

An Android App For Food Allergy Sufferers To Search For And Review Restaurants

DT265

Higher Diploma in Computing

**Name: Maeve Rooney**

**Supervisor: Michael Collins**

**<Second Reader>**

School of Computing

Dublin Institute of Technology

**<Date>**



Abstract

Declaration

I hereby declare that the work described in this dissertation is, except where otherwise stated, entirely my own work and has not been submitted as an exercise for a degree at this or any other university.

Signed:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

<Student Name>

<Date>

Acknowledgements

Body text

Table of Contents

1. Introduction

1.1 Background

1.2 Project Objectives

1.3 Project Challenges

1. Technologies Researched

2.1 Introduction

2.2 Eclipse and Java

2.3 Third party web server – x10hosting.com

2.4 TextWrangler and php, mysql

2.5 Android platform

2.6 Google Maps API

1. Architecture & Design

4.1 Introduction

4.2 System Architecture

4.3 Use Case Design

4.4 Design Methodology

4.5 Development Environment

4.6 Web Server and MySQL Design

4.7 Local SQLite database

4.8 User Interface Design

4.9 List of Features

1. Development & Implementation
   1. Details of each component within the project, problems encountered and resolved, challenges overcome or worked around.
   2. Identify key development components
2. System Validation
   1. Testing: What testing was performed, why it was selected and what are the key use cases within the project.
3. Project Plan
   1. Project Plan analysis and review of how it changed from the initial proposal including explanation of what changed and why, and suggestions on how to address this if the project was repeated.
4. Conclusion
   1. Analysis of the projects key elements identify the key learning obtained from the project and recommendations and suggestions for how the work can be improved on continued into the future.
5. Bibliography
6. Appendix

­

*(Note: For bibliography use IEEE or Harvard referencing styles)*

**Problems**

* Starting activity from map overlay item- finding context
* Passing paramters to async task
* Using name pair values to post http request to php
* Getting context in itemized overlay class
* Pass restaurant id into new activity to change content dynamically
* Progress dialog while switching activities
* Phone and email
* Mergeadapter for listview with header in restaurantview
* Store passwords with salt encryption
* Use ajax to check if unique username or email already in use
* Favoutrite icons on map markers
* Check if user logged in on map and list view
* Show only restaurants with rating of more than 2 for each of users allergies
* Write review. Locate on map
* Add review to restauratn on map
* Place marker at users current location. Drag to change and make note of new geopoint
* Add heart to list view
* Blacklist
* Put rating bar indicator in alertbox

1. **Introduction**

1.1 Background

1.2 Project Objectives

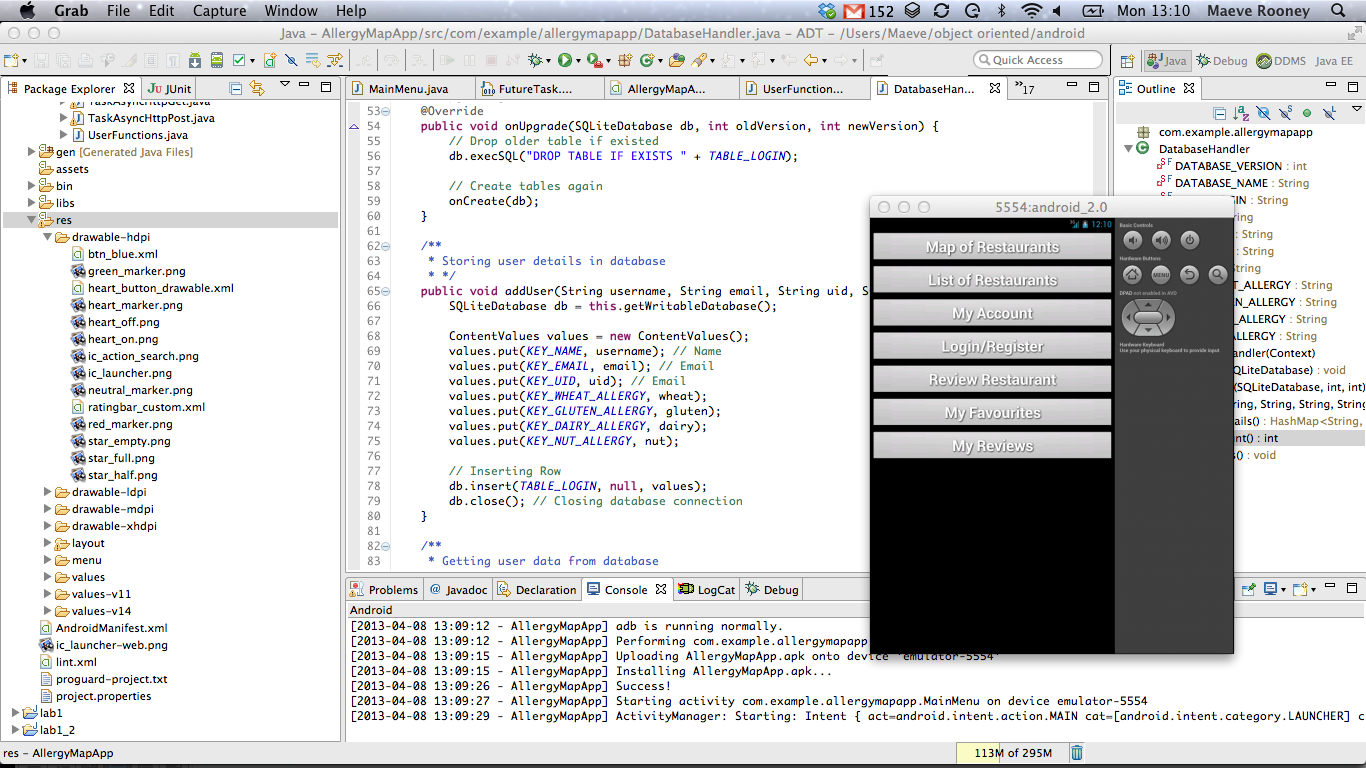
1.3 Project Challenges

1. **Technologies Researched**

2.1 Introduction

2.2 Eclipse and Java

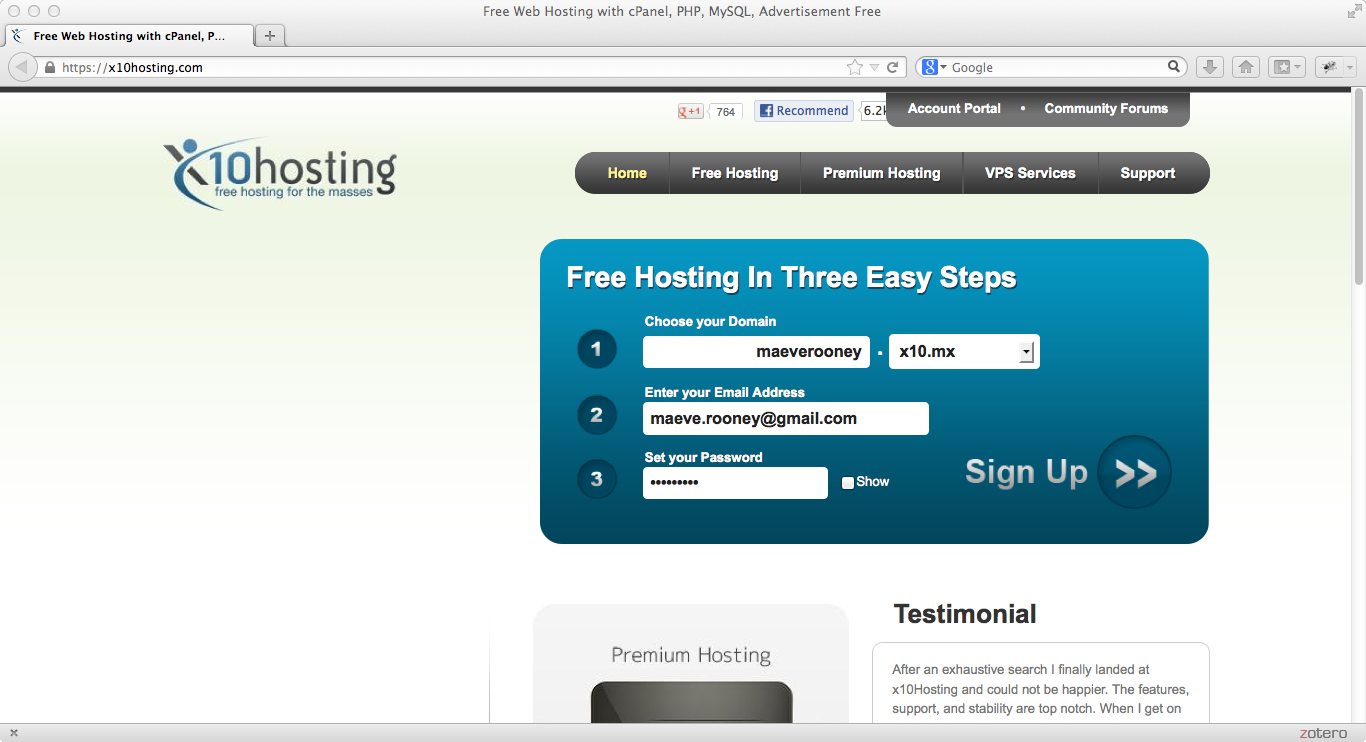
Eclipse Development Environment



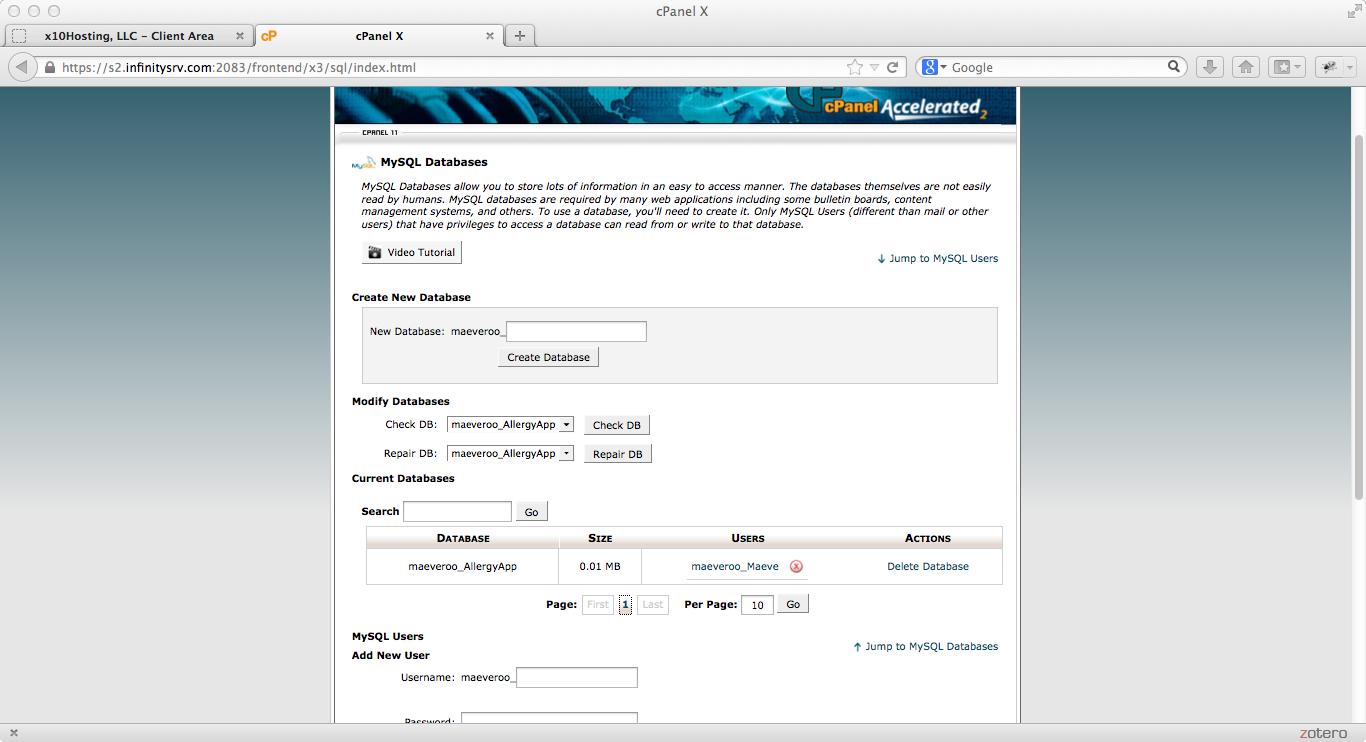
**Development Device – Samsung Galaxy Ace**

2.3 Third party web server – x10hosting.com

Setup free Website with MySQL



**Setup MySQL database using web servers cPanel**

****

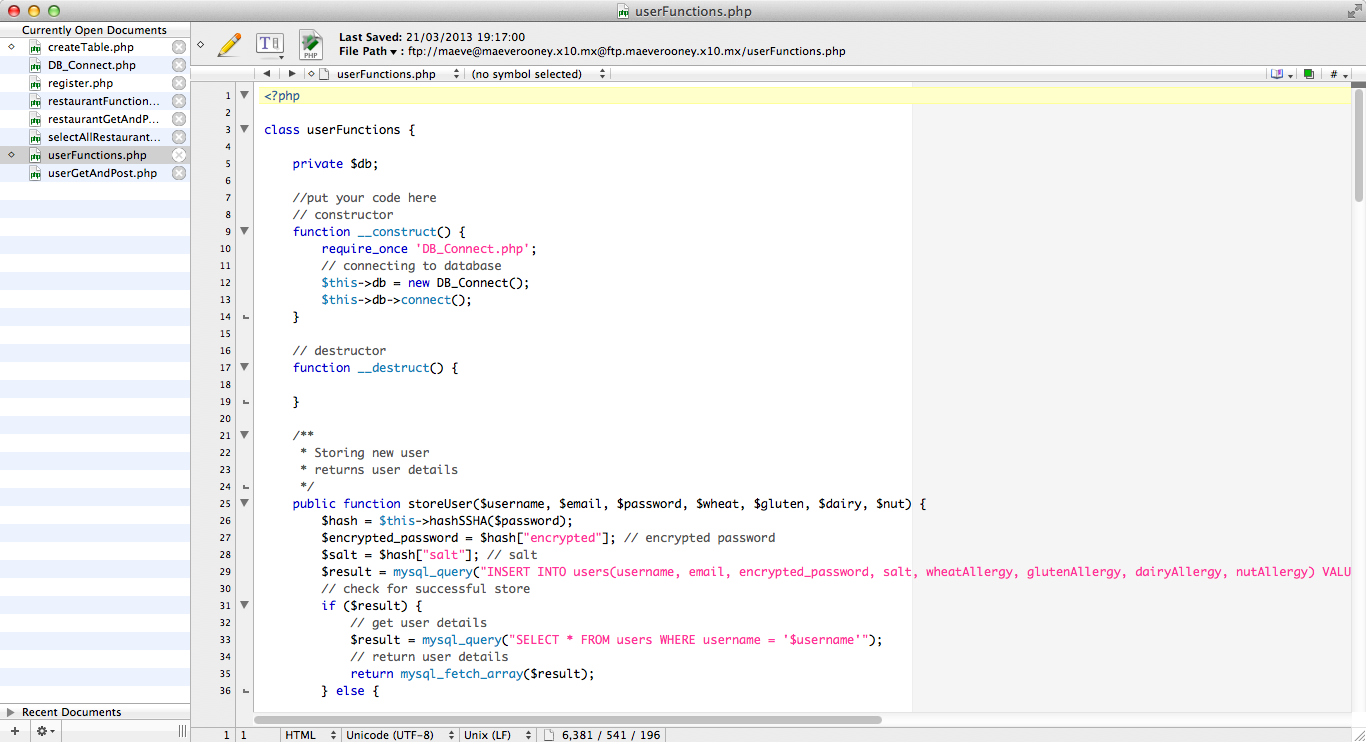
2.4 FTP Program – CyberDuck

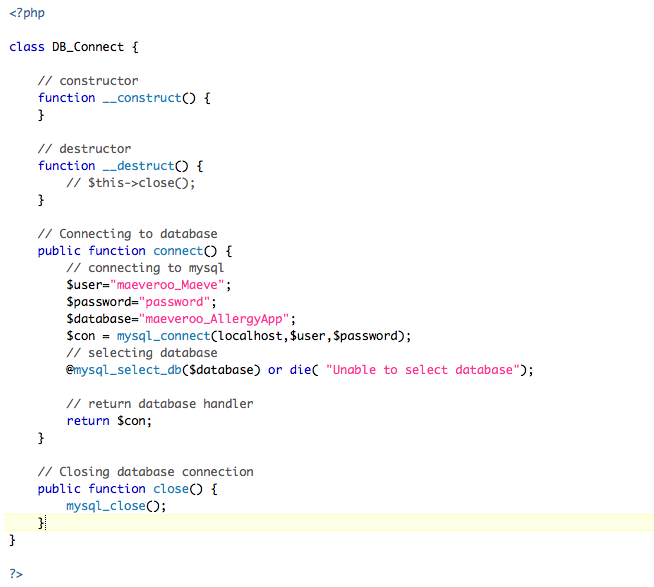
Connect to web server to manage file transfer



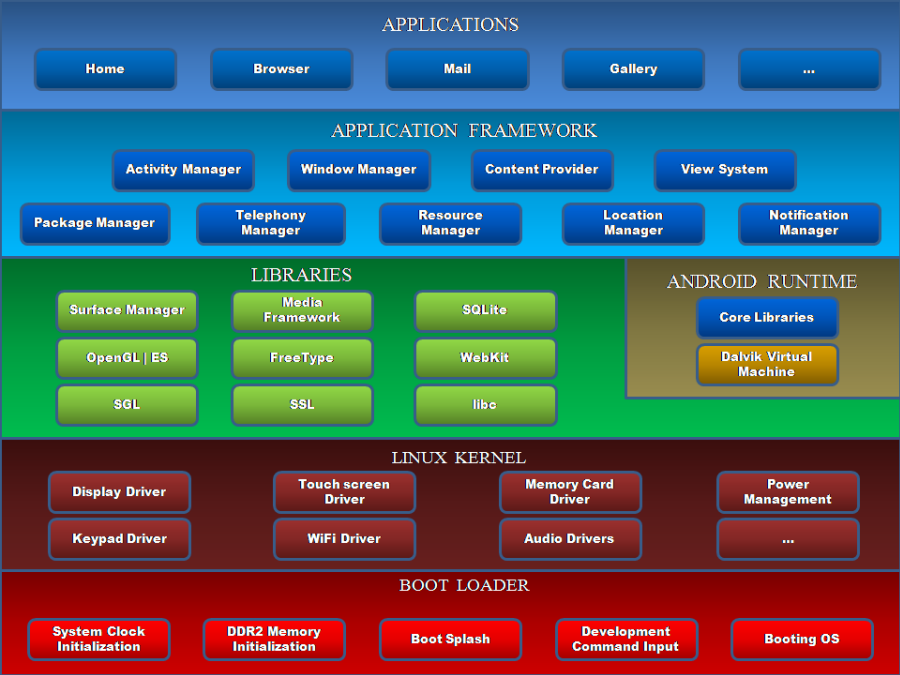
2.5 TextWrangler and php, mysql

TextWrangler used to write php and html for web server

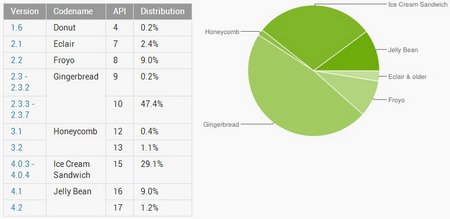


**PHP script to connect to MySQL database**

2.6 Android platform



**Versions of Android in Market 2013**

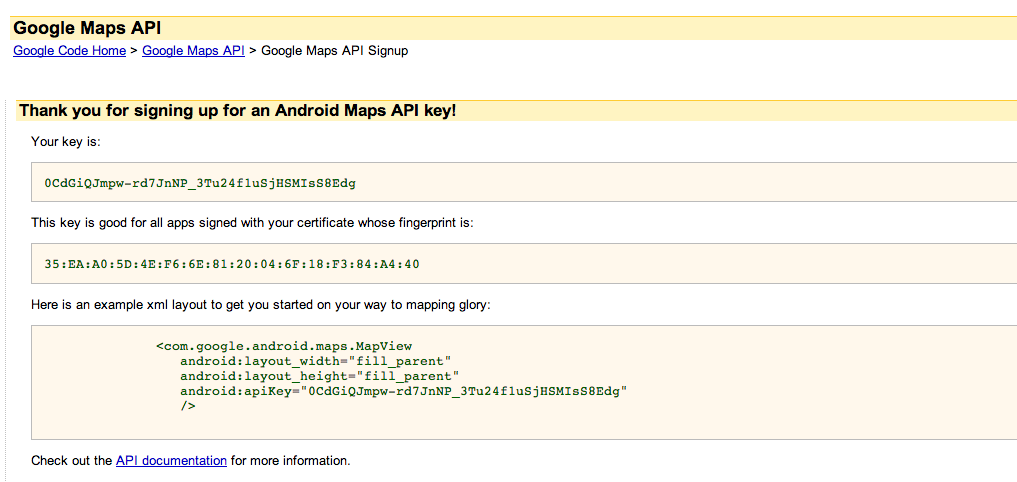
****

This app is optimised for the JellyBean (API 17) version but is working well on the Gingerbread version which the Samsung Galaxy Ace runs on.

2.6 Google Maps API

To develop an Android app that will display Google Maps data using the API provided in the Maps external library, you must register with the service and get a Google Maps Android API v1 Key.

**Get a Google maps key for android**

****

**Use key in xml layout of MapView activities**

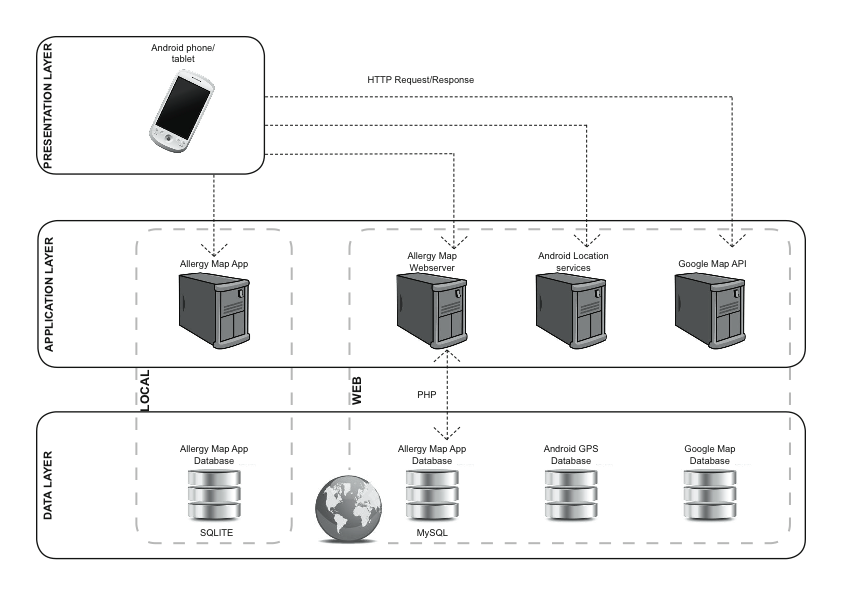


1. **Architecture & Design**

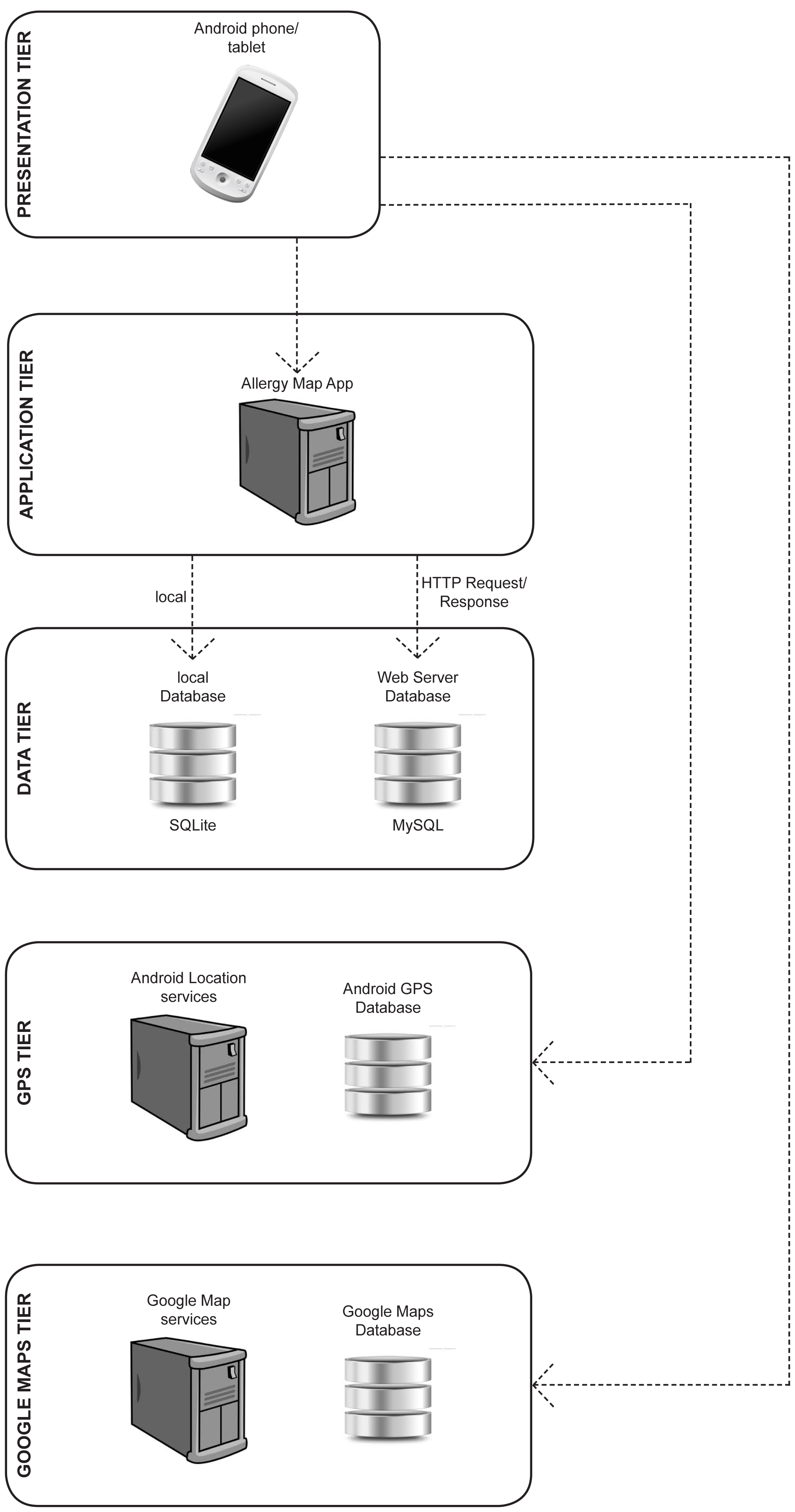
**4.1 Introduction**

**4.2 System Architecture**

**Initial System Architecture Diagram**

****

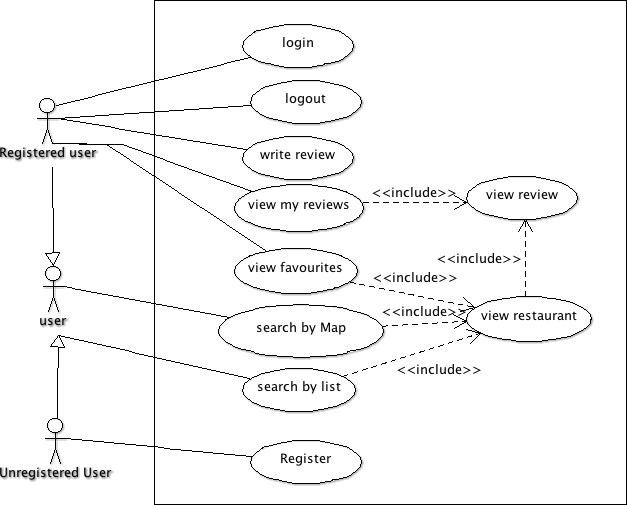
**Final System Architecture Diagram**



**5 Tiers**

1. Presentation Tier.
2. Application Tier.
3. Data Tier.
4. Android GPS Tier.
5. Google Maps Tier.

**4.3 Use Case Design**

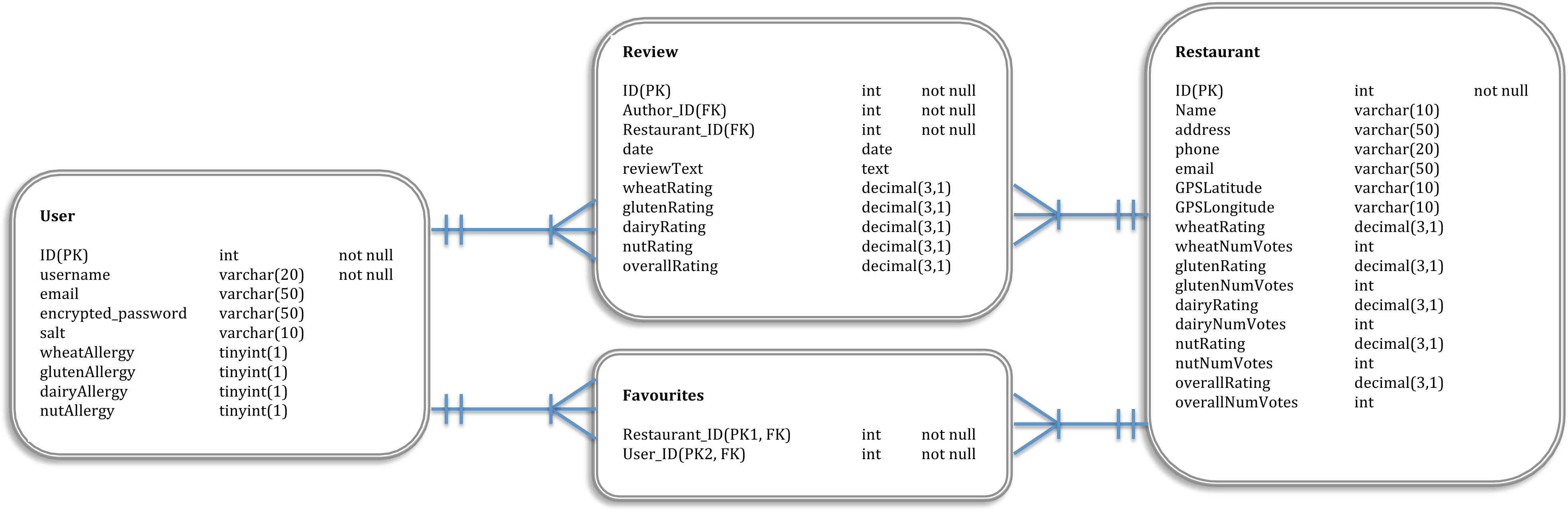
****

**4.4 Design Methodology**

**4.5 Development Environment**

**4.6 Web Server and MySQL Design**

**ERD Diagram for Web Server MySQL database**

****

**User Table**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| FIELD | TYPE | SIZE(byte) | NULL | DEFAULT |
| ID (PK) | int | 4 | no | Auto-increment |
| username | varchar | 20 | No |  |
| email | varchar | 50 | no |  |
| Encrypted\_password | varchar | 50 | no |  |
| Salt | varchar | 10 | no |  |
| wheatAllergy | tinyint | 1 | yes |  |
| glutenAllergy | tinyint | 1 | yes |  |
| dairyAllergy | tinyint | 1 | yes |  |
| nutAllergy | tinyint | 1 | yes |  |

**Restaurant Table**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| FIELD | TYPE | SIZE(byte) | NULL | DEFAULT |
| ID (PK) | int | 4 | no | Auto-increment |
| name | varchar | 30 | No |  |
| address | text | <=65535 | yes |  |
| email | varchar | 50 | yes |  |
| GPSLatitude | varchar | 10 | no |  |
| GPSLongitude | varchar | 10 | no |  |
| wheatAllergy | decimal | 8 | yes |  |
| wheatNumVotes | int | 4 | yes |  |
| glutenAllergy | decimal | 8 | yes |  |
| glutenNumVotes | int | 4 | yes |  |
| dairyAllergy | decimal | 8 | yes |  |
| dairyNumVotes | int | 4 | yes |  |
| nutAllergy | decimal | 8 | yes |  |
| nutNumVotes | int | 4 | yes |  |
| overallRating | decimal | 8 | yes |  |
| overallNumVotes | int | 4 | yes |  |
| numFavourites | int | 4 | yes |  |

**Review Table**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| FIELD | TYPE | SIZE(byte) | NULL | DEFAULT |
| ID (PK) | int | 4 | no | Auto-increment |
| Author\_ID (FK) | int | 4 | no |  |
| Restaurant\_ID (FK) | int | 4 | no |  |
| date | date | 3 | no |  |
| reviewText | text | <= 65535 | yes |  |
| wheatRating | decimal | 8 | yes |  |
| glutenRating | decimal | 8 | yes |  |
| dairyRating | decimal | 8 | yes |  |
| nutRating | decimal | 8 | yes |  |
| overallRating | decimal | 8 | yes |  |

**Favourites Table**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| FIELD | TYPE | SIZE(byte) | NULL | DEFAULT |
| User\_ID (PK1, FK) | int | 4 | no |  |
| Restaurant\_ID (PK2, FK) | int | 4 | no |  |

**4.7 Local SQLite Design**

**User Table**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| FIELD | TYPE | SIZE(byte) | NULL | DEFAULT |
| ID (PK) | int | 4 | no | Auto-increment |
| Unique\_ID | int | 4 | no |  |
| username | varchar | 20 | No |  |
| email | varchar | 50 | no |  |
| wheatAllergy | tinyint | 1 | yes |  |
| glutenAllergy | tinyint | 1 | yes |  |
| dairyAllergy | tinyint | 1 | yes |  |
| nutAllergy | tinyint | 1 | yes |  |

The local database stores one table that is used to check if a user is registered/logged in. It does not store the user’s password locally for security reasons. A connection to the web server database has to be made to validate the user against the password stored there. The primary key of the user in the web server database is stored in the local database in the field “Unique\_ID”. This way, any altering of the web server database (eg adding a review) can point to the correct user using the “Unique\_ID” stored locally. It minimizes the amount of calls to the web server needed, until the user logs out.

**4.8 User Interface Design**

|  |  |
| --- | --- |
| Macintosh HD:Users:Maeve:Dropbox:Android Allergy App:infographics:screen grabs:main menu.tiff  Main Menu | Macintosh HD:Users:Maeve:Dropbox:Android Allergy App:infographics:screen grabs:register.tiff  Register An Account |
| Macintosh HD:Users:Maeve:Dropbox:Android Allergy App:infographics:screen grabs:map view.tiff  Search By Map | Macintosh HD:Users:Maeve:Dropbox:Android Allergy App:infographics:screen grabs:list view.tiff  Search By List |
| Macintosh HD:Users:Maeve:Dropbox:Android Allergy App:infographics:screen grabs:restaurant view.tiff  Restaurant View | Macintosh HD:Users:Maeve:Dropbox:Android Allergy App:infographics:screen grabs:review screen.tiff  Review Restaurant |

**4.7 List of Features**

1. **Development & Implementation**
   1. Details of each component within the project, problems encountered and resolved, challenges overcome or worked around.
   2. Identify key development components
2. **System Validation**
   1. Testing: What testing was performed, why it was selected and what are the key use cases within the project.
3. **Project Plan**
   1. Project Plan analysis and review of how it changed from the initial proposal including explanation of what changed and why, and suggestions on how to address this if the project was repeated.
4. Conclusion
   1. Analysis of the projects key elements identify the key learning obtained from the project and recommendations and suggestions for how the work can be improved on continued into the future.
5. **Bibliography**
6. Appendix