Selected Topic : Learning material to help users learn the computer network class, especially TCP/IP and OSI models, ARQs, Routing protocols, TCP congestion control, including quiz, and solutions.

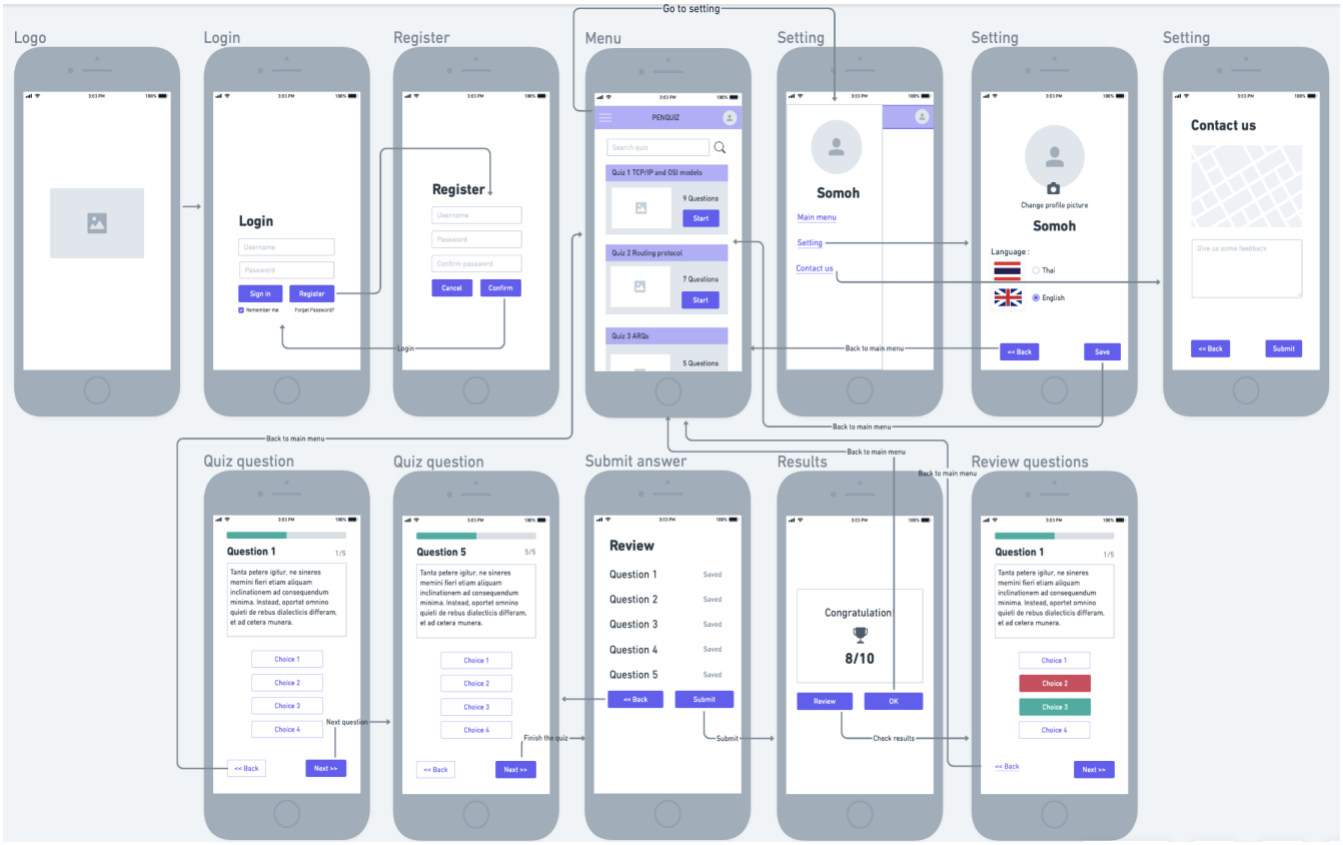
PENQUIZ - Application Overview

The application will basically give the users the best practice for computer network content within the area specified above. Firstly, the user needs to create his/her profile in order to keep the record and the best score. Then, the user will choose a quiz which they want to do. After they choose, the application will show the content list in various topics of computer network content. To make the user access quiz quicker, the application has a search feature to suit this. Another feature is that users can keep tracking his/her best score for a quiz, the application will allow users to track the mood when they finish the quiz. The application supports both English and Thai.

The following are application features included in the PENQUIZ mobile application

1. **Quiz Time**
    a. The application will provide the computer network content. The level of the quiz would be simple and give basic knowledge to the users. The quiz will have one screen per question.
2. **Score calculation with reviews**
    a. The score of the quiz will be calculated at the end of the quiz. The application will show a score in the dialog box and provide the review for each question for users.
3. **Searching Quiz**
    a. The application has the search box for the users to search the typical name of the quiz. If users search TCP model, the quiz about TCP model will show in the application. This will allow users to do the quiz they want.
4. **Create your profile**
    a. The application can create the profile and allow a user to change language in the application.
5. **Challenge yourself - Keep tracking your best score**
    a. The application will keep tracking the best score of each quiz. This will allow the users to challenge themself
6. **Contact us**
    a. The application allows a user to contact or give feedback to the developer team of this application. It also displays the location service of our headquarters on a map

Wireframe


Logo
Login
Register
Menu
Setting
Setting
Setting
Go to setting
Login
Register
Cancel
Confirm
PENGUIZ
Quiz 1 TCP/IP and OSI models
9 Questions
Start
Quiz 2 Routing protocol
7 Questions
Quiz 3 ARQs
5 Questions
Somoh
Main menu
Setting
Contact us
Back to main menu
Change profile picture
Somoh
Language :
Thai
English
<< Back
Save
Contact us
Give us some feedback
<< Back
Submit
Back to main menu
Quiz question
Quiz question
Submit answer
Results
Review questions
Question 1
Tanta petere igitur, ne sineres memini fieri etiam aliquam inclinationem ad consequendum minima. Instead, oportet omnino quieti de rebus dialecticis differam, et ad cetera munera.
Choice 1
Choice 2
Choice 3
Choice 4
Next question
<< Back
Next >>
Question 5
Finish the quiz
Review
Question 1 Saved
Question 2 Saved
Question 3 Saved
Question 4 Saved
Question 5 Saved
<< Back
Submit
Congratulation
8/10
Review
OK
Check results
Back to main menu

PENQUIZ - Checkpoint Overview

In the previous checkpoint, we have created two activities including a login system and register system on the PENQUIZ application and keep data on the firebase. In this third checkpoint, we focus on five activities consisting of implementing a home page, implementing contact us page, implementing a setting page, implementing quiz question page ,and adding quiz questions on firebase

The objective of the project

This project aims to encourage users to learn the material from the computer network class by doing an online quiz on the mobile application. The users have to create an account for the first login before using an application so the users can access an application to do online quizzes. Once users login to the system users can select any topics that display on the Home page. All quiz questions will be pulled from the database, firebase, and display on each page while users are doing a quiz. Moreover, users can see the final score when they finish doing a quiz. After users complete the quiz, they can give any feedback or comment about the system to the administrator on the Contact Us page such as provide more questions of each topic.

The application design

- **Home Page** = show the quiz objects for users to choose the quiz they want to do

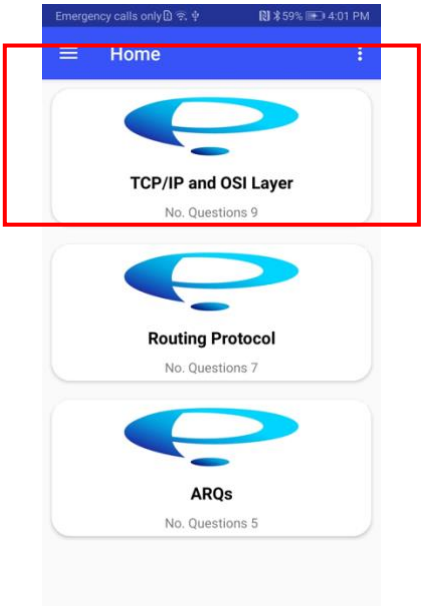
1. This fragment contains a `RecycleView` with `Quizes` object retrieved from database by

```
quizesRef=FirebaseDatabase.getInstance().getReference().child("Quizes")
```

2. Therefore, we use `FirebaseRecycleOptions` to generates `Quizes` object

```
options: FirebaseRecyclerOptions<Quizes> =
FirebaseRecyclerOptions.Builder<Quizes>().setQuery(quizesRef,
Quizes::class.java).build()
```

3. We use `FirebaseRecyclerAdapter` to generate custom adapter that we implemented in `CustomViewHolder` class. Therefore, for each of quiz object, it will show quiz name and number of questions.



- **Quiz Questions Page** = show questions according to users' selected topic in a home page. Each topic provides a different number of questions. Each question page will display question text and question number at the top page as well as display 4 options with the description in square block*(Figure 1)*. **In this checkpoint,** when users select an option, it will navigate users to the next question page and also display the green block if the answer is correct*(Figure 2*). Otherwise, it will display the red block and show the correct answer*(Figure 3)*. However, we haven't created the intent for the next button and the back button yet.

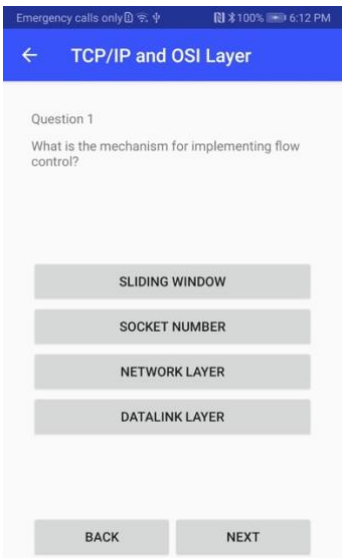


Figure 1 : Display question on each question page

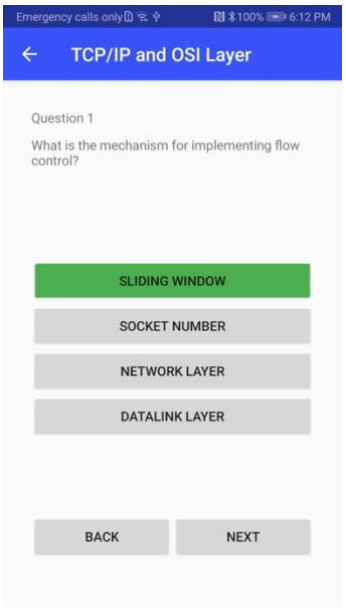


Figure 2 : Display the green block if users select correct answer

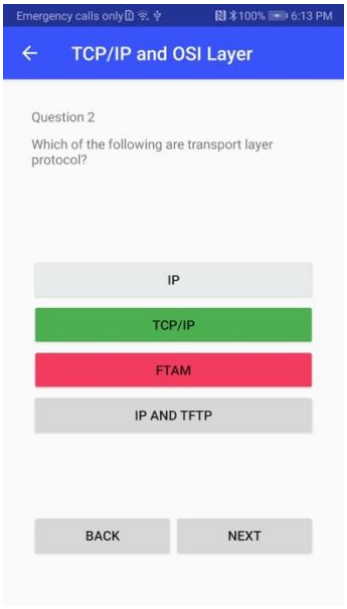
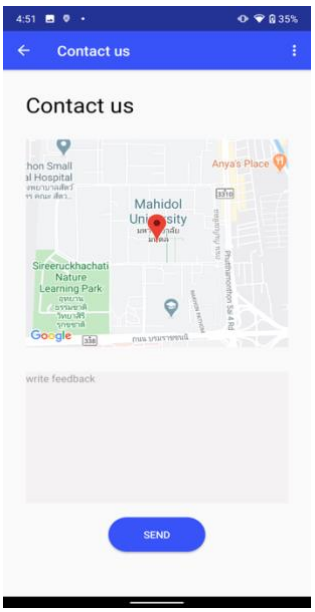
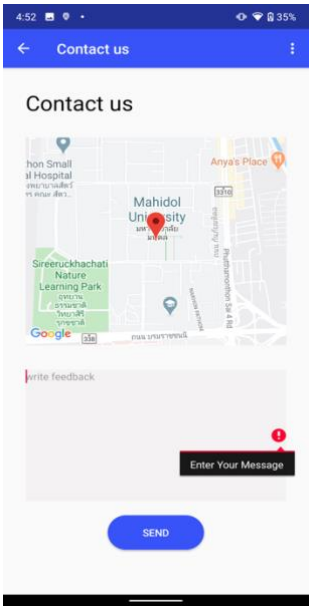
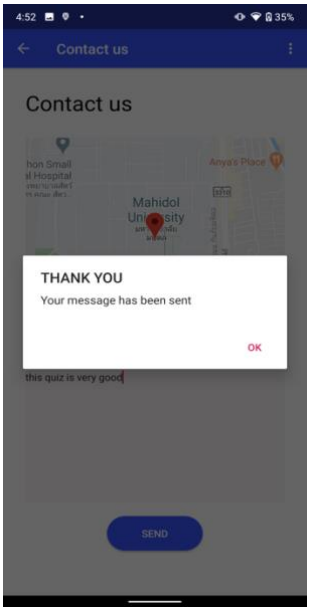


Figure 3 : Display the red block if users select wrong answer

- **Contact us Page =** show the application developer's company and giving feedback to them. If the users want to give feedback or see the map of developer's company, users can go through contact us menu in the navigation menu.



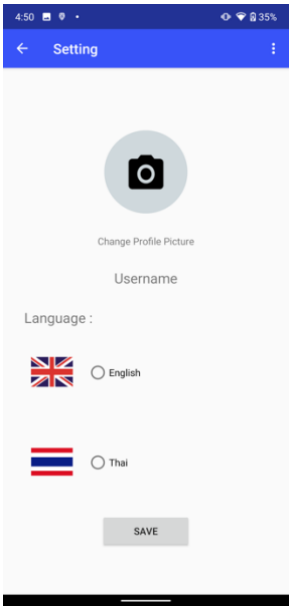




  - **Google Map API** :
    Contact us page will mainly show the location of the application developer's company, Mahidol university, by using Google Map API. In order to create a Google Map, we need `google_map_key` given by Google. The Mahidol University location will display in a square block with a marker.

  - **Giving feedback** :
    Users are able to give feedback to application developer's by typing in the text box below the Google Map and clicking the submit button. After the submit button is clicked, the alert dialog blog will be shown to display a thank you message. However, if the users click the send button without input any message, the message block will alert an error to notify users to input some message.

- **Setting Profile Page** = This page shows the profile of the user. Users can set the language and picture of the user's profile. If the users want to change the default setting of the application, users can go through setting menu in the navigation menu.



  - **Camera** :
    In this checkpoint, we create layout for setting page. The circle imageButton will show the selected image from the users by upload their image. By default, the image inside imageButton is camera icon which give a sense for users when they want to change his/her profile's picture. Also, username of the current user will display under the circle imageButton.

  - **Language** :
    The application provides multiple languages which are English and Thai. The user can select the language by selecting the one radio button for the language that user requires.

- **Database design**

  - **Feedback from user** : When users write some feedback on the mobile application and click submit button, all feedback will be transmitted and stored in the firebase database. On the firebase, there is a root called Messages that will save the real time feedback from each user. Every time users send feedback, each feedback will be pushed into a sender ID branch and feedback will be saved according to their unique key *(Figure 4)*. The unique key of each user is generated once users register for the first time. For

example, this unique key, 1TVY1ZAYciWL5VsG3aSxn2SrZVu2, is generated for the first user.

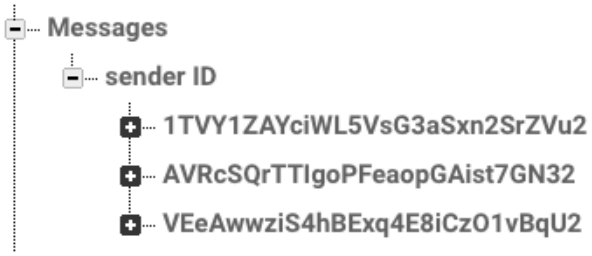


Figure 4 : display path storing the real time feedback according to unique key of each user

To push the real time feedback object in firebase database, the feedback object will be save to specific database reference by write operation like,

```
myRef.child("Messages").child("senderID").child(senderID.toString()).push().setValue(message)
```

When the feedback object is saved in the database, the object will be mapped according to specific path location in a nested fashion*(Figure 5)*.

Figure 5 : Display how each feedback object stored in a nested fashion

- **Quiz questions** : our group stores questions about the computer network area including TCP/IP and OSI models, Routing protocols, and ARQs on the firebase. On the firebase, it will have another root to retrieve quiz questions called Quizzes. This root contains three sub branches to represent the identification(ID) of each quiz topic. For example, the TCP/IP and OSI model

topic has ID 1, routing protocols topic has ID 2, and ARQs topic has ID 3. With this database design, it helps us to retrieve the questions of each topic or information of each topic easier when we want to read content from the firebase and display it on the application. Moreover, each sub branch consists of three child nodes which are the important information of each topic including Questions, total number of questions, and question topic*(Figure 6)*.

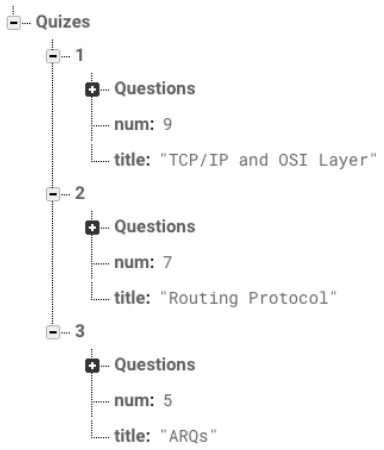


Figure 6 : Display the database child location

In the Questions child, it contains the identification number of each question ranging from 0 to the last number of questions. Each identification number consists of several descendents of child nodes including question text, choice from a to d, and correct answer of that question. Therefore, we can retrieve any information from the firebase by following the path that we specify on the firebase*(Figure 7)*.

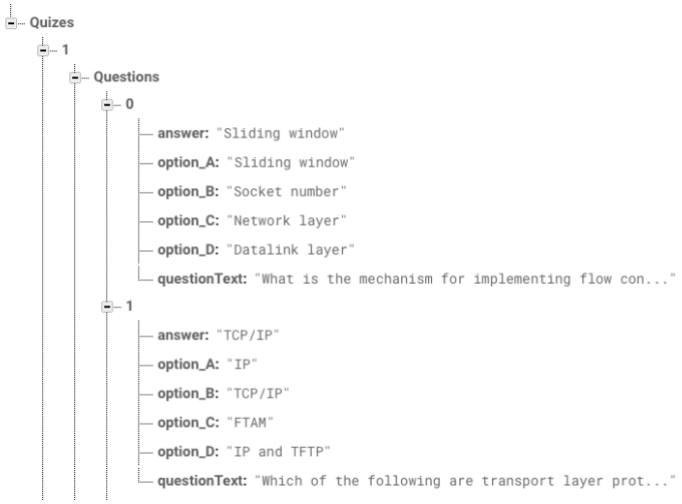
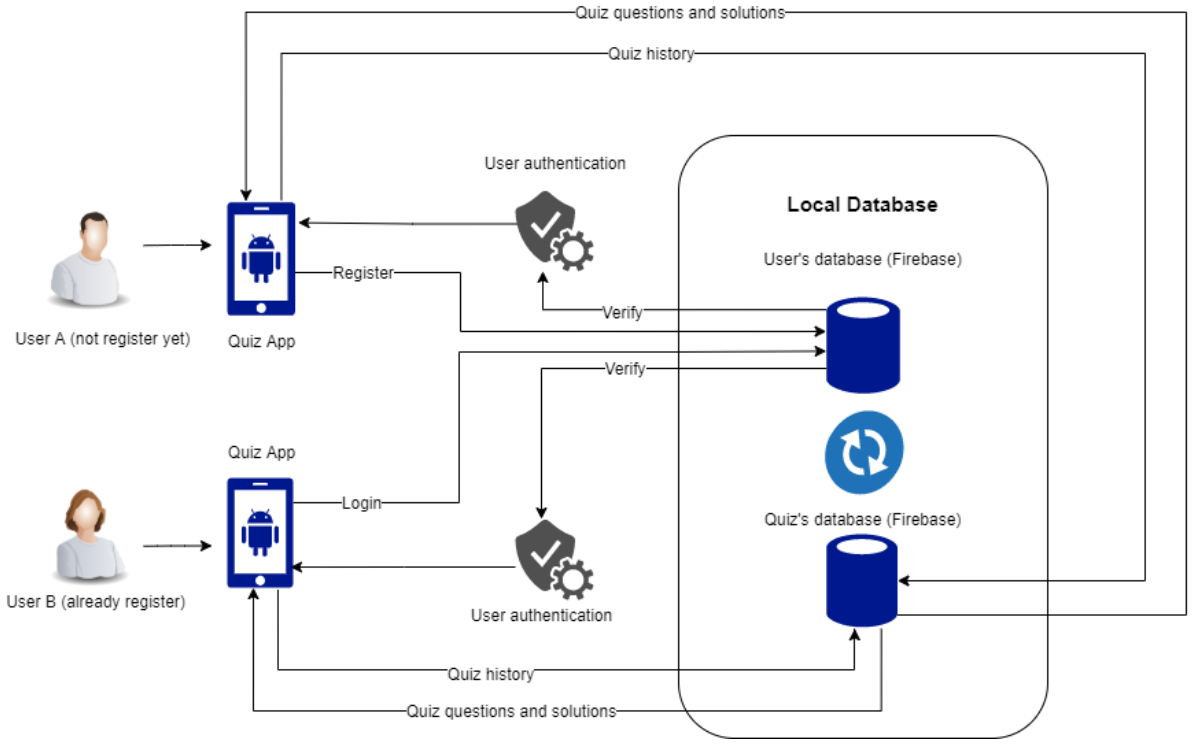


Figure 7 : Display list structure of questions

The system architecture

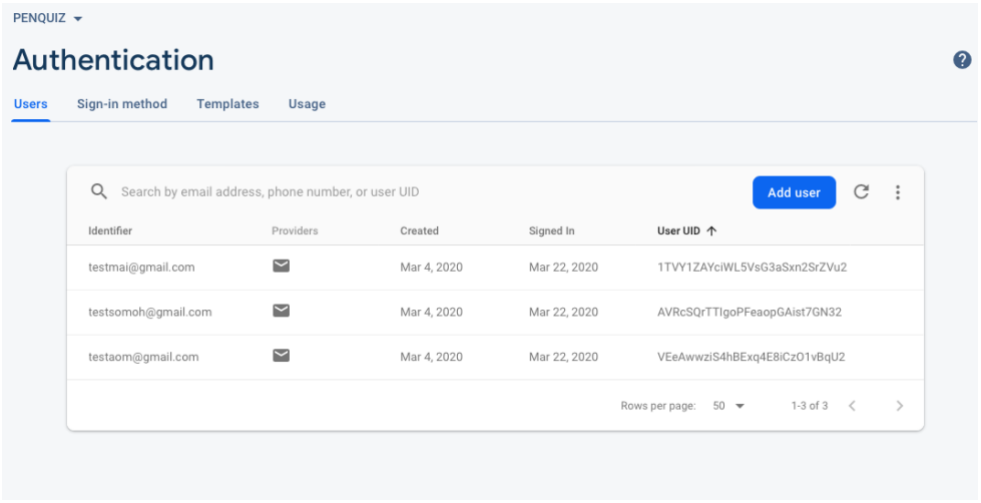


From the picture, it illustrates the system of quiz service. It has two cases of users which are user A and user B.

1. **User A (who does not register yet)**
    - Firstly, users have to register before using an application, the information of the user required are username and password. Therefore, the system will create the user's database on firebase and bring that information to keep in the database.Next, the system will tell the users to do the users authentication and then the users can log in now. After that, users can access the quiz service. When user do the quiz, system will create the quiz's database on firebase for keeping
2. **User B(who already do register)**
    - Users can access the quiz without doing the register. We already create the username and password (username : testsomoh@gmail.com , password: 123456) that keep in the user's database on firebase. Therefore, users can do the quiz. And when users do the quiz, the system will keep it on the database and send it to the quiz app.

3. **Quiz's Database**

- In this database, it keeps several quizzes that users require to do. One quiz consists of QuestionID,questionText, correct answer,option 1,2,3 and 4. When a user does the quiz, the system will show which is the wrong and true choice. After that, when the user finishes doing the quiz. It will show the score of the quiz that you get from the quiz that you choose.



4. **User's Database**

- After the user has done registration, the system will keep information about the user. There are UserID, username, password, sigin_date,login_date. Moreover, unique key of each user will be generated.