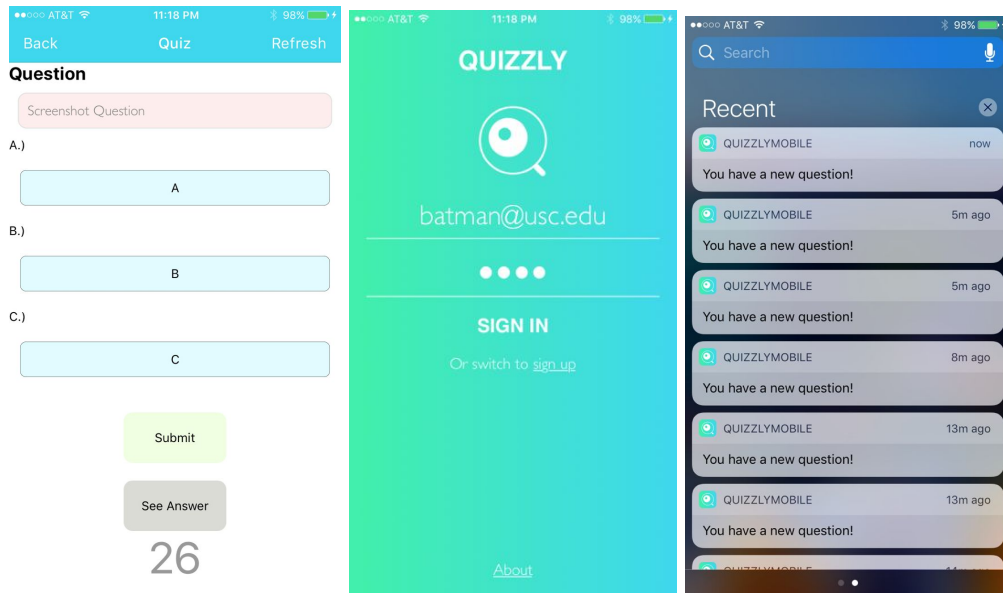


Description

Quizzly is a platform that provides convenience for professor to give out a quiz and for students to take it in class using their smartphones.



Stakeholder:

Reem Alfayez

Team:

Jacob Rey

Anthony Tu

Eddie Hays

Rahul Chander

Wilson Yeh

Installation

- Install React native: <https://facebook.github.io/react-native/docs/getting-started.html#content>
- [brew install node](#)
- [brew install watchman](#)
- [npm install -g react-native-cli](#)

Run

*iOS

- cd to the folder QuizzlyMobile/
- run the following command
[react-native run-ios](#)

*Android

- Current development doesn't support Android deployment.

Contribute

Special thanks to Professor Miller and Reem Alfayez for supporting the project.

License

The project is a USC CSCI 401 Capstone Class Project.

The following documentation serves as a reference to the project structure and function that has been written in the Quizzly Mobile App for iOS.

1. File Structure
2. Detailed File Explanation

1. File Structure.

Log In Page ----- Entrance.js
Home Page ----- Courses.js
Specific Course Page ----- Course.js
Quizzes Page ----- QuizzList.js
Specific Quiz Questions Page ----- QuestionsList.js
Answer Question ----- Questions.js

2. Detailed File Explanation

[AnswerQuestion.js](#)

Displays the question object received and prompts the user to answer. This page is triggered when the professor asks a question in class and it can be either through the notification or student clicks on 'Take Quiz' button.

[AnswerQuiz.js](#)

Displays the quiz object received and prompts the user to answer. It links to a QuestionList.js object and eventually utilizes the AnswerQuestion.js object for the user to send out a response. This page is triggered when the professor asks a quiz in class and it can be either through the notification or student clicks on 'Take Quiz' button.

[Answers.js](#)

Takes in a question object and displays its answer. It is triggered once the user clicks on the 'See the Answer' button under the question page from 'View My Score' Section.

[Course.js](#)

Acts as the home page once the user logs in. It has two buttons in display. One is Take Quiz and the other is View My Score.

CourseRow.js

Serves as a wrapper for the Courses.js file.

Courses.js

Receives course names from the database and displays them in table.

Entrance.js

Serves as the first page that user sees upon opening the app. It handles log in, sign up and authentication functionality.

Login.js

Serves as the mechanism that allows the application to gain access to the server and pull the corresponding information.

NavBar.js

Serves as a component file for the Navigation Bar at the top of each page.

PushNotification.js

Serves as a dedicated file for push notification functionality within the app. It's mainly being used under the condition that professor asks a question or quiz.

QuestionRow.js

Shares similar structure as CourseRow. It serves as a component wrapper for Questions.js file. The difference is CourseRow has a correct answer rate component while QuestionRow doesn't.

Questions.js

Takes in a question object and display the answer and question text in the page. It is triggered once user click on the question inside the QuestionList.js.

QuestionList.js

Triggered once the user clicks on the quiz object in QuizzList Page. It contains a list of questions that belong to that quiz.

QuizzList.js

Allows the user to see what quizzes they have for a specific course. It shows the taken and untaken quizzes.

QuizzList_temp.js

Serves as the template for how the QuizzList will be displayed.

QuizzRow.js

Similar to QuestionRow.js. It serves as a wrapper for Quizz Object displayed in the quiz list.