Gamification of Application for Diverse Facial Emotion Data Collection

Rob Godfrey 😎 Loelle Lam 🤭
ICS 496 Capstone Project, Spring 2024
Sponsor: Dr. Peter Washington
Project Manager: Jing Zheng

Introduction

- There is a lack of labeled facial expression data beyond the six basic emotions (happiness, sadness, fear, anger, surprise, disgust).
- Recognizing complex human emotion is becoming more important as computer systems continue to evolve - deciphering emotions from facial expressions can help improve human-computer interaction.
- There is a demand for systems that can respond intelligently to natural human emotional feedback (e.g., in retail, a system that can gauge the effectiveness of customer service).

Methodology

- Issue-Driven Project Management (IDPM)
  - An agile project management technique.
  - Leveraged the GitHub project board to track project issues, assign each issue an owner, and aim for a deliverable every week.
  - Scheduled weekly meetings: one with the project manager and one internally with the team.
  - Project board visually represented project status, organizing tasks into "To Do," "In Progress," and "Done" columns.
- Responsibilities
  - Rob: Communications Lead, Requirements Lead, Meeting Scribe
  - Loelle: Technical Lead, Test Lead

The application's game page, where users must match their recorded videos with the correct emojis.

Solution

- The Emoji Challenge
  - Users are presented with emojis and must act out the emojis within the allotted time
  - A quiz section prompts users with their recorded videos and challenges them to associate each video with the emoji they acted out
  - Leaderboard to display top performers
  - Questionnaire for user feedback
- Revamped UX to have a gamified appearance
- Users can refer others to earn a monetary reward
- Users have the ability to download their own videos

Flask B HTML JS CSS aws SQLA

Challenges

- Implementing social media (TikTok, X, Instagram) sharing for user videos
  - We applied for authentication tokens to facilitate user video sharing. However, our application was stalled as it required a Terms of Service and Privacy Policy, leading to the feature being dropped due to the inability to meet approval requirements.
- Converting video file types and adding text overlay to video
  - We had working solutions on our local machines, but when we tried to use the code on our server, we ran into issues. We ended up having to rewrite the code for the server environment.

Takeaways

- HTML Templating
  - We learned how to structure and design web pages using HTML templates. This included creating reusable components and layouts to maintain consistency across the website.
- Flask
  - Working with Flask provided us with insights into building dynamic and interactive web applications using Python. We acquired knowledge in routing, handling requests and responses, and integrating database functionality.
- AWS
  - Accessing S3 and EC2 via their respective APIs allowed us to learn more about third-party integrations.