows users to search with a keyword and the scan a list of results.

4) Structured Results

Structured results should provide the ability to remodel the search results. The optimal format of the results to embed rich snippets through digging deeper into the data. For example, summarizing results and representing them as a chart or graph. It is often better to present a picture for a thousand words or take the least step to allow users to understand the data well. This pattern plays well with best first and could benefit from personalization.

3.2.4 Analysis

This section compares and discusses the search user interface design patterns from three design patterns as introduced above. As these design patterns may have different terminologies for the same pattern, so the terminology for each specific pattern is unified first. Then the popularity of design patterns and how these patterns can or cannot be applied to the implementation and design of the MyVolts mobile application are analyzed.

1) Terminological Unification

Table 3.3 presents the terminological unification for search user interface design patterns.

Table 3.3: Terminological Unification for Search User Interface Design Patterns

Terminology	Nudelman's	Neil's	Morvillie & Callender's
Auto-Completion	Auto-Completion	Auto-Completion	Auto-Completion
Pull to Refresh	Pull to Refresh	N/A	N/A
Tap-Ahead	Tap-Ahead	N/A	N/A
Voice Search	Voice Search	User's Effort	N/A
Barcode Search	N/A	User's Effort	N/A
Search History	N/A	User's Effort	Personalization
Search Form	N/A	Search Form	N/A
Explicit Search Options	Search Options	Explicit Search Option	N/A
Refine Search	Dedicated Search	Dynamic Search	Faceted Navigation
Scoped Search	N/A	Scoped Search	N/A
Lazy Loading	N/A	Lazy Loading	Lazy Loading
Personalization	N/A	N/A	Personalization
Best First	N/A	N/A	Best First
Faceted Navigation	N/A	N/A	Faceted Navigation
Structured Results	N/A	N/A	Structured Results

The following list presents the terminological unification for the overlapped patterns.

- **Auto-Completion**

  All three design patterns have the same terminology about the auto-completion.
- **Voice Search**

  Nudelman introduced voice search, while Neil also categorized it as user's effort.
- **Search History**

  Neil included search history as the part of the user's effort. Morvillie and Callender also mentioned search history in the personalization pattern.
- **Explicit Search Options**

  Nulelman and Neil both mentioned the design of explicit search options.
- **Refine Search**

Dedicated search pattern and dynamic search pattern are both referred to refine search, and Morvillie and Callender also mentioned refine search in the faceted navigation pattern.

2) Popularity Analysis

Table 3.4 illustrates the comparison of popularity between different search UI design patterns. The comparison result clearly shows that the auto-complete pattern and refine search pattern were highly recommended from all three guidelines to be applied into the design of mobile search feature.

Voice search pattern, search history pattern, and explicit search options patterns are also popular patterns which were recommended by 2 search UI design patterns. And the rest are recommended by the single search UI design pattern.

Table 3.4: Popularity of Design Patterns

Patterns	Popularity
Auto-Completion	3
Pull to Refresh	1
Tap-Ahead	1
Voice Search	2
Barcode Search	1
Search History	2
Search Form	1
Explicit Search Options	2
Refine Search	3
Scoped Search	1
Lazy Loading	1
Personalization	1
Best First	1
Faceted Navigation	1
Structured Results	1

3) Identification of Patterns to Be Used for MyVolts

The following list discusses how each pattern can or cannot be applied to the MyVolts mobile application.

- **Auto-complete & Refine Search**

  Auto-complete pattern and refine search pattern are most popular search user interface design patterns, so they are selected for the MyVolts mobile application.

- **Lazy Loading**

  Pull to refresh pattern and lazy pattern are actually very similar, they are both providing the functionality to allow the user to automatically update or load more results without a clicking button. However, pull to refresh pattern may be only suitable to be designed in the social media mobile application or any others mobile applications which require high timeliness. Lazy loading pattern can be widely used in any type of mobile applications to allow results to be displayed progressively and immediately. So the MyVolts app will only implement the lazy loading pattern.
- **Voice Search**

  As the mobile usability challenge of data entry method was addressed in section 2.3.2, it is important to implement different data entry methods. Voice search pattern allows users to search on the move, so it is able to resolve the usability challenge and improve the usability of MyVolts mobile application.
- **Barcode Search**

  Barcode search allows quick responses as they can be read quickly by a smart phone, which can significantly improve the efficiency of search. Therefore, MyVolts mobile application will implement voice search pattern and barcode search pattern as alternative ways for users to search.
- **Tap-ahead**

  Nudelman recommended tap-ahead pattern. This pattern could work well if there are many different categories of the product because it is able to improve the search efficiency by instructing users to construct their queries. However, MyVolts mobile application only focuses on selling power supplies, so it is not necessary to implement the tap-ahead pattern as it may make the auto-complete list more complicated and confuse users. In addition, the diagonal arrow icon on the right side of the query may also confuse users and let them think it is just a sign to search directly.
- **Search Form & Faceted Navigation & Scoped Search**

  Search form pattern, faceted navigation pattern, and scoped search pattern are more complicated and useful for filtering with multiple conditions, which is not applied the MyVolts mobile application.

- **Explicit Search Options**

  As considered the mobile application will be implemented on the android platform, it should follow the android design guidelines and appropriately implement the search option on the action bar.

- **Personalization & Best First**

  Personalization and best first pattern use the intelligent way to improve the quality of the search results. However, it requires the algorithms to be implemented on the server side, which is not a case in the current project.

- **Structured Results**

  Considered the simplicity principles as discussed in section 2.3.1, the structured results pattern should be applied when designing the search result list. And each product picture should be presented to allow users to verify the product information quickly and simply.

- **Best-selling**

  However, no these search design guidelines indicated the best-selling pattern. As recreational users are attracted to popular trends, implementing best-selling pattern is able to instruct them with the highest rated brands or items [53]. This pattern can be applied into MyVolts mobile application because it can help users especially for people without rich experience on power supply to know which products are most popular.

Through comparing and analyzing three design pattern guidelines, the right column in table 3.5 presents the design patterns which are selected for the MyVolts mobile application.

Table 3.5: Comparison of Search UI Design Patterns from Literature and Identification of Patterns Selected for the MyVolts Mobile App

Patterns	Nudelman's	Neil's	Morvillie & Callender's	MyVolts
Auto-Completion	Yes	Yes	Yes	Yes
Pull to Refresh	Yes	No	No	No
Tap-Ahead	Yes	No	No	No
Voice Search	Yes	Yes	No	Yes
Barcode Search	No	Yes	No	Yes
Search History	No	Yes	Yes	Yes
Search Form	No	Yes	No	No
Explicit Search Options	Yes	Yes	No	Yes
Refine Search	Yes	Yes	Yes	Yes
Scoped Search	No	Yes	No	No
Lazy Loading	No	Yes	Yes	Yes
Personalization	No	No	Yes	No
Best First	No	No	Yes	No
Faceted Navigation	No	No	Yes	No
Structured Results	No	No	Yes	Yes
Best-selling	No	No	No	Yes

3.2.5 Summary

This section introduced three different search pattern guidelines. Through comparing an analyzing, the search patterns including auto-completion, voice search, explicit search option, refine search, lazy loading, and structured results will be applied to the MyVolts mobile application design.

3.3 Chapter Summary

This chapter reviewed the existing Linked Data mobile applications, through comparing and discussing, the technical architecture was decided for further implementation of MyVolts mobile application. In addition, three search design pattern guidelines were introduced, later comparison and analysis were presented in order to decide which search patterns can be applied to the design of the user interface for MyVolts mobile application.

Chapter 4 Design

The purpose of this chapter is to present the design of the MyVolts mobile application which has been created to investigate the research question. The design of the application user interface followed the Constantine & Lockwood's 6 user interface design principles (Section 2.3.1). Considering the Linked Data challenges (Section 2.1.2) and mobile usability challenges (Section 2.3.2), this chapter also presents a proposed solution in response to the addressed challenges.

The structure of this chapter is divided into three sections, which presents the system requirements (Section 4.1), technical approach (Section 4.2), and interative user interface design (Section 4.3).

4.1 System Requirements

This section presents a system use case diagram and discusses the major system requirements which will be designed and implemented to the MyVolts mobile application. The MyVolts use cases were established in conjunction with MyVolts Ltd, so it was ensured that the major functionalities from functions already present in the MyVolts web application were covered.

Figure 4.1 presents the MyVolts mobile application use case diagram. The desired function of the MyVolts mobile application is to match product, so it has included the basic functionalities associated with the MyVolts web application including search and results display functions. Apart from the basic functions, through reviewing the state of the art design patterns, the alternative search methods were decided to be implemented into the MyVolts mobile application including voice search and barcode search.

The application should also implement a functionality to track users' searching history and display their recently viewed items so that they can recall them easily. In addition, users may want to know which manufacturers are most popular, so it is also worth to provide a rich experience for the users which allows users to select from hot manufacturers.

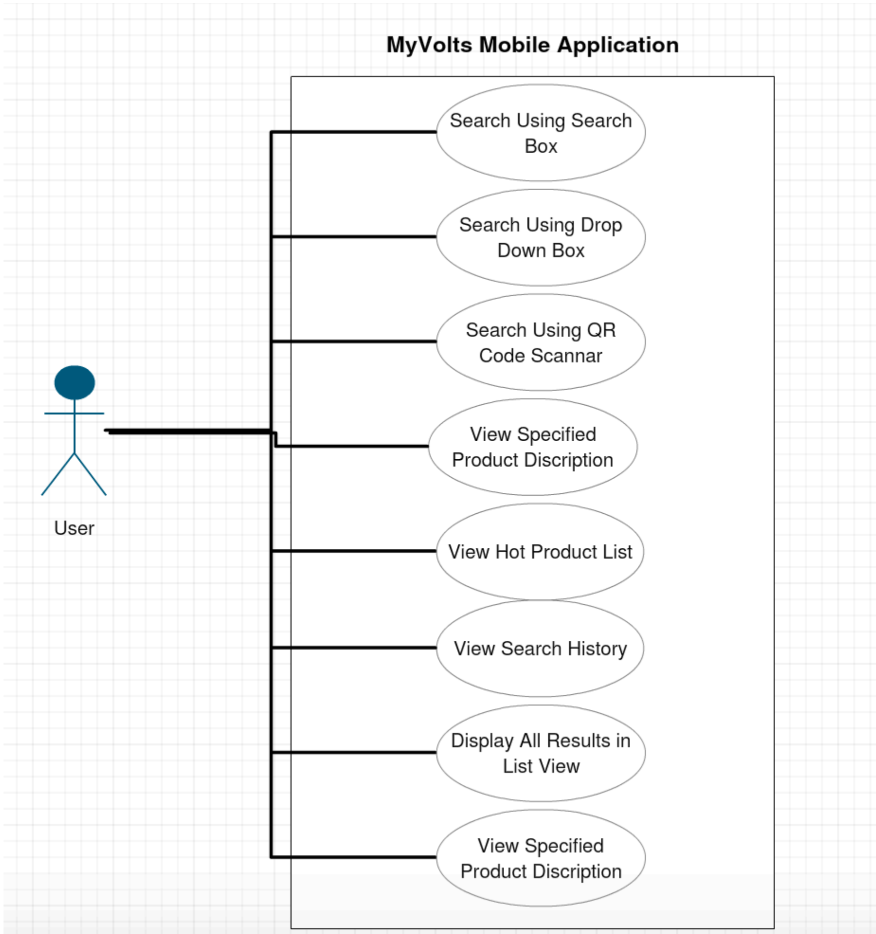

Figure 4.1: MyVolts Use Case Diagram

Based on the use case diagram, discussions with MyVolts and the literature review in Chapter 3, the following requirements were established for the MyVolts mobile application. It was important to interact with the MyVolts system, users should be able to perform without knowing the underlying technologies including RDF and SPARQL. Hence, the following requirements were posited.

REQ 1: Access MyVolts dataset through the remote SPARQL endpoint.

REQ 2: Provide a method to filter MyVolts dataset to match user request and display the search results in the list.

REQ 3: Provide a method to allow users to search by search box.
REQ 4: Provide a method to search by drop down box.
REQ 5: Provide a method to search by voice recognition.
REQ 6: Provide a method to search by QR Code.
REQ 7: Provide a method to display product description specified by user selection.
REQ 8: Be usable to common users, not only expert.

According to the Linked Data challenges, the following requirements were taken into account when designing the MyVolts mobile application.
REQ 9: Avoiding to frequently query the SPARQL endpoints when visualizing Linked Data into the single user interface. Try to implement asynchronous methods instead or provide the link widgets which allow users to click and load information.
REQ 10: As the Linked Data dataset can be changed frequently, try to implement an alternative method to allow users to search the real time information as well when there is a need to do a local cache.

Based on the addressed mobile usability challenges, the following requirements were posited.
REQ 9: Providing necessary feedbacks in relation to network status.
REQ 10: Using appropriate widget to display appropriate amount of information.
REQ 11: Using different level of resolution elements for different types of screens.
REQ 12: Implementing different user input methods to support different types of user activities.

4.2 Technical Approach

This section presents a technical approach in relation to the overall system design. It includes overall system architecture, data retrieval and storage, and screen revolution.

4.2.1 System Architecture

Figure 4.2 presents the overall system architecture of MyVolts mobile application.

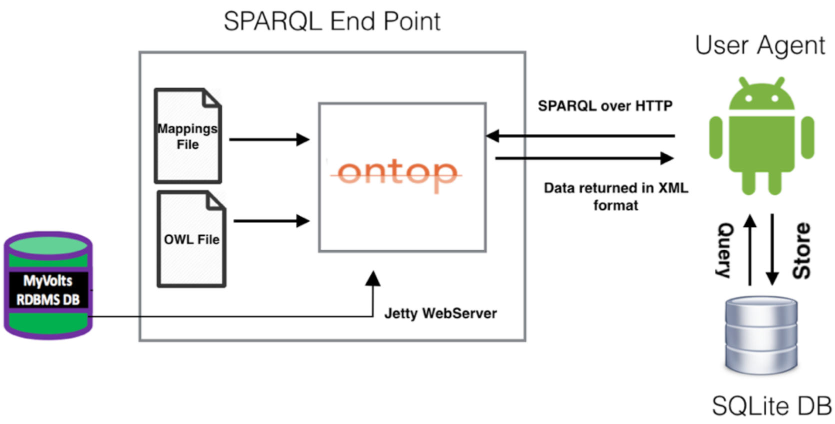

Figure 4.2: MyVolts Overall System Architecture

The MyVolts Linked Data dataset is hosted in Theme E of the ADAPT Research Centre. MyVolts system was decided to build on a client-server architecture as discussed in section 3.1.6. The MyVolts mobile application queries the server (remote SPARQL endpoint) by sending a HTTP request. The server obtains the request and retrieves data from MyVolts relational database using ontop, and then issues a HTTP Get back to the client with the response described in XML format.

For further details on the app architecture see section 4.3.1.2.

4.2.2 Data Retrieval and Storage

The MyVolts application retrieves data by running a SPARQL query over HTTP. Based on design requirements, the query may be comprised of a set of filters to satisfy search results. The application also contains a local SQLite database which used to cache information when necessarily.

All of the functionalities provided in this application are searching and browsing, so SPARQL SELECT queries were created to retrieve information from remote endpoints. Figure 4.3 shows an example to use SPARQL SELECT queries. The FILTER keyword was used to filter the results based on the different user purpose. And the LIMIT keyword was used to limit the range of results output.

```
PREFIX: <http://myvolts.com#>
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
SELECT ?prod_id ?name  WHERE {
?prod_id :product_name ?name .
}
```

Figure 4.3: SPARQL Query for Retrieving All Product IDs and Names

4.2.3 Major System Components

Figure 4.4 presents the implementation architecture of the MyVolts mobile application.

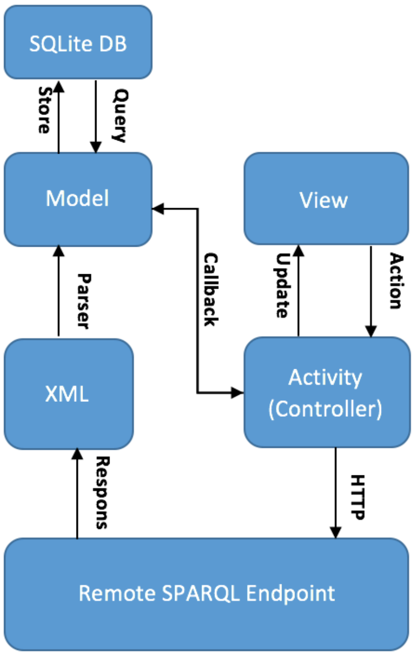

Figure 4.4: MyVolts Mobile App Implementation Architecture.

The MyVolts system was implemented using MVC (Model View Controller) architecture. The *model* component is responsible for managing the business logic. In MyVolts project, the *model* component is working on handling network status and data from HTTP response or data from local SQLite database. The *view* component is responsible for handling user's input or output and visualizing data for the updates from the *model* component. The *controller* component contains the logical functionalities of the system, which is responsible for handling users' actions. The implementation for each user interface of prototype 1 are discussed as following.

4.3 Mobile Application User Interface Design

This section introduces the user interface design and implementation approach. The approach consists of an initial prototype and a final version prototype.

4.3.1 Prototype 1

Prototype 1 was the initial prototype designed for gathering the early feedback. It consists of the query user interfaces which allow users to construct their search queries and the following requirements based on the initial requirements were posited.

REQ 3 & 4 & 5 & 6: Provide a set of predefined options to allow users to select a search method. This should be visible to users.

REQ 2a: For the user selecting search box, it should provide the auto-completion function.

REQ 2b: For the user selecting search box, it should provide the voice search function.

REQ 2c: For the user selecting search box, it should allow them to search from popular list and search history list. These two lists should be visible and clickable to users.

REQ 4a: For the user selecting drop down box, it should enable a quick response to present the option list for each box.

4.3.1.1 Paper Prototyping

This section presents the paper prototyping for prototype 1 and design of different of search patterns.

Home User Interface

Home user interface was designed for allowing users to make option from three different search methods including search by search box, search by drop down search box, and search by QR code scanner. Figure 4.5 presents the paper prototyping for *home* user interface 1 and figure 4.6 presents the paper prototyping for *home* user interface 2.

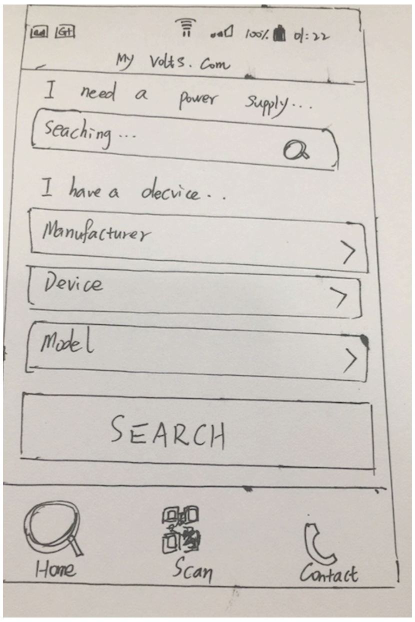

Figure 4.5: Paper Prototype for Home UI 1.

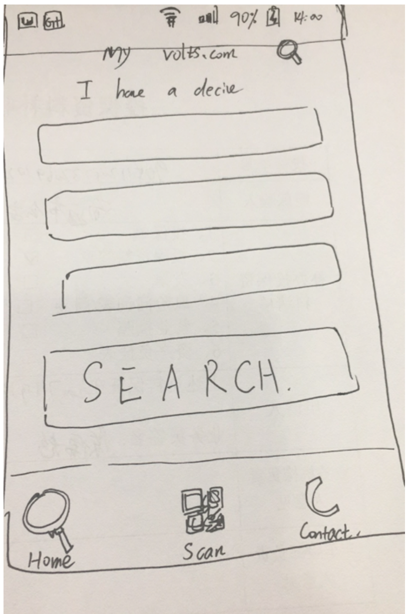

Figure 4.6: Paper Prototype for Home UI 2.

The drop down search box allows users to search by manufacturer, device, and model reference, this can be done by selecting the target item from each select list. Therefore, three select boxes were designed. In order to make select boxes clickable, right arrow icon was designed to accompany with each select box. The QR code scanner option was designed to be placed on the horizontal navigation bar, which aims to make the option more clickable and visible to users.

However, there are two design options for designing search box. One option is to design the home user interface as similar as MyVolts web app. The other option is to following android design pattern guidelines as discussed in section 3.2.1, which recommends to place the search box option on the action bar. Considered from the HCI interaction model as discussed in section 2.2.2, users rely on the previous experience from similar apps or any other available information. The design option 1 was decided to be applied to the implementation of the home user interface.

Search User Interface

The search user interface was designed for users who choose to search products using search box. It consists of a search box, a hot list, and a search history list. Figure 4.7 presents the paper prototyping for *search* user interface 1 and figure 4.8 presents the paper prototyping for *search* user interface 2.

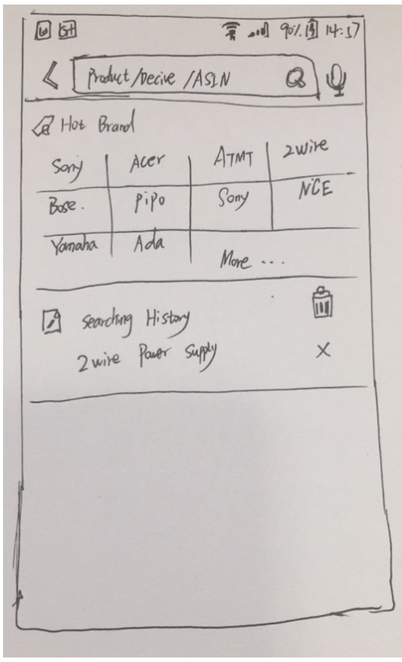

Figure 4.7: Paper Prototype for Search UI 1.

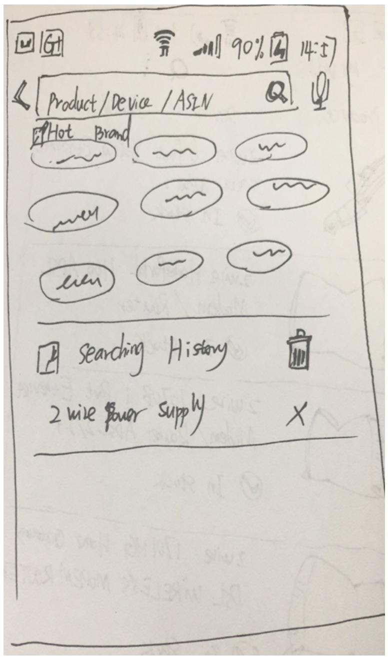

Figure 4.8: Paper Prototype for Search UI 2.

The voice search button was designed to be placed near the search box, which aims to be more clickable and visible to users. The search history list was considered to be implemented with the functions which allow users to remove a single item or all items at once. This aims to allow users to control the items. There are two options for designing hot list. One is to implement grid view, and the other is to use tag cloud approach. However, through the research, it was found nielson[9] indicated "*Tag clouds were a huge fad in 2009 and have actually been a fad for several years. Even so, usability studies show that most normal users don't know what they are and don't know how to deal with them.*" So considered the potential usability issue from tag cloud approach, the grid view was adopted to implement the hot list.

[9] https://www.nngroup.com/articles/tag-cloud-examples/

4.3.1.2 Prototype 1 Implementation

Home User Interface

Users normally want to get immediate response for options, so in order to improve efficiency of each select box, all of options were cached and stored in the local SQLite database. Each select box provides the responsive functionality based on the selection of another select box. The open source image processing library Google ZXing[10] (Zebra Crossing) was applied to implement the QR code scanner. Considered each QR code has to represent a unique product, so the ASIN (Amazon Standard Identification Number) was decided to be encoded as the QR code content. After the decoding of the QR code, the *controller* will get the notification and send the HTTP request to the remote SPARQL endpoint with the ASIN number in order to get the matched product information. Figure 4.9 represents the screenshot of the *home* user interface.

Search User Interface

A good auto-suggestions have to consider to reduce the user input effort, so it is necessary to provide suggestions as quickly as possible. However, it normally takes time to communicate with the server via network, so product names were cached into the local database initially. The RecognizerIntent[11] API was decided to implement voice search, which provides constants for supporting speech recognition through starting an intent using Google server. Figure 4.10 presents the screenshot of the *search* user interface.

[10] https://zxing.github.io/zxing/apidocs/
[11] https://developer.android.com/reference/android/speech/RecognizerIntent.html

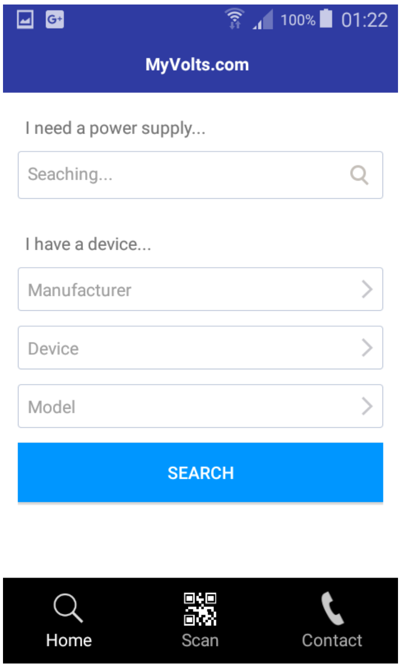

Figure 4.9: Screenshot of the Home UI.

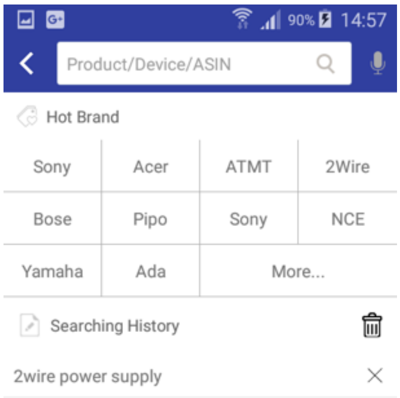

Figure 4.10: Screenshot of the Search UI.

4.3.2 Prototype 2

Prototype 2 was a completed version of MyVolts mobile application. It was designed to achieve full functionalities as well as improvements from prototype 1. This version added another two user interfaces including result list user interface and product description user interface, which provide they ability to allow users to view all matched products and details. As the MyVolts mobile applications was designed to retrieve information from the remote SPARQL endpoint via networks, it also considered to provide appropriate feedbacks for situations in relation to network unreachability. The following requirements were posited.

REQ 2a: Provide an option to allow users to refine the search.

REQ 2b: Provide a method of pagination and lazy loading, this should control the speed of the presenting of upcoming items.

REQ 2c: Provide an option to allow users to back to the top of the list in the results list screen.

4.3.2.1 Paper Prototyping

This section presents the paper prototyping for prototype 2 and design of different of search patterns.

Results List User Interface

The result list user interface was designed for displaying user search results. Figure 4.11 presents the paper prototyping for *search* user interface 1 and figure 4.12 presents the paper prototyping for *results list* user interface 2.

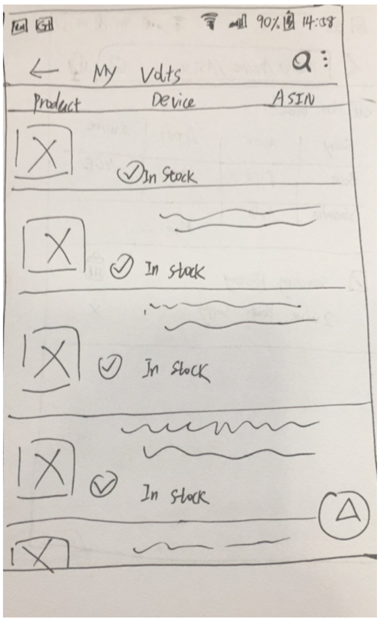

Figure 4.11: Paper Prototype for Results List UI 1.

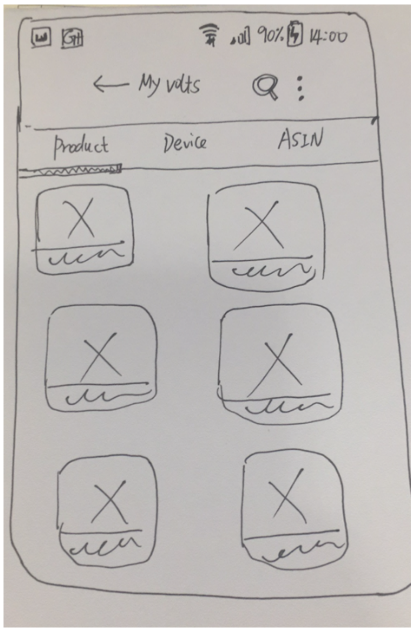

Figure 4.12: Paper Prototype for Results List UI 2.

There were two designs for displaying search results. One is to present results on the list view, and the other is grid view. A key factor to make a decision between the list view and the grid view is how much information a user needs in order to choose between items. Generally, details in Lists, and pictures in Grids. Considered the power supply can be very difficult to allow users to verify mainly through pictures, especially for users without rich knowledge of power supplies or cables. So the list view was decided to be implemented to display the search results.

Considering the search result page can be very long, it is more efficient to provide the functionality which allows users to back to the top of the screen by displaying a floating button at the right bottom of the screen (the back-to-top button). As the floating button may hide the details of the information, it will disappear when users scroll down the screen, and appear when users scroll up the screen.

Product Description User Interface

Product description user interface was designed for displaying related information about the specific product including its technical specifications. Figure 4.13 presents the paper prototyping for *search* user interface 1 and figure 4.14 presents the paper prototyping for *product description* user interface 2.

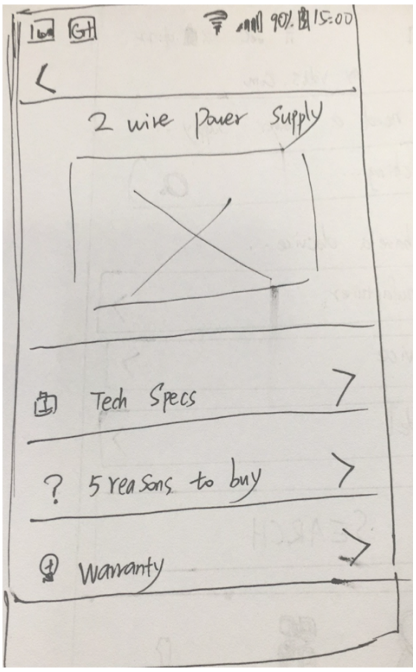

Figure 4.13: Paper Prototype for Prod Desc UI 1.

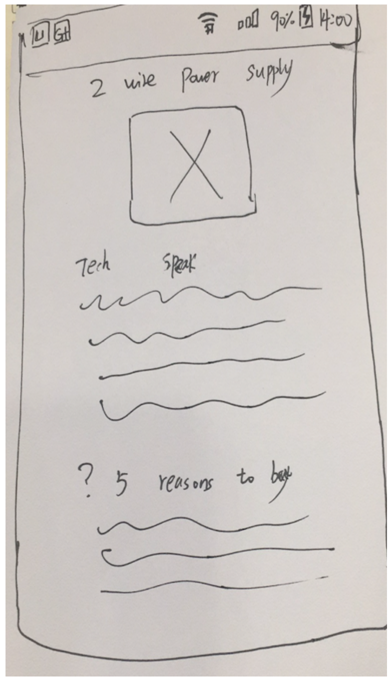

Figure 4.14: Paper Prototype for Prod Desc UI 2

There were two design options for implementing the *product description* user interface. The first design provides the users with a clickable button to load and display the corresponding information. The other design displays information below each label directly as similar as MyVolts web app. As a single product may have multiple technical specifications, it would be very slow to retrieve all data with technical specifications at once and display on the user interface. Therefore, the design one was selected.

4.3.2.2 Prototype 2 Implementation

1) List View

As images can be powerful to take users' attention and provide useful messages for users to verify their expected products, each item in the list consists of an image with text-based information rather than heavy text only. Considered images normally have a big size[12], the images in the list view were designed to be loaded asynchronously in the background using AsyncTask API[13].

In order to improve the efficiency of the system, each time the system only retrieves 10 items by using the SELECT LIMIT query and it will automatically load more items when the user scrolls to the last item. Considered users have different scrolling speed, the coming new items will be slowed down to load in the screen in order to allow users to differentiate the new items and the early items.

Considered to visualize the product description screen, the unique *product_id* will be cached. When the user is clicking the specific item in the results list, the *result list* activity will send a HTTP request to the remote SPARQL sever with the *product_id* in order to get the corresponding product description. Figure 4.15 presents the screenshot of the *results list* user interface.

2) Search Box

As the refine search pattern was designed to be implemented in the MyVolts mobile applications, a search icon was placed on the action bar followed the android design pattern guidelines. Figure 4.16 presents the screenshot of refine search screen which allows the user to refine the search.

3) No Record Hint

When there are no available results at the begin of a refine search screen, it is necessary to give a feedback to users by placing an appropriate hint on the screen as shown in figure 4.18.

4) Network Unreachability Hint

When users are doing online searches, it is very important to check the network status firstly. If the network is unreachable, the screen should give an appropriate hint to ask users to check their mobile settings or move to a place with network coverage. An appropriate hint was designed to be placed on the search screen when there is no available network. Figure 4.17 presents the screenshot of network unreachability hint.

---

[12] https://developer.android.com/training/improving-layouts/smooth-scrolling.html
[13] https://developer.android.com/reference/android/os/AsyncTask.html

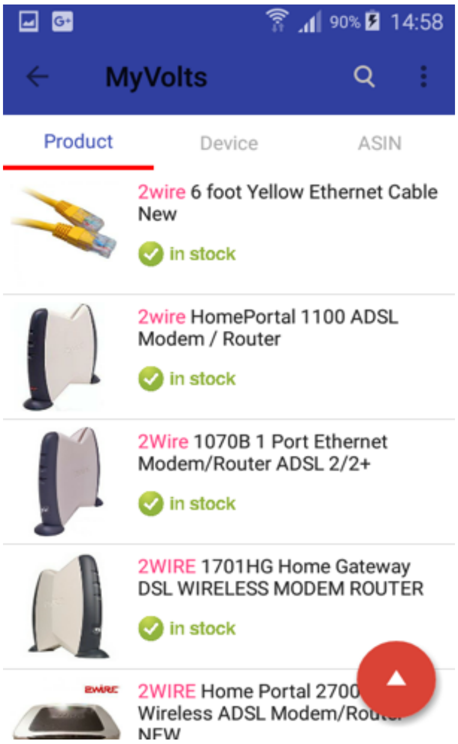

Figure 4.15: Result List UI

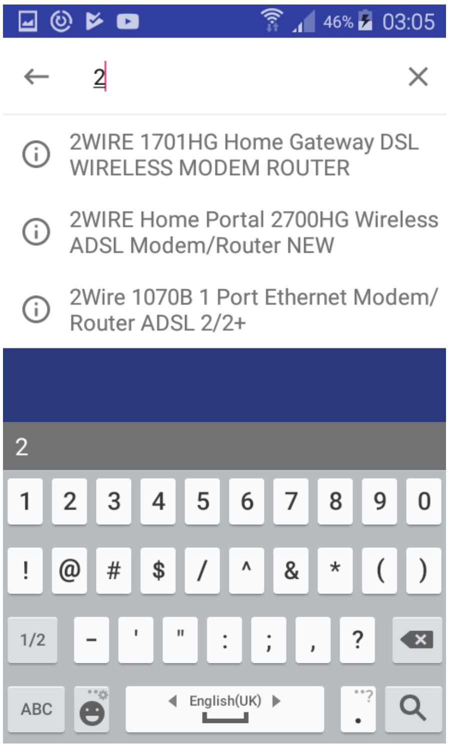

Figure 4.16: Refine Search

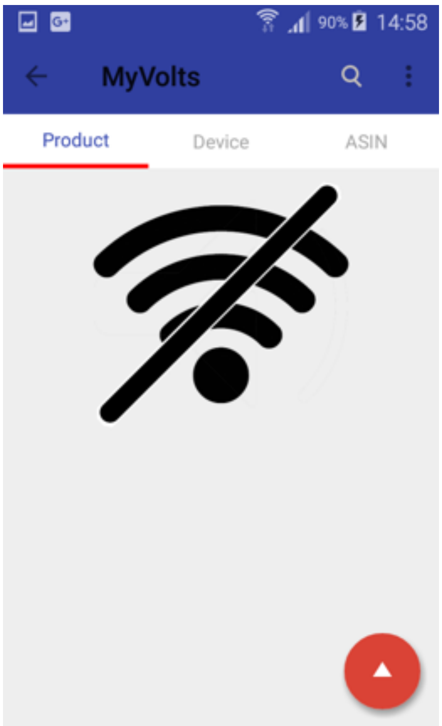

Figure 4.17: Network Unreachability Hint

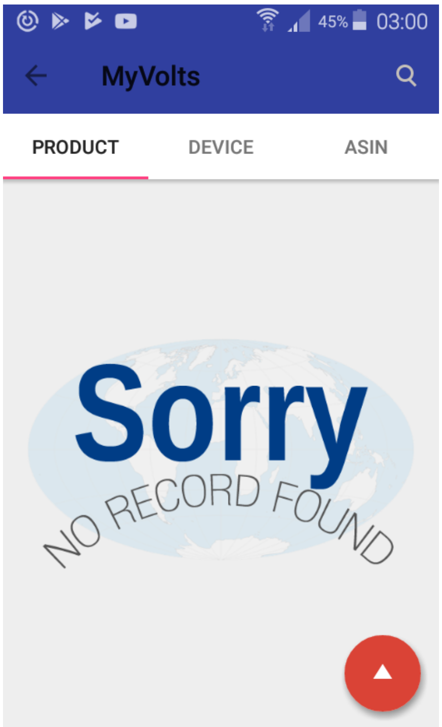

Figure 4.18: No Record Hint

Product Description User Interface

Figure 4.19 presents the screenshot of the *product description* user interface. The product description user interface contains different labels to allow users to click and get the corresponding information. When the user clicks the label, the information will be visualized through sending a HTTP request to the remote SPARQL server. Figure 4.20 presents the example when the user clicks the *Tech Specs* label.

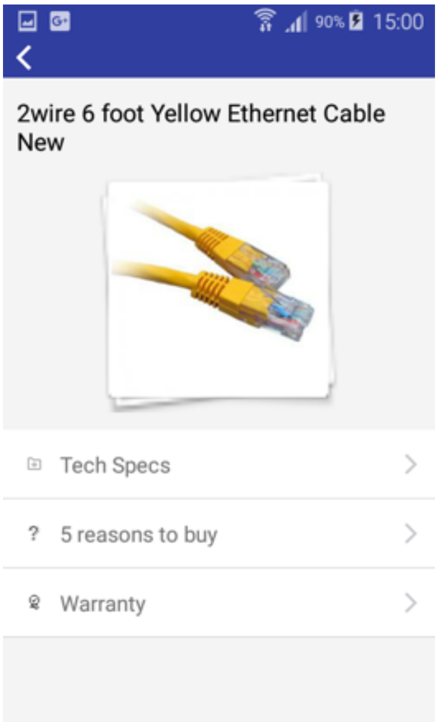

Figure 4.19: Product Description UI

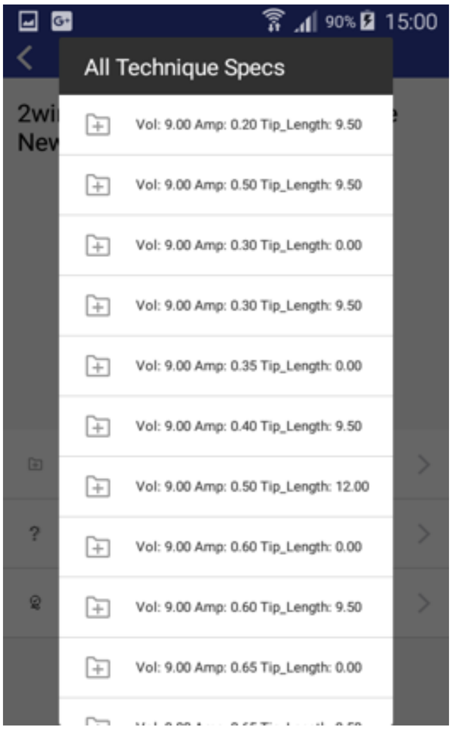

Figure 4.20: Technical Specifications List

4.4 Chapter Summary

This chapter presented the design and implementation of MyVolts mobile application, which will be used for the further evaluation. The evaluation will be carried out with iterative experiments. The design of prototype 1 will be used to gather early feedback (see Section 5.2), and the prototype 2 will be used to measure the usability of the full version MyVolts mobile application (see Section 5.3).

Chapter 5

Evaluation

This chapter aims to evaluate the usability of the MyVolts app. The evaluation has been carried out into four experiments including three lab-based experiments and one web-based online questionnaire.

5.1 Introduction

Based on the different usability purpose, the MyVolts app evaluation approach is divided into four stages. (Stage 1) Considering the requirements could change rapidly, an initial prototype was evaluated by gathering the early feedback in order to achieve time effective. (Stage 2) The second experiment aimed to observe and measure user behaviors after the full development. (Stage 3) The purpose of the third experiment was to compare the difference between MyVolts mobile app and web app in order to find out whether the mobile app could achieve the equivalent usability as web app. (Stage 4) The last experiment was done by analyzing users' subjective impression and preferences on MyVolts mobile app features based on the answers and feedbacks from a web-based online questionnaire. Various types of participants including the employees from MyVolts Ltd, people who either have CS background or not, and potential users were recruited to involve the app testing.

The structure of three experiments is broken down into three sections

(Section 5.2, 5.3, 5.4 & 5.5), which presents the experimental method, data collection, and analysis results for each individual experiment. Finally, overall conclusions are discussed.

5.2 Experiment 1 (Evaluation Stage 1)

This initial experiment aims to investigate the user experience of the initial prototype through getting the early feedbacks from the users.

5.2.1 Hypothesis

H1: The search user interface design for user input design is user-friendly. User-friendly here is defined as the participants expressed that the search user interface design for user input is clarity and not confused.

5.2.2 Experimental Method

In this experiment, participants were not required to perform the tasks. Instead of allowing them to interact with the app, they were given a demonstration of the initial prototype with the explanation of the design ideas. When the demonstration finished, they were given the SUS questionnaire and it though collective decision-making so that they agreed a score for each question and feedbacks together.

5.2.3 Participant Recruitment

In this experiment, 5 staff members from MyVolts including a database engineer, managing director, operations manager were invited to take part in the usability testing. People from MyVolts Ltd has the richest previous experiences with the product, so it is very useful to get feedbacks in order to make sure the functionalities of the app are correct. In addition, potential suggestions could be useful to improve the usability for the future design and implementation as they have rich experience on designing and implementing the web application and dealing with customer queries.

5.2.4 Data

Table 5.1 illustrates the SUS score for the initial experiment.

Table 5.1. SUS Questionnaire and Scores

Question		SUS Score	
1	If I need a power supply, I think I would like to use this system frequently.	Agree	3
2	I found the system unnecessarily complex.	Disagree	3
3	I thought the system was easy to use.	Agree	3
4	I think that I would need the support of a technical person to be able to use this system.	Disagree	3
5	I found the various functions in this system were well integrated.	Disagree	3
6	I thought there was too much inconsistency in this system.	Neutral	2
7	I would imagine that most people would learn this system very quickly.	Disagree	3
8	I found the system very cumbersome to use.	Agree	3
9	I felt very confident using this system.	Disagree	3
10	I needed to learn a lot of things before I could get going with this system.	Agree	3
Total			29

(**From the users' feedback, the first question was refactored as almost of them don't need power supply*)

5.2.5 Analysis

1) Overall Satisfaction

The overall satisfaction of 72.5% was calculated based on the standard calculation method as discussed in section 2.4.2. Figure 5.1 shows the mean SUS scores for each adjective rating based on the study, the rating is then matched to "good" which is in the third quartile [57].

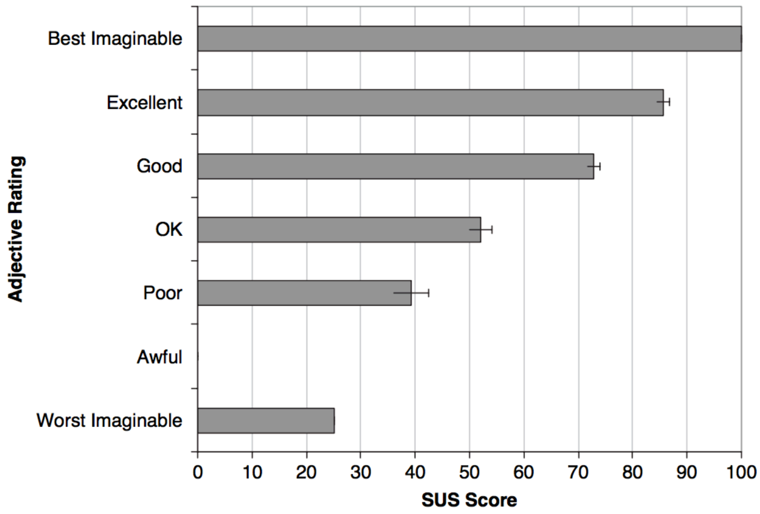

Figure 5.1: Mean System Usability Scale (SUS) Scores for Each Adjective Rating.

2) User Feedback

Figure 5.2 presents user feedback. The main feedbacks are concentrated on the overall usability, hot products feature, and history list feature. Participants all thought the app has the right complexity and they also agreed that the hot products and history list features are useful to improve the app usability.

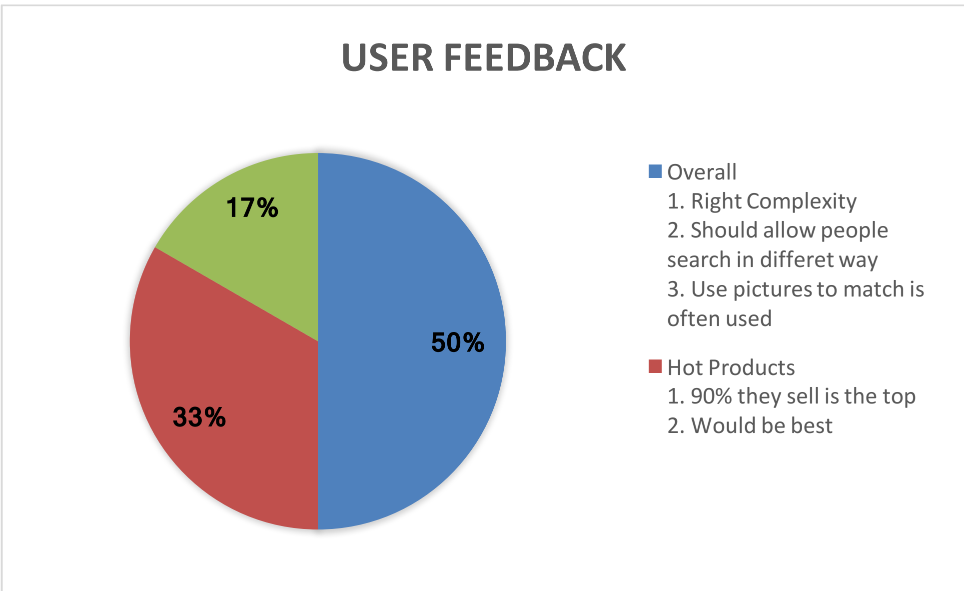

Figure 5.2. User Feedback for Experiment 1.

5.2.6 Conclusion

Through calculating the SUS score, it is clearly that the overall usability of the initial prototype achieved good usability. Based on the feedbacks, participants also indicated that the app has the right complexity, which means the design of the initial prototype was clarity. Therefore, it was shown that the search user interface design for user input is user-friendly. In addition, it was found that the following factors could significantly improve the usability of the search functionality for the mobile apps.

1. The search history list is a factor to improve usability of search functionality.
2. The hot recommendation list is a factor to improve usability of search functionality.
3. Users prefer to use images to verify products.
4. Users like to use different ways to search.

5.3 Experiment 2 (Stage 2)

This experiment aims to determine whether the second prototype achieved the improvement from initial prototype by measuring user behavior and performance.

5.3.1 Hypothesis

H2: The MyVolts app achieved a higher usability and a higher SUS score than initial prototype.

As discussed in section 2.4.1, the evaluation of the MyVolts app followed ISO 9241-11 usability standard. Thus, the following three sub hypothesis were posited.

H2a: The MyVolts app is effective.

H2b: The MyVolts app is efficient.

H2c: Overall, users are satisfied with the MyVolts app.

5.3.2 Experimental Method

This experiments conducted a lab-based experiment. Nielsen Norman Groups[14] recommends the tasks scenarios should be realistic and actionable. The tasks are representative to the main purpose of the app, so these scenarios focused on the app search functionalities.

Table 5.2 lists the user goals for the task scenarios.

[14] https://www.nngroup.com/articles/task-scenarios-usability-testing/

TABLE 5.2: User Goals of Task Scenarios

Task	User Goals
1	Find the product by given product name using search box. This task is designed for measuring users' behaviors when they know the products are exactly they want. They may have different performances to complete this task such as using autocomplete list or typing keywords.
2	Match the products by given device information using drop down search box. This task aims to investigate users' behaviors how they use select boxes to complete the task.
3	Find the product by given key words using search box. This task is to investigate users' behaviors when they have a basic idea of what kind of product they want.
4	Find the product which was searched. This task is related to task 1, which aims to get feedback of how easy they can find the product in the history list.

To evaluate the full version prototype, four scenarios were designed to the usability of the app based on the list of suggestions [60]. During the investigation. a paper-based task scenarios list was provided to the participants using plain language and the minimum instructions were given before the tasks being performed. Table 5.3 illustrates all of four task scenarios used in this experiment.

TABLE 5.3: Task Scenarios for Experiment 2

Task	Scenarios
1	Use app search box to find the correct MyVolts cable "1 to 4 DC Power split cable 2.1mm".
2	Use app drop down search box to find a power supply which matches the device "Router" made by "2wire" with the model reference "2700HG", then select the first matched product.
3	Use app search box to find a "Yellow Ethernet cable" which made by 2wire.
4	Use search history list to find the power supply/cable which was searched in **task 1**.

To record the user behaviors, an underlying program was designed to track the users' performance including completion time and number of clicks for each widget. The benefits of this design is to make sure the participants were staying in the controlled experimental environment, which helped to improve the accuracy of the test results.

The simplest way to calculate the task completion time is to subtract start time from end time as shown below.

$$Task\ Completion\ Time = End\ Time - Start\ Time$$

After the participants finishing the tasks, each of them was asking to fill the SUS questionnaires and some feedbacks were interviewed

5.3.3 Participant Recruitment

There were three groups of participants with a total of 11 people involved in this experiment This included 4 employees from MyVolts Ltd, 4 people with CS (Computer Science) background, and another 3 people with non-CS background. The participants apart from MyVolts employees are my friends and classmates. Among these participants, only one person from MyVolts Ltd had previous experience on how to interact with the prototype from the initial experiment. However, as the app has evolved, that should have no big impact into the experimental findings. All other participants had no previous interactive experience with the app.

5.3.4 Data

Table 5.4 displays the participant task metrics for experiment 2. It is clear that all participants completed all tasks and 9 out of 11 achieved errors free.

TABLE 5.4: Participant Task Metrics for Experiment 2

Metrics / Participants	Task Completed	Completion Time (sec)				SUS Score (%)	# of Errors
		Task 1	Task 2	Task 3	Task 4		
1	4	67	52	67	10	75	0
2	4	32	39	81	24	95	2
3	4	24	38	26	12	90	0
4	4	41	28	36	8	83	0
5	4	21	28	35	17	93	0
6	4	17	13	15	4	93	0
7	4	18	17	20	3	100	0
8	4	10	14	20	6	85	0
9	4	43	52	36	18	85	1
10	4	48	30	32	16	88	0
11	4	22	32	38	22	100	0

Table 5.5 illustrates the full error opportunities and total number for each task.

TABLE 5.5: Opportunities for Errors using MyVolts Mobile Application

Task ID	Types of Possible Errors	Total Errors per Task
1	Mistyping a product name.	6
1	Accidentally clicking an item from hot recommendation.	
1	Accidentally clicking an item from history record list.	
1	Select wrong item from autocomplete list.	
1	Accidentally clicking search button with no input.	
1	Using drop down search box instead.	
2	Accidentally clicking search box.	4
2	Selecting wrong item.	
2	Missing item when scrolling the screen.	
2	Selecting wrong product.	
3	Mistyping product key words.	5
3	Accidentally clicking an item from hot recommendation.	
3	Accidentally clicking an item from history record list.	
3	Accidentally clicking search button with no input.	
3	Selecting wrong product.	
4	Using search box instead.	3
4	Using drop down search box instead.	
4	Accidentally clicking an item from hot recommendation.	

Table 5.6 illustrates the errors which were occurred during the users' performance.

TABLE 5.6: Task Errors Distribution

Error ID	Task ID	Errors Descriptions
1	**2**	Accidentally clicking search box.
2	**2**	Missing item when scrolling the screen.
3	**3**	Accidentally clicking an item from hot recommendation.

5.3.5 Analysis

1) Effectiveness

Table 5.7 illustrates the binary success record for each participant. It is obvious that every participant completed all tasks during the experiment, and all tasks has achieved 92.31% completion rate with 95% confidence interval.

TABLE 5.7: Binary Success Record

Participant	Task 1	Task 2	Task 3	Task 4
P1	1	1	1	1
P2	1	1	1	1
P3	1	1	1	1
P4	1	1	1	1
P5	1	1	1	1
P6	1	1	1	1
P7	1	1	1	1
P8	1	1	1	1
P9	1	1	1	1
P10	1	1	1	1
P11	1	1	1	1
Average	100%	100%	100%	100%
Confidence Interval (95%)	86.7%	86.7%	86.7%	86.7%

Figure 5.3 illustrates the successful completion rate for each task. According to a study from Sauro, the average task completion rate is 78%[15]. Thus it can be shown that MyVolts app has high effectiveness as it achieved high completion rate.

[15] http://www.measuringu.com/blog/task-completion.php

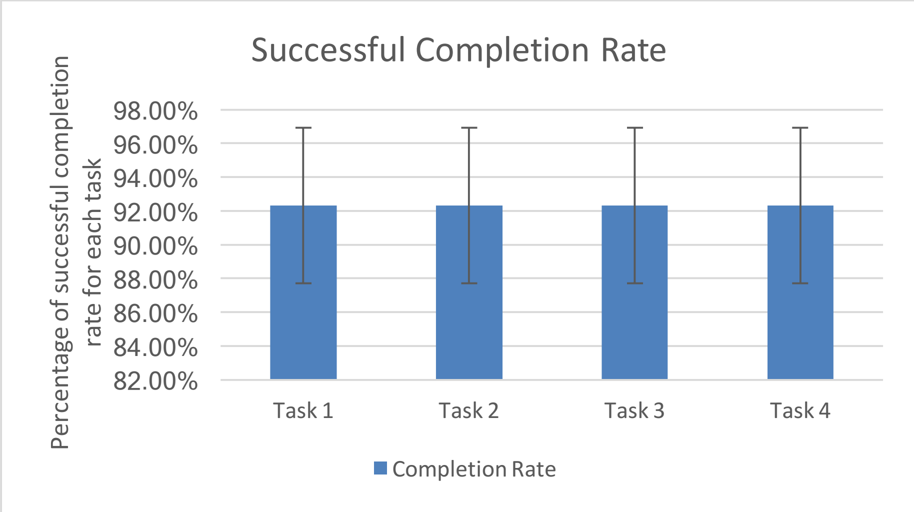

Figure 5.3: Successful Completion Rate for Each Task.

2) Efficiency

Table 5.8 illustrates the mean completion time for each task specified by overall and each group of participants. We can simply find from the data in the table that

Finding 1. People with CS background is able to spend less time on completing the tasks than others.

TABLE 5.8: Mean Completion Time for Each Task

	Task 1	Task 2	Task 3	Task 4
MyVolts	38.38	38.32	47.47	12.32
CS	15.92	17.16	21.40	5.91
Non-CS	35.68	36.82	35.24	18.50
All	29.99	30.77	34.70	12.24

In order to calculate the overall efficiency, one simple approach can be used is to calculate the percentage of the total time taken by all users who successfully completed the tasks in relation to the total time taken by all users[16]. After the calculation, MyVolts app has achieved 100% overall relative efficiency.

$$Overall\ Relative\ Effeciency = \frac{\sum_{j=1}^{R}\sum_{i=1}^{N} n_{ij}\, t_{ij}}{\sum_{j=1}^{R}\sum_{i=1}^{N}\ t_{ij}} \times 100\%$$

3) Satisfaction

[16] http://usabilitygeek.com/usability-metrics-a-guide-to-quantify-system-usability/

Figure 5.4 and 5.5 shows the SUS score for each participant and the overall frequency distribution. As shown on the graph, more than half participants agreed that the app has an excellent usability. After calculation, the overall satisfaction with 95% confidence level is 85.3%, which can be mapped to the "excellent" usability [57].

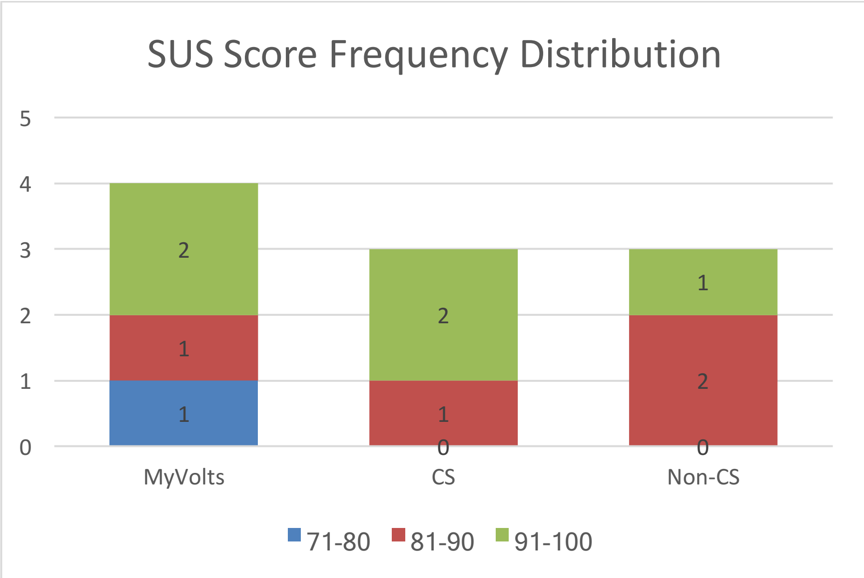

Figure 5.4: SUS Score Frequency Distribution.

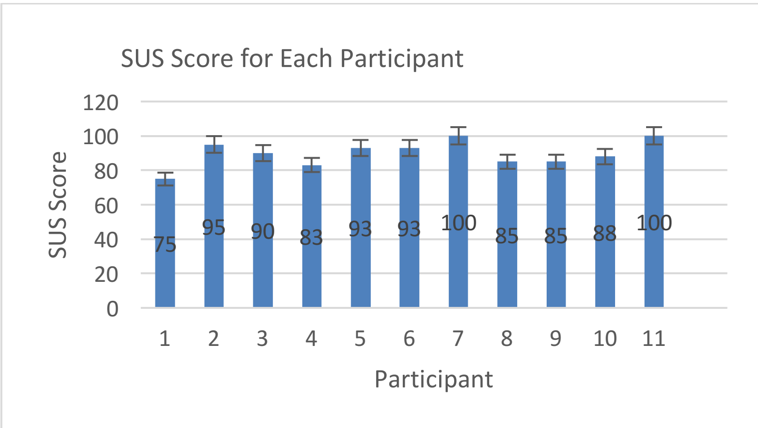

Figure 5.5: SUS Score for Each Participant.

4) User Error Occurrence Rate

There are different methods or techniques to calculate it, in this project the THERP (Technique for human error rate prediction) was picked as the standard as it provides cleared and detailed instructions [58]. The value of error occurrence rate for each task as shown in table 5.9.

TABLE 5.9: Error Occurrence Rate

	Task 1	Task 2	Task 3	Task 4
# of Errors Occurred	0	2	1	0
# of Errors Opportunities	6 * 11 = 66	4 * 11 = 44	5 * 11 = 55	3 * 11 = 33
# Occurrence Rate	0	5%	2%	0

Based on the errors occurred in the tasks, we can find that

Finding 2. According to error 2, even the name of items is displayed in alphabetic order, users still have possibility to miss their target one.

Finding 3. According to error 3, users don't prefer to type many words subconsciously, when they performing a task, they would like to find the simplest way with least steps to achieve the goal.

5) Number of Steps to Achieve the Goal

TABLE 5.10: Mean number of click

Task ID	Number of Clicks
1	3
2	5
3	3
4	2

The number of steps to achieve the goal is useful to indicate how simple and quick to complete the tasks. Table 5.10 illustrates the average number of clicks (steps) for achieving each task. As the task 1 and 3 are both required to use the search box while the task 2 is required to use the drop down box, and they are all used to find a specific product, it is obvious that using search box to complete the tasks required less steps than using drop down box. Therefore, considered with the consequence and error 1, we can conclude that

Finding 4. Using search box is more efficient and simpler.

6) Overall Usability Measurement

It is very useful to combine all related usability metrics into a single one to measure the overall system usability. As SUM (Single Usability Metric) has a general agreement from the standard ISO 9241[59], which could be worked well for the evaluation. The three main dimensions of usability in this project are effectiveness, efficiency and satisfaction, thus, the usability metrics involved in the calculation of the SUM should be the completion rate, completion time, errors rate, and satisfactions. The SUM was calculated using the SUM calculator developed by Sauro[17] as shown in table 5.11.

Table 5.11: Work Sheet for SUM Calculation

	SUM								
Task	**Low**		**High**	**Completion**	**Satisfaction**	**Time**	**Errors**	**Graph Low**	**Graph High**
Task 1	75.2%	93.2%	98.0%	86.7%	85.3%	77.5%	97.1%	18.0%	4.9%
Task 2	72.6%	92.0%	97.7%	86.7%	85.3%	79.5%	88.8%	19.4%	5.8%
Task 3	77.0%	94.5%	98.6%	86.7%	85.3%	84.7%	94.2%	17.5%	4.2%
Task 4	75.4%	94.3%	98.5%	86.7%	85.3%	82.0%	94.6%	18.9%	4.2%
Overall SUM	75.1%	93.6%	98.2%						

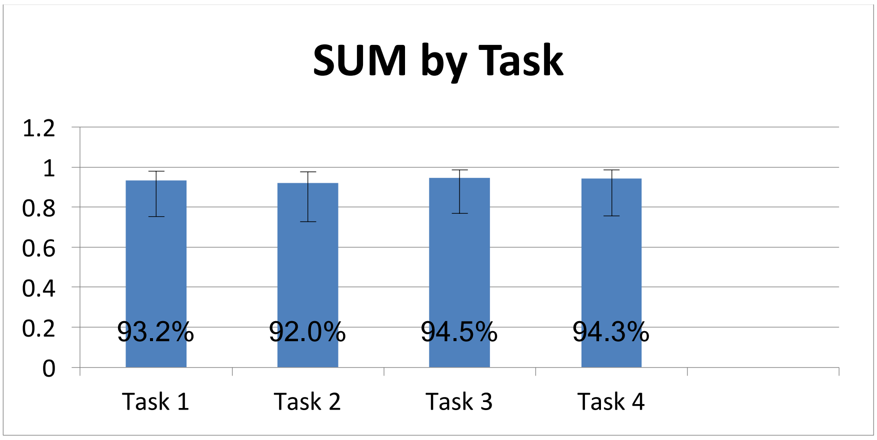

Figure 5.6: Figures of SUM by Task.

According to a study, the average SUM score is 65%. The study also assigns each SUM scores block to a percentile rank as shown in figure 5.7. The overall SUM score of the MyVolts app is 93.6%, which is in the 99th percentile, therefore, we can conclude that MyVolts app has achieved a high usability.

[17] https://measuringu.com/sum-2/

SUM %	Percentile Rank
55	25%
68	50%
78	75%
82	85%
84	90%
87	95%
91	99%

Figure 5.7: Percentile Rank for SUM Score.

7) Overall Usability Comparison by Groups

As there are three groups of participants involved in this experiment, there is significant to analyze the difference of performances between them. To compare them, each of usability metrics including task completion rate, task completion time, and satisfaction were converted into the percentage [13]. Then the mean scores can be used to measure each group's performance and satisfaction. In order to make sure there is a statistically significance between three samples, multiple *t*-tests [62] was used to calculate. To avoid a Type 1 error [62], the alpha was adjusted ($\alpha = 0.167$). Table 5.12 presents the overall usability score for each participant.

Table 5.12: Overall usability score per participant for MyVolts Web App

Participant #	Time per Task (sec)	Tasks Completed (of 3)	Time %	Tasks %	SUS %	Mean %
1	34	3	34%	100%	85%	83%
2	23	3	96%	100%	93%	96%
3	22	3	100%	100%	73%	91%
4	39	3	56%	100%	80%	79%
5	31	3	71%	100%	88%	86%
6	30	3	73%	100%	90%	88%

Table 5.13 illustrates the results of running multiple t-test for three samples, the results have shown that more than one test result were less that alpha, so it can be found that there is a significant difference between these three samples.

Table 5.13: Multiple t-Test for three samples

Samples	**P (T<=t) one-tail**	**Significant Difference**
MyVolts vs CS	0.00780379359805537	Yes
MyVolts vs non-CS	0.318763414619333	No
CS vs non-CS	0.0393806489983411	Yes

The overall usability comparison between three groups of participants as shown in figure 5.8. Through the graph, we can easily find that

Finding 5: Generally, people who have technical background feel more confident to use the mobile app.

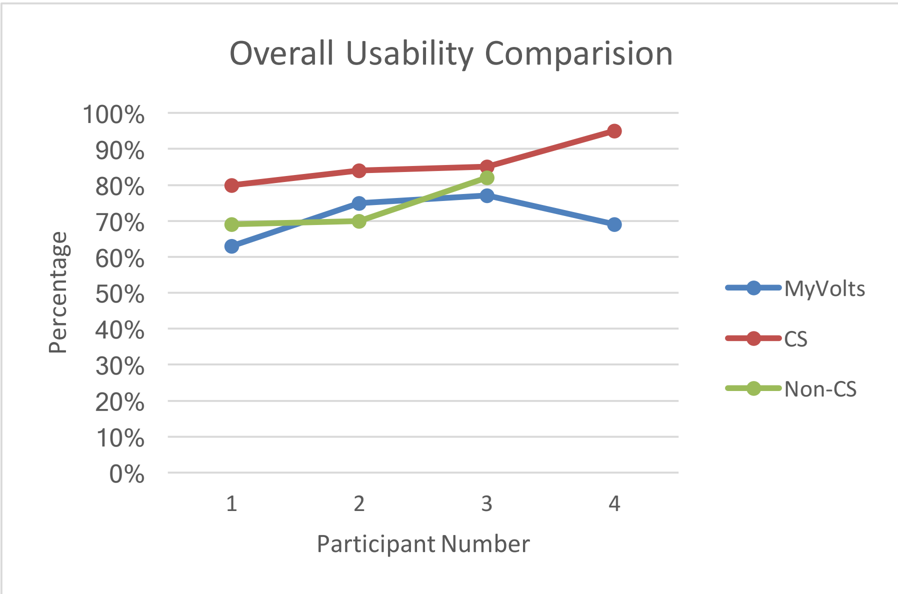

Figure 5.8: Overall Usability Comparison between Three Groups.

8) Comparison between Prototype 1 and Prototype 2

The comparison of overall satisfaction between the first two experiments as shown in table 5.14. Based on the data, it is very clear that through the design and development, MyVolts app has achieved a significant improvement with 12.8%.

TABLE 5.14: Comparison of Overall Satisfaction between Two Experiments

Experiment ID	Overall Satisfaction
1	72.5%
2	85.3%

9) User Feedback

Based on the users' feedback, participants are all agreed with the high usability of the system. Figure 5.9 presents their comments on different features. Some of them think that an attractive interface or innovative technologies are very helpful to gain more returning users. Almost of them agreed that a short tutorial is necessary to allow first time users to learn how to use the system quickly, but depends on the different skill levels for the users, it should allow users to close the instruction at any time they want. Through the users' feedbacks, the following findings were found.

Finding 6: People expect that they can refine search anywhere at any time.

Finding 7: People like innovations of technologies.

Finding 8: People prefer to close First time users' tutorials by themselves instead of only display for one time.

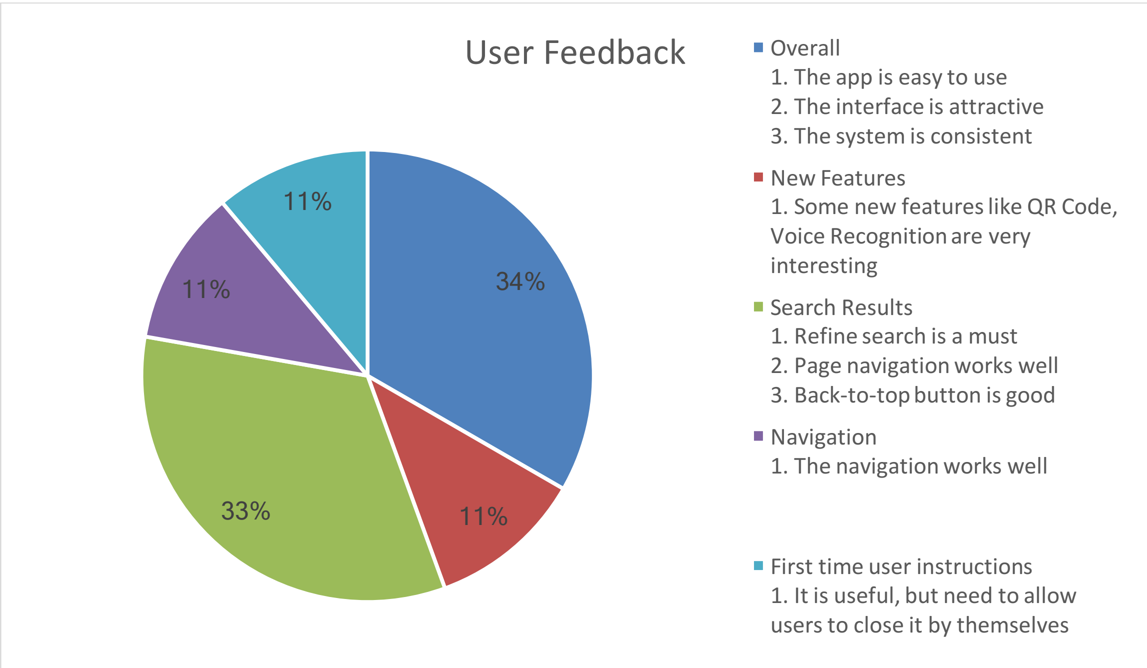

Figure 5.9: User Feedback for Experiment 2

5.3.6 Conclusion

Through analyzing different usability metrics, it is obviously that all results after calculation are higher than the average results. Through comparison, prototype 2 also achieved a higher SUS score than initial prototype. Therefore, it can be concluded that prototype 2 has achieved a higher usability than initial prototype.

5.4 Experiment 3 (Stage 3)

This experiment aims to investigate the difference of usability between MyVolts web-based app and mobile app.

5.4.1 Hypothesis

H3: It is possible for MyVolts mobile app to achieve equivalent usability as MyVolts web

5.4.2 Experimental Method

In this experiments, two sample *t*-test method was applied to analyze the difference between two means in order to indicate whether there was a significant difference between two samples. Therefore, the null hypothesis ($\mathrm{H_o}$) and alternative hypothesis ($\mathrm{H_a}$) were posited[18].

$$\mathrm{H_o}: \mu_1 = \mu_2$$

$$\mathrm{H_a}: \mu_1 \neq \mu_2$$

As there are multiple usability metrics, they were combined together to generate the percentage of the overall usability for each participant. Then these scores were used to calculate the p value to identify whether there was a statistical significant difference or not. If there was a significant difference, different usability metrics would be used to determine whether there is a possibility for MyVolts mobile app to achieve the equivalent usability as MyVolts web app. Otherwise, it would be failed to reject the null hypothesis, which means it may need more evidence to make the samples statistically significant.

The evaluation conducted a lab-based experiment. Each participant was given the first three tasks from the experiment 2 as the MyVolts web app doesn't implement the search history functionality. Each participant then was required to fill the SUS questionnaire.

5.4.3 Participant Recruitment

There were totally 6 people invited to take part in this experiment. The participants my friends and classmates. In order to make sure there is no bias,

[18] μ_1 *is the mean usability score of MyVolts mobile applcication*, μ_2 *is the mean usability score of MyVolts web application.*

they are all fresh people with no previous experience on interacting with MyVolts mobile app. All participants have rich experience with new generation technologies.

5.4.4 Data

Table 5.15 displays the raw data collected during the experiment including completion time per task, task completion status, and SUS score for each participant.

TABLE 5.15: Participant Task Metrics for Experiment 3

Participant #	Task Competition Time (sec)			Task Completed			SUS Score
	Task 1	Task 2	Task 3	Task 1	Task 2	Task 3	
1	19	20	64	1	1	1	85%
2	27	25	23	1	1	1	92.5%
3	42	25	22	1	1	1	72.5%
4	24	27	39	1	1	1	80%
5	69	32	21	1	1	1	87.5%
6	37	23	30	1	1	1	90%

Table 5.16 illustrates the potential error opportunities using MyVolts web app.

TABLE 5.16: Opportunities for Errors using MyVolts Web App

Task ID	Types of Possible Errors	Total Errors per Task
1	Mistyping a product name.	5
1	Using top search bar instead.	
1	Select wrong item from autocomplete list.	
1	Accidentally clicking search button with no input.	
1	Using drop down search box instead.	
2	Accidentally clicking search box.	4
2	Selecting wrong item.	
2	Missing item when scrolling the screen.	
2	Selecting wrong product.	
3	Mistyping product key words.	4
3	Using top search bar instead.	
3	Accidentally clicking search button with no input.	
3	Selecting wrong product.	

Table 5.17 displays the errors made by participants during the experiment.

TABLE 5.17: Task Errors Distribution

Error ID	Task ID	Errors Descriptions
1	3	Using top search bar instead.
2	1	Using top search bar instead.

5.4.5 Analysis

Table 5.12 illustrates the overall usability score per participant for MyVolts web app and table 5.18 illustrates the overall usability per participant for MyVolts mobile app.

TABLE 5.18: Overall usability score per participant for MyVolts Mobile App

Participant #	Time per Task (sec)	Tasks Completed (of 3)	Time %	Tasks %	Rating %	Mean %
1	67	3	15%	100%	75%	63%
2	32	3	31%	100%	95%	75%
3	24	3	42%	100%	90%	77%
4	41	3	24%	100%	83%	69%
5	21	3	48%	100%	93%	80%
6	17	3	59%	100%	93%	84%
7	18	3	56%	100%	100%	85%
8	10	3	100%	100%	85%	95%
9	43	3	23%	100%	85%	69%
10	48	3	21%	100%	88%	70%
11	22	3	45%	100%	100%	82%

Through calculation, the p value for two-tail is 0.017 which is much less than 0.05 as shown in table 5.19. Therefore, we can reject the null hypothesis and prove that there is a significant difference between two samples. The following usability metrics are used to investigate the difference of usability between two apps.

TABLE 5.19: t-Test: Two-Sample Assuming Unequal Variances

	Mobile	*Web*
Mean	0.771818182	0.871666667
Variance	0.008476364	0.003576667
Observations	11	6
Hypothesized Mean Difference	0	
df	14	
t Stat	-2.700887721	
P(T<=t) one-tail	0.008613338	
t Critical one-tail	1.761310136	
P(T<=t) two-tail	0.017226677	
t Critical two-tail	2.144786688	

*(*df refers to degrees of freedom)*

1) Effectiveness

Table 5.20 illustrates the binary success record for experiment 2. It is obvious that every participant completed all three tasks so that we can conclude MyVolts mobile app has achieved the equivalent effectiveness as MyVolts web app.

TABLE 5.20: Binary Success Record

Participant	Task 1	Task 2	Task 3	Task 4
P1	1	1	1	1
P2	1	1	1	1
P3	1	1	1	1
P4	1	1	1	1
P5	1	1	1	1
P6	1	1	1	1
Mean	100%	100%	100%	100%
Confidence Interval (95%)	80%	80%	80%	80%

2) Efficiency

The difference of efficiency can be measure by comparing the mean time spent on completing tasks as shown in table 5.21. It shows that users spent less time on completing tasks using MyVolts web app than mobile app, but the different of 2.28 sec is not very big.

TABLE 5.21: Mean Task Completion Time for Two apps

Tasks	Mobile App (sec)	Web App (sec)
Task 1	29.99	33.12
Task 2	30.77	25.07
Task 3	34.70	30.44
Mean	31.82	29.54

3) Satisfaction

Table 5.22 illustrates the SUS score distribution and mean SUS score with 95% confidence interval for MyVolts mobile app and web app.

TABLE 5.22: SUS Scores Distribution

SUS Scores (95% Confidence)		
	Mobile	Web
	75%	85%
	95%	92.5%
	90%	72.5%
	82.5%	80%
	92.5%	87.5%
	92.5%	90%
	100%	
	85%	
	85%	
	87.5%	
	100%	
Mean	85.3%	84.6%

Figure 5.10 displays the frequency distribution of SUS scores. It clearly shows that the distribution of SUS score for both mobile app and web app is between 70 and 100.

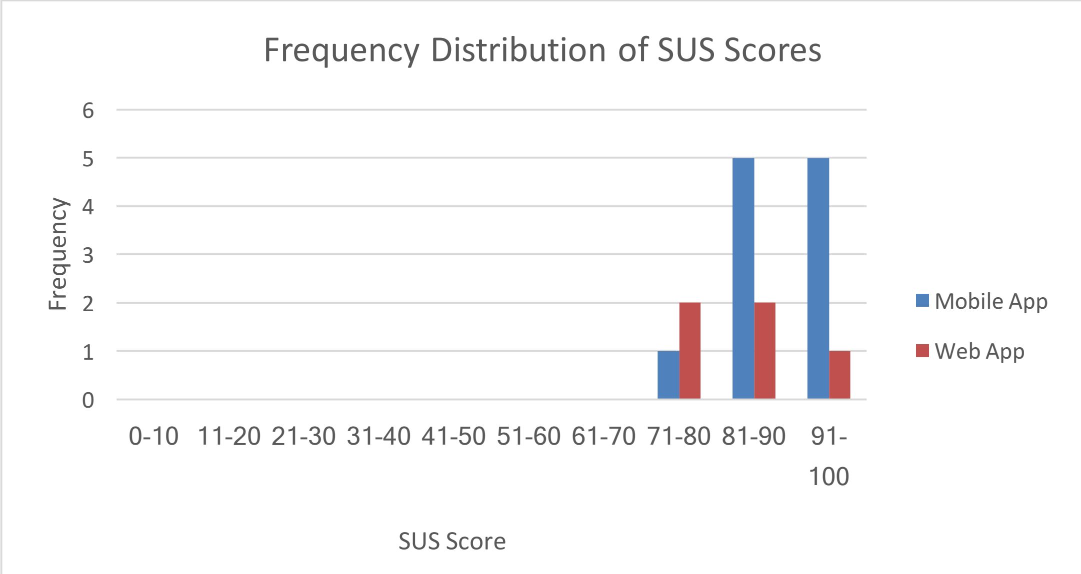

Figure 5.10: Frequency Distribution of SUS Scores

Through the data discussed above, it is easy to find that the mean SUS score of 85.3% from MyVolts mobile app is few higher than the mean SUS score of 84.6% from MyVolts web app. It can be concluded that MyVolts mobile app has achieved the higher satisfaction.

4) User Error

Based on the error distribution as shown in table 5.23, it is obvious that two participants encountered the same issue which they both input the words in the same and wrong search box. So we can found that

Finding 9: Appropriate number of search boxes in the same page should be considered to make users not confused.

TABLE 5.23: Error Free Rate Comparison

Errors Free Rate	Mobile App %	Web App %
Task 1	100	97
Task 2	95	100
Task 3	98	96
Mean	97.67	97.67

5) Overall Usability Measurement

Table 5.24 and 5.25 illustrate the overall usability score of MyVolts mobile app and web app by calculating their SUM scores. Through comparing the difference of SUM scores between two apps, it clearly shows that the two SUM scores are very close with only 2.1% difference.

TABLE 5.24: Overall Usability of MyVolts Mobile App

	SUM								
Task	**Low**		**High**	**Completion**	**Satisfaction**	**Time**	**Errors**	**Graph Low**	**Graph High**
Task 1	75.2%	93.2%	98.0%	86.7%	85.3%	77.5%	97.1%	18.0%	4.9%
Task 2	72.6%	92.0%	97.7%	86.7%	85.3%	79.5%	88.8%	19.4%	5.8%
Task 3	77.0%	94.5%	98.6%	86.7%	85.3%	84.7%	94.2%	17.5%	4.2%
Overall SUM	74.9%	93.2%	98.1%						

TABLE 5.25: Overall Usability of MyVolts Web App

	SUM								
Task	**Low**		**High**	**Completion**	**Satisfaction**	**Time**	**Errors**	**Graph Low**	**Graph High**
Task 1	58.9%	90.7%	98.6%	80.0%	65.3%	81.1%	88.2%	31.8%	7.9%
Task 2	59.0%	91.4%	98.7%	80.0%	65.3%	80.5%	92.9%	32.4%	7.3%
Task 3	59.1%	91.0%	98.7%	80.0%	65.3%	83.8%	85.9%	31.9%	7.7%
Overall SUM	59.0%	91.1%	98.7%						

5.4.6 Conclusion

Through analyzing the difference of usability between MyVolts mobile app and web app by measuring different usability metrics including effectiveness, efficiency, satisfaction, user error, and overall usability. As shown in table 5.26, MyVolts mobile app and web app has achieved the equivalent effectiveness and user error free rate, and the difference of efficiency and satisfaction between the two apps are very close. Therefore, we can conclude that it is possible for MyVolts mobile app to achieve the equivalent usability as MyVolts web app.

TABLE 5.26: Usability Comparison Summary

Usability Metrics	**Mobile App**	**Web App**
Effectiveness (%)	100	100
Effectiveness (95% Confidence Interval)	86.7	80
Efficiency (sec)	31.82	29.54
Satisfaction (%)	85.3	84.6
User Error Free Rate (%)	97.67	97.67
SUM (%)	93.2	91.1

5.5 Experiment 4 (Stage 4)

This experiment aims to analyze the feedbacks from MyVolts potential customers.

5.5.1 Hypothesis

H4: Users agreed that MyVolts mobile app is able to achieve high usability.

5.5.2 Experimental Method

In this experiment, the potential users' feedbacks were analyzed by asking them to fill a web-based questionnaire[19]. The design of the questionnaire took into account users' subjective impression and preference by providing them screenshots and app design ideas. It aims to gather their overall impression regarding to the app usability and preference of different search features. In addition, it is also important to analyze the connection between users' preference and their background. Therefore, few background questions were also designed such as their age, gender, and frequency of product purchase.

5.5.3 Participant Recruitment

In this experiment, MyVolts contacted 204 recent customers by sending them an email and got 14 responses. As the customers all had rich experience on purchasing power supplies online, their feedbacks are significant to help to measure whether the app usability are successful or not.

5.5.4 Data

Table 5.27 and 5.28 illustrate the raw data of user response for each question. Table 5.29 lists the main comments given by participants.

[19] https://docs.google.com/forms/d/e/1FAIpQLSdR7rsNyF-I8wg46GOJPhqww1LW-CIp1p-D3Bnv1aHOk4KY9w/viewform

TABLE 5.27: Responses for Participants 1-7

Participant / Questions	1	2	3	4	5	6	7
1. Do you think you would use this app?	Agree	Strongly Agree	Neutral	Agree	Agree	Agree	Neutral
2. Which search feature would you prefer to use to find a product?	Search Box	Search by Manufacture, Device Name, and Model Number	Search by Manufacture, Device Name, and Model Number	Search by Manufacture, Device Name, and Model Number	Search by Manufacture, Device Name, and Model Number	Search Box	Search by Manufacture, Device Name, and Model Number
3. Do you think it is important to have more than one way to search in the app?	Very Important	Very Important	Moderately Important	Important	Very Important	Important	Moderately Important
4. What is your gender?	Male	Male	Male	Male	Male	Male	Female
5. What is your age?	45 years or older	45 years or older	45 years or older	35-44 years old	35-44 years old	45 years or older	45 years or older
6. How often in the last year have you bought a replacement power supply/cable?	Once	2-3 times	2-3 times	4-10 times	Once	2-3 times	Once
7. When buying a replacement part did you know which product to buy or did you have to look it up?	Sometimes	Yes	Yes	Sometimes	Yes	Sometimes	Yes
8. If you answered yes or sometimes to the last question, where did you search for information on replacement power supplies?	Google	Google	Google	Google	Google	Google	Google

TABLE 5.28: Responses for Participants 8-14

Questions \ Participant	8	9	10	11	12	13	14
1. Do you think you would use this app?	Strongly Agree	Agree	Strongly Agree	Strongly Agree	Agree	Agree	Strongly Agree
2. Which search feature would you prefer to use to find a product?	Search Box	Search by Manufacture, Device Name, and Model Number	Search Box	Search Box	Search by Manufacture, Device Name, and Model Number	Search Box	Search Box
3. Do you think it is important to have more than one way to search in the app?	Important	Very Important	Very Important	Very Important	Important	Important	Very Important
4. What is your gender?	Male	Male	Female	Male	Male	Male	Female
5. What is your age?	45 years or older	45 years or older	35-44 years old	35-44 years old	25-34 years old	35-44 years old	25-34 years old
6. How often in the last year have you bought a replacement power supply/cable?	Once	4-10 times	Never	Once	Once	2-3 times	Once
7. When buying a replacement part did you know which product to buy or did you have to look it up?	Yes	Sometimes	Yes	Yes	Sometimes	Yes	Yes
8. If you answered yes or sometimes to the last question, where did you search for information on replacement power supplies?	Manufacturer or retailer web-sites	Other	Google	Google	Google	Community Forums	Google

TABLE 5.29: User Comments

	Comments
1	All good.
2	It would be worth including close-up photos of the plugs on the ends of cables as this often helps to identify which cable will fit a particular device.
3	I really had no problem usually hard for me to navigate no problems at all.
4	Great idea, although as a Blackberry user I don't personally use apps as such.
5	Is it really likely that someone will enter an ASIN?
6	It looks great, simplicity is the key I find with apps but with the right functionality.
7	It looks great.

5.5.5 Analysis

The questionnaire includes 5 usability questions and another 5 background research questions.

Q1. Do you think you would use this app?

Table 5.29 displays response distribution for overall subjective impression. It indicates that there is no disagreement of the app usability, 85.7% of participants agreed that they were satisfied with the overall impression of the app.

TABLE 5.29: Responses to Overall Subjective Impression

	Number of Responses	Percentage
Strongly Agree	5	35.70%
Agree	7	50%
Neutral	2	14.30%
Disagree	0	0
Strong Disagree	0	0

Q2. Which search feature would you prefer to use to find a product?

Figure 5.11 displays user search feature selection preference distribution. It shows that half of them prefer using search box to find the product and another half prefer to use drop down box. However, there is no one chose the QR code scanner. Through analyzing users background, it is clear that 50% of participants are more than 45 years old and 35.7% of them range from 34 to 44 years old, the rest ranges from 25-34 years old. Therefore, we can find that

Finding 10: Mature people prefers traditional search methods rather than new innovation.

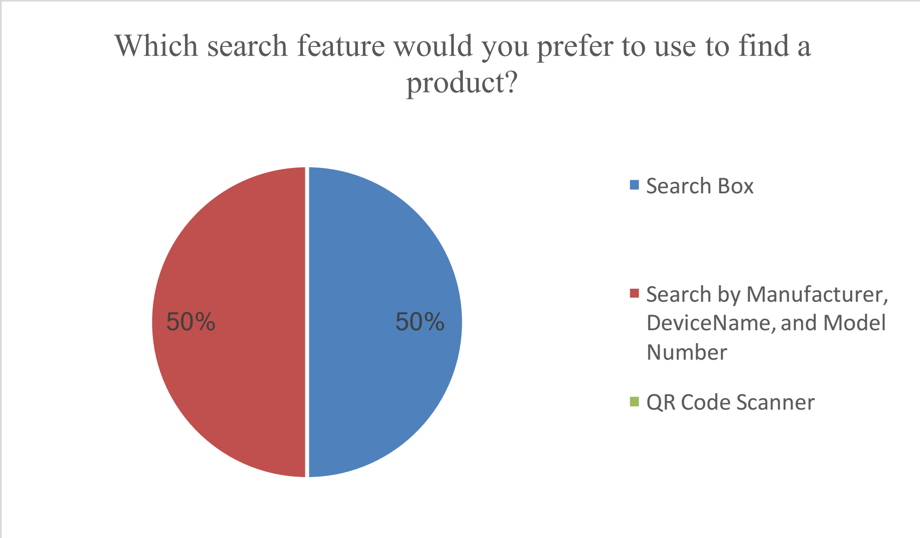

Figure 5.11: User Preference for Search Features Selection.

Figure 5.12 displays the distribution of whether participants know which product to buy when buying a replacement based on their search feature preference. For participants prefer using search box, 71% of them know which product to buy. While for participants choosing drop down box, 57% of them know which product to buy. Based on the data analysis, we can find that

Finding 11: Most of people know which product to buy before they are going to buy a replacement.

Finding 12: People using drop down box have more uncertainty of which product to buy rather than using search box.

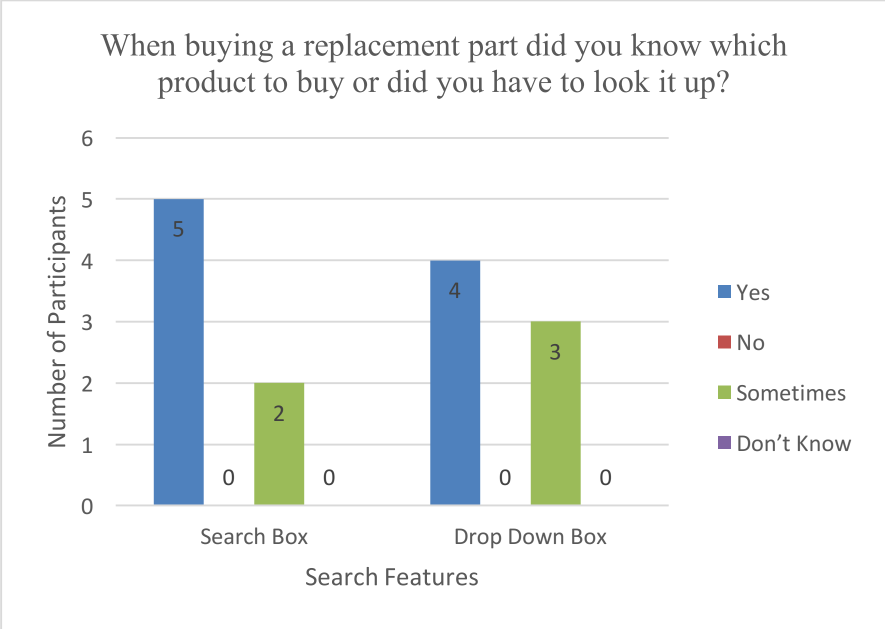

Figure 5.12: Users' Knowing about the Product when Buying a Replacement.

Q3. Do you think it is important to have more than one way to search in the app? Figure 5.13 displays the percentage of user thoughts about the importance of implementing multiple search features. The figure shows that about 86% of participants think it is important to have more than one way to search in the app. Therefore, we can find that

Finding 13: Implementing multiple search features is a factor to improve the usability of the search app.

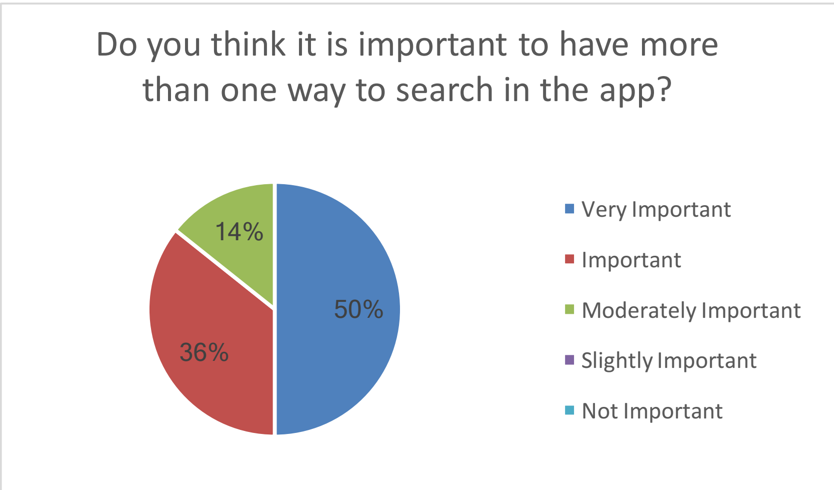

Figure 5.13: Users' Thoughts about Multiple Search Features Importance.

Q4. Where did you search for information on replacement power supplies?

Figure 5.14 displays the percentage of each channel where participants search information for which replacements suit their devices. 79% of participants indicate that Google is the best choice for them to find the potential matched replacement. This can be concluded that

Finding 14: Most of people prefer to use search engines to find out which product is most valuable.

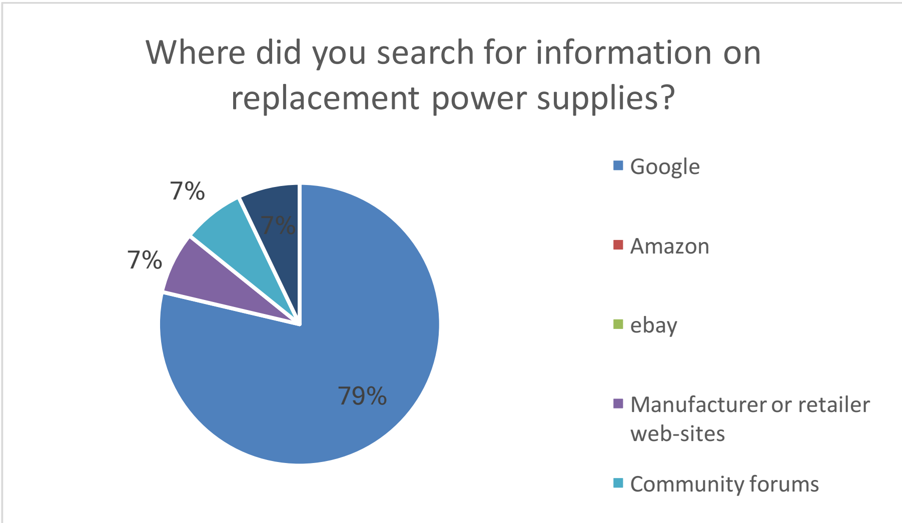

Figure 5.14: How People Search for Information on Replacement.

Q5. Other Comments.

As shown in table 5.25, 5 participants out of 7 are satisfied with the app usability and functionality.

5.5.6 Conclusion

As discussed above, it is clearly that 12 participants out of 14 agreed that the app has a good usability. 86% participants thought it is very important to provide different ways to search the product, and MyVolts mobile app implemented alternative ways such as search by input, search by drop down box, and QR code scanner. In addition, many participants also gave good compliments. Therefore, it can be concluded that the MyVolts mobile app is able to achieve high usability.

Chapter 6

Conclusions

The study aimed to investigate the possibility of a consumer-oriented, product matching Linked Data mobile application to achieve high usability. Through researching the background knowledge and the state of the art technologies, the proposed solution which aimed to support the further evaluation was designed. This project carried out with iterative usability studies with prototype refinements. Based on the analysis of these usability studies, different factors which could affect the usability such as the app features were observed and modified.

In this chapter, we summarize our project major findings, results, and contributions. In addition, the possible future work is discussed.

6.1 Progress on Research Question

As a reminder, here is the research question stated in chapter 1:

To what extent can consumer-oriented, product matching mobile applications based on Linked Data achieve high usability?

Overall, the MyVolts mobile application has achieved high usability. It can be reflected from the overall results from first three usability experiment: (1) The first experiment has shown that the initial prototype achieved "good" usability according to the SUS scale. (See Section 5.2) (2) The second experiment has shown that the second prototype achieved excellent SUS usability score and a higher usability than the initial prototype. (See Section 5.3) (3) The third experiment has shown that it is possible for MyVolts mobile application to achieve the equivalent usability as the pre-existing MyVolts web application for product matching. (See Section 5.4)

In addition, it was found from the last experiment that MyVolts mobile application interface design and features have got positive user feedback for example there was no disagreements with the app usability. (See Section 5.5)

6.2 Progress on Research Objectives

In this section, the main results and findings were concluded in relation to each research objective.

6.2.1 RO1: Review of the State of the Art

Overall, this research objective was satisfied. This project reviewed the background of the State of the Art including the overview of the Linked Data technologies, HCI, mobile usability, and evaluation methodologies, which covered the essential materials to support the further studies. Through reviewing the foundation of HCI design and well-known HCI model, it was found that user-centered design was highly recommended to improve the system usability. Based on the review of mobile usability, the mobile usability principles were highlighted. The review also indicated the potential issues and challenges towards applying Linked Data technologies (See Section 2.1.2) and achieving high usability on mobile devices (See Section 2.3.2).

Five Linked Data mobile applications were discussed and categorized in relation to technical approaches. Based on the comparison of these existing Linked Data mobile applications, it was decided to take the traditional client-server approach to retrieve information from the remote SPARQL endpoints and display into the user interfaces of the MyVolts application. In order to design a usable user interface allowing users to search the item, different user interface design patterns in relation to the app search functionality were reviewed and decided.

6.2.2 RO2 & RO3 & RO4: Design and Implementation of the Mobile Application

This research objective was achieved through iterative design and implementation of the MyVolts mobile application. The system use cases and the system requirements were outlined in relation to the implementation of the MyVolts mobile application. The proposed technical approach was also presented to support the implementation of the mobile application. Table 6.1 illustrates how the requirements to be applied to MyVolts mobile application.

The design and implementation of the mobile application carried out with two iterative stages. The initial prototype including *home* user interface and *search* user interface was implemented aiming to provide the basic query user interface in order to get the early feedbacks. The second prototype was improved based on the feedbacks from the initial prototype such as the additional features (e.g. search results list).

Table 6.1: How the requirements to be applied to the MyVolts mobile application

	Requirement	**Status**	**Comment**
R1	Information Access	Done	This requirement was achieved by querying the SPAQL endpoints.
R2	Information filters and visualize search results	Done	This requirement was achieved by providing different search methods and a List View to display the search results.
R3	Search by search box	Done	This requirement was achieved by providing method to allow users to search by these options.
R4	Search by drop down box	Done	
R5	Search by voice recognition	Done	
R6	Search by QR Code	Done	
R7	Visualize product description	Done	This requirement was achieved by providing a *Product Description* user interface to present information.
R8	Be usable to common users	Done	The users were able to use the app functions without relevant technical background.
R9	Avoid frequently querying the SPARQL endpoints	Done	This requirement was achieved by applying asynchronous method (e.g. image presentation) and the button to allow users to click and load information(e.g. lables in the *Product Desc* UI).
R10	Implement the real time search function	Done	Apart from the cached information to be presented in the drop down box select list, users can also search from the search box if there is no desired selection.
R11	Provide necessary feedbacks	Done	The MyVolts mobile application provides feedbacks including message dialogs and color changes.
R12	Use appropriate widget to display appropriate amount of information	Done	This requirement was achieved by discussing and comparing different widgets in order to design which one is best suited.
R13	Enable different level of resolution for different types of screens.	Done	Each element was sized to be displaying on the screen using percentage. Images were replicated and resized to adapt different screen resolution.
R14	Implement different user input methods	Done	Different user input methods were implemented including text input, voice recognition, and select box.

6.2.3 RO5: Evaluation of the Mobile Application

The evaluation of the MyVolts mobile application was established based on multiple relevant usability metrics such as efficiency, effectiveness, and satisfaction. The results from the initial prototype has achieved "good' usability (see section 5.2.4). The user feedbacks have indicated that the search patterns selected for MyVilts mobile application including different search methods, the search history and the popular recommendations can significantly improve the app usability.

The second experiment aimed to evaluate whether the second prototype achieved high ISO usability and higher SUS score from the initial prototype or not. The results have shown that usability metrics including effectiveness, efficiency, satisfaction have achieved higher value than average. Based on the comparison of overall usability scores between three groups of users including employees from MyVolts Ltd., people with CS background, and people from non-CS background, the results have shown that people with CS background feel more confident to interact with the mobile application. Finally, the SUS scores were compared between the initial prototype and the second prototype, it was found that the second prototype achieved much higher satisfaction than the initial one. In addition, through analyzing user errors, it was found that it is possible for users to miss their target item in the select box and they don't like to too many words. And through comparing the number of steps to achieve the same goal between using search box and drop down search box, it was found that using search box is more efficient.

The third experiment focused on the comparison of the usability between MyVolts mobile application and a pre-existing MyVolts web application. In order to indicate whether the two samples of the data were significantly different, the two sample t-test method was applied and the results have shown that there is a significant difference between two samples. The usability metrics were used to compare the usability between two applications in this experiment including effectiveness, efficiency, satisfaction. It was found the scores from two applications of all usability metrics were very closed, so we can conclude that it is possible for MyVolts mobile app to achieve the equivalent usability as web app.

The final experiment focused on the feedbacks from an online questionnaire for MyVolts customers that showcased the mobile app UI design and requirements. It was found that users agreed that the MyVolts mobile application has achieved high usability as there is no disagreement of their overall impression. The feedback also has shown that people using drop down search box have more uncertainty of which product to buy rather than using search box. And users also agreed that implementing multiple search feature can significantly improve the usability of the mobile application.

6.3 Future Work

The future work can be considered from some issues that have been identified from user feedbacks (2 & 5) in experiment 4 including adding the additional images of the product in order to allow users to identify the product easily and indicating whether users would like to search by ASIN. In addition, considered the limitation of amount of experiment participants, the possible future work can expand the evaluation among larger size of participants.

6.4 Final Remarks

Overall, it can be concluded that it is possible that consumer-oriented, product matching Linked Data mobile applications achieve high usability. The experiments include four iterative stages. And it covered different types of participants including employees from MyVolts Ltd, MyVolts customers, people with CS background, and people with non-CS background.

Users were able to complete tasks with minimum technical support and only occurred a total 3 errors. Users also thought the MyVolts mobile application is very easy to use. And they complemented the simplicity of the mobile application with the right functionality. Some users found they really had no problem with the app consistency. MyVolts was very happy with the design of the app and planned to do a follow-on project with the ADAPT center to build a full live version of the app for their customers.

The Linked Data challenges were overcome by implementing the alternative real time search method and avoiding to query the server frequently. The mobile usability challenges were also resolved by designing appropriate widgets for appropriate amount of information, implementing different input methods, and providing feedbacks for network unreachable conditions.

Bibliography

[1] W3C. What is the semantic web? http://www.w3.org/2001/sw/ Accessed: 7 July 2017.

[2] Bizer, C., Heath, T., Berners-Lee, T. (2009) Linked data - the story so far. International journal on semantic web and information systems. 5(3):1-22.

[3] Hausenblas, M. (2009) Exploiting linked data to build web applications. IEEE Internet Computing. 13(4): 68-73.

[4] Tillmann, N., Moskal, M., de Halleux, J., Fahndrich, M., Bishop, J., Samuel, A., Xie, T. (2012) The future of teaching programming is on mobile devices. In: Proceedings of the 17th ACM annual conference on Innovation and technology in computer science education, ACM. 156–161.

[5] Calvanese, D. (2017) Ontop: Answering SPARQL queries over relational databases. International journal on semantic web and information systems. 5(3):1-22.

[6] Yus, R., Pappachan, P. (2015) Are apps going semantic? a systematic review of semantic mobile applications. In SEMWEB.

[7] Dadzie, A., Rowe, M. (2011) Approaches to visualizing linked data: a survey. Semantic Web. 2(2):89-124.

[8] Dey, A., Abowd, G., Salber D (2001) A conceptual framework and toolkit

for supporting the rapid prototyping of context-aware applications. Hum-Comput Interact. 16:97-166.

[9] Zhang, D., Adipat, B. (2005) Challenges, methodologies, and issues in the usability of testing mobile applications. International Journal of human-computer interaction. 18(3):293-308.

[10] Akyildiz, I., Wang, X. (2005) A survey on wireless mesh networks. IEEE radio communications. 23-39.

[11] Siek, k.A., Rogers, Y. Connelly,K,H. Fat finger worries: how older and younger users physically interact with PDAs. In; Proceedings of INTERACT'05. 267-280.

[12] Brooke, J. (1996) A "quick and dirty usability" scale. In: Jordan Pw, Thomas B, Weerdmeester BA, McClelland AL, es. Usability evaluation in industry. Abingdon, UK: Taylor and Francis. 293-308.

[13] Tullis, T., Albert, B. (2012) Measuring the user experience. Morgan Kaufman.

[14] Altman, DG. (1998) Confidence intervals for number needed to treat. BMJ.

[15] Sauro, J., Lewis, J.R. (2005) Estimating completion rates from small samples using binomial confidence intervals: comparisons and recommendations. In: Proceedings of Human Factors and Ergonomics Society 49th Annual Meeting, Orlando, Florida.

[16] Sauro, J., Lewis, J.R. (2005) Estimating completion rates from small samples using binomial confidence intervals: comparisons and recommendations. In: Proceedings of Human Factors and Ergonomics Society 49th Annual Meeting, Orlando, Florida. 2100-2104.

[15] Stefik, M., Lavendel, G. (1997) Research and advanced technology for digital libaries. In: Proceedings of the First European Conference, Pisa, Italy.

[16] Sauro, J., Lewis, J.R. (2009) Correlations among prototypical usability metrics: evidence for the construct of usability. In; Proceedings of CHI. 1609-1618.

[17] Sauro, J., Lewis, J.R. (2010) Average task times in usability tests: what to report?. In; Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, Atlanta, Georgia, USA.

[18] ISO/IEC. (1998) 9241-14 Ergonomic requirements for office work with visual display terminals (VDT)s – Part 14 Menu dialogues. ISO/IEC 9214.

[19] Harrison, R., Flood, D., Duce, D. (2013) Usability of mobile applications: Literature review and rationale for a new usability model. Journal of Interaction Science. 1(1): 1-16.

[20] Abran, A., Khelifi, A., Suryn, W., Seffah, A. (2003) Usability meanings and interpretations in ISO standards. Software Quality Journal. 325-338.

[21] Ferre, X., Juristo, N. ,Windl,H., Constantine, L. (2001) Usability basics for software developer. IEEE Software. 22-29.

[22] Armen, G. (2015). 5 UX KPIs You Need to Track. https://designmodo.com/ux-kpi/#comments Accessed: 16 July 2017.

[23] Constantine, L.L., Lockwood, L.A.D. (1999) Software for use: a practical guide to the models and methods of usage-centered design. Reading, MA: Addison-Wesley.

[24] Picking, R., Grout, J. , Mcginn,J., Crisp, Grout, H. (2010) Simplicity, consistency, universality, flexibility and familiarity: the SCUFF principles for developing user interfaces for ambient computer systems. International Journal of Ambient Computing and Intelligence. 2(3) 40-49.

[25] Hewett, G. (1992). ACM curricula for human-computer interaction. http://old.sigchi.org/cdg/index.html. Accessed: 16 July 2017.

[26] Community, L,D. (2001). Linked data – connect distributed data across the web. http://linkeddata.org Accessed: 16 July 2017.

[27] W3C. (2001). Vocabularies. https://wwww.w3.org/standards/semanticweb/ontology Accessed: 22 July 2017.

[28] W3C. (2001). Vocabularies. https://wwww.w3.org/standards/semanticweb/ontology Accessed: 22 July 2017.

[29] Prud'hommeaux, E., Seaborne, A. (2017). SPARQL query language for RDF. W3C. http://www.w3.org/TR/rdf-sparql-query/. Accessed: 22 July 2017.

[30] P´erez, J., Arenas, M., Gutierrez, C. (2016) Semantics and Complexity of SPARQL. In: Proceedings of 4th International Semantic Web Conference (ISWC). Athens, GA, USA.

[31] Berners-Lee, T. (2006). Linked data – design issues. https://www.w3.org/DesignIssues/LinkedData.html . Accessed: 22 July 2017.

[32] Umbrich, J., Hausenblas, M., Hogan, A., Polleres, A., Decker, S. (2010) Towards Dataset Dynamics: Change Frequency of Linked Open Data

Sources. In: Proceedings of WWW2010 Workshop: Linked Data on the Web.

[33] W.R., Hage, N., Stash, Y., Wang, L., Aroyo. (2010) The Semantic Web: Research and Applications. In: Proceedings of 7th Extended Semantic Web Conference ESWC. Berlin. 330-331.

[34] Millard, I., Glaser, H., Salvadores, M., Shadbolt, N. (2010) Consuming multiple linked data sources: Challenges and Experiences. In: Proceedings of First International Workshop on Consuming Linked Data. 1-12.

[35] Bono, E. (1998) Simplicity. Penguin, London.

[36] David, K. (1997) Interactivity and Multimedia Interfaces. Instructional Science. 79-96.

[37] Shneiderman, B. (1998) Designing the User Interface.

[38] Renaud, K., Cooper, R. (2000) Feedback in humancomputer interaction: Characteristics and recommendations. In: Proceedings of South African Institute of Computer Scientists and Information Technologists. Annual Research Conference, Cape Town, South Africa. 105-114.

[39] HEWETT, T. T. ET AL., EDS. (1992) ACM SIGCHI Curricula for Human-Computer Interaction. ACM Press, New York.

[40] Suchman, L. A. (1987) Plans and Situated Actions: The Problem of Human Machine Communication. University of Cambridge Press, Cambridge, U.K.

[41] Norman, D.A. (1998) The Psychology of Everyday Things. Basic Books, New York.

[42] Vogel, D., Kennedy, D. M., Kwok, R. (2009) Does using mobile device applications lead to learning?. Journal of Interactive Learning Research. 469-485.

[43] Nathan, T. (2015) How to Use the System Usability Scale (SUS) to Evaluate the Usability of Your Website? http://usabilitygeek.com/how-to-use-the-system-usability-scale-sus-to-evaluate-the-usability-of-your-website/. Accessed: 16 July 2017.

[44] Rakkolainen, I., Vainio, T. (2001) A 3D City Info for mobile users. Computers & Graphics. 619-625.

[45] Becker, C., Bizer, C. (2008) DBpedia Mobile: A Location-Enabled Linked Data Browser. In: Proceedings of World Wide Web 2008 Workshop: Linked Data on the Web.

[46] Pietriga, E., Bizer, C., Karger, D., Lee, R. (2006) Fresnel: A browser-independent presentation vocabulary for RDF. In: Proceedings of International Semantic Web Conference. 158-171.

[47] Wilson. M., Russell. A., Smith. D., Owens, A. Schraefel, M. (2005) A browser-independent presentation vocabulary for RDF. In: Proceedings of End User Semantic Web Workshop and ISWC2005. Galway, Ireland.

[48] Seaborne. A., Bristol. H,L. (2004) RDQL a query language for RDF. W3C Member submission.

[49] Ermilov, T., Heino, N., Tramp, S., Auer, S. (2011) OntoWiki mobile - knowledge management in your pocket. In: Proceedings of the ESWC2011.

[50] Cano, A.E., Dadzie, A., Hartmann, M. (2011) Who's who – a linked data visualization too for mobile environments. In: Proceedings of the ESWC2011.

[51] Corsar, D., Edwards, P., Baillie, C., Markovic, M., Papangelis, K., Nelson, J. (2013) GetThere: A Rural Passenger Information System Utilising LinkedData & Citizen Sensing. In: Proceedings of the ISWC 2013 Posters & Demonstrations Track a track within the 12th International Semantic Web Conference. Sydney, Australia. 85-88.

[52] Auer, S., Bizer, C., Kobilarov, G., Lehmann, J., Cyganiak, R., Ives, Z.G. (2013) DBpedia: A nucleus for a web of open data. In: Proceedings of 6th International Semantic Web Conference (ISWC). 722-735.

[53] Walker, R. (2012) Designing for Ecommerce: The First Impression. https://adpearance.com/blog/designing-for-ecommerce-the-first-impression. Accessed: 17 August 2017.

[54] Nudelman, G. (2013) Android design patterns: interaction design solutions for developers. Wiley, Mar.

[55] Neil, T. (2012) Mobile design pattern gallery: UI patterns for mobile applications. O'Reilly Media, Inc.

[56] Morville, P., Callender, J. (2010) Search patterns: design for discovery. Sebastopol, CA.

[57] Bangor, A., Kortum, P.T., Miller, J.T. (2008) Technique for human error rate prediction (THERP) International Journal of Human-Computer Interaction 24. 574-594.

[58] Swain, A.D. (1964) Technique for human error rate prediction (THERP). Symposium on the Quantification of Human Performance.

[59] Sauro, J., Kindlund, E. (2005) A method to standardize usability metrics into a single score. In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems. Portland, Oregon, USA.

[60] Shaw, D. (1996) Handbook of usability testing: How to plan, design, and conduct effective tests. Journal of the American Society for Information Science. 47(3): 182-192.

[61] Babich, N. (2016) Mobile ux design: user-friendly search. https://uxplanet.org/mobile-ux-design-user-friendly-search-51e5f78f5a1e. Accessed: 26 August 2017.

[62] Ramsey, P.H. (2016) Exact Type I error rates for robustness of Student's t test with unequal variances. Journal of Educational Statistics, 5:337–349.

[63] Sum, X., May, A.A. (2013) A comparison of field-Based and lab-Based xxperiments to evaluate user experience of personalized Mobile Devices. In: Proceedings of the Hum-Comp. Hindawi.