OTP Verification System

Python Capstone Project

Prashantsagar Uppara (S8083)

Contents

Project Charter

Problem Statement	Project Requirements
• Developing an OTP (One-Time Password) verification system in Python. The system should generate a 6-digit OTP and send it to the user's email address for verification • Upon receiving the OTP, the user should enter it into the system for validation • If the entered OTP matches the generated OTP, access should be granted; otherwise, access should be denied	• Implement a function to generate a 6-digit OTP randomly • Develop a function to simulate sending the OTP to the user's email address • Create a function to prompt the user to enter the OTP received in their email • Implement a function to verify if the entered OTP matches the generated OTP • Ensure proper error handling and user-friendly prompts throughout the system • Allow the user to retry OTP entry in case of incorrect input

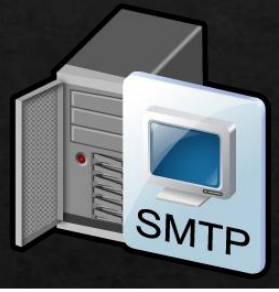
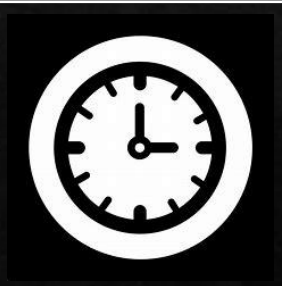
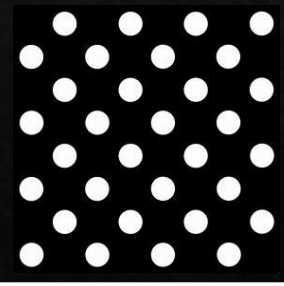
Project Charter

Project Objective	Project Scope
• Creating an OTP verification system capable of sending OTP to the user and verifying it	• Creating python-based OTP verification system • Building user friendly & secure GUI, if possible • Documentation & test cases

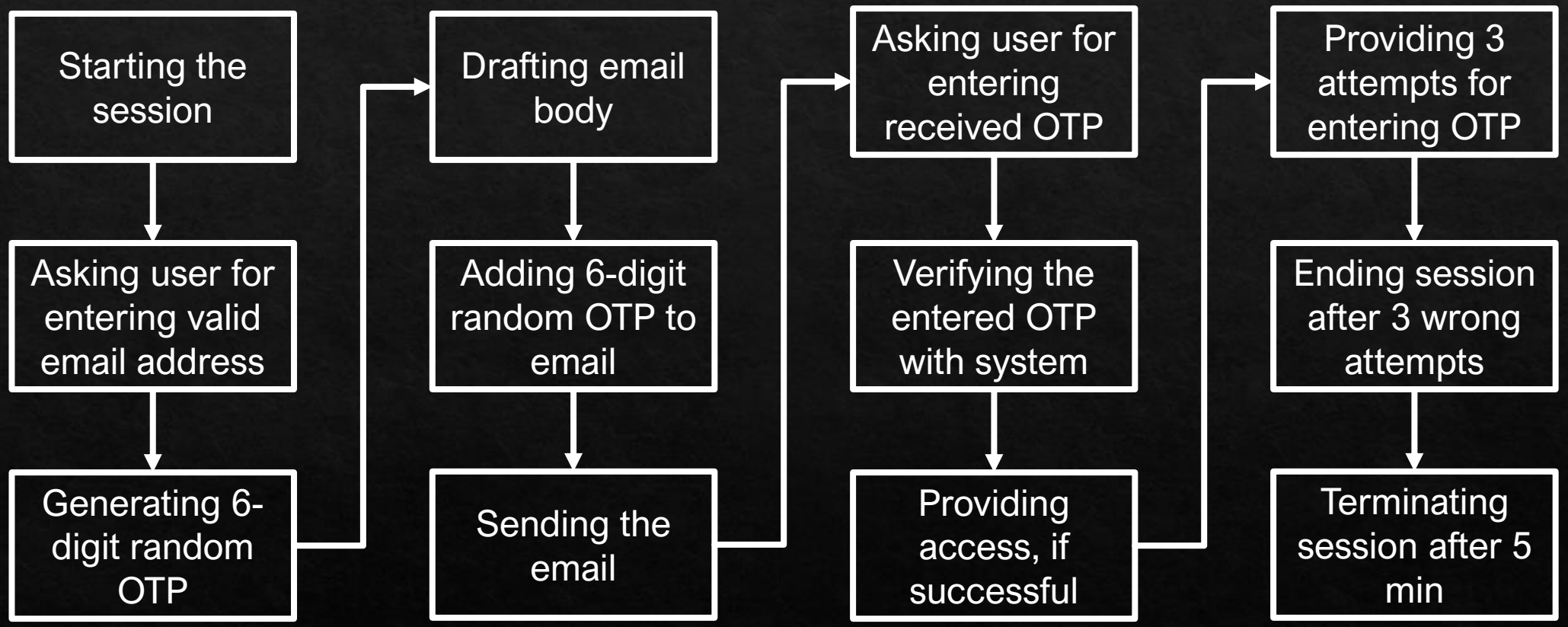
IDE Used

Jupyter Notebook

Libraries Used

		
SMTPLIB Simple Mail Transfer Protocol	email 4.0.2 (MIME) Multipurpose Internet Mail Extensions	PyQt6 Python bindings for cute version6
		
Time	re Regular Expressions	random Generate pseudo-random numbers

Process Flow



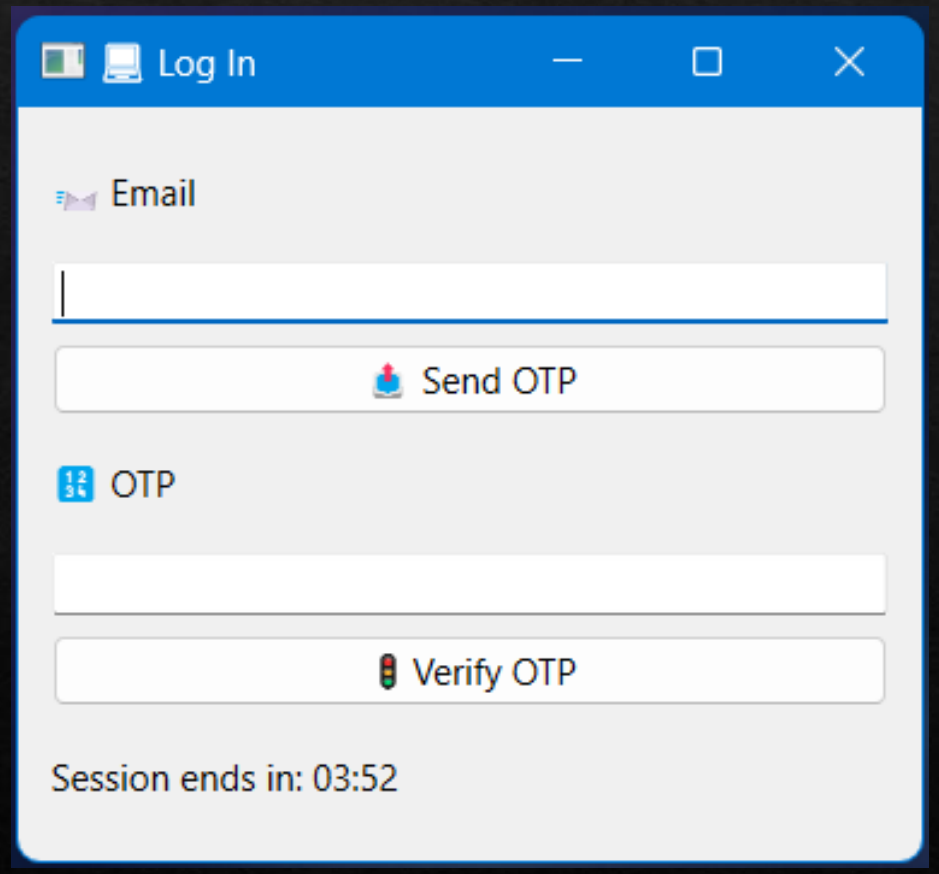
Robust Features Inbuilt In The System

- Each session is valid for 5 minutes, after which the session terminates automatically & no OTP can be generated once the OTP is sent to the entered email address
- Entered email format is validated before OTP generation, random strings are not accepted
- After sending the OTP to email, message is shown to the user
- 3 attempts are provided for entering OTP, after each attempt input box is frozen for 30 sec for ensuring that the user checks & enters right OTP
- After 3 attempts, session is terminated
- Random strings & characters are not accepted as OTP

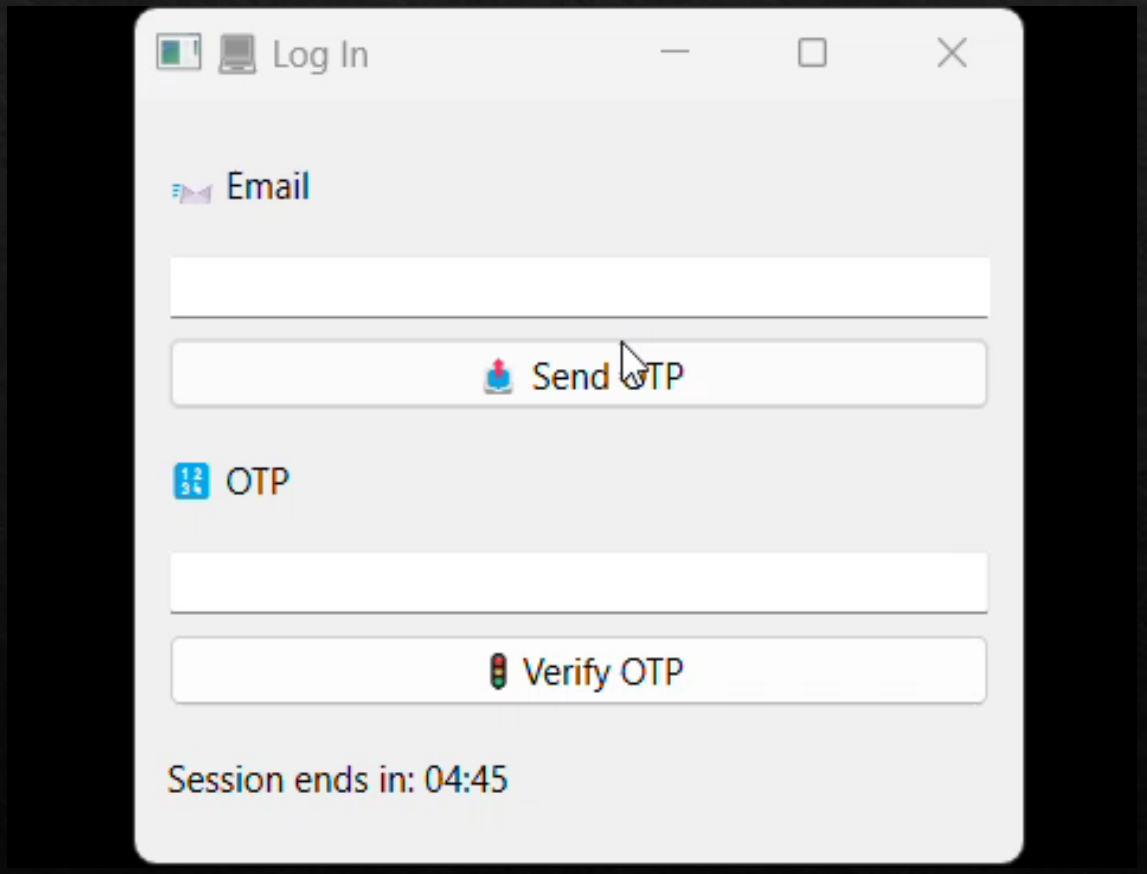
Challenges Faced

- Generating appropriate email body with subject & email body
- Generating GUI form
- Integrating GUI with Jupyter notebook
- Integrating considered features in the code & GUI
- Generating time-based sessions & making it work parallelly with other considered features

User Friendly Graphical User Interface

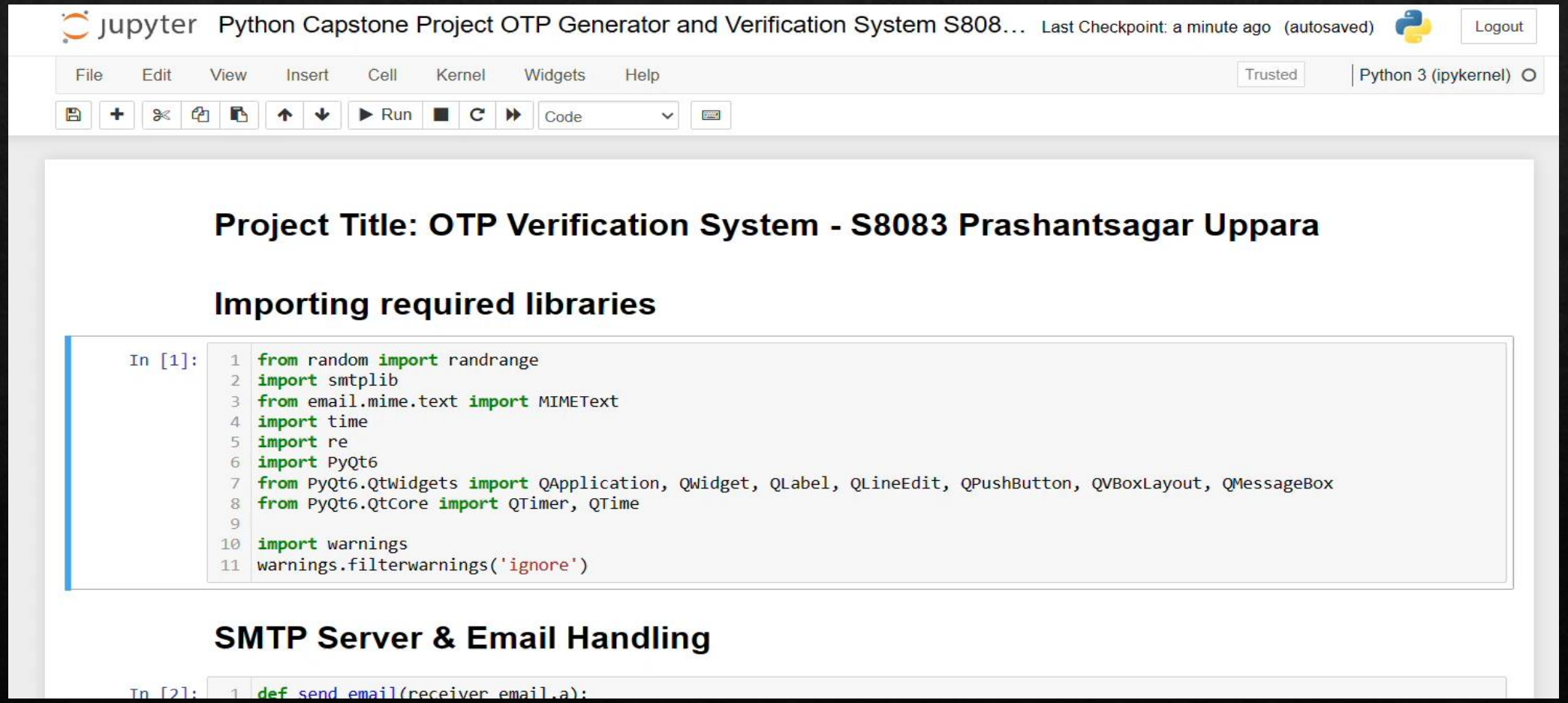


GUI (OTP verification form)



Play the above video to check working

Code Snippet



Play the above video to check the code

Questions

Thankyou