National College of Ireland

BSc in Computing

2013-2017

Jelico Bachin

Cyber Security

X13301011

X13301011@student.ncirl.ie

Bestcarparks App

Technical Report

Table of Contents

Executive Summary

This project consists of developing a mobile application for my customer Bestcarparks. The mobile application will consist of multiple linked interfaces allowing Bestcarparks customers to book parking on the go from their Apple iPhone smartphones. The application will allow users to securely pay for their booking from their mobile device. The development of this application is very important as the customers of Bestcarparks will have the option of booking parking on the go while they can see the live spaces available in each car park. The security importance is significantly high in this project as the application will be accepting user sensitive information such as passwords. It is important to make sure the application is secure this is done through user sessions, server side validation and secure user passwords. The database is secured and the password are encrypted using PHP password_default encryption. Real time car parking availability will be displayed to the user to ensure customers are aware of the congestion in a car park and the number of free spaces. Users will have the possibility to edit bookings that have not expired, allowing them to change the date, location and vehicle registration number. Users can add, edit and delete saved vehicles at any time using the app. These details are then grabbed when making a booking and everything is automated to make it quick and easy when making a booking. The application will show users the available parking locations with a summary of prices and a live google map of the location. None of the booking services, user details will not be available until the user logs in or registers an account. Payments shall be virtually processed through the Stripe Payment system.

1 Introduction

The aim of this document is to provide all the technical details of the Mobile Application developed for Bestcarparks Ireland. The application will provide multiple booking features described in the document below.

Background

The idea of this concept inspired me early last year. After conducting market research in the car parking business in relation to applications for smartphones or smart devices, it led to the discovery that a parking app already existed for the Dublin City Council parking spaces located throughout Dublin City. The app was called Parking Tag and allowed users to pay for public parking from their smartphone. The main factor that customers look at when booking a car park space is the daily rate.

The daily rates were very high compared to any of the private car parking companies available in Ireland and no daily, weekly, monthly or yearly parking services were available. This made me conduct market research particularly in the private car parking companies, how easy it is for users to find information about daily rates, car park locations etc. but particularly if a mobile application was available that allowed users to book parking. Conducting research also led to the discovery finding out that no mobile application was available for private car parking companies and this gave me the idea of developing such app.

After finalizing the concept idea, a meeting was arranged by me with the Managing Director of Bestcarparks Ireland. Bestcarparks is a car parking company established and located in Ireland. It is the largest car park company in Ireland with over 5,000 car park spaces in Ireland in more than 13 parking sites.

After fully explaining the idea to the managing director he was more than happy to get on board. After the meeting, my idea was presented to a team of judges in the college who accepted the idea and gave me permission to move into the development stage and provided me with all the content they had available.

Aims

The aim of the project is to build an application that allows users to book car parking from their iOS devices designed for Bestcarparks Ireland securely. The application shall have multiple linked interfaces based of a HTML5 website created by me using Web Views this will allow customers to login, register, book and pay for their car parking but also receive information about the prices of carparks, parking types available and live spaces and directions to any car park owned by the company. The application will allow users to register an account and login at any time requested by the user. Strong user passwords and sessions will protect the users activity and accounts, users will not be allowed have multiple accounts and this will be checked by the email address provided at registration. The application will allow users to pay their parking from their mobile device, users can view their booking history and have the flexibility of editing valid bookings not past the booking date. Users can add, edit and delete vehicles registered on the account at any time. The application and website will be security tested before release to ensure all the content is secure before release. The app shall be available to download from the Apple Store in early June 2017.

Technologies

The mobile application will be built for iOS using the swift language. Swift is a general purpose, multi – paradigm compiled language developed by Apple. Swift is used to develop iOS, macOS, watchOS, tvOS and Linux based applications. Web Views in swift allow displaying web pages from a local host or remote server and I will be using web views to display the relevant content for the application. I will be using HTML5, Bootstrap, PHP and JavaScript to create the relevant content that will be displayed on the application. An InnoDB will used to store all of the information and will be the application database, this can be accessed using PhPMyAdmin and SQL queries will be created to display the relevant information when required. XCode 8.3 will be used to develop the application and the in-app simulator will allow the simulation of the application to test new features as they are implemented. My personal iPhone will help with the testing of the application. XAMPP 7.1.1-0 will allow the creation of a virtual web server on my machine where the web files and database files will be stored for the development of the application, testing and presentation, later they will be moved to the Bestcarparks servers.

Structure

Section 1 – Introduction, will explain the background and aims of this project while the technologies section will give a brief explanation of the programming language and devices used to build the application. While structure explains the structure of this document.

Section 2 – System, this section describes the requirements and features implemented in the application. Use case diagrams accompany all the features in this project. The non-technical requirements are described in the non-technical requirements section of the document. The architecture of the application is represented in a class diagram. The graphical user interface mock-ups are used to represent the applications features and how a user will be presented with these features. The testing section will explain the testing methods that will be used to test the application and give results to ensure that the application is programmed correctly without bugs and errors to give the user the best experience possible.

Section 3 – Conclusion, this section gives an overall conclusion on this document and the application.

Section 4 – Further Development, describes the future opportunities for the application in the near future and any upgrades that can be added to make the application beat any competition in the future if any.

Section 5 – References, information that was used to complete this document, the application.

Section 6 – Appendix, this section contains a copy of the project proposal, the monthly journals and supervisor meetings.

2 System

2.1 Requirements

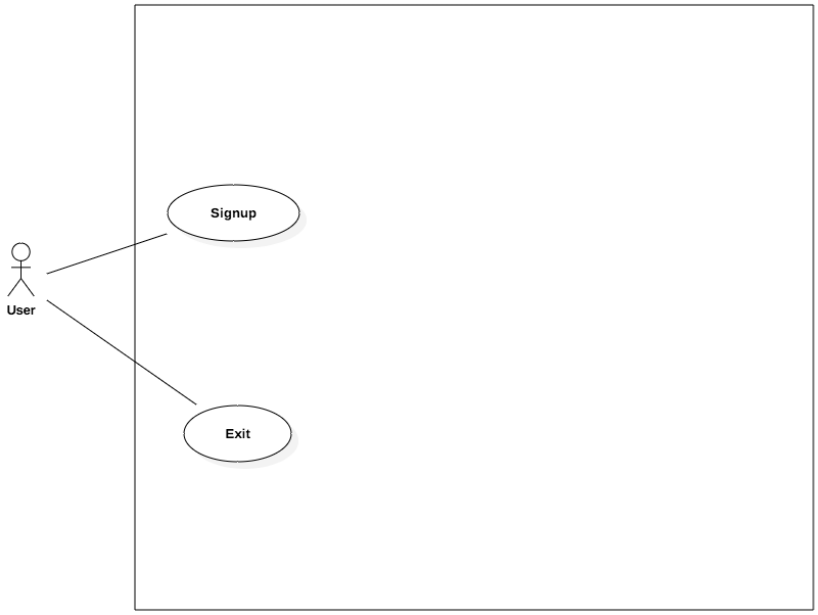
2.1.1 Functional Requirements

Requirement 1 < Signup >

Figure 1: This use case shows the interaction between the users, the application which will process the signup request of a user. A MySQL InnoDB database will store all the user information. The priority of this use case is high as the user needs to an account to use the application services.

Use Case

Figure 1 Signup Use Case



Detailed Use Case Description

(Figure 1)

Name: Signup Use Case

Description: This use case describes the signup process

Flow of Events:

Activation: Use case starts when the user presses the 'Signup' button

Basic Flow:

1. The use case begins when the user presses the 'Signup' button.
2. User enters his details in the fields provided by the application. (First Name, Last Name, Email, Address, Password, Address, Town, County, Mobile Number)
3. User presses 'Sign up' button.
4. The application validates the fields.
5. In the case the user details are valid:
5.1 The application will send an acknowledgement for the signup process being successful and provide a link to the login page.
5.2 The user's registration details will be stored in the database.
In the case the user details are not valid:
5.3 The application will ask user to recheck the fields for errors or
5.4 The user's details will not be saved in the database and the user will not have an account to use.

Alternative Flow:

1. Invalid Email
2. Email already registered
3. Password does not meet the required characteristics
4. Invalid email address format

Termination: User is presented with a signup successful notification message and a link to login.

Special Requirements: None

Preconditions: For a signup to be successful an internet connection is needed to connect to the database.

Post-conditions:

Success Conditions:

1. The user creates an account
2. The user has access to the application services
3. The user can login into the application
4. The user details are stored in the database

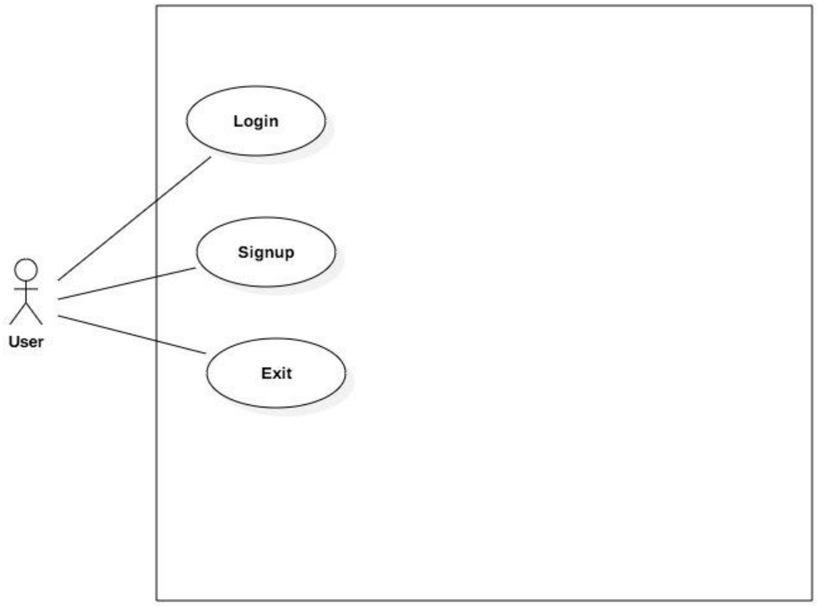
Failure conditions:

1. The user has no access to the application services
2. The user cannot login into the application
3. The user details are not stored in the database

Requirement 2 < Login>

Figure 2: This use case shows the interaction between the user which will try to login with the details registered to use the application services, the application will validate the login request. The application will retrieve the login details and match the entered password with the saved password. The priority of this use case is high as the user needs to login to use the application services.

Figure 2 Login Use Case



Detailed Use Case Description

(Figure 2)

Name: Login Use Case

Description: This use case describes the Login process

Flow of Events:

Activation: Use case starts with the user pressing the ‘Login’ button

Basic Flow:

1. The use case begins when the user presses the ‘Login’ button.
2. User enters his details in the fields provided by the application. (Email and Password)
3. User presses ‘Login’ button.
4. The application validates the fields.
5. The user is then logged in.

5.1 In the case the user details are valid:
The application will send an acknowledgement for the login process being successful.
The application will redirect the user to the ‘My Details Interface’

5.2 In the case the user details are not valid: The application will ask user to recheck the ‘Email/ Password fields for error.

6. The application will redirect the user to a new interface after a successful login.

Alternative Flow:

1. Invalid Email
2. Password entered is not correct

Termination:

User is presented with a new interface.

Special Requirements: None

Preconditions: For a login to be successful an internet connection is needed as the application will retrieve the user login details from the database.

Post-conditions:

Success Conditions:

1. The user enters email and password used when registering
2. The user has access to the application services
3. The user will be logged into the application
4. The user details are retrieved from the application database

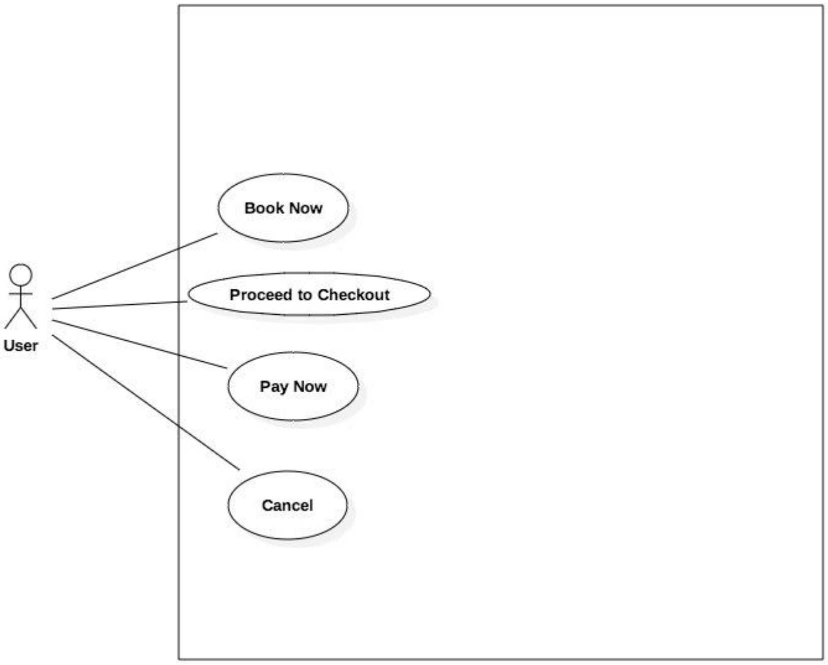
Failure conditions:

1. The user has no access to the application services
2. The user cannot login into the application

Requirement 3 < Book & Pay Now >

Figure 3: This use case shows the interaction between the user and the application when making a booking. A payment will be processed. The priority of this use case is high as this is the most important feature of the application and the scope of the application is that users can book parking on the go.

Figure 3 Book & Pay Now Use Case



Detailed Use Case Description

(Figure 3)

Name: Book & Pay Now Use Case

Description: This use case describes the booking process

Flow of Events:

Activation: Use case starts with the user pressing the ‘Book Now’ button

Basic Flow:

1. The use case begins when the user presses the ‘Book Now’ button.
2. User chooses the booking details to meet their requirements using the application graphical interface. (Location, Date, Spaces Required, Duration)
3. The application shows the total cost of the booking.
4. User presses ‘Book Now’ button.
5. The application validates the fields and presents the next stage.
6. The user is asked to confirm their booking details and the user chooses a vehicle stored on their profile, the user is asked to accept the company T&C’s.
7. User presses ‘Proceed to Checkout’ button to move to next stage.
8. The application validates the fields and presents the next stage.
9. User is presented with the Stripe interface and there payment gets processed.
10. User presses ‘Pay Now’ button.
11. The application validates the fields.
12. The booking is processed.

13.1 In the case the details are valid:

The application will show a confirmation screen showing the booking is successful.

The booking will appear in the My Bookings section immediately.

10.2 In the case the details are not valid:

The application will ask user to recheck the booking fields

The application will ask user to recheck the payment details entered

Alternative Flow:

1. Invalid Date
2. No Location was chosen from the list provided
3. The car park is full or closed
4. Product (Duration) required is invalid
5. User didn’t chose a Vehicle registration number from their list
6. Stripe Error
7. No Vehicles saved

Termination: User is presented with a confirmation message

Special Requirements: None

Preconditions: For a booking to be successful an internet connection is needed to connect to the application booking services.

Post-conditions:

Success Conditions:

1. The user is presented with a booking successful message
2. Application processes payment and stores, it
3. The user will have a valid booking.

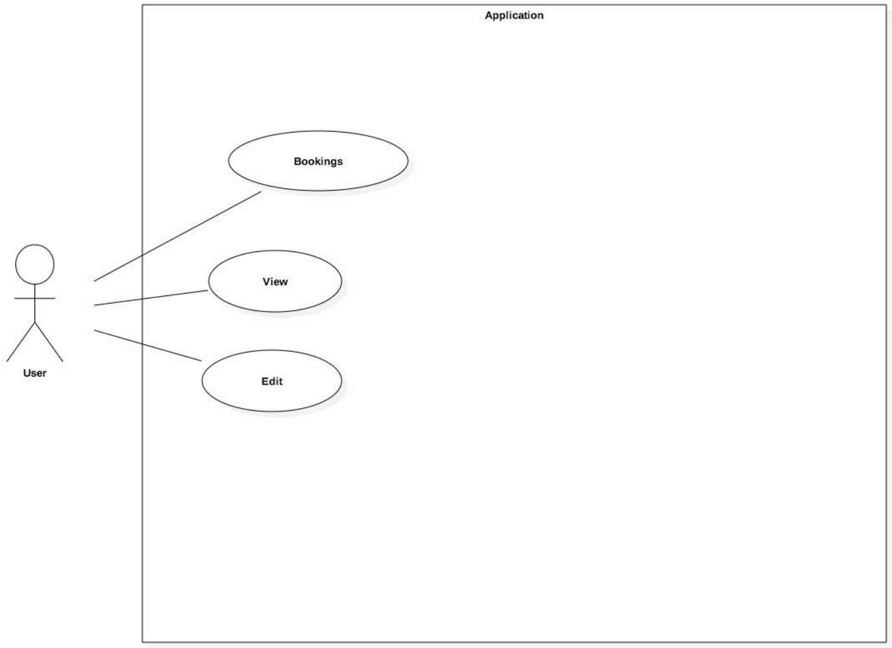
Failure conditions:

1. The user is not redirected to the payment interface
2. The booking is not successful
3. The booking is not saved in the database
4. The application will not process a payment

Requirement 4 < My Bookings>

Figure 4: This use case shows the interaction between the user and the application.

Figure 4 **My Bookings Use Case**



Detailed Use Case Description
(Figure 4)

Name: My Bookings Use Case

Description: This use case describes the My Bookings section.

Flow of Events:

Activation: Use case starts with the user pressing the “My Bookings” link in the navigation bar after they are logged in.

Basic Flow:

1. Use case starts with the user pressing the “My Bookings” link in the navigation bar.
2. User gets provided with an interface that provides all their current and previous bookings.
3. If the user requires to edit a booking, they press the ‘edit booking’ button provided on a non-expired booking and an interface will provide a user to modify their parking.
4. The bookings are sorted by current and expired bookings and the booking is available to edit until the booking date.
5. ***In the case the edit button is pressed:***

5.1 The application will retrieve all the details regarding the booking and provide the edit interface.

5.2 The user will have the option of modifying the Car Park, Reservation Date and VRN. The user will then press the ‘change booking’ button, after pressing this button the booking will be processed and changed.

5.3 The booking is then modified in the database and saved.

In the case the edit is not valid:

5.4 The application will tell why the booking cannot be edited.

5.5 The booking shall not be modified.

Alternative flow:

1. No current bookings
2. No expired bookings
3. All bookings are past the booking date and cannot be edited

Termination: User navigates to a different page or closes application.

Special Requirements: None

Preconditions: For a user to have access to my bookings or to edit a booking there must be a valid internet connection.

Post-conditions:

Success Conditions:

1. The user views his current and expired bookings.
2. The user is displayed with a view of his parking history
3. The user has the option to edit an existing booking.

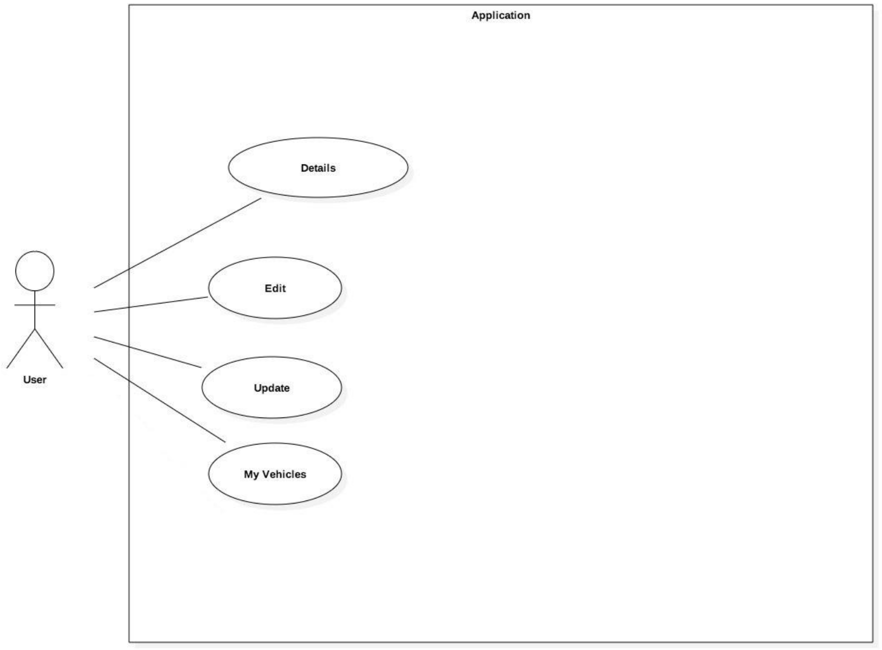
Failure conditions:

1. The user cannot view his current and expired bookings.
2. The user is not displayed with a view of his parking history
3. The user is not able to edit an existing booking.

Requirement 5 < My Account>

Figure 5: This use case shows the interaction between the user and the application.

Figure 5 **My Account Details Use Case**



Detailed Use Case Description
(Figure 5)

Name: My Account Use Case

Description: This use case describes the My Account section.

Flow of Events:

Activation: Use case starts with the user pressing the “My Account” link in the navigation bar after they are logged in.

Basic Flow:

1. Use case starts with the user pressing the “My Account” link in the navigation bar.
2. User gets provided with an interface that provides all their personal details.
3. User gets provided with a button to view their saved vehicles.
4. If the user requires to edit their personal details, they press the ‘edit details’ button provided and an interface will provide a user to modify their personal details.
5. The new details are then saved once the users presses the ‘update my details’ button. A status message will be provided once the user is redirected with a confirmation or error message.
6. ***In the case the edit button is pressed:***

6.1 The application will retrieve all the details regarding the user that are saved in the database and provide the edit interface.

6.2 The user will then have the option to update the old details or go back to the previous page.

6.3 The details are then modified in the database and saved.

In the case the edit is not valid:

6.4 The application will tell why the personal details cannot be edited.

6.5 The personal details shall not be modified.

Alternative flow:

1. No details modified
2. No option to edit details

Termination: User navigates to a different page or closes application.

Special Requirements: None

Preconditions: For a user to have access to their personal details and for a user to be able to modify these details there must be a valid internet connection.

Post-conditions:

Success Conditions:

1. The user can view their personal details
2. The user can edit their personal details information
3. The user’s information has been update and the new information is provided.

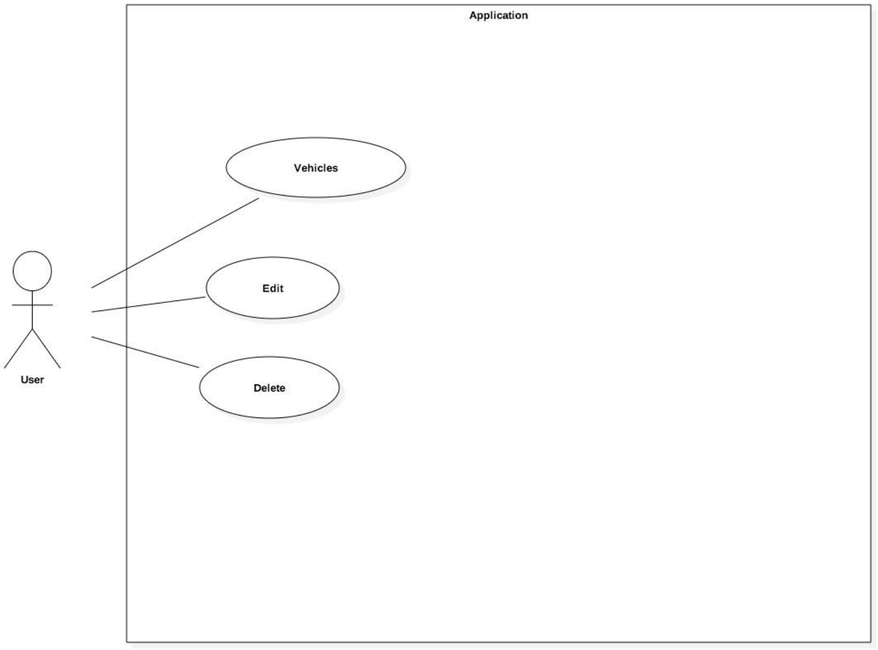
Failure conditions:

1. The user cannot view their personal details.
2. The user cannot edit their personal details.
3. No information has been updated.

Requirement 6 < My Vehicles>

Figure 6: This use case shows the interaction between the user and the application.

Figure 6 **My Vehicles Use Case**



Detailed Use Case Description
(Figure 6)

Name: My Vehicles Use Case

Description: This use case describes the My Vehicles section.

Flow of Events:

Activation: Use case starts with the user presses the My Vehicles button under their personal details.

Basic Flow:

1. Use case starts with the user presses the My Vehicles button under their personal details.
2. User is provided with an interface displaying all their saved vehicles. Details about the car such as VRN, Make and Color will be displayed.
3. User is provided with 2 buttons underneath the details. One button called Edit Car and another one called Delete Car. The user is also provided with a New Vehicle button which allows the user to add a new vehicle.
4. If the user requires to edit one of their saved vehicle details, they press the 'edit details' button provided under the details of the vehicle they wish to edit. The user then can edit the car VRN, Make and Color, he can then update these details.
5. If the user wishes to delete a saved vehicle they press the delete car button situated under the details of the desired vehicle. This will then delete the car from the database.
6. If the user wishes to add a new vehicle to their account, they can but only after entering all the details required.
7. A status message will be provided once the user is redirected with a confirmation or error message.

In the case the add, edit, delete are not valid:

7.1 No new vehicle will be added.
7.2 The chosen vehicle will not be edited or deleted.

Alternative flow:

1. No button available to add a vehicle
2. No option to edit or delete car

Termination: User navigates to a different page or closes application.

Special Requirements: None

Preconditions: For a user to have access to their vehicles and for a user to be able to modify these details there must be a valid internet connection.

Post-conditions:

Success Conditions:

1. The user can view their vehicles saved and details of each.
2. The user can add a new vehicle
3. The user can edit vehicle information
4. The user can delete vehicle information
5. The vehicle's information has been updated and the new information is provided.
6.

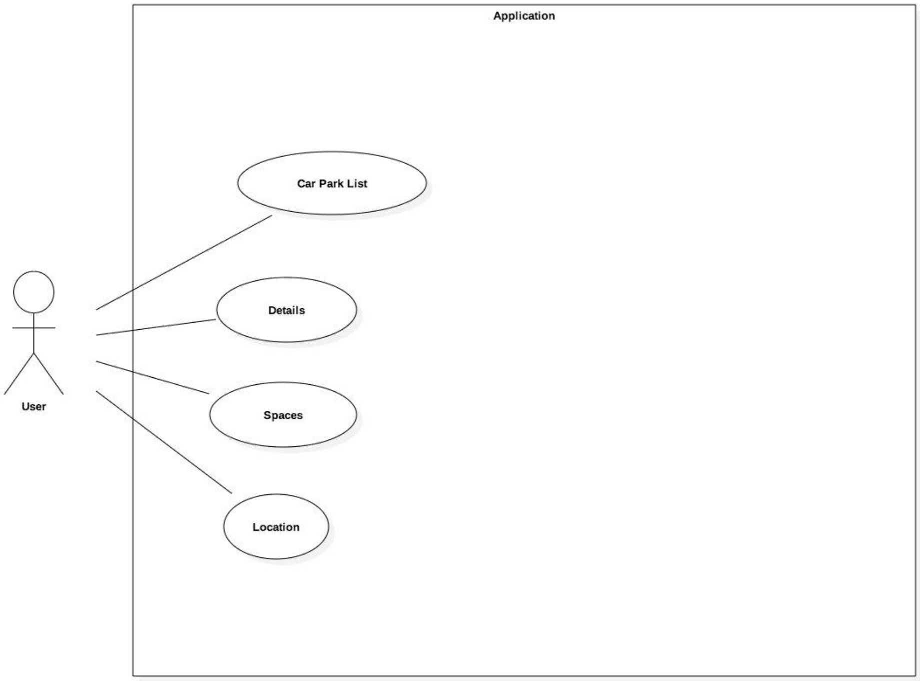
Failure conditions:

1. The user cannot view their vehicles saved.
2. The user cannot add a new vehicle
3. The user cannot edit the vehicle chosen.
4. The user cannot delete the vehicle chosen.
5. No information has been updated.

Requirement 7 < Find A Car Park >

The user will be presented with a list showing the car parks available, the user can click on the car parks to receive information about each car park, live spaces and a map of where the car park is located.

Figure 7 **Find A Car Park Use Case**



Detailed Use Case Description

(Figure 7)

Name: Find A Car Park Use Case

Description: This use case describes the Find A Car Park functionality of the application.

Flow of Events:

Activation: Use case starts when the user presses the 'Find A Car Park link from the navigation bar.

Basic Flow:

The use case begins when the user is presented with the interface and a list of car parks.

1. User is presented with an interface that gives a list of car parks.
2. The user clicks on a car park, information about the car park prices, opening hours and live spaces is provided along with a map of the location of the car park.
3. Opening the location in Google Maps on the iPhone will allow a user to get directions to the car park.
4. User is also provided with a Book Now button which allows a user to make a booking in the selected car park, if the user is not logged in he will not be able to make a booking and will be redirected to the login page.

Termination: User changes view, user presses book now, user closes app.

Special Requirements: Internet Connection Required.

Preconditions: Valid internet connection is needed to display the map and to connect to Googles servers.

Post-conditions:

Success Conditions:

1. The user is presented with a list of car parks.
2. The user is presented with info about a car park and a map showing the location of the app.
3. The user can get directions to the car park.

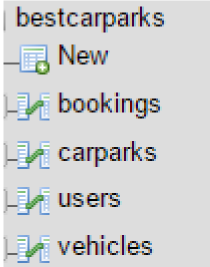
Failure conditions:

1. No car parks shown on the list.
2. No internet connection to display map

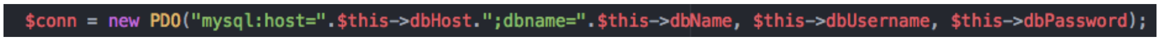
2.1.2 Data Requirements

In this section I will explain and describe the data requirements of the application which are essential to the applications features mentioned above.

Database server: The server type is MariaDB running version 10.1.21 which is a source distribution version. The database is running on protocol version 10 and the server charset is UTF-8. The database uses a mysqli extension which is a relational database driver used in the PHP programming language to provide an interface with MySQL databases. I have used this extension in my project along with the PHP Data Objects (PDO) API to retrieve, store and delete data from the database using the application. Using the PDO API I was able to use named parameters, object mapping, prepared statements and stored procedures in a fast performance environment. All of the customers details, bookings etc. will be stored on this database. The database consists of 4 main tables (bookings, carparks, users and vehicles) which are all stored in one database named Bestcarparks.



bestcarparks
- New
- bookings
- carparks
- users
- vehicles



```
$conn = new PDO("mysql:host=".$this->dbHost.";dbname=".$this->dbName, $this->dbUsername, $this->dbPassword);
```

2.1.3 Non-Functional Requirements

Performance/Response time requirement

Performance will be a major aspect in the overall application. The application should be capable of being opened on older iOS firmware versions and older spec mobile devices without any response time issues. As the application, will be connected hosted on a remote server and connected to a backend database factors like network speeds and issues need to be considered when a user tires to book car park spaces or wants to access their account to view their details, booking history, saved vehicles etc. Unhandled network errors could result in a user not being able to use the application booking services. Therefore, the application should be able to deal with such issues. A loading indicator is also implemented in the applications web view to show a user if the page they require has fully loaded or not. This shall give a user an indication of how slow/ fast the performance of the application is and if they are connected to a network with slow network speeds.

Availability requirement

The application must be available constantly as the user will use the application services to book and pay for their required parking needs. Once a user has downloaded and installed the application from the Apple App Store the application should be fully functional and accessible to the user. The application should be free from downtime and if any bugs or errors appear they must immediately be repaired or removed to insure the application services remain up and running to the users.

Operating System Requirements

As the application, will be developed for iOS devices the user must have an Apple iPhone to use the application services. If the user doesn’t an Apple iPhone unfortunately he won’t be able to use the application services.

Security requirement

The application is using a Web View that pulls a PHP file directory that has all the applications content and files. An Inno DB is being used to store the information. The application files must be secure to avoid any attacks or information leakage. The application has been designed to only accept passwords that contain the strong password criteria, sessions have been implemented to restrict access on the booking features until a user has logged in. The application is security tested to discover what threats the application is vulnerable to if any.

Reliability requirement

In many systems reliability is a big consideration during the development stage. If a system keeps crashing or has a lot of software bugs this will affect the overall reliability of the application and affect the user's use of the application services. I will take a range of measures into account ensuring that all software bugs have be eliminated through varies testing methods such as verification, validation, integration testing, functional testing, system testing and UAT Testing. Logical errors will be removed where possible. The system should be able to cope with minor issues that may arise because of internal factors therefore making it reliable.

Maintainability requirement

Software developers regularly release varies updates and bug patches. I will plan to write the code in a way where it's easily readable with or without comments, this means that I will be reminded what each piece of code has the functionality for hen re-editing the code. Another factor which needs to be considered will be the systems structure. I will develop the system in different modules. Each module will provide certain functionality such as performing transactions with the remote database. The modular design and structure will enable updates and patches to be released quicker and be implemented in easily compared to systems which have no definitive structure. I will also use agile development which allows certain structures to be put in place that will allow various changes to be made during or after the implementation stage of development.

Reusability requirement

The reuse of code will be a big factor when minimising development time. Certain methods can be reused repeatedly once a decided bug free implementation idea has been finalised. This will speed up development time and provide time to ensure the application is secure and free of any vulnerabilities.

2.1.4 User Requirements

In this section I will explain the user requirements. If a user wants to use this application, he must meet the following requirements:

- **iPhone:** The user must possess an iPhone on which to run the application
- **iOS10:** The user should preferably have iOS 10 installed, but the application is not limited to iOS 10 and will be compatible with older firmware versions.
- **iTunes App Store:** The user should have the App Store installed on their device, to download the application.
- **Internet Access:** The user will need to have internet access to use the application and download it.

2.1.5 Environmental Requirements

In this section I will explain the environmental requirements that are essential when developing the application.

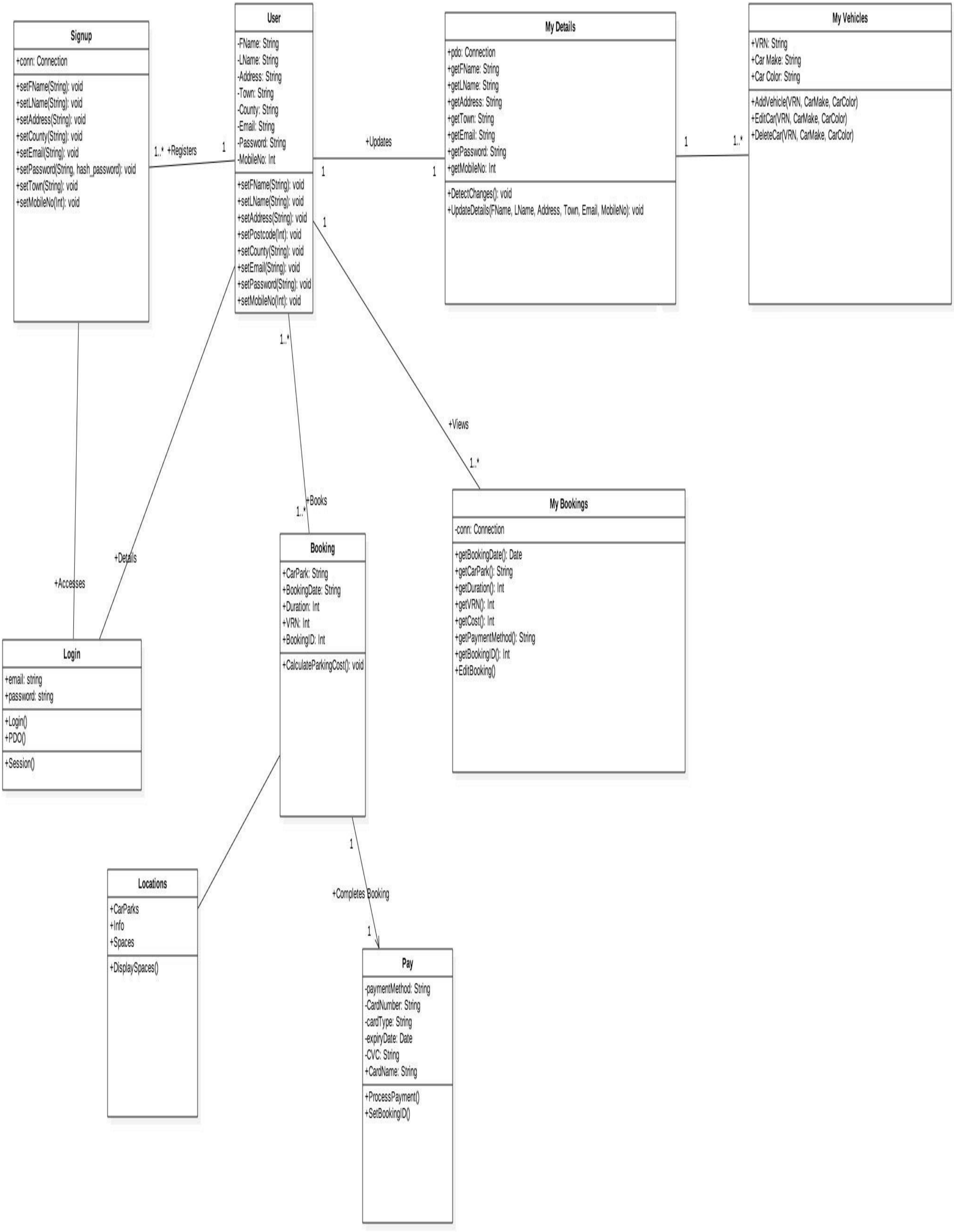
- **MacBook:** A MacBook is required to run XCode.
- **iPhone 7 & 7 Plus:** Needed to run the application and perform testing.
- **XCode:** The application is developed on Swift and this is the default environment provided by Apple.
- **Internet Access:** This is required to access Apple documentation and other various resources in the building process of the application and code.
- **Apple Development Certificate:** To run the application on your phone an Apple developer certificate is required.

2.1.6 Usability Requirements

For a user to use the applications booking services he must have an account. It is assumed that the user knows how to register and how to complete forms asking for personal details and most importantly how to choose a secure password.

2.2 Design and Architecture

The application is built using the Swift iOS language. The application consist of a web view that displays a web site that was created by myself using HTML5, Bootstrap, CSS, JavaScript, JQuery and PHP. A MySQL database is used to store information and prepared statements are used to display the relevant information on the correct pages. Bootstrap allows a web site to be mobile responsive, this allowed me to create a website that would turn into an application programmed correctly. An activity indicator is displayed using the Swift language to show when the page has fully loaded. This is a spinner which will spin in the middle of the app until the page has loaded then it will disappear this feature will appear on all pages. The CSS was designed according to Bestcarparks needs and using their existing content that I have partially created before entering my 4th Year of college when I gained experience in the web development area. The CSS offers a fluid transitions when a user goes through multiple pages on the site. A user session is created when the user logs in into their account and is stored until the app is closed. Stripe payments system is incorporated to accept payments, test card functionality is only working at the moment.


Signup
+conn: Connection
+setFName(String): void
+setLName(String): void
+setAddress(String): void
+setCounty(String): void
+setEmail(String): void
+setPassword(String, hash_password): void
+setTown(String): void
+setMobileNo(Int): void
User
-FName: String
-LName: String
-Address: String
-Town: String
-County: String
-Email: String
-Password: String
-MobileNo: Int
+setFName(String): void
+setLName(String): void
+setAddress(String): void
+setPostcode(Int): void
+setCounty(String): void
+setEmail(String): void
+setPassword(String): void
+setMobileNo(Int): void
My Details
+pdo: Connection
+getFName: String
+getLName: String
+getAddress: String
+getTown: String
+getEmail: String
+getPassword: String
+getMobileNo: Int
+DetectChanges(): void
+UpdateDetails(FName, LName, Address, Town, Email, MobileNo): void
My Vehicles
+VRN: String
+Car Make: String
+Car Color: String
+AddVehicle(VRN, CarMake, CarColor)
+EditCar(VRN, CarMake, CarColor)
+DeleteCar(VRN, CarMake, CarColor)
1..*
+Registers
1
+Updates
1
1
1
1..*
1
1..*
+Views
1..*
+Books
1..*
+Details
+Accesses
Login
+email: string
+password: string
+Login()
+PDO()
+Session()
Booking
+CarPark: String
+BookingDate: String
+Duration: Int
+VRN: Int
+BookingID: Int
+CalculateParkingCost(): void
My Bookings
-conn: Connection
+getBookingDate(): Date
+getCarPark(): String
+getDuration(): Int
+getVRN(): Int
+getCost(): Int
+getPaymentMethod(): String
+getBookingID(): Int
+EditBooking()
1
+Completes Booking
1
Locations
+CarParks
+Info
+Spaces
+DisplaySpaces()
Pay
-paymentMethod: String
-CardNumber: String
-cardType: String
-expiryDate: Date
-CVC: String
+CardName: String
+ProcessPayment()
+SetBookingID()

Class Diagram

Signup Class

Before a user can gain access to the application booking services they must first create an account. The register user class is responsible for gathering this information from the user class. In conjunction with the user class and the database the information entered is stored.

User Class

The user class is used to gather the information required to login to the Booking class. Each variable in the user class is set to a String with the appropriate setters and getters. This information is stored in the database and is retrieved each time the user logs into the application booking services.

Locations Class

The GoogleMaps car park locations are stored in this class, real time data regarding car spaces is stored in this class.

Login Class

The login class has direct contact with the user class and the database. When the user enters their email address and password the login class will check the database and if the correct password has been entered the user can access the application booking services.

Booking Class

This class allows users to use the booking services of the application. Users book their desired parking through this class. This class is associated with the Pay class.

Pay Class

This class allows users to pay for their desired parking. This class is directly associated with the Booking class as both classes need each other to function. This relates to the database as when a booking is successful, it is stored in the database. It also communicates with Stripe payments to authorize a valid card.

My Bookings

This class allows users to view their previous bookings and edit current bookings. This class relates to the database as the user’s bookings are retrieved from the database.

My Details

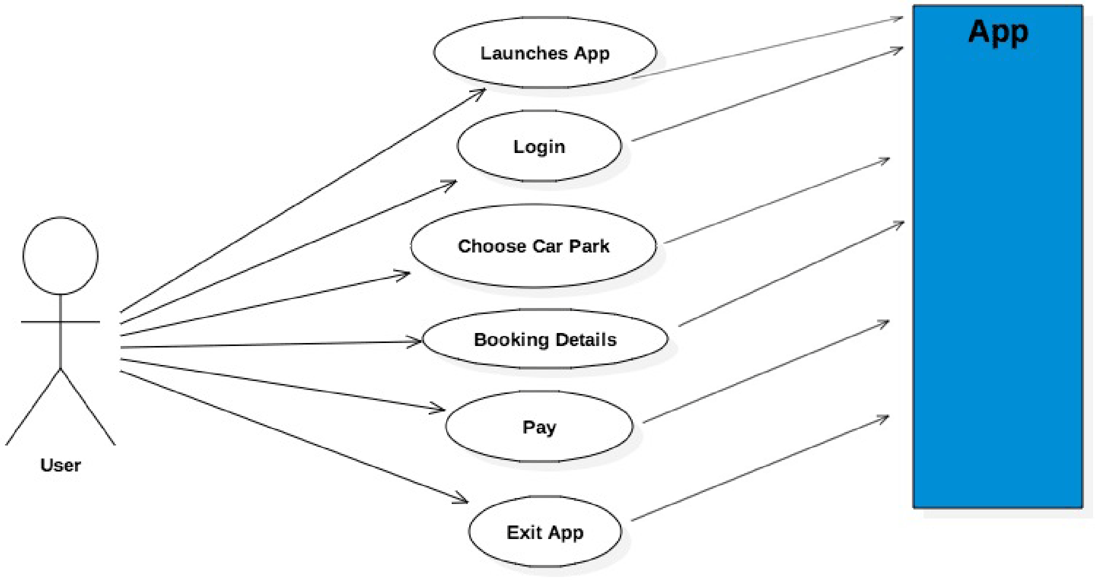
This class allows a user to view their personal details and change any of these details when they desire. This class works with the database to retrieve the current user details and store new user details. This class is associated with the User class as the user enters these details.

My Vehicles

This class allows a user to add new vehicles, view their stored vehicles and edit any of these details when they desire but also delete the vehicle details. This class works with the database to retrieve the vehicles details and store new details.

Use Case Diagram

The class diagram shows the process a registered user would take to book a car park that has spaces available on the application.



2.3 Implementation

The purpose of this section is to describe the technologies used in the implementation of the iPhone application.

2.3.1 Technology Overview

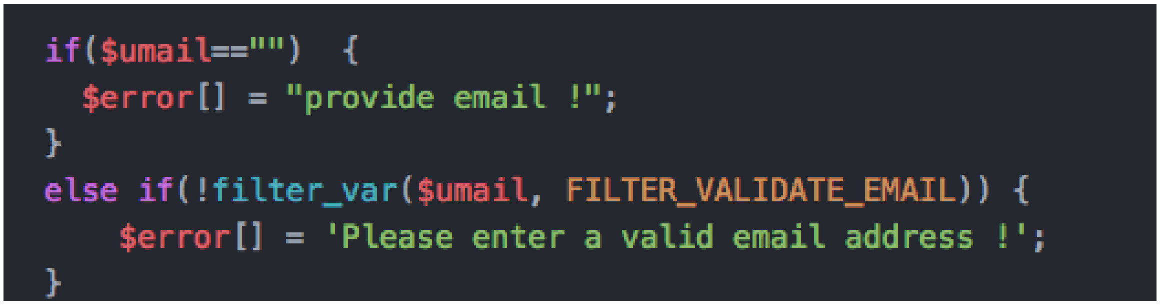
The approach to building an iPhone application that was taken at the start of the development was the native app development route but this quickly changed once I became more familiar with the swift capabilities. The app was created using Apple's swift language and developed through their building environment which is XCode. After learning the limitation of the swift language, I had to build a mobile responsive website that was going to be used for the application. The following technologies and frameworks allowed me to develop the website for the application (HTML5, Bootstrap, PHP, SQL, CSS and JavaScript). I then used Stripe payments to process payments.

2.3.2 Security

Security is very important to the application and had to be implemented everywhere possible. The following security features were implemented:

2.3.2.1 Server Side Validation

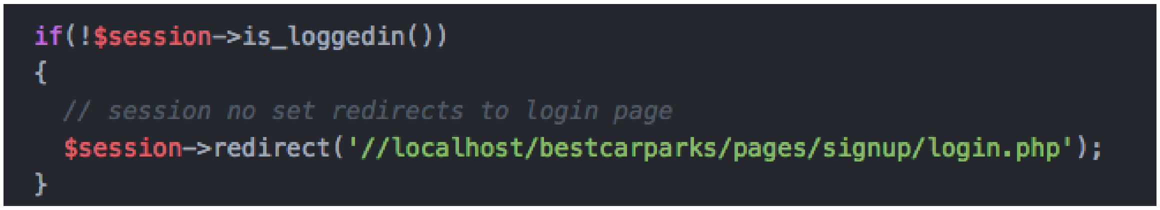
Server side validation occurs the input submitted by the user is being sent to the server and validated using PHP on the server itself. After the validation, has occurred the feedback is presented to the user in an error message format etc. Using server side validation can protect against client side scripting and dangerous inputs to the server. Server side validation was implemented throughout my whole source code any time an input was being sent to the server. An example of server side validation can be found in the registration form below.



```
if($umail=="")  {
  $error[] = "provide email !";
}
else if(!filter_var($umail, FILTER_VALIDATE_EMAIL)) {
    $error[] = 'Please enter a valid email address !';
}
```

2.3.2.2 Sessions

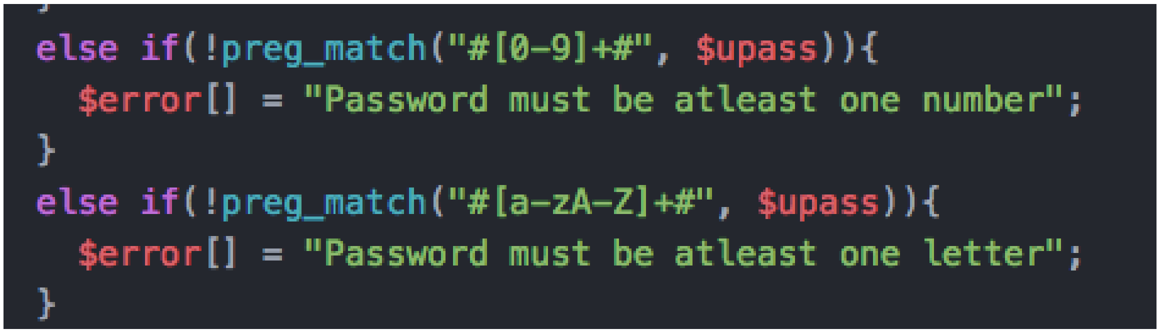
In PHP 5 sessions create a method of storing information in variables that can be used across multiple pages, unlike a cookie the information is not stored on the user’s computer or application. It helps protect pages that you don’t want a user to access until the user has logged in or registered which helped protect the booking pages of the application. Session variables hold information about the user logged in and all the information is available across all pages of the application. Once the user closes the application the session is destroyed and the variable is reset to its normal state not storing any information until the user logs in, this helps protect a user’s account if the user has lost its phone or if someone is using their phone without their permission. If the user tries to access the booking pages he will be redirected to the login page and then be granted access upon valid login. The session used in my code can be seen below.



```php
if(!$session->is_loggedin())
{
  // session no set redirects to login page
  $session->redirect('//localhost/bestcarparks/pages/signup/login.php');
}
```

2.3.2.3 Strong User Passwords

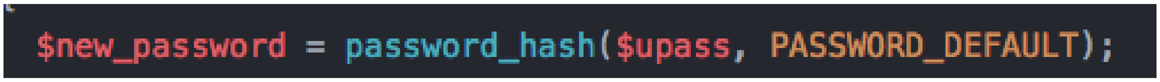
A strong password helps protect a user account against brute force hacking. Having a strong password takes the time of cracking a password to a maximum of 100 million years. A strong password must include a mixture of numbers, symbols, capital letters and lower case letters. I have implemented this feature in my project to help protect users accounts from being brute force or compromised easily. Strong passwords a very rarely guessed due to their complexity and they are a must in every project developed. An example of a part of my implementation of this feature can be seen below.



```php
else if(!preg_match("#[0-9]+#", $upass)){
  $error[] = "Password must be atleast one number";
}
else if(!preg_match("#[a-zA-Z]+#", $upass)){
  $error[] = "Password must be atleast one letter";
}
```

2.3.2.4 Password Hashing

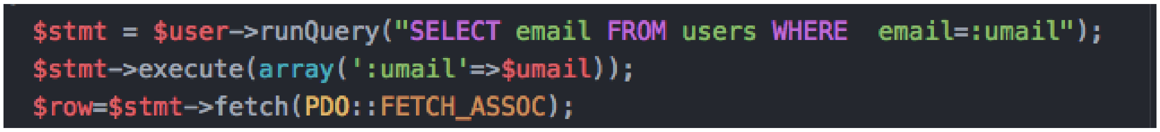
Implementing the password_hash in PHP 5 helps create a new password hash using a strong one-way hashing algorithm that cannot be decrypted and the value is also encrypted in the database. This is very important as the user passwords must be encrypted in the database in case of a database compromising. The passwords are encrypted via the crypt() hashing method and are very hard to decrypt. I have implemented this method when a user registers a new account and therefore all passwords are encrypted, this can be seen below.



```
$new_password = password_hash($upass, PASSWORD_DEFAULT);
```

2.3.2.5 Email registered

A method was created to check if the email address submitted at registration is already registered with another user or with the user trying to register. This method helps protect from users trying to register 2 or more accounts with the same email, it also offers data integrity and it is a security feature that helps avoid multiple accounts that may cause the database problems. The method implemented in my application can be found below.



```
$stmt = $user->runQuery("SELECT email FROM users WHERE  email=:umail");
$stmt->execute(array(':umail'=>$umail));
$row=$stmt->fetch(PDO::FETCH_ASSOC);
```

2.3.3 Technologies

2.3.3.1 XCode

XCode is the integrated development environment for MacOS that contains a suite of software development tools developed by Apple that allows the software development for macOS, iOS, tvOS and watchOS. XCode was first released in 2003 and started on version 1.0 and its now on 8.0. XCode supports Swift, Objective-C, Ruby and many more and has many supporting frameworks that can be added such as Facebook SDK, Twitter SDK and Google SDK adding plenty of new development features. At the very start of the development phase I was using XCode version 8.1 and I finished the development stage on Version 8.3.2, every version of XCode is updated when a major new update of iOS, MacOS software or firmware update is released. XCode offers in app simulators which allow you to test your application on many different firmware's and iPhone models. This was very beneficial to my development phase and testing phase as I could get a clear understanding if my application was compatible with all software and iPhone models. After performing testing which can be found in this report I found that there were certain code modifications I needed to make the application compatible with all iPhones, this helped me ensure that the application is working as expected on any iPhone and it was a very beneficial factor to have rather than having to borrow all versions of iPhones.

2.3.3.2 Swift

Per Apple's developer website and build notes swift is the new programming language for all of Apples devices that builds on Objective-C. Swift offers safe programming patterns and adds modern features to make programming easier and more flexible, with the help of the much-loved Cocoa and Cocoa Touch frameworks it is an opportunity to reimagine how software development looks. With the help of Swift I implemented an Activity Indicator that would spin until the page on the Web View had fully loaded.

2.3.3.3 HTML5, Bootstrap, JavaScript

HTML5 is a markup language that is used for presenting and structuring content from the World Wide Web. I used the HTML5 language in the development of the website to display the content that was required to make the website as professional as possible and to the customer's standards. With the help of HTML5 I could display divs, images, tables etc.

Bootstrap is a framework that easily and efficiently scales your websites and applications using a single code base, bootstrap scales everything from phones to tablets to desktops with its CSS media queries. With the help of Bootstrap, I created a website that adapted perfectly into my Web View and transformed into an application style website just what I needed. With the help of Bootstrap, I was also able to get an extensive and beautiful documentation for common HTML elements, dozens of custom HTML and CSS components and awesome JQuery plugins that helped me beautify the site and button links and many more features.

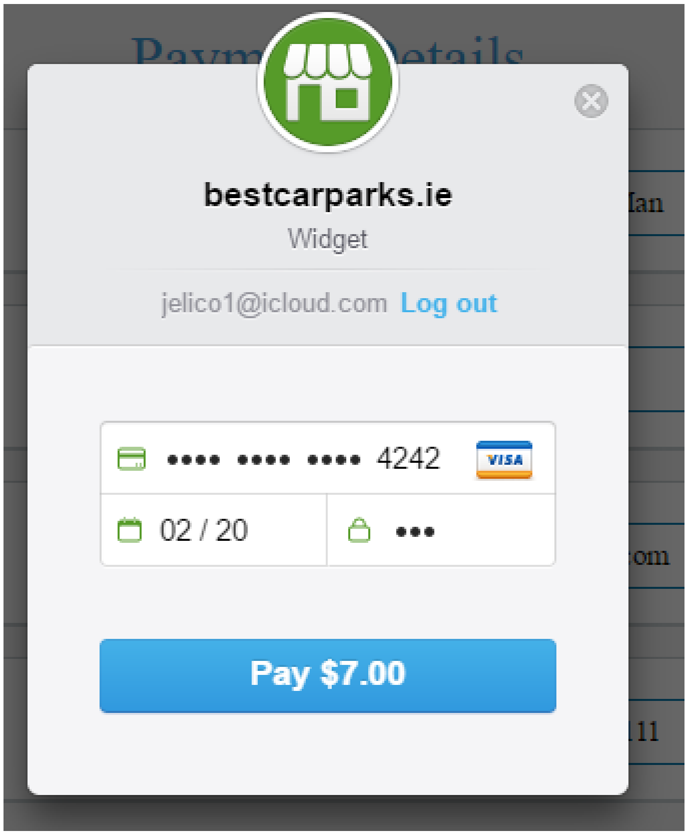
JavaScript is a dynamic high-level programming language. JavaScript is prototype-based with first-class functions making it a multi-paradigm language that supports object-orientated programming. JavaScript helped me implement price calculations on car parking bookings making them dynamic not requiring any page changes and with the use of HTML5 these figures could then be spanned to not allow the user to change the price of the booking. JavaScript also allowed me to implement a feature that would not allow the user to go back or refresh the page while they are on the final step of the booking process this was beneficial and was also a security flaw that I could cover.

2.3.3.4 PHP

PHP is a scripting language that is especially suited to server-side web development. PHP runs on a web server and any code requests are executed by the PHP runtime. With the help of PHP, I could connect to a SQL database that was stored on a server using the mysqli connection PHP offers due to this connection being made I could display the user's information that was stored in the database when the user required it. Information such as personal details, vehicles registered on the account and booking history was all displayed using PHP. User sessions were added thanks to PHP offering security by restricting certain pages to public users until they were logged in, this ensured site security and it also ensured that users could not make a booking until they registered or logged in. Prepared Statements were implemented using the PDO API framework which enabled to make features like the edit booking, change personal details etc. PDO also allowed the implementation of CRUD functionality in the website which was crucial to completing the requirements set out for the project and it helped achieve all the functionality. PHP also offers server side security which adds server side inputs which cannot be manipulated by the user and insures system integrity which is very important for the application.

2.3.3.5 Stripe Payments

Stripe payments offer a secure payment system without having to keep the users details in the database. Stripe offers users to register with a valid phone number and email address, registering will allow users to retrieve their stored payment methods for future bookings. A test card is offered by Stripe which was used to test bookings and was given to users to test. Stripe is a great solution in offering secure payments.



jelico1@icloud.com

•••• •••• •••• 4242

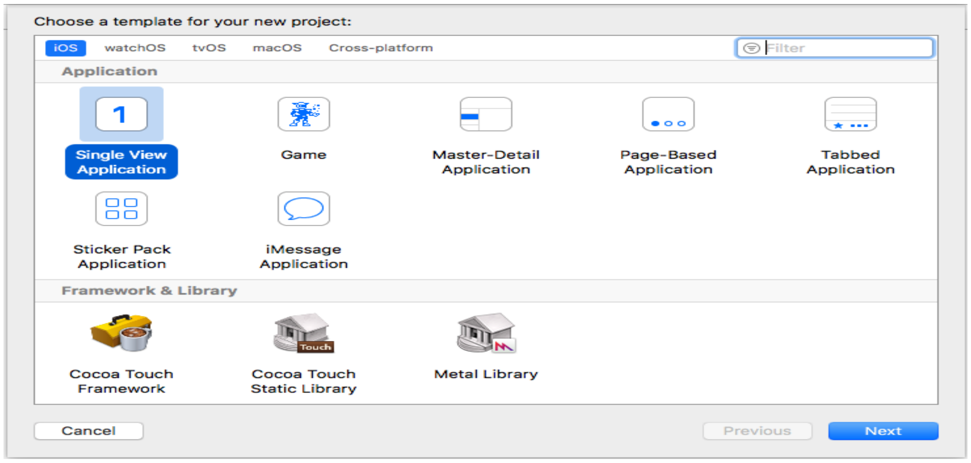
02 / 20

•••

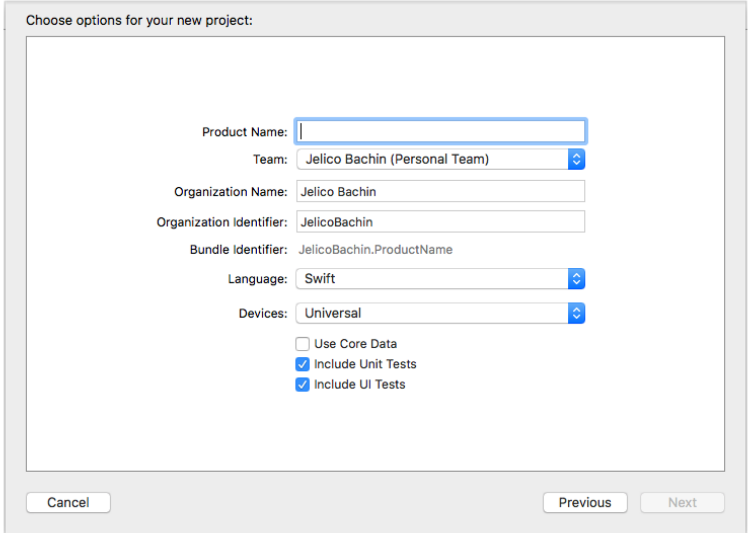
2.3.4 Procedures

2.3.4.1 Project Creation

To create the application a template had to be chosen from the variety of templates offered by XCode and Apple. After researching the internet, I discovered that Single View Application was the most suitable for my needs. To create the project, you click file > new > New Project then you are presented the templates and a template must be chosen.



After choosing the template you must choose the Project (App) name, programming language and Apple Developers account. It is also very important that the devices are set to universal as this enables the compatibility of all models of the iPhone.


Choose options for your new project:

Product Name:
Team: Jelico Bachin (Personal Team)
Organization Name: Jelico Bachin
Organization Identifier: JelicoBachin
Bundle Identifier: JelicoBachin.ProductName
Language: Swift
Devices: Universal

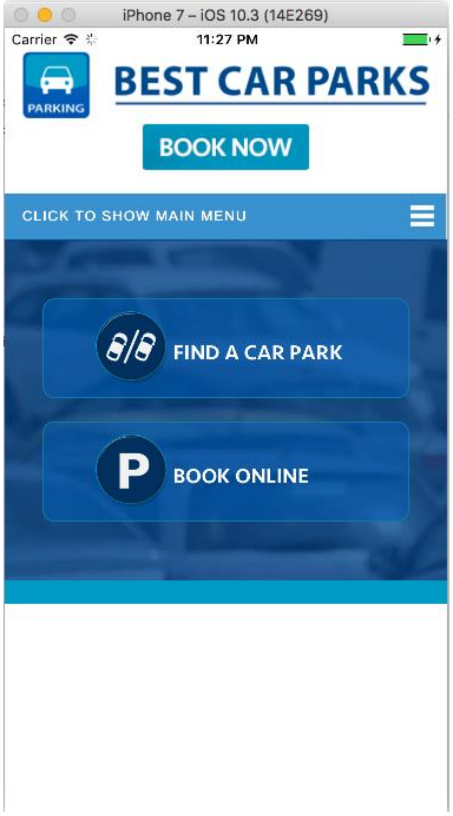
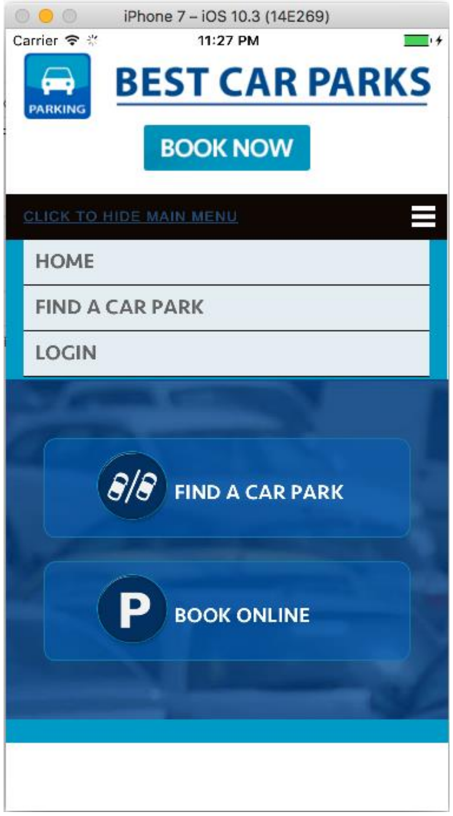
□ Use Core Data
☑ Include Unit Tests
☑ Include UI Tests

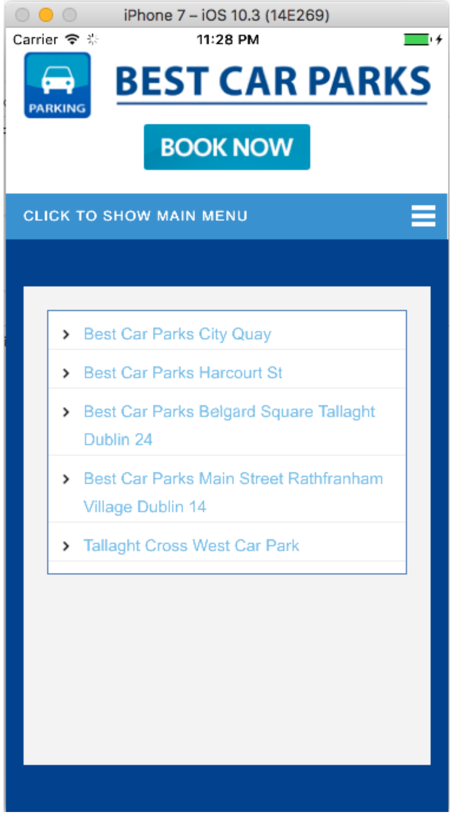
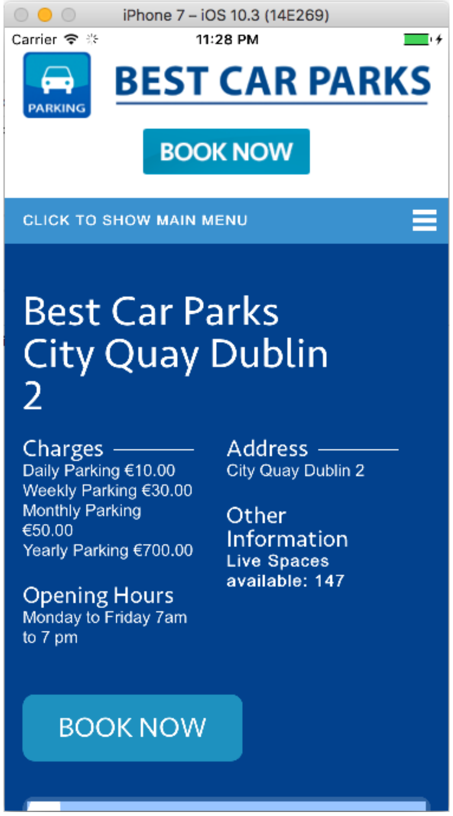
2.3.4.2 Custom Classes

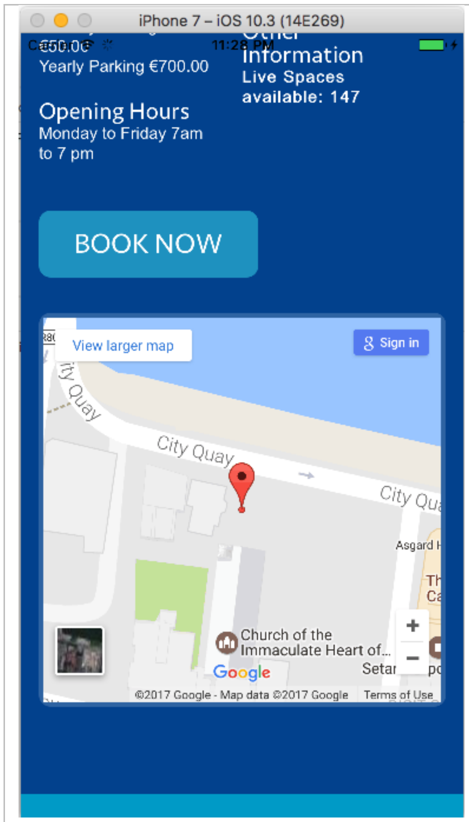
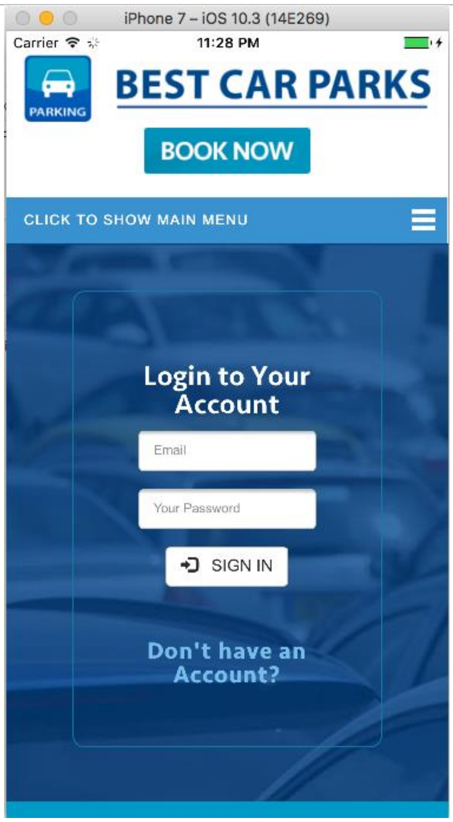
In XCode each view must be controlled by a class. I created a class called Web which was assigned to the Web View interface. This class was then connected to the view and it allowed me to implement the Web UI feature which helped integrate the load request NSURL method which loads the URL that you request. I then called my localhost server to display the mobile responsive website the application. I was then also able to implement a UIActivityIndicatorView which animates until the web page is fully displayed this gave the application a more iOS look and made it look more professional.

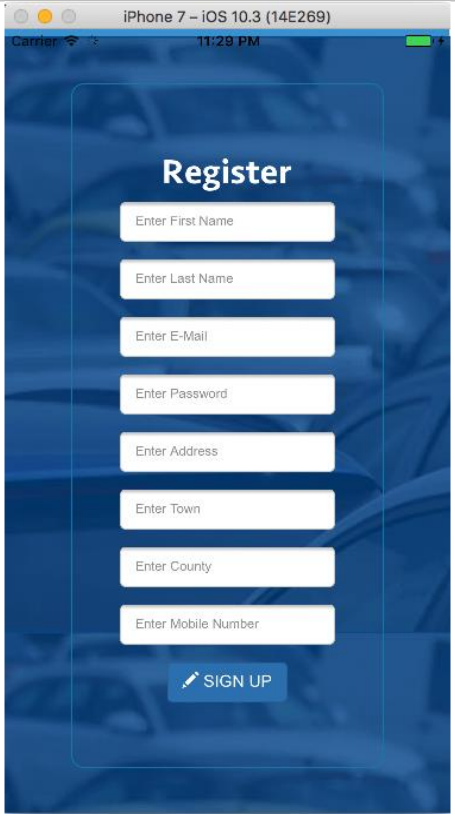
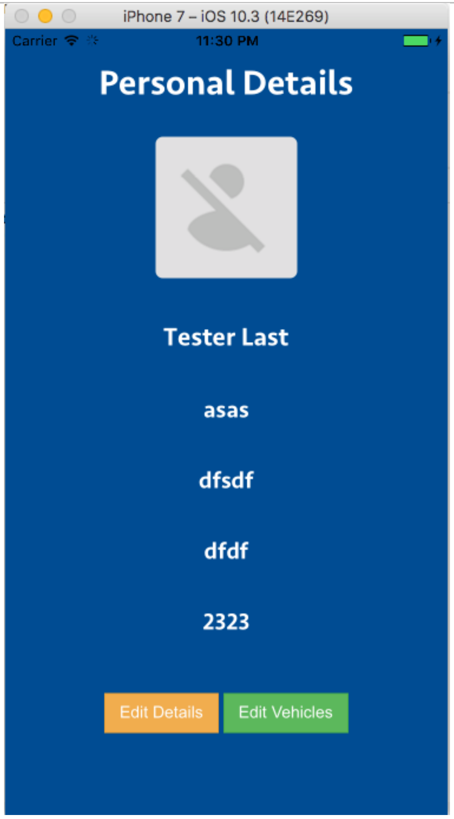
```
@IBOutlet weak var progressIndicator: UIActivityIndicatorView!

 webView.loadRequest(NSURLRequest(url: NSURL(string: "http://localhost/bestcarparks/pages/home.php")! as URL) as URLRequest)

progressIndicator.startAnimating()
```

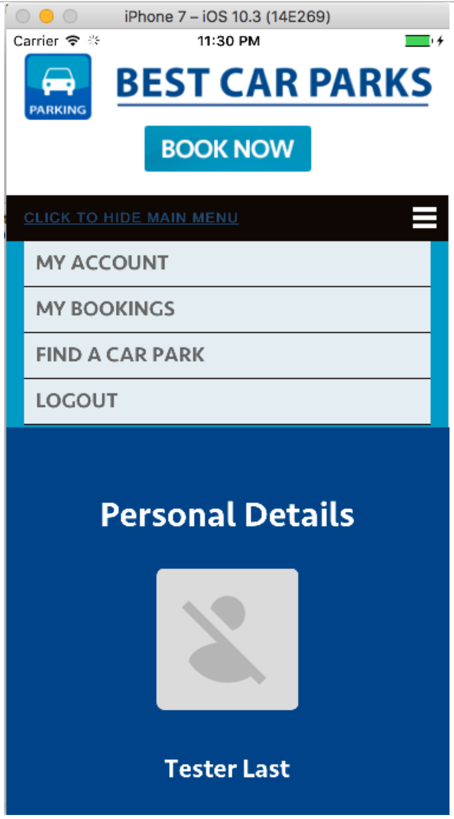
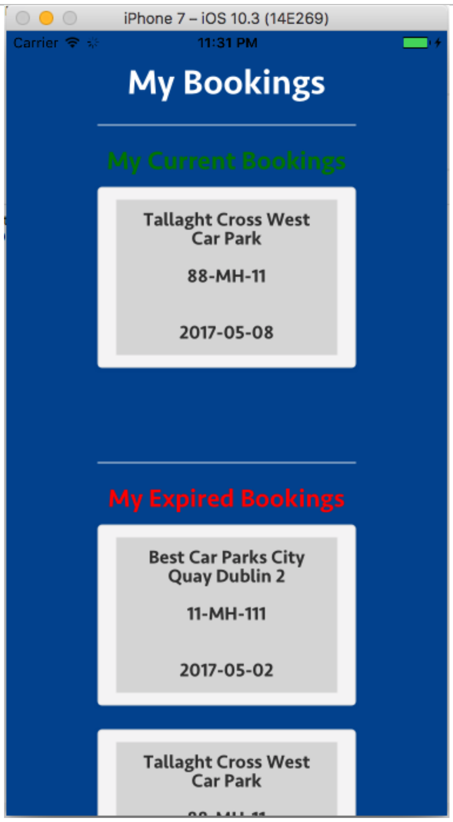
2.4 Graphical User Interface (GUI) Layout

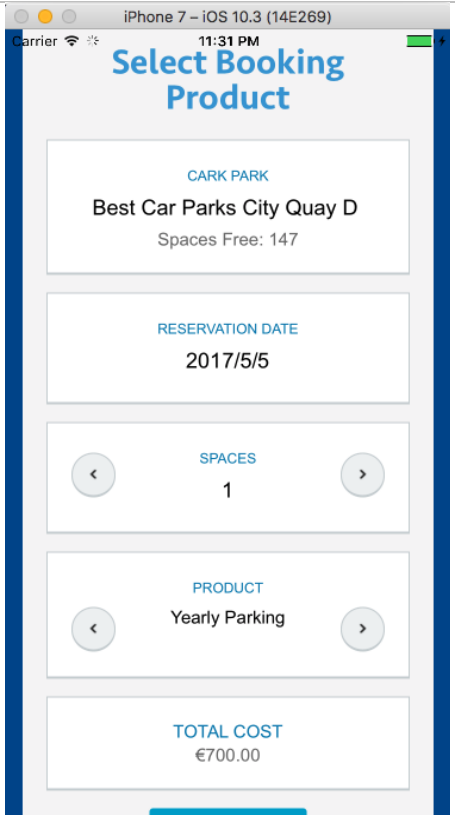
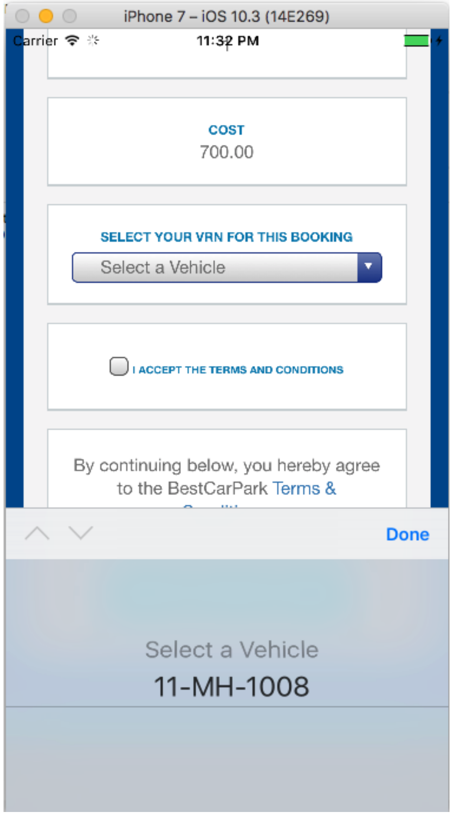
 iPhone 7 – iOS 10.3 (14E269) Carrier 11:27 PM PARKING BEST CAR PARKS BOOK NOW CLICK TO SHOW MAIN MENU FIND A CAR PARK BOOK ONLINE	 iPhone 7 – iOS 10.3 (14E269) Carrier 11:27 PM PARKING BEST CAR PARKS BOOK NOW CLICK TO HIDE MAIN MENU HOME FIND A CAR PARK LOGIN FIND A CAR PARK BOOK ONLINE
This is the interface the user is presented when they launch the application.	This shows the navigation bar the user is provided with before logging in.

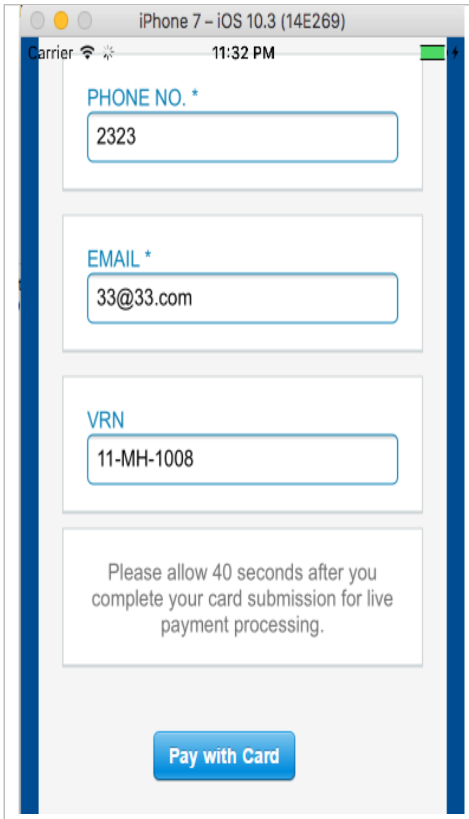
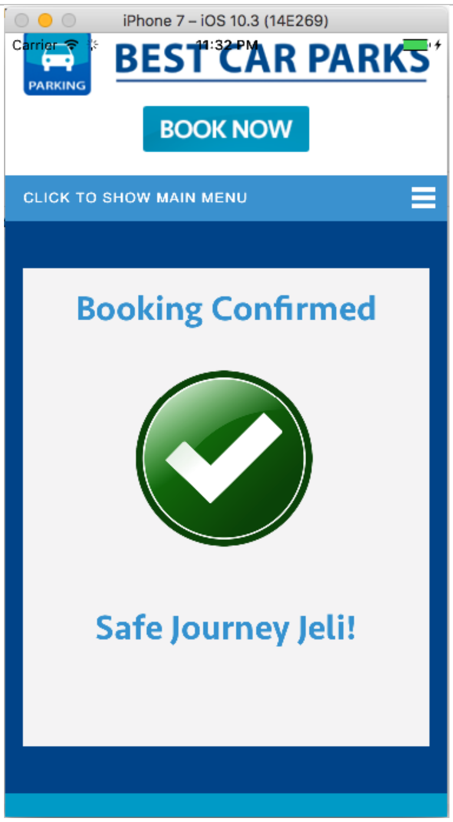
 iPhone 7 – iOS 10.3 (14E269) Carrier 11:28 PM PARKING BEST CAR PARKS BOOK NOW CLICK TO SHOW MAIN MENU Best Car Parks City Quay Best Car Parks Harcourt St Best Car Parks Belgard Square Tallaght Dublin 24 Best Car Parks Main Street Rathfranham Village Dublin 14 Tallaght Cross West Car Park	 iPhone 7 – iOS 10.3 (14E269) Carrier 11:28 PM PARKING BEST CAR PARKS BOOK NOW CLICK TO SHOW MAIN MENU Best Car Parks City Quay Dublin 2 Charges Daily Parking €10.00 Weekly Parking €30.00 Monthly Parking €50.00 Yearly Parking €700.00 Opening Hours Monday to Friday 7am to 7 pm Address City Quay Dublin 2 Other Information Live Spaces available: 147
This is the interface the user chooses the Find A Car Park option from the navigation pane	When the user chooses, a car park this is the interface he/she is displayed with. This provides prices, opening hours and live spaces available in the car park.

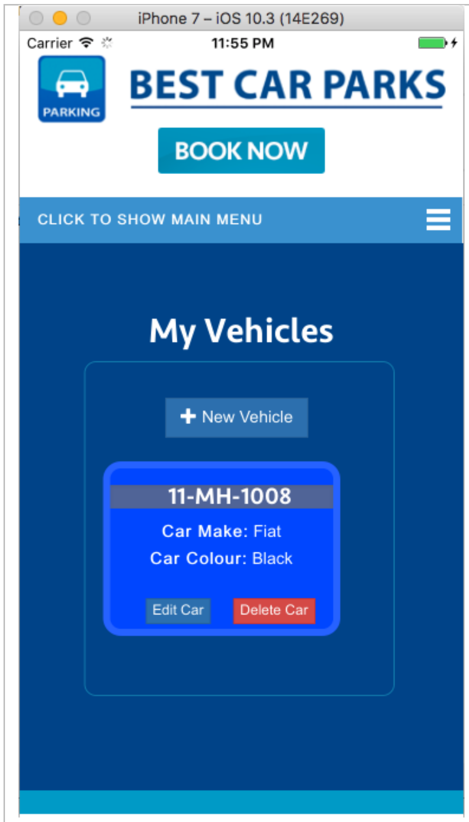
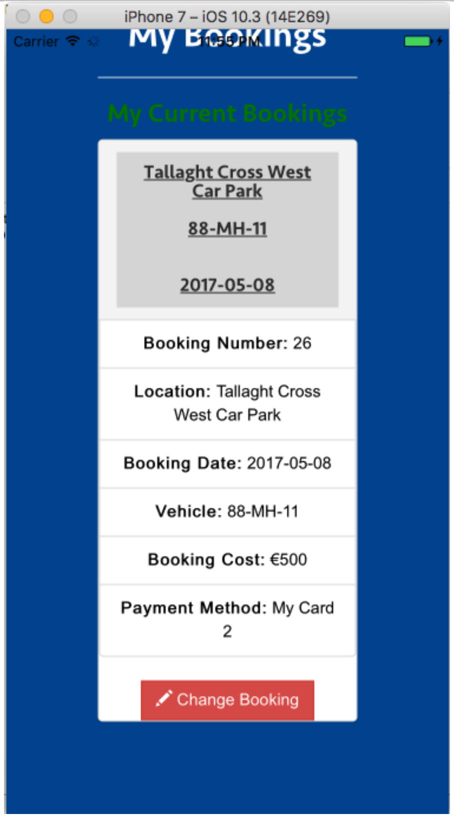
 iPhone 7 – iOS 10.3 (14E269) Carrier 11:28 PM €50.00 Yearly Parking €700.00 Other Information Live Spaces available: 147 Opening Hours Monday to Friday 7am to 7 pm BOOK NOW View larger map Sign in City Quay Asgard H Church of the Immaculate Heart of... Google ©2017 Google - Map data ©2017 Google Terms of Use	 iPhone 7 – iOS 10.3 (14E269) Carrier 11:28 PM PARKING BEST CAR PARKS BOOK NOW CLICK TO SHOW MAIN MENU Login to Your Account Email Your Password SIGN IN Don't have an Account?
This interface shows a map of the car park location under the car park details.	This interface shows the login screen of the application.

 iPhone 7 – iOS 10.3 (14E269) Carrier 11:29 PM Enter First Name Enter Last Name Enter E-Mail Enter Password Enter Address Enter Town Enter County Enter Mobile Number SIGN UP Register	 iPhone 7 – iOS 10.3 (14E269) Carrier 11:30 PM Edit Details Edit Vehicles Personal Details Tester Last asas dfsdf dfdf 2323
This is the registration screen where the user registers an account.	After the user, has logged they will be presented with the following interface showing their personal details and giving them access to their details and vehicles.

 iPhone 7 – iOS 10.3 (14E269) Carrier 11:30 PM PARKING BEST CAR PARKS BOOK NOW CLICK TO HIDE MAIN MENU MY ACCOUNT MY BOOKINGS FIND A CAR PARK LOGOUT Personal Details Tester Last	 iPhone 7 – iOS 10.3 (14E269) Carrier 11:31 PM My Bookings My Current Bookings Tallaght Cross West Car Park 88-MH-11 2017-05-08 My Expired Bookings Best Car Parks City Quay Dublin 2 11-MH-111 2017-05-02 Tallaght Cross West Car Park
This is shows the options a user is presented with in the navigation bar after the have logged in.	This interface shows the users bookings. The bookings are split into current bookings and expired bookings.

 iPhone 7 – iOS 10.3 (14E269) Carrier 11:31 PM Select Booking Product CARK PARK Best Car Parks City Quay D Spaces Free: 147 RESERVATION DATE 2017/5/5 SPACES ‹ 1 › PRODUCT ‹ Yearly Parking › TOTAL COST €700.00	 iPhone 7 – iOS 10.3 (14E269) Carrier 11:32 PM COST 700.00 SELECT YOUR VRN FOR THIS BOOKING Select a Vehicle ▾ □ I ACCEPT THE TERMS AND CONDITIONS By continuing below, you hereby agree to the BestCarPark Terms & Done Select a Vehicle 11-MH-1008
This is the interface the user is presented after they have pressed the book now on the car park they have chosen. The spaces available in the car park is displayed. The user chooses their product date etc.	After the user confirms their details they are told to choose a vehicle from their vehicles stored on the account.

 iPhone 7 – iOS 10.3 (14E269) Carrier 11:32 PM Pay with Card PHONE NO. * 2323 EMAIL * 33@33.com VRN 11-MH-1008 Please allow 40 seconds after you complete your card submission for live payment processing.	 iPhone 7 – iOS 10.3 (14E269) Carrier 11:32 PM PARKING BEST CAR PARKS BOOK NOW CLICK TO SHOW MAIN MENU Booking Confirmed Safe Journey Jeli!
The user is presented with a selection of their booking details and the VRN they choose. They are then asked to pay by card using the stripe interface.	This is the confirmation interface a user is presented with after paying for a booking.

 iPhone 7 – iOS 10.3 (14E269) Carrier 11:55 PM PARKING BEST CAR PARKS BOOK NOW CLICK TO SHOW MAIN MENU My Vehicles + New Vehicle 11-MH-1008 **Car Make:** Fiat **Car Colour:** Black Edit Car Delete Car	 iPhone 7 – iOS 10.3 (14E269) Carrier 11:55 PM My Bookings My Current Bookings Tallaght Cross West Car Park 88-MH-11 2017-05-08 **Booking Number:** 26 **Location:** Tallaght Cross West Car Park **Booking Date:** 2017-05-08 **Vehicle:** 88-MH-11 **Booking Cost:** €500 **Payment Method:** My Card 2 Change Booking
This is the interface that shows all the users saved vehicles on their account providing them with the option to edit the car or delete the car from their account.	When the user views their current bookings, they are provided with a change booking option that allows them to modify the booking details.

2.5 Testing

Test Plan

A test plan is a document detailing the objectives, target market, internal beta team and the processes involved in a specific beta test for a software or hardware product. I developed a test plan to ensure that all the testing I wanted to carry out was achieved. The different types of testing methods I had planned to carried out can be compared to the actual testing methods carried out below, the green highlighted fields display the testing methods that were performed . A copy of my test plan can be viewed below.

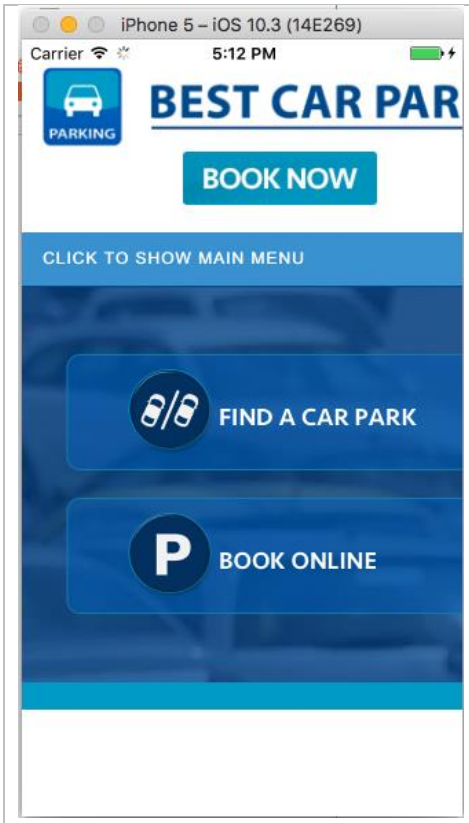
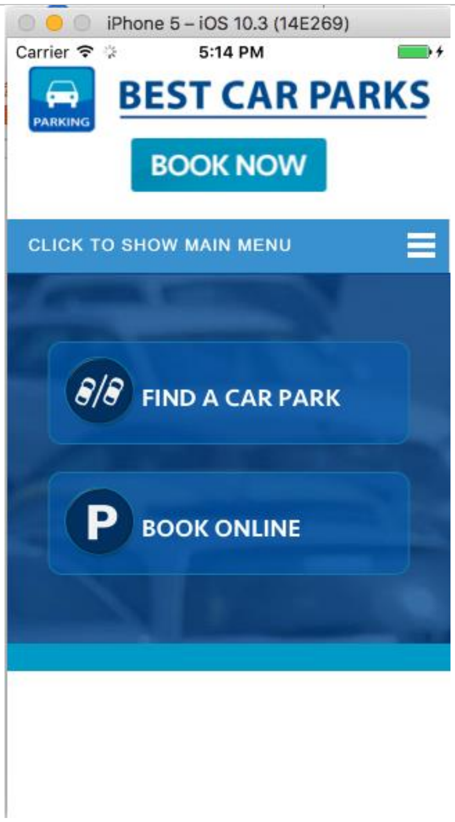
Approach	Type Of Testing	Manual Testing		Automated Testing on Device
		Using Device	Using Simulator	
Standard Testing	Integration Testing	Yes	Yes	No
	System Testing	Yes	Yes	Yes
	Usability Testing	Yes	Yes	No
	Acceptance Testing	Yes	Yes	No
	Verification	Yes	Yes	No
Special Types of Testing to address specific challenges	Compatibility Testing	Yes	Yes	Yes
	GUI Testing	Yes	Yes	No
Type of testing more relevant for enterprise mobile business application	Performance Testing	Yes	Yes	No
	Security Testing	No	No	Yes
	Synchronization Testing	Yes	Yes	Yes

Verification

Verification is the process to make the sure that the application satisfies the conditions imposed at the start of the development phase, it is done to make sure the application behaves the way it is designed to and for the purpose its built. Verification has been conducted thought the application to make sure each feature behaves the way it should per the requirements. It is very important to conduct verification on all components of the application to avoid the release of a version with bugs. After conducting verification on all features, I have discovered that the VRN was not saving in the database when making a booking and it was crucial that this bug was discovered and solved as the booking system would have been compromised and not functioning correctly.

System Testing

System testing is testing to ensure that the application works in different environments (older iPhones, older firmware versions) as expected. System testing is done with fully system implementation and environment. This type of testing falls under the class of black box testing. XCode's simulator was used to test the application under different environments and ensure the application is running as expected. This helped ensure that the application was adaptable to any iPhone devices provided by Apple. The application was first tested on the iPhone 5 in the simulator which was produced in 2012. After running the application on the iPhone 5, I made the discovery that the application was not loading on the full screen and half of the interface was not accessible due it being beyond the user reach. This bug was fixed using the auto constraint on the Web View in XCode. The testing then carried on to the iPhone 6, iPhone 6 Plus, iPhone 7 and iPhone 7 Plus. Results were all good and the application was displayed correctly. The results of the test can be viewed below.

 iPhone 5 – iOS 10.3 (14E269) Carrier 5:12 PM PARKING BEST CAR PAR BOOK NOW CLICK TO SHOW MAIN MENU FIND A CAR PARK BOOK ONLINE	 iPhone 5 – iOS 10.3 (14E269) Carrier 5:14 PM PARKING BEST CAR PARKS BOOK NOW CLICK TO SHOW MAIN MENU FIND A CAR PARK BOOK ONLINE
iPhone 5 before constraints were added.	**iPhone 5** after constraints were added. The content all now looks perfectly scaled.

Usability Testing

Usability testing is done to the perspective of the client to learn how user friendly the GUI is and how easily can the client use the interface, if the design meets the standards of the application users and if how easy and efficient the design is, this testing method falls under the class of black box testing. After putting all the factors into consideration, a survey was drafted by myself to ask 3 normal customers that are frequent car park and iPhone users to test the application and document their opinions on the design, GUI and how efficient the application is. The 3 users were picked from McDonald's Head Office my place of work, they were asked to use the application for 5 minutes at least and to answer 5 questions after. The results of the users can be found below.

Q1) How often do you use iPhone applications?
Daily
Q2) What is your first impression of the application?
Very Neat, right to the point
Q3) Do you like the design? Explain why
Yes, colors are very good on the eye
Q4) Is the application user friendly and easy to use?
Application is easy to use and everything is accessed easily.
Q5) Would you book parking using this application?
Yes 100%

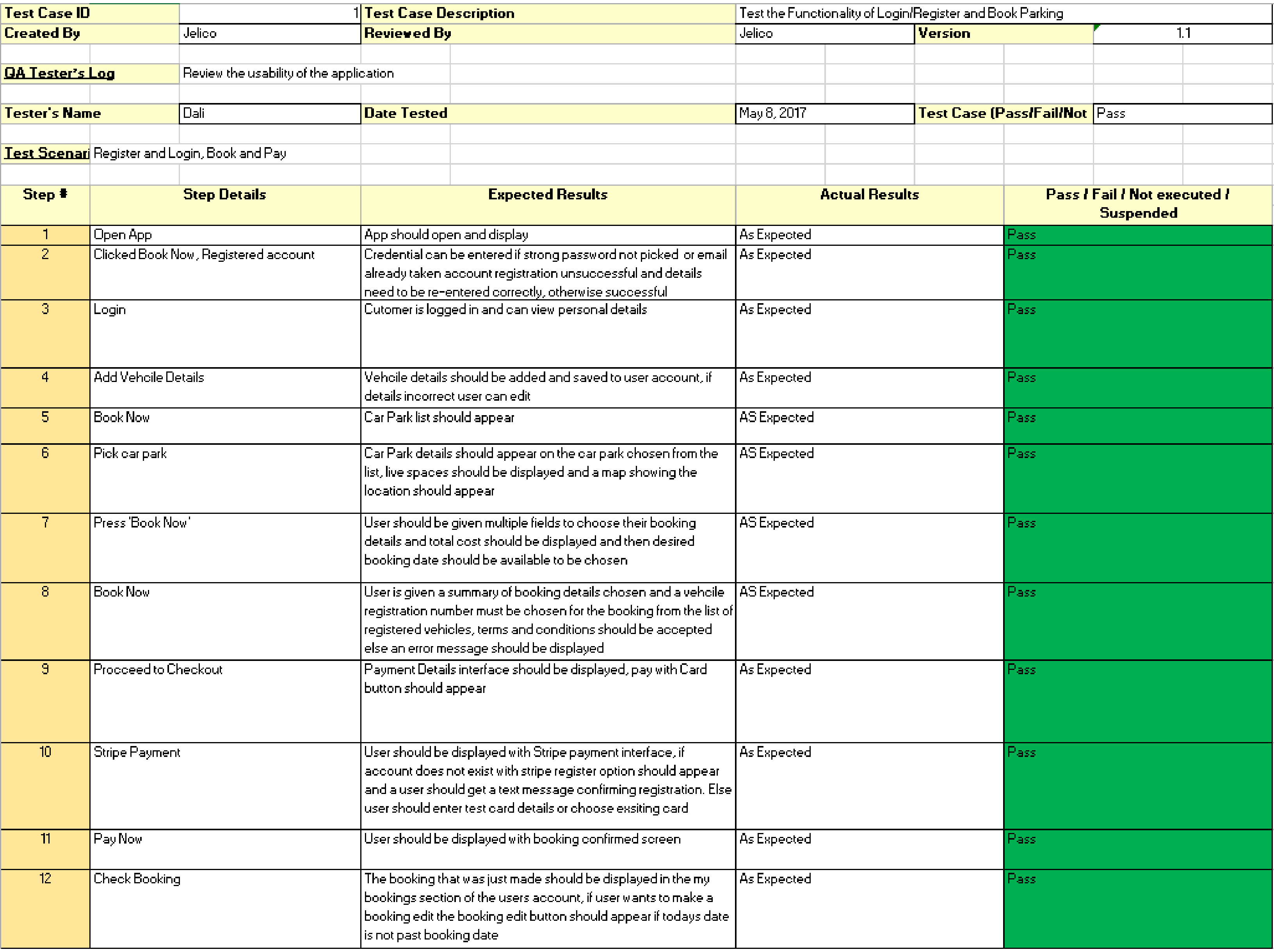
Q1) How often do you use iPhone applications?
Daily
Q2) What is your first impression of the application?
Impressive
Q3) Do you like the design? Explain why
Yes, the navigation bar makes everything be accessed easily
Q4) Is the application user friendly and easy to use?
Yes, and I’m not very technological
Q5) Would you book parking using this application?
When needed yes.

Q1) How often do you use iPhone applications?
Once a day
Q2) What is your first impression of the application?
Good
Q3) Do you like the design? Explain why
Yes, all the main features are accessed ok
Q4) Is the application user friendly and easy to use?
User friendly yes, all ok
Q5) Would you book parking using this application?
Sure

Acceptance Testing

This type of testing is done by the customer to ensure that the delivered product meets the requirements as the customer expected. A test case helps with acceptance testing as a test case is a set of conditions or variables under which a tester will determine wheter a system test satisfies requirements or works correctly.

A test case was developed by myself and my brother was asked to test the application as he had never seen my application until today. While he was testing the application, I was busy documenting all the results. I then took the results recorded and started to fill out the test case with the results I had written down. I began to document the step details my brother took, the expected results, the actual results and if the step passed or failed. After I documented the results I realized I had to make some changes and improvements to my application. This testing helped me realize the errors a customer could have faced if they were not repaired before app release. Some recommendations were made by the tester and they have been added. The results of the test can be found below.



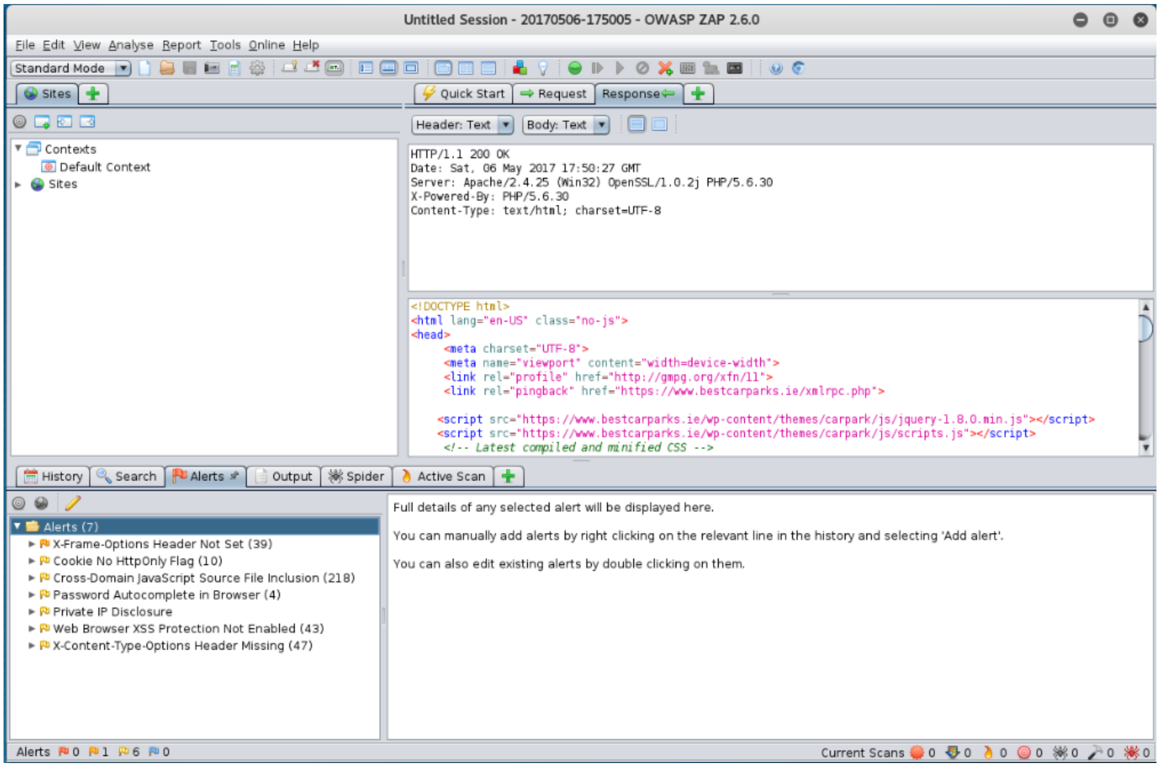
Test Case ID	1	**Test Case Description**	Test the Functionality of Login/Register and Book Parking		
Created By	Jelico	**Reviewed By**	Jelico	**Version**	1.1
QA Tester's Log	Review the usability of the application				
Tester's Name	Dali	**Date Tested**	May 8, 2017	**Test Case (Pass/Fail/Not**	Pass
Test Scenari	Register and Login, Book and Pay				

Step #	Step Details	Expected Results	Actual Results	Pass / Fail / Not executed / Suspended
1	Open App	App should open and display	As Expected	Pass
2	Clicked Book Now, Registered account	Credential can be entered if strong password not picked or email already taken account registration unsuccessful and details need to be re-entered correctly, otherwise successful	As Expected	Pass
3	Login	Cutomer is logged in and can view personal details	As Expected	Pass
4	Add Vehcile Details	Vehcile details should be added and saved to user account, if details incorrect user can edit	As Expected	Pass
5	Book Now	Car Park list should appear	AS Expected	Pass
6	Pick car park	Car Park details should appear on the car park chosen from the list, live spaces should be displayed and a map showing the location should appear	AS Expected	Pass
7	Press 'Book Now'	User should be given multiple fields to choose their booking details and total cost should be displayed and then desired booking date should be available to be chosen	AS Expected	Pass
8	Book Now	User is given a summary of booking details chosen and a vehcile registration number must be chosen for the booking from the list of registered vehicles, terms and conditions should be accepted else an error message should be displayed	AS Expected	Pass
9	Procceed to Checkout	Payment Details interface should be displayed, pay with Card button should appear	As Expected	Pass
10	Stripe Payment	User should be displayed with Stripe payment interface, if account does not exist with stripe register option should appear and a user should get a text message confirming registration. Else user should enter test card details or choose exsiting card	As Expected	Pass
11	Pay Now	User should be displayed with booking confirmed screen	As Expected	Pass
12	Check Booking	The booking that was just made should be displayed in the my bookings section of the users account, if user wants to make a booking edit the booking edit button should appear if todays date is not past booking date	As Expected	Pass

Security Testing

Security testing is carried out to ensure your application and source code is protected and free from security threats. Using Kali Linux security tests can be carried out on Web Applications. After scanning the Web Application directory using OWASP Zap a tool for finding security threats that refer to the OWASP Top 10 I found that the application was vulnerable to 7 types of attacks but some were not relevant as the source code was hosted on a local server and these threats disappear when the source code is moved to a domain hosted on a remote server.

The results also showed that password autocomplete was available and not locked down which is a security threat. With the help of OWASP Zap this security flaw has been removed. The results of the security test can be found below.



```
HTTP/1.1 200 OK
Date: Sat, 06 May 2017 17:50:27 GMT
Server: Apache/2.4.25 (Win32) OpenSSL/1.0.2j PHP/5.6.30
X-Powered-By: PHP/5.6.30
Content-Type: text/html; charset=UTF-8
```

```
<!DOCTYPE html>
<html lang="en-US" class="no-js">
<head>
     <meta charset="UTF-8">
     <meta name="viewport" content="width=device-width">
     <link rel="profile" href="http://gmpg.org/xfn/11">
     <link rel="pingback" href="https://www.bestcarparks.ie/xmlrpc.php">

    <script src="https://www.bestcarparks.ie/wp-content/themes/carpark/js/jquery-1.8.0.min.js"></script>
    <script src="https://www.bestcarparks.ie/wp-content/themes/carpark/js/scripts.js"></script>
     <!-- Latest compiled and minified CSS -->
```

Alerts (7)
- X-Frame-Options Header Not Set (39)
- Cookie No HttpOnly Flag (10)
- Cross-Domain JavaScript Source File Inclusion (218)
- Password Autocomplete in Browser (4)
- Private IP Disclosure
- Web Browser XSS Protection Not Enabled (43)
- X-Content-Type-Options Header Missing (47)

Full details of any selected alert will be displayed here.

You can manually add alerts by right clicking on the relevant line in the history and selecting 'Add alert'.

You can also edit existing alerts by double clicking on them.

Integration Testing

Integration testing is the phase in which individual features or modules are combined and tested as a group this type of testing occurs before validation testing. Integration testing was performed throughout my project every time a new feature or functionality was added as it helped me see how all the functionalities connected and worked together. I had to particularly perform this type of testing after I added the login and registration functionality to ensure that there was a communication with the database. When user sessions were added this type of testing helped ensure the functionality was working as a communication between the login and the user session was vital to store the user session and display the user’s correct details and give access to the booking services. Results were recorded and any errors found were fixed.

GUI Testing

GUI testing is the process of testing the systems graphical user interface of the application under test. GUI testing involves checking the screens with the controls like menus, buttons, icons and all types of bars etc. GUI is what the user sees and it is important this type of testing is carried out to ensure a fluid and error free GUI is offered to the user. I had performed GUI testing to make sure that all the buttons such as ‘Book Now’, ‘Edit Vehicles’ etc. we’re functioning correctly and the GUI was displayed once the button had been clicked. I had discovered that the logo was not displaying any GUI once clicked and this error was quickly resolved. All other GUI’s of the application were functioning correctly.

3 Conclusions

The project is developed for Bestcarparks Ireland. This is a disadvantage in my opinion but it is also an advantage as the development and implementation of features becomes much easier and less stressful as all the objectives and features of the application are clearly stated in the planning stage. HTML5, PHP and SQL have allowed me to create the source code for the website that the application will implement and display. PHP has allowed me to add user sessions making the users usage of the application more secure but has also made the development of the application easier as users details can be retrieved by sessions. Using the Web View Interface provided by XCode allowed me to build a website that was mobile responsive and adaptable for the iPhone this meant that application was not limited to the capabilities of the Swift programming language which are very limited, a SQLite database can only be added which was not helpful as the application was intended for multiple users and not specifically for one user but with the implementation of PHP I was able to overcome this issue . The application developed is the only iOS application that is available in Ireland for a private car parking company. The tests of the application were all positive and myself and the customer (Bestcarparks) were very happy of the end product and are excited to see the application live on the App Store in the near future(June 2017). The application development process was very enjoyable and the results of the application show the amount of time and effort that were applied into the development of this application. I hope to be developing this type of project in the future.

4 Further development

This application being developed is specifically designed for Bestcarparks Ireland therefore further developments are linked to the company’s needs but not only. Updates will be released to improve performance of the application but to also fix any bugs that will arise. The following further improvements can be released:

Language

For a further development, the application will have different language support for ease of access to tourists when visiting Ireland.

Parked Location

The application could have the ability to allow the user to add a pin to a map allowing the user to store where he has parked his car to make it easy for the user to remember where he has parked.

Loyalty System

The application could have a loyalty system rewarding the user with gifts or bonus points for using the application to purchase parking.

Push Notifications

A Notification should be sent to the user’s phone once a booking is confirmed and a reminder should also be sent to remind the user of the booking.

5 References

Irish Life | Parking Ireland | Park Rite. 2017. *Irish Life | Parking Ireland | Park Rite*. [ONLINE] Available at: http://www.parkrite.ie/carparks/irish-life. [Accessed 09 Jan 2017].

Irish Parking Association - Ireland's Parking Industry Representative. 2017. *Irish Parking Association – Ireland's Parking Representative*. [ONLINE] Available at: http://www.parkingireland.ie/. [Accessed 09 Jan 2017].

CodeProject. 2017. *What is software testing? What are the different types of testing? - CodeProject*. [ONLINE] Available at: https://www.codeproject.com/Tips/351122/What-is-software-testing-What-are-the-different-ty. [Accessed 06 May 2017].

JSFiddle. 2017. *Edit fiddle - JSFiddle*. [ONLINE] Available at: http://jsfiddle.net/zunrk/. [Accessed 06 May 2017].

ios - Add an activity indicator on a uiwebview - Stack Overflow. 2017. *ios - Add an activity indicator on a uiwebview - Stack Overflow*. [ONLINE] Available at: http://stackoverflow.com/questions/19475350/add-an-activity-indicator-on-a-uiwebview. [Accessed 06 May 2017].

iTech Empires. 2017. *Login Registration System with PHP Data Object (PDO)*. [ONLINE] Available at: http://www.itechempires.com/2016/06/php-login-registration-system-with-php-data-object-pdo/. [Accessed 06 May 2017].

Pradeep khodke. 2017. *PHP Login and Registration Script with PDO and OOP | Coding Cage*. [ONLINE] Available at: http://www.codingcage.com/2015/04/php-login-and-registration-script-with.html. [Accessed 06 May 2017].

Testing. 2017. *Testing*. [ONLINE] Available at: https://stripe.com/docs/testing. [Accessed 09 May 2017].

** References provided thought the code also **

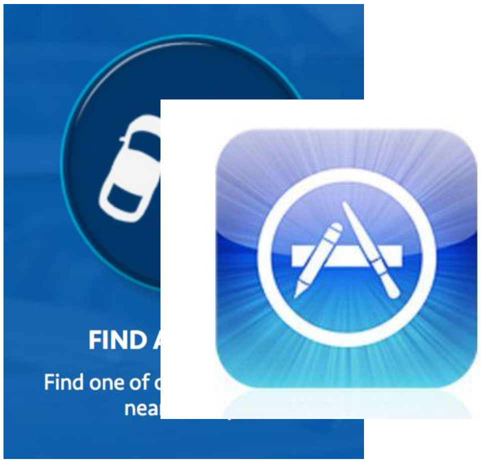
6 Appendix

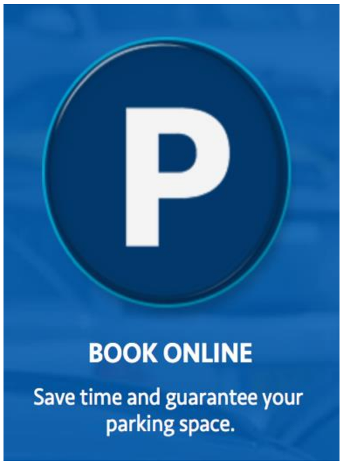
Project Proposal

Bestcarparks iOS Application

PARKING

BEST CAR PARKS





Jelico Bachin X13301011

X13301011@student.ncirl.ie BSc (Hons) in Computing

Specialisation Cyber Security October 2016

Objectives

To design and develop an application for Bestcarparks Ireland. This application targets anyone that needs to book a car parking space on the go. The application will provide a fluid user experience with an uncomplicated flow between screens when users are using the application.

Users will can view where the car parks of Bestcarparks are located and provide directions to each. The car parks will be displayed on a map and the users will can receive directions to any of the car parks they choose from their location.

If a customer needs to find information about the company, the application provides details of the car park company and their car parks.

Users will can register an account with the company to allow them to save time on future bookings. They will also be able to login to their accounts at any time using their registration username and password.

The application will have an online payment system to allow users to pay for their car parking from their mobile device, this is done to provide users with the best parking experience possible. Users will also have the option to pay clamping fines they have received. A loyalty system will be integrated to ensure the customer is rewarded when booking through the app.

Users must login into their account through the two-factor authentication login process to increase account security.

The application will also have integrated a security feature to ensure a customer cannot book multiple spaces in different car parks at the same time and it will also have integrated a security feature to allow users to get notified if they have spent over a certain amount of money on bookings per day.

Background

I first thought of the concept for this idea several months ago when I needed to attend a meeting in Dublin City Centre and was uncertain of where car parks were located and their daily rates. If an app like this existed, it would have made the journey to my meeting less stressful as I would have known where to park my car.

After conducting some research in the car parking business regarding applications for smartphones or smart devices. I also discovered that a parking app already existed for the Dublin City Council parking spaces located throughout Dublin City. The app was called Parking Tag and allowed users to pay for public parking from their smartphone. The main factor that customers look at when booking a car park space is the daily rate.

The daily rates were very high compared to any of the private car parking companies available in Ireland. This made me conduct some research particularly in the private car parking companies, how easy it is for users to find information about daily rates, car park locations etc. but particularly if a mobile application was available that allowed users to book parking. I also discovered that no mobile application was available for private car parking companies and this gave me the idea of developing such app.

After coming up with the idea I contacted a bunch of car parking companies regarding my idea and enquired if they wanted to get on board. I set up a meeting with the Managing Director of Bestcarparks Ireland. Bestcarparks is a car parking company established and located in Ireland. It is the largest car park company in Ireland with over 5,000 car park spaces in Ireland in more than 13 parking sites.

After fully explaining the idea to the managing director he was more than happy to get on board. He provided me with feedback regarding technical functionalities of the mobile application and provided me with some extra features that he will like to be implemented. I then presented this idea to the judges at my project pitch and the idea was accepted. I was ready to move forward with this project.

Technical Approach

The project task is to create a mobile application for iOS for Bestcarparks Ireland. This will include research into iOS development particularly in the swift language for smartphones and iPads.

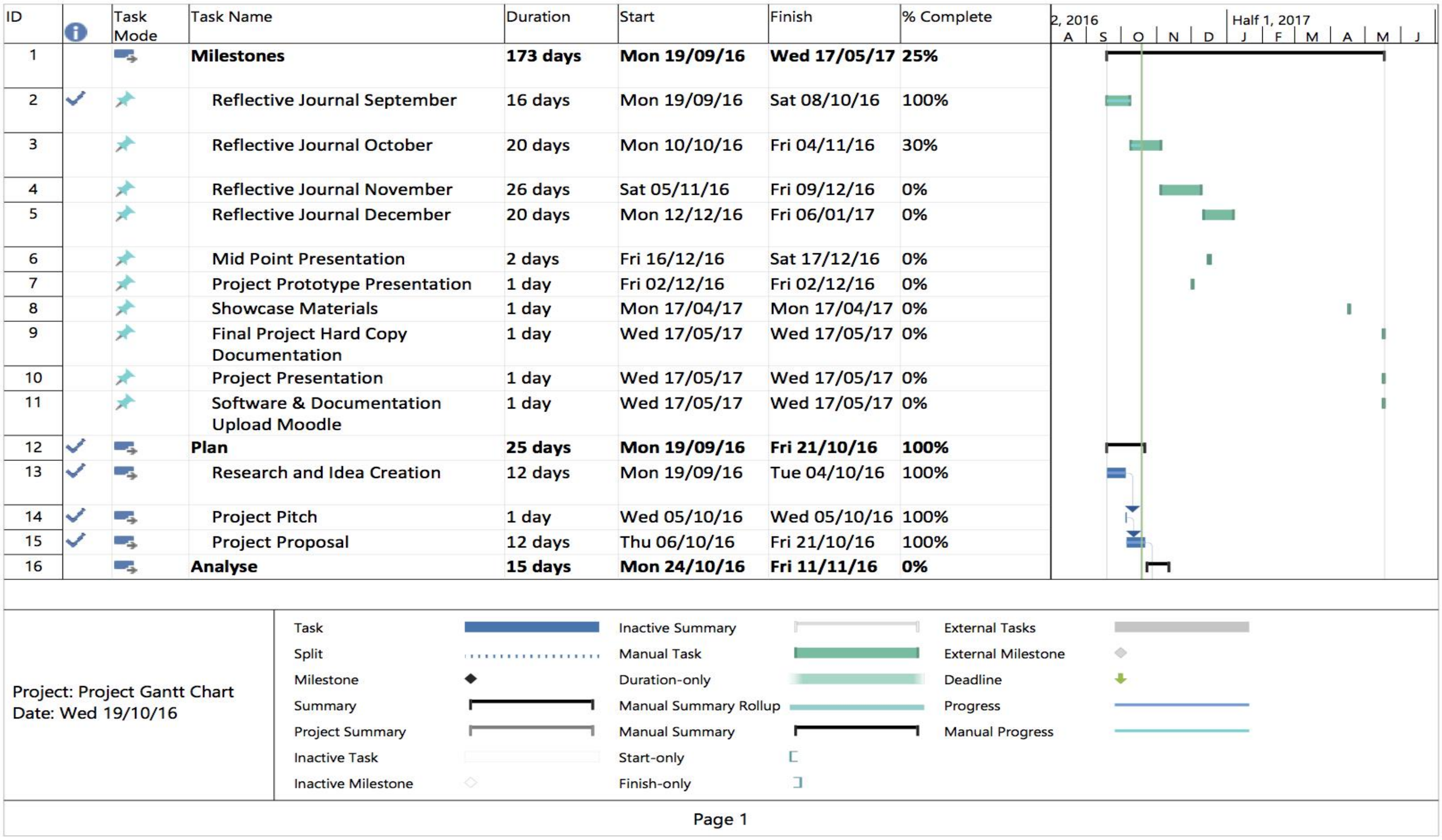
- Define the functional requirements, non-functional requirements and scope of the project, and develop UI wireframes.
- Research the swift language and identify different libraries and technologies that can be used in the application
- Research XCode IDE features to speed up the development of the application.
- The project will involve researching and creating a two-factor authentication login.
- The project will involve researching and creating a method to get the users login details when a user presses the ‘Login’ button.
- The project will involve searching the integration of Realex payments in an iOS application for the swift language.
- The project will involve creating a database and connecting it to my application to maintain customer bookings etc.
- Once a user has logged into their account their account details must be displayed correctly.
- Once a user has logged in, the application must allow them to book and pay for their car parking for the duration requested.
- The project will involve researching and creating a security feature to not allow the users to book multiple car parking spaces in the same period.
- The project will involve researching and integrating a Google Maps API to display the car parking locations on a map and offer directions from the user’s location if requested.

Special Resources Required

The project will be developed in XCode for iOS and will be using an online server to provide and host an online database. This will be hosted on www.bestcarparks.ie, using the PHPMyAdmin user interface.

- Apple iPhone running iOS 10.
- XCode for Mac using the latest version.
- An Apple Developers account.

Project Plan



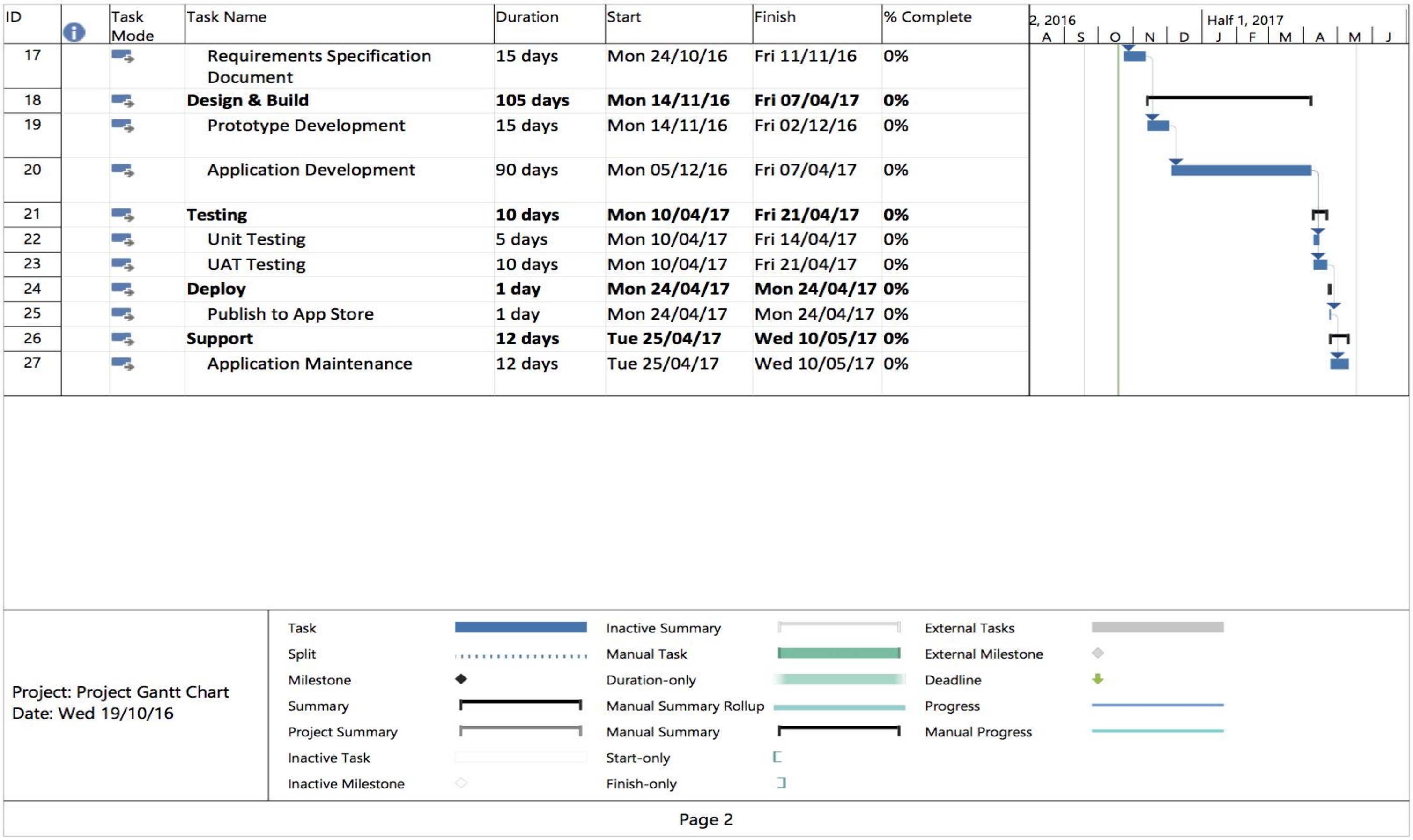
ID		Task Mode	Task Name	Duration	Start	Finish	% Complete
1			**Milestones**	**173 days**	**Mon 19/09/16**	**Wed 17/05/17**	**25%**
2	✓		Reflective Journal September	16 days	Mon 19/09/16	Sat 08/10/16	100%
3			Reflective Journal October	20 days	Mon 10/10/16	Fri 04/11/16	30%
4			Reflective Journal November	26 days	Sat 05/11/16	Fri 09/12/16	0%
5			Reflective Journal December	20 days	Mon 12/12/16	Fri 06/01/17	0%
6			Mid Point Presentation	2 days	Fri 16/12/16	Sat 17/12/16	0%
7			Project Prototype Presentation	1 day	Fri 02/12/16	Fri 02/12/16	0%
8			Showcase Materials	1 day	Mon 17/04/17	Mon 17/04/17	0%
9			Final Project Hard Copy Documentation	1 day	Wed 17/05/17	Wed 17/05/17	0%
10			Project Presentation	1 day	Wed 17/05/17	Wed 17/05/17	0%
11			Software & Documentation Upload Moodle	1 day	Wed 17/05/17	Wed 17/05/17	0%
12	✓		**Plan**	**25 days**	**Mon 19/09/16**	**Fri 21/10/16**	**100%**
13	✓		Research and Idea Creation	12 days	Mon 19/09/16	Tue 04/10/16	100%
14	✓		Project Pitch	1 day	Wed 05/10/16	Wed 05/10/16	100%
15	✓		Project Proposal	12 days	Thu 06/10/16	Fri 21/10/16	100%
16			**Analyse**	**15 days**	**Mon 24/10/16**	**Fri 11/11/16**	**0%**

Project: Project Gantt Chart
Date: Wed 19/10/16

Task		Inactive Summary		External Tasks	
Split		Manual Task		External Milestone	◇
Milestone	◆	Duration-only		Deadline	↓
Summary		Manual Summary Rollup		Progress	
Project Summary		Manual Summary		Manual Progress	
Inactive Task		Start-only	[		
Inactive Milestone	◇	Finish-only	]		

Page 1

ID 2 (Reflective Journal September)



ID		Task Mode	Task Name	Duration	Start	Finish	% Complete
17			Requirements Specification Document	15 days	Mon 24/10/16	Fri 11/11/16	0%
18			**Design & Build**	**105 days**	**Mon 14/11/16**	**Fri 07/04/17**	**0%**
19			Prototype Development	15 days	Mon 14/11/16	Fri 02/12/16	0%
20			Application Development	90 days	Mon 05/12/16	Fri 07/04/17	0%
21			**Testing**	**10 days**	**Mon 10/04/17**	**Fri 21/04/17**	**0%**
22			Unit Testing	5 days	Mon 10/04/17	Fri 14/04/17	0%
23			UAT Testing	10 days	Mon 10/04/17	Fri 21/04/17	0%
24			**Deploy**	**1 day**	**Mon 24/04/17**	**Mon 24/04/17**	**0%**
25			Publish to App Store	1 day	Mon 24/04/17	Mon 24/04/17	0%
26			**Support**	**12 days**	**Tue 25/04/17**	**Wed 10/05/17**	**0%**
27			Application Maintenance	12 days	Tue 25/04/17	Wed 10/05/17	0%

Project: Project Gantt Chart
Date: Wed 19/10/16

Page 2

This task is a milestone which involves creating a Reflective Journal for the month of September explaining everything that I have done towards my project. This Reflective Journal will need to be uploaded on Moodle before the 30th of September.

ID 3 (Reflective Journal October)

This task is a milestone which involves creating a Reflective Journal for the month of October explaining everything that I have done towards my project. This Reflective Journal will need to be uploaded on Moodle before the 31st of October.

ID 4 (Reflective Journal November)

This task is a milestone which involves creating a Reflective Journal for the month of November explaining everything that I have done towards my project. This

Reflective Journal will need to be uploaded on Moodle before the 30th of November.

ID 5 (Reflective Journal December)

This task is a milestone which involves creating a Reflective Journal for the month of December explaining everything that I have done towards my project. This Reflective Journal will need to be uploaded on Moodle before the 31st of December.

ID 6 (Mid-Point Presentation)

This task is a milestone it is a presentation given to several lecturers in NCI.

ID 7 (Prototype Presentation)

This task is a milestone it is a presentation of my project prototype given to several lecturers in NCI.

ID 8 (Showcase Materials)

This task is a milestone it will require me to have all the materials prepared for the project presentation.

ID 9 (Hard Copy Documentation)

This task is a milestone it will require me to have all the materials printed out and made into a booklet. I will need to submit this booklet to my college project supervisor.

ID 10 (Project Presentation)

This is task is a milestone I will be presenting my project to the lectures of NCI in May 2017.

ID 11 (Upload Moodle)

This task is a milestone it will require

ID 13 (Research & Idea Creation)

This task is involves researching multiple tasks leading to the creation of an idea. This is an essential task in the planning of the project.

ID 14 (Project Pitch)

This task involved presenting my idea to a team of judges who decided if my idea was accepted or declined.

ID 15 (Project Proposal)

This task involved creating a proposal document for my project idea.

ID 17 (Requirements & Specification Document)

This task involves creating a document containing the requirements and technical specifications regarding my project application.

ID 19 (Prototype Development)

This task involves developing a prototype of my application that will be showcased at the project prototype presentation in December. ID 20 (Application Development)

This task involves developing my application that should be fully functional.

ID 22 (Unit Testing)

This task involves carrying out Unit Testing on my code to determine whether parts of my code are fit for use.

ID 23 (UAT Testing)

This task involves carrying out UAT Testing on my application using real world scenarios to determine whether the application is functioning as intended.

ID 25 (Publishing Application)

This task involves publishing the application to the App Store so that customers of Bestcar parks can download and use the application.

ID 27 (Application Maintenance)

This task involves fixing any bugs that may appear with the application when it is live on the App Store.

Technical Details

The project will be an iOS app for an Apple iPhone. The application will be developed using Swift programming language for its logic and UI. The application will also contain a database to store information of the users, their booking etc. This will be created using PHP and MySQL.

The database would be used for the retrieval of customer details and bookings and would be implemented on a host using *PHPMyAdmin* as an interface on the server side and PHP backend in the mobile application to communicate with the server.

Evaluation

Unit testing

Unit testing will be implemented to run tests on individual functions or classes within the application, separately from the overall application. These unit tests will be used to highlight potential areas where bugs may occur later in the integration testing. Unit tests will be done locally within the use of XCode in the development environment and through tests on my personal Apple iPhone 6s Plus and iPhone 6s and through the XCode simulator.

User Acceptance Testing

User Acceptance Testing will be implemented to ensure that my application can handle required tasks in real-world scenarios, per specifications. These tests will be carried out on my personal devices and the test will be carried out by members of my family or close friends with a non-programming background.

Monthly Journals

September

Student name: **Jelico Bachin**

Programme: **BSc in Computing, Cyber Security Stream**

Month: **September**

My Achievements:

After 9 months of work placement I returned to college. The first week back felt odd as I was used to going to an office instead of college. I finally realised I returned to college when the word ‘projects’ was repeated non-stop in various sentences by my lectures and my classmates. I wasn’t very eager to returning to 4th year as I knew I would be under a lot more pressure this year than previous years.

After undergoing my placement I felt I gained a lot of new knowledge that will help me successfully get through this final year.

I spent this month giving a lot of thought towards my final year project. During the summer I didn’t spend any time thinking about the project by not doing this I felt it put me in a lot of pressure in the previous recent weeks. After a lot of research and thought I decided to make an iOS application for a car parking company. After spending a day trying to get in contact with various car parking companies, Bestcarparks Ireland was interested in my idea and they wanted to collaborate with me. The iOS application will allow users to search for any car parks the company provides and get directions to it from their iOS app and it will also allow them to book parking on the go and from their device. This app will also allow users to register an account to save time when booking a space in the future.

My contributions to the projects included, gathering my ideas and presenting my project pitch to a panel. My project was approved with some revisions suggested.

My Reflection:

I felt, it worked well for me to brainstorm multiple ideas on a piece of paper in order to find the perfect idea. Also due to the fact that I am a driving commuter parking in the city centre I thought that a parking app would be the perfect solution for finding parking on the go.

Now that my project was approved, I felt like some of the pressure was taken of my shoulders but looking ahead towards finishing this project is daunting. Due to the amount of projects we have received thought all of our modules managing my time is going to be the key of finishing this project successfully.

Intended Changes:

Next month, I will try to work on my project prototype and start working on my specifications document.

It was recommended to me that I meet with my supervisor to discuss my project and to implement more features in my project around my specialisation area.

October

Student name: **Jelico Bachin**

Programme: **BSc in Computing, Cyber Security Stream**

Month: **October**

My Achievements:

This month was all about the project proposal. I started doing the project proposal earlier in the month but I only finished it close to the submission deadline due to the other workload that was involved in my other modules. The project proposal was actually quite an enjoyable document to do as it reminded me of how I got my idea but it also gave me a few extra ideas to include in my project. The technical approach section of the proposal was quite challenging as I had to research all the technologies I am going to use for my project.

I then had to go on and create a Gantt Chart for my project doing this was quite challenging as the last time I had created a Gantt Chart was back in my first year of college. I had to spend some time researching these charts. I also had to research to see exactly what should be included in such a chart and the appropriate headers to use for each task. After I fully finished my Gantt Chart and the rest of my proposal document I reviewed and uploaded the document. The following week I started my Requirements Specification document. I started by looking at the previous documents I have created to refresh my memory.

After I was clear on the requirements required for this document I started to create my Use Case diagrams for my project. At first this was quite challenging as I needed to find some software that required me to create these Diagrams as Visio was not compatible with my MacBook but after I downloaded StarUML there wasn't anything stopping me from designing my use case diagrams.

My contributions to the projects included writing up my Project Proposal and uploading it to Moodle and creating my Use Case Diagrams for my Requirements Specification.

My Reflection:

I felt, it worked well to view refresh my memory browsing the previous Project Proposal I have done in my previous year of college as I was fully aware of what my tasks were. I also felt that when you are developing use case diagrams it is good to brainstorm your main project features under multiple headings to see exactly what is required in your use case diagrams and to discover your actors.

Intended Changes:

Next month I will finish and upload my requirements specification to Moodle and I will develop my Project Prototype.

I have only received my supervisor recently and I will meet with them to discuss my project.

November

Student name: **Jelico Bachin**

Programme: **BSc in Computing, Cyber Security Stream**

Month: **November**

My Achievements:

This month, was very stressful there were a lot of CA's and assignments to be handed in. But at the same time this had to put all the stress to the side and carry on with my work and I had to remain focused. After handing in my project requirements I was ready to start developing the prototype. Developing the prototype was quite a challenge as I had to get used to the XCode software and its capabilities. After conducting research in and around the software I could power on.

I started developing my GUI's for my mobile app and linked them all together. After running many simulations of the app, I was happy with the way the GUI's were linked and I was ready to add functionality to them. I started off by adding the GoogleMaps SDK which allowed me to add the locations on a map which later will allow directions to be given to a user. I then implemented a login system connected through firebase which ensured that only registered users were allowed access the booking system. I then went on and added in the booking GUI allowing users to choose the date, time, duration, car registration number. I added in the pay now section which allowed users to pay for their parking but everything was hardcoded as this was only a prototype.

I managed to also start writing my technical report and overall I was quite happy with my progress, putting aside all the stress and workload this semester.

My Reflection:

I felt, it worked well to meet regularly with my supervisor and ask him for every piece of advice or any question or topic I was unsure about. I also felt it worked great to interact with fellow class mates and discuss certain topics together and unite to come up with solution to help all.

Intended Changes:

Next month, I will try to fully finish my prototype and upload my technical report to Moodle after performing a few changes.

Supervisor Meetings:

Date of Meeting: 24/11/2016

Items discussed: Prototype, extra features, queries regarding report, requirements spec feedback

Action Items: Document ate technical report as per feedback and develop prototype with extra features recommended.

December

Student name: **Jelico Bachin**

Programme: **BSc in Computing, Cyber Security Stream**

Month: **December**

My Achievements:

This month, was all about the mid-point presentation and preparing myself to get everything right to get the highest grade possible. I started off by fully polishing off my prototype and making sure I was ready to present and that my prototype was free from errors. By double checking my prototype earlier and finishing it completely it enabled me to add in any extra features that would make my prototype different from the other students presenting a prototype.

I had to create my PowerPoint presentation for the mid-point presentation and I had to ensure I included all the required requirements. Also, I had to focus on my technical report which was required for my mid-point presentation. I had to ensure that my technical report was of a high standard because I want to achieve the highest grade possible.

My Reflection:

I felt, it worked well to meet regularly with my supervisor and ask him for all the advice that he had for me regarding my project. I felt that listening to his advice is very important and every student should listen to their supervisor as they are only trying to help you achieve a high grade which I am very grateful for. I also felt that it was very important to communicate with other students in your class or close college friends to exchange advice between each other and gain ideas from each other.

Intended Changes:

Next month, I must focus on exams as we are in the examination period for semester one. After I finish my exams I will start working on the final application and plan what I must do this semester to successfully complete this project to a high standard.

Supervisor Meetings:

Date of Meeting: 8/12/2016

Items discussed: Prototype recommendations, technical report feedback and recommendations

Actions:

- Add in recommended features into prototype.
- Change given sections in technical report.

January

Student name: **Jelico Bachin**

Programme: **BSc in Computing, Cyber Security Stream**

Month: **January**

My Achievements:

This month, was all about studying for exams hoping to boost up my overall average which has worked out well for me. There was a lot of stress for me this month as I had to study for 4 exams and because 3 were one after the other this made it even harder for me. I was always tempted to continue working on my project but I knew exams were the priority for now. Straight after the exams were finished I continued with my project. I started the implementation process which led me on a very dark road. I discovered the capabilities of the swift language and thus I had to completely change the implementation of my project. I now must create a website and then make an iOS application with multiple controllers and web views for the application to work and accept bookings and payments.

My Reflection:

I felt, that achieving good grades in my exam results boosted my confidence. I was happy with my mid-point presentation grade as it showed that I have been putting in the effort over the past few months. The grade also boosted my confidence knowing I can achieve this again in the showcase presentation. I am happy with my project and I hope to continue enjoying the next couple of weeks of my final semester.

Intended Changes:

Next month, I will hope to have 60% of the implementation complete to give me time to test features and test if they are working as expected. I will meet with my supervisor more often as I felt this was key to success last semester.

Supervisor Meetings:

Date of Meeting: 6/02/17

Items discussed: Project language change

Actions:

- Continue with implementation and development

February

Student name: **Jelico Bachin**

Programme: **BSc in Computing, Cyber Security Stream**

Month: **February**

My Achievements:

This month the focus was implementing my details section of the project which I successfully managed to do. Implementing other features was in my timeline but I have started doing assignments and studying for other modules also. There was a bit of pressure and stress on me this month for some private reasons which affected my work rate especially towards the project. I am feeling confident and I am sure I can recover the time lost over the next couple of months if I keep the head down and stay focused.

My Reflection:

I felt that making a timeline plan of that features are to be implemented at a certain time you will always stay on track. It is very important to stay motivated and focused as this is will help achieve the end goal which is completing the project and getting a high mark.

Intended Changes:

Next month I hope to complete most of the feature implementation to give me more time to focus on documents and touch ups but also implementing additional features.

March

Student name: **Jelico Bachin**

Programme: **BSc in Computing, Cyber Security Stream**

Month: **February**

My Achievements:

This month the focus was maintaining a high work rate and balancing the rest of the modules assignments together with the main project. I managed to complete a lot of tasks that I had set out for myself and this helped me conclude that my progress was better than I expected. I am happy with the direction my project is heading towards and I am excited to go over the final hurdle in this course.

My Reflection:

I felt that balancing the workload into tasks helped me realize how well the development stage of my project is heading towards. I will focus on getting the project finished by completing the testing and the documents that are required. This semester was very enjoyable compared to last and I had no problem getting out of bed and going to lectures.

Intended Changes:

Next month I hope to complete the project and the rest of its deliverables. I shall conduct the testing required and complete the documents and submit the project. I also look forward to the project showcase.