Software Aim Document for "notNTA" Project

# Project Overview

The "notNTA" project aims to gamify the competitive exams environment, such as JEE and NEET, by creating an interactive and engaging platform for students. The project will involve a series of phases, starting with a codenames-type game where two teams compete by answering past year questions (PYQs). The project will expand with added features such as customizable game settings, animations, additional games, leaderboards, anti-cheat mechanisms, and more.

# Phase 1.1: Basic Game Implementation

In this phase, the basic game structure will be created. The game involves two teams, red and blue, who take turns answering multiple-choice questions (MCQs) from past year questions (PYQs). Each team gets 2 minutes to answer a question, and they score points based on their accuracy. The game consists of 5 rounds, and the winner is the team that answers the most questions correctly.

# Phase 1.2: Game Settings and Host Controls

This phase will introduce settings for the game. The host will have control over the following aspects:

* - Ability to kick users from the game.
* - Modify game settings, such as the number of rounds and time per question.
* - Adjust the types of questions (topics) and difficulty levels.

# Phase 1.3: Enhanced Gaming Experience

In this phase, the game will be enhanced with better animations, a more engaging user interface, and attractive cards for questions and scoring. This will further gamify the experience and make it visually appealing.

# Phase 2.1: Additional Games and Exams

The second major phase will introduce more games and extend the platform to cover additional competitive exams. This includes introducing dynamic time-based and difficulty-based scoring, along with bonus questions for extra points.

# Phase 2.2: Leaderboards and 1v1 Platform

This phase introduces leaderboards to track performance across games. Additionally, a 1v1 platform will be created, where users can challenge each other to one-on-one matches. The platform will also include a ranked 1v1 mode, where users can answer PYQs to climb the ranks.

# Phase 2.3: Anti-Cheat Measures and Monitoring

To ensure fair gameplay, monitoring systems and anti-cheat mechanisms will be implemented in ranked matches. This will help maintain the integrity of the competition.

# Phase 3.1: Practice Platform and Story-Based Games

In this phase, a practice platform will be added to allow users to practice individually. Additionally, story-based games will be introduced, featuring characters and narratives. These games will focus on complex subjects like Operating Systems and Computer Networks, making learning more immersive.

# Conclusion

The limitations of the project are dependent on the technical stack used by the development team. The project has the potential to scale up to a large extent and is only limited by the imagination of the developers. The choice of technology and architecture will play a crucial role in the project’s success.

# Systemic Database Architecture and Technologies

The following technologies can be used to build the "notNTA" platform:

* - Frontend: React, Vue.js, or Angular for an engaging user interface.
* - Backend: Node.js with Express.js to handle server-side operations.
* - Database: MongoDB or PostgreSQL for storing user data, questions, and scores.
* - WebSockets: Socket.IO to manage real-time communication between clients and server.
* - Game Logic: Server-side game logic can be handled with Node.js, including scoring and time management.

Socket.IO will work with the database by connecting the client to the server for real-time interactions. Game data, such as player scores, game states, and question choices, will be stored in the database and updated in real-time as players make their moves. This ensures smooth communication between players and keeps the game in sync. For example, when a player selects an answer, it is sent to the server via Socket.IO, which checks the answer against the database, updates the score, and sends the result back to all players.

Add ons: the schema and structure is not idealized yet and will require rnd to figure the best architecture out with the mapped out plan.

As the steps further go up the complexity of backend and sockets will increase