Sprint 3

1. A clear and understandable use of "**Web Services**" is requested that be incorporated into the system - merely for demonstration purposes. This can be for calculation of the charges per journey.
2. The distance between two addresses can be calculated via **Google map services**, which can be used in a Web Service that calculates the charge of taxi services*.*
3. It can also be useful for the system to use "**Filtering**" for admin user authentication purposes- a clear example of this would also be favourably considered (“**Filtering**” is discussed in Chapter 13 of reference book, HF Servlet & JSP).

Marks for sprint 3 are allocated as follows:

## Step 1: The Test Cases.

1. Login to the system as a typical user (e.g. user = "admin", password= "admin") and create a Session lasting for 20 minutes.
2. List all drivers and all customers registered for the head office
3. Show the list of jobs for each driver upon request
4. Create a daily report (including the turnover, and the number of customers served)
5. List all customers served per day including the destinations and the charge incurred.
6. Let a customer book a cab for a particular date and time - (if it is the 1st time, the office should add a new customer record, the office should assign the customer to an available driver).
7. Add more drivers to the list
8. Create a customer invoice - showing incurred cost and VAT for a total cost.
9. Remove a driver
10. Change the price of a destination

Each successful test case is worth 4 marks (total 40).

Each test must be demonstrated sequentially and as described. Marks could be lost – e.g. for a lack of clarity or failure to demonstrate tests sequentially. Do not combine the Test Cases - each must be a distinct operation. Normally a test will either pass or fail.

If you think there may be some ambiguity in the specification or Test Cases you must check for clarification with the tutor in your normal practical class. False assumptions may cost you marks.

## Step 2: Design Walk Through.

You should briefly and concisely explain the design and implementation of the following features of your system (45 marks)

* The MVC pattern and project file structure (15 marks)
* The relational database (10 marks)
* Sessions (5 marks)
* Web Services (develop and use) (10 marks)
* Using Filters (5 marks)

## Step 3: Team Work Evidence.

Your team work will be assessed based on the evidence you provide via GitHub version control software and other team-based software development tools. (15 marks)

* Evidence of each group member’s contribution (5 marks)
* Task management and allocation (5 marks)
* Clear history of collaboration (5 marks)

Marking Criteria:

MVC pattern

* Comfortable and optimal use of MVC patters, clear split of the roles over M, V and C.
* Clear explanations of the terms, and concepts with all limits and opportunities.

Database and SQL operations

* Comfortable and optimal use of DB operations, well-embedded in model classes, and successfully displaying the results via viewers
* Connection is managed once in application scope and used ahead
* Clear understanding of wider context of the knowledge with all relevant limits and implications

Sessions

* Comfortable and optimal use of sessions, successfully activated/inactivated on change of users
* Comfortably using the sessions scope for attributes
* Clear understanding of the knowledge and technology

Developing and using WS

* Comfortable development, deployment and use of WS for the aimed purpose
* Comfortable use of cloud service
* Very good understanding of the terms and concepts
* Clear awareness of the limits and implications

Use of Filters

* Comfortable use of Filters for authentication purpose
* Comfortable knowledge demonstrated for Filters
* Clear awareness of the limits and implications

Teamwork Evidence

* Excellent records of individual contribution
* Tasks are clearly identified, allocated and workload balanced
* Excellent history of collaboration
* Excellent communication