

# Blueprint: A Framework for Collaborative Success

Requirements Specification & Technical Roadmap

Isaac Trejo Mendoza, Jacob McPherson, Jeongha Park, and Obadiah Sieg

# Introduction



- Problem: group projects are typically hard to manage and the tools out there are far too complicated to warrant using them in the first place.
- Solution: a simple project management web application that is free to use and compares everyone's schedules to find the best time for them to meet and collaborate on group projects.
- Core philosophy: minimalist and student focused.



Home  
Notifications  
Goals

Spaces 

Everything  
**D Design**  
**E Engineering**  
**P Product**

Dashboards 

Docs 



Project



Box



Board

Timeline

Doc

Whiteboard

## Workload



Brendan

23

Not done

6

Done

### TIME ESTIMATE

5d 15h

Not done

6h

Done

⚠ 19 tasks without estimate

▼ READY (8)

▼ IN PROGRESS (14)

▼ REVIEW (14)



12

Not done

1

Done

▼ READY (1)

▼ IN PROGRESS (17)

▼ REVIEW (8)



Amy

12

Not done

3

Done

### TIME ESTIMATE

1d 15h

Not done

3h

Done

⚠ 17 tasks without estimate

▲ READY (8)

TIME EST.

User research - P... 2h

Wireframe new... 0.5h

Review & Appro... 2h

▼ IN PROGRESS (2)

▼ REVIEW (3)

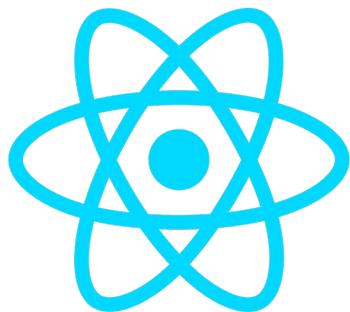
# Overview

- Tools & Technologies
- Project Requirements
- Tentative Schedule

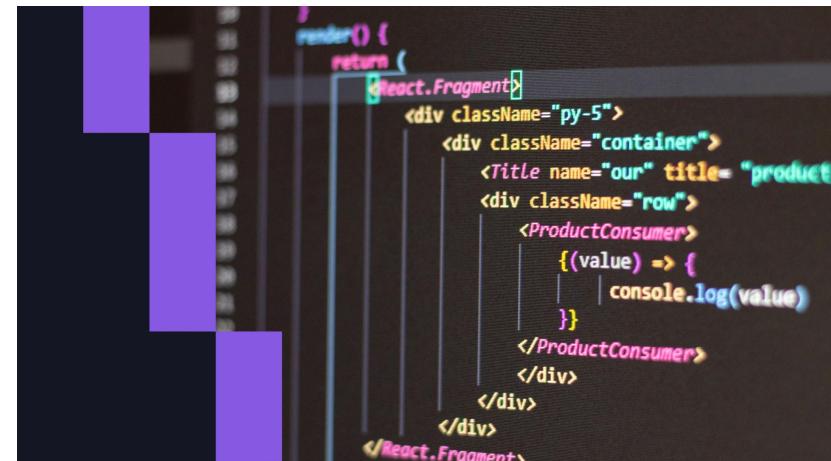
# Tools & Technologies

React	Front end with a component-based, reactive U.I.
Supabase	Back end for a real time database and Google OAuth 2.0.
Vercel	Hosting and deployment of the website.
Figma	Design and rapid inspiration of the G.U.I.

# Front End



# React



A screenshot of a code editor showing a React component. The component is named 'ProductConsumer' and contains a title and a row of products. The code uses React Fragments and functional components.

```
render() {
  return (
    <React.Fragment>
      <div className="py-5">
        <div className="container">
          <Title name="our" title="product" />
          <div className="row">
            <ProductConsumer>
              {(value) => {
                |   |   console.log(value)
              }}
            </ProductConsumer>
          </div>
        </div>
      </React.Fragment>
```

# Back End



The screenshot shows the Supabase Backend interface. At the top, there's a navigation bar with Home, + Add, Library, Plugins, Auth, Backend, and More. The main area has tabs for Layout, Data, Actions, and a search bar. On the left, there's a sidebar with Collections (properties selected), a New button, and a Variables section with a Create global variables button. The main workspace is divided into sections: properties (Select a source: Supabase, Dynamic), Configuration (Table properties), Query configuration (Pagination: Off, Filter, Sort), and Fetch data (This collection is fetched automatically). To the right, a Data panel shows a preview of the data with a Success message. The data is listed in two sections: 0 and 1, each with an object containing fields like id, title, description, price, bedrooms, bathrooms, address, and square\_feet.

```
Success
0
id: "e8c027c8-73fa-40ac-b380-f11a3676d42"
title: "Modern Downtown Loft"
description: "Stunning modern loft with
price: 450000
bedrooms: 2
bathrooms: 2
address: "123 Downtown Ave"
square_feet: 1200
1
id: "c324dc36-9f30-4c18-b700-1ab9bdc70b3"
title: "Suburban Family Home"
description: "Spacious family home with
price: 650000
bedrooms: 4
bathrooms: 3
address: "456 Maple Street"
square_feet: 2800
> 2 Object
> 3 Object
> 4 Object
> 5 Object
> 6 Object
> 7 Object
> 8 Object
> 9 Object
```

# Server

▲ Vercel

The screenshot shows the Vercel DevTools interface. On the left, a sidebar displays deployment logs for various projects, including "acme-store" and "acme-japan". The main area features a code editor with the "index.html" file open, showing basic HTML structure. To the right is a file tree showing the project structure with files like "index.html", "favicon.ico", and "robots.txt". At the bottom, a "CHECKS" section indicates a pending build. A tooltip for the "acme-store" deployment provides details about recent changes.

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8" />
    <meta name="viewport" content="width=device-width" />
    <link rel="icon" type="image/svg+xml" href="/favicon.ico" />
    <title>Acme Store</title>
    <meta name="description" content="Acme Store" />
  </head>
  <body>
    <div id="root"></div>
  </body>
</html>
```

VERCEL

DEPLOYMENTS

- acme-store 1m ago
- acme-japan-web 2m ago
- acme-store 5m ago
- acme-japan-web 13m ago
- acme-store 20m ago
- acme-store 45m ago
- acme-store 3h ago
- acme-store 6h ago
  - Visit store-df563c.vercel.app
  - fix: as an AI model I'm...
  - feature-a acme/store
  - Build Logs READY
- acme-japan-api 11d ago

PROJECTS (2)

- acme-store 1d ago from 5cd76e2 via GitHub
- acme-japan acme-japan - 28id

index.html - acme-store

FILES

- \_next
- api
- download-files
- send-email
- report-user
- 404
- 500
- favicon.ico
- globals.css
- index.html
- robots.txt

acme-store

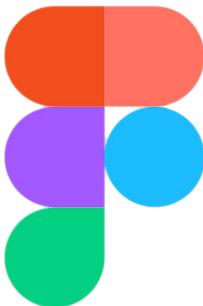
demo.vercel.store

fix: center buttons and rename variables

- 11 more tests are added
- change theme colors from purple to peppermint

Ln 13, Col 7 ⚒ HTML ▲ Building

# Additional Tools



# Figma

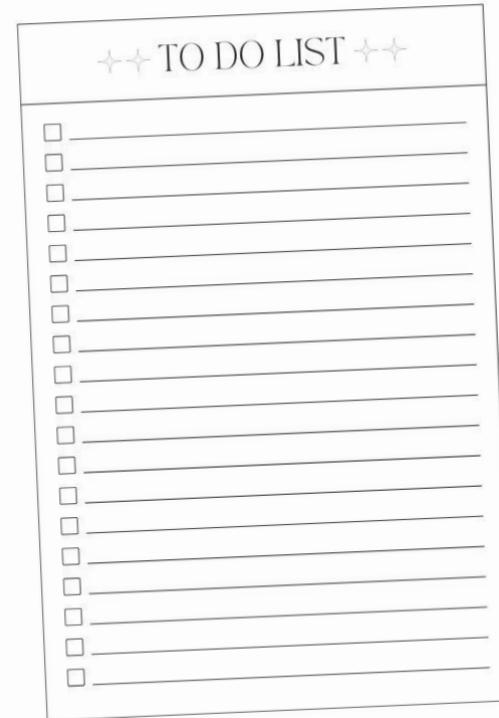
The screenshot displays the Figma application interface. On the left, the sidebar contains a navigation menu with sections like 'Pages', 'Components', 'Assets', and 'Code'. Under 'Pages', there are items such as 'Round 1 review 5/8', 'Player explorations', 'Sign-in screen', and 'Archive'. Under 'Components', there are categories like 'M / Tempo / Playlist Detail', 'M / Tempo / Filters', 'M / Tempo / Home', and 'Frame'. The main workspace shows a blurred image of a person looking at a city skyline at night. Overlaid on this are several UI elements: a pink button labeled 'Auto', a search bar with a magnifying glass icon and a '24' notification, a red button labeled 'Developer', and a green button labeled 'Designer'. Below the workspace, the text 'Palms Sylvan' is visible. The right side of the interface features the 'Inspect' panel, which shows details about selected components like 'M / Heading' and 'Dev resources'. It also includes a 'Component' section with a preview of the 'Discover' component and a 'Props' section with properties for 'Variant', 'State', and 'Destructive'. At the bottom, there's a 'Layout' section with a visual representation of a box with dimensions 344x100 and padding of 16px on all sides.

# Project Requirements

- Task Management Component
- Calendar Component
- Notification System
- Design: examples and non examples
- Dashboard

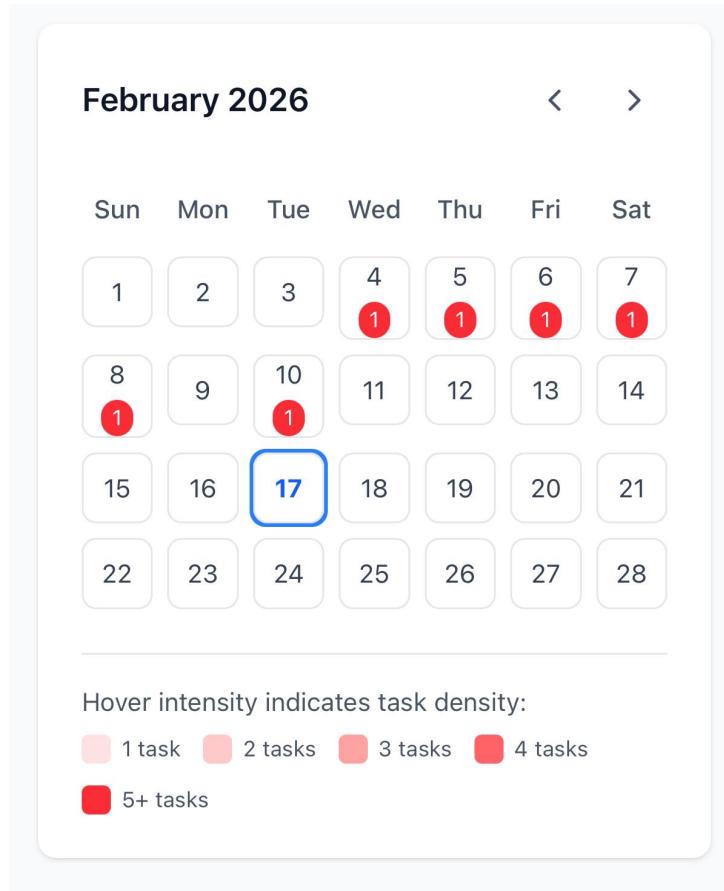
# Task Management Component

- A dynamic task list: gives the user the ability to view the tasks, the members assigned to them, and their priority at a glance.
- Color-coded priorities: red for high, orange for medium, white for low, and green for completed tasks.
- Completion animation for when a task is finished.
- A “Past Tasks” view



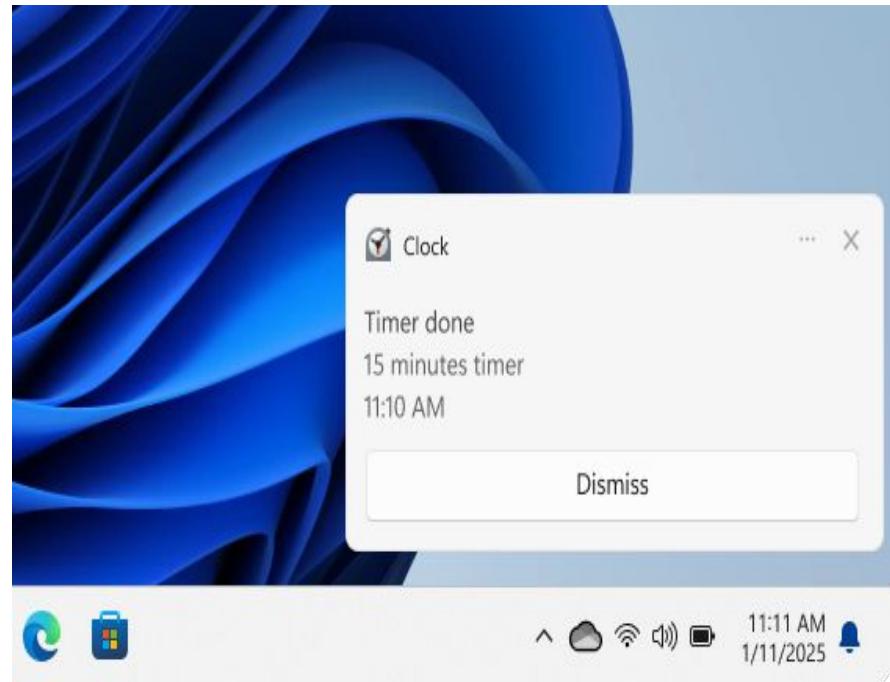
# Calendar Component

- Monthly grid view
- Selection for other months will be above
- The current month will be the default one shown
- Each day on the calendar will have a red indicator depending on if there are any tasks due that day. (The darker the indicator means more tasks due that day)
- Clicking any day on the calendar will show all the tasks assigned to that day
- Tasks will be automatically be “Unassigned” until the users assigns a due date for that specific task



# Notification System

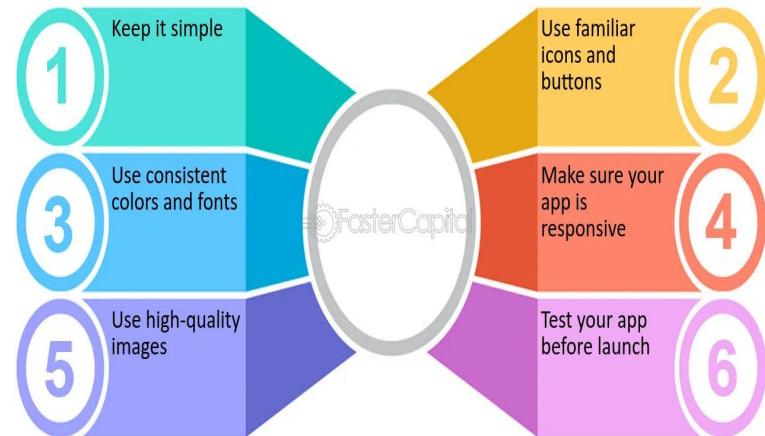
- 24 hours before due dates
- Audible alert
- Title and Task (Details)
- Daylight Saving Time  
Notifications Alert users  
when DST begins or ends



# Design

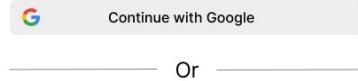
- Entertaining elements
- Replace the word task with quest
- Quest Rewards - 30 minute break, snack time, Play video games
- Encourage users to enter the work zone comfortably.

## Designing a user friendly interface



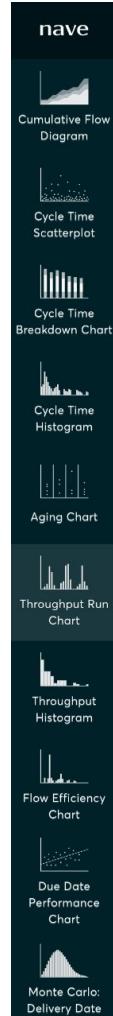
# Design Examples

- Minimalist
- Easy to digest.
- No information overload.



By clicking continue, you agree to our [Terms of Service](#) and [Privacy Policy](#)

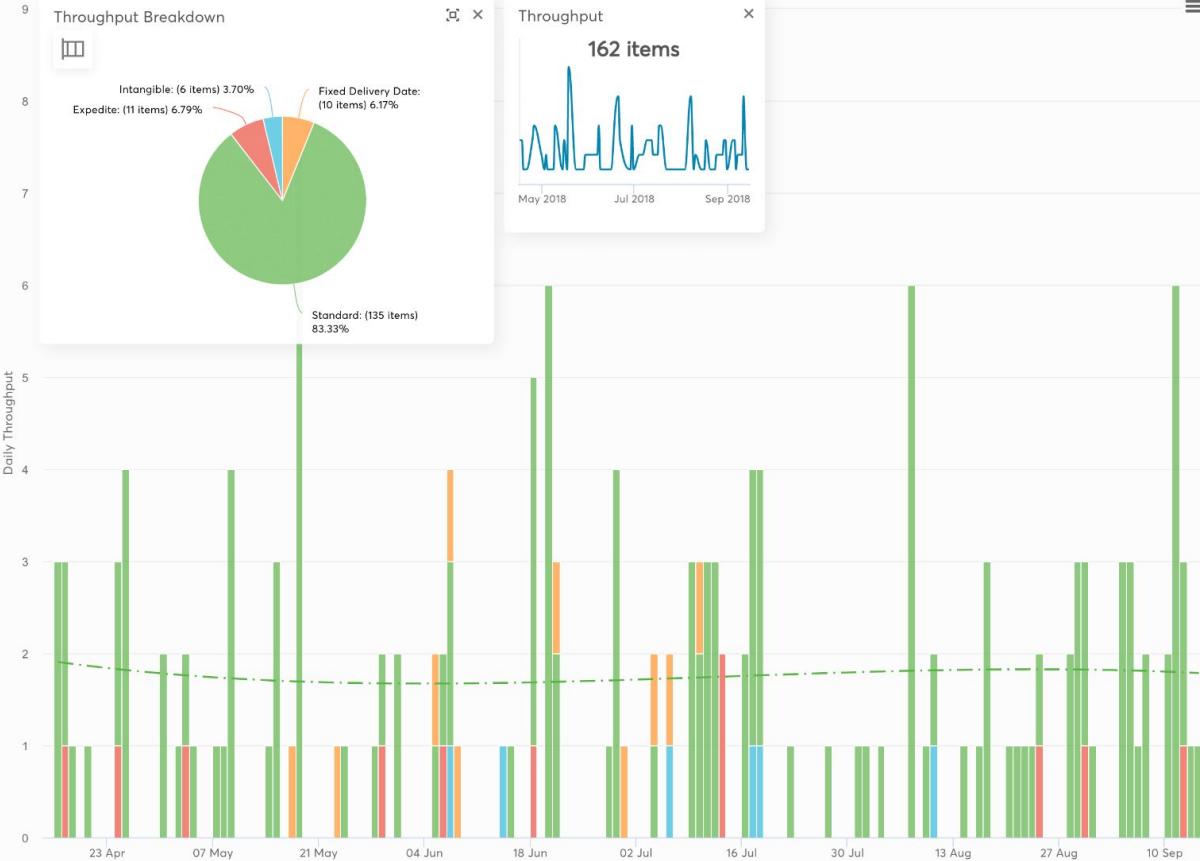
# Unacceptable Design Examples



Development



## Throughput Run Chart ⓘ



## Controls

## ▼ Trendline

- 
- Enabled

## ▼ Group by

- 
- Day
- 
- 
- Week
- 
- 
- Month

## ▼ Filters

## ▼ Lists

- 
- Select all
- 
- 
- To do
- 
- 
- Development
- 
- 
- Code review
- 
- 
- Code review (Done)
- 
- 
- Testing
- 
- 
- Testing (Done)
- 
- 
- Deployment
- 
- 
- Done

## ▼ Labels

- 
- Select all
- 
- 
- Expedite
- 
- 
- Fixed Delivery Date
- 
- 
- Intangible
- 
- 
- Standard

## &gt; Members



The screenshot shows the Asana interface for a workspace named "Lindy's Worksp...". The main view is titled "Book Deliveries" and displays a table of deliveries. A modal menu is open at the top left, triggered by a blue "+" button. The menu options include "New Item", "Search", "Person", "Filter", "Sort", "Hide", and three dots. Below this is a dropdown menu with options: "New Board", "New Doc", "New Dashboard", "Choose from templates", "Import data", "Apps", "New Form", and "New Folder". The "New Dashboard" option is highlighted with a blue box and has a blue arrow pointing to it from the left. The main table lists deliveries for "Koo's Nest" on various dates with different vehicles and drivers. At the bottom, there is a section titled "▼ Kids Fiction" with a single item listed.

Item	Driver	Vehicle	Date	Hours
		Van 1	Jul 9	1
		Truck	Jul 13	3
		Van 2	Jul 14	2
Koo's Nest		Bicycle	Jul 25	1
		Van 1	Jul 18	5
		Van 2	Jul 20	2

▼ Kids Fiction

Item	Driver	Vehicle	Date	Hours
James and the Giant Peach		Truck	Jul 13	1

 Mirha

**Project**

- Hope project 

 **Dashboard**

 **Tracking**

 **projects**

 **work History**

**Tools**

-  **Inbox**
-  **Setting**

 **ADD NEW TASK**

 **JOE MAX**  
TEAM LEADER

## Dashboard

13 March 2021

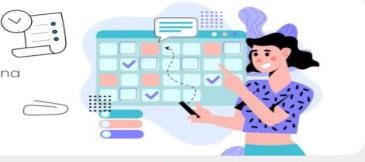


 13 March 2021

**March 2021**

27	28	1	2	3	4
5	6	7	8	9	10
11	12	13	14	15	16
17	18	19	20	21	22
23	24	25	26	27	28
29	30	31	1	2	3

**Welcome To Your Task Management Area**





**Total work**



**Task Percentage**



**Work Progress**

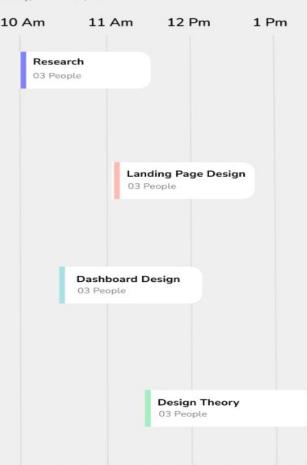
**Ongoing Project**  
**Dashboard Design**  
90 % Complete  
Start Date : 10 Jan End Date : 29 Jan

**Ongoing Project**  
**App UI UX Design**  
30 % Complete  
Start Date : 12 Jan End Date : 25 Jan

**Working Status**



**Upcoming**  
Wednesday, 13 March, 2021





# Dashboard

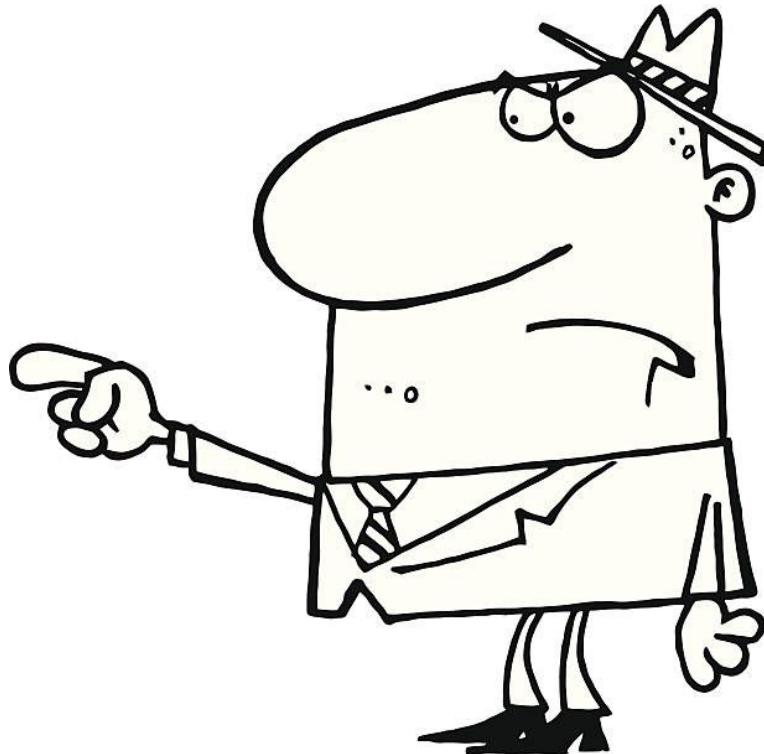
Google OAuth2



A completion bar to show the user's progress.



# Roles & Permissions



- Every user can be assigned a role from the creator (“Founder”) of that specific Blueprint
- These roles are Manager and User
- Managers will have full “Create, Read, Update, and Delete” privileges or (CRUD) for every tasks and members under the Blueprint
- Users can update their own assigned tasks and view the project’s overall status

# Tentative Schedule

Week 1: Feb 9-15	Finalize requirements document, initialize GitHub, and setup Supabase and Vercel
Week 2: Feb 16-22	Begin the front end with React and configure the database with Supabase
Week 3: Feb 23-Mar 1	Implement Google OAuth and begin laying out the dashboard
Week 4: Mar 2-8	Create the task list, calendar components, and hooks for the database
Week 5: Mar 9-15	Implement the task list and calendar and begin making the logic for the completion bar
Week 6: Mar 16-22	Create and implement the notification system, implement roles, and test for bugs
Week 7: Mar 23-29	Conduct an overall assessment of the website and fine tune the website with “Past Tasks”
Weeks 8-12: March-May	Create and implement gamification and ChatGPT API call, if time permits

# Conclusion



- Blueprint is the solution to group projects being hard to manage and there simply being too many project management tools out there that aren't simple enough for an academic environment
- Blueprint will be simple and easy-to-use, catering specifically to students
- We will be using React, Supabase, Vercel, and Figma to do this
- If time permits, we will implement gamification affects and an API call to Chat GPT to make suggestions for the user

# Questions?