Kontroller

Vision Document

Version <3.0>

Revision History

| **Date** | **Version** | **Description** | **Author** |
| --- | --- | --- | --- |
| 02/06/2025 | 1.0 | Write detailed document on the vision for this project | Nguyen Gia Nghi |
| 24/07/2025 | 2.0 | Revised All Vision Document | All members |
| 25/07/2025 | 3.0 | Revised All Vision Document | All members |
|  |  |  |  |

Table of Contents

1. Introduction 4

2. Positioning 4

2.1 Problem Statement 4

2.2 Product Position Statement 4

3. User Descriptions 4

3.1 User Profiles 4

3.2 User Task and User Environment 5

3.3 Alternatives and Competition 5

4. Product Features 5

5. Non-Functional Requirements 5

Vision (Small Project)

# Introduction

The purpose of this document is to collect, analyze, and define high-level needs and features of the Kontroller

system. It focuses on the capabilities needed by gamers and reviewers, and why these needs exist. The details of

how the Kontroller platform fulfills these needs are detailed in the use-case and supplementary specifications.

This Vision document provides an overview of the platform's goals, target users, high-level features, and product

positioning within the competitive landscape of digital entertainment logging and social platforms.

# Positioning

## Problem Statement

| The problem of | keeping track of games played and finding personalized recommendations or reviews, giving game critics a space to voice their opinions. |
| --- | --- |
| affects | gamers, casual players, game reviewers, and social players of all sorts |
| the impact of which is | users struggle to remember or share their gaming experiences, receive quality suggestions, critics don’t have a platform to efficiently share their thoughts and reviews |
| a successful solution would be | a social platform that allows users to log their gameplay, game critics to rate/review games, create lists, and interact with the game community in a personalized environment. |

## 

## 

## 

## 

## 

## 

## Product Position Statement

| For | all kinds of gamers and game critics |
| --- | --- |
| Who | wants to rate, review, track the games they’ve played |
| The (product name) | Is Kontroller |
| That | allows gamers to reflect on the games they’ve played and their gaming journey |
| Unlike | Letterboxd, Backloggd, IGDB |
| Our product | can guarantee a rich social experience for gamers with curated features and exclusive interaction systems including lists, stats, and reviews. |

# User Descriptions

## User Profiles

## Persona 1: Casual Gamer – Nguyễn Văn An

* Name/User Type: Casual Gamer
* Avatar: Male university student wearing gaming headphones
* Perspective: Gaming is a way to relax after school, not too focused on achievements
* Role/Responsibilities: Logs games played on weekends, reads friends’ reviews
* Skills: Basic to intermediate computer skills
* Experience:
  + Computer: Uses laptop daily for online learning
  + Games: Plays Genshin Impact, FIFA, and some mobile games
* Usage Frequency: 2–3 times per week
* Age/Gender: 20 years old, male
* Culture/Interests: Interested in Japanese culture, anime, football
* Habits: Usually plays on Saturday nights, watches game reviews on YouTube
* Goals/Expectations: Wants weekly recaps, easy-to-use interface, smooth mobile UI
* Limitations: Limited time due to school, unfamiliar with complex apps
* Other Attributes: Doesn’t need many advanced features, prefers fast loading and simplicity

## Persona 2: Game Content Creator – Trần Thị Bích

* Name/User Type: Content Creator / Streamer
* Avatar: Female streamer with a full PC setup, webcam, and professional microphone
* Perspective: Wants to build a personal brand by sharing high-quality gaming content
* Role/Responsibilities: Writes reviews, shares experiences, highlights key moments in games
* Skills: Tech-savvy, skilled in video editing and digital content creation
* Experience:
  + Computer: Uses a high-performance PC for livestreaming and editing
  + Games: Plays RPGs, horror, and indie games on Steam
* Usage Frequency: Almost daily
* Age/Gender: 22 years old, female
* Culture/Interests: Passionate about content creation, cosplay, active in Discord communities
* Habits: Plays at night, frequently shares screenshots/logs on social media
* Goals/Expectations: Needs tagging system, stylish review tools, fast sharing features
* Limitations: Needs quick publishing — limited time for complicated steps
* Other Attributes: Prefers a professional UI, wants to embed YouTube or TikTok links

## User Task and Environment

### **Task 1: Create a Game Log**

* **Goal:** Users want to record a gaming session (e.g., played Genshin Impact for 2 hours, completed a main quest).
* **Performed by:** Casual gamers, hardcore gamers, and content creators
* **Environment:**
  + At home, using a laptop or phone
  + May also log quickly while out using a mobile-friendly layout
* **Frequency:**
  + Casual gamer: 2–3 times per week
  + Hardcore gamer: Daily, after each gaming session
  + Content creator: After each livestream or video session
* **Constraints:**
  + Must be logged in
  + Requires a stable internet connection
  + Should not lose unsaved data (auto-save or temporary local storage is needed)
* **Other Info:**
  + Draft feature recommended
  + Ability to attach screenshots, links, mood, or platform

### **Task 2: Write and Share a Game Review**

* **Goal:** Users want to share detailed impressions and ratings after completing a game
* **Performed by:** Primarily content creators or reflective gamers
* **Environment:** At home, using a desktop or laptop for writing
* **Frequency:** Infrequent — after finishing a game (every 1–2 weeks or months depending on the game)
* **Constraints:**
  + May require a minimum character count
  + Needs basic formatting options: bold text, insert image/video
  + System should suggest tags or game genres
* **Other Info:**
  + Allows star rating and user comments
  + Edits allowed after posting
  + Option to share on social media or export a “review card” as an image

### **Task 3: Tag Friends in a Gaming Session (Multiplayer Tagging)**

* **Goal:** Users want to log a session and tag friends who played with them (similar to tagging in a photo)
* **Performed by:** Users with friends or guildmates in the system
* **Environment:** Usually on a computer or phone, after a gaming session
* **Frequency:** Depends on game type — roughly 1–2 times per week for co-op or MMO games
* **Constraints:**
  + Can only tag connected friends
  + Must prevent spam or unrelated tagging (confirmation may be required)
* **Other Info:**
  + Can assign roles (e.g., Healer, Tank, Support)
  + May display on friends’ timelines
  + Suggest friends based on recent activity or previous sessions

## Alternatives and Competition

| Competitor | Strengths | Weaknesses |
| --- | --- | --- |
| Backlogged | Serve the basic needs of game logging and reviewing | Pretty simple, not a lot of advanced features |
| IGDB | Rich game database | Logging and reviewing games are possible but not the main purposes so lacks lots of features |
| Steam | Renowned game store, numerous features, good for tracking games | Not built as a social media app but rather as a game store so it doesn't allow interactions between players. |

# Product Features -> functional user requirements

## User Authentication and Authorization

* + Sign up, log in, log out securely.
  + Supports email/password and social login (e.g., Google, Facebook).
  + Forgot password and account recovery.

## Game Browsing and Searching

* + Browse game database fetched from IGDB.
  + Search by title, genre, platform, release year, etc.
  + View detailed information about a game.

## Game Logging and Diary

* + Add games to the user's diary with timestamps.
  + Write personal logs and notes for each entry.
  + Delete or edit diary logs.

## Review and Rating System

* + Write, update, or delete reviews.
  + Rate games on a 1–10 scale with optional text.
  + Like and comment on reviews.

## PlayLater and Watchlist Management

* + Add/remove games to/from PlayLater list.
  + Maintain a “Watchlist” of games to track.

## Social Features

* + Follow or unfollow other users.
  + View activity feeds (reviews, ratings, logs).
  + Block or ban users to control visibility.

## List Management

* + Create public or private custom game lists.
  + Add/remove games from lists.
  + Share lists and explore lists made by others.

## Friend Discovery and Popular Trends

* + See games popular among friends and the public.
  + View activity by followed users and general trends.

## User Profile Customization

* + Update avatar, bio, and favorite games.
  + Display recent activity, review stats, and profile metrics.

## Admin Panel

* + Moderate reported content.
  + View, ban, or manage user accounts.

# Non-Functional Requirements 🡪 non-functional user requiremens

## Performance

* The platform should load major pages (e.g., homepage, game detail) within **2 seconds**.
* Should support up to **10,000 concurrent users** with minimal latency.

**Scalability**

* The backend must be modular to allow horizontal scaling.
* Database indexing and caching strategies must support increasing traffic and content.

## Security

* Use HTTPS across the app.
* Passwords must be encrypted.
* Prevent XSS, CSRF, and SQL injection attacks.

## Maintainability

* Code must follow clean architecture and .NET coding standards.
* Modular services for features like Auth, Game API, Review, etc.
* Unit and integration tests must be written for all critical modules.

## Reliability and Availability

* The system should have 99.9% uptime.
* API endpoints must have timeout handling and retry strategies.
* Backend should have fail-safe logging for unexpected errors.

## Usability

* Modern, responsive UI compatible with all major browsers and screen sizes.
* Clear, consistent layout.
* User-friendly navigation and forms.

## Documentation Requirements

* Provide online user documentation and help sections.
* Developer documentation must cover API specs, database schema, and integration guidelines.
* README.md and setup instructions must be present in the repository.

## Technology Constraints

* **Backend:** .NET (C#)..
* **Frontend:** HTML, CSS, JS.
* **Database:** SQL.
* **External APIs:** IGDB for game metadata.