Requirement Document

Alexandros Gonatas
alexthecipher@gmail.com

Alexander Spetz
spetzal@student.chalmers.se

Haoning Ma
mahaoning21@outlook.com

Xingrong Zong
xingrong@student.chalmers.se

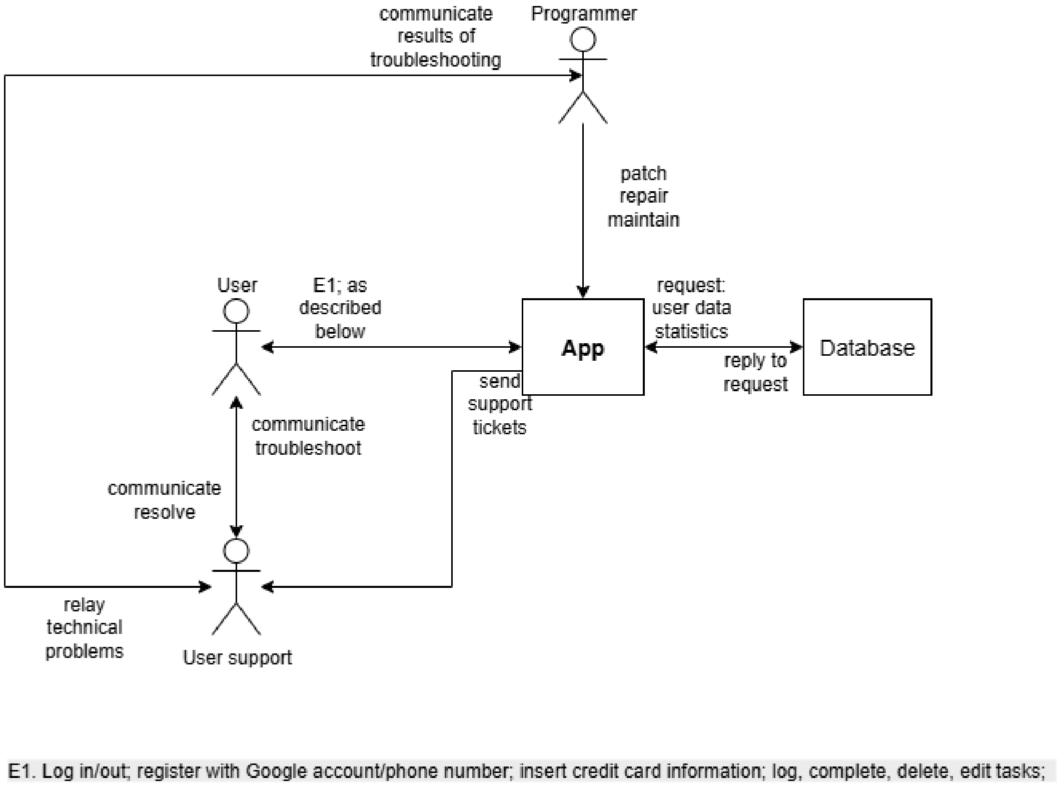
September 2025

Contents

1 High-level Description

1.1 Goal and scope

The goal of the application is to help users overcome procrastination and improve productivity by turning task management into an engaging game. The user should log their daily tasks in the application, and upon completion, they are granted rewards. These rewards can be spent on the game part of the application.



E1. Log in/out; register with Google account/phone number; insert credit card information; log, complete, delete, edit tasks;

Figure 1: Context Diagram

The context diagram (Figure 1) describes the relation of our system to other systems and agents who interact with it.

The **application** requests user data and statistics from the central database, an external system which contains this information. The **database** must reply appropriately to the system's request within a short time frame.

The **user** must register on the application with their e-mail, Google account or phone number. They can also log in/out. They can log tasks and perform these actions on them: create, edit, delete, view, complete. Once a task is completed, the user must receive rewards. Lastly, they can interact with the game environment provided by the app.

The application must ask the user for permissions: notifications, access to their calendar. The application must provide a contact center option, in case of errors or questions, which will generate a ticket for customer support agents.

The **customer support agents** must receive the tickets generated by the contact center. They must communicate with the users to ask for further information concerning their request and to update them about their ticket's status. Also, they can communicate with the programmers of the application in case of a technical error they are not able to resolve.

The **programmers** must maintain and repair the application. They can communicate with the customer support agents in case of a technical error and update them on the status of their query.

1.2 Business case and stakeholder map

Perspective	Illustration	Example
Problem/Issue	Real-life demands	People often set goals but lack self-discipline, existing apps do not provide encouragement by feedback, whether it is statistics or words of wisdom
Value	Opportunities are created by this project	Provides a cross-platform, data-driven discipline tool, increasing user loyalty
Investment gain	Financial gain and branding	Early stage ads and premium subscription; potentially collaborate with other applications and make a profit chain
Risks and restraints	Possible and potential challenges and solutions	How to differentiate from current existing market competitors and how we can attract more users

Nowadays, people use digital tools for all sorts of tasks: documents, diagrams, software, websites, etc. However, they often get distracted by either browsing or by the overwhelming, constant stream of information when trying to deal with a task. This leads to progress delay that often causes anxiety and procrastination for individuals, and lowers the efficiency of work for organizations.

There are many productivity apps already, but in reality, people still get frustrated when they are not completing tasks in time. Is it because those apps are not helpful? A simple answer is no, since each individual reacts and progresses differently, and apps are designed to serve diverse users.

With current technology trends, it will also be innovative to collaborate with Artificial Intelligence (AI) to, for example, collect user data to provide a customized, productive plan.

The expected benefits include:

- For users – improved concentration, higher task completion rate, reduced anxiety, and a fun, motivating way to cultivate long-term discipline.

- For stakeholders – stronger customer engagement, potential for workplace productivity programs, and positive impact on mental well-being.

From a business perspective, the project creates new revenue streams through premium subscriptions, in-app advertising, and B2B licensing to educational institutes or companies. It also enhances brand value by positioning us as an innovator in the digital productivity and wellness market.

Risks include intense competition, user-privacy concerns, and challenges in long-term engagement. These will be mitigated through clear market differentiation (AI-personalization + gamification), robust data-protection policies compliant with GDPR, and continuous user-experience testing to maintain high retention.

Name - Stakeholder name	Relationship - relation to system under construction	Representative - Name and contact info of one person	Sentiment / Power - How they feel about this project and whether they can influence it
End User	Daily use, provides feedback	Target student group representative	Positive / High
Investor	Provides funding	University course sponsor	Neutral / High
Developer Team	Builds and maintains	Group members	Positive / High
GDPR Authority	Legal compliance	N/A	Neutral / High

Stakeholder Map, Estimated result of future Elicitation

1.3 Core functionality

UML diagram (note: has to focus on the actions performed by the APP, not the actors):

- Actors: User, Notification Service
- Use cases:
  - Manage To-Dos
  - Receive Reminders
  - Earn Rewards
  - View Progress Statistics

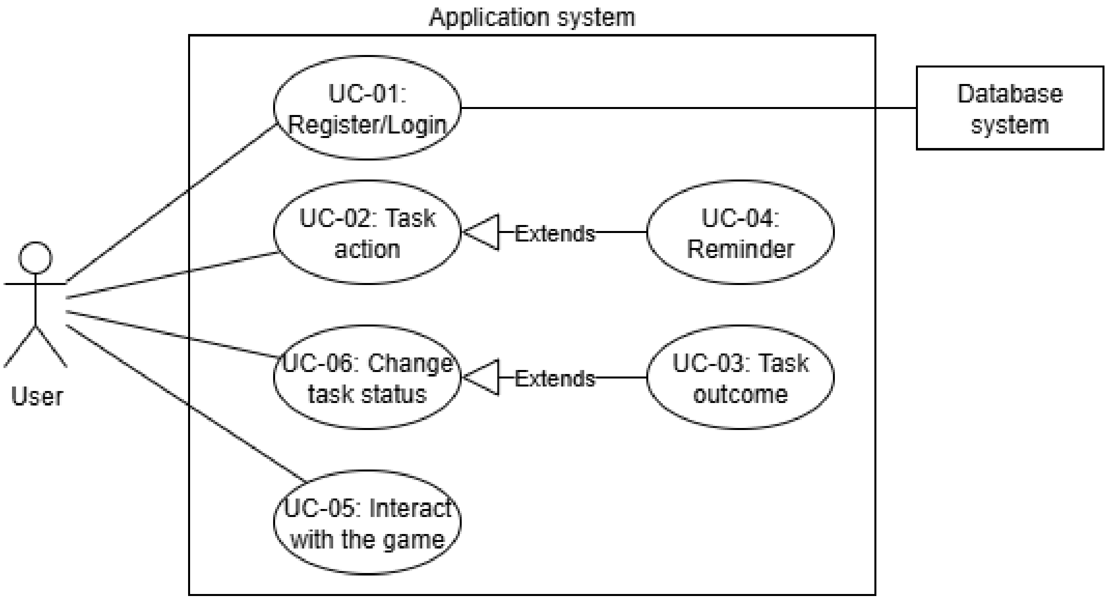


Figure 2: Use Case Diagram

UC-01: Register/Login. This initial action requires the user to either create an account or use their account to login the application. It is required as their data is unique and this is the optimal way to store them. The user's data is stored in the database, which means that the database system has to provide and verify the user's identity.

UC-02: Task action. The user can create, edit, delete, view or change the status of tasks.

UC-03: Task outcome. Each task has a reward attributed to it. If a user has missed the deadline on one of their tasks or completes one of their tasks, they will be met with missed or gained rewards.

UC-04: Reminder. The user receives a reminder for a task in the form of a mobile notification.

UC-05: Interact with the game. The user can user their rewards from UC3 to interact with the in-built game of the app. The game is yet to be defined.

UC-06: Change task status. The status of the task has changed. This triggers UC3.

ID	Use Case	Description
UC-01	Register/login	Adds user or logs in user
UC-02	Task action	User creates, deletes, edits or views a task
UC-03	Task outcome	A tasks either fails or succeeds
UC-04	Set Reminder	User sets a due date and notification
UC-05	Interact with game	User interacts with the game aspect somehow
UC-06	Change task status	User changes the status of task

Use-Case table

1.4 Performance Requirements, Specific Quality Requirements, Constraints

This part lists the main non-functional requirements for our goals.

User lifecycle:

- Activation: a new user finishes one task on day 1.
- Retention: users come back in week 1 and week 4.
- Conversion: some users buy premium.
- Trust: privacy, security, stable service.

Should we have the user lifecycle somewhere else? or not at all?

Quality targets (high level) (Estimated numbers, any feedback appreciated)

- QR-01 Task completion speed: p95 from tapping "Complete" to reward shown $\leq$ 500 ms online; $\leq$ 2s offline cache. Priority P0.
- QR-02 App start time: launch to interactive "Today" median 1.5 s, p95 2.5 s (worst 2.5 / 4). Priority P0.
- QR-03 Availability: monthly uptime for create/open/complete $\geq$ 99.5% (worst 99.0 %). Priority P0.
- QR-04 Notification on time: reminders delivered within 60 s $\geq$ 95 %. Priority P1.
- QR-05 Offline use: add/complete offline; sync after reconnect $\leq$ 10 s. Priority P1.
- QR-06 Privacy and GDPR: data export $\leq$ 48 h; account deletion $\leq$ 7 d. Priority P0.

- QR-07 Security baseline: TLS $\geq$ 1.2, encryption at rest, no secrets in client, pass baseline checks. Priority P0.

Constraints (high level):

- Platforms: iOS 15+, Android 10+, Web (latest 2 major browsers).
- Data: only what we need for core loop and subscription; analytics opt-in; no sensitive categories.
- Policies and payments: follow App Store/Play and GDPR; only subscription and cosmetic items (no cash out).

Prioritization:

- P0: QR-01, QR-02, QR-03, QR-06, QR-07.
- P1: QR-04, QR-05.

2 User Requirement Specification

2.1 Data requirements

In order to describe the data requirements of the application, an ER diagram is used to illustrate the relationships between entities that exist within the application. For this revision, the data requirements are kept at a high level, thus the diagram appears minimal. For that reason, a data dictionary was used to describe in detail the attributes of these entities.

2.1.1 ER-Diagram

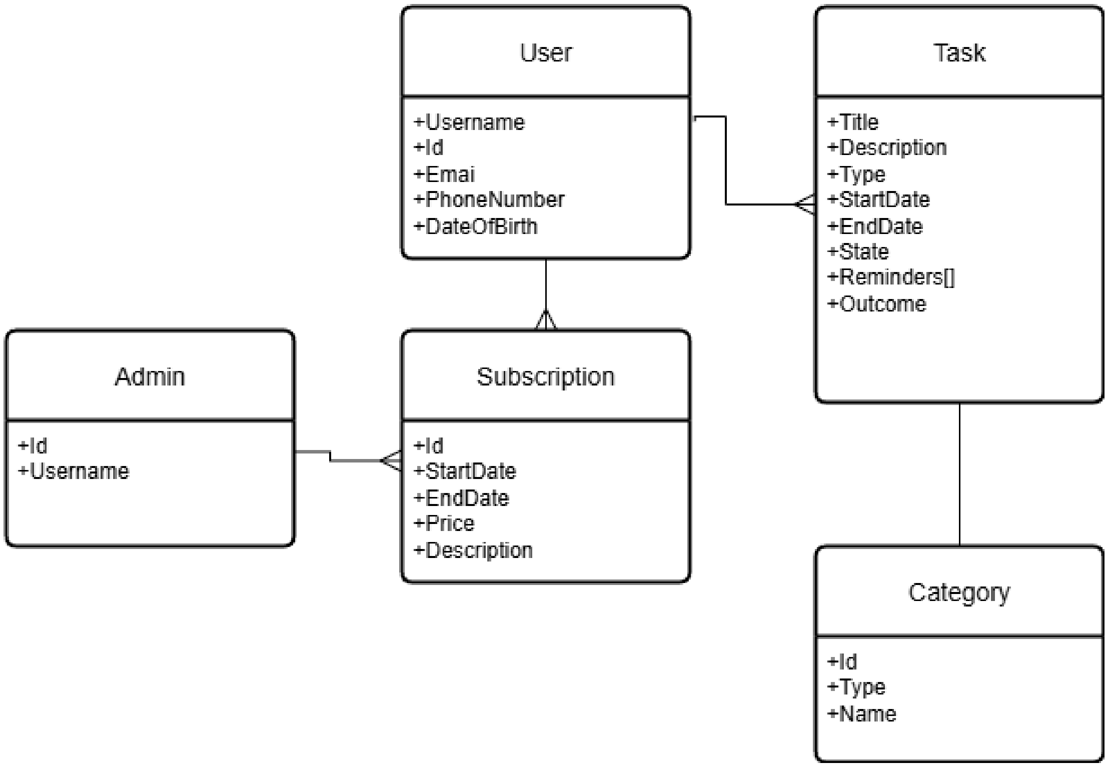


Figure 3: ER-diagram

2.1.2 Data dictionary

Class: User
The "User" class represents each user of the application. It holds the necessary logic regarding a user such as the username. The user class can also have a subscription as well as several tasks.

Attribute	Type	Description
Id	Int	A unique id that represents each user
Username	String	A username for the user
Phone number	Tel	A provided phone number by the user
E-mail	Email	A provided e-mail by the user
Date of birth	Date	The user's birth date

Class: Task

The ”Task” class represents a task in the application. The type of a task can be attributed to categories such as a study task or a sports task.

Attribute	Type	Description
Title	String	The title of the task
Description	String	A description of the task
State	Category	Contains the status of the task
Start date	DateTime	The start date and time
End date	DateTime	The end date and time
Reminder	Time	A user can set one or many reminders for a task
Type	Category	The type of the task
Outcome	Category	The outcome of the task

Class: Subscription
The ”Subscription” class represents the subscription plan a user can have.

Attribute	Type	Description
Id	Int	The unique id of the subscription plan
Price	Int	The cost of the subscription
Start date	DateTime	The date and time the user chose the subscription
End date	DateTime	The date and time the subscription expires for a user
Description	String	The description of the subscription plan

Class: Administrator
The ”Administrator” class represents the highest-level account responsible for the app. The Administrator provides the subscription plans.

Attribute	Type	Description
Id	Int	The id of the account
Username	String	The username of the account

Class: Category
The ”Category” class represents a general entity that is used to describe attributes, like the ’Outcome’ of a ’Task’. The ’Type’ is an enum that describes the different categories; for example, it separates ’Outcome’ categories from ’State’ categories. This attribute will be useful for showing the correct categories in their corresponding fields in the application.

Attribute	Type	Description
Id	Int	The unique id of the category
Type	ENUM	The type of the category
Name	String	The name of the category

2.2 Functional requirements

2.2.1 Task Creation and Management

User Story: As a user, I want to create, edit, and delete so that I can manage my workload effectively.

- Description: Users can add unlimited tasks, edit their details (title, description, category, deadline), and delete them when no longer needed.
- Stakeholders and Business Value:
  - Users: Essential for organizing personal work; reduced friction in task entry.
  - Investors/Owners: A basic requirement for user retention - without smooth task management, retention drops.
- Success Measure: Users report that entering tasks feels fast and intuitive.

2.2.2 Categories and Projects

User Story: As a user, I want to group tasks into categories/projects, so that I can organize complex goals.

- Description: Tasks can be tagged under categories (Work, Study, Health, etc). Filter and sort views allow quick navigation.
- Stakeholders and Business Value:
  - Users: Helps them manage multiple areas of tasks; reduces clutter.
  - Investors/Owners: Supports upselling advanced organization tools.
- Success Measure: Users create ≥ 1 category on average in their first week.

2.2.3 Deadlines and Scheduling

User Story: As a user, I want to set deadlines and recurring tasks so that I can plan my time better.

- Description: Users can assign due dates/times, create repeating tasks (daily, weekly, custom).
- Stakeholders and Business Value:
  - Users: Increases usefulness beyond simple lists; supports productivity habits.
  - Project Owners: Improves retention.
- Success Measure: % of users who schedule tasks grows monthly.

2.2.4 Notifications and Reminders

User Story: As a user, I want reminders for tasks so that I don't forget what I planned.

- Description: Push notifications, in-app banners for approaching deadlines or streak loss.
- Stakeholders and Business Value:
  - Users: Improves success rate in completing tasks.
  - Investors/Owners: Increases daily active use and reduces churn.
- Success Measure: Number of tasks completed on time increases after enabling reminders.

2.2.5 Reward Allocation

User Story: As a user, I want to receive points for completing tasks so that I stay motivated.

- Description: Completing tasks grants points or virtual rewards.
- Stakeholders and Business Value:
  - Users: Keeps engagement high and gamifies productivity.
  - Investors/Owners: Encourages premium upgrades to raise the daily rewards.
- Success Measure: $\geq$ 30 % of active users engage with the reward system weekly.

2.2.6 Reward spending

User Story: As a user, I want to spend the points in something interesting, because I want to feel like my rewards matter and have impact.

- Description: Using rewards to "buy" items that benefit an in-app game.
- Stakeholders and Business value:
  - Users: Causes excitement, sets personal goals, increases motivation for completing tasks.
  - Investors: Differentiates app from existing task management systems, creates brand/name value.
- Success Measure: New demands for the game per month, high-normal rate users spend their rewards.

2.2.7 Progress Tracking and Badges

User Story: As a user, I want to keep track of my progress, because I want to know if I have progressed and what I failed to do, in order to draw motivation or determination for the next days.

- Description: Acquiring an X-day streak, getting badges for completing certain tasks and
- Stakeholders and Business Value:
  - Users: Motivation, clear picture of progress, measurable success/failure.
  - Investors/Owners: Encourages retention.
- Success Measure: Positive user feedback for badges(?).

2.2.8 Analytics and Statistics

User Story: As a user, I want statistics about my task completion so that I can reflect on my productivity.

- Description: Charts of completed vs. pending tasks, reward earnings, and time-to-completion.
- Stakeholders and Business Value:
  - Users: Supports self-improvement.
  - Investors/Owners: Data provides insights for feature optimization.
- Success Measure: Feedback shows statistics aid goal-setting.

2.3 Detailed Performance Requirements, Specific Quality Requirements, Constraints

Tag: GIST: A short description to help understanding ambition/PLAN: The level at which success can be claimed SCALE: The scale of measurement used to quantify the statement METER: The process or device used to measure using the SCALE MUST: The minimum level required to avoid failure STRETCH: The best if everything goes perfectly WISH: A desirable level of achievement RECORD: The best-known achievement PAST: Previous results that may be used for comparison TREND: A set of historical data or an extrapolation of this STAKEHOLDER: A person or organisation materially affected AUTHORITY: The person, group or level of authorization allocated DEFINED: The official definition of a term

2.3.1 QR-01: Task completion speed

- Tag: Performance
- Description: Time from tap “Complete” on a task to reward screen shown.
- Scale: milliseconds (p95).
- Goal: $\leq$ 500 ms online.
- Stretch: $\leq$ 300 ms.
- Fail: $\geq$ 2 s.
- Priority: P0
- Owner: Dev team.
- Rationale: Responsiveness is crucial for habit-forming.

Is the above correct as an example of PLanguage?

Quality Requirements:

- QR-01 Task completion speed: milliseconds from tap “Complete” to reward/pet visible on client (p95). Target $\leq$ 500 ms online; $\leq$ 2 s offline cache; worst 1.5 s / 3 s.
- QR-02 App cold start: time to interactive “Today” (median/p95). Target 1.5 s / 2.5 s; worst 2.5 s / 4 s.
- QR-03 Availability of core flows: monthly uptime for create/open/complete. Target 99.5%; worst 99.0%; planned downtime announce $\geq$ 48 h before.
- QR-04 Notification timeliness: percent of due reminders delivered within 60 s of planned local time. Target $\geq$ 95%; default $\leq$ 3 per day; respect quiet hours.
- QR-05 Offline continuity: can add/complete without internet; sync on reconnect $\leq$ 10 s; no duplicate records.
- QR-06 Privacy and GDPR: data export $\leq$ 48 h; deletion $\leq$ 7 d (including backups per policy); store only needed fields (email, timezone); analytics opt-in.
- QR-07 Security baseline: TLS $\geq$ 1.2; encryption at rest; no hard-coded secrets in clients; pass basic mobile/web checks.

Quality scenarios:

- QS-1 Instant reward (QR-01): tap “Complete” and reward appears almost at once (p95 $\leq$ 500 ms).

- QS-2 Quick first use (QR-02): first open shows "Today" fast; user can add first task quickly (1.5 s / 2.5 s).
- QS-3 Offline then sync (QR-05): no internet, user completes tasks; after reconnect, sync finishes within 10 s.
- QS-4 Trust action (QR-06/QR-07): user requests account deletion; done within 7 days; no PII kept in logs.

2.4 Proposed prioritization

2.4.1 Next release

For the next release, we are planning to perform our elicitation techniques. This is pivotal for the project, as the elicitation will dictate the actual user needs and may change the direction of the project. The functional requirements will become more detailed.

Additionally, the next release will introduce the game factor that is part of the application. The description of the game will provide in-depth details for the reward system - a guide on how to spend rewards and in which way they benefit the users. This step is important in persuading the stakeholders of the importance of gamification, as we believe that the main driving force of the users will be the game of the application, acting as an agent to battle their procrastination.

Lastly, we will delve deeper in the business logic this project will follow, as in the marketing techniques, the balance between free features and premium subscription, and include citation for research showing common behavior that leads to procrastination.

2.4.2 Second release

In the second release, we will:

- Add an electronic pet (e-pet) as the game part of the application.
- Explore ways in which the e-pet will be interesting to interact with, such as customizing it and its habitat.
- Add a streak system, meaning that completing all or most part of daily tasks for X days in a row, the user will be awarded extra rewards.
- Decide which extra rewards are appropriate.
- Explore the concept of paying to save one's streak, if a user has missed a day.

2.4.3 Future releases (optional)

In future releases, we will:

- Integrate a house/habitat/environment for the e-pet.
- Generate weekly reports and statistics for each user.
- Add a store where a user can spend their rewards to customize their e-pet and its habitat.
- Explore the option of integrating other apps that may be relevant to the application, such as the system calendar or the 'Fitness' app most mobile phones have.
- Explore a cross-platform option, in which a user can have all information available in more than one device or device type.
- Research how users would benefit from interacting with other users and their pets on the app.

3 System Requirement

3.1 System requirements

3.2 UI prototype

3.3 Detailed data requirements

3.4 Acceptance tests (optional)