Individual Report: Heju

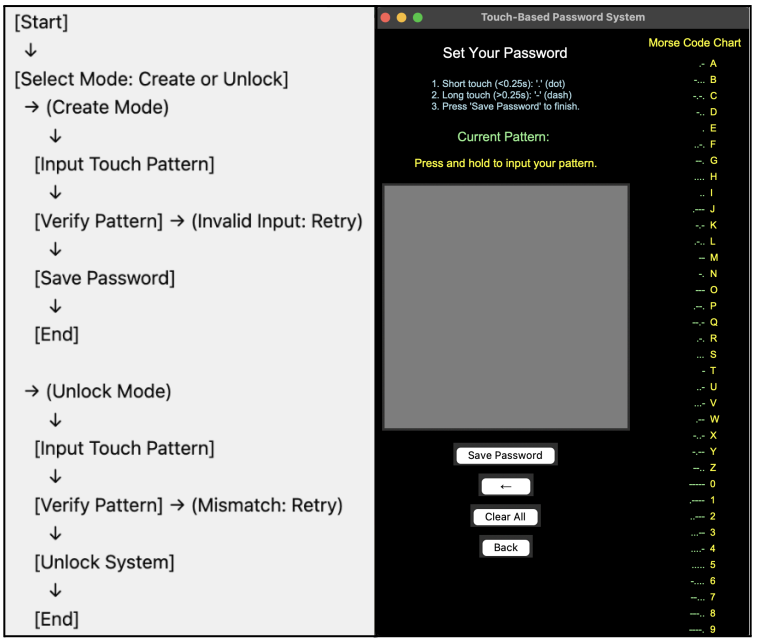
The project, *"Touch-Based Morse Code Password System: Development and Data Visualization,"* involved designing an innovative system that enhances password security through touch-based inputs. As a key contributor, my primary focus was on developing this System.

My responsibilities included designing and implementing the system's core functionalities, ensuring seamless user interaction, and integrating features for data logging. This work required attention to technical detail and user experience to create a robust and reliable system that effectively met project objectives.

Role and Responsibilities

As a key contributor to the project, I focused primarily on **developing the Touch-Based Morse Code Password System**. My responsibilities included designing and implementing the system's core functionality, ensuring seamless user interaction, and integrating features for data logging.

Key Contributions



The system development phase focused on designing and implementing the GUI for the Morse Code Password System using Python and the Tkinter library. Core features included real-time touch detection to distinguish between short (.) and long (-) inputs based on touch duration, password creation and unlocking mechanisms, and error handling with feedback messages to guide users during input. These features ensured the system was user-friendly and functionally robust.

Integrated data logging was a crucial aspect of the project, as it involved implementing a system to log user interactions accurately. The focus was on touch durations and input patterns, ensuring data accuracy and consistency. The collected data included success and failure rates, which were analyzed as part of the data visualization process. This approach ensured the data was accurately captured and seamlessly logged for future analysis.

Collaboration with Sean played a key role in optimizing the data visualization and analysis process. Together, the team ensured that the logged data met the required format and structure. This involved aligning the CSV structure with key variables, such as touch durations and input types, and generating a file named "touch_durations.csv" for further analysis. Sean's feedback enhanced the system's usability and efficiency, ensuring the project met its goals effectively.

touch_durations

Timestamp	Touch Duration	Input Type	Success/Fail

(Your touch duration results from the program will be recorded on the ‘touch_durations.csv’ file like this left chart)

Support for data collection included providing participants with clear instructions and system training to ensure accurate interactions. Troubleshooting assistance was offered during initial trials to address user

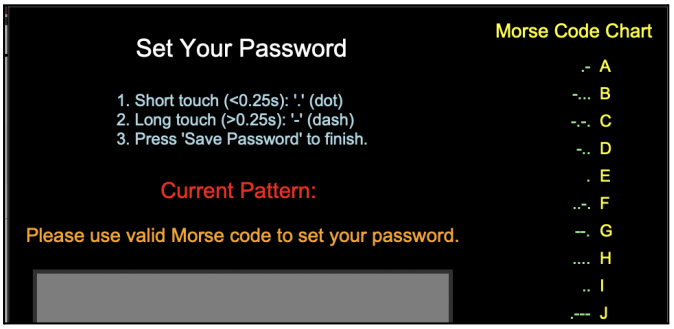
issues, and the system was refined based on feedback. Additionally, guidance was provided throughout the data collection process, ensuring smooth and reliable user interaction with the system.

Challenges and Solutions

```
def stop_timer(self, event):
    elapsed_time = time.time() - self.start_time
    input_type = "short" if elapsed_time < 0.25 else "long"

    self.log_touch_data(elapsed_time, input_type)

    if self.morse_code and (time.time() - self.last_input_time) > 0.5:
        if self.morse_code[-1] != " ":
            self.morse_code.append(" ")

    if elapsed_time < 0.25:
        self.morse_code.append('.')
        self.action_label.config(text="Short touch recorded (.)", fg="lightblue")
    else:
        self.morse_code.append('-')
        self.action_label.config(text="Long touch recorded (-)", fg="lightblue")

    self.last_input_time = time.time()
    self.update_display()
```

One major challenge was real-time touch detection, which required a reliable mechanism to differentiate between short and long touches. To address this, timer-based logic was developed to measure touch durations accurately. Extensive testing with edge cases ensured the solution's robustness, providing consistent performance under various conditions.

Another challenge involved ensuring seamless data logging for compatibility with data visualization processes. This was resolved by collaborating with Sean to structure the logged data in a CSV format, capturing essential details such as touch durations, input types, and timestamps. Additionally, the system was adjusted to reset invalid entries, maintaining data consistency and reliability automatically.



The final challenge focused on integrating user feedback to enhance the overall user experience during password creation and unlocking. To resolve this, on-screen messages were added to notify users about the validity of their inputs and the system's status. This approach provided clear and timely feedback, ensuring a smooth and intuitive interaction process. If you don’t use passwords from the chart, there is going to be an error message like this.

Reflection and Learning

This project provided me with an invaluable opportunity to deepen my understanding of GUI development and data integration. Through collaboration with Sean, I realized the importance of constant communication and feedback in refining and improving a system. This experience underscored the value of teamwork in achieving project success.

Another key lesson was balancing simplicity and functionality. Designing a system that is both user-friendly and functional requires iterative testing and continuous improvements. This process taught me the significance of maintaining usability without compromising the system's core functionalities.

Finally, I gained practical insights into handling real-world data. Implementing a robust data logging mechanism demonstrated how analyzing user behavior data can drive system enhancements and optimizations. This experience emphasized the importance of integrating data-driven strategies into system design and development.

Based on my experience, several potential improvements could enhance future iterations of the project. First, the system's adaptability can be improved by allowing customization of touch duration thresholds to better accommodate various user needs. This would make the system more versatile and user-friendly for a wider audience.

Additionally, incorporating advanced error analysis features could provide users with more detailed and informative feedback. By refining error messages and guidance, users would have a clearer understanding of how to interact with the system effectively and resolve issues independently.

Lastly, integrating AI-based predictions could significantly reduce input errors. By leveraging intelligent algorithms to predict user behavior and correct potential mistakes this would not only improve accuracy but also streamline the overall user experience, making the system more efficient and reliable.

Conclusion

As the lead developer of the Morse Code Password System, I took pride in building a functional and intuitive system that effectively supports data collection and analysis. This project not only enhanced my technical skills but also emphasized the importance of collaboration and user-centric design. By working closely with Sean, we successfully aligned system functionality with data-driven insights, creating a meaningful and impactful project.