Inspiration

It all began with a creative spark to reimagine the Trojan War using modern technology. Inspired by ancient mythologies and their intricate webs of relationships, we envisioned a platform where Odysseus himself could manage alliances, rivalries, and divine interventions—much like a strategist commanding a battlefield. The concept of TrojanNet, a fictional social network for manipulating the social dynamics of Troy, pushed us to think about how these ideas could take shape in code.

Odysseus' cunning mind and his reliance on both human and divine relationships became our guiding principle. We wanted to build something that combined mythological intrigue and modern programming practices, like object-oriented programming (OOP) and SQL databases to make our ideas scalable and reusable, and interactivity, so users could step into Odysseus' shoes and influence the world around him.

Thus, Odysseus' Control Panel was born—a web application that lets users manipulate loyalty, influence, and trust in a way Odysseus himself would approve of.

What it does

Using object-oriented programming (OOP), and SQL database management the app models relationships between characters, with each having unique attributes. Users can interact with dropdown menus to select characters and actions, and PHP handles the backend logic to adjust values and update the interface in real time. This allows for an immersive experience where users can experiment with the impact of their decisions, mirroring the strategic mindset of Odysseus himself.

In addition to our manipulation panel for Odysseus, our social site includes a Trojan style wordle game that aims to distract those in Troy, enabling Odysseus to have the upper hand in his attack.

How We built it:

- Conceptualization and Design:
    - The initial idea was to create an interactive platform where users could step into the role of Odysseus and influence relationships with other characters from the Trojan War. Our goal was to use modern programming practices to simulate the complex interactions of loyalty, trust, and influence
    - The design process focused on:
        - Modeling characters as dynamic objects with attributes and behaviors.
        - Making the system interactive so Odysseus could modify these attributes and immediately see the effects of his decisions.
- Object-Oriented Programming (OOP)
    - To represent the characters in a flexible and scalable way, we chose Object-Oriented Programming as our approach. This allowed us to:

        - Create a class for characters, each with unique attributes such as loyalty, trust, influence.
        - Encapsulate behaviors such as adjusting trust levels or influence into methods, making the system modular and easy to extend.
- Frontend Development
    - The frontend was designed with user experience in mind. We used HTML and CSS to create the structure of the page, which included:
        - Dropdown menus to allow users to select characters (e.g., Odysseus, Achilles) and actions (e.g. adjust loyalty, trust, influence).
        - Input fields for users to enter numeric values for modifying attributes like trust levels.
        - A dynamic, user-friendly interface that updated immediately after the user submitted their choices.
- Backend Development
    - PHP
        - The backend of the application, built using PHP, handled the logic of modifying and updating the attributes of characters. PHP processes the form inputs submitted by users and adjusts the values in real time based on their selections. Key actions that PHP handles include:
            - Form handling: PHP processes the user input (such as selecting a character and action) and adjusts the corresponding attributes (e.g., trust level, loyalty,influence).
            - Dynamic data updates: After each action, PHP updates the page to reflect the new values of loyalty, trust, and influence.
    - Database and SQL
        - This website uses a database that we implemented using Mr Engel's, our teacher's, server. The database contains each person's name, role, loyalty and influence. We decided to use a database so the website could update instantly and store data based on if the users decide to attack or not. This was also fresh in our mind and familiar to all of us since we just recently learned this.
- Testing and Refining
    - We used an iterative development process to refine the program by starting with a simple system of characters and attributes. We added more complex features like trust levels. We constantly tested the system, ensuring it was intuitive, functional, and engaging for the user.
- Final Touches
    - Site design
        - The site includes 2 main sections. The control panel for Odysseus and our Trojan wordle game. The page essentially has 2 cards that adjust to the size of the user's screen.
    - Logo
        - Color Palette: We selected a rich gold and red color scheme. Gold represents wisdom, power, and the divine favor Odysseus receives from

the gods, while red symbolizes the bloodshed, passion, and intensity of the Trojan War. Together, they create a visually striking and meaningful contrast.
    - The Trojan Horse: The main element of the logo is a stylized representation of the Trojan Horse. Its simple, bold shape captures the idea of strategic deception and cleverness. The horse has complex details demonstrating the complexities of the Trojan social network.

Challenges and Accomplishments

- Modeling Complex Relationships
    - Person classes with the right attributes and behaviors was challenging, especially when trying to model the many layers of influence, loyalty, and trust.
    - We solved this by making a database to organize our information and keeping our classes simple and easy to use.
- Maintaining Real-Time Interactivity:
    - Ensuring that changes to character attributes (like trust, loyalty, and influence) were reflected instantly on the frontend, while keeping the system responsive, required careful handling of form submissions and PHP logic.
- Scalability and Extensibility:
    - Designing the system to easily accommodate new characters and actions without overcomplicating the structure was a challenge. We needed to ensure the system could grow as new features were added.
    - This was also solved by our database, which allows new information to be added easily without adding more complex classes.
- User Interface Design:
    - Creating an intuitive and visually appealing interface that conveyed the complexity of the relationships and actions without overwhelming the user was a balancing act.
    - We used Bootstrap for the base of our design to ensure simplicity and ease of use for Odysseus and other users.

How to Use It:

- Spreading Rumors
    - When clicking on the "Spread Rumors" button, you may affect the loyalty, influence, and trust of each person signed up for TrojanNet.
        - Loyalty: Their loyalty to the city and its army
        - Influence: Their importance, how much influence they have over others
        - Trust: How much they are trusted by the city.
    - In the resulting modal, you must select the action, value, and target of your rumor
        - Action: Whether you will affect their perceived loyalty, influence, or trust

        - Value: How much the attribute will be affected (can be positive or negative)
        - Target: Who the rumor will target
    - Now, all you have to do is spread the rumor! Everything else will be taken care of in the backend.
- City Analysis
    - In addition to spreading rumors, TrojanNet also allows you to analyze if you should attack now or wait.
    - This works by analyzing the loyalty and trust within the city. If either the highest loyalty or highest trust level is below a set level, the city is ready to attack.
- Logosle: Ancient Wordle
    - In addition to having a strategic battle plan, TrojanNet offers a fun alternative to participating in the battle: You can play “Logosle” or Ancient Wordle (because Logos means word). This allows Odysseus to sharpen his mind while preparing for battle, while also distracting any enemies who are also using TrojanNet.