MODULE-4 UPDATING THE TEACHER'S TIMETABLE

4.1. DESCRIPTION OF THE MODULE

The "LeaveReq" module and "TeacherNotification" module of a part of main module SignInfoPage the application, these modules allows the user to request leave to the admin and update the timetable accordingly. Here's the description of the following module.

SignInfoPage: The primary objective of this code is to consolidate all the methods and functions relevant to teachers into a single module, called "SignInfoPage," which extends the functionality of another module called "TeacherStorage."

JavaFX is a framework for creating desktop applications with a graphical user interface (GUI). In this case, the code aims to create an interactive interface for teachers, allowing them to perform various tasks related to their work. The application uses JavaFX layout containers, event handling, and styling to achieve this.

By extending the "TeacherStorage" module, the "SignInfoPage" module inherits or enhances the functionalities provided by the "TeacherStorage" module. It includes methods and functions specific to the sign-in information page, such as updating timetables, applying for leave, displaying notifications, and sending emails.

The graphical user interface (GUI) is constructed using JavaFX layout containers. Layout containers are components that arrange other components (buttons, labels, text fields, etc.) in a structured manner. Examples of JavaFX layout containers include VBox (vertical box), HBox (horizontal box), GridPane (grid-based layout), and BorderPane (border-based layout).

The event handling aspect of the code deals with defining actions or behaviors that occur when the user interacts with the GUI elements. For instance, clicking a button might trigger a method that updates the timetables or displays notifications. Event handling allows the application to respond to user input and perform the necessary actions accordingly.

Overall, the provided code aims to create a user-friendly and efficient sign-in information page for teachers using JavaFX. It combines various functionalities into a single module, enabling teachers to access and perform their tasks conveniently through the graphical user interface.

LeaveReq: The "LeaveReq" module is designed to facilitate a user-friendly graphical user interface (GUI) that enables users to easily request leave from the administrator. When initiated, the module opens a window that presents several input fields and controls, carefully designed to collect all the necessary information for requesting an absence on a specific day.

Through this intuitive interface, users can conveniently enter details such as the desired date of leave, the reason for the absence, and any additional comments or relevant information. The GUI provides a seamless experience, allowing users to navigate and interact with the various fields effortlessly.

By utilizing the "LeaveReq" module, users can efficiently communicate their leave requirements to the administrator, streamlining the leave request process and enhancing overall productivity.

Within the input grid layout, the following information is collected:

1.)Select Day: The user can choose a day of the week from a list view.

2.)Select Period: The user can choose a time slot from a list view.

3.)Mandatory Grounds of Leave: The user can enter the reason or grounds for leave in a text field.

4.)Select Date: The user can choose a date from a date picker.

The user needs to provide values for these input fields and click the "Request" button to submit the leave request. The code validates whether all the input fields are filled, and if so, it saves the leave request data separated by comma to a file named "LeaveFile.txt" and displays a success message.

This request is sent to the admin end, where the admin can either accept or reject. The details of the “Leave.File.txt” is stored in new file for further actions as per the action performed by the admin.

The module employs appropriate event handlers and input validation to ensure the entered data is processed correctly and necessary actions are performed accordingly.

TeacherNotification: The "TeacherNotification" module is an important component of a system designed to handle notifications related to leave requests

for teachers. Its purpose is to display notifications regarding the acceptance or rejection of leave requests and to send notifications to other available teachers regarding the availability of free periods. Additionally, it reserves those free periods for the teachers.

To provide a user-friendly interface, the module defines event handlers for button clicks, allowing users to interact with the system. It relies on two data files, namely "ForTeachersLeave.txt" and "user_data.txt," to gather the necessary information for generating notifications and identifying free periods. By processing and displaying this information, the module ensures that teachers are kept informed of important updates.

Consider the scenario where a teacher applies for leave, and their leave request is approved. In this case, the module generates a message specifically notifying the user about the approval of their leave. Simultaneously, other teachers who have free periods during the same time receive a message urging them to reserve the classroom for their own use.

When a teacher receives a leave approval or rejection notification, a "Confirm" button appears as part of the message. This button serves as a means for the user to provide a response. Upon clicking the "Confirm" button, the message is updated accordingly based on the user's response. For example, if the teacher confirms their acceptance of the leave approval, the message may display a confirmation message or provide additional information related to the approved leave. On the other hand, if the teacher confirms their acceptance of the leave rejection, the message might include alternative options or instructions.

By incorporating these functionalities, the "TeacherNotification" module streamlines the process of notifying teachers about leave requests, managing free periods, and facilitating their responses. It enhances communication within the system and promotes efficient utilization of resources such as classrooms.

Overall, the SignInInfo module provides a user-friendly interface for users accessibily information, saves the details, and checks for potential matches among found items. It facilitates the process of allocating proper image.

4.2 MOTIVATION OF THE MODULE.

The module plays a crucial role in offering convenient and interactive functions for users. It ensures that data is stored and utilized in a responsible manner by the

administrator. Based on the available data, the module facilitates the provision of services to the users.

Ease of Use: The module utilizes a user-friendly graphical interface with clearly labeled input fields and controls. This design approach simplifies the reporting process, making it accessible even to users with limited technical skills. The intuitive layout and familiar form-like structure help users navigate and fill out the required information effortlessly.

Data Collection: The module collects various details about the date, time, slot. This comprehensive data collection ensures that users can provide as much information as possible, facilitating the matching process and assisting the user with accurate details.

User Interaction: By offering a dedicated module for providing information, the application empowers users to take an active role in the updating process. It allows them to provide detailed information about the date of leave, and also allows to allocate a slot for the free user according to their own wish.

Integrated Notification System: This module send the user notifications according to the admins answer. This integration ensures that users are promptly notified of any potential matches, providing them with valuable information and facilitating efficient communication between users and the application.

4.3 RELEVANCE OF THE MODULE IN THE SYSTEM:

The module plays a crucial role in the system by facilitating efficient communication and interaction between the users and the system. It serves as a bridge, connecting the users' actions with the system's responses.

Firstly, the module's ability to process and display information from the "ForTeachersLeave.txt" and "user_data.txt" data files is vital for the system's operation. It ensures that teachers are kept up-to-date with important updates, such as leave approvals or rejections. This feature enhances the system's transparency and allows for a smoother workflow.

Secondly, the module's event handlers for button clicks enable users to interact with the system effectively. For instance, the "Confirm" button allows users to respond to notifications, thereby promoting active user engagement.

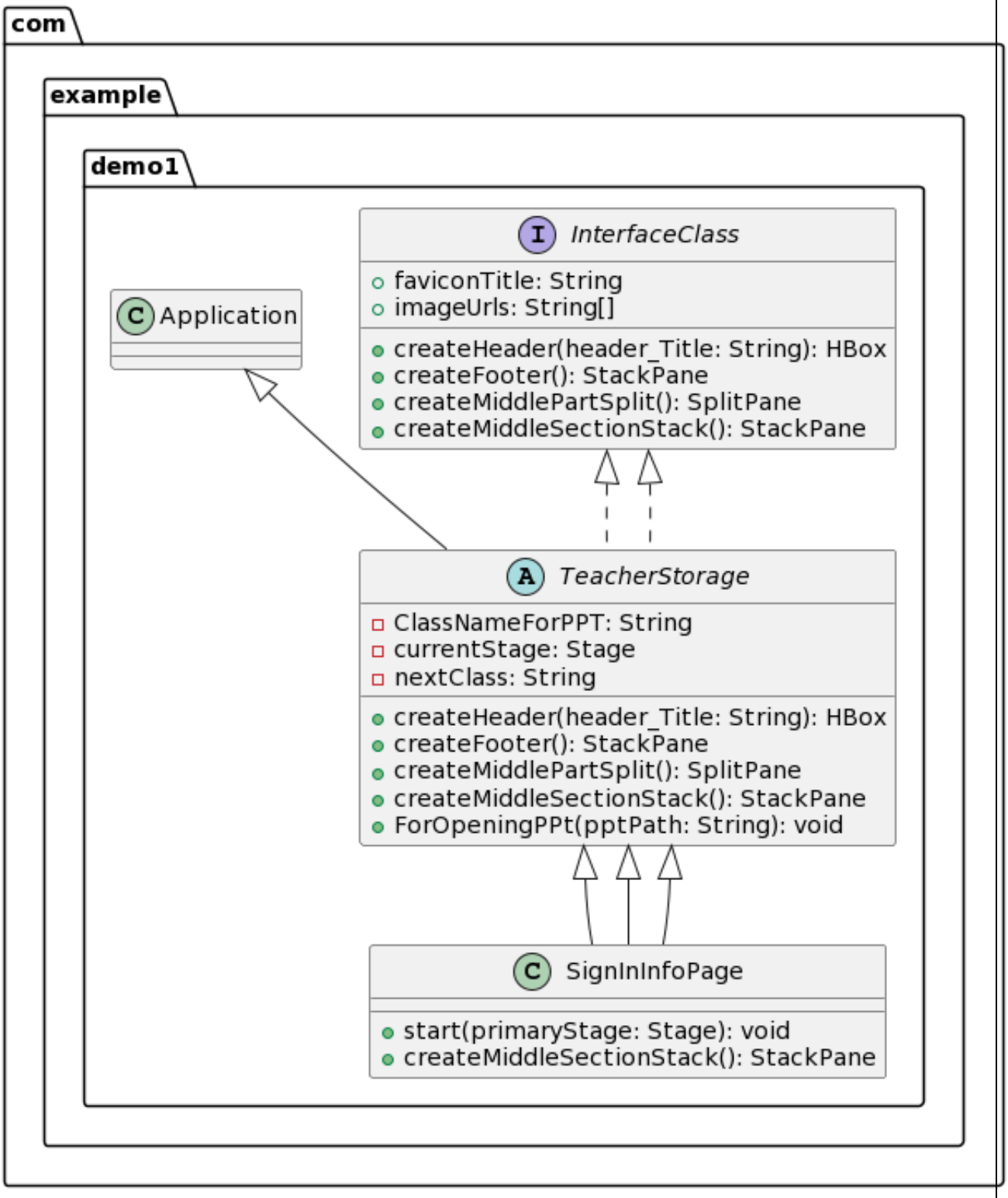
Thirdly, the module's user-friendly graphical interface enhances the system's usability. It simplifies the reporting process, making it accessible to users with varying technical skills. This feature is particularly relevant in a diverse user environment, ensuring that all users can navigate and use the system with ease.

Lastly, the integrated notification system is a key feature of the module that enhances the system's efficiency. By sending user notifications based on the admin's responses, the module ensures that users are promptly informed of any updates. This feature not only provides users with valuable information but also facilitates efficient communication within the system.

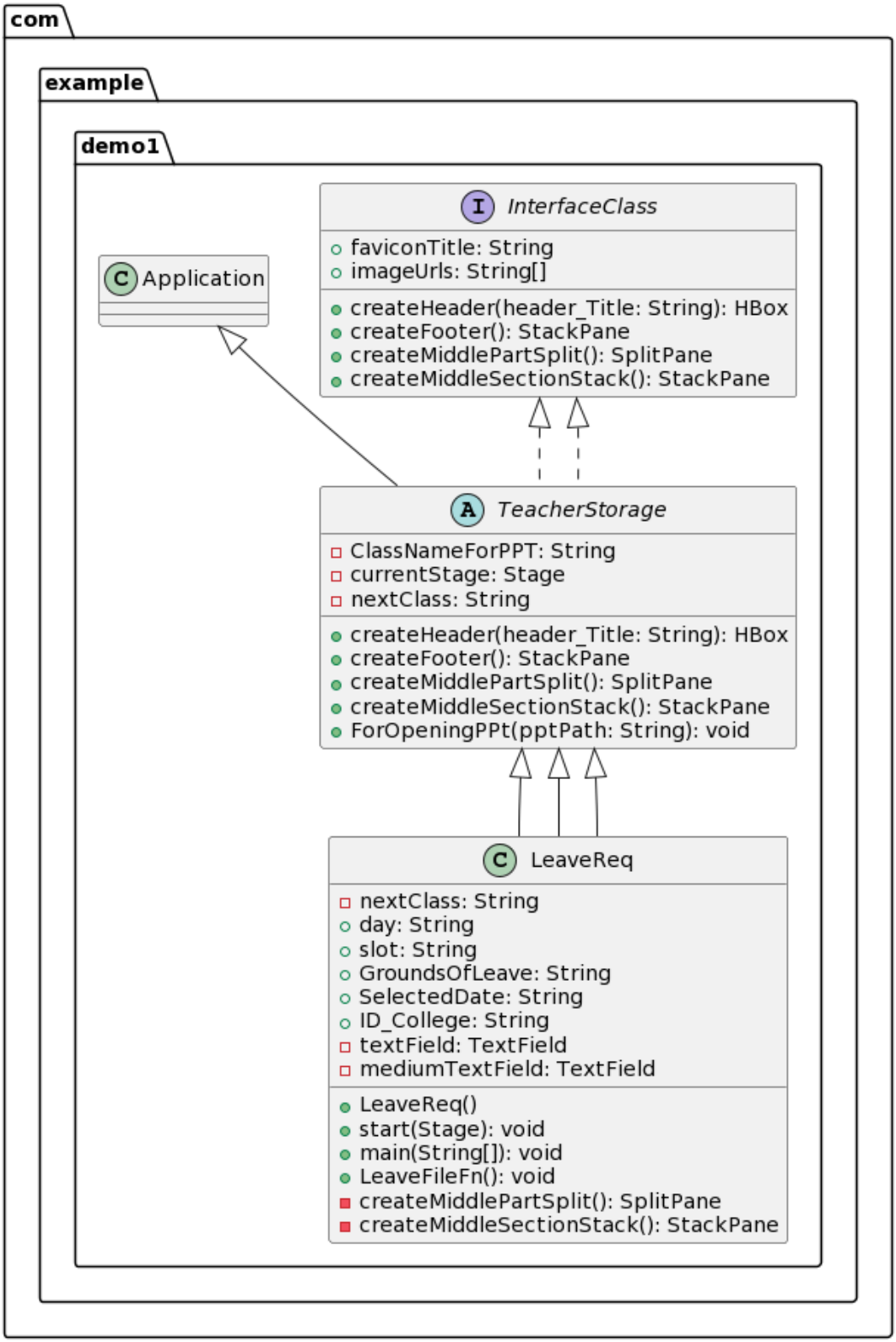
In conclusion, the module's relevance in the system lies in its ability to facilitate user-system interaction, enhance system transparency, promote user engagement, improve system usability, and ensure efficient communication.

4.4. UML OF THE MODULE

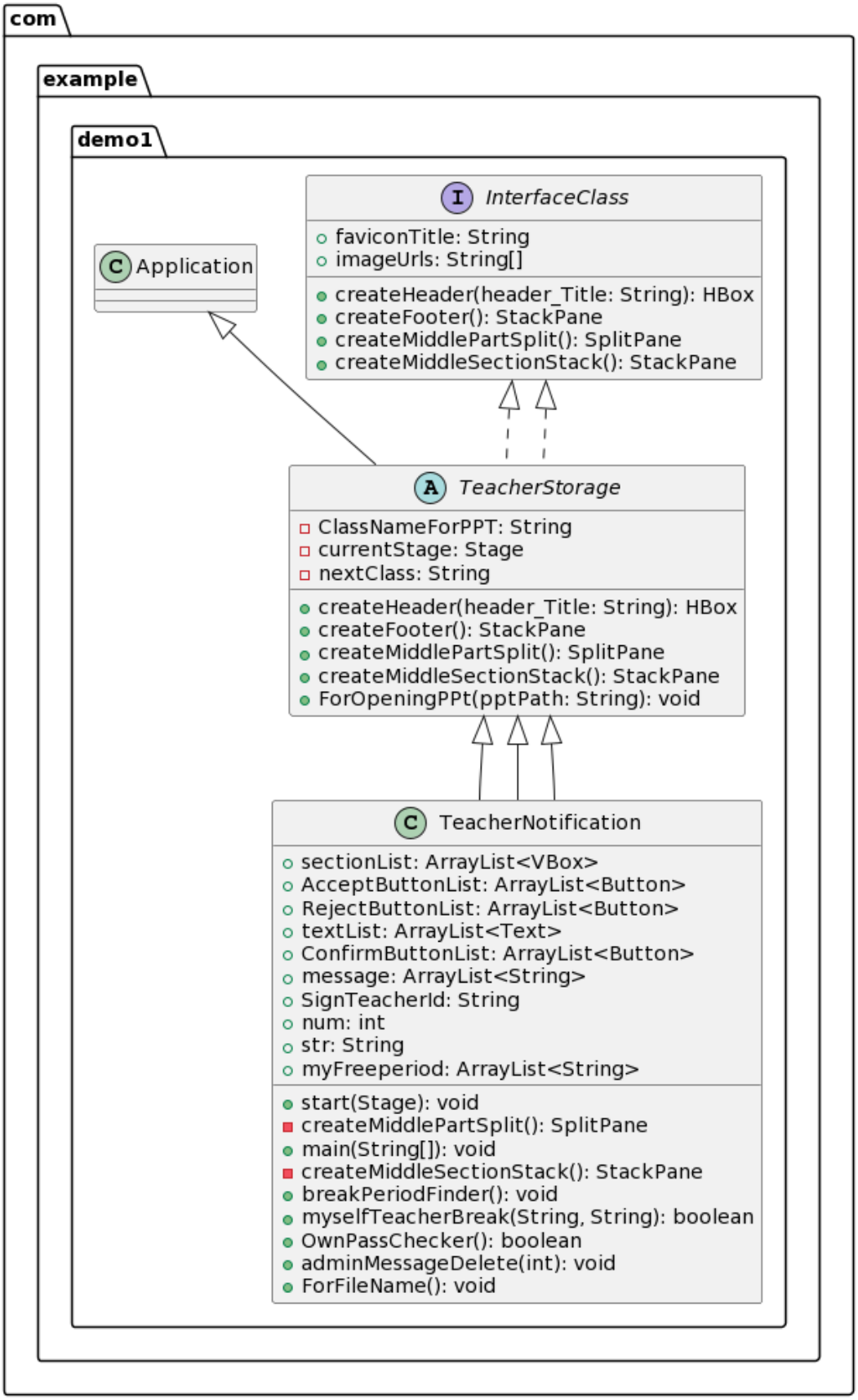
4.4.1. UML of “SignInInfoPage” Class.



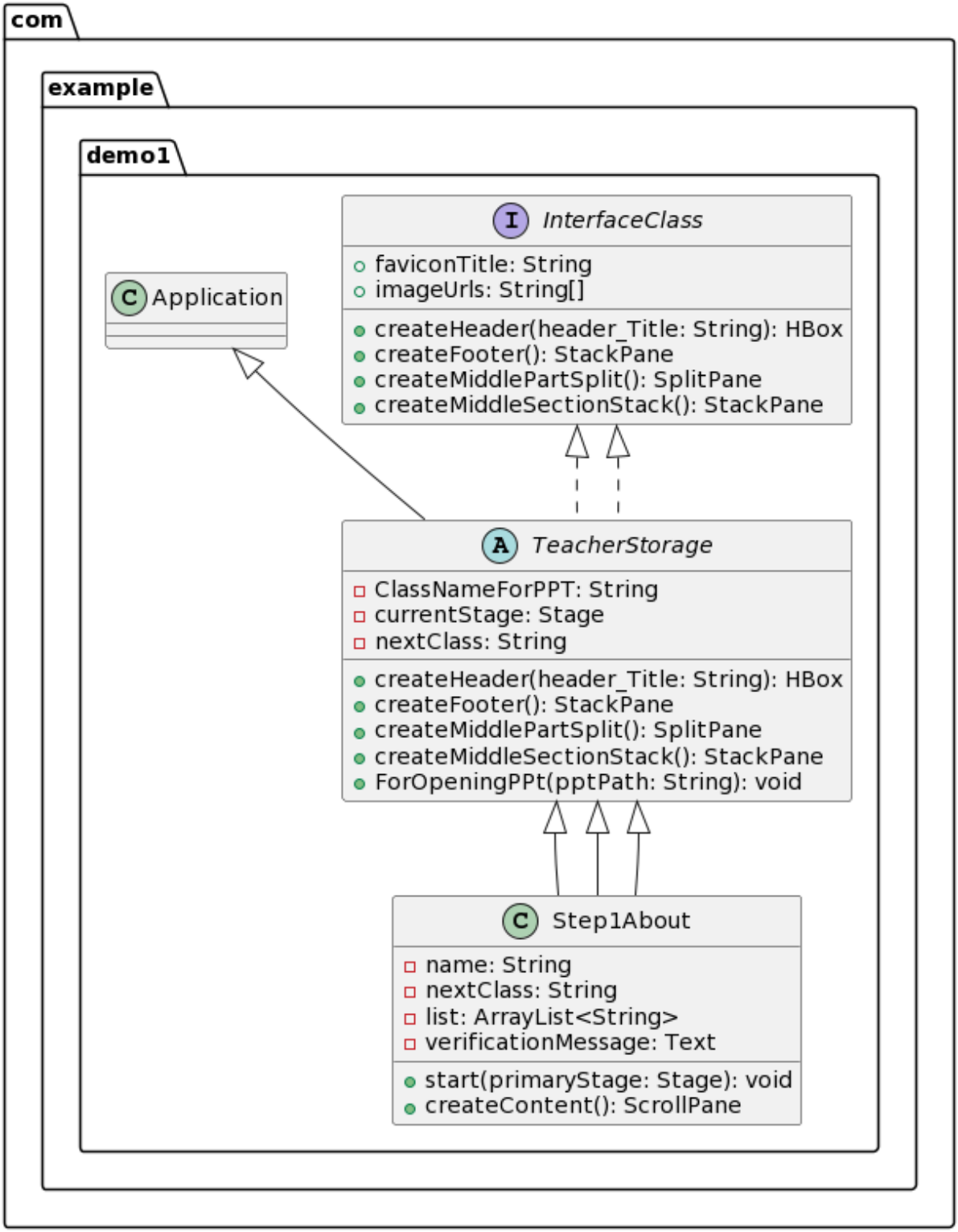
4.4.2. UML of “LeaveReq” Class



4.4.3. UML of “TeacherNotification” Class.

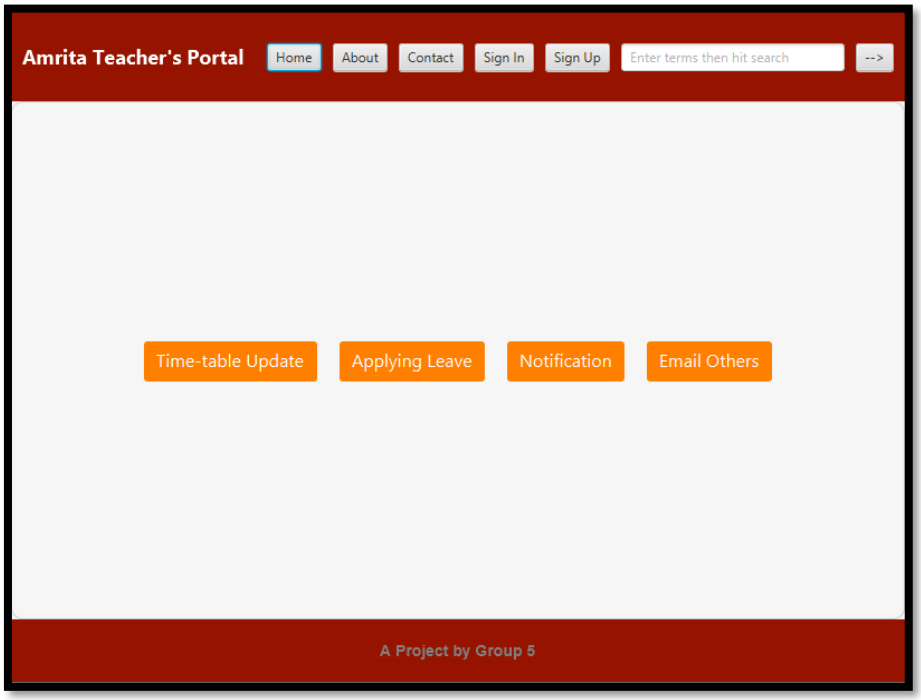


4.4.4. UML of “Step1About” Class.

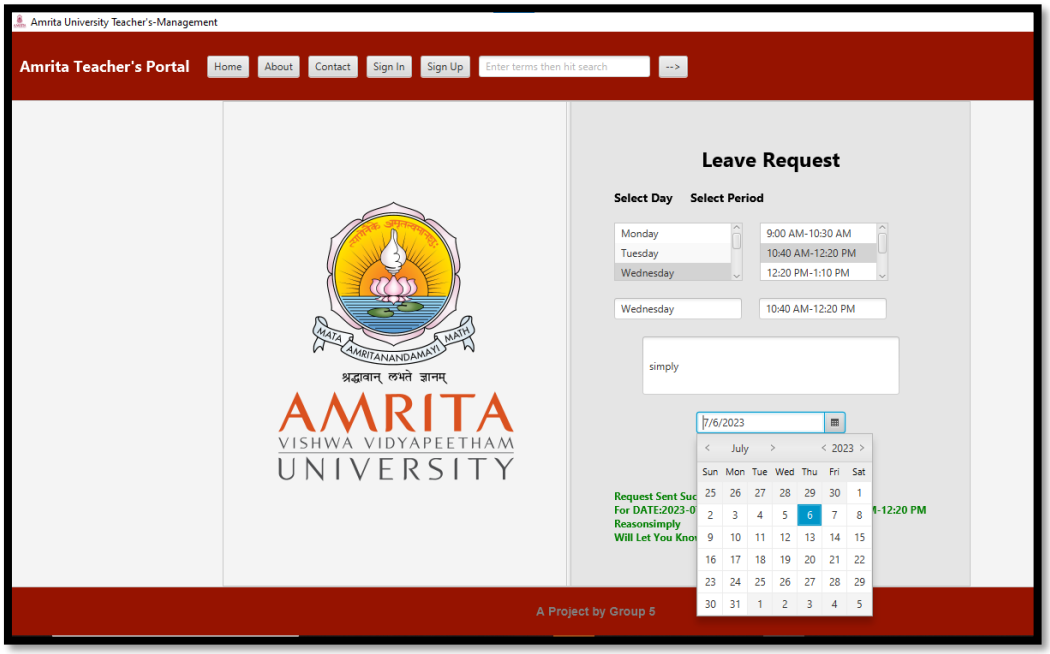


4.5. OUTPUT OF THE MODULE

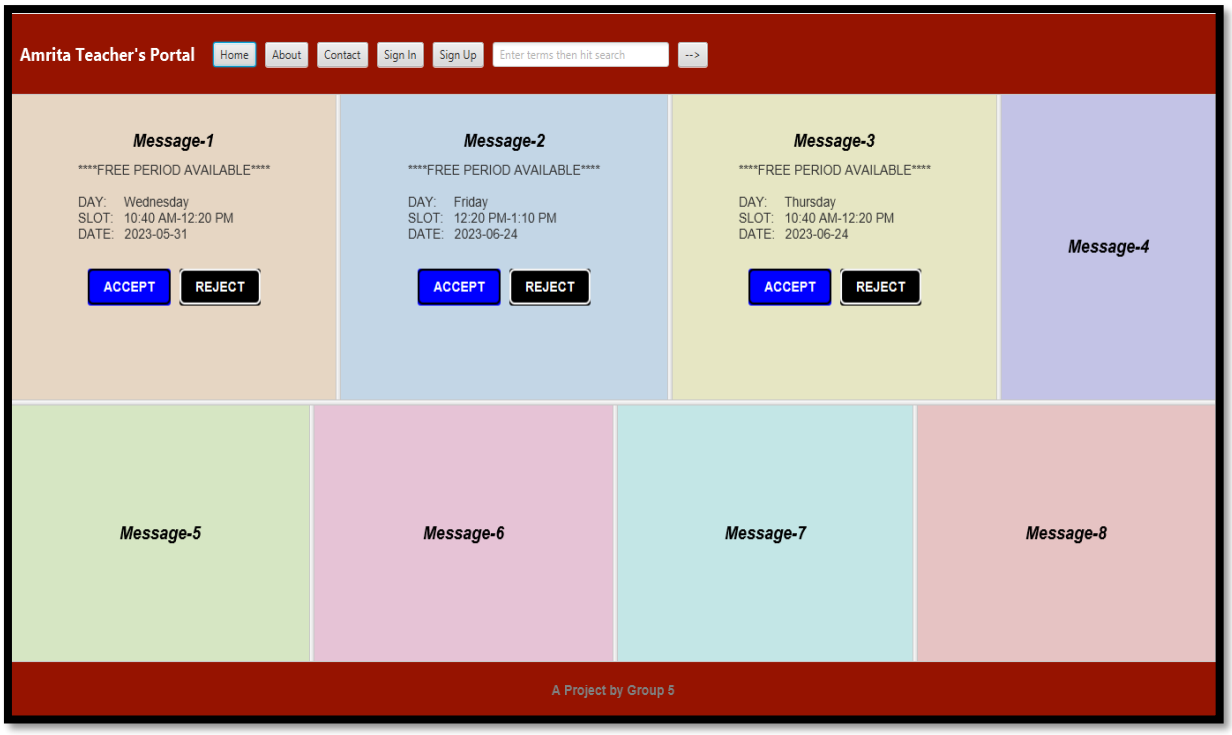
4.5.1. SignInInfoPage



4.5.2. LeaveReq



4.5.3. TeacherNotification



4.5.4. Step1About