

# Documentation for UniVerse

## Project Overview

UniVerse is a tool created to help students navigate their academic and career journeys by linking them with mentors and streamlining the college application process. The system provides features for signing up, logging in, and role-specific capabilities for both students and mentors, guaranteeing a smooth user experience.

## Patterns Used in the Code

### 1. Singleton(Creational)

**Purpose:** Ensures a single instance of UserManager to manage the state of registered users across the system.

**Implementation:** The UserManager class follows the Singleton pattern using a private constructor and a static getInstance() method.

### 2. Factory Method(Creational)

**Purpose:** Encapsulates the creation of Student and Mentor objects based on user input.

**Implementation:** The UserFactory.createUser() method abstracts the instantiation logic for Student and Mentor objects, enhancing modularity and scalability.

### 3. Observer(Behavioural)

**Purpose:** Notifies mentors when a new student selects them, enabling real-time updates.

**Implementation:** The Mentor class implements the Observer interface, and the update(Student student) method is invoked whenever a student chooses a mentor.

### 4. Facade(Structural)

**Purpose:** Simplifies interactions with the system through a unified interface (MenuFacade).

**Implementation:** The MenuFacade class provides an accessible entry point for user registration, login, and navigation of student and mentor menus.

### 5. Decorator(Structural)

**Purpose:** Enhances the display of user information without altering the User class.

**Implementation:** The UserDecorator class extends user details dynamically, providing formatted information about Student and Mentor objects.

### 6. Strategy(Behavioural)

**Purpose:** Differentiates the behavior and logic for students and mentors, ensuring separation of concerns.

**Implementation:** Role-specific methods like `handleStudentMenu` and `handleMentorMenu` in the `MenuFacade` class encapsulate distinct functionalities for each user type.

## **Main characteristics and process flow**

### **User Registration:**

Users can register as either Student or Mentor.

Students select their university preferences and mentors specify their expertise.

### **Login:**

Role-based login allows access to functionalities tailored to the user's role.

### **Student Menu:**

Access university application checklists.

Select courses and connect with mentors.

View all registered students.

### **Mentor Menu:**

View assigned students.

Access educational materials.

### **Dynamic Mentor-Student Interaction:**

Students can select mentors based on their expertise.

Mentors receive real-time notifications about new students.

### **Enhanced User Display:**

The `UserDecorator` dynamically provides detailed user information during interactions.

**In conclusion,** The UniVerse provides a comprehensive platform for students and mentors to collaborate effectively. By simplifying the university application process and offering tailored tools for academic and career guidance, the system addresses key challenges faced by users.