

FAIRWEATHER APP

Scrum Master [Timi Ogunjobi] : Timi Ogunjobi

timiasservice@gmail.com

Sprint Planning Documentation

1. Sprint Planning Notes

This document summarizes the outcomes of the sprint planning meeting, providing clarity on the sprint goals, selected user stories, and task breakdowns.

Sprint Details

- **Sprint Number:** Sprint 1
- **Sprint Duration:** 2 Weeks (10 Working Days)
- **Sprint Start Date:** [Will Communicate]
- **Sprint End Date:** [Will Communicate]
- **Sprint Team:**
 - Frontend Developers (2)
 - Backend Developers (2)
 - UX/UI Designer (1)
 - QA Engineer (1)
 - DevOps Engineer (1)

Sprint Goal

The primary goal of **Sprint 1** is to develop the core user registration and authentication features, allowing users to create accounts and log in securely. This is part of building the foundation of the *Fairweather* platform.

Selected User Stories for Sprint 1

The following user stories were selected based on priority, estimated effort, and available team capacity:

User Story ID	User Story	Story Points	Priority	Acceptance Criteria
US-01	As a new user, I want to register using my email, social media, or Google account.	3	High	Users can create accounts using different methods.
US-02	As a registered user, I want to log in securely using my credentials.	2	High	Users can log in securely and access their accounts.
US-03	As a user, I want to reset my password via email.	2	Medium	Users can reset their password using a secure email-based process.
US-04	As a user, I want to enable multi-factor authentication (MFA) for additional security.	5	High	MFA is implemented, and users can enable/disable it from their accounts.

Task Breakdown for Sprint 1

Task ID	Task Description	Assigned To	Estimated Hours	Dependencies	Status
T-01	Design user registration form UI/UX for web	UX/UI Designer	8	None	To Do
T-02	Implement user registration API	Backend Developer 1	12	Database setup	To Do
T-03	Set up OAuth 2.0 for social media login integration	Backend Developer 2	10	T-02	To Do
T-04	Develop frontend login form for web and mobile	Frontend Developer 1	8	T-01	To Do
T-05	Implement password reset feature via email	Backend Developer 1	10	Email service setup	To Do
T-06	Set up MFA (e.g., Google Authenticator integration)	Backend Developer 2	14	T-02	To Do
T-07	Frontend implementation of MFA feature	Frontend Developer 2	12	T-06	To Do
T-08	Unit testing of registration, login, password reset, and MFA	QA Engineer	10	T-02, T-05, T-06	To Do
T-09	Set up CI/CD pipeline for continuous integration	DevOps Engineer	8	T-02	To Do
T-10	Sprint Review preparation and demo setup	Scrum Master [Timi Ogunjobi]	4	T-01 to T-08	To Do

2. Capacity Planning Documentation

Capacity planning helps ensure the team’s workload aligns with their available capacity for the sprint.

Team Capacity Calculation

- **Sprint Duration:** 2 weeks (10 working days)
- **Hours per Day:** 6 hours (assuming focus time, excluding meetings and breaks)
- **Total Available Capacity:** 6 hours x 10 days = 60 hours per team member

Team Member	Role	Total Hours Available	Planned Hours	Buffer Hours	Capacity Utilization (%)
Frontend Developer 1	Frontend Dev	60	48	12	80%
Frontend Developer 2	Frontend Dev	60	50	10	83%
Backend Developer 1	Backend Dev	60	50	10	83%
Backend Developer 2	Backend Dev	60	52	8	87%
UX/UI Designer	UX/UI	60	40	20	67%
QA Engineer	QA	60	40	20	67%
DevOps Engineer	DevOps	60	36	24	60%

Availability Chart

Day	Frontend Developer 1	Frontend Developer 2	Backend Developer 1	Backend Developer 2	UX/UI Designer	QA Engineer	DevOps Engineer
Day 1	Available	Available	Available	Available	Available	Available	Available
Day 2	Available	Available	Available	Available	Available	Available	Available
Day 3	Available	Available	Available	Available	Available	Available	Available
Day 4	Available	Available	Available	Available	Available	Available	Available
Day 5	Available	Available	Available	Available	Available	Available	Available
Day 6	Available	Available	Available	Available	Available	Available	Available
Day 7	Available	Available	Available	Available	Available	Available	Available
Day 8	Available	Available	Available	Available	Available	Available	Available
Day 9	Available	Available	Available	Available	Available	Available	Available
Day 10	Available	Available	Available	Available	Available	Available	Available

Work Allocation

Tasks are allocated based on the estimated hours and availability:

- **Frontend Developers:** User login, MFA frontend, form design, and sprint review preparation.
- **Backend Developers:** User registration, API development, OAuth setup, password reset, and MFA backend.
- **UX/UI Designer:** Design user registration forms and refine user interfaces for login/MFA.

- **QA Engineer:** Unit testing and preparation for sprint demo.
- **DevOps Engineer:** CI/CD setup and integration with the codebase.

3. Task Board Setup

Task Board Setup for Sprint 1 (Kanban Board)

Task Status	Task ID	Task Description	Assigned To	Comments/Notes
To Do	T-01	Design user registration form UI/UX for web	UX/UI Designer	No dependencies
	T-02	Implement user registration API	Backend Developer 1	No dependencies
	T-03	Set up OAuth 2.0 for social media login integration	Backend Developer 2	Dependent on T-02
	T-04	Develop frontend login form for web and mobile	Frontend Developer 1	Dependent on T-01
	T-05	Implement password reset feature via email	Backend Developer 1	Dependent on email service setup
	T-06	Set up MFA (e.g., Google Authenticator integration)	Backend Developer 2	Dependent on T-02
	T-07	Frontend implementation of MFA feature	Frontend Developer 2	Dependent on T-06
	T-08	Unit testing of registration, login, password reset, and MFA	QA Engineer	Dependent on development completion
	T-09	Set up CI/CD pipeline for continuous integration	DevOps Engineer	No dependencies
	T-10	Sprint Review preparation and demo setup	Scrum Master [Timi Ogunjobi]	Dependent on overall sprint progress
In Progress	T-01	Design user registration form UI/UX for web	UX/UI Designer	Work started
Review	T-02	Implement user registration API	Backend Developer 1	QA testing required
Done	None			

Daily Scrum Reports

1. Daily Standup Notes

Daily Standup Notes capture the key points discussed during the daily standup meeting, focusing on what each team member accomplished yesterday, what they plan to accomplish today, and any impediments they're facing.

Sample Daily Standup Notes (Day 3 of Sprint 1)

Date: [07/01/2021]

Duration: 15 Minutes

Facilitator: Scrum Master [Timi Ogunjobi]

Attendees:

- Frontend Developer 1
- Frontend Developer 2
- Backend Developer 1
- Backend Developer 2
- UX/UI Designer
- QA Engineer
- DevOps Engineer

Team Updates

Team Member	Yesterday's Work	Today's Plan	Impediments Raised
Frontend Developer 1	Completed UI design for the login form (Web).	Begin frontend development of user registration form (Web).	No blockers reported.
Frontend Developer 2	Worked on the MFA frontend design mockup.	Implement frontend MFA logic and validation (Web).	Awaiting backend API for MFA integration (T-06).
Backend Developer 1	Developed registration API; set up database for user profiles.	Test registration API and start work on the password reset.	Issues with database connection in dev environment.
Backend Developer 2	Worked on OAuth setup for social login integration.	Continue OAuth integration and begin MFA setup.	Delays due to API documentation gaps from the vendor.
UX/UI Designer	Designed the registration form UI for web and mobile.	Design user login form UI for mobile.	No blockers reported.
QA Engineer	Prepared test cases for user registration and login.	Begin testing registration API and UI components.	Needs access to the latest code deployment for testing.
DevOps Engineer	Configured CI/CD pipeline for automated builds.	Deploy code updates to the testing environment.	No blockers reported.

Additional Notes

- The team is aligned with sprint goals, but there are some technical blockers that need immediate attention to prevent delays.
- Frontend Developer 2 needs backend API support for MFA integration; backend team to prioritize API readiness.
- Backend Developer 1 to resolve database connectivity issues with DevOps assistance.

2. Impediment Log

The Impediment Log tracks issues or blockers raised by team members during the sprint. It includes the date when the impediment was reported, the team member who reported it, the status of the impediment, actions taken, and resolution details.

Sample Impediment Log for Sprint 1

Impediment ID	Date Reported	Reported By	Impediment Description	Status	Assigned To	Action Taken	Resolution/ETA
IMP-01	[07/01/2021]	Backend Developer 1	Database connection issues in the development environment. Delay in receiving API documentation for OAuth from vendor.	Open	Backend Dev, DevOps	DevOps investigating server configs and firewall.	Resolution expected by [Date].
IMP-02	[07/01/2021]	Frontend Developer 2	Limited access to the latest code deployment for testing.	In Progress	Backend Developer 2	Contacted vendor for updated documentation.	Pending vendor response by [Date].
IMP-03	[07/01/2021]	QA Engineer	Backend API for MFA is not ready for integration.	Open	DevOps Engineer	DevOps to enable QA access to the testing server.	Resolution expected by end of day.
IMP-04	[07/01/2021]	Frontend Developer 2	API documentation gaps delaying OAuth setup.	Open	Backend Developer 2	Backend team to prioritize MFA API completion.	Estimated completion by [Date].
IMP-05	[07/01/2021]	Backend Developer 2		In Progress	Backend Developer 2	Engaged vendor support for detailed docs.	Expected update within 2 working days.

3. Impediment Log Details

Impediment Resolution Process

1. **Identification:** Team members identify impediments during daily standups or work.
2. **Logging:** Scrum Master [Timi Ogunjobi] logs the impediment in the Impediment Log.
3. **Prioritization:** The Scrum Master [Timi Ogunjobi], in consultation with the team, prioritizes impediments based on their impact on the sprint.
4. **Assignment:** The Scrum Master [Timi Ogunjobi] assigns the resolution of the impediment to the appropriate team member or external party.
5. **Follow-up:** The Scrum Master [Timi Ogunjobi] monitors progress and updates the log until the impediment is resolved.
6. **Resolution:** Impediments are closed once resolved, with details recorded for future reference and learning.

Key Metrics for Impediment Management

- **Time to Resolution:** Measures the average time taken to resolve impediments, helping to identify potential process improvements.
- **Number of Open Impediments:** Indicates how many unresolved issues exist at any given time, helping to assess the team's progress in removing blockers.
- **Repeat Impediments:** Tracks whether certain impediments reoccur, suggesting areas for process optimization or risk management.

Sprint Review Summary and Increment Documentation

1. Sprint Review Summary

The **Sprint Review Summary** captures feedback from stakeholders, evaluates the sprint's achievements, and outlines potential impacts on the Product Backlog.

Sprint Details

- **Sprint Number:** Sprint 1
- **Sprint Duration:** 2 Weeks (10 Working Days)
- **Sprint Start Date:** [Will Communicate]
- **Sprint End Date:** [Will Communicate]
- **Facilitator:** Scrum Master [Timi Ogunjobi]
- **Attendees:**
 - Product Owner
 - Development Team
 - Stakeholders (Product Sponsor, Marketing Team, Customer Support Lead)
 - Scrum Master [Timi Ogunjobi]

Sprint Goal

The goal of **Sprint 1** was to implement core user registration and authentication features, including email registration, login, password reset, and multi-factor authentication (MFA). The team aimed to build a strong foundation for user management on the *Fairweather* platform.

Sprint Achievements

The following user stories were completed and demonstrated during the sprint review:

User Story ID	User Story	Status	Demonstrated Functionality
US-01	As a new user, I want to register using my email, social media, or Google account.	Completed	Demonstrated registration with email, Facebook, and Google.
US-02	As a registered user, I want to log in securely using my credentials.	Completed	Showcased secure login with email and password.
US-03	As a user, I want to reset my password via email.	Completed	Demonstrated email-based password reset functionality.
US-04	As a user, I want to enable multi-factor authentication (MFA) for additional security.	Completed	Showcased MFA setup and login with Google Authenticator.

Feedback from Stakeholders

Feedback was collected from the Product Owner, marketing, and customer support teams. Below is a summary of the key points discussed:

Stakeholder	Feedback	Impact on Product Backlog
Product Owner	Registration process is smooth, but onboarding guidance (e.g., tooltips or pop-ups) should be added for clarity.	New user onboarding guidance user story added to backlog for prioritization.
Marketing Team	Positive response to the social login integration, suggesting further exploration of LinkedIn as an option.	LinkedIn integration added as a low-priority item to the backlog.
Customer Support Lead	Suggested a clearer error message for MFA failures to enhance user experience during login.	User story to refine error messages for MFA failures added to backlog.
Product Sponsor	Impressed with the demo but requested faster response times during registration.	Performance optimization tasks added to the backlog for future sprints.

Additional Discussion Points

- **Feature Demonstration:** Each user story was demonstrated live, with the development team explaining the implemented features and handling questions from stakeholders.
- **User Story Completion Rate:** 100% of the planned user stories were completed within the sprint.
- **Product Backlog Refinement:** Based on the feedback received, three new user stories were added to the Product Backlog:
 1. Add onboarding guidance for new users.
 2. Implement LinkedIn as an additional social login option.
 3. Refine error messages for MFA failures.

2. Increment Documentation

The **Increment Documentation** details the features and functionalities completed during the sprint, as well as their readiness for release.

Overview of Sprint 1 Increments

The increment achieved during Sprint 1 focused on building the core user registration and authentication system, setting up the initial user management infrastructure on the *Fairweather* platform.

Completed Increments

Feature/Functionality	Description	Status	Ready for Release	Additional Notes
User Registration	Users can register using email, Facebook, and Google accounts.	Completed	Yes	Minor UI adjustments suggested for social login buttons.
Secure User Login	Users can log in using email and password. Secure authentication implemented with OAuth 2.0.	Completed	Yes	Meets acceptance criteria; no issues identified during testing.
Password Reset via Email	Users can reset passwords via a secure email-based process.	Completed	Yes	Ready for release; includes validation for email input.
Multi-Factor Authentication (MFA)	MFA using Google Authenticator has been implemented to enhance user security.	Completed	Yes	Clear setup and successful login using MFA verified.

Increment Demonstration

Each feature was demonstrated to stakeholders, showing the following workflows:

1. User Registration Workflow:

- New users can register by entering their email, setting a password, and receiving a verification email.
- Users can also register using social media (Facebook and Google), with OAuth 2.0 handling authentication.
- Error handling includes messages for invalid inputs and existing accounts.

2. User Login Workflow:

- Users can log in securely using their credentials (email and password).

- The login process includes checks for incorrect passwords and locked accounts after multiple failed attempts.

3. Password Reset Workflow:

- Users can initiate a password reset by entering their email, receiving a reset link, and setting a new password.
- Password reset links are valid for a limited duration to ensure security.

4. Multi-Factor Authentication (MFA) Workflow:

- Users can enable MFA from their account settings, scan a QR code with Google Authenticator, and verify the setup.
- Login with MFA requires entering a time-based one-time password (TOTP) after entering the regular password.

Testing and QA Summary

- **Unit Testing:** Completed for all increments with 100% coverage of registration, login, password reset, and MFA features.
- **Integration Testing:** Successfully tested the integration between frontend and backend components, including social login APIs.
- **User Acceptance Testing (UAT):** Conducted with select users, yielding positive feedback for user experience and functionality.
- **Performance Testing:** Basic performance checks indicated acceptable response times for registration and login; however, further optimization is required as the user base scales.

Ready for Release

All the increments from Sprint 1 are considered **ready for release** into the development environment. The features will be deployed in a staging environment first, followed by additional testing before the official launch.

Sprint Retrospective Documentation

1. Retrospective Notes

The **Sprint Retrospective** allows the team to reflect on the past sprint, celebrating successes and identifying areas for improvement.

Sprint Details

- **Sprint Number:** Sprint 1
- **Sprint Duration:** 2 Weeks (10 Working Days)
- **Facilitator:** Scrum Master [Timi Ogunjobi]
- **Participants:**
 - Product Owner
 - Frontend Developers
 - Backend Developers
 - UX/UI Designer
 - QA Engineer
 - DevOps Engineer

Retrospective Summary

- **Date of Retrospective:** [07/01/2021]
- **Duration:** 45 Minutes

Structure of Retrospective Meeting

The retrospective was divided into three main sections: **What Went Well**, **What Didn't Go Well**, and **What Can Be Improved**. The team participated in an open discussion using a collaborative online whiteboard (e.g., Miro, Mural) to gather insights.

What Went Well

Positive Aspects	Details/Impact
User Story Completion	All planned user stories were completed, with 100% acceptance during the sprint review.
Smooth Collaboration	The team worked well together, with clear communication and willingness to help each other.
Quick Issue Resolution	Impediments were addressed promptly, reducing potential delays.
Effective CI/CD Integration	The CI/CD pipeline setup was successful, allowing for smooth code integration and testing.
Stakeholder Feedback	Positive feedback from stakeholders on user registration and MFA features.

What Didn't Go Well

Challenges	Details/Impact
API Documentation Delays	Delays in receiving complete OAuth API documentation affected the integration timeline.
Database Connectivity Issues	Development was slowed down due to initial database connectivity problems, requiring DevOps intervention.
Testing Environment Access	The QA Engineer experienced limited access to the testing environment, delaying the testing process.
User Onboarding Gaps	During the review, it was noted that the registration process lacked onboarding guidance for new users.

What Can Be Improved

Suggestions for Improvement	Proposed Actions
Faster API Documentation Access	Establish stronger communication with third-party vendors to ensure timely access to documentation.
Improve Development Environment Stability	DevOps team to perform regular environment checks to prevent connectivity issues.
Early Access for QA	Grant QA team earlier access to the testing environment to reduce delays in the QA process.
Add Onboarding Guidance for Users	Product Owner to prioritize a user story for implementing onboarding guidance in the next sprint.

2. Improvement Action Plan

The **Improvement Action Plan** outlines specific steps the team will take to address issues identified in the retrospective and enhance the process in the upcoming sprint.

Improvement Action Items for Sprint 2

Improvement Area	Action Item	Responsible Team Member	Deadline	Expected Outcome
API Documentation Access	Establish a clear communication channel with third-party API vendors, including setting response SLAs. DevOps team to perform pre-sprint	Scrum Master [Timi Ogunjobi], Backend Devs	Start of Sprint 2	Faster resolution of API-related dependencies.
Environment Stability	checks on the development and testing environments to prevent issues.	DevOps Engineer	Ongoing	Fewer technical issues in the development environment.
Early QA Access	Grant QA team access to the testing environment as soon as features are code-complete, not at sprint end.	DevOps Engineer, QA Engineer	Mid-Sprint 2	QA team can start testing earlier, reducing testing delays.
User Onboarding Guidance	Prioritize a new user story for onboarding guidance in Sprint 2, focusing on tooltips and pop-ups during registration.	Product Owner	Sprint Planning	Improved user experience during registration for new users.

3. Team Health Check

The **Team Health Check** documents the overall morale, collaboration, and process effectiveness based on insights from the retrospective. It helps monitor the team's well-being and effectiveness over time.

Sprint 1 Team Health Check

Category	Rating (1-5)	Team Comments	Suggestions for Improvement
Morale	4	The team felt positive about achieving the sprint goal and the overall collaboration.	Continue to celebrate small wins and recognize individual contributions.
Collaboration	5	Strong communication and teamwork across frontend, backend, and QA.	Maintain daily check-ins and encourage open feedback during standups.
Process Effectiveness	3	CI/CD setup was effective, but testing environment access was limited, causing delays.	Implement early QA access and regular development environment checks.
Workload Balance	4	Workload was generally well-balanced, but some backend tasks took longer than expected due to dependencies.	Consider allocating more time for complex backend tasks in sprint planning.
Responsiveness to Impediments	4	Impediments were addressed promptly, reducing the impact on the sprint timeline.	Continue prioritizing rapid impediment resolution.
Stakeholder Engagement	5	Stakeholders provided constructive feedback, showing active engagement.	Continue to involve stakeholders in sprint reviews and planning.

Team Health Check Summary

- **Overall Rating:** 4/5
- **General Sentiment:** Positive and motivated.
- **Key Strengths:** Strong collaboration and stakeholder engagement.
- **Areas for Improvement:** Faster QA access, improved API communication, and proactive environment checks.

Release Planning Documentation

1. Release Plan

The **Release Plan** outlines the goals, target dates, features, dependencies, and potential risks associated with the upcoming release of the *Fairweather* platform.

Release Overview

- **Release Name:** *Fairweather v1.0 – MVP Release*
- **Release Type:** Major Release (Minimum Viable Product)
- **Target Release Date:** [07/01/2021]
- **Release Owner:** Product Owner
- **Release Manager:** Scrum Master [Timi Ogunjobi]
- **Development Duration:** 3 Sprints (6 Weeks)

Release Goals

- Deliver a fully functional MVP of the *Fairweather* digital library platform with core features including user registration, login, password reset, subscription management, content discovery, and basic content consumption (reading and video).
- Gather user feedback from early adopters to refine the platform and prioritize future features.

Features Included in the Release

The table below provides an overview of the key features planned for *Fairweather v1.0* release, along with their associated user stories.

Feature	User Stories	Sprint	Status	Dependencies
User Registration	US-01: Register using email, social media, or Google account.	Sprint 1	Completed	None
User Authentication	US-02: Secure login with MFA; US-03: Password reset via email.	Sprint 1	Completed	OAuth integration, MFA setup
Subscription Management	US-05: View, select, and purchase subscription plans; US-06: Secure payment gateway integration	Sprint 2	In Progress	Payment gateway setup, subscription API readiness
Content Discovery	US-04: Search and filter content; US-09: Content recommendations.	Sprint 2	In Progress	Content indexing, search API setup
E-Book Reader	US-07: Basic e-book reading with font size and background adjustments.	Sprint 3	Planned	Frontend and backend synchronization for e-book

Feature	User Stories	Sprint	Status	Dependencies
				content
Video Playback	US-09: Video player with subtitles and playback options.	Sprint 3	Planned	Video encoding and CDN setup

Release Timeline

Milestone	Date	Description
Start of Development	[07/01/2021]	Development begins for the MVP release, starting with Sprint 1.
End of Sprint 1	[07/01/2021]	Completion of user registration, login, and basic subscription setup.
End of Sprint 2	[07/01/2021]	Completion of subscription management and initial content discovery.
End of Sprint 3	[07/01/2021]	Completion of e-book reader and video player features.
QA/UAT Testing	[07/01/2021]	QA and User Acceptance Testing for all completed features.
Final Review and Bug Fixes	[07/01/2021]	Final review, addressing critical bugs identified during testing.
MVP Release to Beta Users	[07/01/2021]	Release of <i>Fairweather v1.0</i> to a select group of beta users.
Public Release	[07/01/2021]	Full public release of <i>Fairweather v1.0</i> on web and mobile.

Dependencies

Dependency	Description	Impact if Delayed	Mitigation Plan
OAuth Integration	Required for secure login and social media account authentication.	User authentication may be delayed, affecting overall release.	Engage vendor early for API support and testing.
Payment Gateway Integration	Necessary for subscription purchase and payment processing.	Subscription management will be incomplete, affecting monetization.	Prioritize payment gateway setup early in Sprint 2.
CDN Setup	Needed for video streaming and e-book content delivery.	Content playback may be slower or delayed for users.	Use a backup CDN provider for initial release if needed.
UAT Access	Beta users must have early access for effective feedback.	Lack of feedback could delay feature refinement.	Grant early access to a small group of testers by end of Sprint 2.

Risks

Risk	Description	Impact	Mitigation Strategy
Integration Delays	Delays in third-party API integration (OAuth, payment gateway).	May push release date forward, impacting the MVP launch.	Establish strong communication channels with third-party vendors.
Limited UAT Coverage	UAT feedback may be limited due to the small beta user group.	Some issues may go undetected until public release.	Increase UAT focus in the final sprint and extend beta testing.
Performance Issues	Potential slow response times during peak usage, especially for content streaming.	May impact user experience and retention.	Implement load testing and optimize CDN setup before release.

2. Release Notes

The **Release Notes** provide an overview of the features, improvements, and fixes delivered in the current release of the *Fairweather* platform, along with known issues that need attention.

Fairweather v1.0 Release Notes

Release Overview

- **Release Name:** *Fairweather v1.0 – MVP Release*
- **Release Date:** [07/01/2021]
- **Type of Release:** Major Release (Minimum Viable Product)
- **Platform:** Web and Mobile
- **Release Status:** Deployed to Production

New Features

The following new features are included in *Fairweather v1.0*:

Feature	Description
User Registration	Users can now register using email, Facebook, or Google accounts, with a secure verification process.
Secure Login	Users can log in securely with MFA support, enhancing account protection.
Password Reset	Users can reset passwords via a secure email process to regain access to their accounts.
Subscription Plans	Users can view, select, and purchase subscription plans with integrated payment gateway support.
Content Discovery	Users can search for content, filter results, and receive recommendations based on preferences.
E-Book Reader	Basic e-book reading functionality with adjustable font size and background color options.
Video Player	Users can watch video content with subtitles, playback quality options, and full-screen mode.

Improvements

- **User Interface Enhancements:**
Improved navigation flow, better button responsiveness, and updated visuals for the registration form.
- **Error Handling:**
Improved error messages for login failures, including invalid credentials and MFA issues.

Bug Fixes

Bug ID	Issue Description	Resolution
BUG-01	Slow response time during registration process.	Optimized backend API calls to improve response speed.
BUG-02	MFA setup failing for some users.	Adjusted MFA integration to handle edge cases and device compatibility issues.
BUG-03	Payment gateway not processing certain card types.	Updated payment processing logic to accept additional card types supported by the gateway.

Known Issues

The following issues are known in the *Fairweather v1.0* release:

Issue ID	Description	Impact	Workaround
ISSUE-01	Slow content load times during peak hours.	May cause slight delays in content playback.	Users are advised to retry after a few seconds.
ISSUE-02	Limited LinkedIn login integration.	LinkedIn login is currently unavailable.	Users should use email, Facebook, or Google login.
ISSUE-03	Incomplete onboarding guidance for new users.	Some users may find the registration process unclear.	Tooltips will be added in the next release.

Important Notes

- **User Feedback:** We encourage users to provide feedback through the app's feedback feature or by contacting customer support.
- **Security:** Users are advised to enable MFA for enhanced account security.
- **Next Release:** The next planned release will focus on performance optimization, additional social login options, and improved onboarding guidance.

Continuous Improvement Documentation

1. Lessons Learned Documentation

The **Lessons Learned Documentation** captures insights and knowledge gained throughout the project, particularly after major milestones or releases. This document helps the team identify successes, understand challenges, and refine future processes.

Lessons Learned Overview

- **Project:** *Fairweather* Digital Library Platform
- **Milestone:** MVP Release (*Fairweather v1.0*)
- **Facilitator:** Scrum Master [Timi Ogunjobi]
- **Participants:**
 - Product Owner
 - Development Team
 - UX/UI Designer
 - QA Engineer
 - DevOps Engineer
- **Date:** [07/01/2021]

Structure of Lessons Learned

The lessons learned are categorized into three areas: **What Went Well**, **What Didn't Go Well**, and **Recommendations for Improvement**. The team's feedback was gathered through a workshop facilitated by the Scrum Master [Timi Ogunjobi].

What Went Well

Positive Aspect	Description	Impact
Effective Sprint Planning	Sprint planning sessions were productive, resulting in clear scope definition and achievable goals.	Contributed to a 100% user story completion rate for Sprint 1.
Collaboration Among Team	The team showed strong collaboration, openly sharing information and supporting each other.	Improved task progress and impediment resolution throughout the sprint.
Successful CI/CD Implementation	CI/CD pipeline setup facilitated frequent builds and testing, reducing integration issues.	Allowed faster feedback loops and ensured high-quality deliverables.
Stakeholder Engagement	Regular involvement of stakeholders in sprint reviews and backlog refinement led to useful	Enhanced product alignment with user needs and business goals.

Positive Aspect	Description	Impact
	feedback.	

What Didn't Go Well

Challenges	Description	Impact
Delayed API Documentation	Late receipt of third-party API documentation affected OAuth integration and testing.	Caused minor delays in authentication-related tasks.
Limited Testing Environment Access	QA team experienced delays in accessing the testing environment, affecting early bug identification.	Slowed down the overall QA process in Sprints 1 and 2.
Ambiguous User Requirements	Initial user stories for onboarding guidance lacked sufficient detail, causing rework.	Resulted in additional refinement during Sprint 2.

Recommendations for Improvement

Area for Improvement	Proposed Solution	Action Items	Responsible
API Documentation Delivery	Establish a Service Level Agreement (SLA) with third-party vendors for timely documentation delivery.	Product Owner to coordinate with vendors and clarify documentation timelines.	Product Owner, Backend Team
Early QA Involvement	Grant QA team early access to testing environments to ensure faster issue identification.	DevOps to provide earlier access during code completion.	DevOps Engineer, QA Engineer
User Story Refinement	Improve user story refinement by including more details in acceptance criteria and onboarding needs.	Product Owner to work with stakeholders to refine stories in backlog grooming.	Product Owner

Overall Summary

- **General Sentiment:** Positive, with good team collaboration and process adherence.
- **Key Lessons:** Early communication with third-party vendors is essential, and QA involvement should be maximized early in each sprint.
- **Action Items:** Implementation of the proposed solutions is set for the next sprint cycle.

2. Agile Coaching Logs

The **Agile Coaching Logs** document notes from coaching sessions, workshops, or training activities conducted with the team to improve Agile processes, communication, and collaboration.

Agile Coaching Session Overview

- **Project:** *Fairweather* Digital Library Platform
- **Coaching Session Title:** Sprint Retrospective Workshop
- **Facilitator:** Scrum Master [Timi Ogunjobi] (also acting as Agile Coach)
- **Participants:** Development Team, Product Owner, QA Engineer
- **Date:** [07/01/2021]
- **Session Type:** Workshop (Online)

Agile Coaching Session Objectives

- Enhance the team's understanding of the sprint retrospective process.
- Improve the team's ability to identify and implement process improvements.
- Foster a culture of continuous feedback and adaptation.

Coaching Session Agenda

1. Introduction:

Overview of the retrospective process and its importance in the Agile methodology.

2. Exercise – Start, Stop, Continue:

Team members listed activities they would like to start doing, stop doing, or continue doing during sprints.

3. Group Discussion:

Discuss insights from the exercise and brainstorm actionable steps.

4. Review of Previous Sprint Improvements:

Discuss whether the previous sprint's improvements were implemented effectively and what impact they had.

5. Feedback Session:

Gather feedback from team members on the effectiveness of the coaching session and how it

helped them in understanding Agile practices better.

Key Outcomes of the Session

Topic Discussed	Details	Agreed Actions	Responsible
Start: More Frequent Refinement	Team members agreed to conduct backlog refinement sessions mid-sprint to ensure clarity of user stories.	Schedule mid-sprint refinement sessions starting from Sprint 3.	Product Owner, Scrum Master [Timi Ogunjobi]
Stop: Long Planning Meetings	Some planning meetings were considered too long, affecting team energy levels.	Keep sprint planning meetings under 90 minutes.	Scrum Master [Timi Ogunjobi]
Continue: Daily Standups	The team agreed that daily standups were effective for communication and alignment.	Maintain current daily standup format and timing.	Scrum Master [Timi Ogunjobi]

Coaching Feedback

- **General Feedback:** Team members found the session useful, especially the “Start, Stop, Continue” exercise, which provided clarity on next steps.
- **Suggestions for Next Session:**
Focus more on improving estimation techniques (e.g., using Planning Poker) to enhance sprint planning accuracy.

Follow-up Actions

- **Mid-Sprint Refinement:**
Product Owner to coordinate with the team to start mid-sprint refinement sessions for better clarity on user stories.
- **Shorter Planning Meetings:**
Scrum Master [Timi Ogunjobi] to ensure future planning meetings are more concise by keeping the agenda focused and time-boxed.

Miscellaneous Documentation

1. Meeting Agendas and Minutes

Meeting Agenda Template

This template can be used for different Scrum ceremonies like **Sprint Planning**, **Sprint Review**, and **Sprint Retrospective**.

Meeting Agenda for Sprint Planning

- **Meeting Title:** Sprint Planning – Sprint 2
- **Date & Time:** [Will Communicate]
- **Duration:** 90 minutes
- **Facilitator:** Scrum Master [Timi Ogunjobi]
- **Participants:** Product Owner, Development Team, QA Engineer, DevOps Engineer

Agenda

1. **Introduction and Sprint Goal Setting** (10 minutes)
 - Review the overarching goal for Sprint 2.
 - Product Owner outlines the priorities for the sprint.
2. **Backlog Review** (20 minutes)
 - Product Owner presents the top-priority user stories from the backlog.
 - Team members discuss story details, requirements, and acceptance criteria.
3. **Estimation of User Stories** (25 minutes)
 - Team estimates story points for each user story using Planning Poker.
4. **Task Breakdown** (20 minutes)
 - Team breaks down selected user stories into smaller tasks.
 - Tasks are assigned based on team capacity.
5. **Sprint Plan Review** (10 minutes)
 - Review the sprint board setup.
 - Finalize sprint capacity and adjust as needed.

Meeting Minutes for Sprint Planning – Sprint 2

- **Meeting Date:** [07/01/2021]
- **Facilitator:** Scrum Master [Timi Ogunjobi]
- **Participants:** [Appended]

Discussion Summary

1. Sprint Goal:

- The goal of Sprint 2 is to complete the core subscription management features, including payment gateway integration.

2. Selected User Stories:

- US-05: View subscription plans (3 points)
- US-06: Purchase subscription (5 points)
- US-07: Manage subscription (4 points)

3. Estimations:

- User stories were estimated using Planning Poker.
- The team agreed on the story points for each story.

4. Task Assignments:

- Backend Devs: API development for subscription management.
- Frontend Devs: UI for subscription plans and management.
- QA: Prepare test cases for subscription features.

5. Decisions Made:

- All selected stories are within capacity, with a 10% buffer for unexpected issues.

Action Items

- [Backend Developer 1]: Start working on subscription API setup.
- [Frontend Developer 1]: Begin UI design for the subscription page.
- [QA Engineer]: Create test cases for subscription workflows.

Meeting Agenda Template for Sprint Review

- **Meeting Title:** Sprint Review – Sprint 1
- **Date & Time:** [Will Communicate]
- **Duration:** 60 minutes
- **Facilitator:** Scrum Master [Timi Ogunjobi]
- **Participants:** Product Owner, Development Team, Stakeholders

Agenda

1. **Introduction and Sprint Overview** (10 minutes)
 - Scrum Master [Timi Ogunjobi] provides an overview of completed sprint goals and user stories.
2. **Feature Demonstration** (25 minutes)
 - Team demonstrates completed features to stakeholders.
3. **Feedback Session** (15 minutes)
 - Stakeholders provide feedback on features and user stories.
4. **Product Backlog Updates** (10 minutes)
 - Product Owner discusses potential backlog adjustments based on feedback.

Meeting Minutes for Sprint Review – Sprint 1

- **Sprint Achievements:** All user stories related to user registration and authentication were completed.
- **Feedback:** Stakeholders suggested improving error messages and adding user onboarding guidance.
- **Next Steps:** Add new stories to refine onboarding guidance and error messages in the Product Backlog.

2. Team Charter and Working Agreements

The **Team Charter** establishes a shared understanding of the team’s purpose, roles, and collaboration norms.

Team Charter Overview

- **Project:** *Fairweather* Digital Library Platform
- **Purpose:** Build a subscription-based digital platform that delivers books and video content to users.
- **Vision:** Provide seamless access to digital content with a focus on personalization, security, and user engagement.
- **Team Values:** Transparency, collaboration, continuous improvement, accountability, and respect.

Team Roles & Responsibilities

Role	Responsibilities	Team Member(s)
Product Owner	Defines the product vision, manages the backlog, and prioritizes user stories.	[Insert Name]
Scrum Master [Timi Ogunjobi]	Facilitates Agile processes, removes impediments, and ensures team alignment.	[Insert Name]
Frontend Developers	Develops user-facing interfaces for web and mobile apps.	[Insert Names]
Backend Developers	Develops APIs, manages databases, and ensures backend functionality.	[Insert Names]
QA Engineer	Ensures product quality through testing and validation.	[Insert Name]
DevOps Engineer	Manages deployment, CI/CD pipelines, and infrastructure stability.	[Insert Name]

Working Agreements

Category	Agreement
Daily Standups	All team members must attend daily standups at 9 AM, keeping updates concise (2 minutes per person).
Sprint Planning	All team members must participate in sprint planning meetings, prepared with story estimates.
Code Reviews	Every pull request must undergo a code review by at least one other developer before merging.
Communication	Use Slack for quick communication and Jira for task updates; meetings should be kept within set time frames.
Decision-Making	Decisions are made collaboratively, with the Product Owner having the final say on product features.
Retrospective Feedback	All team members should provide constructive feedback during retrospectives to improve processes.

Conflict Resolution

- Conflicts should be resolved openly and quickly, with the Scrum Master [Timi Ogunjobi] mediating when necessary.
- If a conflict cannot be resolved within the team, the Scrum Master [Timi Ogunjobi] will escalate it to the Product Owner.

3. Team Roster and Contact Information

The **Team Roster** helps ensure effective communication among team members.

Team Roster

Name	Role	Email	Phone	Time Zone
[Name 1]	Product Owner	some1@ourmail.com	[+1 (555) 1234]	EST
[Name 2]	Scrum Master [Timi	some1@ourmail.com	[+1 (555) 5678]	EST
	Ogunjobi]			
[Name 3]	Frontend Developer	some1@ourmail.com	[+1 (555) 9101]	PST
[Name 4]	Frontend Developer	some1@ourmail.com	[+1 (555) 1123]	PST
[Name 5]	Backend Developer	some1@ourmail.com	[+1 (555) 3145]	EST
[Name 6]	Backend Developer	some1@ourmail.com	[+1 (555) 5167]	EST
[Name 7]	QA Engineer	some1@ourmail.com	[+1 (555) 7189]	EST
[Name 8]	DevOps Engineer	some1@ourmail.com	[+1 (555) 9210]	CST

Emergency Contact Protocol

In case of urgent communication needs (e.g., production issues or major blockers):

- **First Contact:** Scrum Master [Timi Ogunjobi]
- **Backup Contact:** Product Owner
- Use phone calls or direct messages on Slack for immediate reach.