Table of Contents:

1. Functional Requirements
2. Non-Functional Requirements
3. Constraints
   1. Social Constraints
   2. Economic Constraints
   3. Technical Constraints
   4. Legal Constraints
   5. Usability Constraints
4. Solution Scope
5. User Characteristics
6. Technical Environments
7. C2 Analytical Tools – Appendix
   1. Use Case Diagram
   2. Data Flow Diagram
   3. Context Diagram

# Functional Requirements

|  |  |  |
| --- | --- | --- |
| Code (FR\*\*) | Functional Requirement | Comment |
| FR01 | Users must be able to **sign up** and **log in** securely. | Supports authentication via **JWT, OAuth, and social login** (Google, Facebook, X, etc.). |
| FR02 | Passwords must be **encrypted** before storage. | Enhances security and prevents unauthorized access. |
| FR03 | Users must be able to **create, modify, and delete study plans.** | Core functionality for organizing study sessions. |
| FR04 | Study planner must support **recurring sessions.** | Allows users to set up weekly/daily study routines. |
| FR05 | Users must be **matched based on predefined criteria.** | Matchmaking algorithm pairs users based on subjects, availability, and learning styles. |
| FR06 | Users must be able to **accept or decline study partners.** | Provides control over matchmaking choices. |
| FR07 | Chat messages must be **sent instantly** between study partners. | Uses **WebSockets (Socket.IO)** for real-time messaging. |
| FR08 | Study partner availability must be **updated dynamically.** | Ensures users can see when partners are online or available. |
| FR09 | Users can track **study hours and progress.** | Logged data generates productivity insights. |
| FR10 | Badges and streaks must be **awarded for achievements.** | Encourages consistent study habits through **gamification.** |
| FR11 | Users must be able to **subscribe to premium features (**Study Gold). | Unlocks advanced analytics and matchmaking tools via **Stripe, PayPal, or Square.** |
| FR12 | Transactions must be **securely processed** via third-party services. | The system **cannot store credit card details directly**—must comply with **PCI-DSS security.** |
| FR13 | Users should receive **email notifications** for key updates. | SendGrid integration enables reminders for study sessions and match updates. |
| FR14 | Privacy settings must allow users to **control matchmaking visibility.** | Users can decide whether they appear in matchmaking searches. |
| FR15 | User data must be **encrypted and stored securely.** | Ensures compliance with **GDPR and other privacy laws.** |
| FR16 | Users must be able to **delete their accounts** permanently. | All associated data should be erased upon request. |
| FR17 | System must track **user activity analytics.** | Uses Google Analytics API to monitor engagement trends. |
| FR18 | Subscription renewal and cancellation must be **clearly defined**. | Users should be able to manage payments and upgrades easily. |
| FR19 | UI must be **responsive and accessible** across devices. | Optimized for **mobile and desktop users** with adaptive design. |
| FR20 | External APIs must only send **necessary data.** | Prevents excessive data sharing with third-party services. |

# Non-Functional Requirements

|  |  |  |
| --- | --- | --- |
| Code (NFR\*\*) | Non-Functional Requirement | Comment |
| NFR01 | The system must be **available 99.9% of the time.** | Ensures minimal downtime for a smooth user experience. |
| NFR02 | Response time for API requests must be **under 200ms.** | Guarantees fast and efficient study matchmaking & planner updates. |
| NFR03 | All user data must be **encrypted** during transmission and storage. | Uses **AES encryption** for sensitive information. |
| NFR04 | The system must comply with **GDPR and other privacy laws.** | Protects user data and ensures legal compliance. |
| NFR05 | The UI must be **responsive and accessible** across devices. | Optimized for **mobile and desktop users** with adaptive design. |
| NFR06 | Users must be able to **navigate the app intuitively**. | Follows standard **UI/UX principles** for seamless usability. |
| NFR07 | The matchmaking algorithm must generate results in **less than 5 seconds.** | Ensures quick pairing without excessive processing delays. |
| NFR08 | Transactions must be **secure and compliant** with PCI-DSS standards. | Payment processing via Stripe, PayPal, and Square must be fully encrypted. |
| NFR9 | Notification system must have **less than 1-second latency.** | Push notifications must be instant for study reminders and updates. |
| NFR10 | The system must support **multi-language options.** | Enables accessibility for global users. |
| NFR11 | The system should allow for **at least 1GB of storage per user.** | Supports storing study logs, preferences, and communication history. |
| NFR12 | Logs and analytics must be processed **daily** for optimization. | Tracks user activity trends via Google Analytics API. |
| NFR13 | The system must integrate with **third-party APIs** efficiently. | Ensures smooth operation between SendGrid, analytics, and payment processing. |
| NFR14 | Error handling must provide **clear messages and recovery options.** | Users must understand failures and possible fixes without frustration. |

# Constraints

## Social Constraints

|  |  |  |
| --- | --- | --- |
| Code (SC\*\*) | Social Constraint | Comment |
| SC01 | Must support diverse learning styles. | Study plans and matchmaking must be adaptable for different types of learners. |
| SC02 | Needs features to encourage collaborative learning. | Users should feel comfortable studying with matched partners. |
| SC03 | Accessibility features must support disabled students. | Includes text-to-speech options and compatibility with assistive technologies. |
| SC04 | Must prevent social exclusion. | Matchmaking should avoid bias and ensure fair pairings. |
| SC05 | Should encourage healthy study habits. | Gamification should promote balance and discourage unhealthy overworking. |
| SC06 | Must respect cultural differences in learning approaches. | Users from various backgrounds may prefer different study methods. |
| SC07 | Requires community moderation tools. | Users should be able to report inappropriate behaviour. |

## Economic Constraints

|  |  |  |
| --- | --- | --- |
| Code (EC\*\*) | Economic Constraint | Comment |
| EC01 | Must be affordable for students. | Pricing models need to consider student budgets. |
| EC02 | Subscription model must be financially sustainable. | Revenue must cover hosting, security, and ongoing development costs. |
| EC03 | Needs low-cost marketing strategies. | Organic growth through social media and partnerships with educational institutions. |
| EC04 | Must minimize server and maintenance costs. | Cloud hosting should be reliable while being affordable |
| EC05 | Requires efficient payment processing. | Fees from third-party payment gateways (Stripe, PayPal) must be factored into revenue. |
| EC06 | Needs a free tier to attract users. | Offers basic features for free with an option to upgrade. |
| EC07 | Should support student discounts. | Making premium features affordable encourages adoption. |

## Technical Constraints

|  |  |  |
| --- | --- | --- |
| Code  (TC\*\*) | Technical Constraint | Comment |
| TC01 | Must handle large amounts of user data efficiently. | The database should scale with user growth without slowing down. |
| TC02 | Requires strong encryption for sensitive data. | Protects authentication and payment information. |
| TC03 | Matchmaking algorithm must operate within 10 seconds. | Long delays reduce engagement and user satisfaction. |
| TC04 | Real-time chat must work without noticeable latency. | Users expect instant messaging between study partners. |
| TC05 | Must support cloud hosting and CDN for scalability. | Ensures fast content delivery and minimal downtime. |
| TC06 | API requests must complete in less than 200ms. | Improves overall system responsiveness. |
| TC07 | Must integrate third-party APIs securely. | SendGrid for emails, Stripe/PayPal for payments, etc. |
| TC08 | System should support future AI-enhanced features. | The architecture should allow future upgrades like AI-driven study recommendations. |

## Legal Constraints

|  |  |  |
| --- | --- | --- |
| Code  (LC\*\*) | Legal Constraint | Comment |
| LC01 | Must comply with Australian Privacy Act 1988. | Ensures personal data is collected, stored, and used lawfully. |
| LC02 | Must follow Victorian Privacy Laws (Privacy and Data Protection Act 2014). | Regulates data handling by organizations operating in Victoria. |
| LC03 | Payment processing must adhere to PCI-DSS security standards. | Protects financial transactions using Stripe, PayPal, or Square. |
| LC04 | Must comply with Australian Consumer Law (ACL). | Ensures fair business practices, including subscription terms and refunds. |
| LC05 | Must follow Copyright Act 1968 when using study materials. | Ensures any shared resources are properly licensed or attributed. |
| LC06 | Requires Terms & Conditions agreement in compliance with Australian contract law. | Defines user responsibilities and legal protections for Studyr. |
| LC07 | Needs a Privacy Policy in line with Australian privacy principles (APPs). | Informs users about data collection, storage, and usage policies. |
| LC08 | Must comply with Child Safe Standards (Victoria) if under-18 users are involved. | Requires appropriate safeguards and parental consent where necessary. |
| LC09 | Must allow users to opt out of data sharing per Australian privacy regulations. | Protects user control over personal data used for analytics or third-party services. |
| LC10 | Must prevent harassment or inappropriate behaviour in messaging features. | Requires moderation tools and reporting systems for user safety. |

## Usability Constraints

|  |  |  |
| --- | --- | --- |
| Code | Usability Constraint | Comment |
| UC01 | Interface must be intuitive and beginner friendly. | Users should be able to navigate easily without prior technical knowledge. |
| UC02 | Must be responsive on different devices. | The app must work well on smartphones, tablets, and desktops. |
| UC03 | Should minimize clutter and complexity. | Simple UI design improves usability. |
| UC04 | Must load within a reasonable time (under 3 seconds). | Slow apps discourage user engagement. |
| UC05 | Needs clear error messages and guidance. | Users must understand how to fix common issues without frustration. |
| UC06 | Requires flexible customization. | Users should be able to adjust study plans and notification preferences. |

# Solution Scope

**Features Included in Scope**

* **Study Planning & Tracking**
  + Users can create, edit, and delete study schedules.
  + Supports recurring study sessions for long-term planning.
  + Provides visual analytics for study streaks, time management, and performance trends.
* **Study Partner Matchmaking**
  + Automated matchmaking based on subject, availability, and learning styles.
  + Secure in-app chat for study collaboration.
  + Compatibility scoring system to recommend ideal study partners.
* **Gamification & Engagement**
  + Streak tracking with rewards and leaderboard rankings.
  + Study challenges to boost motivation.
  + Achievement badges for consistency and productivity.
* **Notifications & Communication**
  + Real-time push notifications for study reminders.
  + Email alerts for new matches and upcoming study sessions.
  + Updates on study partner availability for scheduling adjustments.
* **Subscription Model (Study Gold)**
  + Premium analytics for study habits and performance.
  + Enhanced matchmaking features (priority-based pairing).
  + Advanced insights into study productivity trends.
* **Security & Compliance**
  + End-to-end encrypted messaging for study discussions.
  + Secure authentication (OAuth, JWT) for user accounts.
  + Compliance with Victorian privacy laws and Australian Consumer Law.

# User Characteristics

Primary Users: Students

The audience for Studyr consists of high school and university students who require structured study planning, productivity tracking, and collaboration. These users face challenges in managing coursework, staying motivated, and finding study partners with similar goals and schedules. They need an easy-to-use platform that helps them organize study sessions, set reminders, track progress, and efficiently match with compatible study partners based on shared subjects and availability.

Secondary Users: Self-Improvement Learners & Professionals

Beyond academic students, Studyr also serves self-learners and professionals studying for personal growth, certifications, or career advancement. These users typically prefer self-paced study methods rather than formal peer matchmaking. While they may not require collaborative tools, they value structured scheduling, productivity insights, and personalized reminders to maintain discipline in their independent learning.

Usability Needs & Accessibility

Studyr must ensure a user-friendly, intuitive interface that works across desktop and mobile platforms, allowing seamless study planning regardless of device preference. Accessibility features such as text-to-speech compatibility, and adjustable study settings help diverse users engage effectively. The platform caters to different study habits by offering customizable scheduling, engagement tracking, and motivational tools to keep users accountable.

Social & Engagement Preferences

Students and professionals alike benefit from gamification elements that make studying more engaging. Features like study streaks, badges, and leaderboard rankings help users stay consistent with their goals while fostering a sense of accomplishment. For those using matchmaking, peer interaction through chat and shared study sessions enhances learning effectiveness and builds a supportive study community.

Studyr is designed to adapt to different learning styles, whether structured or flexible, solo or collaborative. Its dynamic scheduling, personalized insights, and motivational tools position it as an essential platform for efficient and engaging study management.

# Technical Environment

## Front-End Development

**Technology:** React.js (JavaScript)

* **Purpose:** Provides an interactive and responsive UI across desktop and mobile.
* **Key Features:**
  + Component-based structure for modular development.
  + Uses Axios for efficient API communication with the back end.
  + Implements React Router for smooth navigation between study planner, matchmaking, and profile pages.
  + Material-UI or Tailwind CSS for sleek UI design.
* **Performance Considerations:**
  + Optimized for fast loading times and minimal re-rendering using React’s useEffect & useMemo hooks.

### Back-End Development

**Technology:** Node.js with Express.js

* **Purpose:** Manages authentication, data processing, and matchmaking logic.
* **Key Features:**
  + RESTful API endpoints for study planner, matchmaking, and user management.
  + Uses Middleware for logging, security, and request validation (e.g., Express Validator).
  + Implements rate limiting and caching to optimize API response times.
* **Performance Considerations:**
  + Utilizes asynchronous processing (async/await) for faster execution of API calls.

## Database Management

**Options:** PostgreSQL (Relational) or MongoDB (NoSQL)

* **Purpose:** Stores user profiles, study schedules, messaging history, and matchmaking preferences.
* **Key Features:**
  + PostgreSQL (Structured) → Ideal for relational data like user authentication and study session logs.
  + MongoDB (Flexible) → Best for managing unstructured data such as chat messages and user activity tracking.
  + Indexing strategies to boost database query performance.
* **Performance Considerations:**
  + Database sharding for distributing large amounts of data across multiple servers.
  + Uses ORM tools like Sequelize (PostgreSQL) or Mongoose (MongoDB) for easier data manipulation.

## Authentication & Security

**Technologies:** OAuth 2.0, JWT, HTTPS

* **Purpose:** Ensures secure login and protects user data.
* **Key Features:**
  + OAuth integration for Google, Facebook, and Microsoft authentication.
  + JWT-based session management to prevent unauthorized access.
  + Implements bcrypt encryption for storing passwords securely.
  + HTTPS enforcement for encrypted user communication.
* **Compliance:**
  + Fully adheres to Victorian privacy laws and Australian Consumer Law.
  + PCI-DSS compliant for secure payment transactions.

## Payment & Subscription Management

**Technologies:** Stripe, PayPal, Square

* **Purpose:** Enables secure transaction processing for Study Gold subscriptions.
* **Key Features:**
  + Secure payment gateway integration.
  + Automated subscription renewal and billing cycle management.
  + Refund and cancellation policies aligned with Australian Consumer Law.
* **Compliance:**
  + PCI-DSS standards for transaction security.

## Analytics & Monitoring

**Technologies:** Google Analytics, LogRocket, Prometheus

* **Purpose:** Tracks user activity, engagement trends, and system performance.
* **Key Features:**
  + Google Analytics monitors study session interactions and user retention.
  + LogRocket provides error tracking and debugging.
  + Prometheus tracks API performance and server load.
* **Performance Considerations:**
  + Automated alerts for unusual traffic or downtime.

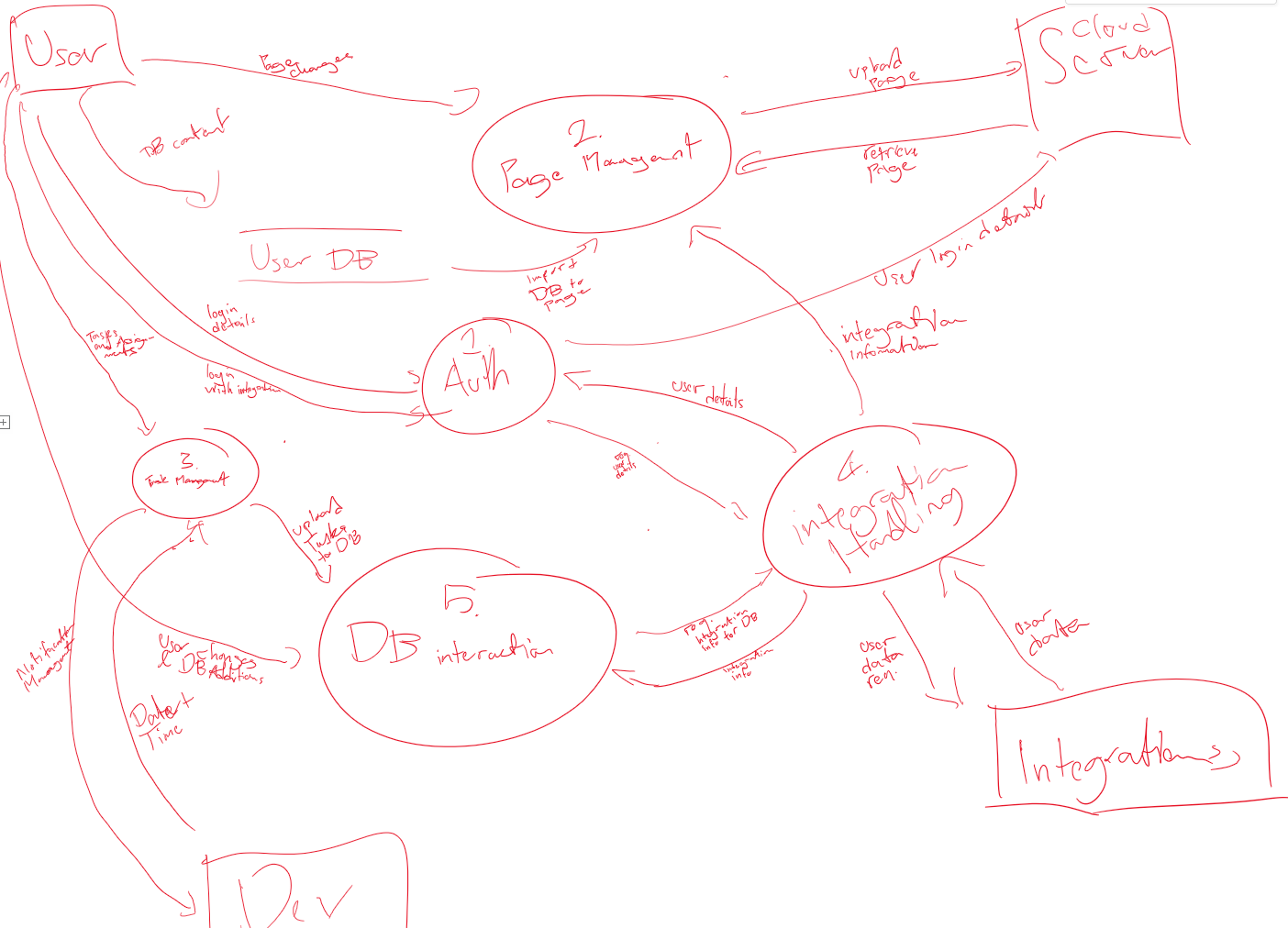
# C2 Analytical Tools – Appendix

## Context Diagram

A diagram of a solution

AI-generated content may be incorrect.

## Data Flow Diagram



## Use Case Diagram