Group 2
Heather Wright
Emma Holden
Laura Jiminez-Hernandez
Danielle Parker-Jones
Xinyun Zhuo
Adaeze Eke

CODE FIRST GIRLS

Project Report Female Icons Top Trumps

August 2023

Table of Contents

Introduction

Aims and Objectives

The group collaborated to develop a card battle game called 'Female Icons Top Trumps', which, simply put, is a deviation from the popular game 'Top Trumps'. Through research, we understood that this popular game is not readily available online to play interactively. To address this gap, we developed and created a unique online platform to provide this, and infused the game with a feminist theme to add a unique dimension to it.

We have created a fun and interactive game that can be played online. Top Trumps is a well established and loved game around the world, but we've added our own unique and educational spin on it.

We aim to teach the player about influential women from history, whether they are politicians, musicians, scientists, inventors or revolutionaries.

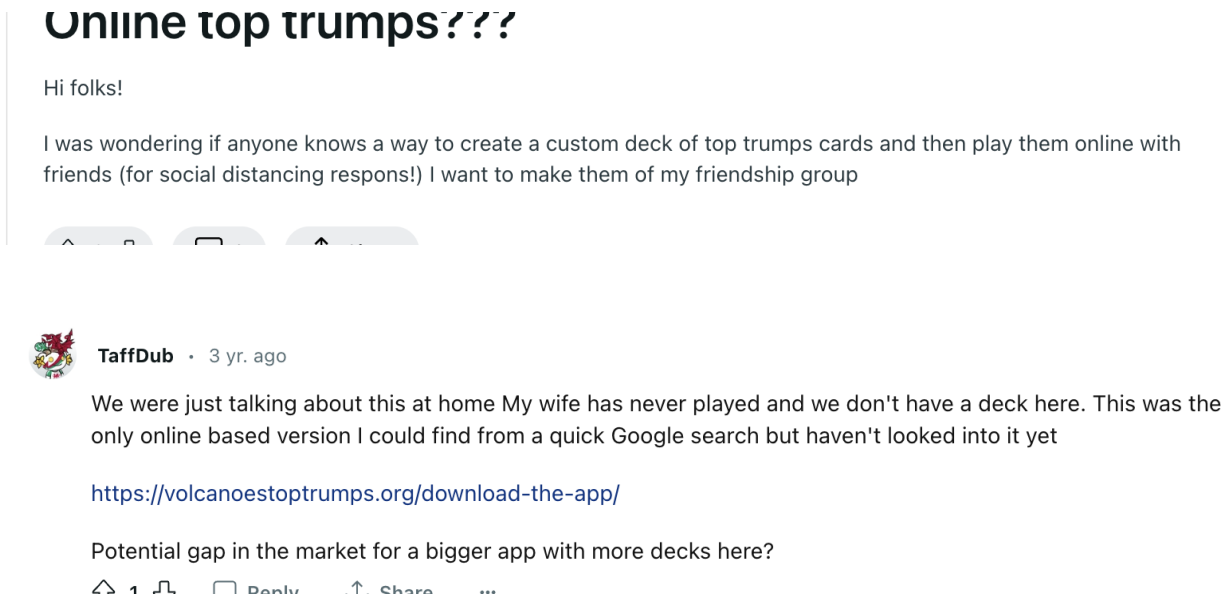
Existing Solution and Research

Share of gaming audiences who ever play games online in the United Kingdom (UK) from 2017 to 2022, by gender

During research, we found that the number of female game players was rising and there was a gap in the market for female-aimed games.

We also found that there were a limited amount of online Top Trump games online, and especially ones that were customisable for the user. We decided

that we could build a function that allows users to input their own characters for future scalability.



Taken from reddit^.

The two main online top trumps games were very niche topics and did not appeal to a wide demographic:

https://volcanoestoptrumps.org/download-the-app/

http://www.supercarworld.com/cgi-bin/setuptrumps.cgi

Background and SWOT

This game serves a dual purpose, aiming to entertain and educate players. It seamlessly merges elements of entertainment, education, and strategy to raise awareness about feminist icons and their impactful contributions, all while engaging players in thought-provoking decision-making challenges. It delivers a distinct and innovative approach to learning and discourse, offering an enjoyable gaming experience that fully immerses participants.

Through its gameplay, the game acquaints players with the remarkable feats and qualities of real-life feminist trailblazers from diverse domains such as art,

academia, activism, science, sports, and politics. Each card symbolises an iconic female figure along with her specific strengths in various attributes. By engaging in the game, players can gain insights into these women and their valuable societal impacts.

The game's primary objective is to empower players by showcasing the remarkable achievements of women across different spheres. It directs attention towards the triumphs of feminist heroes, which can serve as a source of inspiration and enlightenment for players of all genders. Moreover, the in-game challenges could reflect real-world issues demanding attention, effectively fostering awareness and motivating players to take proactive measures.

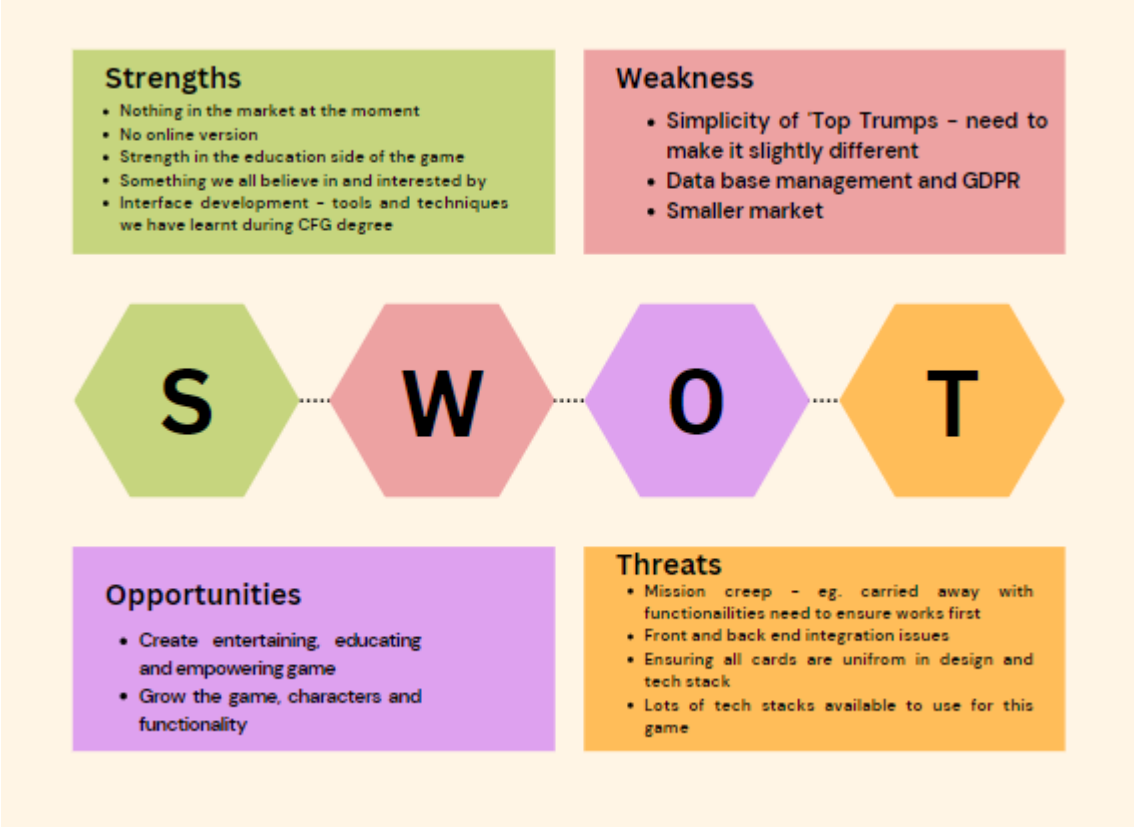
Furthermore, the game prompts players to exercise strategic judgment during gameplay. Optimal card selection, aligned with challenge cards, is key to winning rounds, fostering critical thinking and decision-making proficiencies.

By intertwining educational content with gaming dynamics, the game delivers an avenue for educating players about feminist pioneers in an engaging and interactive format. It fuels curiosity, encouraging players to delve into and appreciate the strides made by individuals who have impacted society. The interactive nature and unpredictability of outcomes also contribute to the overall enjoyment factor.

The game boasts a diverse ensemble of feminist revolutionaries, therefore advocating the notion that heroes and role models can emerge from diverse backgrounds and realms of expertise. This has the potential to dismantle stereotypes and foster inclusivity.

We hope that the game will spark discussions around feminism, women's contributions, and societal hurdles that can be sparked by the game's content. Players may find themselves motivated to embark on further research into the profiled heroes, their accomplishments, and the adversities they encountered.

SWOT:



Roadmap

In order to achieve the goals and complete our project we had to ensure work was complete in a specific order. In order to do this we divided up tasks whilst mapping out a roadmap. The group methodology we decided that would be most suitable for this is the 'Kanban' process. Due to the time restraints and us wanting to be as efficient as possible whilst also ensuring we could produce our best possible work we felt that this was the most viable option.

Using Kanban to implement a pull system helped us to ensure accountability, transparency, and collaboration across the team – leading to better collaboration, less waste, and a predictable delivery for the project.

Female Icons Top Trumps Roadmap

	RESEARCH	DELEGATION	CRITERIA	ADDITIONAL NOTES
PHASE 1	• Pygame • How to create games using python • Top Trumps game understanding (online or physical)	• All • Heather is scrum master	Research phase Free flow/brainstorm.	The aim of this phase is to ensure that everyone is aligned and has a good understanding of the game we are trying to build and come up with ideas/ways we could build this.
PHASE 2	• API • Connecting Python file to SQL database • Creating a card system in python • Allocation of cards for player vs computer • Creating a comparison function to compare attributes of 2 selected cards • We all chose 4 female icons that we admire to use in the game • Front end work to deisgn the card	• Danielle - API • Laura & Adaeze - Main Python file • Emma - Front End • Heather - Project Report & SQL Database • Xinyun - Connecting Python file with SQL database	Create a first draft for the game.	This phase allowed us to visualise what the game will look like and was a great starting point. Because we agreed this would be a first draft, it took the pressure off trying to get it 'perfect'.
PHASE 3	• Test files • Use of key OOP principles and libraries • API endpoints • Decorators • Recursive functions • Regression tests	• All • Regroup to delegate new tasks to each member dependant on where they are upto with tasks from phase 1.	Second draft Working through the checklist to ensure each criteria is met.	The aim of this phase to build upon the first draft of the programme. We will work through the checklist to ensure that each criteria is met. Some elements we have not yet had the lesson on so we agreed to add these at a later date.
PHASE 4	• Class, methods and variables • Decomposition of algorithms • Code layout consistent • Source tree directory structure • File organisation • Exception handling • Unit Test cases • API • OOP • Libraries	• The whole team together.	Quality Control. Getting presentation ready.	The whole group will meet together to read through our code, test that it works and make any necessary improvements. We will also finalise the presentation and decide who presents which parts.

Specifications and Design

The game adheres to a loop-based structure, where rounds continue until either the player's deck or the challenge deck runs out of cards. During each round, the player encounters a challenge card taken from the challenge deck, serving as their opponent. After the player designates a hero card and selects an attribute for the challenge, the game proceeds to compare the player's hero card attribute value against that of the challenge card. The round is won when the player's card attribute value matches or surpasses the challenge card's value.

Employing the principles of object-oriented programming, our code incorporates comprehensible classes, promoting ease of comprehension for other team members and instructors. Each card's distinct attributes and scores are stored within a dictionary.

To facilitate the card allocation process for each player, we have implemented functions such as `fetch_quote()`, `shuffle()`, and `generate_player_cards()`. Rigorous regression tests and unit tests have been taken to validate the functioning of these functions.

The API interface employs POST and GET methods to interact with the database, facilitating extraction of the quotes from the Google Knowledge Graph API which is an API that holds various quotes about well-known individuals throughout time. We imported the SQLconnector library to help us connect the database to the main python file.

We opted for MySQL as our database management system, due to our familiarity with its syntax, which we had learnt at the beginning of the CFGdegree program. The database reflects the card and attribute configurations embedded in the primary codebase. To organise our data, we established two distinct databases, namely ‘challenge_cards’ and ‘player_cards’, each equipped with schemas using tables known as ‘Attributes’, ‘Cards’, ‘Categories’ and ‘Card_Attribute_Score’. The separation was vital in ensuring that the cards allocated to computer players and human players remained distinct. It was imperative that no two players possessed identical cards, as this would disrupt the game's functionality.

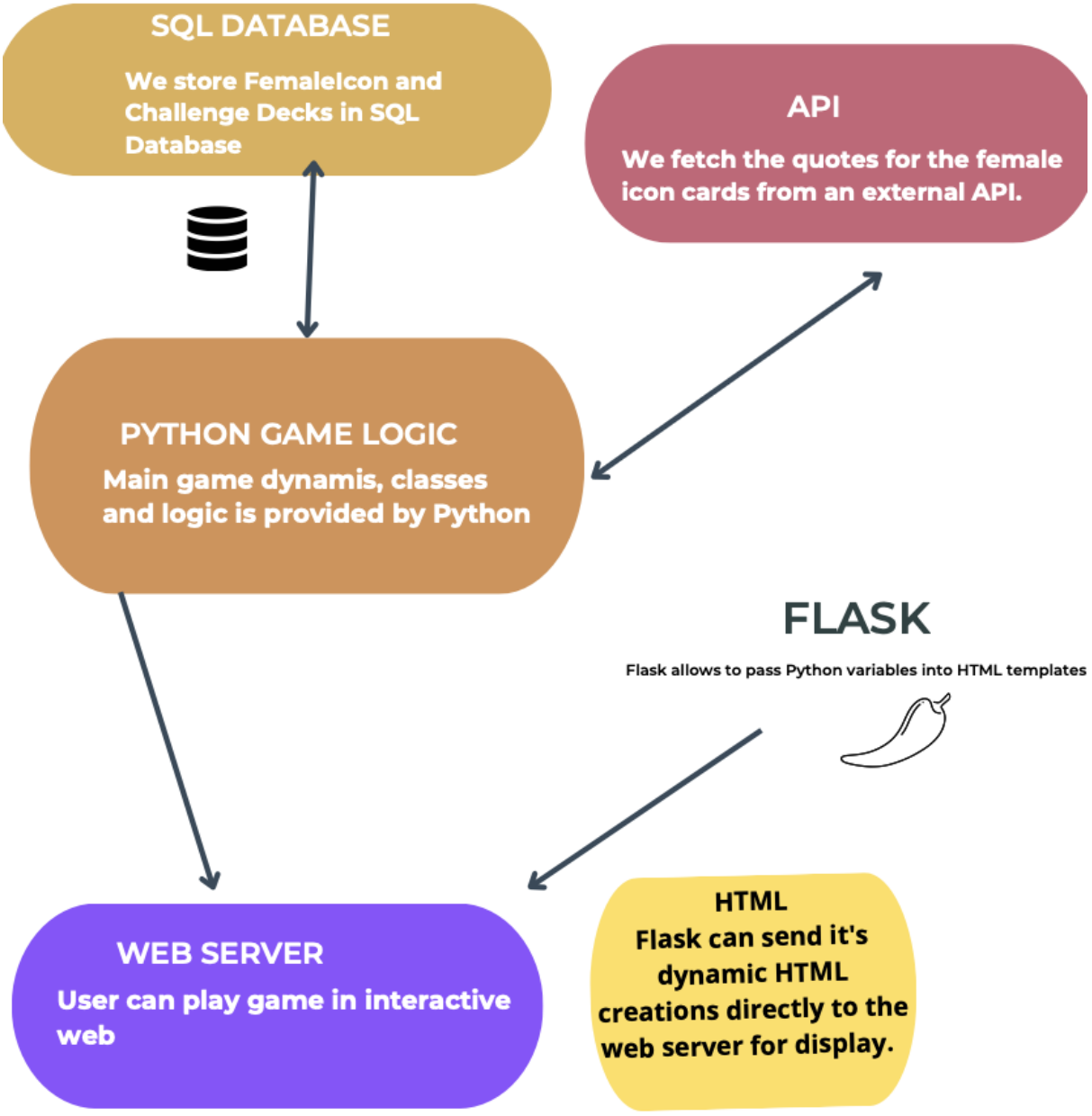
The columns within these tables were designed to specify the attributes of the stored data. We employed primary keys and foreign keys to safeguard the integrity of data relationships across tables.

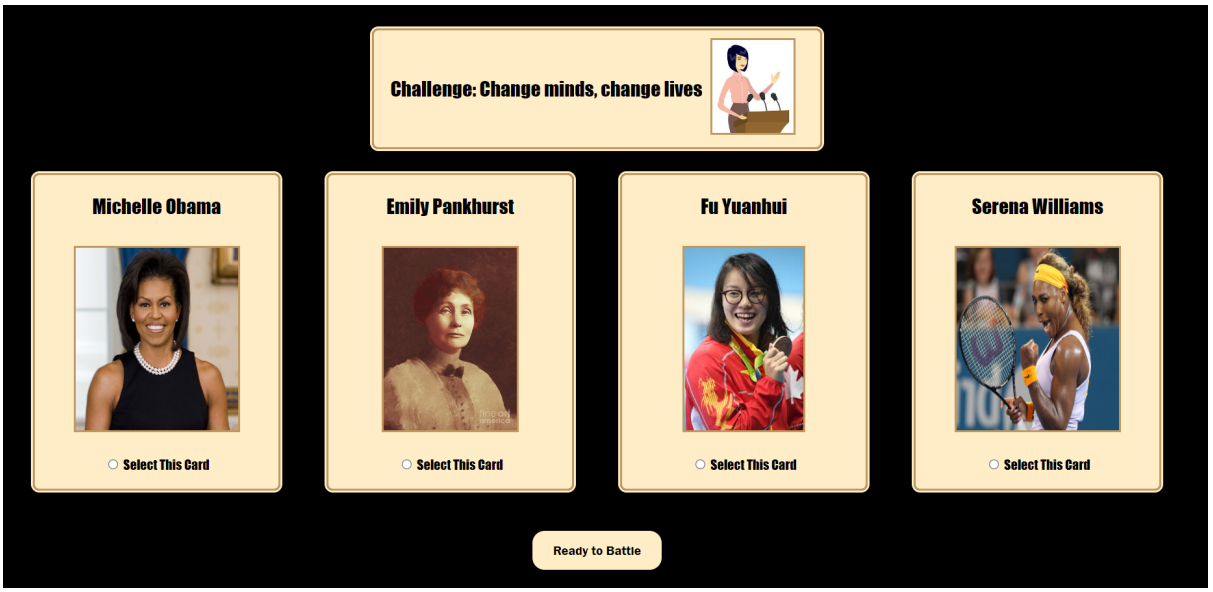
Below, you'll find architectural diagrams that explain our design framework. The utilisation of the Flask library in crafting our web application gave us flexibility and control over design elements. This enabled us to tailor libraries and frameworks to precisely align with our unique project requirements. Notably, Flask served as the foundation for our webpage, seamlessly integrating with SQLalchemy.

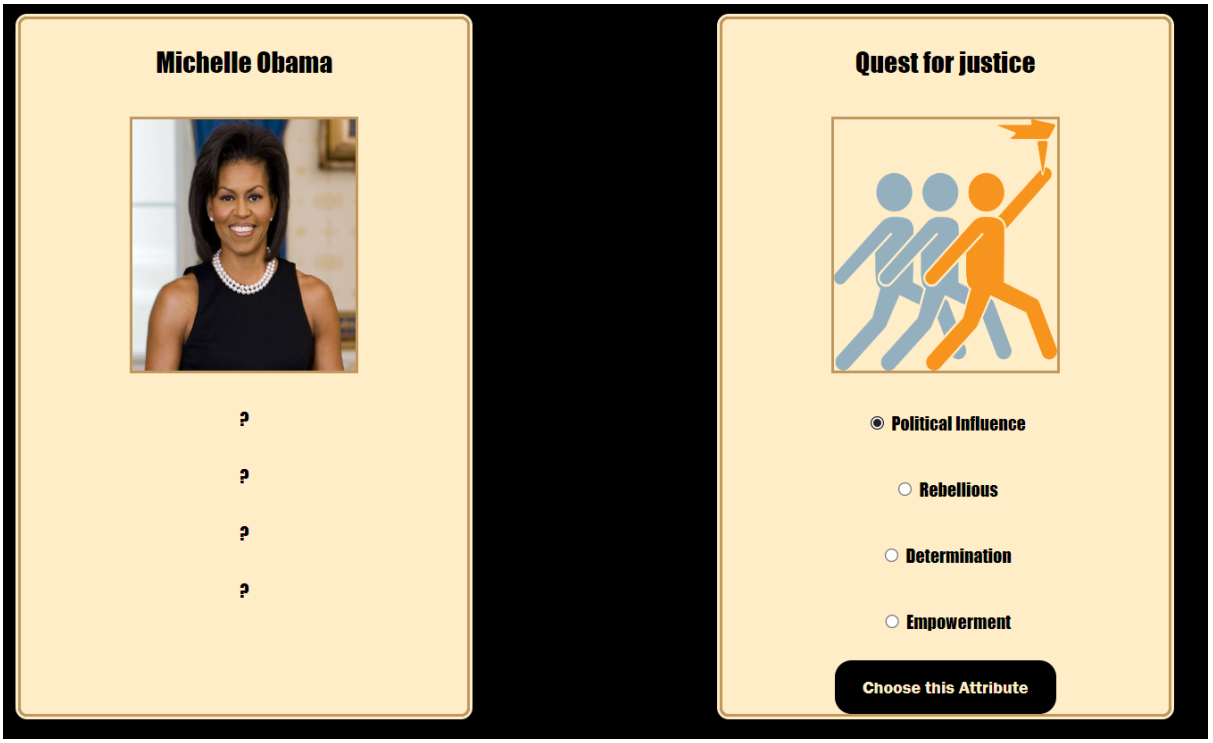
Recognising the importance of user interface (UI) and user experience (UX) design, we prioritised these aspects to enhance engagement for our end users, the players. Our objective was to ensure that the game's rules were conveyed clearly, and that the design adhered to established patterns, facilitating ease of navigation and avoiding confusion among players.

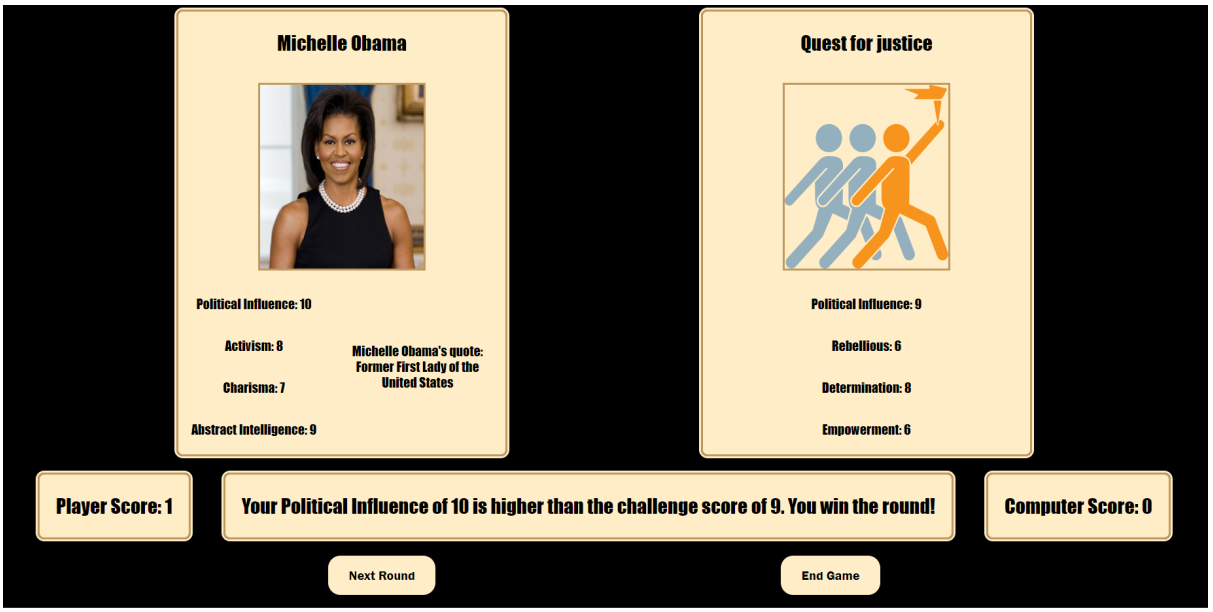
We carefully selected easy-to-read fonts, adjusted text size and colour to direct users' attention to critical content, and curated an inviting color palette. Additionally, we incorporated imagery to captivate users, establishing an engaging game brand that would resonate with and bring joy to our player community.

ARCHITECTURE DIAGRAM

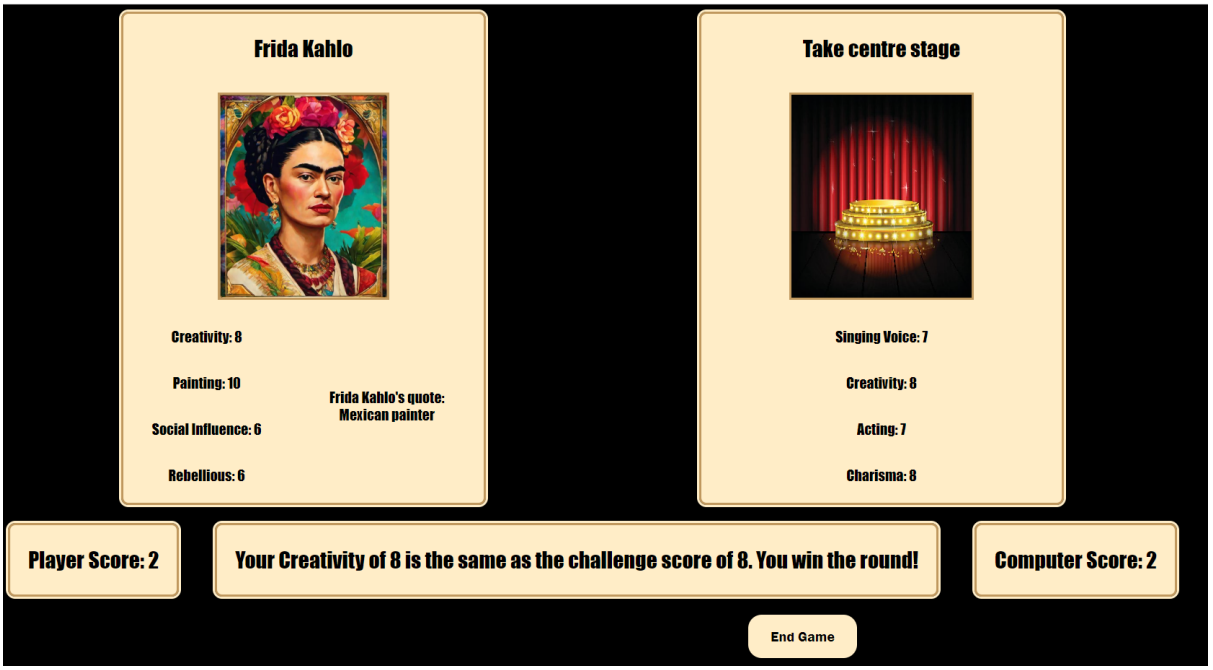


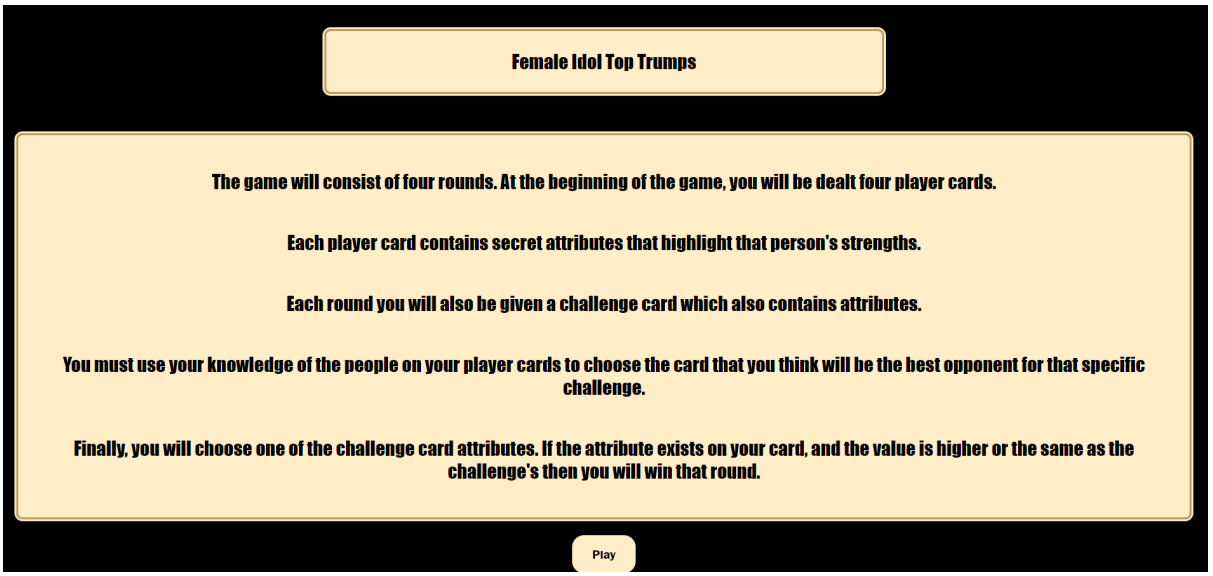

Challenge: Change minds, change lives
Michelle Obama
Select This Card
Emily Pankhurst
Select This Card
Fu Yuanhui
Select This Card
Serena Williams
Select This Card
Ready to Battle


Michelle Obama
?
?
?
?
Quest for justice
Political Influence
Rebellious
Determination
Empowerment
Choose this Attribute


Michelle Obama
Political Influence: 10
Activism: 8
Charisma: 7
Abstract Intelligence: 9
Michelle Obama's quote: Former First Lady of the United States
Quest for justice
Political Influence: 9
Rebellious: 6
Determination: 8
Empowerment: 6
Player Score: 1
Your Political Influence of 10 is higher than the challenge score of 9. You win the round!
Computer Score: 0
Next Round
End Game


Game Over!
Computer Score: 2
Player Score: 2
It's a tie! The battle was intense, but there's still more to conquer!





Implementation and Execution

The initial session involved brainstorming to explore preliminary concepts. A majority, exceeding 50%, of the team expressed a desire to pursue a gaming direction, influencing our choice. In this meeting, we also designated Heather as the scrum master. It was collectively decided that we would mull over ideas during the weekend and reconvene on Monday to share our thoughts. During this discussion, the notion of creating a set of Female Icons Top Trumps emerged, receiving enthusiastic feedback, particularly fitting as our group comprises exclusively female participants.

After reaching a consensus on the Top Trumps concept, our first step was to assess the essential elements required for the project. We categorised these elements into "must have," "should have," and "could have" segments.

Subsequently, we allocated responsibilities, focusing on the "must have" elements and individual preferences. Our intention is to review progress weekly and assign new tasks accordingly.

This allocation was intended to lay a robust foundation for the project, with plans to address project criteria such as unit tests and regression tests once this base is satisfactorily established.

With each member of the team initially owning a particular part of the project it allowed each person to focus their expertise and then share their code with the group for review, ensuring the whole team cast their eyes over the work for a united input and group testing. However this did not create a rigid working environment where teamwork was very much at the heart and team members supported on different parts of the project where it was needed.

Each week we would have team meetings to realign and distribute the work accordingly to correlate with individuals expertise, availability and priority, this allowed us to build rapport as a team and function in a smooth and supportive manner. These meeting notes would then be shared on our Slack group channel for reference to look back on, and if there were any actions for particular individuals.

Halfway through the project, we came to the realisation that our note-taking process was inadequate. Our practice involved discussing and mutually agreeing on tasks during meetings, with the Scrum Master then relaying updates through Slack. Unfortunately, this approach proved ineffective as messages often got lost in Slack, and we lacked clear documentation illustrating the stages of our work and what remained to be done.

In our search for a solution, we explored various tools, including Jira and Notion. Ultimately, we reached a consensus to adopt Trello moving forward. This tool offered user-friendly features, including a Kanban board, which enabled each team member to monitor one another's progress.

Regarding project management, we explored Agile, a well-known methodology widely employed in the software development world. Recognising the importance of effective communication within our remote work setup, we embraced Agile as a high-priority strategy for facilitating team collaboration.

Instead of formulating a rigid plan from the outset, we adopted an Agile approach, which afforded us the flexibility to plan responsively, adjusting our plans based on shifting priorities and workloads.

We worked in one-week sprints, during which we convened weekly to allocate tasks, review completed work the following week, and reallocate tasks accordingly. This approach allowed for constructive feedback and the flexibility to adjust team members' roles if individuals felt uncomfortable with certain tasks.

Implementing the Scrum framework within our relatively small team, we designated specific roles, such as Scrum Master, testers, primary Python file development, front-end design, and API development. However, many team members also collaborated across multiple project categories, offering assistance as needed.

Throughout the implementation phase, each team member worked on their own branch and initiated a pull request when their work was ready for review. This process allowed for peer reviews, comments, and suggestions for changes or improvements.

At the end of each sprint, we conducted a retrospective meeting to assess the work accomplished and provide feedback. This also served as an opportunity for team members to seek assistance if they encountered challenges with specific aspects of the project.

As our project deadline came closer, we initiated daily stand-up meetings at 6 pm, just before our class sessions. We deemed this necessary due to the frequent daily changes and the heightened workload as we aimed to ensure the project met high standards of completion.

Testing and Evaluation

Our testing approach included regression testing, exception handling, and user testing. Utilising GitHub, we documented each pull request, incorporating comments and reviews, and attaching testing files. We encountered some challenges when merging changes into the primary file, resulting in the unintentional overwriting of finalised code files. Consequently, we had to revert these alterations and, subsequently, opted to make the repository public while securing the primary branch to guarantee safety.

To facilitate testing, we established two distinct files: 'Test_feminist_heroes_class.py' and 'test_main.py'. These files imported the 'unittest' library and drew upon various functions from both files for testing purposes. We concentrated our tests on assessing the win status of both player and challenge cards, with particular emphasis on attribute comparisons to ensure the game's proper functionality. For instance, we

employed the 'def test_choose_cards_route(self):' function to validate the appropriateness of data submitted via forms.

Furthermore, we used the 'Testcase' class from the Flask-Testing library to establish a robust testing environment. In the 'main.py' file, a 'requirements.txt' document was included, enumerating external libraries (dependencies) used in this project alongside their precise versions.

We had issues with testing the /battle route (the page where you pick an attribute from the challenge card to battle) because it was looking for the attribute "game.challenge_card" which did not exist. This attribute was not created upon game initialisation - instead it was created during the game. When trying to test only the battle route function in isolation, this attribute did not exist as the previous parts of the game hadn't happened. We solved this by making the game create the challenge_card attribute upon initialisation and giving it one of the challenge cards as a placeholder.

Exception handling was employed to proactively address potential errors during program execution. We integrated assertions to account for all exceptions. In sections of the code where we anticipated exceptions, we wrapped the code

```
try:
    response = requests.get(service_url, params=params)
    data = response.json()
    if 'itemListElement' in data and len(data['itemListElement']) > 0:
        item = data['itemListElement'][0]['result']
        if 'description' in item:
            return item['description']
    return None
except Exception as e:
    print(f"Error fetching quote: {e}")
    return None
```

within a 'try' block. For instance, when fetching quotes from the API, it would generate an error.

Additionally, we rigorously conducted regression testing, with each pull request undergoing review from team members. Any identified bugs or discrepancies affecting the main branch prompted necessary adjustments, which were subsequently committed. The code was merged into the main branch only after we were satisfied with its performance, and this iterative process persisted throughout the project.

We also considered certain system limitations that could impact the program's scalability and usability in the future:

- Data storage/memory: We contemplated whether the program could effectively manage a million users.
- Compatibility with web browsers: We assessed whether the program would be compatible with a variety of web browsers to ensure broad accessibility.
- Response time: We evaluated whether the system would exhibit adequate speed to engage users effectively.
- UX/UI design: We verified whether the design adhered to essential accessibility regulations.
- Compliance: We reviewed whether the program conformed to industry standards that might impose constraints on its functionality.

Summary

In conclusion, our journey to create an online version of 'Top Trumps' with an educational twist has been rewarding and we believe we set out what we hoped to achieve. We started this project with a mission to fill a gap in the online gaming market, provide entertainment, and raise awareness about influential women in history. Through research and a well-thought-out roadmap, we have achieved these goals and more.

Our game not only offers a fun and interactive experience but also serves as a valuable educational tool. It introduces players to a diverse array of feminist icons, shedding light on their remarkable achievements and contributions across various fields.

Our team's approach to project management, adopting the Kanban process and Agile methodologies, ensured that we worked efficiently, collaboratively, and with flexibility. It allowed us to adapt to changing priorities and challenges, ultimately leading to a successful outcome.

The technical implementation of the game, with its object-oriented programming principles, API integration, and database management, reflects our commitment to delivering a reliable product. We encountered challenges along the way, but our dedication to rigorous testing and continuous improvement ensured that the final product met high standards of quality.

We envision our game sparking discussions around feminism, women's contributions, and societal issues. We hope it serves as a source of inspiration for players of all genders and encourages them to explore the stories of the heroes in the game. Additionally, our commitment to inclusivity through diverse representation has the potential to challenge stereotypes and promote a more inclusive society.

In conclusion, the finished product was only achievable due to teamwork, innovation, and a shared commitment to creating something meaningful yet fun and interactive.