CySim

Client - Professor Doug Jacobson

Brady Schlotfeldt - Backend and Systems Developer
Matthew Daoud - Project Manager
Bailey Heinen - Full Stack Development
Ethan Swan - Full Stack Development
Jacob Boicken - Linux Admin and Systems Developer

Project Vision

- Problem Statement
  - What is a good way to test whether a candidate is fit for a cybersecurity role?
  - How can we help cybersecurity professionals practice real world skills?
- CySim’s Goal
  - Simulate real cybersecurity threats through team-based games and scenarios, from traditional Capture-the-Flag and Red-vs-Blue events to industry-specific threat training.
  - Prepare the next generation of cybersecurity professionals for relevant threats.
  - Provide a safe environment for team members to develop security and collaborative skills.

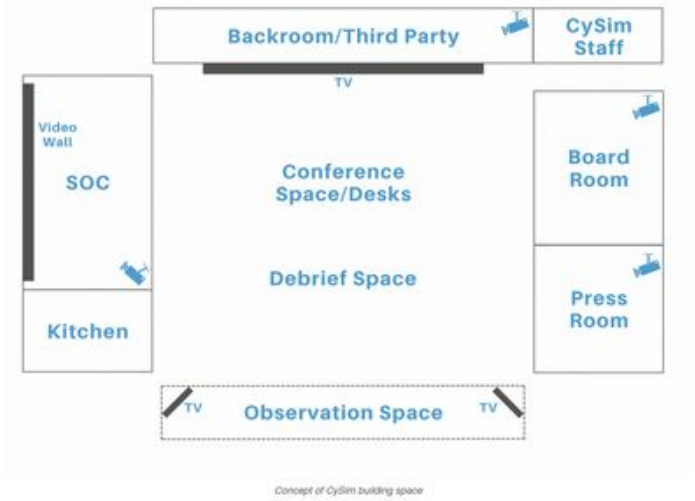
Project Vision Continued

- Who does CySim help?
  - Students and career professionals who want to improve their cybersecurity readiness.
  - Corporations that want test their candidates' security aptitude, or develop their existing teams.
- Which demographics is CySim for?
  - College-aged students with an interest in cybersecurity or a related field of study. Some may have experience from university classes or prior cybersecurity events, but all will be able to participate in events designed to help those of any skill level grow.
  - Career professionals who may have a tech degree or a specialization in cybersecurity. Spanning from younger to middle-aged adults and beyond, these individuals may be a part of a corporate team or academics, and will likely have lots of experience relevant to CySim's collaborative security environment.

Conceptual/Visual Sketch

IOWA STATE UNIVERSITY

Center for Cybersecurity Innovation and Outreach



Requirements

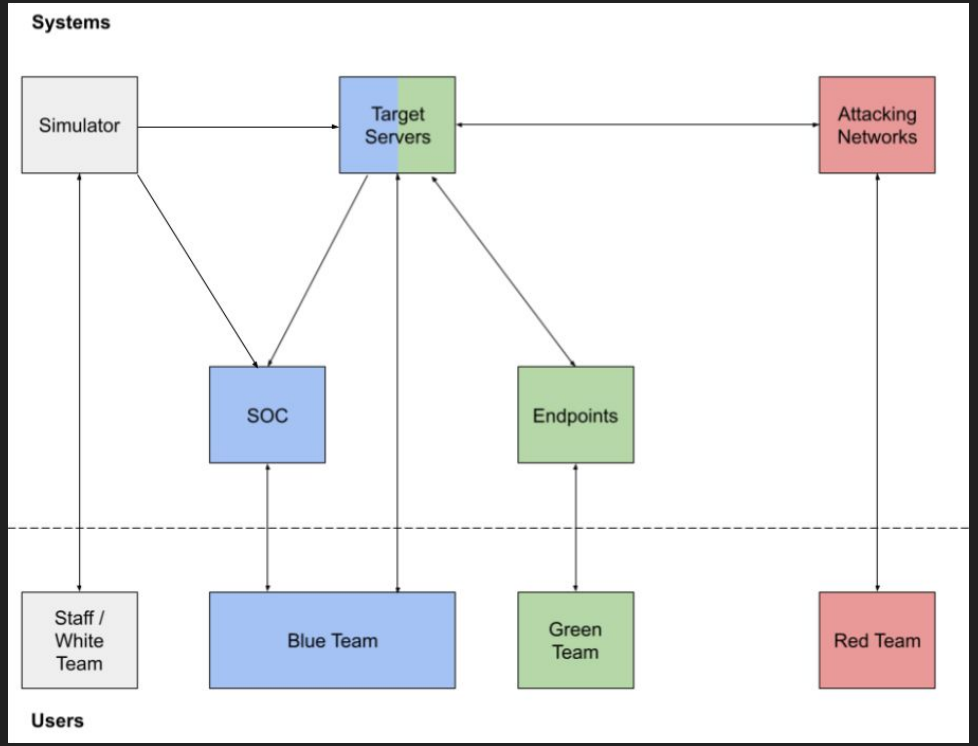
- Functional
  - CySim needs to be accessible for each type of participant, in person and remotely.
  - Situational logic in the backend to accommodate each simulation and game type.
  - Scoreboard design needs to be easily understandable by teams and accurately depict the state of the simulation or game.
- Resources
  - We are provided a server to implement CySim’s backend logic on.
  - We are provided with a domain name to host CySim’s web UI.
  - Using .NET libraries for our backend.

Requirements Continued

- User Experience
  - CySim should provide feedback to user actions. This can be applied in terms of simulated social media activity, or even press conferences and news reports.
  - Both technical and non-technical users should be able to interact with each scenario in CySim.
  - Users should have the ability to tailor a scenario to their various needs and skill levels.
- Environmental
  - CySim will provide users skills that they will be able to apply in real world situations.
  - CySim can offer college students real-world experience before they enter their careers.
  - CySim will continue to stay up-to-date and educate users on new, emerging threats.

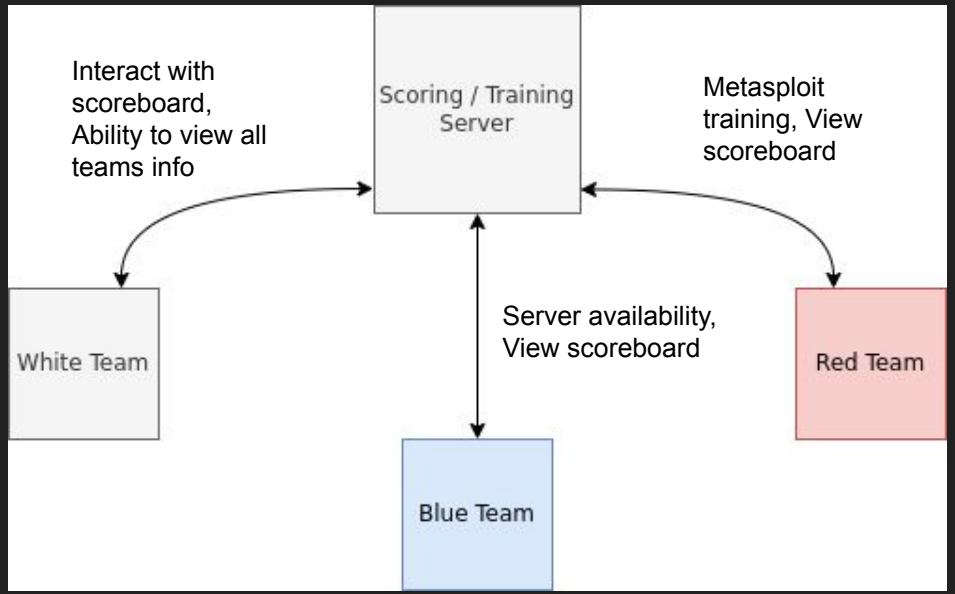
System Design: CySim Field Overview

- CySim field/network
  - White team attack simulator
  - Blue team SOC to report attacks
  - Green team endpoints
  - Red team offensive machines
- Scenarios are configurable to meet each client's needs
  - Students can have preset scenarios
  - Corporate can request custom scenarios depending on the threats and training they want



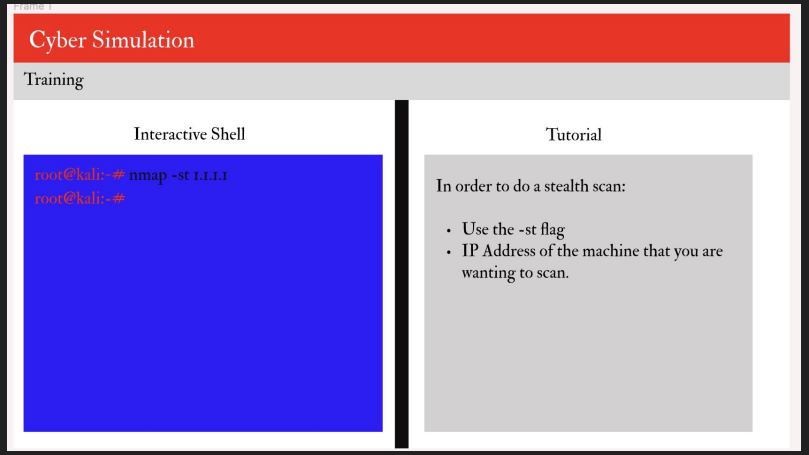
System Design: Web Server and Mock Network

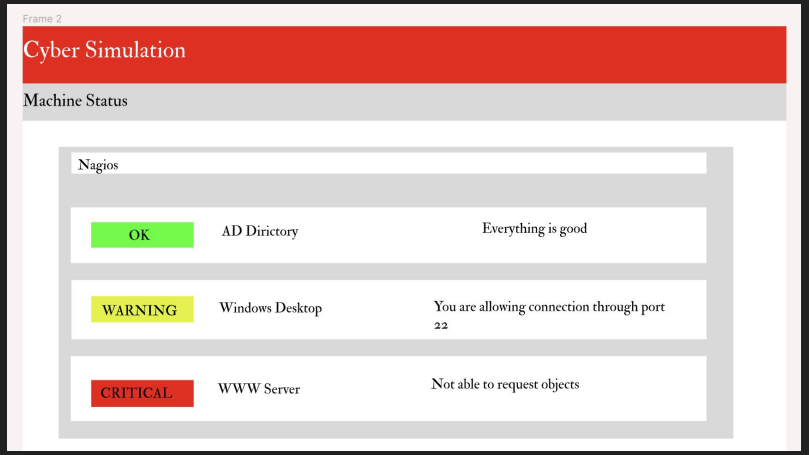
- Web server
  - Home
  - Team Registration
  - Scoring
  - Nagios / Service Uptime
  - Training
  - Anomalies
- Mock CySim Network
  - Machine Access
  - Scenarios
  - Scoring
  - Nagios



System Design: UI

- Components and Modules
  - Team Registration
  - Machines
  - Scoreboard
  - Nagios
  - Training
  - Scenario
- UI/UX
  - CySim Web App
  - VmWare

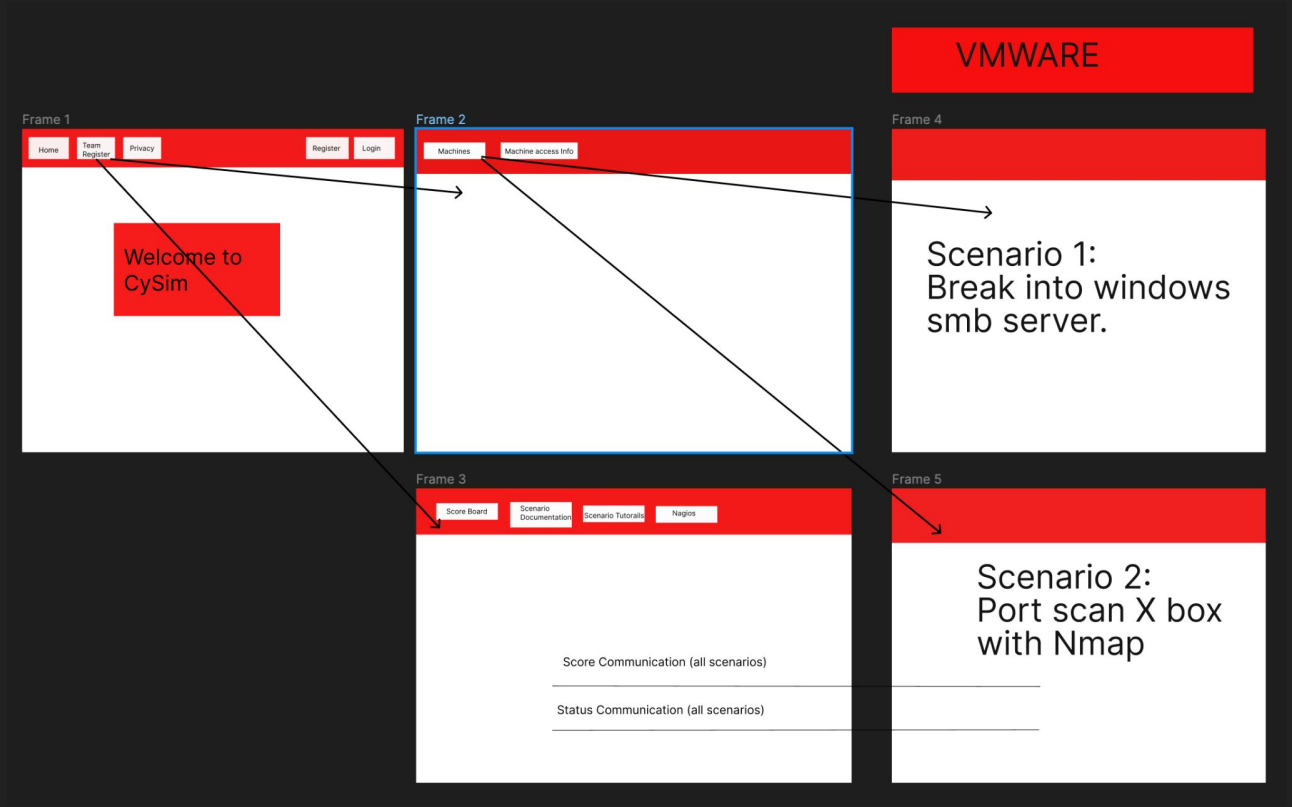




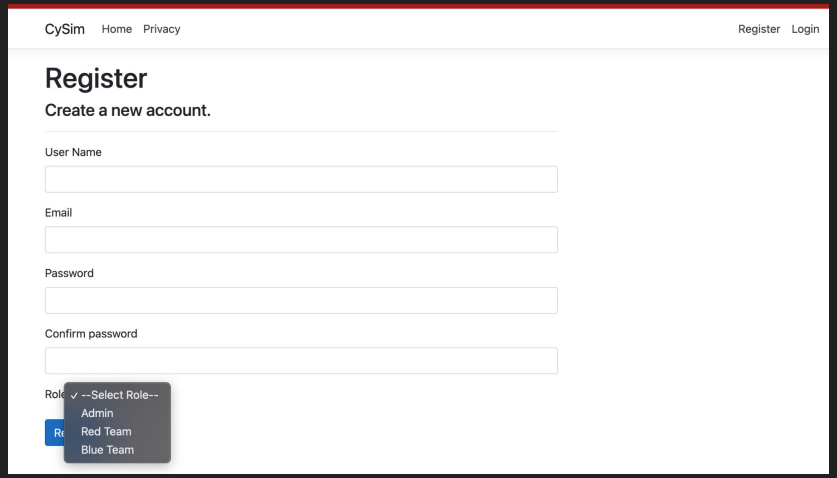
System Design

- Technology
  - VMWare
  - Nagios
  - Visual Studio / Visual Studio Code
  - MsSql Sever
  - Docker
- Frameworks
  - .Net Core
- Libraries
  - Microsoft Identity
  - Serilog Logger
  - Devexpress Reports

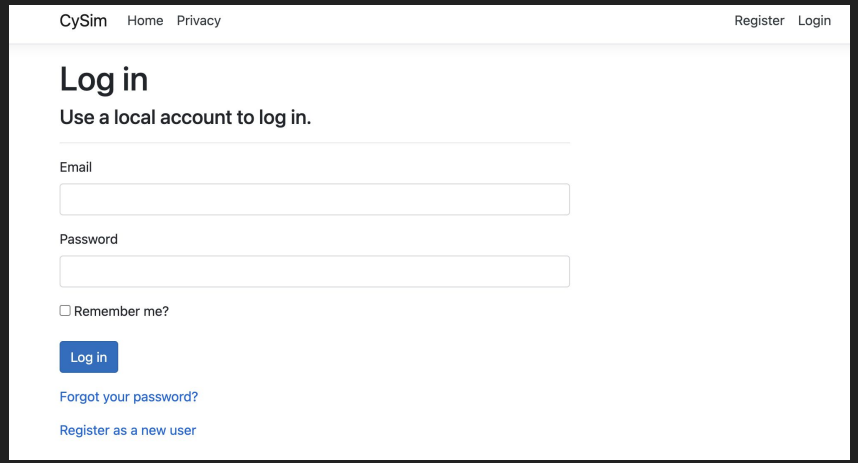
Conceptual Design Diagram



Prototype Implementations: Login and Registration



- This is how a user will register to be a part of CySim

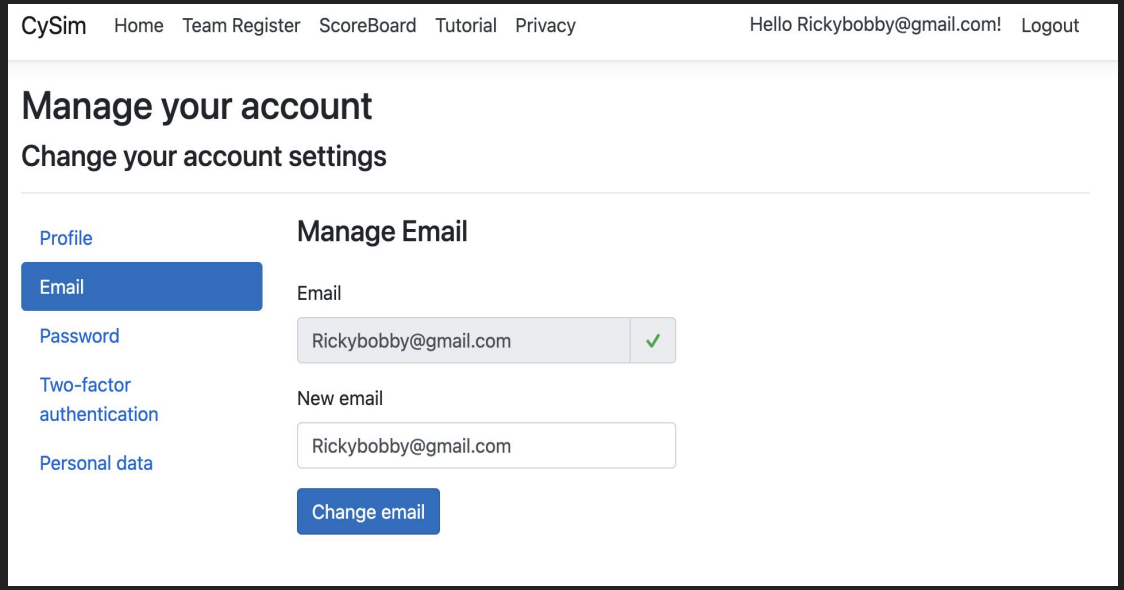


- The user will use their email and password to login.
- If they forgot password they can get a new link email to them to change it.
- They can also click on the registration link and be brought to registration page.

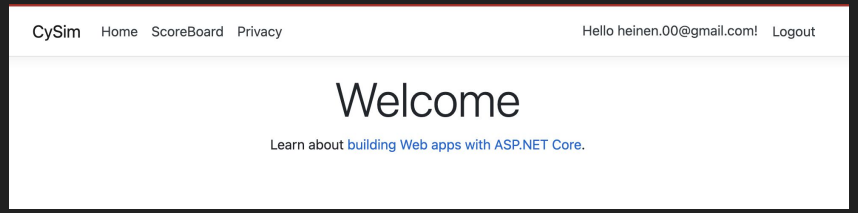
Prototype Implementations: In App User Profile

The user will have the ability to update their user information in app.

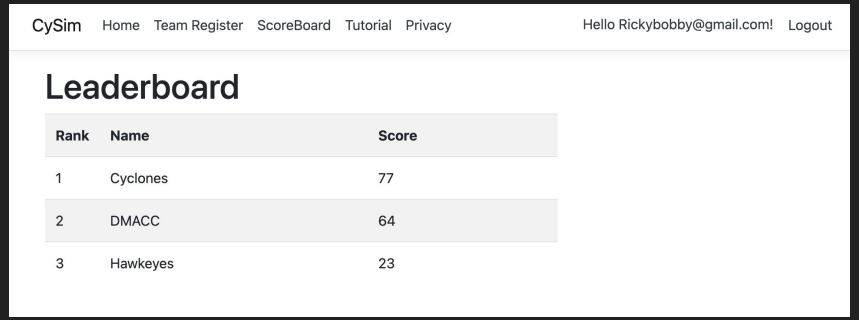
- They can add a phone number to their profile
- They can change their email address
- They can change their passwords
- They can easily add two factor authentication.
- Update their personal data



Prototype Implementations: Loading App View based off of user's role type



- Example of a user logging into our application with the role "Admin".



- Example of the user logging into our application with the role "Blue Team" or "Red Team'

Design Complexity

- CySim Challenges
  - Getting started
    - We need a full stack framework to be able to control all of the data and routes in app.
    - We needed to be able to have our web app be able to read data out of the MsSql Database and act on the data in different ways.
    - We need to manipulate Microsoft Identity to work for CySim
- Design Iterations
  - Login/Registration
    - I was not able to get this two modules to communicate perfectly.
      - I needed to add role at login page so that I can load the users views based off of the login role.
      - There is still minor issues like displaying the user's email address when the users logs in instead of the username.

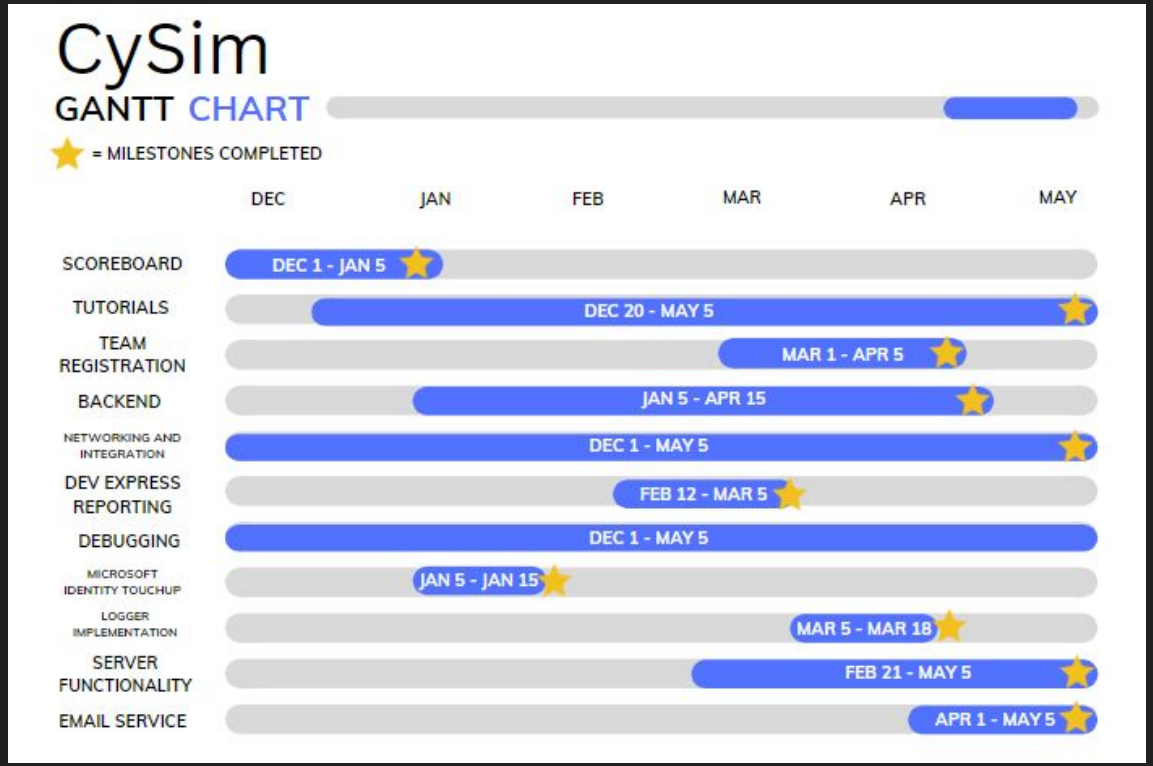
Project Plan

- Assessment of need
  - CDCs, CTFs and Tabletops
- Tasks and Risks
  - Creating the Backend Logic for the various of scenarios within CySim.
    - Risk: Bugs within the logic that we can’t see or don’t find, could lead to scenarios failing and we wouldn’t know why.
      - Mitigation Plan: In depth debugging by the team to ensure that no error goes unnoticed
  - Creating and implementing the CySim field which will display the user interface for all users.
    - Risk: The user interface and the backend logic don’t cooperate in the way that they should and it leads to the CySim field not functioning.
      - This risk has a relatively low risk probability, 0.3.

Project Plan Continued

- Tasks and Risks
  - Creation and Implementation of the Web Application and connecting it to the database.
    - Risk: Doing this we could run into a variety of security issues such as receiving more data in the database than needed or potentially causing an issue for the web application.
      - Mitigation Plan: Double checking the database connection throughout the merge process, and with a solid amount of test cases we will be able to make sure everything works properly.
  - Debugging each stage of the creation process of CySim
    - Risk: Something is missed early on and we don’t see it, which then leads to a chain reaction with a lot of issues.
      - Mitigation Plan: Consistently commenting code, solid communication and debugging each portion will help prevent this.

Schedule and Milestones: Gantt Chart



Schedule and Milestones Continued

- Major Milestones for CySim
  - Creation of the Web Application that will be used to host CySim.
  - Connection of the Web Application and database.
  - User creation and classification between red team blue team and admin.
  - Merging the web application and backend.
  - Creation and implementation of tutorial pages within the web application.
  - Creation of at least one game for CySim to run and utilize.
  - Mail service implementation.
  - Scoreboard design and layout done and works properly.

Testing Plan

- Most testing will take place on the actual platform
- Units to be tested:
  - Web Application and Database
    - User/Team Registration, machine states, scoreboard, etc.
    - Testing done mostly by hand through mock signups and tracking of database updates
    - Ensure compliance with game management and information requirements
  - Backend Scripting and Systems
    - Scripts for scoreboard updates, feedback triggers, automated attacks, etc.
    - Testing through created scenarios and automated testing tools/test scripts
    - Ensure compliance with automated feedback, scoreboard system, red vs blue team operations and scenario handling requirements

Testing Plan Continued

- Integration Testing
  - Web Application and Database will be tested via data tracking
    - Ensuring proper storage of information
  - Database and Backend System integration will be tested through extensive case input
  - Web Application and Backend System integration will tested via simple hands on testing
- Interface Testing
  - Scoreboard and auto scoring features will be tested by hand
    - Mock CDC style events
  - Monitoring signals between web application, and backend to ensure proper reactions
    - Automated systems to ensure signals are constant throughout process

Testing Plan Continued

- System and Acceptance Testing
  - Entire System tested through an extensive but simple test scenario
  - Test scenario will be vetted by client
  - Constant contact during implementation with client to ensure standards are met

Conclusion

- Project is currently on time between the design and implementation stages
  - Prototype web-application is currently in development
  - Networking and Backend design is nearing completion
- Current plans are to proceed as scheduled and expected to reach a functional product by early May 2023 or earlier
- Member Contributions:
  - Matthew Daoud - Project Organization and Systems Design
  - Brady Schlotfeldt - Network and Systems Design
  - Bailey Heinen - Web-Application Prototype Implementation
  - Jacob Boicken - Network and Systems Design
  - Ethan Swan - Web-Application and Systems Design

Thank you for your time!

Questions?