Sprint 4 Plan

Product Name: SpeedFill

Revision Number: 0
Revision Date: 2/28

Sprint Period: Feb 26 - Mar 12th

Goal: Complete remaining backlog from Sprint 3, focused on delivering a minimum viable product with quality of life tasks.

Task List (organized by user stories):

1. [Must Have] [Backlog] As a user, I want to be able to access and reuse previous documents so that I can reuse and reference them later. **[13 Hours]**
   - Task 1: Write a backend hook to convert a PDF into plain text **[2 Hours]**
   - Task 2: Write a backend hook for adding and retrieving PDFs **[3 Hours]**
   - Task 3: Connect dashboard to display user's associated info from backend **[3 Hours]**
   - Task 4: Migrate backend to use Supabase instead of Firebase **[5 Hours]**

2. [Must Have] [Backlog] As a user, I want clear and concise information so that I can easily apply the advice to my job applications. [**8 Total Hours**]
   - Task 1: On resume submit, route to correct result component **[3 Hours]**
   - Task 2: Added client loading state when waiting for API response **[2 Hour]**
   - Task 3: Create JSON schema for proper AI response parsing **[3 Hours]**

3. [Must Have] [Backlog] As a job applicant, I want to be able to automatically fill out job application pages so that I can save time. **[11 Total Hours]**
   - Task 1: Design a brand logo for the google extension **[2 Hours]**
   - Task 2: Design the UI for the google extension, login form and button to autofill page **[4 Hours]**
   - Task 3: Implement frontend components for the extension, including a popup interface for user interaction and login **[2 Hours]**
   - Task 4: Develop a script that scans job application web pages to detect input fields (e.g., name, email, experience, education) **[3 Hours]**

4. [Must Have] [Backlog] As a user, I want an extension to save my information so I can easily access relevant information about myself **[4 Total Hours]**
   - Task 1: Setup connection between google chrome extension and Supabase database to store and retrieve user information **[2 Hours]**
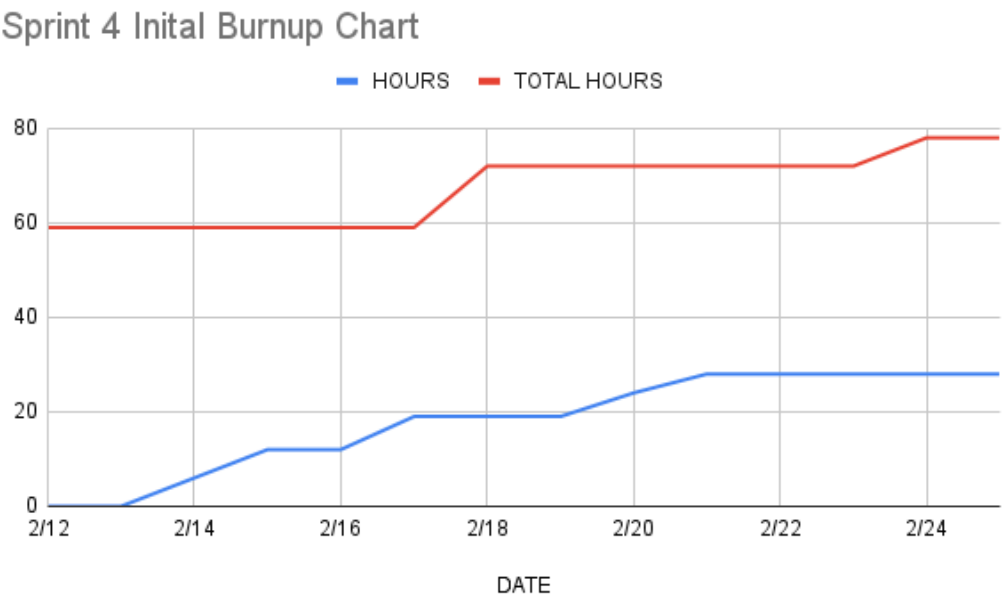   - Task 2: Implement auth within extension to let users login **[2 Hours]**

Team Roles
- Charles - **Product Manager, Backend Developer**
- Andy - **Frontend Developer**
- Colin - **Frontend Developer**
- Chan - **Backend Developer**
- Raj - **Backend Developer**

Initial Task assignment
- Charles
  - **User Story 1** - Task 1, 2, 3, 4
  - **User Story 2 -** Task 1
- Andy
  - **User Story 3** - Task 1, 2, 3
  - **User Story 2 -** Task 2
- Colin
  - **User Story 3** - Task 4
  - **User Story 4 -** Task 1, 2
- Chan
  - **User Story 2 -** Task 1
- Raj
  -

Initial Burnup

Sprint 4 Inital Burnup Chart

Initial Scrum Board

TO DO	IN PROGRESS	DONE	BLOCKED	BACKLOG
		letter and resume analysis [3 Hours]		
	Implement results components for cover letter and resume analysis [6 Hours]			
	Connect results components routing to forms interface and backend [2 Hours]			
		Implement forms page with resume upload, job description box, and additional details area [6 Hours]		
		Write a backend hook for adding and retrieving user data [3 Hours]		
		Write a backend hook for authenticating users [2 Hours]		
Write a backend hook to convert a PDF into plain text [2 Hours]				
				Write a backend hook for adding and retrieving PDFs [3 Hours]
		Implement dashboard UI [6 Hours]		
				Connect dashboard to display user's associated info from Firebase [7 Hours]
	Migrate backend to use Supabase instead of Firebase [4 Hours]			
		Create a better AI prompt to answer additional questions [1 Hour]		
		Implement results components for cover letter and resume analysis [6 Hours]		
		Redo the styling of frontend components with better css practices [4 Hours]		
				Connect results components routing to forms interface and backend [2 Hours]

TO DO	IN PROGRESS	DONE	BLOCKED	BACKLOG
				login form and button to autofill page [4 Hours]
				Implement frontend components for the extension, including a popup interface for user interaction and login [2 Hours]
				Develop a script that scans job application web pages to detect input fields (e.g., name, email, experience, education) [3 Hours]
				Setup connection between google chrome extension and Supabase database to store and retrieve user information [2 Hours]
				Implement auth within extension to let users login [2 Hours]
				Implement backend service to extract key skills, job titles, and industries from a user's resume [5 Hours]
				Develop search feature that uses extracted keywords to scrap relevant job postings [6 Hours]
				Format and display retrieve job listing in interface within extension [2 Hours]
				Develop a web scraper that extracts key titles from job posting [3 Hours]
				Implement logic to highlight and categorize details for better readability [2 Hours]
				Display the extracted job details in a visually clear format in the extension [1 Hours]

Scrum Times

- Monday (Work Session) - **11AM - 12AM**
- Wednesday (Work Session) - **12AM - 1PM**
- Friday (TA Meeting) - **11AM - 12PM**