# III. Architecture Design

## 1. MVC Architecture

### 1.1. Advantages and Disadvantages of MVC Architecture

[To do]

### 1.2. Reason for choosing MVC Architecture instead of Layered Architecture

[To do]

## 2. Architecture Diagram and Deployment Diagram

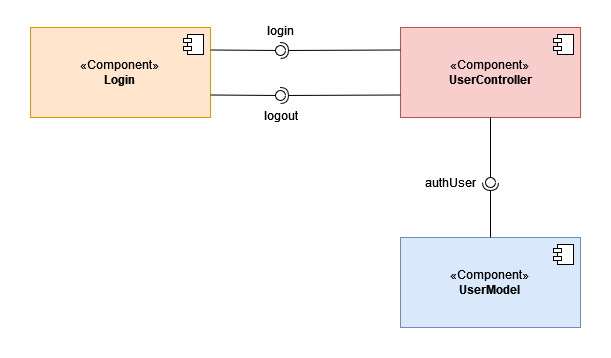
Use a layered architecture to design the HCMUT-SSPS system. Describe how you will present your User Interface. Describe how you will store your data. Describe how you will access external services/ APIs.

**Hint**: draw 01 architectural diagrams for the overall design of HCMUT-SSPS system. Write 01 paragraph for your Presentation strategy, 01 paragraph for Data storage approach and 01 paragraph for API management. The architectural decisions in these paragraphs should be justified and associated with external links for detailed approaches.

[To do]

## 3. Component Diagram

### 3.1. Module: Authentication



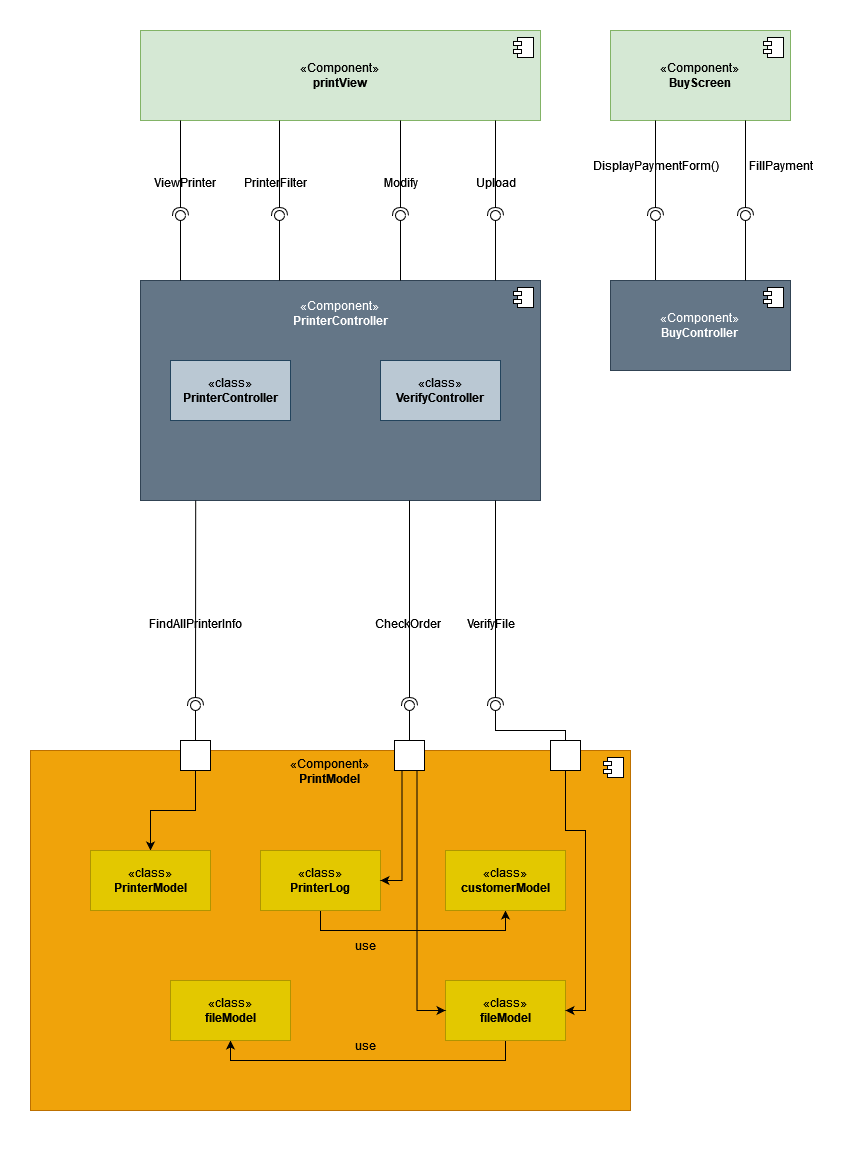
*Figure x: Component Diagram for Authentication Module*

**Description:**

In the component diagram for the Authentication module, the view layer includes a component called **Login** providing the login interface for users to login to their accounts. The controller layer includes a component called **UserController** providing the methods for authenticating the accounts. The model layer includes a component called **UserModel** storing and performing queries on data related to user accounts (authentication).

### 

### 3.2. Module: Printing

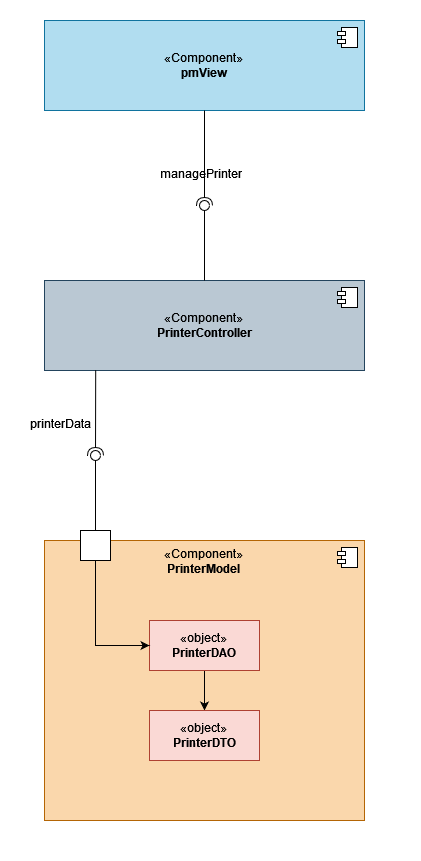


*Figure x: Component Diagram for Printing Module*

**Description:**

In the component diagram of the printing module, the view layer includes two components called **printView -** which provides the interface for the user to perform the operations such as choosing a printer, switching to buying printing pages -and **BuyScreen -** which provides the interface for the user to purchase printing pages. The controller layer includes two components, **PrinterController** and **BuyController**. The **PrinterController** component contains two classes, **PrinterController** and **VerifyController** providing the methods for searching information and choosing the printer. The **BuyController** component provides the methods for creating the buying printing pages form. The model layer includes a component called **PrintModel** for storing and performing operations on printing such as get printer information, verify and store the document file for printing.

### 3.3. Module: Printer Management



*Figure x: Component Diagram for Printer Management Module*

**Description:**

In the component diagram of the Printer Management module, the view layer includes one component called **pmView** providing the interface for SPSO to manage the printers (add, enable, disable printers). The controller layer includes one component called **PrinterController** providing the methods for managing the printers such as add, enable, and disable printers. The model layer includes one component called **PrinterModel** for storing and performing actions related to managing printers such as add, enable and disable printers. Inside the **PrinterModel** contains 2 objects called **PrinterDAO** (to perform operations on the database)and **PrinterDTO** (to store the printer’s data).

The interfaces include:

* **managePrinter**: provides the methods for managing the printers (add, enable, disable).
* **printerData**: provides the API to retrieve the printer’s data for management (add, enable, disable).