



CodeLog

27 Wayfarers



Problem Statement

Broad Problem: In today's increasingly collaborative landscape of software development, the ability to articulate and present one's ideas is essential for creating software that is not only accessible but also effectively communicates its purpose to both oneself and others.

We want to address the ease of documentation and reduction of the need to take action in order to do so.

Specific Problem: As personal passion projects increasingly compensate for limited professional experiences in the workforce, developers require a straightforward method to showcase their projects. Simultaneously, recruiters seek an easily digestible format to grasp the essence of these endeavors.

Our aim is to narrow the comprehension gap between these two user groups by offering a single, user-friendly software solution that is accessible to all.

Current Competitor Landscape



```
Enter 'q' to quit
a) Adds an entry to the diary :
v) Views a diary entry :
s) Let's us search through the diary entries :
Action : █
```



Pure note taking: notepad apps, code editor, command line app - simple but bad at everything

Understand by
User/Recruiters



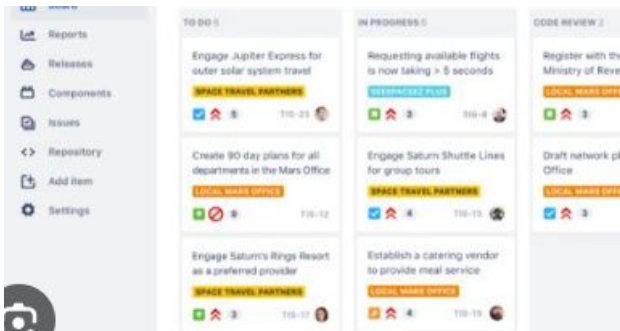
Personal blogs and websites: Github page, Medium, notion - not easy to create and maintain

Create and
Maintain

Record Thought
process



Project management: Github, Atlassian/Jira, Trello, Bitbucket, Google Cloud Source Repositories - lacks structure that is easily understandable by recruiters



Early Career Developer



Name: Jesse

Age: 24

Needs:

A platform to showcase his growing portfolio and technical skills to potential employers.

Background

Recent graduate with a few personal projects and internships.

Characteristics

- Eager to prove technical skills in a competitive job market.
- Keen on learning new coding practices.
- Actively participates in hackathons and coding competitions.

How CodeLog Meets The User

Jesse uses the VS Code widget to quickly document his coding process and decisions, making his projects more understandable to non-technical recruiters.

Senior Career Developer



Name: Emily

Age: 42

Needs:

A platform to demonstrate her technical expertise and leadership on complex projects.

Background

Senior developer looking to transition into a leadership role.

Characteristics

- Mentor to junior developers.
- Regularly leads workshops and technical presentations.
- Focused on sustainable software practices.

How CodeLog Meets The User

Emily uses CodeLog to create detailed project journals that include decision logs and sprint reviews, showcasing her ability to manage and direct software development life cycles.

Dev Company Recruiter



Name: Erika

Age: 29

Needs:

A quick and effective way to assess the technical capabilities and project involvement of potential candidates.

Background

Recruiter for a large tech company, constantly searching for talented developers.

Characteristics

- Always on the lookout for innovative recruitment strategies.
- Values clear and concise documentation of skills and projects.
- Adept at identifying potential through digital portfolios.

How CodeLog Meets The User

Erika uses CodeLog to review comprehensive project journals that give insight into candidates' problem-solving skills and coding proficiency without needing to dive into the codebase.





Solution – Definition

CodeLog is more than a portfolio website; it extends its functionality directly into developer's workflow with a VS Code extension which works as a pop-up widget.

CodeLog is designed to bridge the gap between developers showcasing their personal projects and recruiters seeking to understand these projects quickly and efficiently. It addressed the needs of both users groups (coders & recruiters) by facilitating the creation, sharing, and evaluation of project portfolios in a structured and flexible environment where

- developers can create comprehensive, visually engaging, and easily accessible project portfolios
- recruiters can identify and evaluate potential candidates effectively



Solution – Key Features

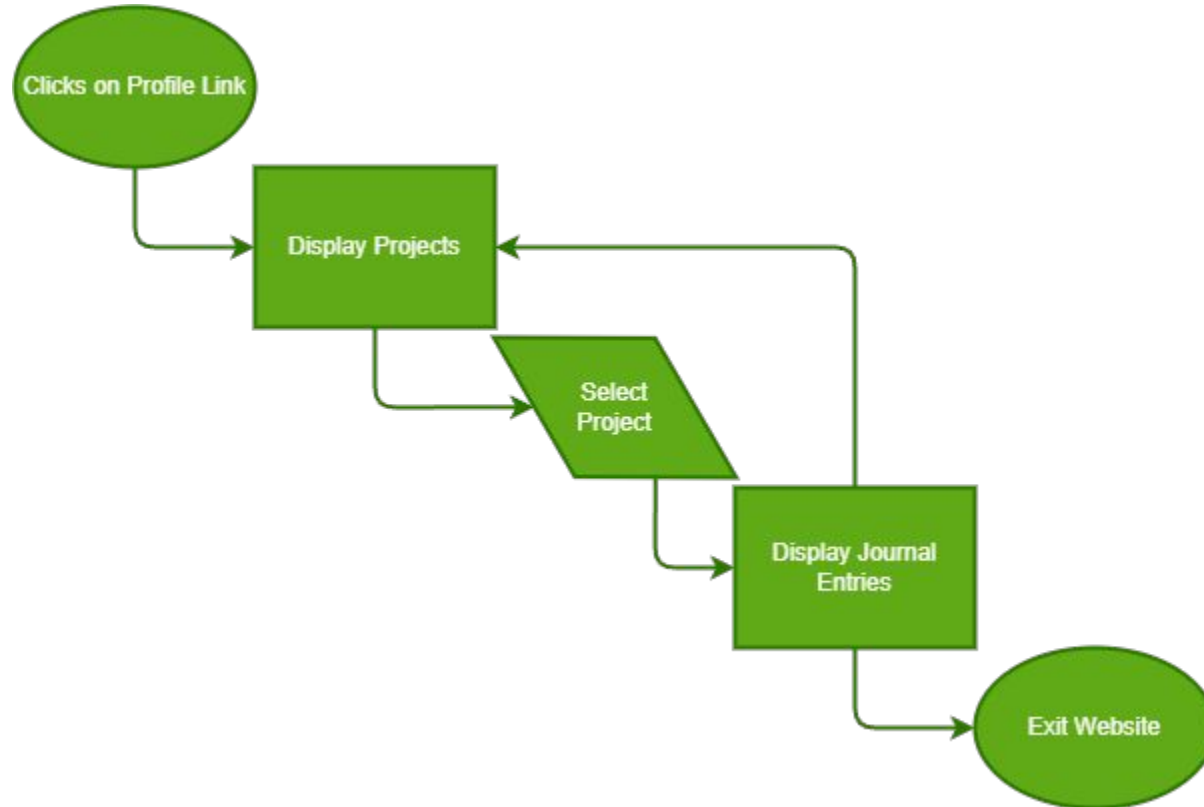
Coders' Perspective:

- A widget that works alongside coders' coding environment (VS Code)
- Users can highlight code and click a button to open this widget, and use the widget to select tags and text to a journal entry. (*The journal entry is based on preset templates*)
- The pop-up widget works as a journal entry. Users can choose one of the templates and start documenting in the given text boxes (*retrospective, sprint review, decision logs, bug review, etc.*)

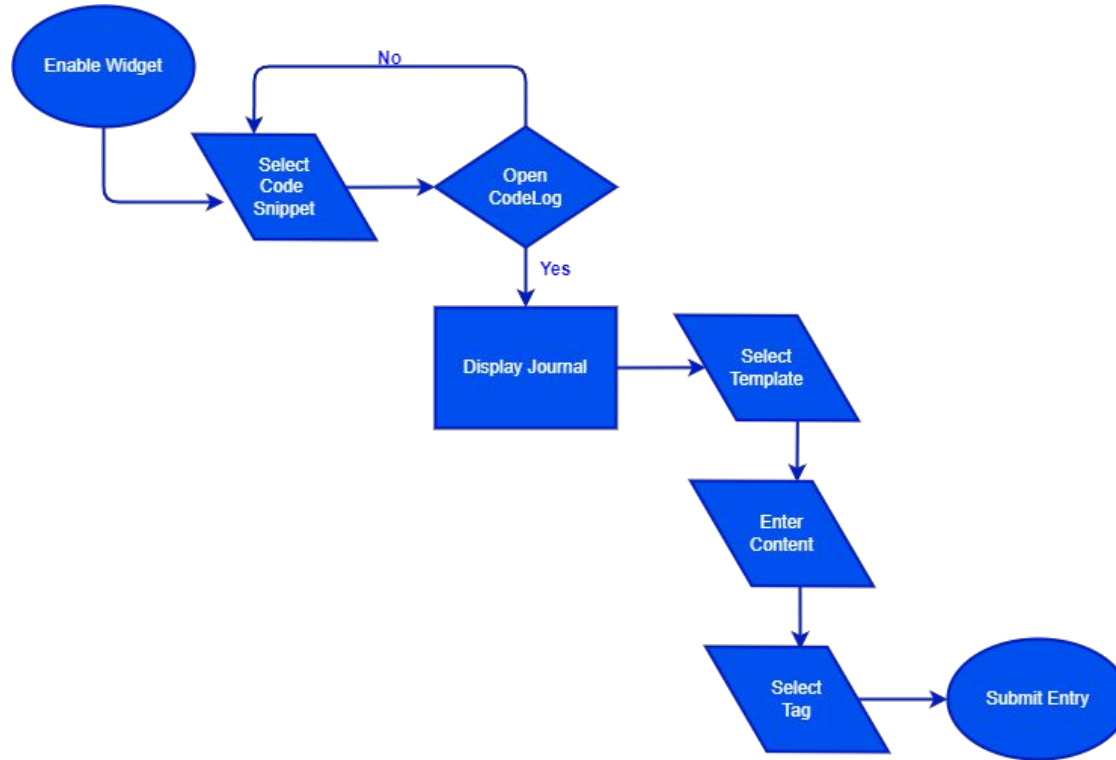
Recruiters' Perspective:

- Check out candidate profile via link on a web application
- Browse through all candidate's documented projects
- Journal entries for each project clearly shows candidate's development progress and personality as well as the respective codes

Website Flowchart



Extension Flowchart



Miro Design Board

https://miro.com/app/board/uXjVKMhqT8E=





Competitive Analysis & Updated Landscape

- A few commonly used competitors: **Notion**, **command line apps**, **GitHub**, **personal websites**

- **Problems:**

- Notion and personal websites require time and effort to set up/maintain
- Command line apps are too basic
- GitHub is not easy to navigate and lacks the overall process that goes into a project
- In general: these products are either too developer- or recruiter-oriented

- **How we will fix these problems:**

- Make both the developer and recruiter user experience enjoyable
- Templates make project organization easy and project navigation pleasant for readers
- Integration with coding environment creates convenience for the developer
- Allows devs to show off project development process for recruiters to easily see



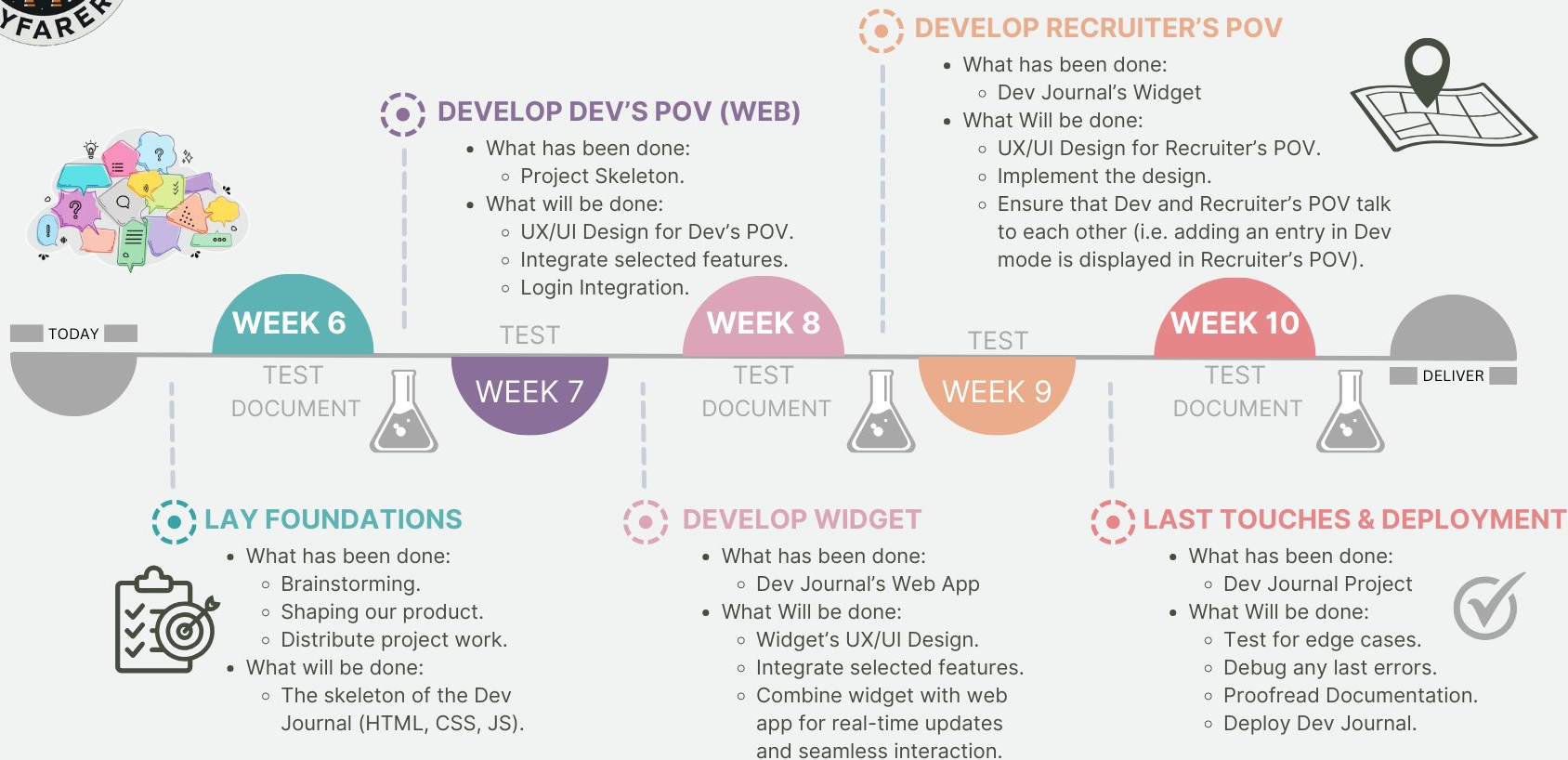
Risks



- **Security Risk**
 - Challenges: Data Privacy and Storing Data.
 - Strategy: Research and select APIs that offer multiple options to securely handle data while protecting user information and project details.
 - Web Storage API is a potential option
 - sessionStorage:
 - Challenges: Data get expired and deleted once the tab is closed, and is unretrievable.
 - Benefits: Can be used for security implementation, such as verification and timeout features.
 - localStorage:
 - Challenges: Since data is stored in plain text and accessible by any script on the same domain, this may lead to potential security risks. Storage is about 5MB. Limited accessibility since the data is stored locally.
 - Benefits: Data never get expired.
- **Technical Risk**
 - Challenges: Complexity of integrating widget/vs code extension.
 - Strategy: Use documentation to understand what is required for adding widget/vs code extension and see if it fits our project timeline.
- **Usability Risk**
 - Challenges: The user interface cannot be too complex while providing sufficient functionality
 - Strategy: Minimize learning curve and improve user experience by providing templates to help users generate their desired outputs.
- **Dependency Risk**
 - Challenges: Reliance and stability of external resources.
 - Strategy: Ensure project uses reliable and stable services (e.g. APIs, and packages, etc.) with an active development team, multiple version history, and available documentation.



Dev Journal Roadmap





Scope of the project

Feasible Scope

1. Portfolio Creation Tool:
 - A simple, intuitive interface for developers to upload and organize their projects
2. Search and Filter Mechanism for Recruiters:
 - Ability to save and revisit portfolios, possibly through a "favorites" system.
3. Feedback and Interaction System:
 - Features for recruiters to leave feedback or contact developers directly through the platform
4. Mobile Responsive Design:
 - Ensuring the platform is accessible and functional across all devices, enhancing user accessibility.

Aspirational Scope

1. AI-Powered Matching System:
 - An AI system that suggests portfolios to recruiters based on past searches, preferences, and interaction history.
2. Real-Time Collaboration Tools:
 - Features like live coding environments or shared whiteboards for on-the-spot assessments during recruiter-developer interactions
3. Comprehensive Analytics Dashboard:
 - Advanced analytics for users to track engagement with their portfolios, such as views, interactions, and feedback.
4. Integration with External APIs:
 - Allowing developers to import projects directly from code repositories or project management tools.



Why CodeLog

The tools available to the general public in today's market do not fulfill the needs of developers and recruiters at the same time. For example, Notion fulfills the needs of recruiters, but is not as convenient for developers, while Github is a good platform for developers, but is not an ideal platform for recruiters.

Therefore, we combined the advantages of these two platforms and designed CodeLog to be a pleasant user experience for both developers and recruiters. In our CodeLog we provide developers with widgets that work with the coding environment (VS Code), so that they can show or document their development process. At the same time, we also had the recruiter's experience in mind, so we made the interface as simple and clean as possible and provided clear diary entries for each project.