

How well do developers track their daily progress?

Importance of having a journal

- **Enhanced Daily Organization:** Dedicated Journal helps developers organize tasks and notes, leading to an increased productivity.
- **Accurate Performance Tracking:** Crucial for performance reviews and career advancements
- **Knowledge Base:** Developers can refer back to their notes upon encountering similar issues

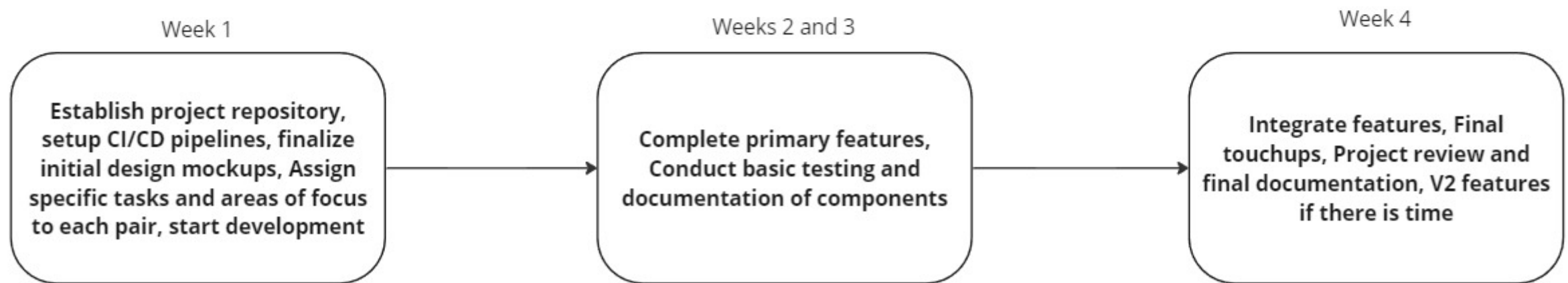
Problems we are aiming to solve

- **Difficulty in Tracking Progress:** Lack of suitable tools makes it challenging for developers to maintain an accurate daily log, affecting demonstration during reviews
- **Lack of Developer-Centric Features:** Absence of essential features like markdown and code snippet support limits the utility of current tools for developers



Project Scope and Commitment

- **Project Timeline:** Due to the nature of this class, we have a **4-week** Development cycle for this project
- **Time Constraint:** Due to this time constraint, we will focus on delivering core functionalities to maximize the MVP's (Minimum Viable Product) impact within the limited time.
- **Rough Project Roadmap:**



Personas/User Stories

Jenny

- Developer
- Likes to stay organized & neat
- Always swamped with several different projects
- Forgets to take breaks



- As a developer who works on many projects & likes to stay organized, I want to have a way to separate different journal entries & tasks for different projects I'm working on.
- As someone who forgets to take breaks, I would like a pomodoro timer to remind me.
- As a developer, I would like to be able to store my notes as Markdown.
- As a developer, I would like to have a function to export a PDF of an entry.
- As a developer, I know how invasive and predatory our data-driven market is, so I care about privacy and open-source software that stays offline to minimize data theft and mitigate my personal privacy breaches.

John

- CS major at university
- English is not his first language
- Procrastinator
- Works late & prefers working with dark mode



- As a procrastinator, I would like to have a deadline countdown timer to keep myself on task & see which tasks I should prioritize.
- As an international student who prefers reading in my native language, I would like a translate option.
- As someone who prefers dark mode, I would like an app that is open source so I can import different visual themes such as dark mode for when I am working late.
- As a CS student, I want to have a calendar view of all my notes to be able to tell which entry is for which day.
- As a procrastinator, I would like to have a feature that marks when I've started/finished a task so that I can improve on time management.

Feature List



Accessibility/Visual:

- Google Translate widget
- Help/Guide (tutorial)
- Scrollability
- Resizable for various screen sizes, allow real time resizing in browser
- Open source - allows users to import visual themes if they want to change app appearance
- Pomodoro timer

Entries:

- Tab for favorite entries
- Popup on screen when entry is clicked
- Separation for different projects (labels, color coded) and ability to toggle between projects
- Attach files
- Link to different entries

Task List:

- Add new task function
- Complete task button
- Edit task function
- Delete task function
- Timer - displays how long is left until due date
- Color code/label by project

Calendar:

- Toggle between different project views
- Dev specific features
- Each day contains notes, entries, and task list

Competitors



- **Day One:** Password protected, allows the option of keeping multiple journals, able to add images, tag feature, sends reminder to journal each day



- **Notion:** Collaboration, real time updates, rich functions, Markdown support



- **Obsidian:** Local Storage, Markdown support, offline access

Why Our App?

- **MarkDown Support**
- **Real Time updates & countdowns**
- **Allows Password**

Our Goals

Local First

- Developers can access their notes without having to access the network

Markdown Support

- Most developers are familiar with markdown syntax and it's versatile in terms of styling

Customization

- We want developers to be able to customize the app to their favorite color scheme through open-source CSS

Accessibility

- We want our app to be accessible on different screen sizes, devices, and to people with disabilities

Accessibility

What

- Translate Functions
- Scrollability
- Resizing function

Why/Benefits

- Allows users that prefer a different languages to translate to their preferred language
- The App won't have any limit on tasks, user can scroll up or down to a specific task
- Instead of having to zoom in & out when using a different screen size, it will resize as needed to their dimensions.

Potential Risks

- Users might accidentally translate to a language that they don't understand and may not know how to change it back.
- Developer risk - Increased complexity for developers

Markdown Support

What

- writing entries in markdown
- ability to include code snippets and images

Why/Benefits

- Commonly used by developers
- little concern of unfamiliarity
- intuitive
- Easy to import/export
- Adaptive to other writing elements
- Allows you to write using an easy-to-read, easy-to-write plain text format, then convert it to structurally valid XHTML (or HTML)

Potential Risks

- A user may not be familiar with markdown, and have to learn it to use our product
- Enabling HTML rendering from raw text input could allow for injection, which could be a security risk
- Markdown without extensions is very limited in what it can do, leading to users being restricted in the creativity as far as designing their dev journals (no footnotes, table of contents, charts, diagrams, etc.)

Open-source Themes

What

- Dark mode color theme (default)
- ability to import a color/style theme via CSS file

Why/Benefits

- Dark mode by default
 - generally preferred by developers
 - softer on the eyes
- Potential accessibility issue with only one color theme, so we provide an opportunity for each user to import their own
- Customizable to the individual
- Familiarity with CSS is assumed, as the users are developers

Potential Risks

- Providing a feature with a wide range of customization could open up issues if the user does something wrong; e.g. writes their file incorrectly and can't change it back?
 - possible solution: providing a template for the user to edit to their liking
- Open customization could also pose time risk for us as developers

Electron

What

- Electron is essentially a "wrapper" for web applications that enables them to run as native applications
- It will allow us to utilize a web development workflow with all the benefits it enables to create desktop applications

Why/Benefits

- This enables us to focus on a local first approach, ensuring no delays due to server latency
- It also removes the need for accounts or other server-reliant features since all data is stored locally

Potential Risks

- Electron is very resource-intensive as it runs an entire instance of Chromium as its wrapper for the application
- We need to write efficient code to ensure runtime and resource usage are within acceptable levels



Security

What

- Developers are often concerned about data security
- We want to protect user data and ensure that potentially sensitive information can't be stolen


How/Benefits

- Utilizing an offline application that is entirely reliant on local storage mitigates concerns of our company accessing user data
- Using encryption methods from Node.js prevents other applications or users from accessing files

Potential Risks

- Data recovery could be made difficult due to encryption
- Local storage enables the user to manually delete user data with no way to retrieve it



Enter Password 

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

Clicking on + (add entry)

		1 <input type="text" value="Entry"/>	2 <input type="text" value="Entry"/>	3 <input type="text" value="Entry"/>	4 <input type="text" value="Entry"/>	5 <input type="text" value="Entry"/>
6 <input type="text" value="Entry"/>	7 <input type="text" value="Entry"/> <input type="text" value="Entry"/>	8 + <input type="text" value="Entry"/>	9	10	11	12
13	14	15	16	17	18	19
20	21	22	23	24	25	26
27	28	29	30	31		

		1 <div>Entry</div>	2 <div>Entry</div>	3 <div>Entry</div>	4 <div>Entry</div>	5 <div>Entry</div>	
6 <div>Entry</div>	7 <div>Entry</div> <div>Entry</div>	8 + 9	10	11	12		
13	14	15	<div>Notes for May 8th 2024</div> <div><div></div><div></div><div></div><div></div><div></div><div></div><div></div></div> <div>Save</div>			18	19
20	21	22				25	26
27	28	29					

Rabbit Holes

Out of bounds (things we won't even consider doing due to the time constraint)

- Real-time edit by multiple users (collaboration)
- Storing notes on the cloud

Possible Rabbit Holes in our Current Features

- Learning how to use the new libraries could be a longer process than we anticipate
- The resizing function may propose difficulty with the calendar element on different screen sizes

Tools and Tech Stack

- **Electron:** The main wrapper around our web application
- **Markdown Rendering Library:** A library that parses markdown and renders it as html tags
- **Exporting html as pdf Library:** A library that does this to allow developers to export their notes