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

Currently, the mobile application delivers simple quiz questions to users. We 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 That Are Finished:

- 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.
- Dummy Database for Testing
- Display progress: # of questions correct, incorrect, skipped

Future Required Features:

- Setup application to handle communication with KT model
    - Potential API development
- Generating quizzes tailored to a content area recommended by KT model
    - MongoDB data extraction
- Potential local SQL-Database to store current user session info instead of on MongoDB?
- Make the application look aesthetic
- Streamline/Refactor development process

Future Semester 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
- Design chatbot app and integrate with android studio

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. Setting data requirements on what type of data is needed to be collected for the ML models - talk to the KT team and standardize data collection and processing on both ends.
3. Connecting a cloud database to our application and successfully pulling/displaying questions on the screen.
4. Make sure we are designing and optimizing for a mobile environment.

Implementation

Software and Development Tools

- Android Studio
    - Mobile Application
- Database

    - Cloud-based MongoDB
- API for KT communication
    - TBD
- Libraries: TBD
- UI-UX wireframing: Figma
- GitHub for File Management
    - Git Kanban Board for workflow management
- Figma
    - Any additional Prototyping

Project Goal For This Semester

Last semester, we delivered a functioning mobile application that allowed users to select a quiz and work through the questions. The goal for this semester is to set up the mobile application to handle communication with the KT model, allowing the app to recommend specific content areas based on user performance and provide a more targeted learning plan. This will require standardizing what type of data we want to send, what type of data we will receive, and how to process the data. We also want to refine the data extraction from our database to ensure that we are pulling the correct questions of the content area recommended by the KT model. We also want to develop and maintain an aesthetic, and refactor our code to streamline our development process.

Milestone	**Date Due** (Hard deadlines are in **bold**)	**Status**
Project Plan Draft	**02/02/22**	In progress
Getting everyone set up on android studio + solidify proposal	02/07/2022	Not Started
Final Project Plan	02/09/2022	Not Started
Fork last semesters github repo and initial project files + continue onboarding	02/14/2022	Not Started
Refresher on Android Studio	02/21/2022	Not Started
Split into Teams and Start Work Backend: Talk to KT team Front-end: Talk about UI upgrades and standardizing screen development	02/28/2022	
Front end - analytics screen, single sign-on	TBD	Not Started

Continued work - depends on previous progress - see requirements section	TBD	Not Started
Continued work - depends on previous progress - see requirements section	TBD	Not Started
Continued work - depends on previous progress - see requirements section	TBD	Not Started
Continued work - depends on previous progress - see requirements section	TBD	Not Started
Prepare for Demo + Final Touches	TBD* - 1 week	Not Started
Demo	TBD*	Not Started
Project Cleanup and Final Documentation	04/18/2022	Not Started
Final Presentation	**April 27th, 2022**	Not Started
Final Peer Review	**TBD**	Not Started
Project Submission	**May 3rd, 2022**	Not Started

Group Membership

Member Skill Sets

Member	**Time Commitment and Credits**	**Skills and Interests**
Khushi Magiawala	6-8 hours/week (2 credits)	React/JavaScript, Java, SQL, Basic HTML and CSS, Android Studio
Chinmayi Kompella	6-8 hours/week (2 credits)	Java, JavaScript, HTML/CSS, Angular, Android Studio
Krishan Patel	6-8 hours/week (2 credits)	Java, JavaScript, HTML/CSS, Python, C/C++, React/React Native, Node.js, SQL
Neha Lalani	6-8 hours/week (2 credits)	Figma, Java, Python, Basic CSS, Basic HTML

Sakshi Deshpande	4 hours/week (1 credit)	Java, JavaScript, HTML/CSS, React, Python
Abdulaziz Memesh	4 hours/week (1 credit)	Java, Python

Task Assignment and Participation

Team Logistics

- Project Manager/Facilitator - Updates tasks/timelines, checks in on progress, schedules meetings, facilitates meetings
    - Chinmayi Kompella
- Scribe - Takes notes during meetings; works with Project Manager to disseminate notes
    - Khushi Magiawala

Work Division

- Front-end developers + UI/UX redesign - Android studio and coding up UI for Quiz App
    - Neha Lalani
    - Khushi Magiawala
    - Abdulaziz Memesh
    - Sakshi Deshpande
- Back-end developer: (1) standardize and develop KT communication, (2) refine MongoDB data extraction - work with front-end developers
    - Chinmayi Kompella
    - Krishan Patel
- Designing + Developing the Chat-Bot Application (new feature)
    - Neha Lalani
    - Sakshi Deshpande

Communication Resources

- Weekly meetings Mondays 12pm-1pm EST via Teams
- Scheduler: Lettuce Meet
- Google Drive: meeting notes, design documents
- WhatsApp for quick communications