Internet Based Localized Calendar System

Project Proposal

Version 1.0

The Department Calendar Group

~

Duffey, Luke

Joos, Dylan

Rocco, Samuel

~

September 29, 2022

**Table of Contents**

[**Introduction**](#_heading=h.1fkgmnqfp7h1) **2**

[Purpose & Scope](#_heading=h.i8filvfas2qm) 2

[1.1 Purpose](#_heading=h.1cfsi9lybv4k) 2

[1.2 Scope](#_heading=h.o0unqndy76zs) 2

[Rationale & Constraints](#_heading=h.2q1f15nllksb) 3

[1.3 Rationale](#_heading=h.db7cy3tpuzhg) 3

[1.4 Major Constraints](#_heading=h.oj6bma12uzkx) 3

[Definitions & References](#_heading=h.qr6e8i81c3y3) 4

[1.5 Definitions, Acronyms, And Abbreviations](#_heading=h.rtjxqrmrnxwd) 4

[1.6 References](#_heading=h.q1b93hlq8ew8) 4

[**Overview**](#_heading=h.t5rqacgfwa5g) **5**

[Context Model](#_heading=h.16e1uatynfbj) 5

[2.1 Context Model](#_heading=h.idztpl1ug5zh) 5

[Use Case Model](#_heading=h.mfnyunj9ydwk) 6

[2.2 Use Case Diagram](#_heading=h.tl27uhirpcjk) 6

[2.2.1 Use Case Narrative](#_heading=h.jahi34xzu90h) 7

[2.3 Structural Model](#_heading=h.yolbwznr32fu) 8

[Constraints & Risks](#_heading=h.2eiuehxz59qc) 9

[2.4 Constraints](#_heading=h.hunrlhpoa7ed) 9

[2.5 Risks](#_heading=h.k3kd69i1w0wo) 9

[Assumptions & Dependencies](#_heading=h.ki1abyoazo8z) 10

[2.6 Assumptions And Dependencies](#_heading=h.m8xk90o5jer4) 10

[**Requirements**](#_heading=h.i92gsjg0bapc) **11**

[Functional Requirements & Traceability Matrix](#_heading=h.9fwl0rz1px4d) 11

[3.1 Functional Requirements](#_heading=h.yt65eervslw) 11

[3.2 Requirements Traceability Matrix](#_heading=h.4hcvc1yhwmkk) 11

[Nonfunctional Requirements](#_heading=h.h1evevhl6vk) 12

[3.3 Nonfunctional Requirements](#_heading=h.y17dbxjmbzh) 12

# **Introduction**

## Purpose & Scope

### **1.1 PURPOSE**

Our goal is to make a web-based scheduling system designed for a company or other organization, such as a college campus. Other goals we plan to achieve include having times that can be scheduled and locked in place for others in a group to see. A user will have the ability to make groups for certain users that can be notified about scheduling times. A system designed for users to have administrative properties on groups. The ability for administrators to assign users different modes of access, such as editor mode or viewing mode. Finally, the ability to link calendars from separate organizations to connect all calendars to one singular application.

### **1.2 SCOPE**

The system is designed for keeping and regulating schedules for a group or organization. Three major inputs will be accounted for. An input to create, add, or import a new schedule will be utilized by the user to start the calendar. Along with an input to share calendars or specific schedules to other users with the ability to assign an administrator. And an input to choose and view your calendar. Time management algorithms will be used to manage multiple schedules for the user to have a clear understanding of scheduled times with different calendar formats to best suit the needs of the user. Formats will include a monthly, a weekly, and a daily calendar depending on the specific need of the user. The application ability to link third-party calendar applications for a unified time managing experience, where users can keep up with a generalized overview of everything happening at once, as the application will understand the formatting and add each scheduled time from a third-party source to a single-glance display.

## Rationale & Constraints

### **1.3 RATIONALE**

This program is designed for marketing to large organizations. This type of software would be extremely useful for schedule organization of events for both employees and student-based organizations. Our calendar program is focused on group aspects while many other calendar programs focus more on the individual’s calendar. One of these systems is Microsoft Outlook, which is a very well implemented system for linking calendars and email. Outlook focuses more on linking email and calendars instead of scheduling itself. Another example is the Google workplace suite, these programs together allow calendar and scheduling functionality, but work more by appointment scheduling than by checking availability for group meetings and events.

### **1.4 MAJOR CONSTRAINTS**

Some of the biggest constraints are keeping response down to 1 seconds, and ensuring it can be used across all Android, IOS devices, and PCs. The System must be able to handle around 100,000 or more users at a time and must add events to others' calendars in less than 3 seconds as events get added so that users can then use the calendar and not double schedule themselves. The system must not have more than 5 seconds of down time in a week.

## Definitions & References

### **1.5 DEFINITIONS, ACRONYMS, AND ABBREVIATIONS**

Calendar - A way to track how time passes

Groups - A collection of users with one or more users able to schedule events

Administrators (Admin) - people in charge of a group

Events - An object to store information about real world events and activities

Editing mode - allows non-Admin users to make edits to the calendar for their group

Viewing mode - allows anyone to look at the calendar’s events

Local Time - The time as defined by the local governing authority.

### **1.6 REFERENCES**

The link below is for the Microsoft Outlook overview page. We used this overview to help us brainstorm ideas and as a market comparison.

[https://support.microsoft.com/en-us/office/introduction-to-the-outlook-calendar-d94c5203-77c7-48ec-90a5-2e2bc10bd6f8\](https://support.microsoft.com/en-us/office/introduction-to-the-outlook-calendar-d94c5203-77c7-48ec-90a5-2e2bc10bd6f8%5C)

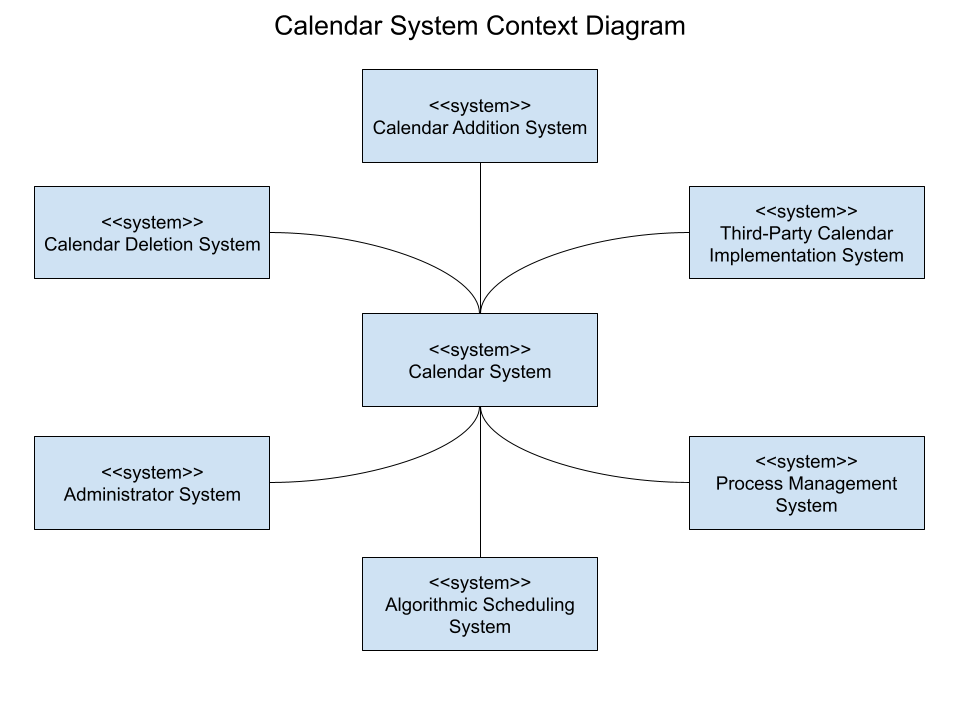
The link below is the introduction and overview page for Google’s workspace. This suite was also used as inspiration as well as a market comparison.

<https://workspace.google.com/individual/?utm_source=google&utm_medium=cpc&utm_campaign=na-US-all-en-dr-bkws-all-all-trial-e-dr-1011401&utm_content=text-ad-none-any-DEV_c-CRE_598599076035-ADGP_Hybrid%20%7C%20BKWS%20-%20EXA%20%7C%20Txt%20~%20Business-KWID_43700071094077977-kwd-298638842561&utm_term=KW_google%20appointment%20scheduling-ST_google%20appointment%20scheduling&gclid=CjwKCAjw1ICZBhAzEiwAFfvFhOE51LXsE7RVxRBRompwkLoiARXg7UFwiz5pCDlmQIsrwDWSmqPtXxoC1jwQAvD_BwE&gclsrc=aw.ds>

# **Overview**

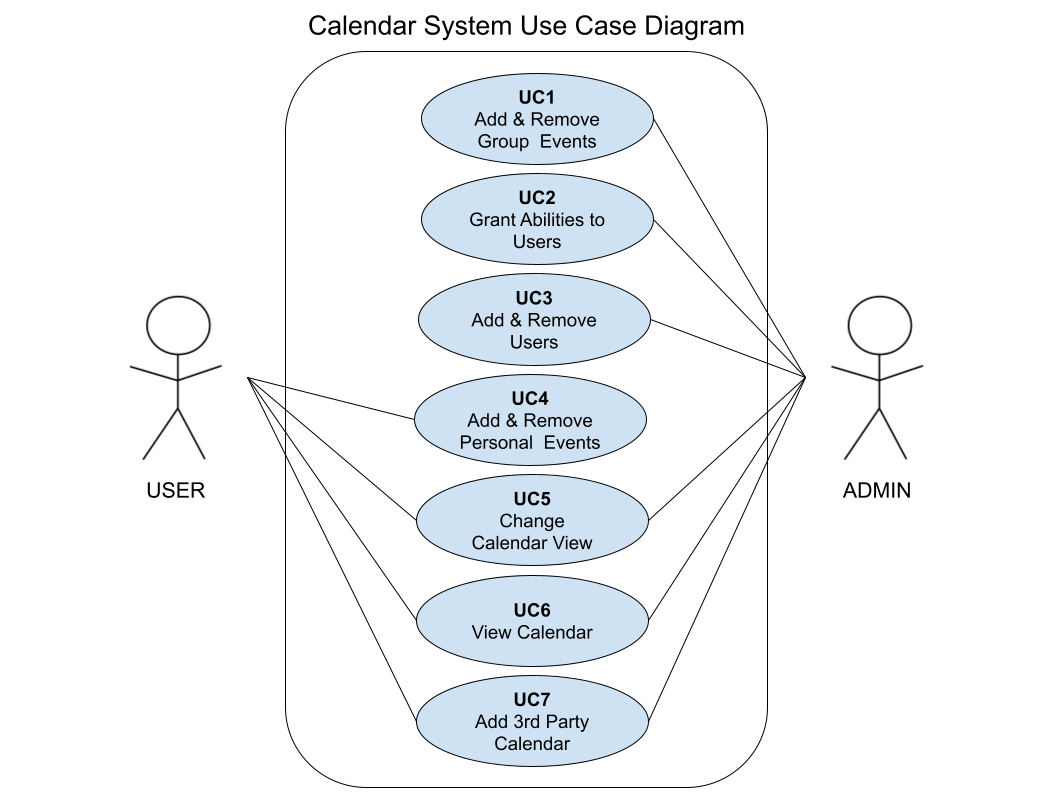
## Context model

### **2.1 CONTEXT MODEL**



## Use Case Model

### **2.2 USE CASE DIAGRAM**



#### **2.2.1 USE CASE NARRATIVE**

Use cases:

UC1 - add and remove group events:

An Administrator needs to schedule a new meeting for a club that they are a part of. The Administrator can then use the calendar program to check the availability of other users in their group and add a meeting to the group calendar. Similarly if a meeting needs to be canceled they may use the program to remove that event from that group’s calendar.

UC2 - grant abilities to users:

An Administrator feels the need to have additional administrators to manage meetings in a workplace. She may add new Administrators from the existing users in the group. Giving them all the privileges an Administrator may have.

UC3 - add and remove users:

An Administrator may need to add new people to the group’s calendar. This Administrator can then add as many people as is necessary to the group calendar.

UC4 - add and remove personal events:

A user may have a party coming up. They want to inform others in their group that they will be unavailable at that time. So they add their event to the group calendar so they may not be scheduled for one on one meetings at those times. Similarly if that party were to be canceled, then they can remove it from their calendar to notify the group that they are available again

UC5 - change calendar view:

A user may have multiple groups that they are a part of and feel that the main calendar is too full. They can then switch between group calendars to only show events related to a specific group.

UC6 - view calendar:

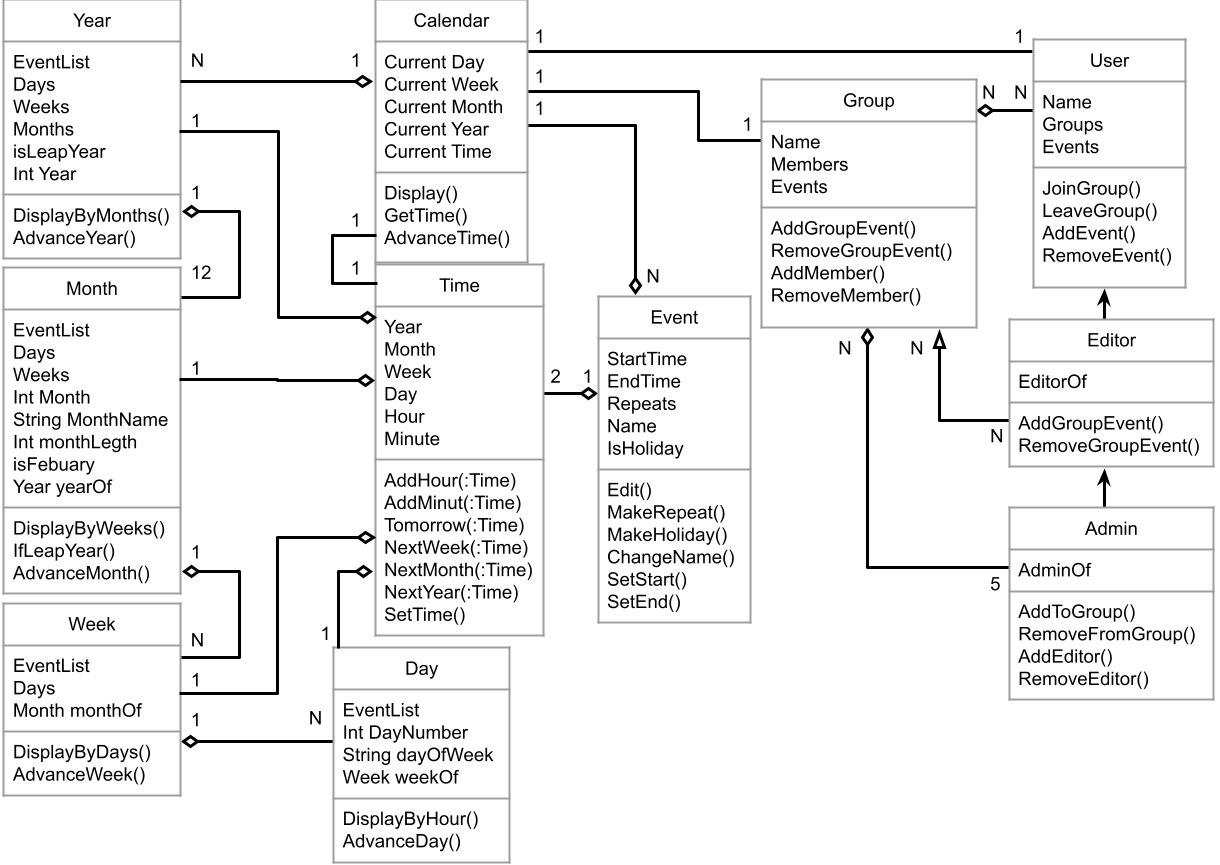
A user wants to check what meetings they have that day. They can open and view their main calendar to show all events for that day.

UC7 - add 3rd party calendar:

A user may have a calendar using another software. To ease transition to new software they may add all of the events on a 3rd party software. They will be imported and assigned to the same dates and times as the previous program.

Structural Model

### **2.3 STRUCTURAL MODEL**

****

## Constraints & Risks

### **2.4 CONSTRAINTS**

Time Constraints give our team one year to complete the product on time.

The client has given no specified financial value for the product.

The product is constrained to hardware that has to be connected to the internet.

Third-Party Constraints from outside companies view on their products use within our product.

Language Constraints when porting the product to different countries with differing dialects.

Log-in Security Constraints, focussing on two-factor authentification when logging in and signing up.

### **2.5 RISKS**

There are multiple large risks associated with developing this program. The first is the market for this particular type of program. Many office environments already pay for the microsoft suite and train on the microsoft suite of programs, including outlook. Those that do not pay for these programs may use the google suite of programs, which many of the programs are free. This market is going to be difficult to advertise to in the short term, possibly even long term. This can be mitigated by creating a rigorous marketing campaign, possibly even considering a free and a paid model of the product.

The second large risk is usage across time zones. Time zones are managed locally, and are not the same size across the globe. For instance China has one time zone, while Australia has many 30 min time zone distinctions. A possible way to mitigate this would be to read from the clock on the hardware.

The final risk associated with this project is accommodating different date display styles throughout the world, like mm/dd/yyyy, and dd/mm/yyyy. Allowing the user to select the date display type may reduce confusion from users across the globe.

## Assumptions & Dependencies

### **2.6 ASSUMPTIONS AND DEPENDENCIES**

There are several Assumptions upon which the System is based. Chief among these is that all users will be on ISO, Android, or have access to the Internet through a browser. Another assumption is that the need for an old calendar disappears after 7 years, and as such no information will be kept for more than 7 years. Furthermore no months, days, or years will be visible 7 years after (ie. if it is 2010, 2003 would be the last visible year).

The system will be dependent on UTC for the purpose of keeping accurate track of time. The System will also be dependent on local laws regarding time and as such will need some way or someone to keep track of changes in local laws. The system is also dependent on the internet to be accessible.

# 

# **Requirements**

## Functional Requirements & Traceability Matrix

### **3.1 FUNCTIONAL REQUIREMENTS**

FR1 The Administrator shall add and delete group calendars, schedules, and specific times to the overall calendar and database.

FR2 The Administrator shall grant the abilities that a user is granted when said user is titled administrator or co-administrator with other co-administrators.

FR3 The Administrator shall add and remove users from group calendars.

FR4 The Administrator shall view and change the view of the calendars.

FR5 The Administrator shall add third-party calendars.

FR6 The User shall add and delete personal calendars, schedules, and specific times to the overall calendar and database.

FR7 The User shall view and change the view of personal calendars.

FR8 The User shall add personal third-party calendars.

### **3.2 REQUIREMENTS TRACEABILITY MATRIX**

|  | FR1 | FR2 | FR3 | FR4 | FR5 | FR6 | FR7 | FR8 |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| UC1 | X |  |  |  |  |  |  |  |
| UC2 |  | X |  |  |  |  |  |  |
| UC3 |  |  | X |  |  |  |  |  |
| UC4 |  |  |  |  |  | X |  |  |
| UC5 |  |  |  | X |  |  | X |  |
| UC6 |  |  |  | X |  |  | X |  |
| UC7 |  |  |  |  | X |  |  | X |

## Nonfunctional Requirements

### **3.3 NONFUNCTIONAL REQUIREMENTS**

NR1 The System shall run on Android, IOS, and internet Browsers

NR2 The System shall handle 100,000 active users

NR3 The System shall respond in 3 seconds

NR4 The System shall not have more than 5 seconds of down time in a week