Final Report: Website for Single Player Mode in Battleship

Katie Schaumleffle

Oregon State University

CS469: Summer 2023

Plan Version 1.0

Introduction:

In this project, I built a Battleship Web App, which turned into a digital adaptation of the well-known two-player strategy board game "Battleship." This rendition reimagines the game, offering players an engaging solo experience where they can put their strategic talent to the test against an AI opponent. Tailored for individuals who prefer competitive and tactical gameplay, the Battleship Game provides an immersive virtual environment for the traditional naval battles.

In this iteration of Battleship, users step into the shoes of a skilled naval commander, embarking on a thrilling one-player game against an AI-controlled opponent. The traditional game has been transformed into a dynamic digital platform, enhancing the interactive experience while preserving the essence of the classic game.

Combining the timeless appeal of the Battleship board game with digital technology, this version enables users to indulge in strategic battles at their convenience. Whether craving mental stimulation during short breaks or seeking the excitement of strategic challenges, the Battleship Game caters to a diverse audience, offering both entertainment and intellectual engagement in a single interactive site.

Users can create their own, unique usernames. Once logged in, they click the “start” button at the top of the screen, and can continue with placing their ships on the board. Once all ships have been placed, then they can shift into “play” mode, where they start attacking their opponent, and strategically guess where their opponents’ ships are, until either the AI or the user has sunk their opponent’s ships and won the game.

Motivation:

The motivation for creating this Battleship Game was driven by the desire to blend the nostalgia of the traditional Battleship game with modern interactive technology. The motivation extended beyond just recreation; the goal was to create an immersive experience where players could strategize, engage in battles, and relive the entertainment of the original game. Additionally, the project provided a valuable opportunity to enhance my programming skills and gain experience in web application development and critical thinking skills required for working through the necessary logic for each step of development. I was particularly interested in the algorithms and logic required to create this game.

Tools and Technologies:

There were several tools and technologies utilized in order to make this project a reality:

1. **Python:**
    a. Purpose: The backbone of the game's development, Python was used to craft intricate game logic and manage backend functionality. Python is a versatile and powerful language which helped with the implementation of the game logic, AI opponent behavior, game status, ships, moves, etc. I used Python to implement the

start_game method, make_move method, ship_placement method, login method and create_user method.

2. **IDE: Visual Studio Code**
    a. Purpose: Visual Studio Code served as the IDE I used for my project. It provided a user-friendly interface and a wide range of extensions and plugins that support efficient coding, debugging, and project management. I have used VS Code a lot in the past, and is the IDE that I am most comfortable with interacting with.

3. **Flask:**
    a. Purpose: Flask was utilized as the web framework to power the creation of the Battleship Game's web application. By harnessing the capabilities of Flask, I was able to streamline the development process by implementing efficient routing and resource management. Flask's robust architecture provided the ideal foundation for building a responsive and interactive platform that seamlessly brings the game to life for users. Since I was already a little familiar with Flask, I chose this framework for its versatility and its capabilities, allowing me to focus more on the game's features and functionality.

4. **HTML/CSS:**
    a. Purpose: The visual and interactive facets of the Battleship Game were shaped by HTML for structuring the content and CSS for styling and visual appeal. It was utilized for designing and styling the UI of the website and was the main technology I utilized when creating the front-end of the website.

5. **JavaScript:**
    a. Purpose: JavaScript was a pivotal element in creating dynamic interactions within the Battleship Game. By integrating JavaScript, I was able to elevate the user experience, enabling real-time updates and responsive behaviors on the frontend. This scripting language empowered me to introduce fluid gameplay mechanics, such as dynamic ship placement, interactive gameplay grids, and instant feedback on game moves. JavaScript's versatility and power allowed me to bridge the gap between user actions and server interactions, ensuring a seamless and engaging user journey throughout the game.

6. **Git/GitHub:**
    a. Purpose: The use of Github ensured seamless version control throughout the project. It was used for tacking changes and ensuring codebase integrity throughout the development process. It even allowed me to roll back to a previous version at one point, when I needed to.

7. **AWS EC2 Instance:**
    a. Purpose: The AWS EC2 (Elastic Compute Cloud) Instance played a critical role in hosting the Battleship Game's web application. By deploying the application on an EC2 instance, I ensured that the game was accessible to users from anywhere, at any time. This scalable and flexible cloud computing resource provided the necessary computational power and storage to accommodate user traffic while maintaining consistent performance. The EC2 instance offered a reliable environment for deploying the backend logic and frontend assets of the game, enabling users to engage in Battleship battles without the need for installing any software locally. This integration of AWS EC2 extended the reach of the Battleship Game, offering an accessible and uninterrupted gaming experience to a wide audience. The EC2 instance's reliability and scalability aligned seamlessly with the project's requirements, enhancing the overall user experience.

By harnessing the capabilities of these tools, the project efficiently navigated the development journey. The project benefited from efficient and streamlined development, robust front and backend frameworks, and secure deployment and hosting.

Work Completed:

The culmination of the Battleship Game project resulted in the creation of a fully functional digital adaptation of the classic board game. I followed the instructor-approved plan, which yielded a comprehensive gaming experience that honored the traditional Battleship rules and mechanics. The game's design revolved around players strategically placing their fleet of ships on a grid, followed by turn-based engagements with the AI opponent to sink the adversary's vessels. The overarching goal remained unchanged: to outmaneuver the opponent by tactically selecting attack coordinates and protecting one's own fleet from harm.

All key components essential to delivering an engaging and immersive experience were incorporated into the final product. The grid arrangement, enabling players to position their ships with precision, was carefully implemented. The turn-based gameplay mechanism, a fundamental aspect of Battleship, was integrated to ensure strategic decision-making and heightened suspense. Visual cues such as hit and miss indicators provided real-time feedback, enhancing the gameplay's excitement and keeping players invested in the outcome of each move. Additionally, the graphical elements were carefully crafted to ensure a visually pleasing and intuitive user interface, promoting an immersive gaming experience.

While the final product achieves high functionality and engagement, given more time, opportunities for refinement would be explored. A more visually captivating user interface would be pursued, enhancing the aesthetics to complement the game. The inclusion of a comprehensive stats board, showcasing the user's victories against the AI, the total number of games played, and more, would add an additional layer of depth to the player's experience. Additionally, components would be centered to bolster visual symmetry, and the homepage would undergo a transformation to embrace more colors and graphics reminiscent of the original Battleship game. These enhancements would contribute to creating an even more user-friendly and visually enticing environment, enhancing the overall user journey and satisfaction.

Throughout the development journey, the project's progression remained in alignment with the original plan. The predefined milestones and goals were consistently met, eliminating the need for major deviations from the initial scope. This translated to timely and effective implementation of all core features. The careful management of resources, efficient allocation of tasks, and proactive mitigation of challenges ensured that the project stayed on track. This consistent execution resulted in a successful manifestation of the envisioned Battleship Game, showcasing the culmination of planned features and functionalities.

Although there weren't any large deviations from the original plan, one minor change I had to make at the last minute was to switch from hosting the site on AWS Amplify to using an EC2 instance on AWS. The build command was not working properly in AWS Amplify, and kept giving me build errors. I was able to switch to hosting it on an EC2 instance, and with a little finesse, was able to get it up and running within a couple of hours.

In summary, the completed Battleship Game project serves as a testament to effective project management and thorough execution. By adhering to the approved plan, the digital adaptation is able to capture the essence of the traditional board game, offering players an engaging and immersive gaming experience. The successful implementation of core features underscores the continuous integration of gameplay mechanics, visual elements, and strategic challenges, resulting in a product that provides both entertainment and intellectual satisfaction.

Retrospective:

The execution of the initial plan for the Battleship Game was largely successful, with the final product closely aligning with the envisioned concept. The game's core mechanics, such as a login feature, ship placement, turn-based gameplay, and hit indicators, were successfully implemented, fulfilling the primary objectives set forth in the plan. The timeline and scope of the project remained consistent, resulting in a cohesive and functional digital adaptation of the classic board game.

Several successes emerged during the project's course. The integration of Flask and JavaScript facilitated the creation of an interactive and responsive web application, enhancing the user experience. The thorough implementation of game mechanics and graphical elements ensured an engaging gameplay environment. Additionally, the effective utilization of version control through GitHub enabled efficient code management, contributing to project integrity.

However, challenges were encountered, primarily related to time constraints. While the project stayed within the established timeline, additional time would have allowed for more advanced features, refinements to the user interface, and comprehensive testing. Balancing between frontend and backend development posed occasional difficulties, requiring strategic time allocation.

In future projects, I would approach time management more rigorously, allocating specific time blocks for feature development, testing, and unforeseen challenges. Enhancing the initial planning phase to encompass potential obstacles would streamline development. Further emphasis on user interface design and user experience testing would be incorporated to ensure an appealing and intuitive user journey. Additionally, seeking opportunities for peer reviews and collaboration throughout the development process could lead to more diverse perspectives and refined outcomes.

Conclusion:

In conclusion, this project has resulted in the creation of a single-player Battleship Web App that successfully demonstrates the classic board game within a digital environment. By skillfully integrating technologies such as Python, Flask, JavaScript, and AWS EC2, the project has delivered an immersive and engaging gaming experience for players. The user-friendly interface, complemented by responsive design and an AI opponent, has provided players with an intuitive and strategic gameplay experience. The user is able to create a username and login, place their ships on a game board, and begin attacking the AI opponent. Once there is a designated winner, the user has the option to continue playing another game or not.

Throughout the development journey, the focus remained steady on creating a visually appealing and intuitive user interface, implementing intricate game logic, and ensuring a seamless experience for players. Although I would have liked to enhance the website to make it more visually appealing, I was more focused on the logic and algorithms of creating the game at this point. The fully functional Battleship Web App was successful in large part due to the diligent plan created within the first week. This serves as a testament to the importance of proper planning and knowing when to make necessary adjustments. Overall, I am very proud of the result achieved.