Computer Science, University of Houston Victoria

Project Overview

Team Name: Ensoft T1 Solution Architects

Caleb Tabora, Ryan Snider –Ainley

Ajmal Sahak, Shearon Fonji

Joseph Dodson, Sarah Crawford

Instructor: Mojtaba Masoumi

COSC 4320 – Software Engineering – Fall 2022

Date: Fall 2022

Main.py

Imports: Starting at the top of the main.py file are a multitude of imports for kivymd to run the user interface.

Class Todocard: Initializes a todo object, including a default label, date, time, and id.

Class user:

* Def load\_task: Loads the tasks.
* Def setname: Sets the name of the user.
* Def setdata: Sets the data of the task, including the date, time, and task name.

Class login:

* Def list\_task(self): Displays all of the tasks
* Def checker(self): Checks the time
* Def refresh(self): Refreshes the task list
* Def build(self): Initializes the login page and buttons on the login page
* Def Logout(self): Initializes a logout
* Def deleteData(self,instance):Deletes a task
* Def get\_id(self,instance): Returns a specific task
* Def login(self): Allows a user to login, checks for an existing login, if there is a username and password that matches, the user is logged in, if not “Account Not Found” is displayed
* Def TaskButton(self): Displays a task button.
* Def on\_save\_date(self,instance,value,date\_range): Saves a task
* Def on\_cancel\_date(self,instance,value): Cancels setting up a task
* Def get\_time(self,instance,value): Gets the time of a task
* Def Dpick(self): Allows the user to pick a date
* Def Timepick(self): Allows the user to set a time
* Def Addtask(self): Adds the task to the task list
* Def Signup(self): Creates the signup page
* Def AccountCreate(self): Allows the user to create an account, a proper username and matching passwords are checked
* Login().run(): runs the app

Login.kv

* Provides the layout for the login screen.
* Includes: an email/username text box, a password text box, a login button, and a signup button

Login.png

* A png image to display on the login page

Signup.kv

* Provides the layout for the signup screen
* Includes: an email/username text box, a password text box, an additional password text box, and a create account button

Signup.png

* A png image to display on the signup page

Task.kv

* Consists of the GUI for the create a task page, along with the methods and calls needed in order to get the time, date, and name related values for the new task instance.
* Includes set time, set date and create task buttons, along with a text box that is used to set the task name.
* Task name will default to “no name” if left blank.

Welcome.kv

* Consists of all the user interface for the login page after the user has successfully completed the login/signup processes during the program’s runtime. Shows a list of all existing tasks, as well as the logout, create task, and refresh functions through GUI.
* Includes a refresh, logout, and create task button, along with a scrolling list of all user created task instances, and a display of the user’s id.

Read\_me.txt

* Provides instructions for running the app on cmd.
* Requires the installation of pip and Python (version 3.9 is recommended).

Requirements.txt

* Includes all the libraries and packages needed to run the app.
* Will install or use current version of all necessary libraries and packages when calling the command listed in the Read\_me text document.

**Sarah additions to main manual entry:**

Are the port and the socket for the messenger definition. I import port and threading to have this accessibility to the applications value. I usually initialize the application to deploy but my other team members did not defining the limitations of technical skills as a group.

**Main1.py**

Initializes socket to socket encrypted communication and validates inputs.

**AndroidChat.kv** is the mobile app visual gui that the user inputs the message into and is accessed by the button addition on the main page.

**Venv** I added as well to create containerized environment for the virtual app.

**Unused imports** removedto reduce the size of code inherently saving money on unessecary cost. Auto merge in git did not work out for the differences of opinions of best coding practices so I went in manually to add the extension files for the added features.