Calendar+

Members

Chad Kinnard, Bo Yang Li, Tyler Marrapese, Warren Overstreet

Abstract

Calendar+ is a calendar run as a dynamic web application that will help users personalize their schedule as they see fit. Through a simple account creation, a user's calendar will be personalized to them. Accessed through the Google Chrome web browser, users can view their calendar from any Windows 10 device. Calendar+ will make it to where users can plan their days, weeks, months, and years according to their own needs. Through browser notifications, users will be reminded of important events that they have scheduled. Time management is an important part of day-to-day life, whether as a student, or a full-time employee, and Calendar+ will help users' productivity.

Description

Web Applications have grown exponentially in the digital world over the past couple of decades. From purely static web pages in the nineties, to the introduction of dynamic web pages through technologies such as Macromedia's Flash of the early 2000s, to the current state of technologies such as JavaScript, HTML5, and various cloud storage options, the web is becoming a great place to run applications. As the web and technology have become a central part of our lifestyles, user accessibility has become more and more important. This is what we aim to implement in this project.

We shall be creating an Online Planner/Calendar, Calendar+, that will allow users to add important events and personalize their schedule as they need. The calendar will allow users to add events such as meetings, birthdays, and others. We will allow all of our users to create unique calendars by having them each create an account with a password and username for them to access, granting them their calendar to create events whilst also keeping their privacy about said events. The calendar will also alert users at a predetermined time interval about upcoming events, reminding them of the event in case they had forgotten or just would like the

extra forewarning. The calendar will have different views to give the user greater flexibility and foresight at upcoming events, this will include a daily view to see events throughout the current day, and a monthly view to see what days of the month contain events. We will also allow users to delete events should they find that they have no need for them anymore or else should the event have been canceled or rescheduled. All user accounts will default to a standard calendar of United States holidays, but we will allow our users to delete any holidays they wish to not appear on their calendar after the calendar account has been created. We shall allow our users to personalize events by allowing users to color-code each event helping readability and allowing users to at a glance to know if a day contains multiple events. The calendar application shall allow users to create events up to ten years in advance, granting our users plenty of time to plan events for the future. The features mentioned so far are the basic features we shall implement over the next few months.

While the features mentioned so far will take a great deal of time to implement, we have several more features we would like to implement if we have the opportunity. We would like to grant our users the ability to select holiday preferences when they create their accounts and personal calendars, such as preset holiday settings based on country of origin or even particular religious holidays. We would also like to grant our users the ability to invite other users of our calendar service to a particular event, if they accept they would instantly have this event added to their calendar allowing many people to have an event but only requiring one person to set it up, reducing the chance of incorrect details about the given event. We would also like to expand upon the event invitation/sharing system to create a form of RSVP Tracking, this would allow our users to know who and how many people were attending a given event should they wish, this would be of great importance for events such as weddings. We would also like to create a form of the to-do list for a given event should preparations need to be in order such as shopping, cleaning, or even a reservation. Given enough time we would like to grant our users the ability to have multiple calendars per account should they, for example, wish to keep their work and personal life calendar events separate, no matter for what reason our users may have, we would like to grant our project this level of flexibility. We would also like to grant our calendar support for different time zones as this could confuse our users in different areas of the world. We would like to further grant our users the ability to personalize their experience and grant our website a Light-mode and a Dark-mode setting to make our website easier on the eyes should our users desire. We would like to extend our website’s compatibility to other browsers to

expand upon who can use our website. Finally, we would like to allow our calendar to support and perform well on mobile devices, further expanding our user support.

Our website will be designed with the Windows 10 operating system at base supported as that is what we, the creators, use. The calendar will be an online service that will allow users to access it from the Chrome web browser, given that it is the leading internet browser worldwide. We shall be implementing our calendar through the use of HTML, CSS, and Javascript to handle the web-based portion of the project and we shall most likely use SQL to handle server-side database management but that portion of the project is undecided at current. Our team will utilize the service Discord to communicate throughout the design and implementation process of the project, given that we all have prior experience. Finally, we plan on using GitHub to use as a source code management system, allowing all team members easy access to the project's up-to-date source code.

Feature List

Required:

- Accounts & Login
- Event Creation (holiday, birthday, meeting, etc…)
- Different views (monthly view, daily view, etc…)
- Event Deletion
- Defaults to U.S. Holidays
- Color coding events
- Reminder System(alerts)
- 10 year scope

If we have time:

- Holiday preferences
- Ability to reschedule events
- Event Invites
- RSVP Tracking & Forwarding
- To-do list
- Multiple calendars

- Multiple time zone support
- Personalization Settings
- Light-mode / Dark-mode options
- Compatibility with other browsers
- Mobile/Desktop port

Will not be implemented:

- Edit event times by resizing event boxes or by dragging and dropping them on the calendar
- Weather Forecast in event descriptions
- Integration with video chat services
- Enterprise Resource Planning software integration
- Automated event creation from emails
- Two Factor Authentication
- Calendar Sharing
- Event Creation Autofill

Technology

Since Windows 10 is the most common desktop operating system, we will be both using and officially supporting it. The same can be said for Google Chrome, which is the most common browser today, with over 60% market share. Since we are creating a web application, we will be using HTML, CSS and Javascript, which are the building blocks of the web, and since the entire team has some experience with them. To allow us to work on the same code together, we will be using GitHub, since it is a version control system coupled with a code repository that allows the team to keep track of changes over time. We plan on setting up a server through Linode using Ubuntu. Since we already have experience using SQL, that will be used for our database which we will need for the calendar. We will be using Visual Studio Code, as it has easy GitHub integration, alongside its own features to allow teams to collaborate on the same project at the same time. Having already used Discord prior to the project starting, we will use that for the majority of our communication outside of class.

Server Information

We will create a Linode account as we will need a server to host our calendar website. We have chosen Ubuntu as our Linux Virtual Machine (VM) distro. Out of the regions given, Atlanta, Georgia seems to be the best location to host the server. We decided to go with the five dollar monthly Nanode 1 GB plan, as it should suffice our workload.

Data Sources

- In the event that we decide to use a data set to populate the holidays rather than filling them out manually, we will need to use something like https://github.com/commenthol/date-holidays – however, it is still undecided if this is the specific one that we would end up using.

Backgrounds

Warren Overstreet

- 4 years of C++
- 1 year of HTML, CSS, and Javascript experience
- 1 semester of SQL and Database experience

Tyler Marrapese

- 4+ years HTML, CSS, JavaScript
- 3+ years C / C++
- 1 year C#

Bo Yang Li

- 1.5 years HTML/CSS/JavaScript
- 1 semester of TypeScript experience
- 3+ years Java
- 2 years Python
- 1 year C/C++
- 2 years git

Chad Kinnard

- 1 year HTML, CSS, and JavaScript
- 3+ years Java
- 1 year Python

Dependencies, Limitations, and Risks

Server implementation could prove to be a risk, but there are many online resources that should prove to be helpful should any difficulties appear. The Coronavirus may limit our ability to meet in person as a group should any of us become infected or the campus close due to rising infections. In the event we are unable to meet in person, we have set up a Discord server so that we can communicate in such circumstances. Inclement weather could prove to be a risk should any classes be canceled such as they were on January 20th, Though we can always communicate through email or the Discord server should we be unable to meet in person as stated before.

Timeline

Week 1: January 18 - January 21

This was the first week of school. We mainly went over the syllabus and met up with our group members. We also brainstormed project ideas.

Week 2: January 24 - January 28

We decided on creating a calendar web application. We spent this whole week working on the project proposal

Week 3: January 31 - February 4

During the first week working on the project, we will begin working on a visual design for Calendar+.

Week 4: February 7 - February 11

During the second week, we will officially begin implementing a basic account & login system for users to have separate data for their calendars.

Week 5: February 14 - February 18

During this time we will be working on implementing multiple different calendar views so that users can view their calendars day by day, week by week, etc.

Week 6: February 21 - February 25

During the 6th week of this course, we will implement event creation and event deletion.

Week 7: February 28 - March 4

Week 7, we will implement color coding in events.

Week 8: March 7 - March 11

This is Spring Break week. Depending on availability of group members during this time, we will make plans accordingly.

Week 9: March 14 - March 18

Week 9, we will begin to implement holidays – we are unsure of how big of an undertaking this will be, so we are allowing two weeks to accomplish this task.

Week 10: March 21 - March 25

Week 10, we will continue implementing holidays.

Week 11: March 28 - April 1

Week 11 we will implement a reminder system (10 minutes before, 20 minutes before, 1 day before, etc).

Week 12: April 4 - April 8

Should there be enough time, we will ensure the compatibility of Calendar+ with other browsers.

Week 13: April 11 - April 15

Should there be enough time, we will deliver a Calendar+ implementation for mobile/other desktop operating systems.

Week 14: April 18 - April 22

Should there be enough time, we will deliver reschedulable events.

Week 15: April 25 - April 29

Should there be enough time, we will integrate a to-do list into the Calendar+ app.

Week 16: May 2 - May 6

Should there be enough time, we will deliver personalization settings and light/dark mode options.