Student Details (Student should fill the content)				
Name				
Student ID				
Scheduled unit details				
Unit code	CIS6003			
Unit title	Advanced Programming			
Unit enrolment details	Year	3		
	Study period			
Lecturer				
Mode of delivery	Full Time			
Assignment Details				
Nature of the Assessment	**Course work**			
Topic of the Case Study	**Online cab service system**			
Learning Outcomes covered	**1,2,3**			
Word count	4000			
Due date / Time	30th August 2022			
Extension granted?	Yes	No	Extension Date	
Is this a resubmission?	Yes	No	Resubmission Date	

Declaration			
I certify that the attached material is my original work. No other person's work or ideas have been used without acknowledgement. Except where I have clearly stated that I have used some of this material elsewhere, I have not presented it for examination / assessment in any other course or unit at this or any other institution			
Name/Signature		Date	
Submission			
Return to:			

Result				
Marks by 1st Assessor		Signature of the 1st Assessor		**Agreed Mark**
Marks by2nd Assessor		Signature of the 2nd Assessor		
Comments on the Agreed Mark.				

For Office use only (hard copy assignments)			
Receipt date		Received by	

STUDENT NAME:	**STUDENT NUMBER:**
Module Number & Title:	**Semester:**
Assignment Type & Title:	
For student use: ***Critical feedback on the individual progression towards achieving the assignment outcomes***	
<u>For the Assessors' feedback</u> **Indicate the Task number strength and Weaknesses and the marks for each task**	
Task No/Question No	**Strengths**
Task No / Question No	**Weaknesses**
Areas for future improvement	
Marks	

Task /Question No	Allocated Marks	Awarded Marks	Remarks
Total Marks			
Name and the Signature of the Assessor			
Date			

Upon successful completion of this module, you will have demonstrated:

- Demonstrate fluency in contemporary programming languages, development tools and environments.
- Evaluate and demonstrate the theory and concepts of contemporary/industry standard programming and design in the software development life cycle.
- Demonstrate awareness of industry standards of professional and ethical software development, software carpentry and codemanship.

Coursework –**Online cab service system**

"GoCheeta" is a well reputed cab service located in Sri Lanka Island wide. It has several branches like Galle, Kandy, Nugegoda, Gampaha, Kurunegala and Jaffna). It has been delivering its services in the mode of a mobile app, and now they want to implement it as in a website either due to the requirement identified at several points.

The customers can get registered first to the system. Then a registered customer can order any type of vehicle available via the website. But once the booking is made, both the **pick-up and destination locations will be restricted only within one exact city, that they choose out of the given list(predefined as per only where branches are located in)**.

There are registered drivers in the system where the customer cannot pick one but depending on the availability of vehicles, the driver associated will be selected and be informed by the system itself.

Customer

Customers can register to the system themselves, search for a respective category of vehicle (you can assume the types of vehicles by your own) in the system. Once they search for a vehicle, the available list of vehicles will be prompted and out of that they can pick up any vehicle available in. They have to choose the source and destination as per in two drop down fields as **Street address** and **city.** They can also view the price for the journey depending on a distance-price scenario which is free to be assumed by you (developer/student). But this price-distance scenario should be reflected and transparent enough in both your documentation and coding. They can view the all the bookings they have made so far along with the details and can leave a feedback once the journey is completed about their entire experience on the tour.

Admin

Admin can add new drivers, vehicles and categories of vehicles to the systems plus having them update, remove and view the details, view all the customer bookings (along with customer details, vehicle and driver details associated with the booking) of all the branches, and view the total sales made by each branch and the total of all the branches at any given time.

Booking of vehicles

Once a particular booking is accepted by the driver, it should be updated with the status of journey (booking) to the customer including pick up and when dropping by.

Drivers

Drivers are all registered in a specific branch and associated with a specific set of vehicles already.

Vehicles

Vehicles are associated exactly with any identified category and it should be registered under a specific branch and should be associated with specific set/ one exact driver

Note:

- *Admins of the system are the officials, specifically appointed by the main branch.*
- *The payments for the customer orders can only be made as cash on delivery.*
- *Once a booking is made, there is no way of cancelling it through the system(any alternative way can be provided like sharing the phone number of drivers through the system and customer can make a call and cancel the booking)*

Provide a well-designed, user friendly system addressing the following features:

- System should have differential access rights to the system users.
- Interactive user friendly interfaces
- Clear implementation of the business flow via the system.
- Design & implement suitable sets of reports, which you think will add more value to the entire business
- Use test driven development and include test classes to test your application

Students are free to make necessary assumption on system design & granting access permissions other than mentioned within the scenario, but all suggestions must be well explained with the Valid reasons.
Students can add any functionality which will enhance the system and make the proposed Solution more comprehensive.

Use Harvard referencing to properly acknowledge all the external sources you use.
Your tasks

Tasks A:
Provide a requirement specification for the proposed system. (06 marks)

Tasks B:
Provide the UML diagrams for the given problem with clear explanations on the design decisions. Derive detailed Use Case diagram, Class diagram & a sequence diagram. Whenever necessary document the relevant assumptions you made.

(09 marks)

Tasks C:
There are many system design patterns available in system development. Critically evaluate singleton, factory and abstract factory design patterns and apply the most suitable design pattern for your system development.

(15 marks)

Tasks D:
Develop an interactive set of interfaces to get the necessary user inputs. Make sure to implement proper validation mechanisms in order to restrict invalid entries to the system. Come up with suitable set of reports, which you will think add more value to your system

i. Your program must be a distributed application with web services

ii. Your program should make use of a proper database to store information

(30 marks)

Tasks E:
Document the test plan and explain how you used test driven development in this scenario and do a test automation to achieve that. This includes test rationale, test plan, test data and proper application of the test plan (LO II)

(15 marks)

Tasks F:
Create user and technical documentation for the developed solution. (10 marks)

Tasks G:
Create your own Git/ GitHub repository which is public to access and upload /deploy the changes of the software project you have developed in it. Share the repo link within the documentation. Update it with several number of versions where modifications applied per each day, that you have applied the new features into which was initially uploaded with. Version control techniques you have used throughout the development should be highlighted and documented properly. Demonstrate workflows deployed with the Git repository.

(15 marks)

Guidelines for the report format
Paper A4 | Margins 1.5" left, 1" right, top and bottom
Page numbers – bottom, right | Line spacing 1.5
Font size
Headings 14pt, Bold | Normal 12pt
Font face- Times New Roman
Referencing and in-text citation should be done strictly using **Harvard Referencing System.**

Marking Scheme

Task (A) contains 6 marks

Criteria	Marks
	Out of 6
Functional requirement specification	3
Nonfunctional requirement specification	3

Task (B) contains 9 marks

Diagrams should be evaluated according to the following criteria.

Criteria	Marks
	Out of 9
Proper use of Object Oriented Design Methodology Use case Diagram · Identification of correct use cases · Identification of correct Actors and associations Sequence Diagram · Implementing identified set of use cases (about 3) as sequence diagrams. Class Diagram · Identification of associated methods, with correct signatures and attributes in each class · Correct identification of relationships	1-3

Average Design · Clear identification of private, public access modifiers & it is visible in the class diagram · Accurate use of <<include> <<extend>> stereo types in use case diagram · Appropriate use of lifelines, messages and objects in proposed sequence diagrams · Correct use of UML notations with minor mistakes Evaluation · Student has given basic description about the design and given a reasonable justification · Effective judgements have been made	4-6
Excellent Design · Highly detailed diagram · Use of OO concepts clearly visible · Backed by relevant assumptions · Multiplicity, navigability aggregation & compositions visible in class diagrams · Excellent use of UML notation Evaluation · Good justification of the design · Judge validity of results · Use critical reflection to evaluate the work and justify with valid explanations Fluency (Of design) · Evidence of critical analysis on different perspectives covering how, use case, class & sequence diagrams support in designing	7-9

Task (C) contains 15 marks

Criteria	Marks Out of 15
Identify the different types of design patterns and there advantages	1-5
Apply the suitable design patterns for system development	6-10
Critically evaluate the impact of design patterns	11-15

Task (D) contains 30 marks

Criteria	Marks Out of 30
Pass · Basic data management system features . · Use a database (simple design) · Have simple Web user interface	0-8
Good · Make a good attempt to follow the three tier architecture. · More sophisticated database design and queries · More sophisticated data representation (e.g. several classes at business logic level) · Separate UI windows for entering results and viewing overall scores.	9-20
Excellent · More sophisticated UI, · Complex functionality (Email alerts/SMS/Innovative aspects) · 3- tier architecture should exit · Appropriate use of more sophisticated database features (e.g. use of stored procedures / functions / triggers to implement business rules) · Reports being proposed to facilitate decision making. · Effective use of sessions / cookies	21-30

Task (E) contains **(15 marks)**

Test Rationale **(5 marks)**

- Provide a concise rationale for the approach adopted. Discuss how you are going to use test driven development.

Devise your test data **(5 marks)**

- Derive test data for the system.

Produce and apply a test plan **(5 marks)**

- Create test classes for your system
- You are to carry out relevant tests and provide documentation detailing the tests used to verify your system**.**
- Demonstrate that the code passes all the tests (use screen-grabbing software and insert images into your submission).
- Use of test automation
- Evaluation of overall success or failure and lessons learned.
- Traceability showing how each requirement is met by the design.

Task (F) contains 10 marks

Criteria	Marks Out of 10
Errors in the documentation	0-3
Acceptable standard of documentation with poor explanations	3-5
High standard of documentation with screen shots & average explanations	5-7
Professional standard of documentation with screen shots & good explanation	5-10

Task (G) contains 15 marks

Criteria	Marks Out of 15
Poor/no any Git version control, deployment, workflows used demonstrated	0-3
Git Repo/GitHub is used in creating a repository only and uploaded the initial version only	3-6
Git repo created, initial version of project uploaded and several versions were updated and deployed with changes but no any workflow or version control technique is demonstrated.	6-10
Git repo creation, accessibility restrictions, versioning and version control techniques, workflow(CI/CD) demonstrated and deployment of changes and the latest updated version is done and demonstrated in documentation	10-15

Final Grading criteria for the coursework:

Marks	Final Grade
>=70	1
69-60	2:1
59-50	2:2
49-40	3
<40	Fail

****