Dipartimento di Ingegneria Civile e Ingegneria Informatica

Ingegneria del software e progettazione web

Final Project Detailed Instructions

Professori:
Davide Falessi
Guglielmo De Angelis

Partecipanti:
Mattia Biga

A.A 2023-2024

Contents

Chapter 1

Software requirements Specification

1.1 Introduction

1.1.1 Aim of the document

This document aims to provide a comprehensive overview of the Time Plan application design process, detailing the various design choices created to ensure that the app meets the specified requirements. By outlining these aspects, the document shows a precise understanding of how the design aligns with software engineering principles.

1.1.2 Overview of the defined system

Time Plan is a desktop application designed to manage work shifts for any business. It facilitates the assignment of shifts to employees, aiming to simplify and streamline shift scheduling. The application is intended for daily use, providing an intuitive and efficient tool for managers to allocate shifts and ensure optimal coverage.

1.1.3 HW e SW requirements

Hardware:

- RAM: At least 2 GB of RAM, although 4 GB or more would be preferable to handle memory

usage peaks.

- CPU: A modern processor, at least a dual-core, although a quad-core might be more suitable for effectively handling multithreading.
- Hard Disk: Sufficient space to install the JVM, and the application, and manage temporary files and logs. At least 500 MB of free space.
- Operating System: macOS 14.5 or higher, compatible with the version of Java in use (Java 21).

Software:

- Visual paradigm for diagram creation
- IntellijIdea Version 17.0.10, for app development
- SceneBuilder for building GUIs

1.1.4 Related system, Pros and Cons

TSheets is a time-tracking and employee-scheduling application that helps businesses manage their workforce effectively.

- Pros: TSheets excels in time tracking, allowing employees to log hours, track breaks, and manage overtime accuracy.
- Cons: The extensive features may be overkill for companies looking for simple manager-employee communication tools without the need for detailed time tracking.

ZoomShift excels in scheduling, allowing managers to create, modify, and manage employee shifts with ease.

- Pros: ZoomShift offers mobile apps that allow employees to view their schedules, request time off, and manage shifts from their smartphones.
- Cons: The primary focus on scheduling and time tracking might be overkill if your main need is to enhance communication rather than manage shifts.

1.2 User story

- As a Manager, I want to generate work shift scheduling for employees, so that each employee knows their assigned shifts.
- As an Employee, I want to see my work shift scheduling, so that I know when to work.
- As an Employee, I want to select a day of absence, so that I can warn my manager of my absence.

1.3 Functional Requirements

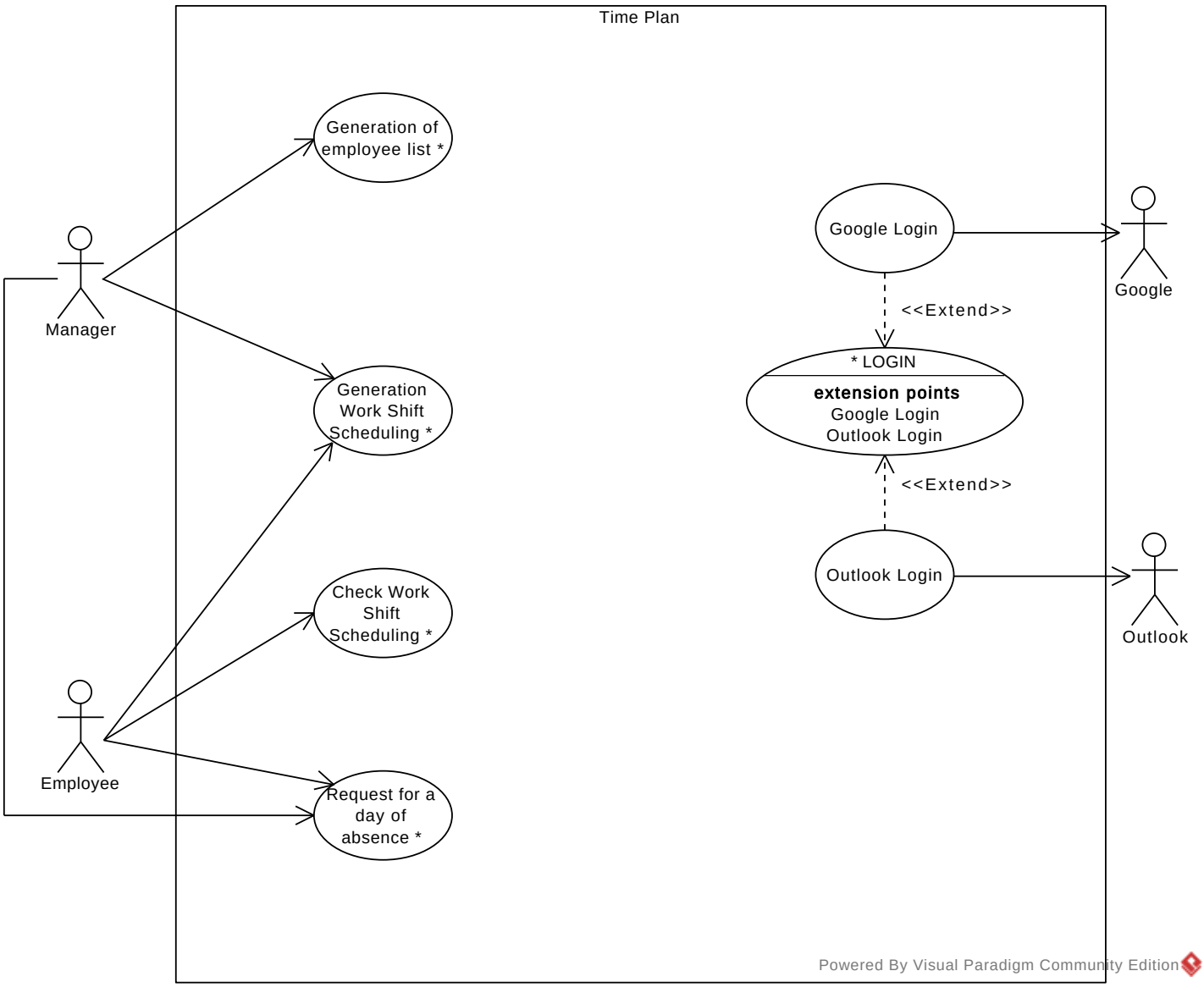
- The system shall generate the calendar for the current month.
- The system shall display the list of all employees.
- The system shall provide three types of login: owner login with email and password, login with Gmail account and login with Outlook account.

1.4 Use Cases

1.4.1 Overview Diagram

Google and Outlook Log In not implemented.

Request for a day of absence not implemented.



1.4.2 Internal steps

Name: *Request for a day of absence*

1. The employee requests a day of absence.
2. The system loads the current month's calendar.
3. The employee selects a day from the calendar.
4. The system verifies that the employee has all permissions for that day of absence.
5. The system prepares an empty motivation form.
6. The employee indicates the motivation form is complete.
7. The system saves the form.
8. The system notifies the manager.

Extensions:

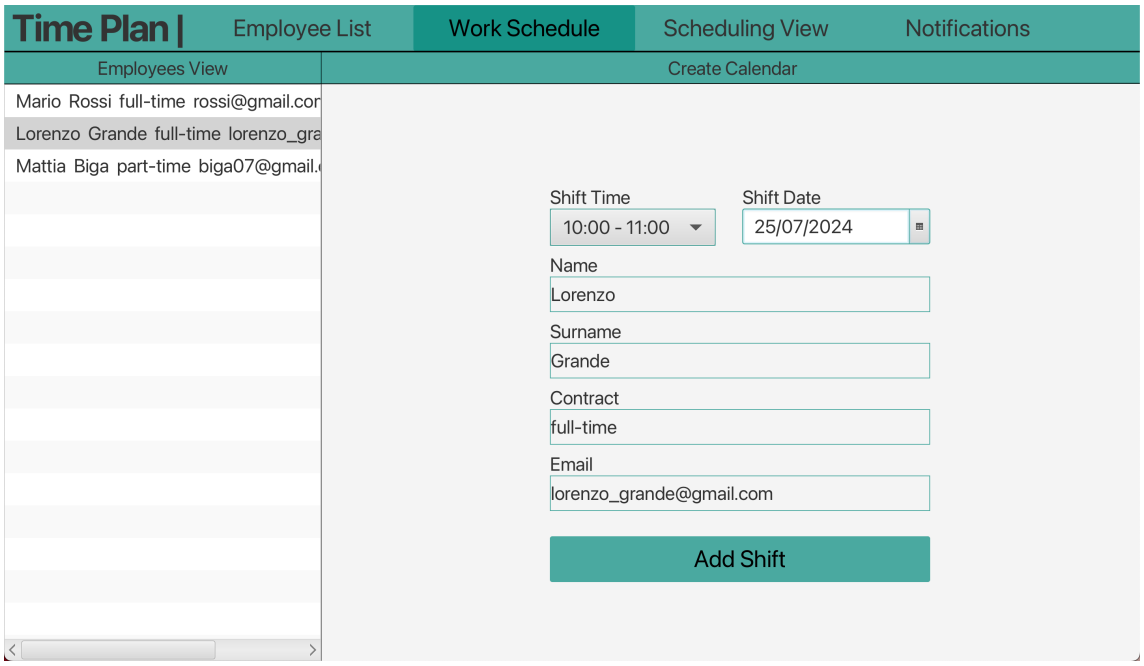
2a. *The system has already loaded the current month's calendar:* The system brings up the previous calendar instead of loading a new one.

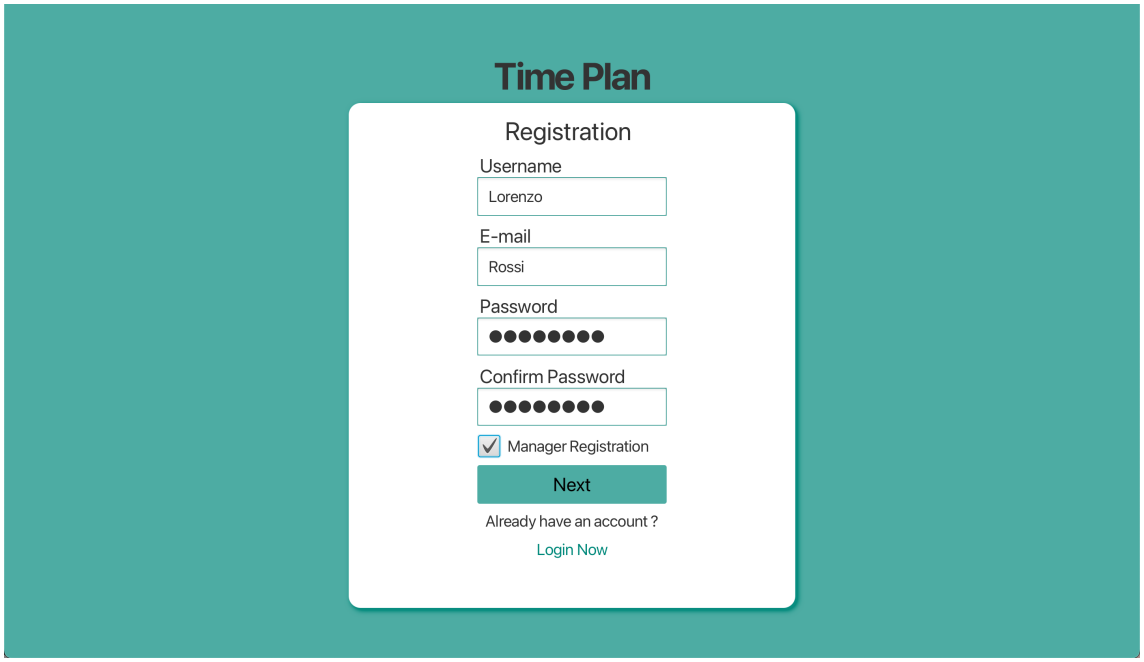
4a. *The employee has not all permissions:* The system notifies the employee and returns to the day selection.

7a. *The save fails:* The system notifies the employee and terminates the use case.

Chapter 2

Storyboards




Time Plan
Registration
Username
Lorenzo
E-mail
Rossi
Password
Confirm Password
Manager Registration
Next
Already have an account ?
Login Now

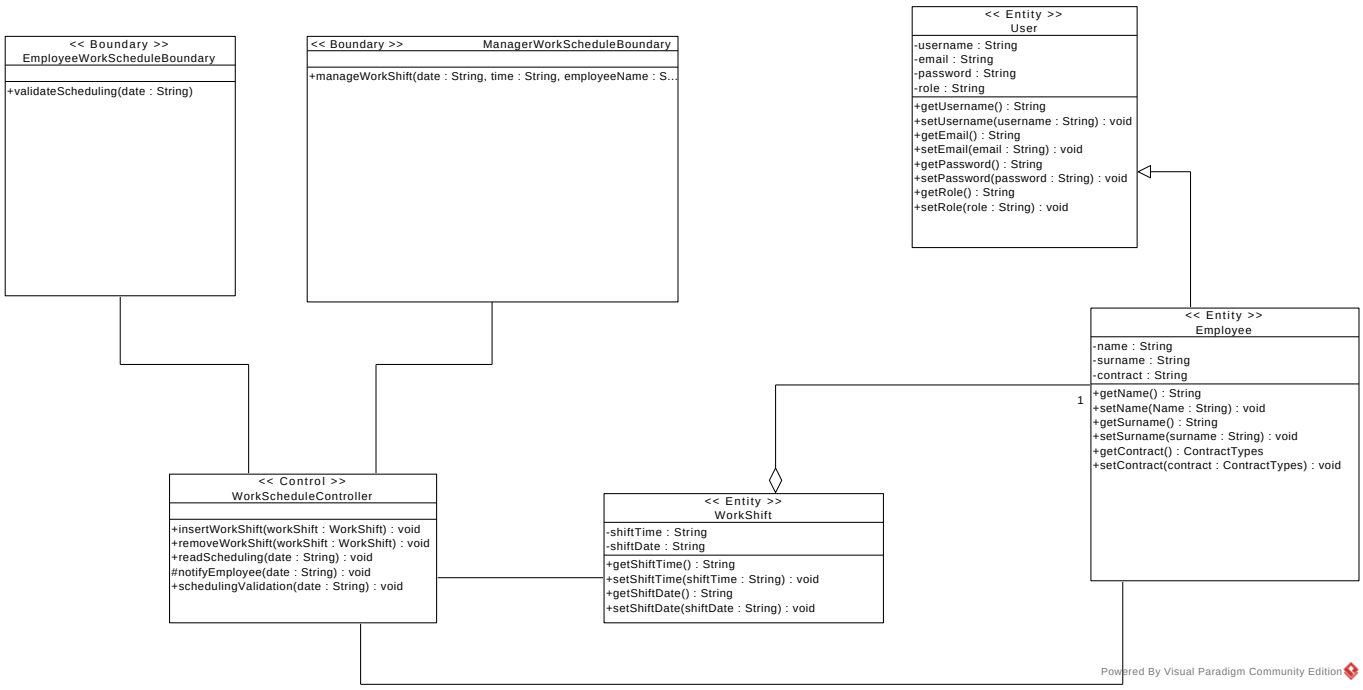
Chapter 3

Design

3.1 Class Diagram

3.1.1 VOPC

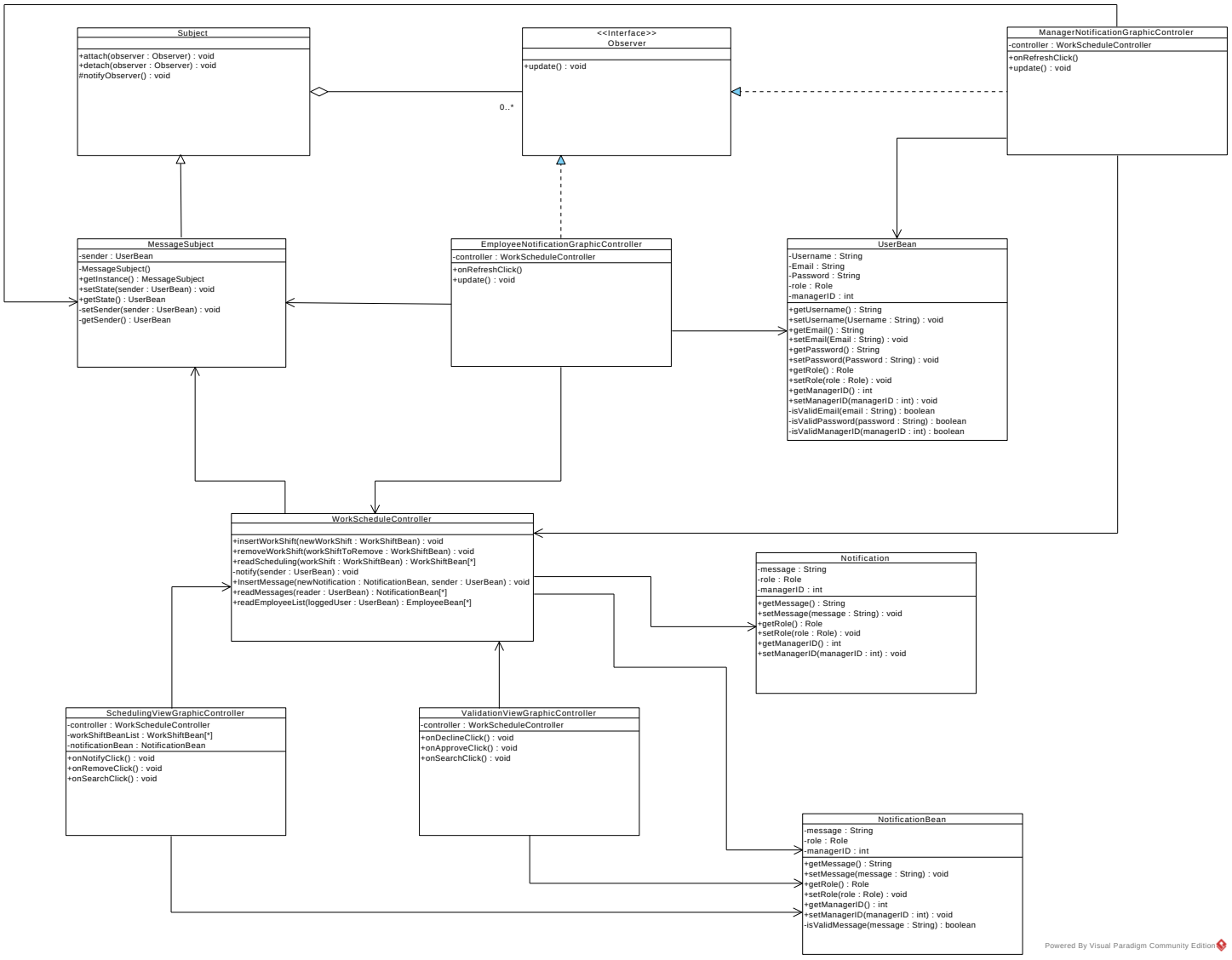
View of participating class about generation work shift scheduling use case



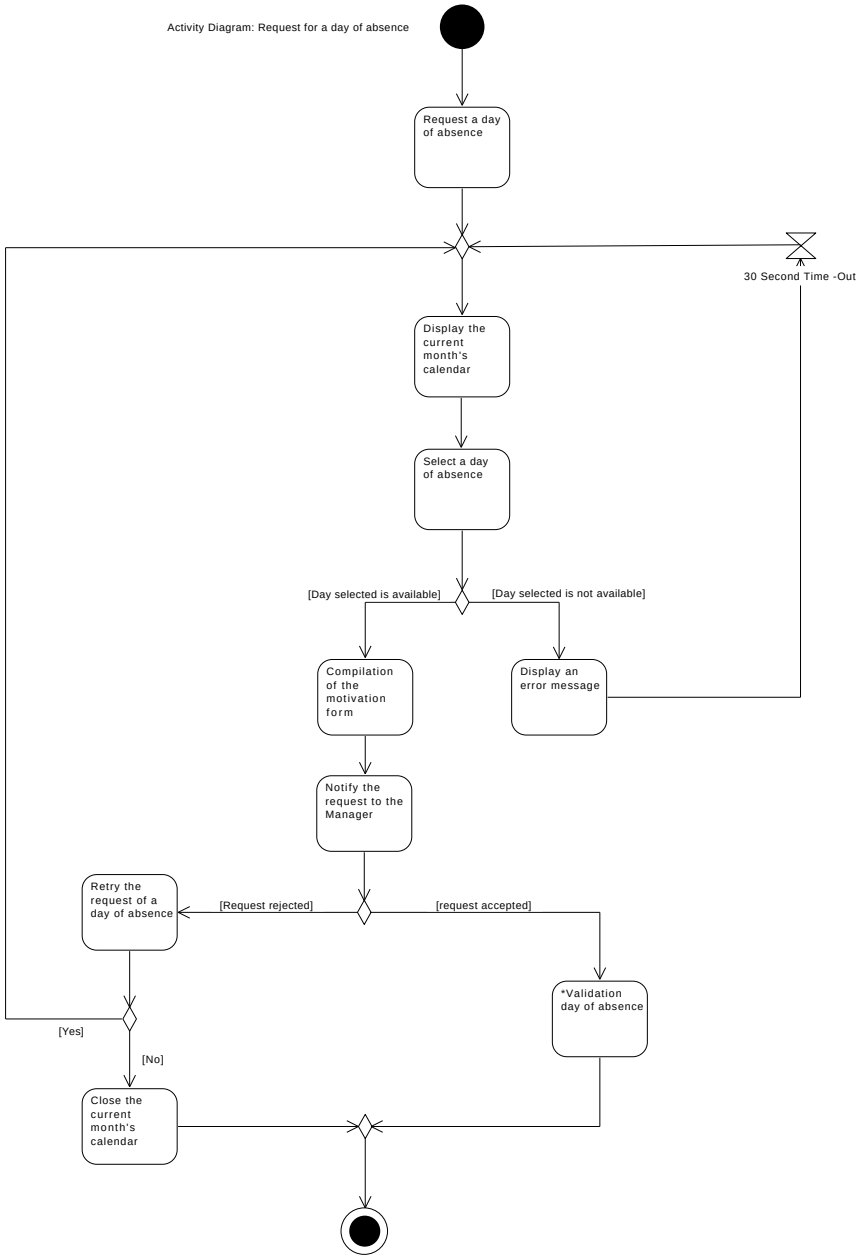
3.2 Design Pattern

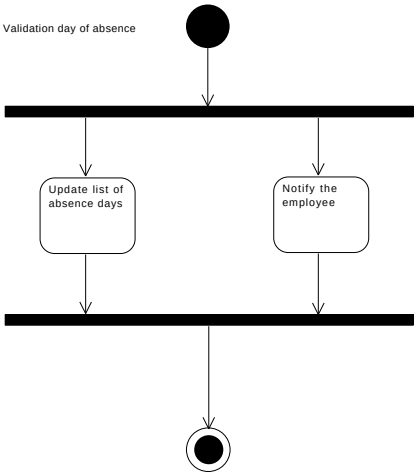
3.2.1 Design Level Diagram

Design level diagram about observer pattern

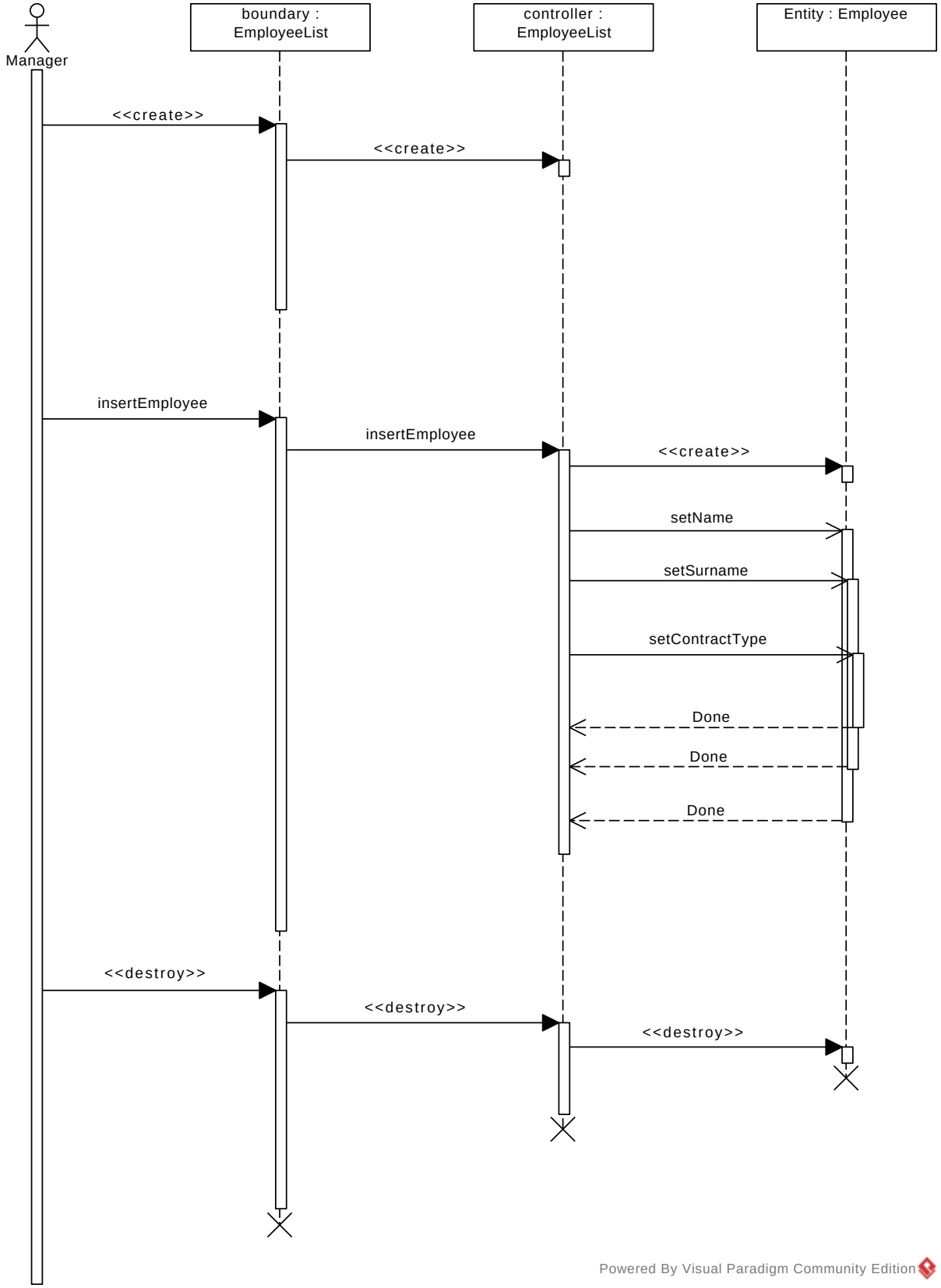


3.3 Activity Diagram

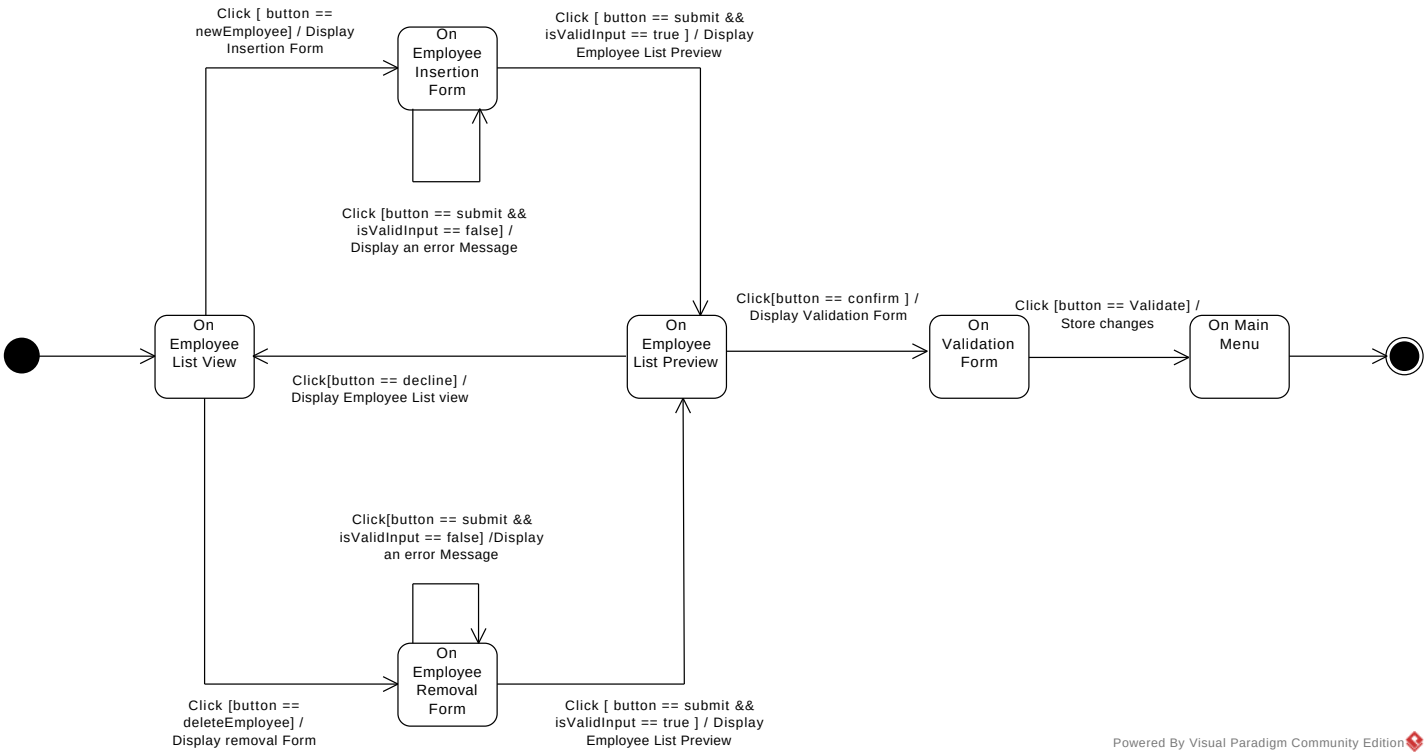




3.4 Sequence Diagram



3.5 State Machine Diagram



Chapter 4

Testing

4.1 Test Cases

- TestEmployeeListController
- TestLoginController
- TestRegistrationController