Graded Unit: Development Stage Report

**New College Lanarkshire**

DANAYAL IFTIKHAR

Table of Contents

[Project Brief (Moodle, n.d.) 2](#_Toc390360863)

[Project Solution 2](#_Toc390360864)

[Review of Interview 1 (Moodle, n.d.) 2](#_Toc390360865)

[Review of Interview 2 2](#_Toc390360866)

[Functional & Non-functional Requirements 3](#_Toc390360867)

[Functional requirements 3](#_Toc390360868)

[Non-functional requirements 3](#_Toc390360869)

[Requirements Evaluation 4](#_Toc390360870)

[Non-functional requirements 4](#_Toc390360871)

[Top Level Use Case Diagram 5](#_Toc390360872)

[Class Diagram 6](#_Toc390360873)

[Admin Activity Diagram 6](#_Toc390360874)

[Customer Activity Diagram 7](#_Toc390360875)

[Login Sequence Diagram 7](#_Toc390360876)

[Booking Sequence Diagram 8](#_Toc390360877)

[Project Management 8](#_Toc390360878)

[Milestones and Resources 8](#_Toc390360879)

[GUI 9](#_Toc390360880)

[Testing 11](#_Toc390360881)

[Test Strategies 12](#_Toc390360882)

[Black Box Testing 12](#_Toc390360883)

[White Box Testing 13](#_Toc390360884)

[References (Moodle, n.d.) 14](#_Toc390360885)

[Bibliography 14](#_Toc390360886)

## Project Brief (Moodle, n.d.)

Simple Car and Van Hire wanted a computerised system that will cut the time taken for a customer to hire a vehicle and increase income. The system should be able to check the availability of vehicles, link to a payment system and ensure cars are allocated to hires taken in by sales staff.

## Project Solution

There will be a web based system that will allow customers to hire vehicles with a linked payment system that should work on any platform and allow sales staff to allocate logbooks and documentation to the customer.

## Review of Interview 1 (Moodle, n.d.)

Prepping for the interview consisted of coming up with a few questions to ask the client about what functions were to be included in the system. During the interview a number of points and information was obtained. The interview allowed me to obtain the initial functional and non-functional requirements which would be used in the initial use case model for the system. This would then be further set into the initial class model to outline the design of the system.

## Review of Interview 2

During the second interview with the client, I was asked to add to the separate the bookings part of the program into “add booking” between a car hire and a van hire. I was also asked to be able to register a customer and expand on “View booking” and “Return Booking.” I needed to develop the system in order for it to show the different types of vehicles that can be hired; these were classed into groups and either Vans or Cars.

# Functional & Non-functional Requirements

## Functional requirements

**Add Booking**: Must be able to make a new booking

**Return Booking**: Must be able to return a vehicle to stock

**Cancel Booking**: Must allow a sales manager to cancel a booking

**Allocate Logbook**: Must allow for a logbook to be created for each hire taken out

**View Bookings**: Allows sales staff to view if a vehicle has been loaned

**Search Booking:** Allows a sales staff to search for an existing booking for a specific hire

**View Vehicles**: sales manager must be able to view amount of vehicles remaining of each group

**Register customer**: Sales manager can allow a customer to register

**Revoke registration**: Sales manager can take a customers’ registration away

**Process payment**: Payment clerk will process the customers’ payment from a vehicle hire

## Non-functional requirements

System must use current software

System has to be compatible on any operating system/platform

System must be web-based

System needs to be complete for June 2014

System needs to be easy to use

# Requirements Evaluation

**Add Booking:** The user must be able to make a new booking to hire a vehicle and fill in the details on the form to do this

**Return Booking**: The sales manager must be able to return a returned vehicle to the stock in order for the vehicle to become available again

**Cancel Booking**: The sales manager should be able to cancel a booking while a vehicle is still hired at any time and make it null

**Allocate Logbook**: The sales staff should be able to allocate a new logbook each time a new hire Is made with the users and vehicles details

**View Bookings**: Should allow the sales staff to be able to view existing bookings that have already been taken out

**Search Booking:** Should let the sales staff to search for a specific booking and display its details and who loaned it by clicking “search” in the AdminWindow and typing in the details

**View Vehicles**: sales manager must be able to view amount of vehicles remaining in the database and from which group they derive from by clicking “View Vehicles” in the AdminWindow

**Register customer**: Sales manager should be able to authorise a new customer to register to the system

**Revoke registration**: Sales manager should be able to withdraw a customer’s account at any time

**Process payment**: Payment clerk should be able to go through payment details with the user and authorise the hire and payment to go through the system

## Non-functional requirements

**Current Software:** The system should be able to correctly run on the already installed software on the company machines

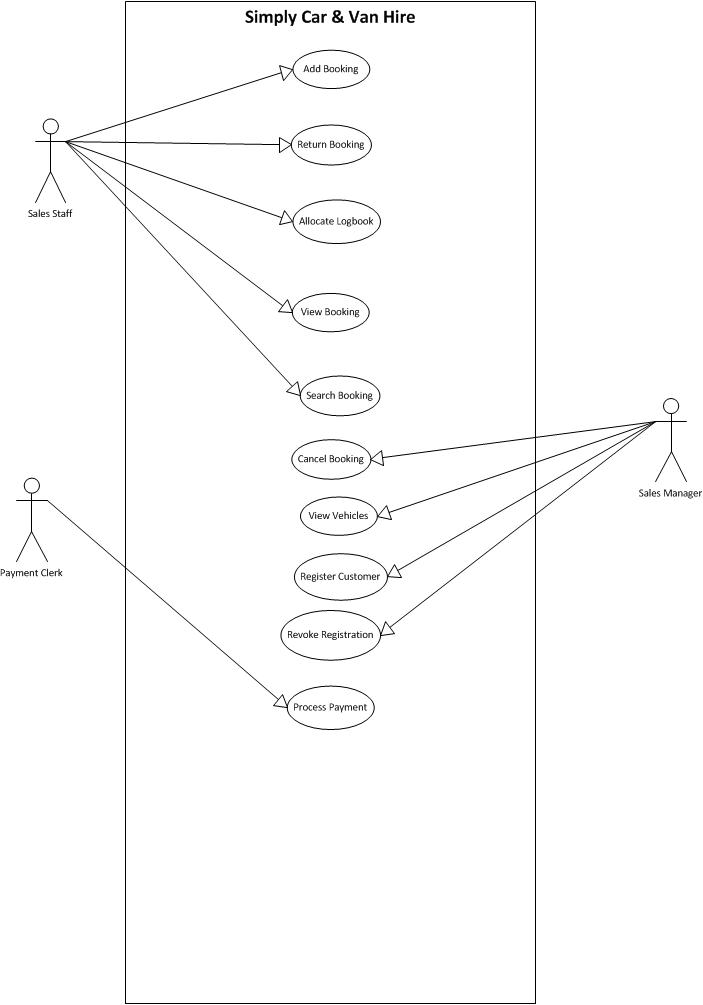
**Any platform:** The system must work on several platforms including Mac OSX and Windows based systems

**Web based:** The system must be able to work as a web page

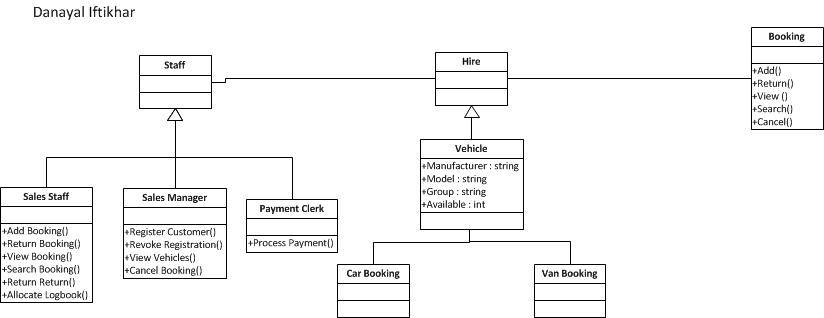
**Completed for June:** The system must be up and running by the 27th of June and tested

**Easy to use:** The system GUI must be user friendly and easy to comprehend and navigate. It should also make it easier for manager to teach new staff how it works by simplifying the system down.

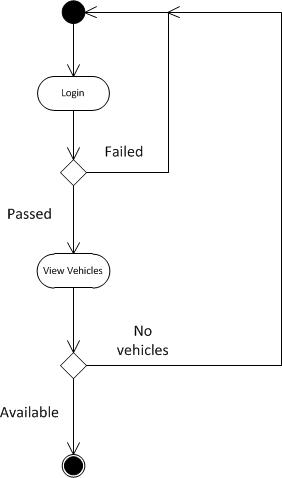
# Top Level Use Case Diagram



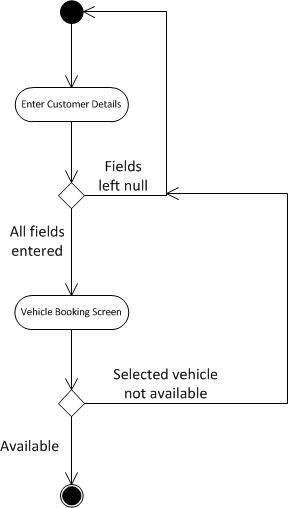
# Class Diagram



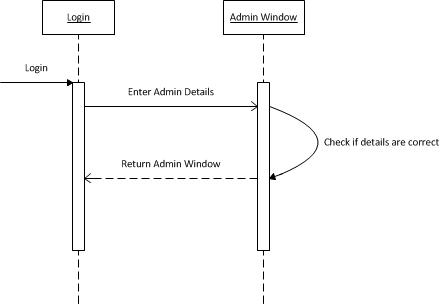
# Admin Activity Diagram



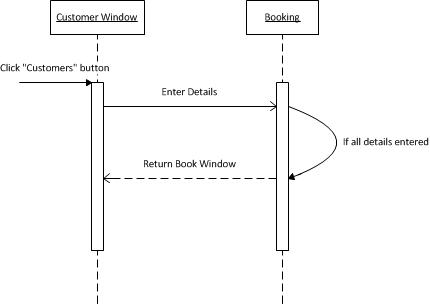
## Customer Activity Diagram



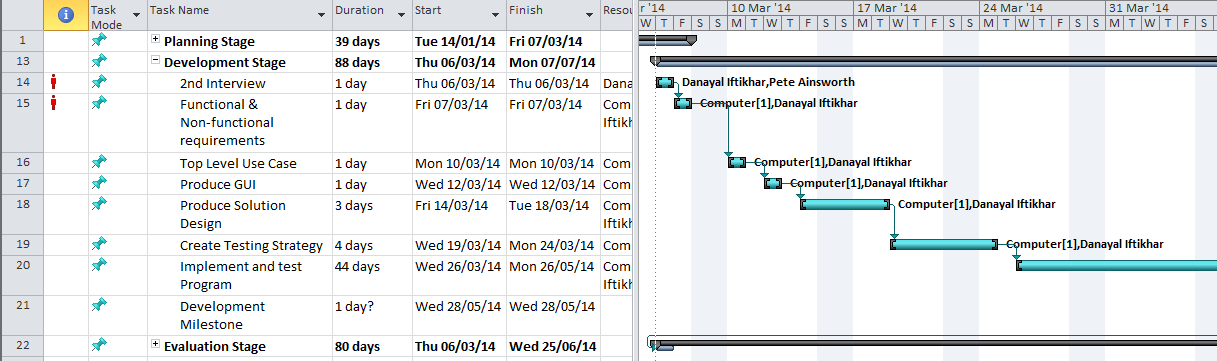
## Login Sequence Diagram



## Booking Sequence Diagram



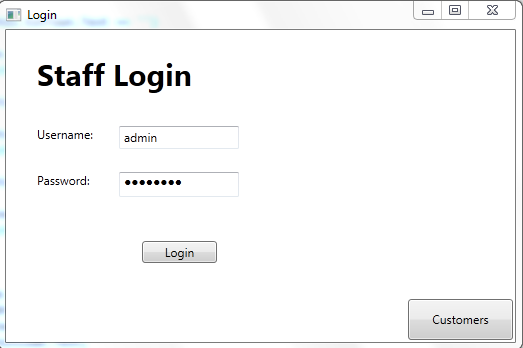
# Project Management



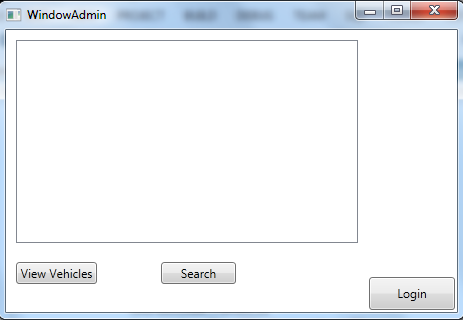
## Milestones and Resources

Several milestones ware identified within the project management for the development of the system. The development milestone was achieved as the interview was done and the client discussed the expansion of certain uses of the system that were to be established. The use case and GUI were fully developed to the specifications of the client. At the end of the development the program was tested with various actions and some errors were found and fixed. At the end the development milestone was met using the resources (Computer) and the developer (Danayal Iftikhar).

# GUI

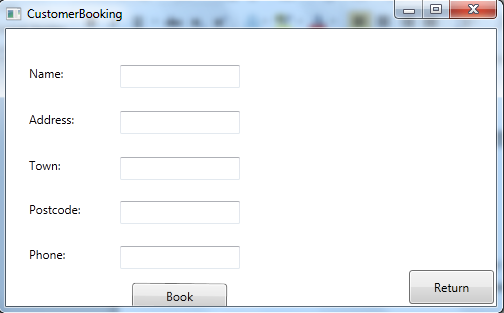
1. Enter the correct admin details

**(Fig.1)**

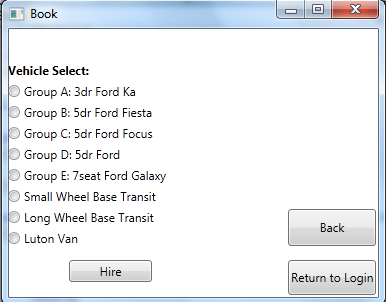
1. The following window will appear for the admin:

From here you can view the amount of vehicles remaining in each group by clicking “View Vehicles” or search for a particular booking that already exists by clicking “Search”

By clicking the “Customers” button on the Login screen (see fig. 1) a customer can go to the booking screen and hire a car by filling in the following details:



From here the customer may make a booking by entering their details and clicking “Book” which will navigate them to the Book screen.



Select your vehicle and click “Hire” to complete the purchase, if your selected vehicle is not available then a message will appear saying “Selected vehicle not available.”

# Testing

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No. | Input | Expected Output | Output | Result |
| 1 | Input no details in the login screen | Messagebox saying “Enter both a username and password” | Messagebox saying “Enter both a username and password” | Pass |
| 2 | Input just a username and no password in the login screen | Messagebox saying “Enter both a password” | Messagebox saying “Enter both a password” | Pass |
| 3 | Input just a password in the login screen | Messagebox saying “Enter both a username” | Messagebox saying “Enter both a username” | Pass |
| 4 | Input correct details on Login screen | Navigates to Admin Window | Navigates to Admin Window | Pass |
| 5 | Click “Customers” on Login screen | Navigates to Booking Window | Navigates to Booking Window | Pass |
| 6 | Click “View Vehicles” in the admin window | Display all available vehicles in listbox |  |  |
| 7 | Click “View Vehicles” in the admin window when no vehicles remaining | Messagebox saying “No vehicles remaining” | Messagebox saying “No vehicles remaining” | Pass |
| 7 | Click “Search” in admin window | Navigates to search window |  |  |
| 8 | Enter everything but a name in the Booking window | Messagebox saying “Please enter a name” | Messagebox saying “Please enter a name” | Pass |
| 9 | Enter everything but an address in the Booking window | Messagebox saying “Please enter an address” | No messagebox | Fail |
| 10 | Enter everything but a Town in the Booking window | Messagebox saying “Please enter a town” | No messagebox | Fail |
| 11 | Enter everything but a Postcode in the Booking window | Messagebox saying “Please enter a Postcode” | No messagebox | Fail |
| 12 | Enter everything but a phone in the Booking window | Messagebox saying “Please enter a Phone number” | No messagebox | Fail |
| 13 | Click “Book” in booking window | Navigates to Book screen | Navigates to Book screen | Pass |
| 14 | Choose option then click “Hire” in Book Vehicle window | Messagebox saying “Selected vehicle hired” | No response | Fail |
| 15 | Choose option then click “Hire” when selected vehicle not available | Messagebox saying “Selected vehicle not available for hire” | No action | Fail |

# Test Strategies

## Black Box Testing

Black box testing is a method used when the user does not know the internal functions of the system itself. This can include functional and non-functional requirement testing. It is mainly used to find out errors in the system, interface design flaws and database errors.

Black box testing can be implemented in all levels of software testing which are: Unit Testing, Integration Testing, System Testing and Acceptance Testing.

**Advantages:**

* Testing is done by a user who knows nothing about the app in order to gain a user’s perspective.
* Tester does not need to know anything about programming or how the system was created
* Testing can be done by someone outside of the developer team or company, creating a different perspective of the system and cutting developer bias.
* Test cases can be made as soon as the specifications are complete

**Disadvantages:**

* Only a limited amount of input can be tested leaving many possible inputs paths untested
* As there cannot be clear specifications for the system, most of the test cases will be difficult to create in the first place
* Tests can be left or scrapped if a developer has already run a test case
* As the user does not know the internals of the system, they cannot point out any errors they can catch out and suggest how to fix it if the developer has missed something out

## White Box Testing

White box testing is a method used when the internal functions and structure of the system are known to the tester. In this method the tester chooses inputs to use which are tested within the code to create an output that they determine. The tester must also have a good knowledge in programming and development, which is beyond the level of standard black box testing.

White box testing can be implemented into several levels of software testing which include: Unit Testing, Integration Testing and System Testing.

**Advantages:**

* Testing can be done earlier in the development stage as the tester will know how the system works
* Testing covers more possible paths and inputs, leaving less gaps and errors

**Disadvantages:**

* Testing becomes very difficult which means a lot of skills and resources will be required from the developer
* Test script maintenance can be a difficult job if the implementation changes too quickly
* This method of testing is more close to the application being in its earlier stages, which means that a lot of tools for implementation might not be available at the time.

# References (Moodle, n.d.)

# Bibliography

Moodle, P. A., n.d. *DH3C 35 Detailed Scenarios.* [Online]   
Available at: http://www.vle.motherwell.co.uk/mod/folder/view.php?id=37539

Moodle, P. A., n.d. *Interview Times.* [Online]   
Available at: http://www.vle.motherwell.co.uk/course/view.php?id=402

Moodle, P. A., n.d. *Reference Help.* [Online]   
Available at: http://www.vle.motherwell.co.uk/course/view.php?id=402