**Deliverable 4: System Prototyping and Testing**

**Food Wastage Application: The Sustainable Spoonful**

**By**

**Lea Thumbiran, Lucinda Zachos, Melany Opperman, Santana Bradbury**

A Mini-Dissertation Submitted as a Partial Requirement for the Bachelor Science in Information Technology: Mobile Application and Web Services

In the Faculty of Information Technology, Eduvos

Supervisors:

Mr. Ian Masaga

Date: 15 June 2023

1. **Student Details**

|  |  |  |  |
| --- | --- | --- | --- |
| Student Details | Student Number | Telephone Number | Email Address |
| Bradbury, Santana | CNZJB3199 | 071 864 4170 | cnzjb3199@vossie.net |
| Opperman, Melany | 6P6NPJX46 | 074 709 6778 | 6p6npjx46@vossie.net |
| Thumbiran, Lea | LMTPQFTH6 | 079 898 2004 | lmtpqfth6@vossie.net |
| Zachos, Lucinda | Y34VR5C17 | 071 878 3138 | Y34vr5c17@vossie.net |

1. **Supervisor Details**

|  |  |  |
| --- | --- | --- |
| Supervisor Name | Faculty | Email Address |
| Masaga, Ian | IT | ian.masaga@eduvos.com |

**Table of Contents**

[Chapter 4: System Prototype Development and Testing 3](#_Toc137856647)

[4.1 Introduction 3](#_Toc137856648)

[4.2 Testing Plan 5](#_Toc137856649)

[4.3 Layouts Development 8](#_Toc137856650)

[4.3.1 User Interface – Landing Page 8](#_Toc137856651)

[4.3.1.1.1 Iteration 1 8](#_Toc137856652)

[4.3.1.1.2 Iteration 2 9](#_Toc137856653)

[4.3.2 User Interface - Registration 10](#_Toc137856654)

[4.3.2.1.1 Iteration 1 10](#_Toc137856655)

[4.3.2.1.2 Iteration 2 10](#_Toc137856656)

[4.3.3 User Interface - Login 11](#_Toc137856657)

[4.3.3.1.1 Iteration 1 11](#_Toc137856658)

[4.3.3.1.2 Iteration 2 11](#_Toc137856659)

[4.3.4 User Interface - Stores 12](#_Toc137856660)

[4.3.4.1.1 Iteration 1 12](#_Toc137856661)

[4.3.4.1.2 Iteration 2 12](#_Toc137856662)

[4.3.5 User Interface – Products 13](#_Toc137856663)

[4.3.5.1.1 Iteration 1 13](#_Toc137856664)

[4.3.5.1.2 Iteration 2 13](#_Toc137856665)

[4.3.6 User Interface – QR code 14](#_Toc137856666)

[4.3.6.1.1 Iteration 1 14](#_Toc137856667)

[4.3.6.1.2 Iteration 2 14](#_Toc137856668)

[4.4 Business Logic Development 15](#_Toc137856669)

[4.4.1 Algorithm 1 - Login 15](#_Toc137856670)

[4.4.1.1.1 Iteration 1 15](#_Toc137856671)

[4.4.1.1.2 Iteration 2 15](#_Toc137856672)

[4.4.2 Algorithm 2 - Register 15](#_Toc137856673)

[4.4.2.1.1 Iteration 1 15](#_Toc137856674)

[4.4.2.1.2 Iteration 2 15](#_Toc137856675)

[4.4.3 Algorithm 2 - Discount 15](#_Toc137856676)

[4.4.3.1.1 Iteration 1 15](#_Toc137856677)

[4.4.3.1.2 Iteration 2 15](#_Toc137856678)

[4.5 Data Access Development 16](#_Toc137856679)

[4.5.1 Database Implementation 16](#_Toc137856680)

[4.5.1.1 Database Creation Class – DatabaseHelper.java 16](#_Toc137856681)

[4.5.2 Data Access Adapters 17](#_Toc137856682)

[4.5.2.1 Accessor Methods 17](#_Toc137856683)

[4.5.2.1.1 Iteration 1 17](#_Toc137856684)

[4.5.2.1.2 Iteration 2 17](#_Toc137856685)

[4.6 Conclusion 17](#_Toc137856686)

[List of Figures 18](#_Toc137856687)

[List of Tables 18](#_Toc137856688)

[References 19](#_Toc137856689)

# System Prototype Development and Testing

## Introduction

Our goal with The Sustainable Spoonful has been to develop a mobile application assist in mitigating food wastage amongst consumers and retailers. Below we will attempt to provide some additional information regarding our implementation plan and our overall strategy when approaching this applications development.

When approaching the system prototype development and testing aspect of The Sustainable Spoonful mobile application we needed to consider the process we would follow to ensure we created a visually appealing, user-friendly mobile application that met our functional requirements. This would involve several stages.

These stages include layouts development, business logic development, data access development and testing.

The layouts development stage was where we attempted to create recreate and build upon our prototype from deliverable 3, which as it stands consists of a user registration screen, a login screen, store listing screen, discounted product screen and the discount code screen. In this stage we need to ensure we are prioritizing the user experience by creating something that is both easy to navigate and responsive while remaining consistent as well as conducting usability testing based on user interactions and experiences. (Soegaard, 2015)

The business logic development stage was where we attempted to translate our business requirements into functional code. During this stage we need to ensure we have clearly defined our business requirements so that can translate it into functional code as well as attempting to follow best practices, ensuring maintainability, readability, and scalability. We can achieve this by adopting an agile approach and adjusting our code throughout the process. (Fowler, 2017)

The data access development stage was where we designed our data access layer that will interact with our applications backend systems and the SQLite database, and finally testing where we meticulously tested our application to ensure its quality and reliability. During this stage we also need to ensure that we maintain data integrity as well as ensuring that user data remains secure. We also need to design an efficient as well as scalable database to minimize response times by optimizing data retrieval. (Buckbee, 2021)

Finally, with regards to testing, we need to establish a testing strategy and create test cases, ensuring we test all features and functionalities of the application as well as documenting reported issues to address them promptly to maintain a stable and reliable application. (Hamilton, 2023)

During these various stages, we have allocated different responsibilities to different team members to ensure clarity and responsibility throughout the planning and development process.

|  |  |  |
| --- | --- | --- |
| **Roles and Responsibilities** | | |
| **Name** | **Role** | **Responsibilities** |
| Melany | Frontend Developer and proofreader | Focus on designing and developing the visual aspects of the application. |
| Lea | Backend Developer and proofreader | Focus on developing the functional aspects of the application |
| Lucinda | Documentation and testing | Compile project documentation |
| Santana |  |  |

Table 1: Roles and responsibilities

Our goal with this project is to apply our research towards building an application that can make a difference.

## Testing Plan

During the stage of this deliverable, we will be evaluating whether the average user would the application easy to navigate. In this case we have chosen to focus on usability testing. We have created a series of test cases for our users to work through, once done we have setup a survey for them to answer and give us a bit of insight into their experience.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Testing Schedule** | | | | |
| **Increment** | **Test Type** | **Test Date** | **Team Members** | **…** |
| 1 | Usability Testing | 18/06/2024 | Lucinda, Lea | Testing individual units/components of the app in isolation. |

Table 2: Test Plan

(Hamilton, 2023)

Objective:

* To understand how real users will interact with our application and make changes based on those results. (Optimizely, 2023)
* Ensure the application is effective, efficient, engaging, error tolerant and easy to learn.
  + **Effective**: How accurately were tasks completed and how often did they produce errors.
  + **Efficient**: Evaluate the time taken to complete tasks.
  + **Engaging**: Gauge the users’ feelings towards the application with surveys or interviews.
  + **Error tolerant**: Create test case scenarios with the potential to cause errors.
  + **Easy to learn**: Try to get test users from different technical knowledge backgrounds to determine how user friendly the application is.

(Quesenbery, 2023)

Scope:

* The test will include the application in its entirety.
  + Main screen
  + Login screen
  + Registration screen
  + Store screen
  + Discounts screen
  + Account Screen
* This includes content and navigation.

(Usability.gov, 2023)

Components:

1. Set up a usability test consisting of various tasks the user would need to accomplish. These tasks will include:
   * Creating an account
   * Logging in to their account
   * Browsing the available discounts
   * View Account
   * Logout
2. Setup up a survey for users to complete after they have tested the mobile application.

(Optimizely, 2023)

Equipment:

* Testing will be conducted via the user’s mobile phones.

Test Cases:

* **Test 1**: The user installs the application and opens it for the first time. They need to register for an account so that they can view the application and view the discounted products on offer.
  + In this scenario, the user would need to open the app, register an account, login and then locate the discounted products screen.
* **Test 2**: The user closes the app after viewing the discounted products screen. If they reopen the application, will they need to re-enter their login details?
* **Test 3**: The user needs to navigate to the accounts screen. If they logout, will they be redirected to the login screen?
* **Test 4**: The user is asked to look for a discount item from Woolworths. Could they easily navigate to it?
* **Test 5**: The user is asked to navigate to the accounts screen and then navigate back to the discounted products page using the menu option. Were they able to easily navigate it?

User Survey Questions:

1. Was everything readable?
2. Were you able to create an account?
3. Were you able to login in?
4. Was the app responsive?
5. Was the app easy to navigate?
6. Did you the app engaging?
7. Did you find the app to be aesthetically pleasing?
8. Were you able to complete test case 1 without any issues? If not, why?
9. Were you able to complete test case 2 without any issues? If not, why?
10. Were you able to complete test case 3 without any issues? If not, why?
11. Were you able to complete test case 4 without any issues? If not, why?
12. Were you able to complete test case 5 without any issues? If not, why?
13. What were your thoughts on the app’s current features?
14. Is there anything you would change?
15. Did you entire any errors?

Survey/ test results:

## Layouts Development

### User Interface – Landing Page

##### Iteration 1

A screenshot of a blue screen

Description automatically generated with low confidence

*Figure 1 - Landing screen when the user opens the application for the first time.*

##### Iteration 2

A screen shot of a cell phone

Description automatically generated with medium confidence

*Figure 2 - Landing Page when the user opens the application for the first time – Emulator Pixel 6 Pro API 30*

This is the screen the user will see when they open the application. From here they can either register for an account or login with their login details. If the user is already logged in, they will be directed to the landing page.

Code Snippet

### User Interface - Registration

##### Iteration 1

A screen shot of a phone

Description automatically generated with low confidence

*Figure 3 - Registration screen so that users can create an account.*

##### Iteration 2

A screen shot of a cell phone

Description automatically generated with low confidence

*Figure 4 – Registration screen*

The registration screen allows the user to register an account. The user will need to fill in all fields of the registration form. If there is successful, their registration details will be stored in the database and the user will be redirected to the login screen. An alert will be displayed notifying the user that registration was successful.

Code Snippet

### User Interface - Login

##### Iteration 1

A screen shot of a phone

Description automatically generated with medium confidence

*Figure 5 - Login Page so that users can login using the account they have just created.*

##### Iteration 2

A screen shot of a cell phone

Description automatically generated with low confidence

*Figure 6 – Login Screen*

The login screen allows the user to enter their login credentials. The email address and password need to match the email address and password stored in the database. Once verified the user will be redirected to the landing page.

Code Snippet

### User Interface – Retail Partners

##### Iteration 1

A screen shot of a phone

Description automatically generated with medium confidence

*Figure 7 – Retail partners page for the Sustainable Spoonful Mobile Application.*

##### Iteration 2

A screen shot of a cell phone

Description automatically generated with medium confidence

*Figure 8 – Retail partner’s page*

Code Snippet

### User Interface – Products

##### Iteration 1

A screenshot of a cell phone

Description automatically generated with medium confidence

*Figure 9 - Discounts listed for a specific store.*

##### Iteration 2

Code Snippet

### User Interface – QR code

##### Iteration 1

A screen shot of a cell phone

Description automatically generated with medium confidence

*Figure 10 - QR code for the selected discounted product that users can scan at the store to collect their item.*

##### Iteration 2

Code Snippet

## Business Logic Development

### Algorithm 1 – Login

Code Snippet:

### Algorithm 2 - Registration

Code Snippet:

### Algorithm 2 - Discount

Code Snippet:

## Data Access Development

### Database Implementation

For our application we have chosen to work with SQLite.

SQLite is an embedded relational database management system that is widely used in mobile development. It is a lightweight, self-contained, reliable database engine that is available locally and does not require any administration to function. SQLite also comes built into Android Studio making it the ideal choice. (SQLite, 2023)

To implement it, we included it in our gradle.build file and created a new class called DatabaseHelper. We then created our tables and provided options for it to be created in other classes when the on create function is called. If it has been upgraded since its last use, we drop the table and recreate it again. We did this to ensure the code remains reusable.

#### Database Creation Class – DatabaseHelper.java

### Data Access Adapters

In this subsection, discuss and illustrate the class used to access the data from the database. Snippets of code can be shown together with screenshots of layouts showing the extracted data.

#### Accessor Methods – Account

## Conclusion

In summary, we have approached the system prototyping and testing phase using multiple methods.

These methods include building on our initial designs to develop a visually appealing application, focusing on the core functionality of our application and using unit testing to ensure it is functioning as it should be, finding the most appropriate database implementation technology to better manage our user data, and combing these aspects to create a functioning mobile application.

Using iterative and incremental design we have managed to build on our application first conceptualized in deliverable one. While this remains an ongoing process, we have made great strides in building on our initial concepts and move further towards a mobile application that meets all our requirements.

# List of Figures

[*Figure 1 - Landing screen when the user opens the application for the first time.* 9](#_Toc137949637)

[*Figure 2 - Landing Page when the user opens the application for the first time – Emulator Pixel 6 Pro API 30* 9](#_Toc137949638)

[*Figure 3 - Registration screen so that users can create an account.* 10](#_Toc137949639)

[*Figure 4 – Registration screen* 11](#_Toc137949640)

[*Figure 5 - Login Page so that users can login using the account they have just created.* 12](#_Toc137949641)

[*Figure 6 – Login Screen* 12](#_Toc137949642)

[*Figure 7 – Retail partners page for the Sustainable Spoonful Mobile Application.* 13](#_Toc137949643)

[*Figure 8 – Retail partner’s page* 14](#_Toc137949644)

[*Figure 9 - Discounts listed for a specific store.* 14](#_Toc137949645)

[*Figure 10 - QR code for the selected discounted product that users can scan at the store to collect their item.* 15](#_Toc137949646)

# List of Tables

[Table 1: Roles and responsibilities 5](#_Toc137949647)

[Table 2: Test Plan 6](#_Toc137949648)

# References

Anon, 2023. *How To Write An Effective Test Summary Report.* [Online]   
Available at: https://www.softwaretestinghelp.com/test-summary-report-template-download-sample/

Buckbee, M., 2021. *Varonis.com.* [Online]   
Available at: https://www.varonis.com/blog/data-integrity  
[Accessed 17 June 2023].

Clifford Stein , Robert Drysdale, Kenneth Bogart, 2011. *Discrete Mathematics for computer scientists.* United states of America: Pearson Education.

Contributors, W., 2023. *Unit testing.* [Online]   
Available at: https://en.wikipedia.org/w/index.php?title=Special:CiteThisPage&page=Unit\_testing&id=1156746873&wpFormIdentifier=titleform

Fowler, M., 2017. *Martin Fowler.* [Online]   
Available at: https://martinfowler.com/  
[Accessed 17 June 2023].

Gaddis, T., 2013. *Starting out with Programming Logic & Design.* United states: Pearson Education.

Gillis, A. S., 2023. *user acceptance testing (UAT).* [Online]   
Available at: https://www.techtarget.com/searchsoftwarequality/definition/user-acceptance-testing-UAT

Hagarty, R., 2002. *Discrete Mathematics for computing.* Malaysia: Pearson Education limited.

Hamilton, T., 2023. *Guru99.* [Online]   
Available at: https://www.guru99.com/unit-testing-guide.html  
[Accessed 17 June 2023].

Hamilton, T., 2023. Integration Testing: What is, Types with Example. *GURU99,* p. 5.

Hamilton, T., 2023. Performance Testing Tutorial – Types. *GURU99,* p. 5.

Hamilton, T., 2023. TEST PLAN: What is, How to Create. *GURU99,* pp. 3-4.

JavaTPoint, 2021. Functional Testing. *JavaTPoint,* p. 12.

JavaTpoint, 2023. Performance Testing. *JavaTpoint,* p. 2.

JRebel, 2023. *How to Use Java Integration Testing.* [Online]   
Available at: https://www.jrebel.com/blog/how-to-use-java-integration-testing

Kolman, B., 1997. *Introductou Linear Algebra with applications.* New Yersey: Prentice -Hall, Inc.

Nalawade, K., 2023. *How to Write Unit Tests in Java.* [Online]   
Available at: https://www.freecodecamp.org/news/java-unit-testing/#:~:text=What%20is%20Unit%20Testing%3F,as%20expected%20before%20releasing%20it.

pandaquests, 2023. *Acceptance testing in JavaScript.* [Online]   
Available at: https://levelup.gitconnected.com/acceptance-testing-in-javascript-ca1db5469fe9

pp\_pankaj, 2023. *Software Testing – Functional Testing.* [Online]   
Available at: https://www.geeksforgeeks.org/software-testing-functional-testing/

RichardJohnsonbaugh, 2009. *Discrete Mathematics.* United states of America: Pearson Education.

Soegaard, M., 2015. *Interaction Design.* [Online]   
Available at: https://www.interaction-design.org/literature/article/usability-a-part-of-the-user-experience  
[Accessed 17 June 2023].

SQLite, 2023. *SQLite.* [Online]   
Available at: https://www.sqlite.org/index.html  
[Accessed 16 June 2023].