

**EMAIL SENDER APPLICATION WITH SMTP**

**Prepared For:**

**Sir Muhammad Usman**

**Group Members:**

**Ahsan Ali 22K-4036**

**Muneeb Ur Rehman 22K-4025**

**Department of Artificial Intelligence**

**FAST-NUCES, KARACHI**

**1. Motivation**

The motivation behind this project stems from the need for a user-friendly, lightweight email client that integrates both SMTP and IMAP functionalities. Traditional email clients often come with excessive features that complicate the user experience. By developing a Python-based application with PyQt5, we aim to provide a streamlined solution for sending and receiving emails while maintaining simplicity and efficiency. This project also serves as an academic exercise to explore the integration of networking protocols (SMTP/IMAP) with GUI frameworks.

**2. Overview**

**2.1 Significance of the Project**

- **Practicality:** The application simplifies email management by combining sending (SMTP) and receiving (IMAP) features in one interface.

- **Academic Value:** Demonstrates the use of Python libraries (‘smtplib’ , ‘imaplib’, ‘email’) and PyQt5 for GUI development.

- **Impact:** Offers a lightweight alternative to bulky email clients, suitable for users with basic email needs.

**2.2 Description of the Project**

The project consists of two modules:

- **SMTP Client:** Allows users to send emails with attachments via a graphical interface.

- **IMAP Client:** Retrieves and displays the latest emails from the inbox, with support for saving attachments.

Both modules are built using PyQt5 for the frontend and Python’s standard libraries for backend functionality.

**2.3 Background of the Project**

- **SMTP/IMAP Protocols:** Fundamental protocols for email transmission and retrieval.

- **PyQt5:** A cross-platform GUI toolkit for creating desktop applications.

- **Security:** Utilizes app-specific passwords (e.g., Gmail’s app passwords) for secure authentication.

**2.4 Project Category**

This is a **Product-based** project aimed at delivering a functional email client for end-users.

**3. Features / Scope / Modules**

**Key Features:**

**1. SMTP Email Sender**

- Send emails with attachments.

- Configure SMTP server settings (e.g., Gmail: `smtp.gmail.com`, port 587).

- Validation for email formats and server connectivity.

**2. IMAP Email Viewer**

- Fetch and display the latest 10 emails from the inbox.

- Decode email headers (subject, sender) and bodies (plain text).

- Save email attachments to a local directory.

**3. User-Friendly GUI**

- Clean, intuitive interface with PyQt5 widgets.

- Responsive design with error handling and status messages.

**4. Security**

- Supports app passwords for secure login.

- Encrypted communication (TLS for SMTP, SSL for IMAP).

**4. Project Planning**

**Gantt Chart:**

| Task | Week 1 | Week 2 | Week 3 | Week 4 |

|------------------------------|-----------|------------|-----------|------------|

| Research & Design | ✓ | | | |

| SMTP Implementation | ✓ | ✓ | | |

| IMAP Implementation | | ✓ | ✓ | |

| GUI Integration | | | ✓ | ✓ |

| Testing & Debugging | | | | ✓ |

- **Responsibilities:**

- **Ahsan Ali**: SMTP module, GUI design.

- **Muneeb Ur Rehman**: IMAP module, error handling.

**5. Project Feasibility**

**Technical Feasibility**

- Achieved using Python’s standard libraries and PyQt5.

- Risks: Dependency on third-party servers (e.g., Gmail’s SMTP/IMAP policies).

**Economic Feasibility**

- Zero development cost (open-source tools).

- Low operational cost (runs on standard hardware).

**Schedule Feasibility**

- Completed within 4 weeks, adhering to the planned timeline.

**6. Hardware and Software Requirements**

- **Hardware:** Any system with Python 3.x support.

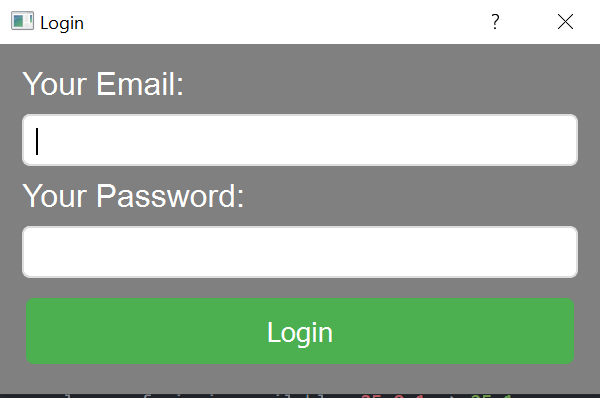
- **Software:**

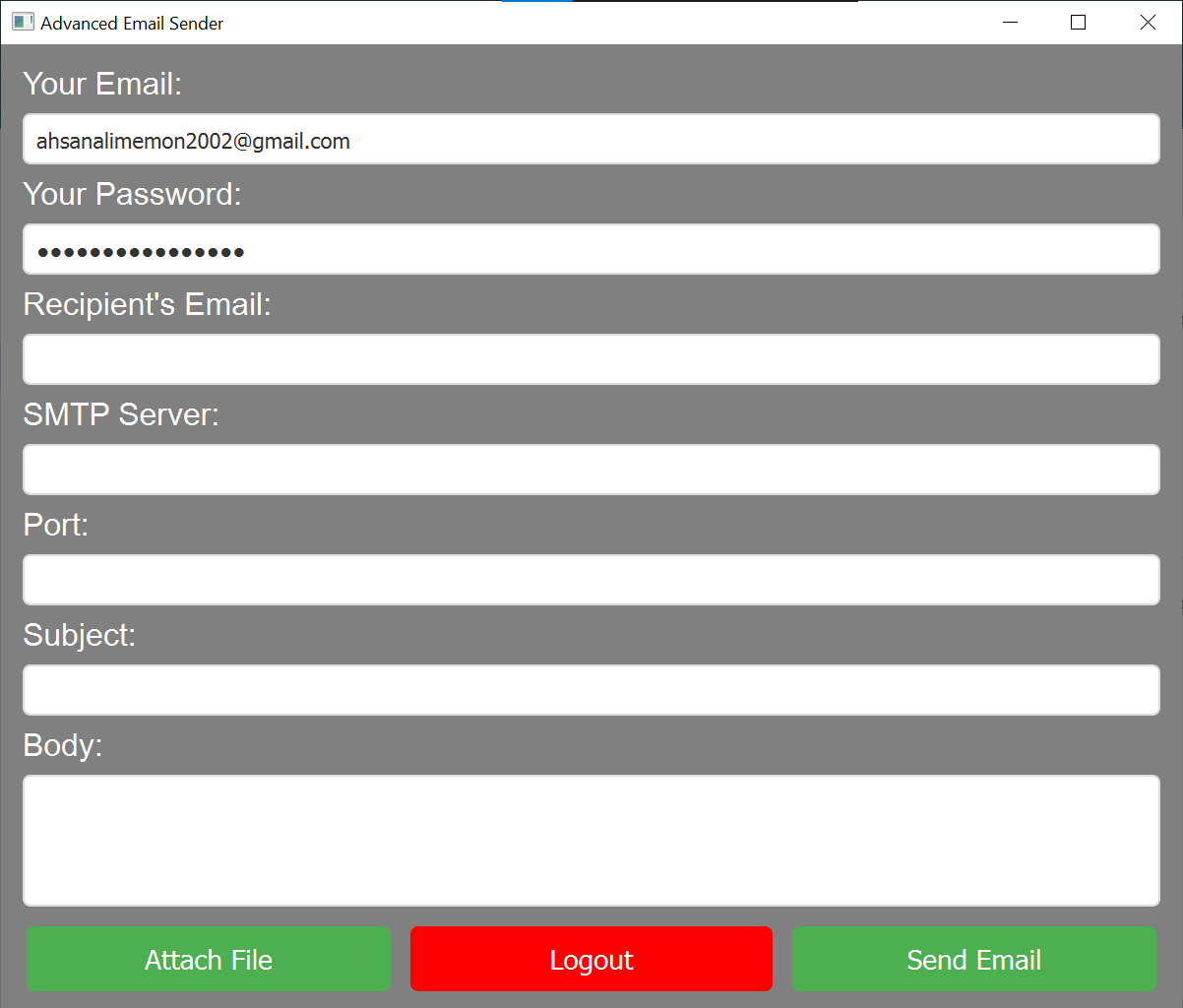
- Python 3.x

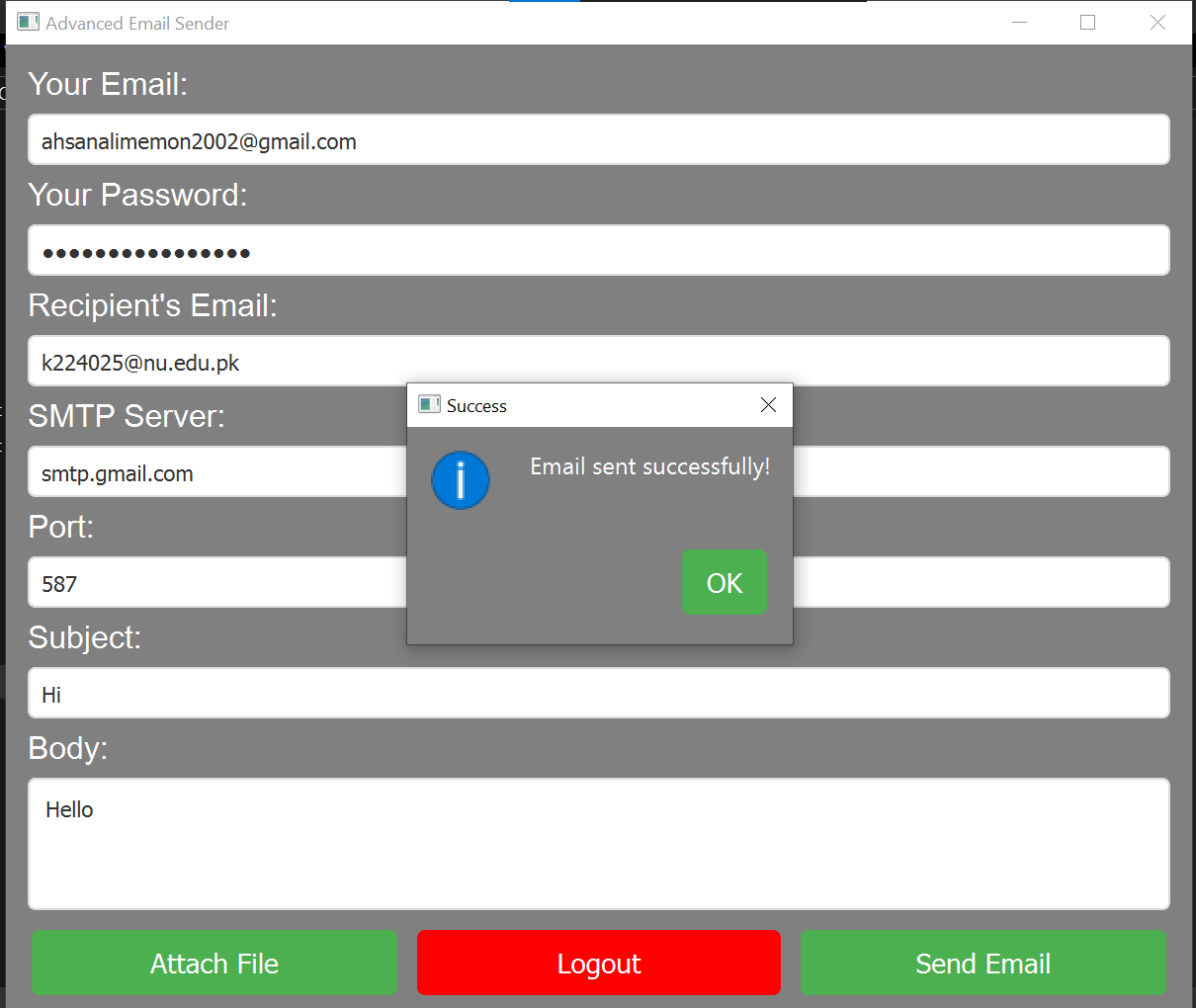
- PyQt5 (`pip install PyQt5`)

- Libraries: `smtplib`, `imaplib`, `email`, `socket`, `re`, `os`.

**7. Diagrammatic Representation of the Overall System**

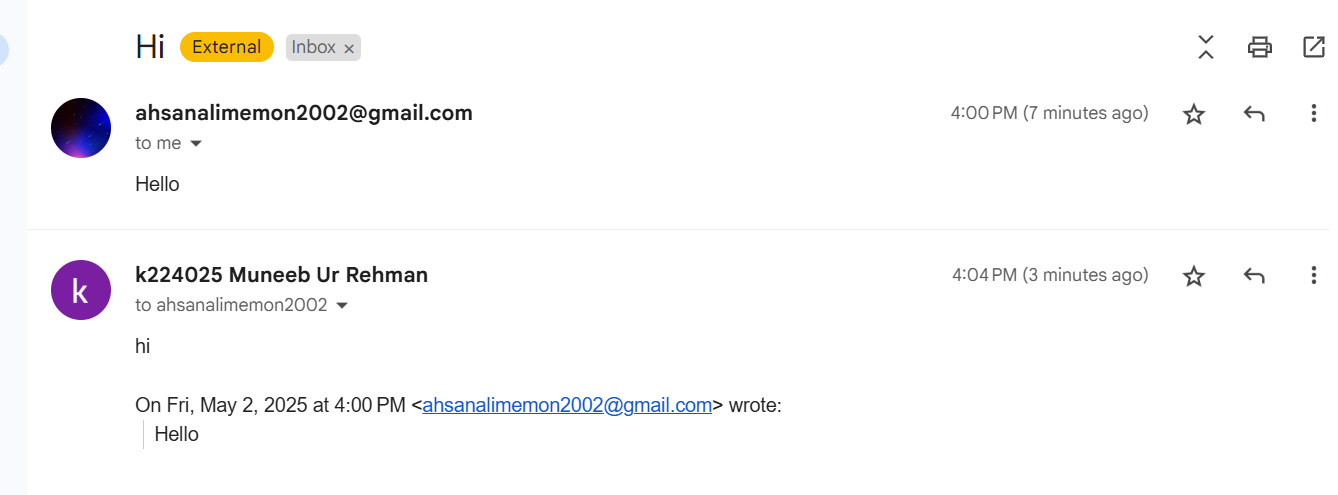
****

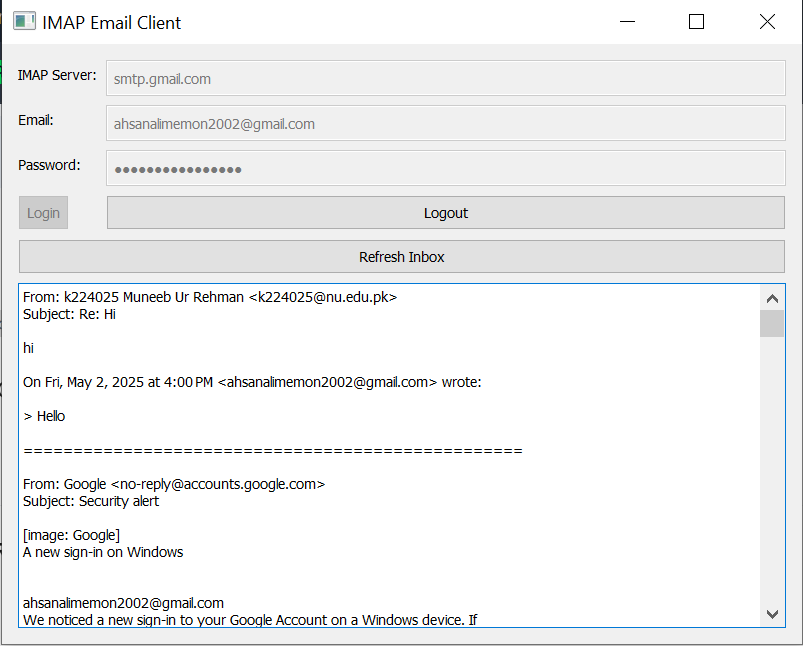
****

****

****

**IMAP:**

****

****

**8. References**

[1] PyQt5 Documentation. Riverbank Computing. https://www.riverbankcomputing.com/static/Docs/PyQt5/

[2] Python `smtplib` Documentation. https://docs.python.org/3/library/smtplib.html

[3] Python `imaplib` Documentation. https://docs.python.org/3/library/imaplib.html

[4] Python `email` Module Documentation. https://docs.python.org/3/library/email.html