Project Proposal

Open source based Learning App

Group 9

07/10/2019

Part 1.Features analysis

The login part of application

Under the consideration that we are going to have two kinds of users- mentors and learners, we plan to implement "one entrance to different users" principle. This means the mentors and learners will have the same registration and login interface and what different is that user must choose their attribute (mentor or learner) at the beginning of registration and login.

Use case: register and login

User	All user
Work flow	As a user, I can register an account for myself by choosing either to be a learner or a mentor and fill all the information needed: email address, username, password, confirm password
Precondition	-
Result	As a user, if I choose learner as my identity, I can only ask questions. If I choose mentor as my identity, I can only explain the question raised by learner.

Main functions for learner

After learner logged into the extension, learner can start ask their question about coding when they are browsing Github. The extension will provide three main functions for learner: asking question by keyword; asking explanation for a whole sentence of coding; asking causal problem.

(1) search by keywords

The learner can search keywords of program, like for, while, if …. Then the chatbot will give the explanation of this keywords and tell how to use these kinds of keywords.

Use case: asking explanation of key words

User	Learner
Work flow	As a learner, I should click the extension and a chatroom interface will be showed to me. Then I input the keywords, which I don't understand, and click the send button.
Precondition	As a learner, I need to login in first
Result	As a learner, I will get response in a few seconds (less than 10 seconds). Either I can receive the explanation from the chatbot or chatbot will give me a link to a mentor who can explain the question for me.

(2) search one completed sentence of program

The learner can ask question by input a whole (completed) sentence of program. Then the chatbot will give explanation of this sentence

Use case: asking explanation of sentence

User	Learner
Work flow	As a learner, I can ask the explanation of a completed sentence (just one line of the program or code) by inputting the sentence into the dialog and

	clicking the send button.
Precondition	As a learner, I need to login in first
Result	As a user, I will get response by waiting for a while (less than one minute), either I can receive the explanation from the chatbot or chatbot will give me a link to a mentor who can explain the sentence for me.

(3) search for general question

The learner can ask questions not related to coding, just daily greeting or normal communication

Use case: daily greeting

User	Learner
Work flow	As a User, I can chat with application by sending greeting words. For example, I can just type “hello”, “how are u?”
Precondition	As a learner, I need to login in first
Result	As a user, I can chat with chatbot and will get response quickly (wait less than 5 seconds to get a response)

Main function for mentors

The mentor can answer the question, which cannot be explained by chatbot, and communicate with learner interactively.

Use case: explain question manually

User	Mentor
Work flow	As a Mentor, I wait for connection from learner and accept it. Then start answer the question raised by the learner
Precondition	User has to login as a mentor
Result	As a Mentor, I teach learner how to do coding and problem get solved

Main function for information collection part

During the process that question get solved by mentors, the application will collect the question raised more frequently (top 5). Then save the correct answer given by mentor into database of chatbot.

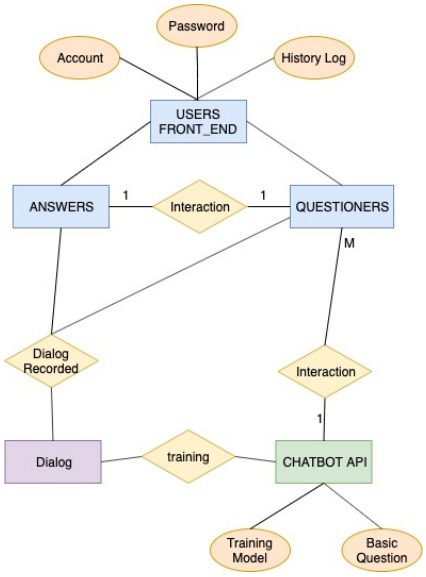
Use case: data collection

User	Training center
Work flow	The application will collect and record the question answered by mentor and save the record into the database. Then we will choose top 5 most question solved by mentors and store the question in the database of chatbot.
Precondition	It needs a period of accumulation
Result	Next time when the same question raised by learner it won’t need to wait the mentor to answer it. The chatbot can solve it automatically.

Part 2. System Design

The main elements of our application include 4 parts. The first part is USER FRONT END, which can split into Answers or Questioners according to the role of users. There are three attributes in this entity; account and password are used for login, and users' history log. The second part is BACK END API, this part is used for control most of the data flow during the interaction between the questioners and answers and achieve most of functions. The third part is CHATBOT API, it offers basic interaction and gives response for simple questions, and it can train some questions and giving feedback. The last part is DATABASE which is for recoding data.

(i) An Entity-relationship (ER Model) of data layer.



(ii) The System architectural diagram below, showing the main system components and interactions of our application.