

# Project Proposal: ITS Quiz App

## Project Description

### Problem to be Solved

- Existing ITS tools are not targeted for mobile devices or on-the-go studying
- Ability to share student-created study tools such as flashcards or quiz questions
- Better recommendations about what to study/what you know through ML models

### Proposed Solution

#### App Functionality

Mobile application that delivers simple quiz questions to users. We can display different formats: multiple choice (MC), vocab/concept flashcards, and short answer (SA). Based on how the student does, it can recommend what they should study next. These recommendations will be determined using a knowledge tracing model (to be developed by the knowledge tracing team).

#### Required Features:

- Display questions in 1 of 3 formats: MC, SA, flashcards
  - MC: multiple-choice → pick one option based on question (system decides whether the user got it right or wrong)
  - Term Matching: type one-word vocab term corresponding to given definition, right or wrong is based on exact string match
  - Flashcards: show front of flash card, user reveals back of flash card, user determines whether they got it right or wrong
- Collect whether user had a correct or incorrect response
- Users can choose different study topics/sets.
- Centralized Model API
  - Data collected by the app can be sent to the model API to get predictions
  - Deploying the model centrally also makes it accessible by other ITS applications
- Dummy Database for Testing
- Display progress: # of questions correct, incorrect, skipped

#### Future Features:

- Students can add their own flashcard sets.
  - May want a desktop interface for creating flashcard sets
- Students can share flashcard sets with other students.
- Short answer question format

## Benefits Of This App

- Convenience: Quick and easy for students to use since the majority of them are on their phone already.
- In the long run, this app will allow users to share study resources they created with other students.
- Provides a targeted learning plan for students who are struggling on a certain concept by prioritizing certain content
  - Many times when students are stuck, they are not sure where they are lacking in knowledge - the ML model would provide them questions that would be specifically catered to their skill set
  - Additional support - supplement/reinforcer to TutorJS application or alternative for students to ChatBot

## Potential Problems and Pitfalls/Other Areas for Research

1. What is the content for the questions? Is it vocabulary from the textbook? Practice problems? Will it differ based on question format (how are certain concepts displayed the best for learning?)
  - Content from the (digital) textbook
  - Questions from ITS - vocabulary, fill in the blank, specific equations, etc.
  - Questions created by students
2. Need to set data requirements on what type of data is needed to be collected for the ML models
3. In regards to future functionality additions, students are sharing flashcard sets with each other: How do we handle a situation where the student-provided answer is incorrect? Will other students be allowed to revise flashcard sets from other students?
4. Make sure we are designing and optimizing for a mobile environment.

## Implementation

### Software and Development Tools

- Android Studio
  - Mobile App
- Database
  - SQLite vs MongoDB (research to decide)
  - Hosting user data + questions API on phone vs cloud-base for short-term testing
- Libraries: TBD
- UI-UX wireframing: Figma
- Unit testing performed through Android Studio
- GitHub for file management
  - Git Kanban Board for workflow management

# Project Goals

The goal for this semester is to deliver a mobile application that will allow users to select a question set and complete a flashcard set. The app will have the ability to collect the results of the study session through an API endpoint. We will use a database to hold dummy questions and answers this semester, but the long-term goal for future semesters is to be able to send answers and receive questions from the Knowledge Tracing model. The app must work efficiently on an Android mobile device.

<b>Milestone</b>	<b>Date Due</b> (Hard deadlines are in <b>bold</b> )	<b>Status</b>
Project Plan Draft	<b>Friday 9/17/21</b>	Completed
Selection of tech stack and development tools for UI, front-end, back-end	Friday 9/24	Completed
Final Project Plan	<b>Friday 9/24/21</b>	In progress
Create github repo and initial project files	Tuesday 9/28	Not started
Environment Setup Completed + UX-UI research on quiz apps	Monday 9/27	Not started
App Architecture/Design Diagrams	Friday 10/1	Not started
UI-UX Research + User Interface Mockups + Database Schemas	Friday 10/8	Not started
Database Setup Complete + Finalize Wireframes	Friday 10/8	Not started
Start Coding Initial Screens + Layouts	Friday 10/15	Not Started
Midterm Peer Review	<b>Friday 10/8/21</b>	Not started
Initial list of functions to test and some test cases	Friday 10/22	Not started
Continue development - set functionality goals	Friday 10/22	Not started
Final Code Improvements and Check-ins	Friday 11/26	Not started
Project Cleanup and Final Documentation	Friday 12/3	Not started

Project Presentation	<b>Wednesday 12/8/21</b>	Not started
Final Peer Review	<b>Wednesday 12/8/21</b>	Not started
Project Submission	<b>Monday 12/13/21</b>	Not started

## Group Membership

### Member Skill Sets

<b>Member</b>	<b>Time Commitment and Credits</b>	<b>Skills and Interests</b>
Skye Aaron	3-4 hours/week (1 credit)	Python, Java, C, basic Android app development, SQL, data analytics, basic HTML and CSS.
Elias Reta	6-8 hours/week(2 credits)	Python, Java, React TSX/JSX, html, css
Khushi Magiawala	6-8 hours/week (2 credits)	React/JavaScript, Java, Basic HTML and CSS
Chinmayi Kompella	6-8 hours/week (2 credits)	Java, JavaScript, HTML/CSS, Angular
Vikas Barevadia	3-4 hours/week (1 credit)	Java, Python, interested in DB portion

## Task Assignment and Participation

Some roles to fill (could be multiple people):

- Project Manager/Facilitator - Updates tasks/timelines, checks in on progress, schedules meetings, facilitates meetings
  - Skye Aaron
- Scribe - Takes notes during meetings; works with Project Manager to disseminate notes
  - Khushi Magiawala
- Advisor on tech stack and development tools
  - Skye Aaron
- UI-UX researcher/designer - Create mockups of how the UI should look and all the different screens
  - Khushi Magiawala

- Skye Aaron
  - Elias Reta
- Front-end developers - Android studio and coding up UI
  - Whole Team
- Back-end developer: (although connecting to KT, we will still need a database this semester)
  - Chinmayi Kompella
  - Vikas Barevadia
- Tester - Writes unit tests for app functionality; Tests front-end
  - Whole Team

## Communication Resources

- Weekly meetings Fridays 10:30-11:30 am ET via Teams
- Scheduler: Lettuce Meet
- Google Drive: meeting notes, design documents
- WhatsApp for quick communications