

CS3354

Homework Calendar App

Payson Eberz | Van Luong | Abuzar Naqvi | Josh Basker | Mattia Moren |
Rishikeshan Bharathi | Andrew Pham | Ali Bidhendi

Project Introduction/Objective

In this project, we will be utilizing software engineering techniques to build a Homework Planner application, which is designed to help users keep track of their assignments and other possible date/event managing with other features such as notifications and integration with university systems.

The objective of this project is to alleviate the stress and confusion that accompanies academic planning with a tool that simplifies the process to put more effort into learning and not worrying or organizing over assignments.

Project Timeline / Scheduling

- Development time = roughly 6 days
 - Begin development May 13th, end May 20th
 - Full time, excluding weekends
 - 5 person team, 50 units of work per week

Cost Estimation

- Development = \$8,400 (Derived using Function Point Model)
 - 50 work per week
 - 5 person dev team
 - \$35/hour (minimal experience required)
- Hardware = \$0
 - Offline app, no servers
- Licenses = \$0
 - Free, open source licenses sufficient
- Personnel = \$0
 - Product releases as-is, minimal to no post-launch maintenance

Functional Requirements

- Add events/assignments/deadlines to their calendar (specifies title, course name, type (test/hw/event/deadline etc.), submission deadline/date, time, description, location). Some fields listed can be null/not applicable.
- Delete events/assignments/deadlines.
- Scroll through their calendar to any date they wish, with options to view in monthly, daily, and yearly modes.
- Receive notifications for events/assignments/deadlines they've created.
- Search for events/assignments/deadlines on their calendars
- Import events/assignments/deadlines from their school's platform calendar (i.e., eLearning calendar) if applicable.
- Link their calendars with friends to see how their schedules match up.

Non-Functional Requirements

Efficiency: Load any view within 2 seconds

Dependability: Uptime of 99.9%, ensuring that users can access their schedules and assignment information almost anytime

Security: Encrypted user data (personal schedules, course details, assignment information) using industry-standard protocols

Regulatory: Comply with the General Data Protection Regulation (GDPR)

Ethical: Not use user data for advertising purposes without explicit consent

Usability: Be accessible to users with disabilities, complying with the Web Content Accessibility Guidelines (WCAG) 2.1 Level AA standards

Performance: Support simultaneous access by up to 10,000 users without significant degradation in response times

Space: Require no more than 150MB of space on the device after installation

Environmental: Prioritize energy-efficient processes to minimize its carbon footprint, including optimizing server usage and encouraging digital over physical task management

Operational: Compatible with both iOS and Android platforms

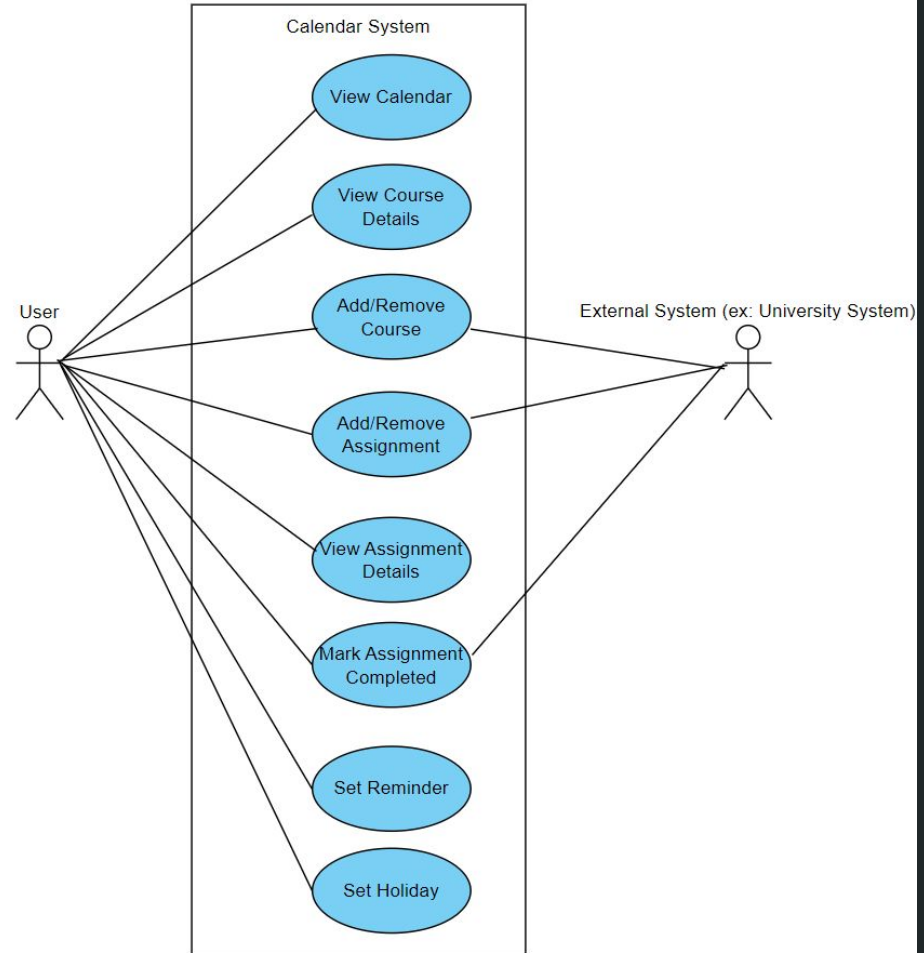
Development: Incorporate continuous integration and continuous deployment (CI/CD) practices to streamline updates and bug fixes

Legislative: Adhere to the Children's Online Privacy Protection Act (COPPA) if it is to be used by children under the age of 13

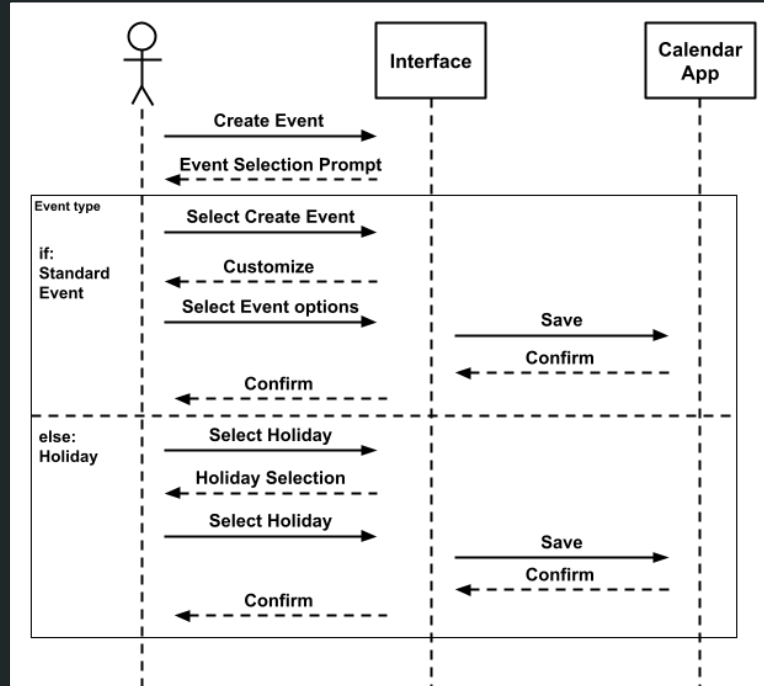
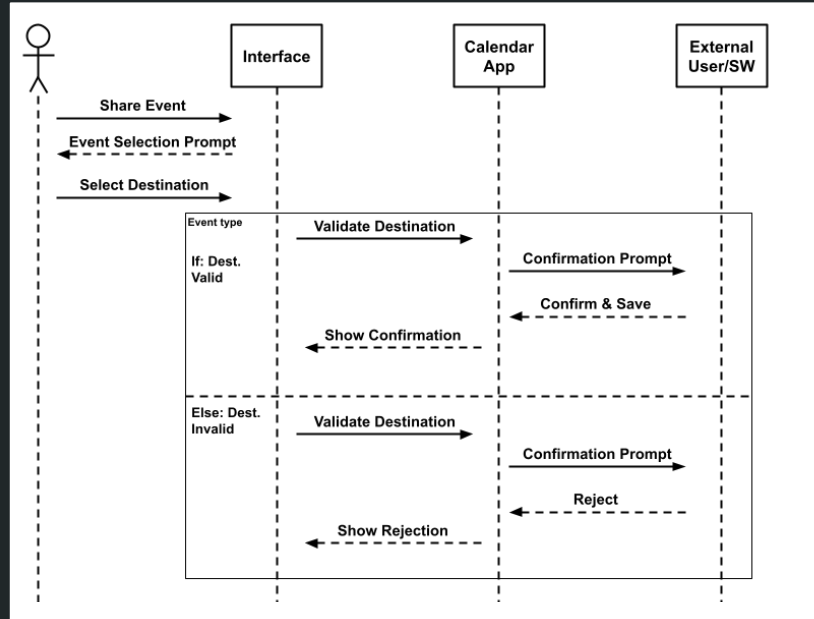
Accounting: Provide detailed logging of user transactions and interactions for auditing purposes

Safety/Security: Include robust authentication mechanisms, such as two-factor authentication

Use Case Diagram

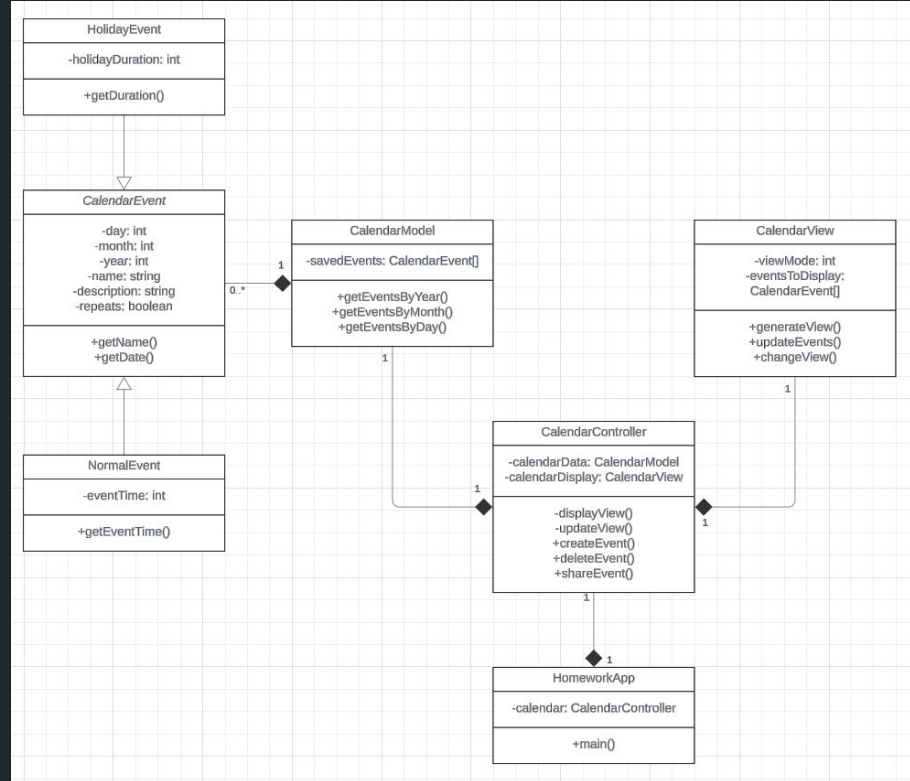


Sequence Diagram



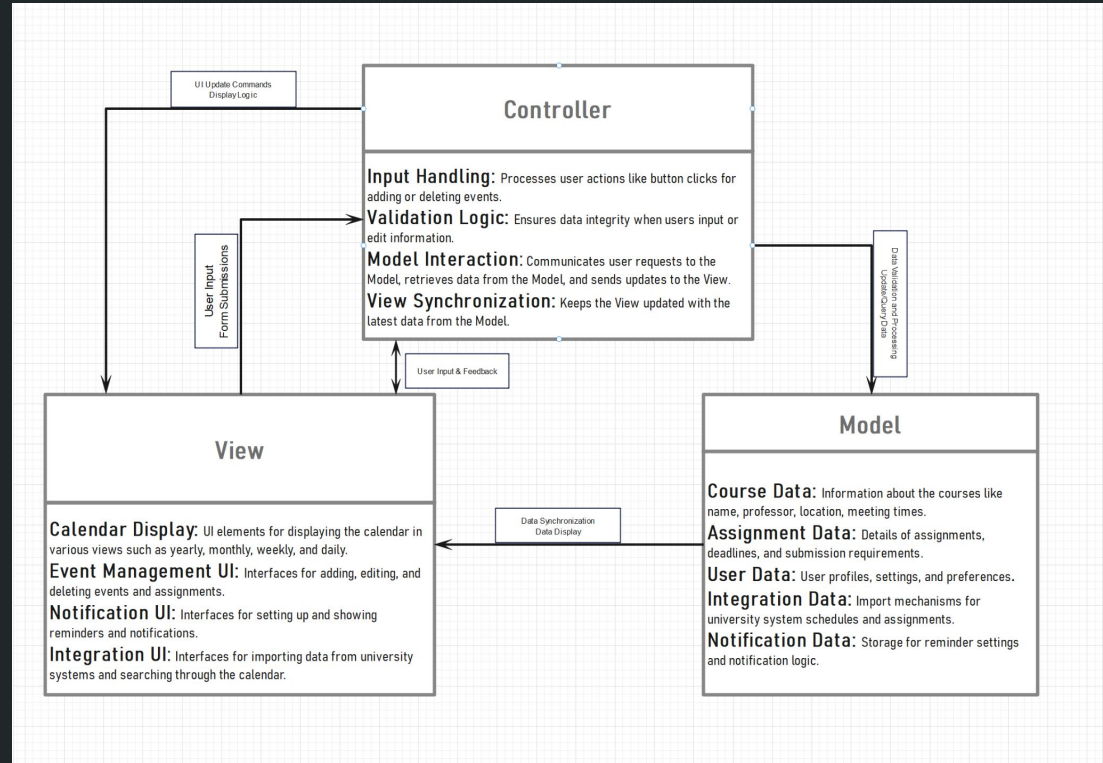
Class Diagram

- Event type flexibility



Architectural Design

- Model View Controller (MVC)



User Interface Design



Colors



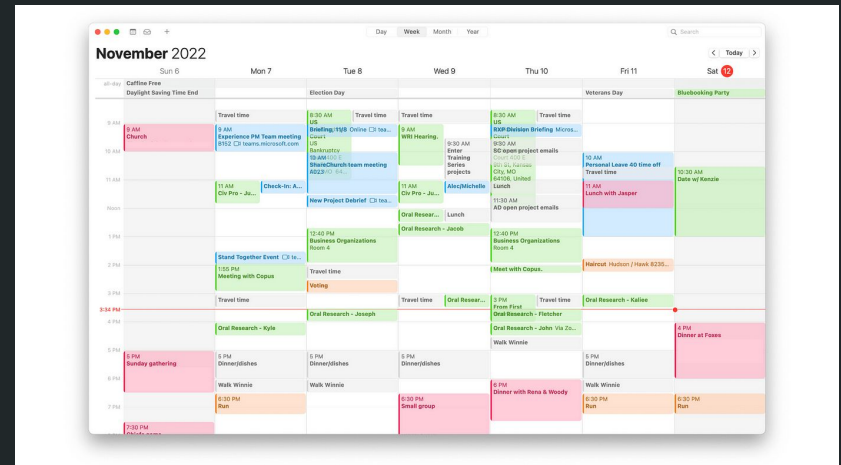
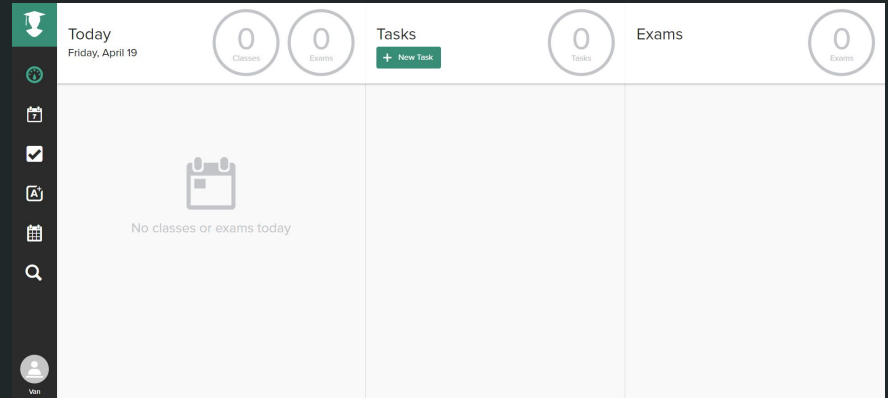
Similar Work / Uniqueness

1. Apple Calendar

- Similarities: daily/monthly/yearly view, being able to add events with a specific date and time
- Differences: our app caters toward students, while Apple's calendar is more general

2. MyStudyLife

- Similarities: caters toward students
- Differences: MyStudyLife is web based, it has a progress circle, tasks and exams are each its own section, and they opted for a more simplistic color scheme.



Conclusion

- Throughout the development process, we remained receptive to feedback, ensuring that our app stood out and tackled problems faced by university students.
- We opted for the Prototyping Model to afford us the flexibility necessary for iterative refinement.
- Despite encountering minor deviations from our initial roadmap, such as adjustments to feature prioritization and user interface enhancements, each decision was guided by our desire to meet the user's needs effectively.
- The collaborative effort among team members was integral to our success, with each member contributing their expertise and dedication to different facets of the project.
- Looking ahead, our focus remains on continuous improvement and user-centric innovation.