VIET NAM NATIONAL UNIVERSITY - HO CHI MINH CITY UNIVERSITY OF TECHNOLOGY

FACULTY OF COMPUTER SCIENCE AND ENGINEERING



**Software Engineering (CO3001)**

**URBAN WASTE COLLECTION AID - UWC 2.0**

Task 3: Architecture design

|  |  |  |
| --- | --- | --- |
| **Teacher:** | Lê Đình Thuận |  |
| **Class:** | L05 - HK221 |
| **Students:** | Trần Nhân Nghĩa | - 2013879 |
|  | Nguyễn Trung Kiên | - 1913858 |
|  | Tưởng Thọ Hiếu | - 1913369 |
|  | Phạm Thanh Tú | - 2014982 |
|  | Nguyễn Đoàn Anh Kiệt | - 1913876 |
|  |
|  |

Ho Chi Minh City, September - 2022

**Contents**

[1. Task Assignment 2](#_Toc118564887)

[2. Architectural approach: WMC 2.0 system 3](#_Toc118564888)

[2.1 View 4](#_Toc118564889)

[2.2 Controller 4](#_Toc118564890)

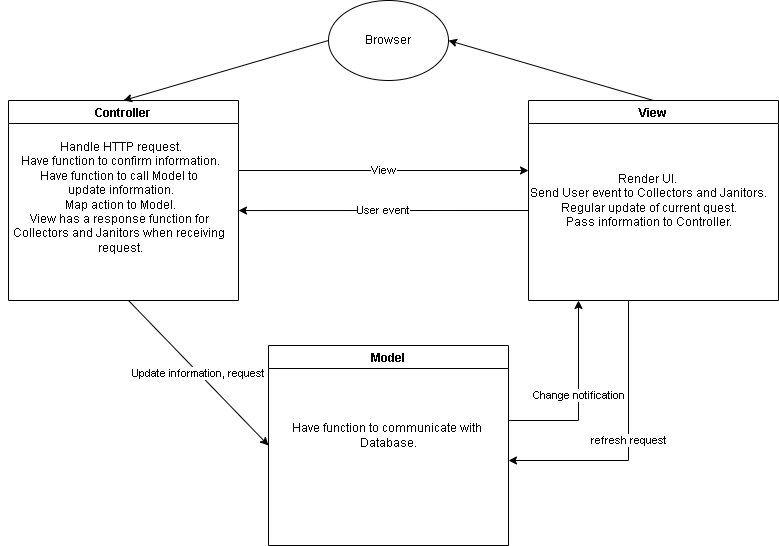
[2.3 Model 4](#_Toc118564891)

[3. Implementation Diagram: Component diagram 5](#_Toc118564892)

# 1. Task Assignment

|  |  |
| --- | --- |
| Trần Nhân Nghĩa | Draw an implementation diagram and describe it and write report |
| Phạm Thanh Tú | Draw an implementation diagram and describe it |
| Tưởng Thọ Hiếu | Describe an architectural approach |
| Nguyễn Trung Kiên | Describe an architectural approach |
| Nguyễn Đoàn Anh Kiệt | Draw an implementation diagram and describe it |

**2. Architectural approach: WMC 2.0 system**



**Link Drive Photo:** https://drive.google.com/file/d/1nN87l4MY27FilchTb6\_MpXagoG6-x4gW/view?usp=sharing

## 2.1 View

* Render the User Interface (UI) of each page:
* Render user login/registration interface.
* Interface of personal information, information about landfills, users, etc. for the main page.
* Export 1 button for “Nhiệm vụ”. A submap for the MCP location.
* Output a form to interact directly with BO or between Janitors, and Collectors.
* Interact with Janitors and Collectors:
* Automatic Slider.
* Show “Danh sách nhiệm vụ” when clicking “Nhiệm vụ”.
* Show “Bản đồ MCP” when clicking on each mission.
* Send the user's operation history to Controller.
* When there is a notification of changed information from the Model, the View will retrieve the data from the Model to refresh and update the new view.

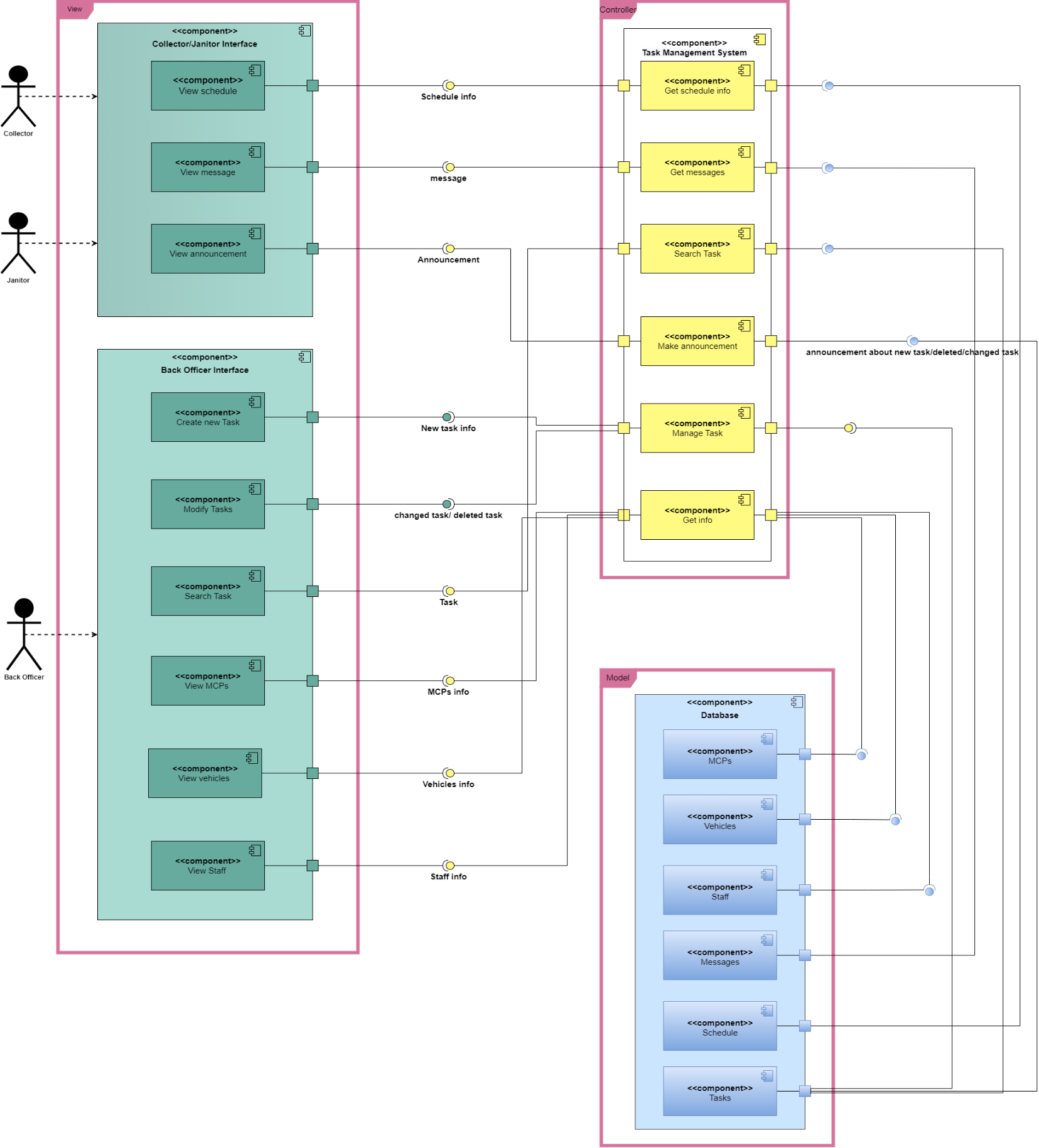
## 2.2 Controller

* Returns a view depending on the requested page.
* Call functions from Model to perform operations on data such as: Register/login, user information; Update tasks for the day; Delete or Reload Task information.
* Call a function from the Model to validate information and data.
* Send feedback to the user with each corresponding request.

## 2.3 Model

* Provide a method to interact with the Database to perform registration/login or user account management.
* Provide methods to perform operations on the Database such as automatically updating information through sensors for MCPs, deleting MCPs if processed, etc.
* When the Model state changes in terms of data as above, it will notify the View to perform a refresh and update the new view.

# 3. Implementation Diagram: Component diagram

****

**Link Drive Photo:**  <https://drive.google.com/file/d/1w844eZHuLLZ7hUZweGAIBOdZ5owL47Y-/view?usp=share_link>

***Description:***

- The system has 3 major components:

* Vỉew component:
* Contains many View component subsets.
* The View components take on the tasks as described description in task 3.1
* The View components use interfaces from the controller to make requests, use interfaces from the model to perform data updates, and provide interfaces for the model to notify about data updates.
* Controller component:
* Contains many Controller component subsets.
* The Controller component takes on the tasks as described description in task 3.1
* Controller component provides interfaces for the view to make requests and uses interfaces from the model to perform database operations.
* Model component:
* Contains many Model component subsets.
* The Model component takes on the tasks as described description in task 3.1
* Model component provides interfaces for the controller to perform operations with the database, providing interfaces so that the view can update data when it receives a message from the model.