Sprint 3 Plan

Product Name: LEMMA
Team Name: LEMMA
Sprint Completion Date: May 19, 2025
Revision Number: 1.0
Revision Date: May 5, 2025

Goal

Expand the application's functionality by implementing the Chrome extension for web content capture, adding automatic connection suggestions between notes, developing natural language Q&A capability, and creating import/export functionality to ensure data portability.

Task Listing, Organized by User Story

US3.1 - As a user, I want a Chrome extension that can save the current webpage to my notes so that I can capture online content.

1. Set up Chrome extension project structure (3 hours)
2. Create browser action popup UI for the extension (4 hours)
3. Add full-page capture functionality (5 hours)
4. Develop communication between extension and desktop app (5 hours)
5. Implement webpage metadata extraction (title, URL, date) (4 hours)
6. Test extension across different website structures (4 hours)
7. Package extension for Chrome Web Store submission (2 hours) Total for user story 3.1: 40 hours

US3.2 - As a user, I want the system to automatically suggest connections between notes based on content similarity so that I discover relationships I might have missed.

1. Design and implement text embedding generation for notes (5 hours)
2. Create algorithm for connection strength assessment (5 hours)
3. Implement background processing for connection analysis (4 hours)
4. Design UI for displaying suggested connections (5 hours)

5. Add user feedback mechanism for suggested connections (3 hours)
6. Implement connection acceptance/rejection functionality (4 hours)
7. Test with diverse note collections (3 hours) Total for user story 3.2: 35 hours

US3.3 - As a user, I want to ask natural language questions about my notes so that I can retrieve information conversationally.

1. Research and select appropriate language model for Q&A (5 hours)
2. Design architecture for local language model integration (6 hours)
3. Implement query processing pipeline (6 hours)
4. Create context retrieval system from note database (5 hours)
5. Develop answer generation functionality (7 hours)
6. Design and implement conversational UI (5 hours)
7. Add citation functionality to reference source notes (4 hours)
8. Test Q&A system with various question types (3 hours) Total for user story 3.3: 45 hours

US3.4 - As a user, I want to import and export my notes in common formats so that I can backup or migrate my data.

1. Define supported import/export formats (Markdown, PDF) (2 hours)
2. Implement Markdown export functionality (4 hours)
3. Create PDF export capability (5 hours)
4. Design and develop import functionality for Markdown files (4 hours)
5. Create batch import capability for multiple files (3 hours)
6. Test import/export with various file formats and structures (3 hours) Total for user story 3.4: 32 hours

Team Roles

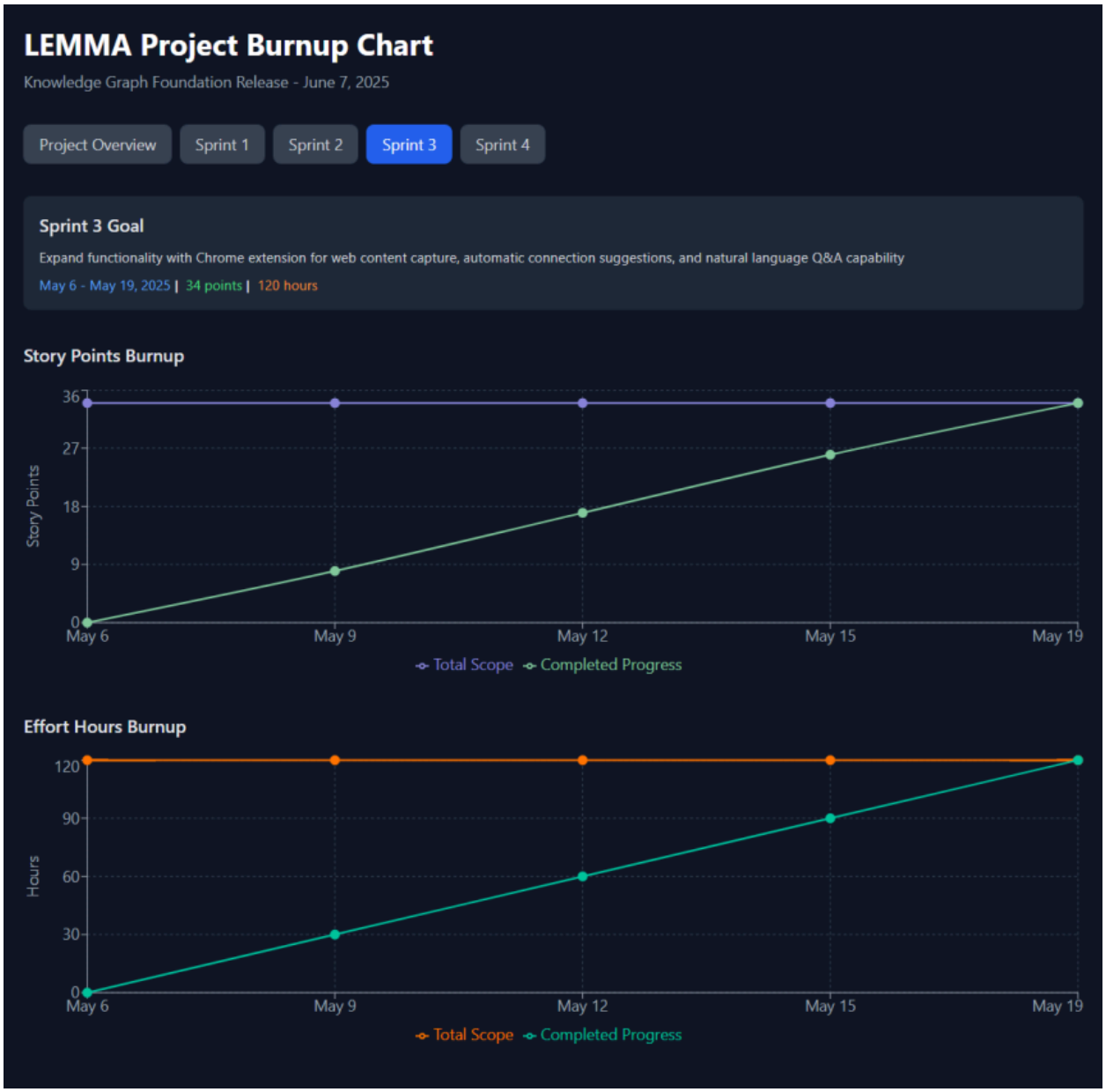
- Project Lead (Cory): Developer (assisting all areas)
- Team Member 1 (Howard): Frontend Developer (UI/UX, Q&A interface)
- Team Member 2 (Jiancheng): Scrum Master, Database/Backend Developer (connections, data processing)
- Team Member 3 (Mason): Chrome Extension Developer (primary extension work)

Initial Task Assignment

- Project Lead: US3.3, Research and select appropriate language model for Q&A
- Frontend Developer: US3.3, Design and implement conversational UI
- Database/Backend Developer: US3.2, Design and implement text embedding generation for notes
- Chrome Extension Developer: US3.1, Set up Chrome extension project structure

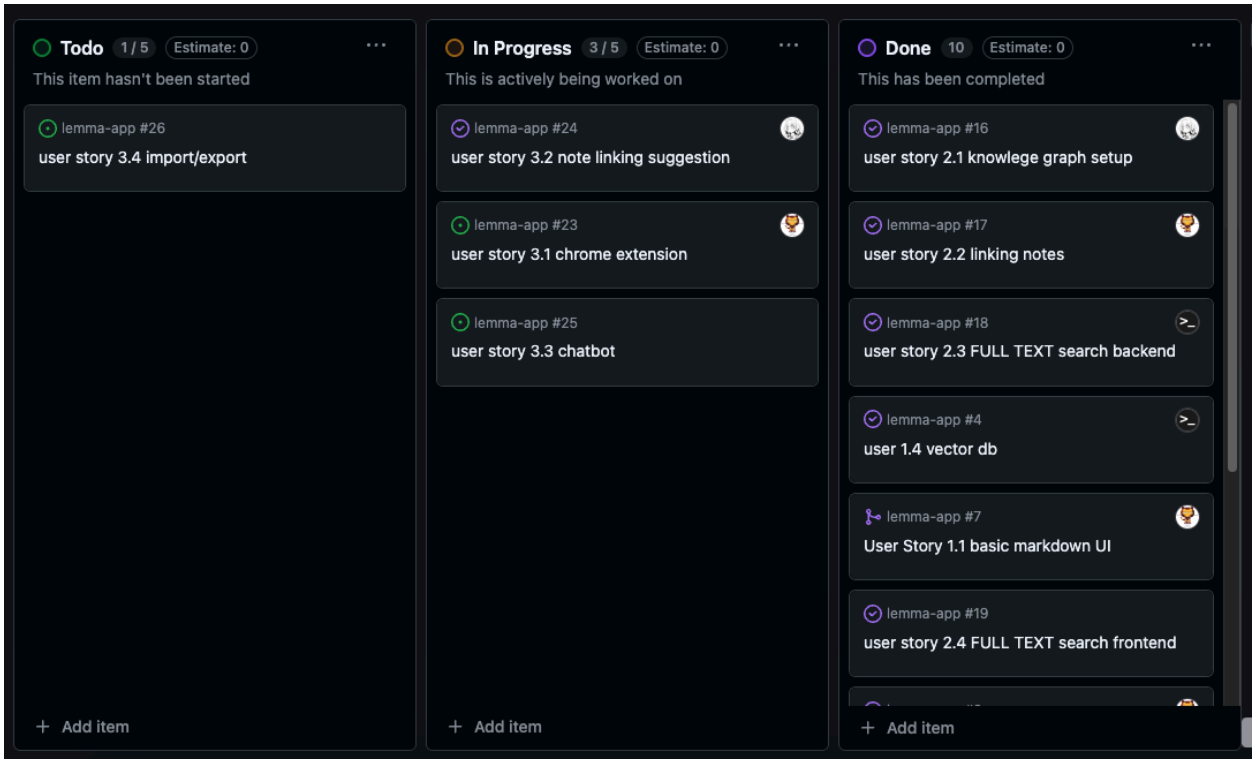
Initial Burnup Chart

[A physical burnup chart is created and located in the lab, labeled "Sprint 3 - LEMMA"]



Initial Scrum Board

[A physical scrum board is set up in the lab with four columns: User Stories, Tasks Not Started, Tasks in Progress, and Tasks Completed. All tasks are currently in the "Tasks Not Started" column, except for the initial assignments which are in "Tasks in Progress"]



Scrum Times

- Monday, Wednesday, Friday at 10:00 AM - Daily standup (TA/tutor will join Friday meeting during lab time)
- Additional ad-hoc scrum meetings will be scheduled as needed

Risk Assessment

1. Language model integration complexity - Mitigation: Begin with simpler model and scale up functionality
2. Chrome extension/desktop app communication challenges - Mitigation: Research and test communication methods early

3. Performance issues with vector similarity search - Mitigation: Implement optimization techniques and test with large datasets
4. Cross-platform compatibility for extensions - Mitigation: Focus on Chrome first with architecture designed for future browser support