VIETNAM NATIONAL UNIVERSITY, HO CHI MINH CITY
UNIVERSITY OF TECHNOLOGY
FACULTY OF COMPUTER SCIENCE AND ENGINEERING

SOFTWARE ENGINEERING (CO3001)

Project (Semester 222)

Urban waste collection - UWC 2.0

Advisor: Mr. Bùi Hoài Thắng
Group: GROUP 6
Students: Nguyễn Trọng Duy - 1852296
Phạm Quang Khánh - 1852459
Bùi Trung Đức - 1852324
Nguyễn Thế Lộc - 1952825
Nguyễn Khương Duy - 1952615

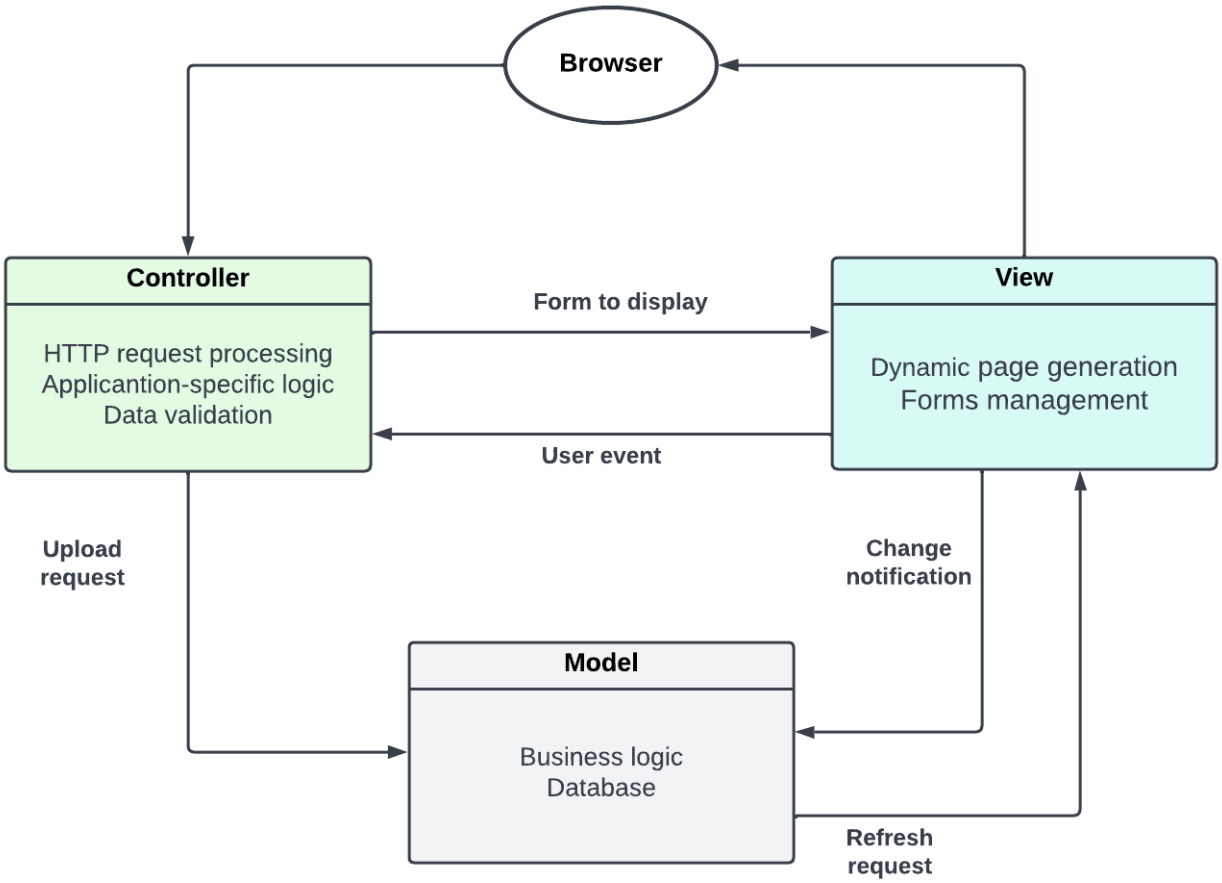
HO CHI MINH CITY, March 2023

Contents

1 TASK 3: ARCHITECTURE DESIGN

1.1 Describe architecture

1.1.1 System Design: MVC Model



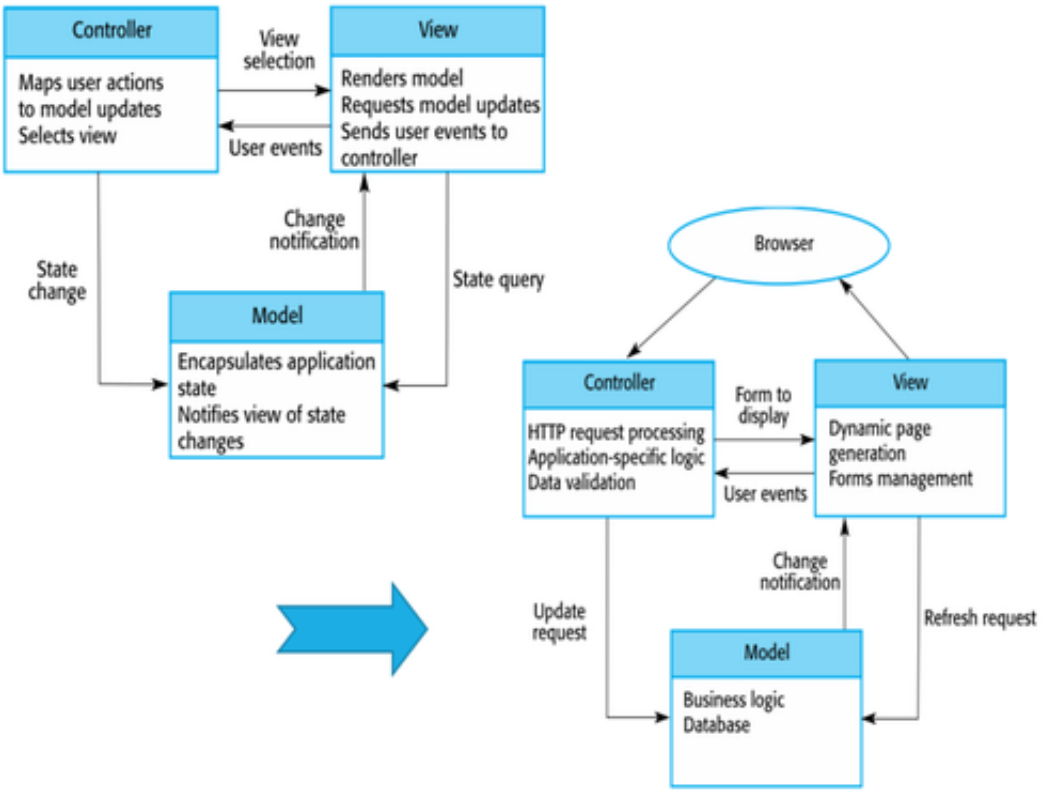


Figure 1: MVC Model

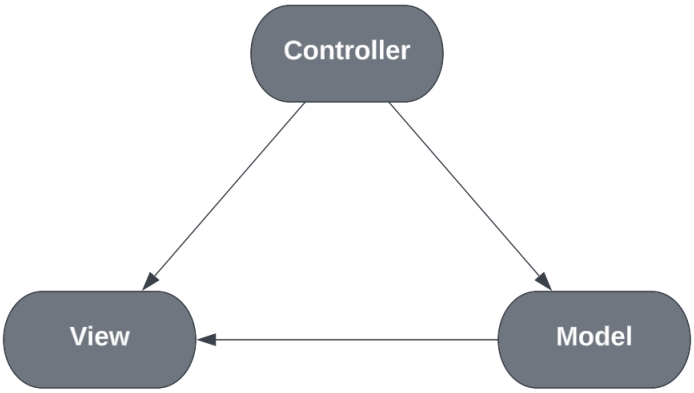
The Model-View-Controller (MVC) is an architectural pattern that separates an application into three main logical components: the model, the view, and the controller

View:
The View component is used for all the UI logic of the application. For example, the Customer view will include all the UI components such as text boxes, drop-downs, etc. that the final user interacts with.

- Render form login, registration, navigation bar, button, etc.
- Render the table of assignment of tasks (Task), work schedule (Calendar),... for Janitors and Collectors.
- Render out the task management table (Task Assignment) for the Back-Officiers.
- Create effects that interact with the user.
- Interact and send data update request to Controller
  - Submit login and registration information.
  - Send information about updated tasks.

  - Sending Click, Submit form events,.. etc of the user

- **Controller** keeps roles:
  - Including Classes/Functions with many different Logic functions
    * Authenticate the user when sending the request to the system
    * Find the optimal path between MCPs
  - Handles user requests via View
    * Get information about assignment schedule
    * View detailed tasks
    * Create, modify and delete tasks
  - Act as an intermediary between Model and View. Controller sends requests updating data to Model. The these requests will continue to be transmitted to the View to update the display interface to match the updated data in the Model for the user to see.



- **Controller - View interaction:** Get images, buttons, or Display the data returned from the Controller to make it possible for the user Easy to observe and operate. This interaction can also have data are not taken directly from the Model, this time it is only for display purposes Just like pictures and buttons.

- **Controller - Mode interaction:** is the processing flow when the Controller continues receive requests and input parameters from the user. Controller will now use Use the classes/functions included in the model to get the correct data as needed by the user.

- Two components View - Model can interact with each other without through Controller. At this time, it only takes on the task of displaying data that without going through any logical processing. Similar to display areas Static on Websites like Block Sidebar.

Model:

- Provide methods to interact directly with the database to perform show system functions.

- Provides methods to perform database operations (add, delete, edit) with features to create and assign tasks, update user information.
- When the state of the model (in terms of data,...) changes as the system needs Quick update to the view => notify the view to do refresh and update view again.

1.1.2 Our Module

1. Task Assignment

Input: Back officer

Output: The confirmation that the task is assigned successfully

Function: Create and assign tasks to Collectors and Janitors

2. Attendance

Input: Janitor and Collector

Output: Job confirmation and completion

Function: Check-in and check-out task

3. View Task

Input: Janitor and Collector

Output: Work schedule and assigned tasks

Function: View task assignment and work calendar

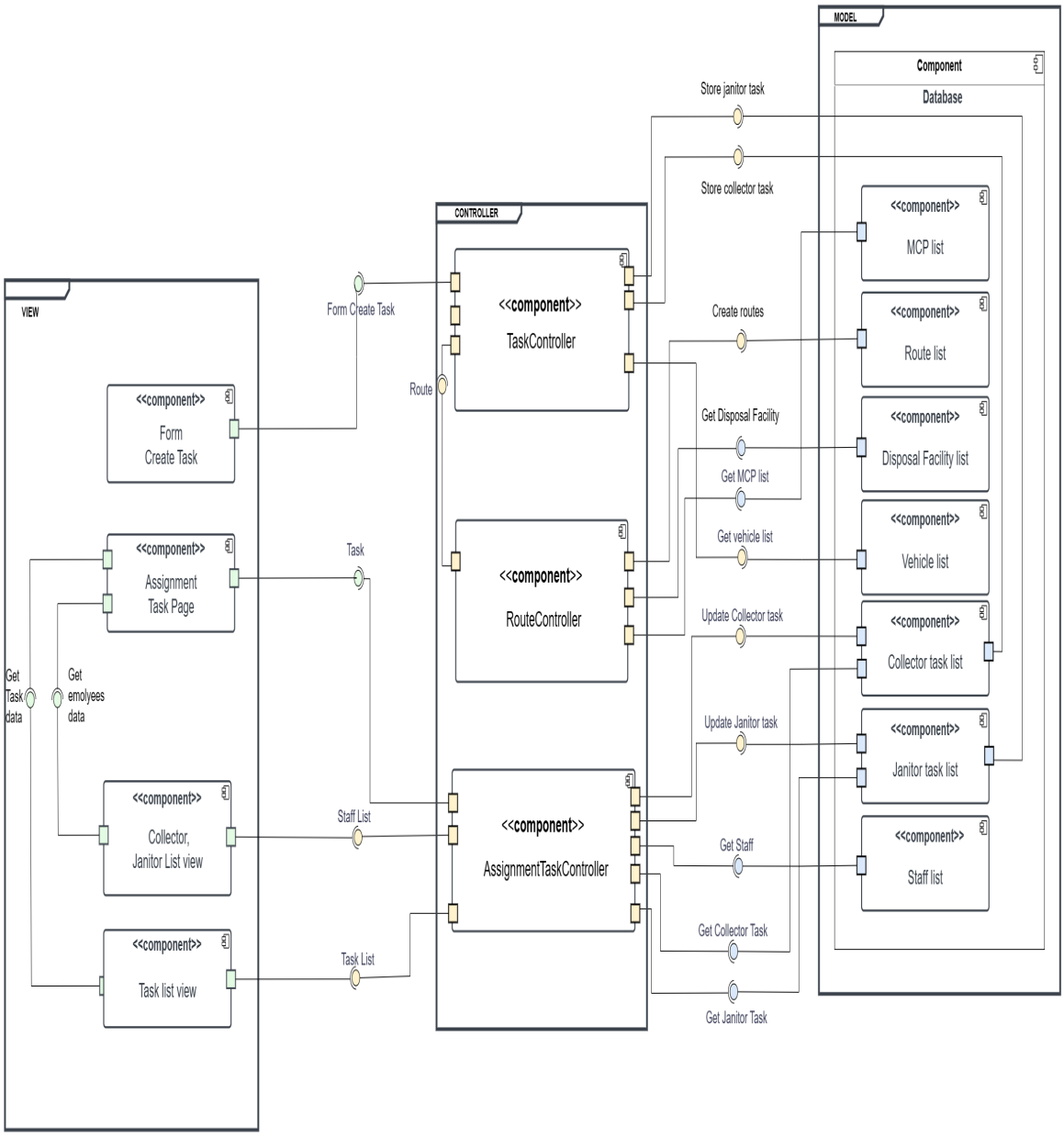
4. Communication

Input: Back Officer, Janitor and Collector

Output: Notification from system and real-time chat

Function: Notification, Chat

1.1.3 Components Diagram



1.1.3.a Diagram Description:

- When the Back officer accesses the task assignment page in the Display Assignment Task Page, information about the list of employees (Janitor, Collector) will be displayed at

the Display Collector, Janitor list component in the View. In addition, information about previously created tasks will be displayed through the Display task list.

- The back officer can select the Create task button to display the task creation form through the Display Form Create Task component. At the Create Task component in the controller, it will use the Create Route, Assign Vehicle components to create routes between the MCPs, and select the vehicle to use to transport the garbage. Then the information of the task will be transferred to the system and saved to the database Janitor Task list and Collector Task list at Model.
- At the task assignment page, the Back officer will select 1 staff from the list of staff, and possibly multiple tasks from the list of tasks. Then assign the selected tasks to that staff. At the Assign Task component in the controller, it will perform processing logic such as the task being assigned having the same schedule as the staff's schedule, ... After processing is complete, it will update the selected employee information on that task and save it to the database.