# Calendar Software

Team 4


Timur Esenaliev

Daniel Hirsh

Wayne Le

Divya Narayan

Baylor Resnick

Meenakshi Sundar Rajan

Lilian Zeng

# Project Objective

The objective of the project is to develop a Calendar Software capable of the standard calendar functions:

1. Inserting events
2. Deleting events
3. Editing events
4. Setting reminders
5. Calendar viewing
6. Calendar sharing

# Cost estimation

Function Point Method for cost estimation:
GFP = 109
PCA = 0.65 + 0.01(42) = 1.07
FP = 109 * 1.07 = 117
Effort = 117 * 5 = 585 hours
Development Cost = 585 * $60/hr = $35,000

| Component | Count | Complexity | FP Weight | GFP |
|---|---|---|---|---|
| External Inputs | 6 | Medium | 4 | 24 |
| External Outputs | 5 | Medium | 5 | 25 |
| External Inquiries | 4 | Medium | 4 | 16 |
| Internal Logical Files | 3 | High | 10 | 30 |
| External Interface Files | 2 | High | 7 | 14 |

Total GFP – 109

# Cost estimation

With a 5-person team working 40 hr weeks, project length should be 585/200 = 3 weeks

| **Hardware Cost estimate:** | **Cost** |
|---|---|
| Cloud Server (AWS EC2) | $200 |
| Database Server (AWS RDS) | $300 |
| CI/CD Build Server | $300 |
| Backup & storage | $400 |

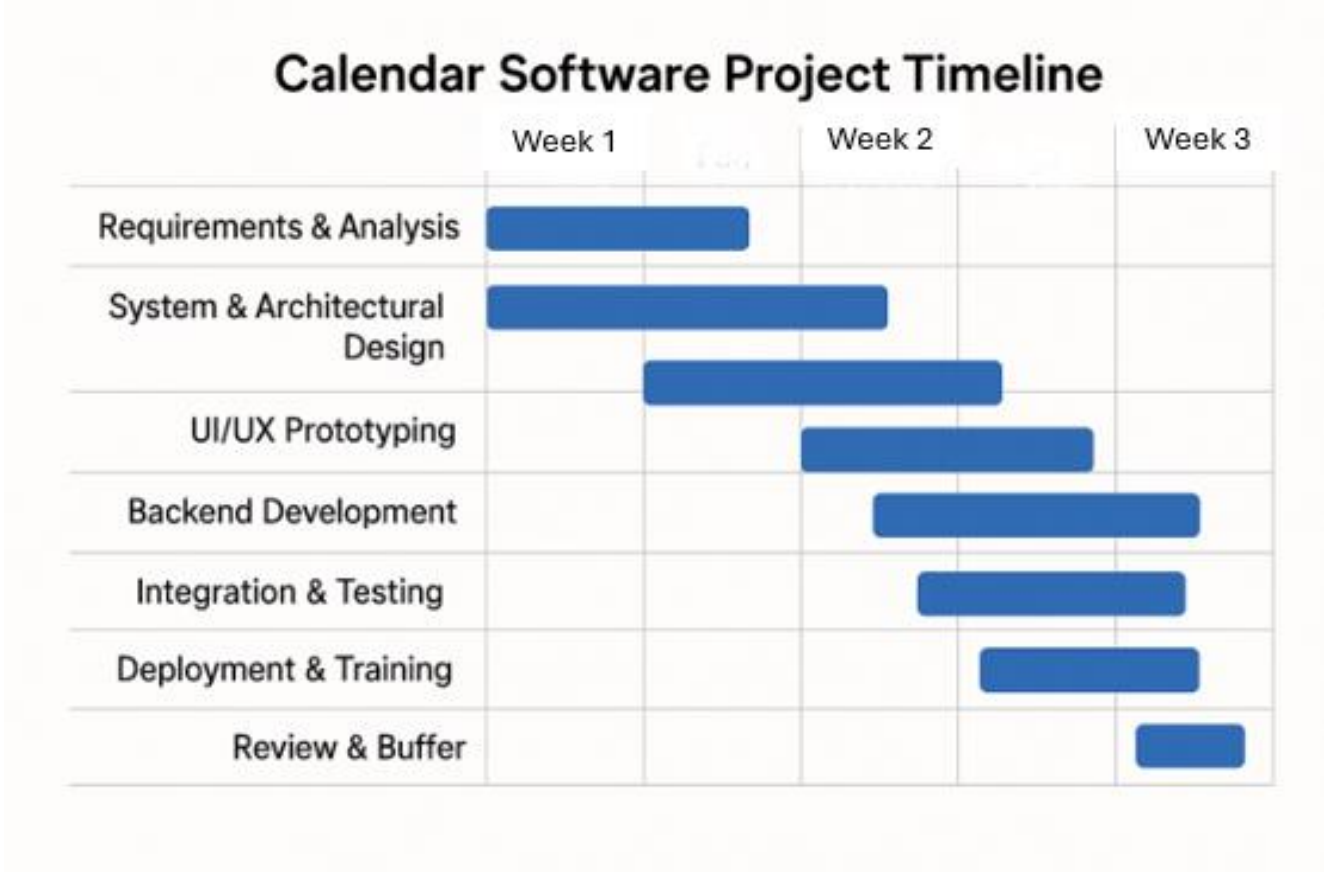
| **Software cost estimate:** | **Cost** |
|---|---|
| JetBrains License (team) | $150 |
| Figma Pro (team) | $250 |
| Email Notification API (SendGrid) | $100 |
| Monitoring Tools (Datadog) | $100 |

| **Personnel cost estimate:** | **Weekly Salary** | **Weeks** | **Cost** |
|---|---|---|---|
| 1 Project Manager | $2,500 | 3 | $7,500 |
| 2 Full-stack Developers | $2,125 each | 3 | $12,750 |
| 1 UI/UX Designer | $1,750 | 3 | $5,250 |
| 1 QA Engineer | $1,625 | 3 | $4,875 |
| Training | | | $3,000 |

| **Category** | **Cost** |
|---|---|
| Development Cost | $35,000 |
| Hardware Cost | $1,200 |
| Software Cost | $600 |
| Personnel Cost | $33,375 |
| **Total Cost** | **$70,175** |

# Project timeline

- Start Date: January 5th, 2026
- End Date: January 23rd,2026
- Total Duration: 3 Weeks
- Assumption: Weekdays M-F, 9am-5pm
- The Scrum process can be utilized for efficiency.



# Functional and non-functional requirements

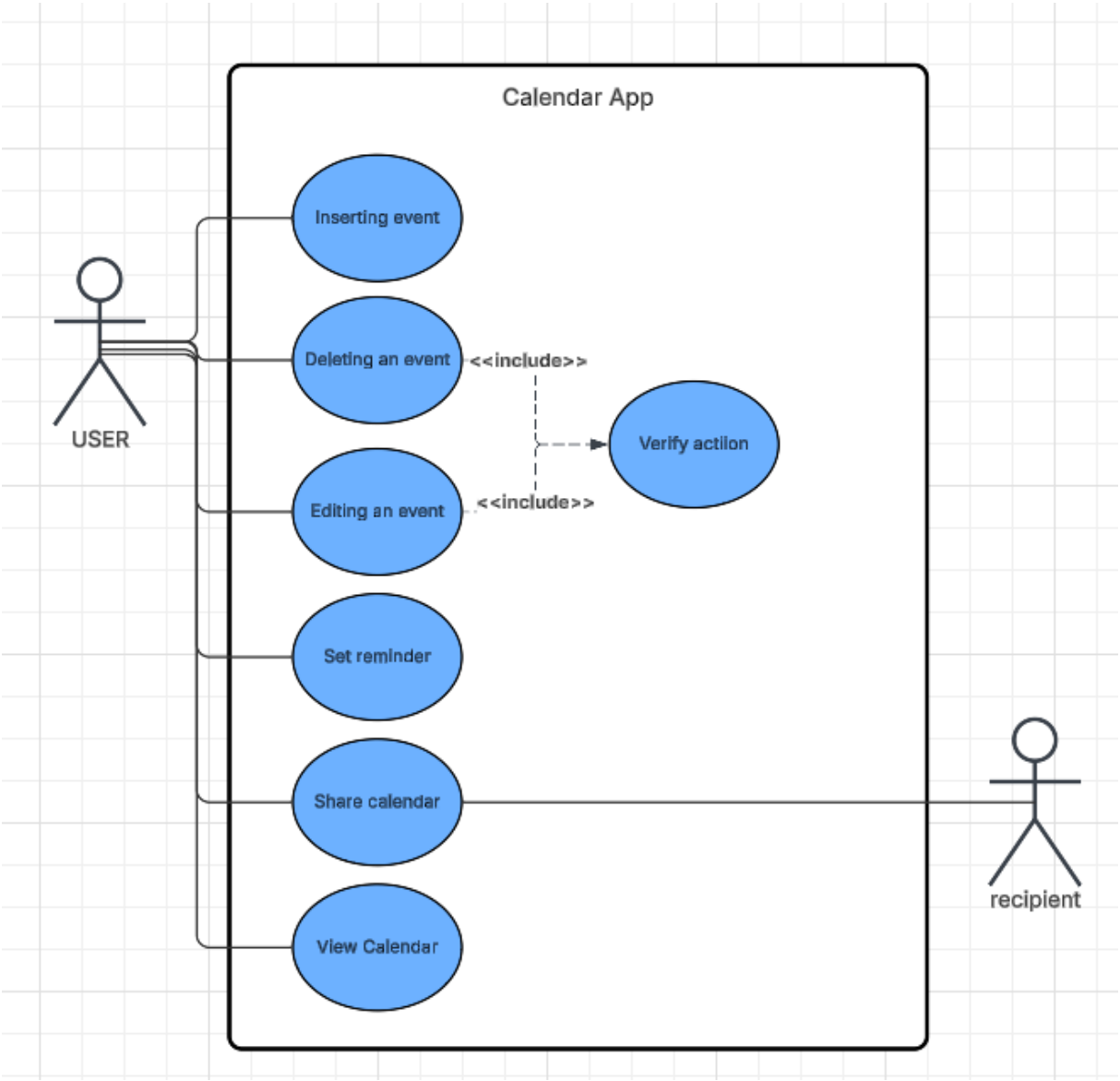
## Functional Requirements:

- Event Management
- Reminders
- Multiple Views
- Conflict Prevention
- Sharing
- Synchronization
- Import/Export
- Ical format

## Non-Functional Requirements:

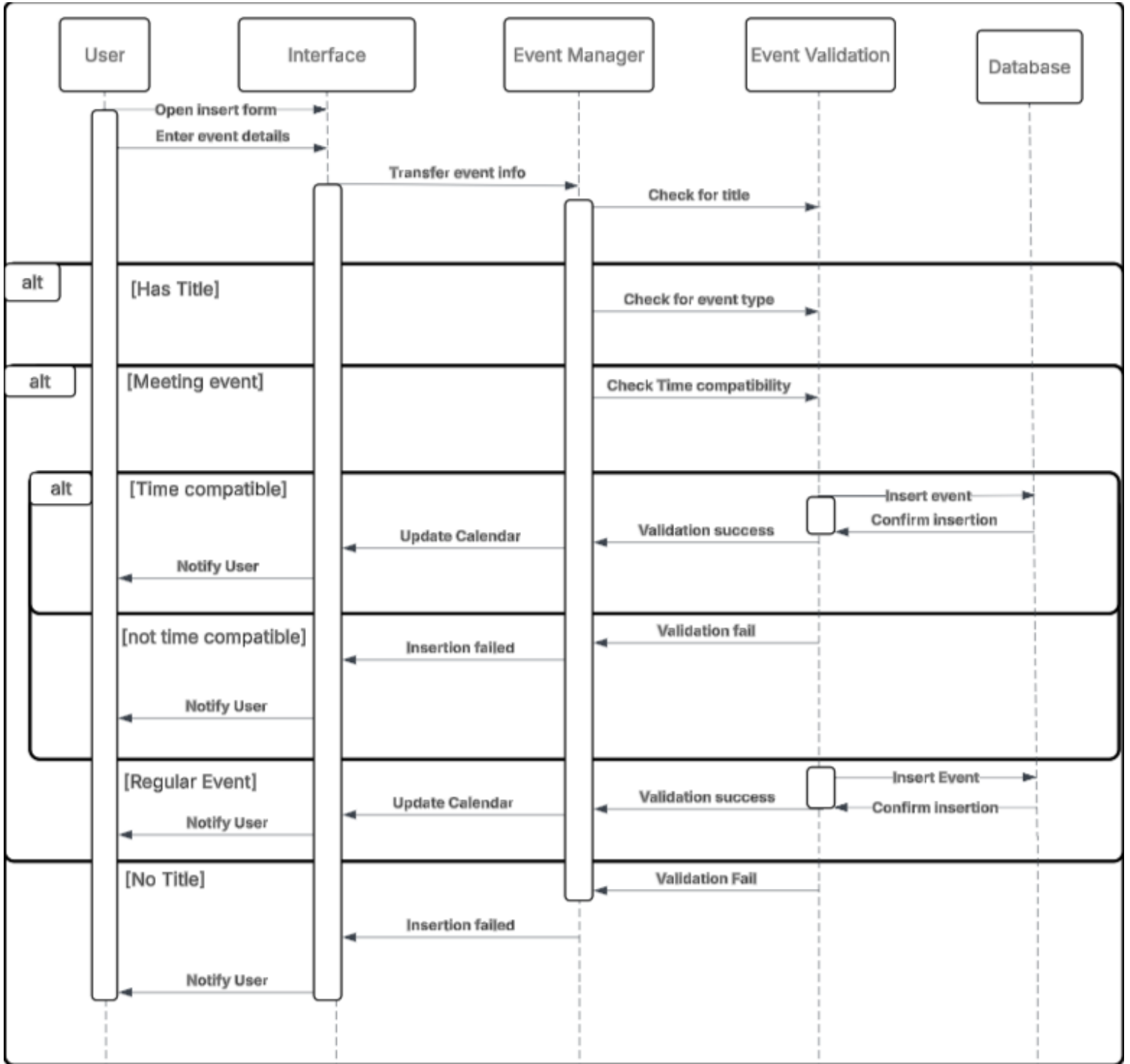
- Product Requirements – Performance, Reliability, Usability, Efficiency
- Organizational Requirements – Security, Maintainability, Portability
- External Requirements – Legal, Ethical

# Use case diagram

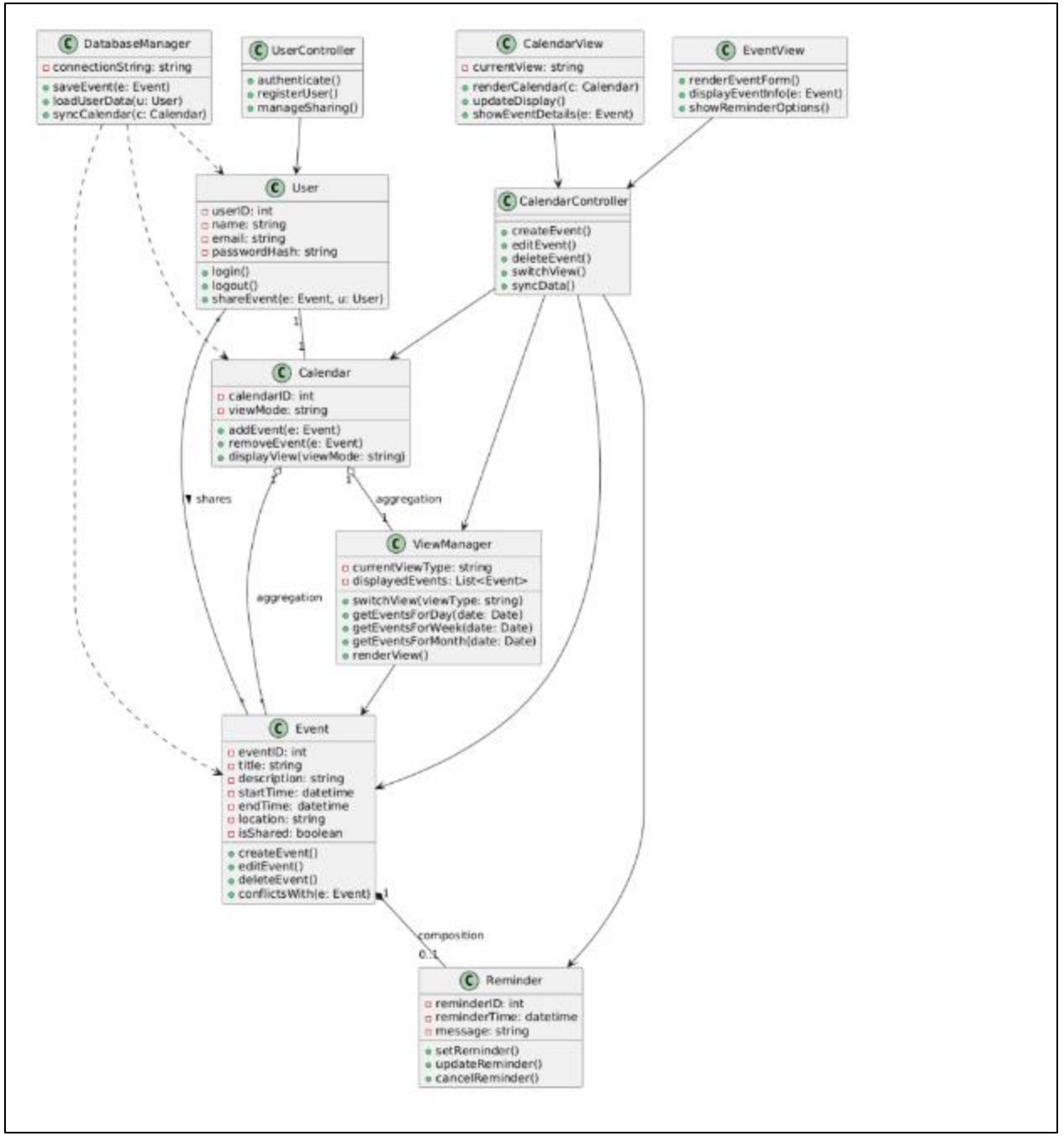


# Sequence Diagram

## Event Insertion



# Class Diagram



# Architectural Design: MVC Pattern