Task 3: Architecture design

Task 3.1: Describe an architectural approach you will use to implement the desired system. How many modules do you plan for the whole WMC 2.0 system? Briefly describe input, output and function of each module?

a. Describe an architectural approach you will use to implement the desired system.

Architectural approach: Model–View–Controller (MVC) is a software architectural pattern commonly used for developing user interfaces that divide the related program logic into three interconnected elements. This is done to separate internal representations of information from the ways information is presented to and accepted from the user.

Model Diagram:

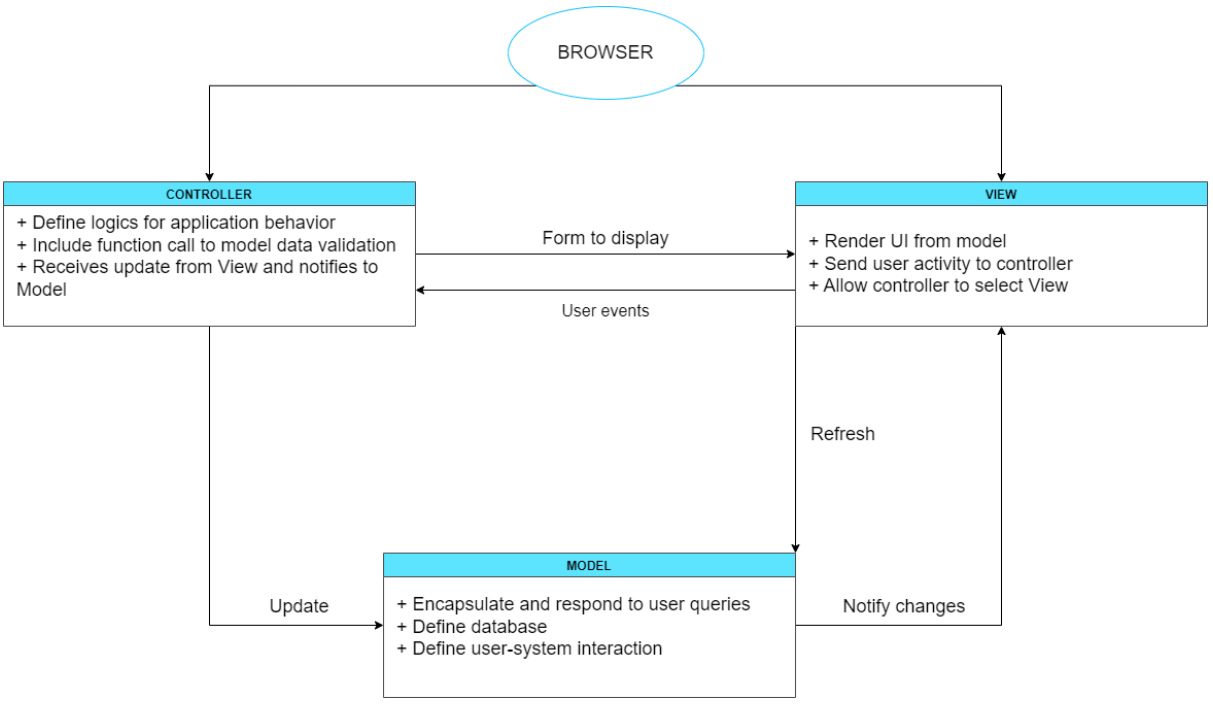

Figure 3.1: MVC model diagram

Description:

Model: Data Structure which has an assignment to cooperate with databases based on functions, methods to modify the data through the instructions of the controller. Model will update the information for view after processing data.

View: Display graphical user interface for users to interact with the system. Views can receive the notify from the controller and information updates from the model.

Controller: Contains behaviors to process the data, verify data. Controller has functions and methods to manipulate the model. Sometimes the controller can notify you.

b. How many modules do you plan for the whole WMC 2.0 system? Briefly describe input, output and function of each module?

We will use 8 Modules is listed in the table below for whole UWC 2.0 system.

Modules	Description
Log in/out (Data validation)	**Input: Registered information from User Output: Main interface – validation successful Function: This module is a compulsory one used to grant access to users (Officer level/ Janitor & Collector level)**
Route planning	**Input: MCPs from Officer Output: Calculated distance Function: Route planning module is meant to be a productivity-oriented module since it helps significantly reduce the human workload which could have been the Officer's responsibility. This calculates the current total distance a J/C has to finish on a daily basis and gives officers chances to maximize J/C work efficiency.**
Schedule	**Input: Information of back officers, janitors or collectors Output: Work calendar Function: Manage all the work schedules of the employees. You can view detailed parameters: location of employees, number of tasks, vehicle parameters,…**
Task assignment	**Input: Information got from the database and route planning Output: Assigned task for all J/C Function: Task assignment is one of the main modules for this application. Based on retrieved information, Officers will assign the tasks (including MCPs location and vehicles) to all J/C available.**
Authenticate	**Input: information of staff Output: verified or unverified Function: Responsible for authenticating the identity of the employees in the system to decentralize control and use of the system's resources.**

Communication	**Input: Message** **Output: Sent messages** **Function: Communication function is an additional function which helps solve unexpected situations during working hours and increase communication between workers.**
Feedback/ Leave application	**Input: Feedbacks/ Leave application form** **Output: Reply/ Leave permit** **Function: This allows J/C to communicate directly with Officers who control their tasks. Moreover, with this module, Officers will be able to optimize their assignment.**
Task check in/check out	**Input: check in/ out for task** **Output: workday recorded in database** **Function: Store the status of task**

3.2. Draw an implementation diagram for Task Assignment module

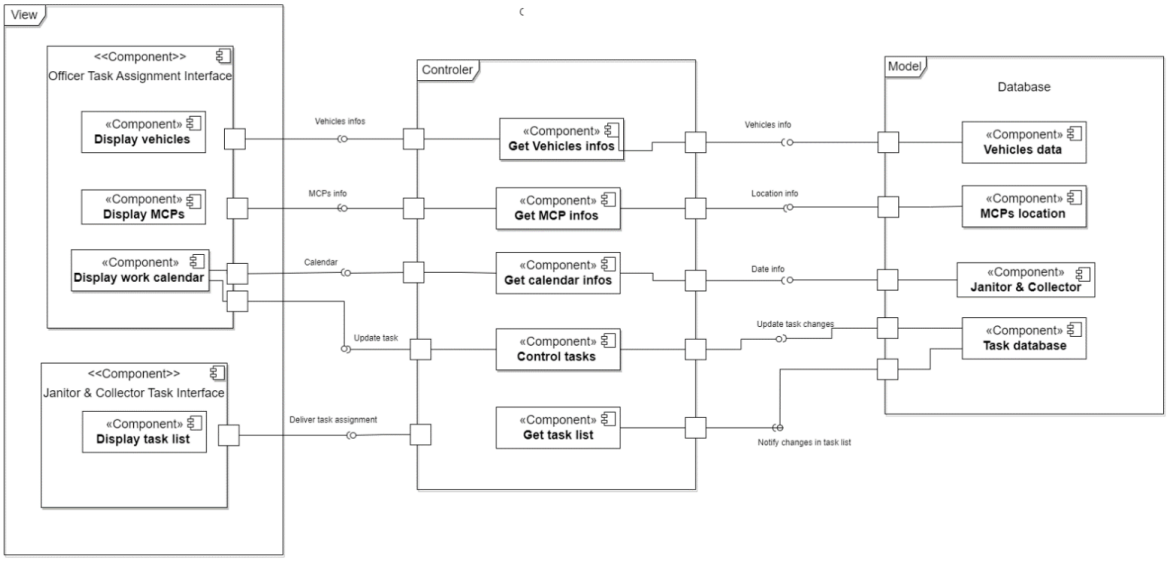

Figure 3.2: Component diagram of the UWC2.0