

Project Plan

Gamehub

Date	:	26th of February 2024
Version	:	1.1
State	:	In Progress
Author	:	Cătălin Popoiu

Version history

Version	Date	Author(s)	Changes	State
1.0	21.02.2024	Cătălin Popoiu	Filled in most sections, such as Project Assignment and Risk	In Progress
1.1	26.02.2024	Cătălin Popoiu	Filled in missing parts, implemented most feedback received	In Progress

Contents

1. Project assignment.....	3
1.1 Context.....	3
1.2 Goal of the project.....	3
1.3 Scope and preconditions	3
1.4 Strategy	3
1.5 End products.....	4
2. Activities and time plan	5
2.1 Phases of the project	5
2.2 Time plan and milestones.....	6
3. Testing strategy and configuration management	7
4. Finances and risk.....	7
4.1 Risk and mitigation	8

1. Project assignment

1.1 Context

My project will be developed as a comprehensive platform intended to serve the global gaming community, connecting gamers, game developers, and enthusiasts in a dynamic social networking environment.

1.2 Goal of the project

The goal of the project is to create a unified platform where gamers can explore, share, and discuss their gaming experiences, discover new games, and connect with the community. It aims to enhance the gaming experience by providing tools for communication, event organization, and game discovery.

1.3 Scope and preconditions

Inside scope:	Outside scope:
1 Secure user authentication and customizable profiles	1 Integration with gaming consoles' proprietary networks
2 A comprehensive game database for discovery and exploration	2 Direct game distribution or sales
3 Community forums for discussions and sharing	3 In-depth analytics for game developers in the initial launch phase
4 Event and tournament organization features	4 Advanced AI-driven personalized game recommendations in the initial launch phase
5 Developer portal for showcasing games	5 Mobile application development
6 Real-time messaging and notifications	
7 Review and ratings system for games	

Preconditions: The project will leverage React for the frontend, Java Script Boot for the backend, MySQL for the database, and other technologies such as Hibernate, JPA, and WebSocket for real-time communication.

1.4 Strategy

My project will adopt an Agile and Scrum methodology, structured around six sprints to ensure iterative development and continuous feedback integration. This strategy supports flexible project management and allows for adjustments based on user feedback and testing results.

1.5 End products

1.Front-end Deliverables

- **User Authentication and Profile Management:** Implementation of secure login systems and customizable user profiles within the React-based front-end application.
- **Community Forums:** A dynamic platform for discussions and sharing amongst users, featuring real-time updates and interactive elements.
- **Event Management System:** User interface for organizing and participating in gaming events and tournaments, with tools for event creation, registration, and management.
- **Developer Portal:** A dedicated section for developers to showcase their games, receive user feedback, and interact with the community through a user-friendly interface.
- **Real-time Messaging System:** Implementation of real-time user interactions and notifications using WebSockets to facilitate immediate communication and updates.
- **Reviews and Ratings:** User interface allowing users to review and rate games, contributing to the community's understanding and evaluation of gaming content.

2.Back-end Deliverables

- **Game Database:** A comprehensive and searchable database of games, managed by a Java Spring Boot RESTful API, allowing for efficient data retrieval and management.
- **Secure Authentication System:** Back-end logic for handling secure JWT-based authentication and authorization processes, ensuring data protection and privacy.
- **Event and Tournament Organization Features:** Server-side logic to support the creation, management, and participation in gaming events and tournaments.
- **Developer Portal Backend:** Services and APIs designed to manage developer interactions, game submissions, and community feedback.
- **Messaging and Notifications System:** Backend implementation for supporting real-time messaging and notifications, ensuring timely communication across the platform.

3.Documentation Deliverables

- **Comprehensive Project Documentation:** Including the project plan, technical specifications, and user documentation, detailing the architecture, design decisions, and functionalities of GameHub.
- **Test Reports and Quality Assurance Documentation:** Documentation of testing strategies, methodologies, results, and quality assurance measures taken throughout the development process, ensuring reliability and performance of the platform.

2. Activities and time plan

2.1 Phases of the project

Sprint 1: Initial Setup and MVP Foundation

- **Objective:** Establish the project groundwork, including Agile workflow, and begin development on critical MVP features.
- **Activities:** Assemble the Agile team, establish the product backlog, set up the development and staging environments, outline the technological stack and architectural design, design and implement secure login/logout functionality, basic profile management, and a rudimentary interface for the game database.
- **Deliverables:** Project backlog, team composition, development environment setup, basic authentication system, user profile management, initial game database structure.

Sprint 2: Core Features Development

- **Objective:** Develop core functionalities that enhance user interaction and content discovery.
- **Activities:** Develop community forums, start integrating real-time messaging, and lay the foundation for game discovery features.
- **Deliverables:** Functional community forums, initial messaging system setup, enhanced game discovery mechanisms.

Sprint 3: Engagement and Interaction

- **Objective:** Enhance features that promote user engagement and interaction within the platform.
- **Activities:** Enhance the messaging system with group chat capabilities, implement event and tournament organization features, and introduce game reviews and ratings.
- **Deliverables:** Advanced messaging features, event management functionality, game review and rating system.

Sprint 4: Developer Portal and Advanced Features

- **Objective:** Launch a developer portal and begin work on advanced features based on earlier feedback.
- **Activities:** Set up a portal for game developers, start on advanced search and recommendation algorithms, and begin integrating third-party APIs for game information.
- **Deliverables:** Developer portal, initial search and recommendation engine, third-party game data integration.

Sprint 5: Refinement and Comprehensive Testing

- **Objective:** Refine features based on user feedback and conduct comprehensive platform testing.
- **Activities:** UI/UX improvements, security enhancements, comprehensive testing (unit, integration, and user acceptance tests), and bug fixing.
- **Deliverables:** Refined UI/UX, security updates, comprehensive test reports, bug fix documentation.

Sprint 6: Finalization and Launch Preparation

- **Objective:** Prepare the product for launch, focusing on final touches, performance optimization, and launch strategy.
- **Activities:** Final UI/UX enhancements, performance tuning, creation of marketing materials, and final preparations for launch.
- **Deliverables:** Launch-ready platform, marketing materials, detailed launch plan.

2.2 Time plan and milestones

The project will be executed over six sprints, utilizing the Agile and Scrum framework to ensure flexibility and responsiveness to feedback. Each sprint will last three weeks, with specific objectives and deliverables.

Sprint justification: The choice of three-week sprints is designed to balance the need for rapid progress with enough time for meaningful feature development, testing, and review. This timeframe allows for continuous integration of feedback and iterative improvement without compromising on quality or scope.

Sprint activities:

- **Sprint planning:** At the beginning of each sprint, I will identify tasks, choose the current focus of the project and set sprint goals.
- **Sprint demo:** At the end of each sprint, a sprint demo will take place to showcase the completed features and improvements to the stakeholders. This allows the reception of feedback and validation of work.
- **Sprint review:** After the sprint demo, a sprint review meeting will be held. During this session, stakeholders and I will discuss the sprint's achievements, review the progress increment and gather feedback for future iterations.
- **Sprint retrospective:** After the sprint review, I will hold a sprint retrospective to reflect on the sprint's performance. This includes identifying areas for improvement, adjusting processes for the next sprint and discussing what went well.

3. Testing strategy and configuration management

Unit Testing

- **Objective:** Ensure individual components function correctly in isolation.
- **Activities:** Develop and execute unit tests for both backend and frontend components to validate logic, functions, and classes.
- **Tools & Technologies:** Utilize JUnit for backend testing and React Testing Library for frontend components.

Integration Testing

- **Objective:** Verify that different modules or services work together as expected.
- **Activities:** Conduct integration tests to ensure that the application's modules interact correctly with each other and with external systems (e.g., databases, third-party APIs).
- **Tools & Technologies:** Use Spring Boot Test for backend integration testing and Cypress or Selenium for end-to-end frontend testing.

System Testing

- **Objective:** Confirm the complete and integrated software product meets specified requirements.
- **Activities:** Perform system testing to validate the application's overall performance, security, and functionality.
- **Tools & Technologies:** Employ tools like JMeter for performance testing and OWASP ZAP for security testing.

User Acceptance Testing (UAT)

- **Objective:** Ensure the software meets user needs and requirements.
- **Activities:** Conduct UAT with target users to validate the user experience, feature set, and usability of the application.
- **Tools & Technologies:** Feedback tools and manual testing based on user feedback scenarios.

4. Risk

4.1.Risk and mitigation

Risk	Prevention activities	Mitigation activities
1 Technical challenges with new technologies	Assess project complexity and adjust it to personal skills	Allocate time for research and potential training
2 Overestimating development capacity	Assess time management and spend time accordingly on development	Prioritize core features and allow for scope adjustment
3 Bugs and issues during deployment	Testing throughout the whole development phases	Keep an older stable version to rollback to in case it is needed
4 Data security breaches	Implement strong encryption methods	Implement a measure that alerts users in case of security breaches and has them change their password