



University of
Bedfordshire

ASSIGNMENT TOP SHEET
Faculty of Creative Arts, Technologies & Science
Department of Computer Science and Technology

Student Ref. No

Unit Code

CIS020-1 / CIS093-1

Unit Name

**Introduction to Software Development
Mathematics and Concepts for
Computational Thinking**

Deadline for Submission(s)

**Friday 13 January, 2023
10.00am (TBC)**

Student's Surname

Student's Forename

Unit Team

Susan Brandreth

Assignment Details:

**Assignment 2 – Group or Individual Project – Case Study – Taxi Booking
System**

Instructions to Student:

Please note:

Work presented in an assessment must be your own. Plagiarism is where a student copies work from another source, published or unpublished (including the work of another student) and fails to acknowledge the influence of another's work or to attribute quotes to the author. Plagiarism is an academic offence and the penalty can be serious. The University's policies relating to Plagiarism can be found in the regulations through the SID website. To detect possible plagiarism we may submit your work to the national plagiarism detection facility. This searches the Internet and an extensive database of reference material including other students' work to identify. Once your work has been submitted to the detection service it will be stored electronically in a database and compared against work submitted from this and other universities. It will therefore be necessary to take electronic copies of your materials for transmission, storage and comparison purposes and for the operational back-up process. This material will be stored in this manner indefinitely.

I have read the above information and I confirm that this work is my own and that it may be processed and stored in the manner described.

Signature (Print Name):

Date:

Extension deadline

SID agrees that the assignment may be submitted ____ days after the deadline and should be marked without penalty.

SID confirmation.....

<p style="text-align: center;">CIS020-1 - Introduction to Software Development CIS093-1 – Mathematics and Concepts for Computational Thinking Assignment 2 – Group or Individual Project</p>

Scenario

Taxi Booking System

Consider the following scenario:

A taxi company would like to provide an online service for customers to book a trip and for taxi drivers to check their upcoming trips. Data needs to be stored in an external file – either a text file or database. The following data, **at least**, would need to be stored and be retrievable - customer data (name, address, email, telephone number, payment method) and trip information (pick up and drop off address, and pickup time and date). Furthermore, you will need to store driver information (name and license plate). Each trip must be assigned to exactly one driver (and trips allocated to a driver must not overlap).

Customers who are not registered yet need to register - providing their name, address, phone number, email address and method of payment. A registered customer should be able to book a trip, ie. provide the pickup address and time as well as the drop off address – preferably via a graphical user interface. Customers should also be able to view their trips and cancel them if needed.

All these operations are only possible after a customer signs in (which means they must be registered). Each new trip booking must be confirmed by the taxi company administrator who must also assign a driver to the booking. Only one driver must be assigned to each confirmed booking. After logging in, taxi drivers must be able to view a list of their upcoming trips.

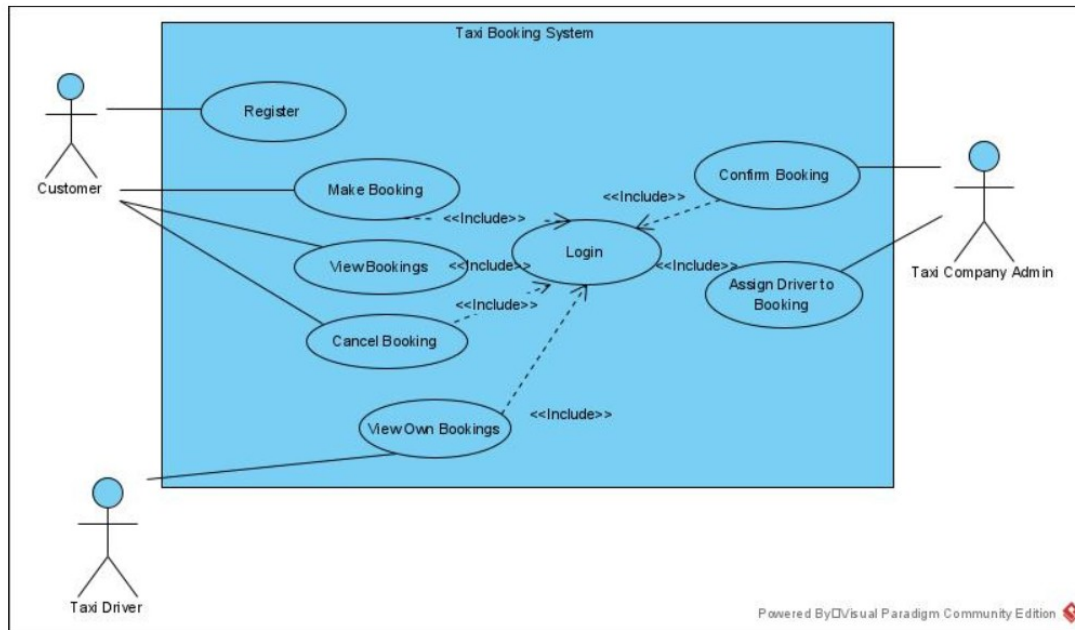


Figure 1: Taxi Booking System Use Case Diagram

Your task is to implement the [Taxi Booking System](#).

The functionality described above is shown in the [UML Use Case Diagram](#) in [Figure 1](#), taking the different user roles (customer, taxi company and taxi driver) into account.

This assignment can be conducted either as an individual assignment or group assignment. Please note in any case it is an individual assessment, which means you will get an individual grade based either on your individual project or your contribution to the group project.

In case of group work, the group needs to attempt to implement the whole system and individual contributions have to be made clear in the group report.

If you choose to conduct the project as individual assignment, you are asked only to implement the system from Customer Use Case(s) point of view.

You MUST state clearly in your submission that this is INDIVIDUAL WORK

FOR CIS020-1

The following [Learning Outcomes](#) will be addressed in this assignment:

- 1 Demonstrate the following knowledge and understanding
 - Knowledge of syntax and semantics of a high level modelling and programming language within the context in which software development takes place.
- 2 Demonstrate the following skills and abilities
 - Develop software based on an accurate model using a high level language following the stages of analysis, design, and implementation and testing

FOR CIS093-1

The following Learning Outcomes will be addressed in this assignment:

- 1 Demonstrate the following knowledge and understanding
 - Comprehend the mathematics that underpins computing problems and the techniques of decomposition, pattern recognition, abstraction and algorithms that are used in programming.
- 2 Demonstrate the following skills and abilities
 - Develop algorithms and software based on computational thinking using a high-level language following the stages of analysis, design, implementation and testing.

Carefully read the '[Taxi Booking System](#)' scenario.

You must investigate the scenario provided and based on your assumptions outline a clear vision of what the system will do and for whom.

What are the different roles for the users interacting with the system? Which functions of the system does each user use, or interact with, and what information needs to be recorded about each of them? What are the functions the system must provide? How will these functions be accessed and organized?

Your task for a group or individual project is as follows.

1. Select critical features of the system and describe them using all three of the following techniques:
 - Use Case Diagrams (Including a discussion of Use Case scenarios and include at least 2 levels of description)
 - Activity Diagrams
 - Class Diagrams
2. Prior to implementation you should develop algorithms for this application. The algorithms can be presented either visually or in text format. [CIS093-1 only]
3. Use Python to code the critical features of the system you have analysed and perform some testing to show that your system works. As well as basic language features, you will be credited by the use of GUIs, File Access and good design techniques.
- 4.

Requirements

You should list the functional, non-functional and usability requirements of the system.

Design

Use Case Diagrams

Sea Level Use Cases

Using the techniques you have learned, you should identify system actors, the most important processes and services the system must provide and the relationships between them.

Scenarios

You should identify and illustrate the most important scenarios or stories about how the system is used. These should include descriptions of how the system is being used for each Use Case.

Fish Level Use Cases

You should identify the lower level processes which support the system at sea level, including the relationships between them and the actors using the system.

Activity Diagrams

You should use Activity Diagrams to describe the steps involved in the tasks and processes you have identified for the system.

Class Diagrams (Optional)

You should use Class Diagrams to describe the structure you have identified for the system only if you use an object-oriented approach.

Database Design

User Interface Design

Implementation

Having thought about how a large scale system would work you should implement a small scale prototype using Python.

Your system should implement the key functions you have identified. You are **not required** to implement an on-line server based system but you should implement a graphical user interface (GUI) if working as a group. However, if working as an individual, you are not required to implement a GUI interface.

If working as an individual, you may use text-based menus similar to the ones you used in lab exercises.

For example:

Login
Search for tools
Create new account
Exit
Enter your choice:

or, you may design Graphical User Interface (GUI) menus if you wish.

Your Python program should demonstrate the features you have identified. Bonus marks will be given for enhanced features in your implementation.

Interface

Your program should be implemented using the high level programming language, Python, and should incorporate a range of Graphical User Interface (GUI) elements in the design (but see above if working as an individual).

Functional logic

You should also write code to implement a version of the basic functions that the system provides using the techniques you have learned - including appropriate data structures and techniques and extending these if you wish.

You are not required to implement a client-server model with network connection.

Report

The report should include the following:

Title Page
Table of Contents

List of Group Members

Introduction/Overview

Why are you doing this work? What you were asked to do? What you did?
Were you successful?

Task Description

This is basically a description of the task as on the Assignment brief. You can paraphrase what is written on the Assignment 2 brief.

Include any assumptions you make about the system.

Project Plan/Schedule

Overview of Functional, Technical (Non-Functional Requirements) and Usability Requirements

Design

- UML Diagrams
 - Uses Case Diagram(s)
 - Activity Diagram(s)
 - Class Diagram(s)
- Database Design
 - Entity Relationship Model (ERM)
 - ERM Diagram
 - Skeleton Tables
 - Data Dictionary
Example is on BREO
- User Interface Design
Sketches of your GUI interfaces

Implementation

You should discuss YOUR application. How was the program created?

Explain how you managed the implementation. How did you ensure all group members actively contributed to writing code? How did you distribute the workload? Was it successful?

What Integrated Development Environment (IDE) did you use? Justify your choice of IDE, Python Framework (if used), packages / libraries.

Did you embed SQL in your client application or did you implement a Client-Server (Sockets Programming architecture. Justify your chosen method, and discuss the advantages and disadvantages of both.

What problems did you encounter during the coding, and explain how you overcame these?

Detailed discussion of your experience coding is required here.

Did you incorporate any novel approaches to coding and meeting the Assignment requirements?

Testing

Include evidence of detailed and robust testing.

There are various ways to document this...

Test Plan / Test Log (tabular format) with date, input/expected output/actual output and screen shot of result.

OR

Documentation of each test – one after the other

You need to include:

- reason for test
- input data or action
- expected result
- actual result
- action if the above are not the same
- screenshot of output

Discussion, Critical Analysis and Reflection

This part is very important. You must discuss the project, group work, and time management skills.

Basically, you are evaluating what you have done. Did you achieve all the required specification of the Assignment brief? If not, why not? Discuss the problems you encountered and how you overcame them.

Did you give yourselves enough time to accomplish this work? Was it harder than you expected? What have you learned from tackling this Assignment?

Did your group work well together? What did you learn about the dynamics of group work?

Did this work enhance your understanding of the Python programming language?
Did it motivate you to learn more Python, and other programming languages?

If you had to undertake this or something similar again would you tackle it differently?

What improvements or enhancements would you recommend for a future iteration of this project?

List of Group Members – Including roles they performed

This can be displayed as a table

Conclusion

What were you asked to do? Did you achieve it?

References

Harvard Referencing System. Any reference must be cited in the text

Appendix

You must include:

Complete project code

Include file/class names

Make sure you code is fully commented

You could include:

Details of you database/table structure and content

Screen shots of your interfaces

Anything else that you might want to include that does not fit well into the main report body.

This exercise will draw on all the skills you have learned so far in Introduction to Software Development unit or the Mathematics of Computational Thinking unit and the amount of work required is equivalent to approximately four weeks of practical sessions.

Deliverables

1. Group or Individual Project Report (5-10 pages with Code Appendices)

Illustrate and justify your choices in analysis and implementation

Provide evidence that you actually tested your code. Include and explain your UML diagram. Any code should go into the appendix, except core classes that

you may explain in the main text. Please do not exceed the page limit as you may be penalized.

This is a formal report and attention should be given to the format of this work. You should include a title page, table of contents, heading and sub-headings (introduction, main part, summary, and appendices). Every screenshot and diagram should have a Fig. No and caption. You are strongly advised to use the report template provided.

EVERY MEMBER OF THE GROUP SHOULD CONTRIBUTE TO THE GROUP REPORT, AD EVERY MEMBER OF THE GROUP NEEDS TO SUBMIT A COPY OF THE GROUP PROJECT.

A [Group or Individual Report](#) template will be provided on BREO.

2. [Presentation / Demonstration \(7 slides max with demonstration of your program - approx. 15-20 mins total\) OR Individual Video Recording](#)

Provide an overview of your work as an individual or group and demonstrate your working system.

PLEASE NOTE: Due to the constraints imposed by COVID it may be necessary for each student to record an individual video using screen capture software with an overview of the work and demonstration of the working system. This should be between 5 and 10 minutes in length and should be uploaded via a link on BREO.

3. [Individual Student Reflective Report \(Group Work Only\)](#)

A confidential Student Reflective Report (one from each student) to be submitted using the appropriate link on BREO. This should reflect on the ease or difficulties that you encountered when completing this group assignment. It could include how you worked with the others in the group to produce the final report and program, the difficulties you encountered and how these problems were overcome, your evaluation of your own contribution to the group work as well as the contribution of others in the group, what you have learned from doing this group work, and how would you do things differently in the future.

4. [Complete program code submitted as a zipped file](#)

Note: The tutors will be happy to advise you at any stage during the development of this program.

[Submission details:](#)

The [Group Report](#) MUST be named with the [Group Number](#). An [Individual Report](#) MUST be named with the student's [University ID](#).

Upload the [.doc](#), [.docx](#) or [.pdf](#) file via the [Assignment 2 Group or Individual Report](#) submission link provided on BREO.

An [Individual Reflective Report](#) MUST be named with your [Group Number AND Student ID](#).

The [zipped program code](#) MUST be named with your [Group Number](#).

Submission of both reports and your zipped program code must take place before [10:00am](#) on [Friday 13 January, 2023](#) using the appropriate links provided on BREO.

Capture and save evidence of submissions in a word document.

You will be asked to demonstrate that the work submitted is all your own work. *Failure to convince us that the work is your own during the presentation may result in failure of the whole group!*

Please note that all submissions will be checked for similarity to other sources, and evidence of plagiarism may be severely penalised.

Grading:

The criteria for grading are as follows.

CIS020-1 / CIS093-1 - Assessment 2 (Group or Individual Project) Marking Grid

	F	E	D	C	B	A	
Report (30%)	No submission	Poor. Little discernible merit.	Not structured and incomplete.	Basic structure and basic content.	Good structure, and acceptable content.	Excellent, professional report. Very good structure and content.	
System Demonstration (40%)	No system/ Insignificant system	Poor system	Basic system or good system (with errors)	Basic system or good system (with a few errors)	Good demonstration of working system	Excellent system and demonstration	
Presentation (30%)	No submission	Poor. Little discernible merit. Demonstrates a poor level of subject knowledge.	Weak. Demonstrates a weak level of subject knowledge.	Acceptable. Demonstrates an acceptable level of subject knowledge.	Good presentation. Demonstrates a good level of subject knowledge.	Very good presentation. Demonstrates a very good level of subject knowledge.	
							Final Grade