<p align="center">**Your MUST read this document carefully**</p>

<div align="center">

CMPS310 Software Engineering
- Group Project -
Milestone-1: Software Requirements Analysis and Design **(12%)**
This is a compulsory project. If you do not submit this, your grade will be zero (0) in this course.
Each group can have <u>four</u> members
Blackboard Submission Due: Midnight <u>October 16, 2024</u>
Fall 2024

</div>

Objectives of the Project

The purpose of this project is to reinforce the concepts discussed in class. It provides an opportunity to apply the concepts and techniques learned to a realistic problem. The main objectives are to practice **<u>Requirements Specification</u>** and **<u>Object-oriented Analysis and Design Modeling</u>** for a software solution.

Background

Assume that Mr. Khalid Hassan is the newly appointed Director of the Qatar Vehicle Registration (QVR) department. Currently, the department offers various services such as vehicle registration renewal, ownership transfer, management of compulsory insurance policies, fitness certificates, accident reporting, and penalty handling. These services are not yet available online to the general public. Mr. Khalid has decided to make these key services available online through a platform called iQVR for the public in Qatar. Recently, Mr. Khalid contacted your group to develop the online system for the iQVR department. He has provided a description of the major functions expected to be available online and has emphasized that the scope of your analysis, design tasks, and system boundary relates completely to QVR.

Currently, iQVR comprises five sections: The Finance section handles payments, the Penalty section manages fines and offences, the Registration section processes renewals and ownership transfers, the Accident section addresses all accident-related matters, and the Technical section manages vehicle fitness. The department uses a computer-based system for managing all payments and employee salaries, which is a standard commercial-off-the-shelf package known as qPay. Mr. Khalid now intends to replace all five sections with an automated online system while keeping qPay as a separate, connected system. He plans to employ only one system administrator to manage this automated process. iQVR should function as described below.

Vehicle owners, insurance companies, and authorized workshops will need a login ID and password to access the iQVR online service. It is assumed that users are already logged in by default, so login functionality does not need to be designed or developed by your team.

To register a new vehicle or renew its registration, the vehicle owner must first purchase an insurance policy from an insurance company. Once the policy is bought, it is immediately recorded in the iQVR system. Each vehicle must have its own insurance policy, and a policy is valid for only one vehicle. The company enters the Vehicle Identification Number (VIN) into the system and attaches a copy of the policy. The system then records this information along with the company data and generates an acknowledgment receipt for the company.

If the vehicle is over two years old, it requires a fitness certificate from an authorized workshop in Qatar. Once the vehicle obtains a fitness certificate, the workshop submits it online to the iQVR system by entering the VIN and certificate number and attaching a copy of the certificate. Each certificate is valid for only one vehicle; however, a workshop can issue multiple certificates. The system stores this information along with the workshop data and generates an acknowledgment receipt for the workshop. A registered vehicle can have only one current owner, but an owner can have multiple vehicles.

To transfer the ownership of a new, unregistered vehicle (one that has not been registered before) to another person, the current owner enters the VIN, vehicle make, model, and year of manufacture into the system. The system then provides the VIN and car details to another system operated by the vehicle manufacturer. This system verifies the VIN and other

specifications with its factory database. If the manufacturer confirms the information, the iQVR system requests the current owner's name and QID. If the manufacturer's response is negative, the system terminates the session with the message "Incorrect vehicle information". Next, the iQVR system communicates with another system run by Qatar Trade Service, using the VIN and QID of the current owner, to check if the owner imported the vehicle from overseas. Qatar Trade Service checks its records and provides a response to iQVR. If the response is positive, the system then retrieves the insurance policy for the vehicle, which was previously submitted by the insurance company. If the response is negative, the system terminates the session with the message "Incorrect ownership." If the insurance policy exists, the system generates a registration number for the vehicle, assigns it to the vehicle, and displays the number to the owner. The current owner then provides the name, QID, address, and mobile phone number of the new owner. The system records these details and assigns the vehicle to the new owner. It designates the current owner as the 'previous owner' and the new owner as the 'current owner'. The system then confirms the ownership transfer by generating a new registration sticker and creating an invoice for the transfer fee, which the new owner will pay later.

However, to transfer a registered vehicle to another person, the current owner enters the VIN of the registered vehicle and the QID number of the current owner. The system first retrieves the registration details of the vehicle. If the retrieved information does not match the provided data, the system terminates the session with the message "Incorrect Information". Otherwise, it initiates the transfer process. The system checks for any unpaid invoices (bills) for registration fees or traffic fines. If there are any unpaid bills, the system terminates the session with the message "Pay the bills first". The payment procedure is explained later. If there are no unpaid dues, the system requests the new owner's details. The current owner provides the name, QID, and mobile phone number of the new owner. The system then assigns the vehicle to the new owner, designates the current owner as the 'previous owner' and the new owner as the 'current owner'. The insurance policy assigned to the vehicle during the last registration remains unchanged; however, the new owner can update the insurance policy later. The system confirms the ownership transfer by generating a new registration sticker and creating an invoice for the transfer fee.

To renew a registered vehicle, the vehicle owner enters the VIN into the system. It retrieves the registration details and checks if the vehicle is over two years old. If it is, the system finds the vehicle fitness certificate issued by an authorized workshop. The system terminates with a message "Get fitness certificate first" if the fitness certificate is not available. If the certificate exists, it retrieves the compulsory insurance policy of the vehicle issued by an insurance company. If a new insurance policy is available, the system checks for any unpaid fines for traffic offences; otherwise, terminate the system with the message "Get insurance policy". If there is any unpaid fine(s), it asks the owner to pay the fine(s) with the message "Pay the bill first''. If there are no unpaid fines (s), the system creates a new registration with the same validation period as specified in the new insurance policy. The insurance policy and the fitness certificate are then attached concurrently with the vehicle registration. The system then prepares a new registration sticker with new validity. It finally creates an invoice for the registration renewal.

For an accident between only two vehicles, if a vehicle accepts that it was his/her fault, he/she enters VINs of his/her and the other vehicle (victim). The system finds vehicle details and registration information for both vehicles. The system then asks to provide the date, time, location, and a brief accident description. The system records this information, and by default, it sets that the vehicle that entered all information is the offending vehicle's owner. The other vehicle is the victim of the accident. The system asks the owner of the offending vehicle to confirm this. Once confirmed, the system stores this. The system then concurrently finds the insurance policy of the offending vehicle and creates an accident report with a unique case number. The iQVR system then sends the accident report to the insurance company, which immediately sends an acknowledgement receipt saved by the iQVR system. The accident report is then stored and available for both parties (owners of the offending and victim

vehicles) to retrieve at any time. Any authorized workshop can retrieve the accident report if the case number is known. However, if there is a dispute regarding an accident, no online reporting is allowed. Both parties must go to a police station to solve the dispute.

Vehicle owners can pay any unpaid invoices or fines for a vehicle using a credit card. To pay, the owner provides the VIN, and credit card details such as number, name of the cardholder, and validity. The system retrieves the vehicle registration details. If the registration details do not exist, the system displays an error message and asks to enter the correct VIN. Otherwise, it lists unpaid invoice(s), if there are any. The owner then selects which of the invoices and fines he/she wants to pay. The system saves the selection, computes the total amount, and then forwards the card details and the total amount to the qPay system for approval. qPay is a separate software system but owned by the OVR department. The qPay checks for the card's validity by contacting the credit card provider (bank). If the card is invalid, it generates an error message to *i*QVR without payment processing. Otherwise, qPay returns an approval advice to *i*QVR. For invalid credit cards, *i*QVR asks the owner to enter valid credit card details. The *i*QVR system records the outcome sent by qPay. It then concurrently records the approval number, prepares a payment receipt, and sets the invoice(s) as paid. It then displays the receipt. Finally, the system updates the list of the unpaid invoices associated with the vehicle.

The traffic police can determine which vehicle has how many red-light offences in the last year, one month or one week. He/she enters the period (day, week, month) and selects the red-light offence type from the system. Based on these criteria, the system finds a list of vehicles that match these criteria. The traffic police select those vehicles that exceed a certain number of red-light offences during this period. The system creates a confiscating order for those vehicles and asks for confirmation from the traffic police. It saves the confirmation, cancels the registration of those vehicles, and informs the owner of each vehicle. The owner receives the order. The system then broadcasts the orders to all police departments. A confiscating order is attached with only one vehicle; however, one can get more than one order.

The system knows the registration details of vehicles, models and the make of each vehicle; previous and current owners of each vehicle such as their name, QID, and address. It also knows the insurance policy of every registered vehicle, such as policy number, validity, and the name of the contact details of the issuing company of the policy. A company can sell many insurance policies. The system also stores information about the fitness certificate of each registered vehicle and the details of the workshop that issued the certificate. It keeps information about the invoices, fines, traffic offences and the involved vehicles. The system knows the details of each accident, such as the date, time of the accident, and the involved vehicles. However, it does not record any information about the traffic police except their login data. The system keeps records of all acknowledgements and responses that it receives from external systems.

Currently, the proposed system can handle 10 million vehicles. However, it is projected that the traffic system will have more than 30 million vehicles within the next 10 years. More new functionalities must be added to the system; some existing ones might be modified or deleted to enhance further capabilities. The core data about the vehicles and the ownership details are confidential; these should not have direct exposure to any general user-level functions. These need to be protected by separating them from the high-level user interface. The software could be portable, which means it can run on various machines such as mobile phones, tab, desktops, etc. From time to time, some functions may need upgrading and modifications, but the interfaces to the rest of the system may remain the same. The software could be distributed in future; different components may run on different machines in different locations. Several backup modules will be managed to support the availability of the systems to the users most of the time. Various software components should be less dependent on each other and more focused.

The traffic police department does not have the budget for more than 10 new technical staff for the project. It has a budget for only 20 new servers. The system must run as a plug-in for the registration system that uses the Oracle database system for data management. Most functions were developed using Java and C. The traffic police department has qualified staff experienced with Oracle, Java and C. The first version of the system should be completed

within three months. Any delay will cost money for the traffic department because the system is expected to start operating precisely after 12 months.

Additional Information

You will likely need additional information about the system's operations to complete this project. In software industry, it is common for software engineers to seek further details when designing a system. For this project, **your Lab Instructor** is your client, and you can direct any questions to him/her. In many cases, you will need to make logical assumptions. You are also encouraged to explore other sources, such as the Internet and related literature, to better understand how similar systems operate, including their core business entities, services, and functions. Please note that the provided description may be unclear or incomplete. It is your responsibility to gather additional information or seek clarifications from available resources. You <u>MUST</u> include justifications for any assumptions you make in your project.

Tasks of Your Project

Milestone 1: Software Requirements Analysis

1. Propose:
    a. A <u>data flow diagram (DFD</u>) of the system: Include major processes, data storage/data files, data flows, and external entities (*20%*).
    b. A <u>use case diagram </u>of the system (*20%*)*:*
    c. Explain which of these two you prefer best to analyze a system and why (*5%*).

2. Develop *use case specifications* for the <u>most complex and key **4** use cases</u> selected from your use case diagram. Your proposed each use case specification should include *brief description, primary actor(s), trigger, pre-condition(s), post-condition(s), normal scenario (actor action-system response table), alternative flows (if any)*. Use the "Use cases specification template" available from Blackboard (4 x 5%= *20%*).

3. Using *object-oriented design fundamentals (objects, inheritance, abstraction, classification, encapsulation, modularity, responsibility-driven design, patterns, etc.)*,

    a) <u>Complete the *design class diagram*</u> of the entire system with all required classes, their attributes and types, relationships (aggregation, generalization, association) with multiplicity where applicable, methods with major parameters, and visibility of attributes and methods (*30%*).
    b) <u>Explain briefly</u> how you applied design principles in your class diagram. In your explanation, give examples from your class diagram (*5%*).

Any assumptions you made regarding the system description must be explained. Assignment components will be evaluated for **accuracy**, **clarity**, **relevancy, and completeness** (especially among components and among artifacts) of your document.

- Each student must upload their electronic copy of Milestone-1 to Blackboard as a *Word Document and .vpp files* **by the due date and time**.

- Your word document and .vpp files must include all diagrams produced in Visual Paradigm.

Each student must submit an individual electronic copy on Blackboard <u>by the due date and time.</u>

Grading scheme for Milestone-1: Requirements Analysis

Tasks	Criteria	Grade
1	(a) DFD (20%), (b) Use case diagram (20%), (c) Explanation (5%)	45
2	4 use case specifications using the template (4 x 5%= 20%)	20
3	Design Class diagram of the entire system (30% for (a); 5% for (b))	35
	Total	**100**

More details will be provided for the subsequent millstones.

Submission requirements

Your assignment group/team MUST comply with the following submission requirements; **otherwise, the grade will be deducted**:

1. Your group should have four members. If you have less than **4** members in your group, please contact your lab instructor. You will form your own group.

2. Each member must submit the electronic copy individually on Blackboard. The cover page of your submission MUST contain the following items in four bullets:

   [Each member of the group will be required to submit a peer group evaluation with an estimate of the contribution of each member to the project. Write the percentage contribution made by each team member so that it adds up to 100%. This evaluation *may* be used to adjust the marks awarded to each team member.]

   - Effort distribution of the student:

     SID:_____STUDENT NAME: _____Effort given_____%
     SID:_____STUDENT NAME: _____Effort given_____%
     SID:_____STUDENT NAME: _____Effort given_____%
     SID:_____STUDENT NAME: _____Effort given_____%

   - Course number_____

   - Submission date_____

   - Theory Class section: **L02 / L03 / L51 / L52** (select one)

   - DECLARATION: *We hereby certify that no part of this project or product has been copied from any other student's work or from any other sources except where due acknowledgment is made in the project. No part of this project/product has been written/produced for us by any other persons.*

Note: If the above submission requirements are not met, up to 10% of marks will be deducted from this milestone.
   3. Be aware of
      - *Submitted work must be the student's own work*
You cannot copy the project from other groups.