

Process

Sprint Planning: At the start of our single sprint, our team conducted Sprint Planning meetings to define sprint goals and select backlog items to work on. Responsibilities were assigned based on team members' skill sets and aligned with project priorities.

Daily Standups: We conducted daily 15-minute standups to share our accomplishments, daily goals, and any obstacles. This fostered transparency and allowed for early issue resolution.

Sprint Execution: Tasks and user stories were managed through Jira. Slack served as our main communication tool for real-time collaboration. GitHub manages version control and code reviews.

Sprint Review: At the end of the sprint, we reviewed completed tasks and gathered feedback. Team members presented their work and discussed the sprint's successes and challenges.

Sprint Retrospective: We reflected on the sprint, identifying successes and areas for improvement. Discussions focused on adjusting processes and role allocations for future projects.

Scrum Documentation

4/5/25 - Goals: Initialize Spend AI Project

- Krish: Set up Django project structure and core app folders
 - Ankit: Initialized Git repository and first commit
 - Sri: Structured models for User and Financial Data
 - Charan: Created authentication system (registration, login, password reset)
 - Abhi: Set up Jira board and initial Scrum documentation
-

4/8/25 - Goals: Create Dashboard and Base Structure

- Krish: Designed the frontend in React
 - Ankit: Integrated home page with the base structure
 - Sri: Added dashboard page to base structure
 - Charan: Structured navigation links
 - Abhi: Structured footer with site links
-

4/10/25 - Goals: Build User Management and Financial Models

- Krish: Created User and Financial Advisor models
 - Ankit: Implemented user registration forms
 - Sri: Created login/logout functionality
 - Charan: Set up the admin site for managing users
 - Abhi: Documented updates and Jira tasks
-

4/12/25 - Goals: Set Up Income and Expense Tracking

- Krish: Defined Income and Expense models
 - Ankit: Added forms for income and expense entry
 - Sri: Integrated with Django Admin for managing entries
 - Charan: Implemented transaction categorization
 - Abhi: Maintained Scrum updates and task progress
-

4/15/25 - Goals: Implement Budgeting System

- Krish: Developed the Budget model
 - Ankit: Created a user view for budget creation and updates
 - Sri: Implemented budget validations and constraints
 - Charan: Integrated budget views with transaction categories
 - Abhi: Updated the admin to manage budget entries
-

4/18/25 - Goals: Create Financial Reports and Charts

- Krish: Designed report views and summaries
 - Ankit: Integrated Google Charts API for data visualization
 - Sri: Filtered reports by time periods
 - Charan: Managed dynamic chart rendering
 - Abhi: Updated Scrum document and sprint progress
-

4/21/25 - Goals: Integrate AI Financial Tips

- Krish: Connected OpenAI API for AI advice generation
 - Ankit: Created the user interface for the AI assistant
 - Sri: Built backend services for AI interactions
 - Charan: Tested AI-based financial advice responses
 - Abhi: Logged feedback and improved AI prompts
-

4/24/25 - Goals: Finalize Deployment

- Krish: Finalized deployment settings
- Ankit: Deployed to the production environment
- Sri: Conducted final testing and bug fixing
- Charan: Prepared demo video and portfolio integration
- Abhi: Finalized Scrum documentation

Scrum Roles		Design Development & Evaluation	
Scrum Master:	Charan Koyaguri	Full-Stack Developer:	Krish Arora, Ankit Chandra
Product Owner:	Ankit Chandra	Front-End Developer	Abhi Nookarapu
Developer:	Sri Ritvik Maddireddy, Krish Arora	Back-End Developer	Sri Ritvik Maddireddy, Charan Koyagrui

Calendar

GMT-05

SUN 19 MON 20 TUE 21 WED 22 THU 23 FRI 24 SAT 25

8 AM

9 AM

10 AM

11 AM

12 PM

1 PM

2 PM

3 PM

4 PM

5 PM

6 PM

7 PM

8 PM

9 PM

10 PM

11 PM

Ankit's Availability 10am - 12pm

Krish's Availability 10am - 12pm

Sri's Availability 10am - 12pm

Surya's Availability 11am - 12:15pm

Krish's Availability 10am - 1pm

Ankit's Availability 11am - 12pm

Group Availability 10:30am - 2:30pm

Charan's Availability 12:30 - 3pm

Group Availability 12:15 - 1:45pm

Charan's Availability 2 - 3pm

Krish's Availability 3 - 5pm

Surya's Availability 3 - 5pm

Mentor TA Meeting 3:30 - 4:45pm

Ankit's Availability 4 - 5pm

Group Availability 5 - 9pm

Ankit's Availability 6 - 8pm

Krish's Availability 6:30 - 8:30pm

Krish's Availability 6:30 - 8:30pm

Krish's Availability 6:30 - 8:30pm

Group Availability 6:30 - 9pm

2. User Stories and Backlog Presentation

JIRA: <https://georgiatechscrum.atlassian.net/>

The screenshot shows the Jira interface for a project named 'ScrumProjectManagement'. The top navigation bar includes 'Your work', 'Projects', 'Filters', 'Dashboards', 'Teams', 'Assets', 'Apps', and a 'Create' button. The left sidebar shows a search bar and a user profile icon labeled 'GK'. The main area displays a 'KAN board' with four columns: 'IDEA 8', 'TO DO', 'IN PROGRESS', and 'TESTING'. The 'IDEA 8' column contains five user stories, each with a title, a checkbox, a key (KAN-42, KAN-37, KAN-34, KAN-38), and a user icon. The 'TO DO', 'IN PROGRESS', and 'TESTING' columns are currently empty, with a '+ Create issue' button in the 'IN PROGRESS' column.

Projects / ScrumProjectManagement

KAN board

Q Search

IDEA 8

As a user, I want to access the app through a responsive graphical user interface (GUI)

☒ KAN-42

As a registered user, I want my financial data to be stored securely and backed up so that I can ensure my information is protected and recoverable.

☒ KAN-37

As a registered user, I want to add, edit, and delete my income so that I can accurately track my earnings over time.

☒ KAN-34

As a user, I want to modify my budget over time so that it reflects my changing financial needs.

☒ KAN-38

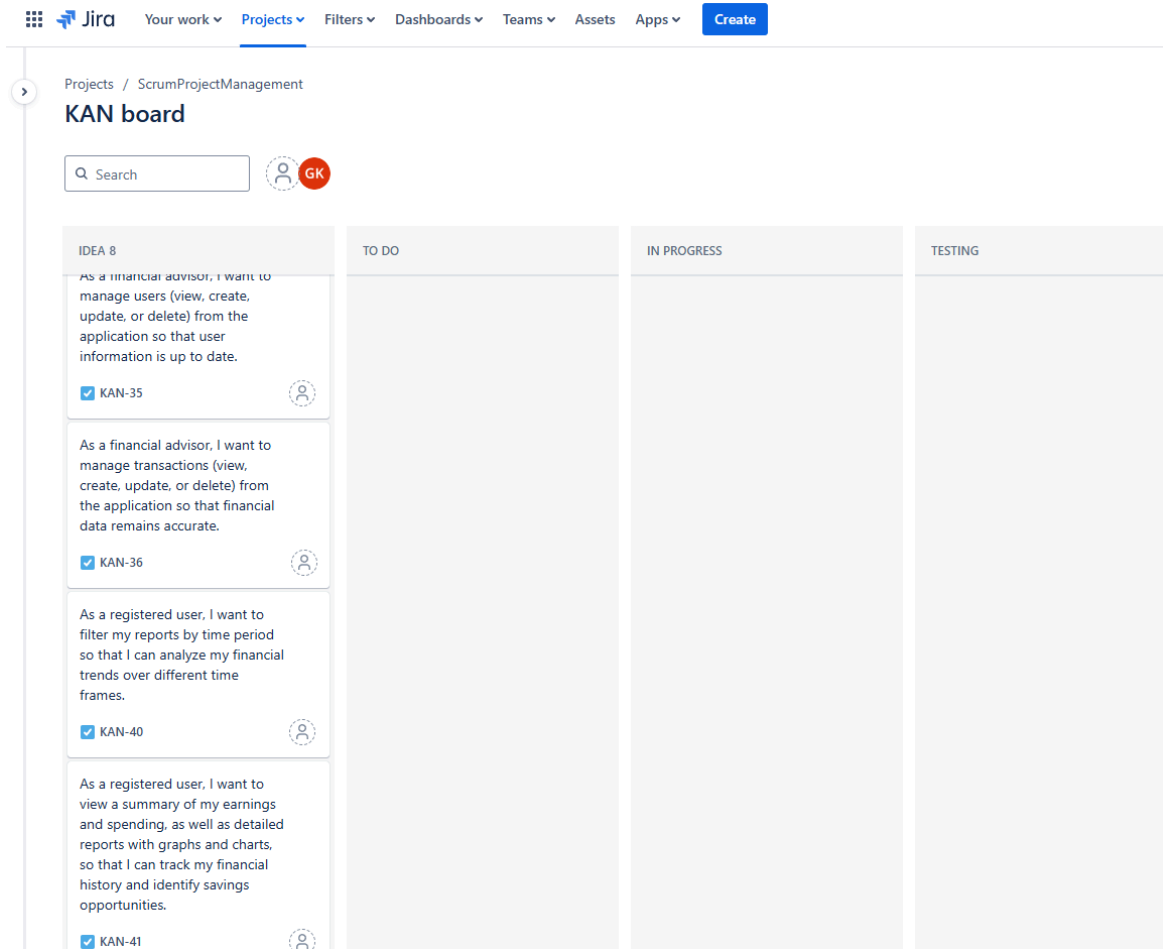
As a financial advisor, I want to manage users (view, create, update, or delete) from the

TO DO

IN PROGRESS

+ Create issue

TESTING



We **prioritize user stories** related to **administration**, managing budgets (**adding/removing categories**), and handling transactions (**adding/editing/removing**). These **core features** are essential for the application’s foundation, ensuring users can track their finances effectively. In the **next sprint**, we’ll shift focus to **financial planning features** and **user account management**, aligning with the project’s overall objectives.

3. When to Meet the TA

Weekly Check-In Times:

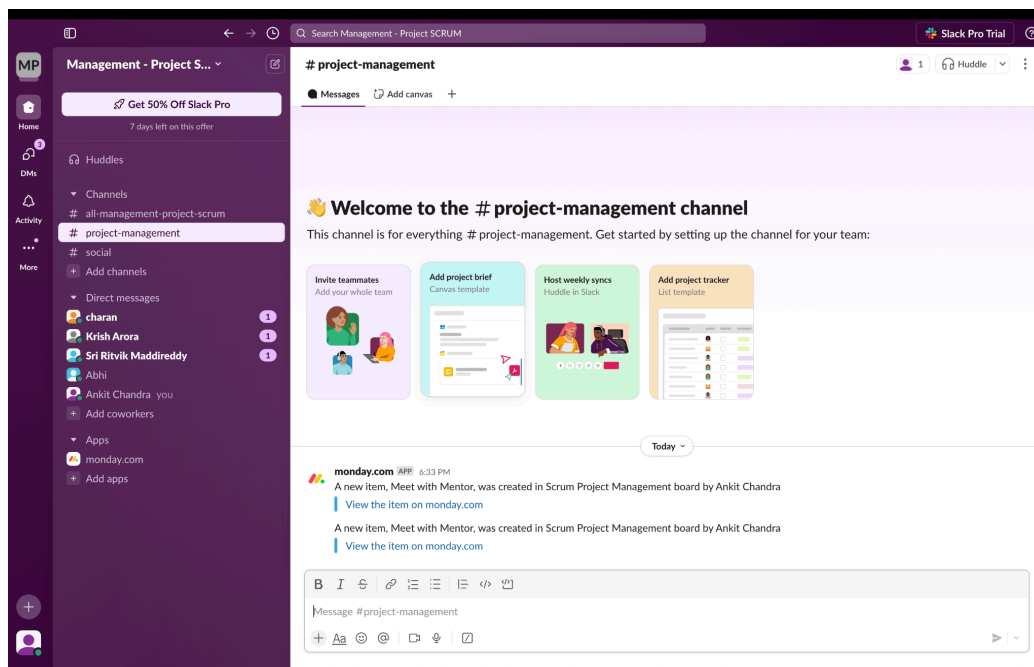
- Tuesday: 3:30 P.M. - 4:45 P.M. (In Class)
- Thursday: 5:30 P.M. - 6:30 P.M. (CCB 267)

4. Staying in Touch

For this project, we’ll be using **Slack** for general communication and **Discord** for direct communication with our mentor TA. Any technical issues will be flagged in **GitHub**, which will

automatically sync with our other workspaces. Additionally, individual priority updates will be posted in the **Slack channel**. We'll address issues and disagreements during our weekly stand-ups and in **Slack channels**. The Project Owner and Scrum Master will lead these discussions, gathering direct feedback from the developers. Our decision-making process revolves around focusing on a user story, suggesting potential solutions, and discussing different approaches. Once we've reached a consensus, the product manager will approve the best solution and work with the scrum master to assign tasks to each developer.

SLACK: <https://managementpro-wsy7439.slack.com/>



Sprint 1 Demo Retrospective

Introduction

Our sprint occurred from 4/5/2025 to 4/24/2025. In the following retrospective, we reflect on what worked well, what did not work well, possible issues, and strategies for improvement. By reflecting on our development process, we aim to improve collaboration and ensure success in future projects.

What Worked Well? What Did Not Work Well?

What Worked Well:

- **Scrum Roles:** We clearly defined team roles and responsibilities, helping us stay organized and maintain consistent progress.
- **Task Management:** We used Jira effectively to track user stories and progress, which helped us prioritize key features like expense tracking, budgeting, and AI tips.
- **Communication:** Slack and Discord helped us maintain real-time updates and foster efficient discussions, especially when collaborating with the mentor TA.

What Did Not Work Well:

- **Testing:** We often moved on to new features without thoroughly testing completed functionality, leading to hidden bugs that surfaced later.
- **External API Integration:** Integration with third-party APIs like Gemini and Google Charts took longer than expected due to insufficient early research.
- **Timeline Crunch:** Towards the end, we experienced time pressure, especially with deployment and bug fixes, because some earlier tasks took longer than anticipated.

Issues and Strategies for Improvement

Primary Issues:

- **Testing:** Limited intermediate testing resulted in missed bugs.
- **External Dependencies:** Delays in understanding and integrating external APIs impacted our final schedule.

Strategies for Improvement:

- **Strategy One:** Dedicate explicit Jira tasks for testing and require peer testing before marking a feature complete.

- **Strategy Two:** Allocate buffer time in future schedules specifically for researching and integrating external services early in the sprint.

Conclusion

Overall, the Spend.AI sprint was highly productive and successful, especially considering the project's technical depth. We plan to continue applying structured planning and communication practices while improving testing discipline and external integration timelines in future projects.