CySim

DESIGN DOCUMENT

sdmay23-11
Client: Professor Jacobson

Team Members:
Bailey Heinen - Full Stack developer
Ethan Swan - Full Stack developer
Brady Schlotfeldt - Backend and Systems Developer
Jacob Boicken - Linux Administrator and Systems Developer
Matthew Daoud - Project Manager

Email: sdmay23-11@iastate.edu
Website: https://sdmay23-11.sd.ece.iastate.edu

Revised: Dec 2 2022 / Version 1.4

Executive Summary

Development Standards & Practices Used

List all standard circuit, hardware, software practices used in this project. List all the Engineering standards that apply to this project that were considered.

- We are using .net core 3.1 which is a full stack development framework for web applications
- Microsoft identity
- We will be following the .net core software security standards by:
    - Parameterized stored procedures talking to database
    - Cleanse file paths, logs and any other place our app is passing untrusted data.
- NIST
- MITRE Attack Framework

Summary of Requirements

List all requirements as bullet points in brief.

- Three types of users
    - Admin -administrators of the application
    - Blue Team – defending users of the application
    - Red Team - attacking users of the application
- Application needs to be quick and responsive
- Easy to change the game being played
- Game Master must have complete control over the scenario
- Implemented Scoring Logic
- Simulated Feedback Systems
- We need to complete all the respected Web-Application Modules
    - Team and CyMail Registration/Setup
    - Score board
    - Tutorials
    - Blue Team Access (Machine)
    - Red Team Access (Kali Box)

-

Applicable Courses from Iowa State University Curriculum

List all Iowa State University courses whose contents were applicable to your project.

- Engl 314: Technical Communication
- Com S 227: Object Oriented Programming
- Com S 228: Introduction to Data Structures
- Com S 309: Software Development Practices
- CprE 230: Cyber Security Fundamentals
- CprE 231: Cyber Security Concepts and Tools
- CprE 430: Network Protocols and Security

New Skills/Knowledge acquired that was not taught in courses

List all new skills/knowledge that your team acquired which was not part of your Iowa State curriculum in order to complete this project.

- How to work a team to bring a real-world application from a thought to real life
- How to start .Net Core project from the start and get everything configured correctly
- How to work Docker
- How to configure MsSql server for a web app
- How to use Microsoft Identity
- How to customize the identity for our use
- How to use Figma
- How to query a database
- How to parameterized queries
- How to set up an email service

Table of Contents

List of figures/tables/symbols/definitions (This should be the similar to the project plan)

Figures:

- Section 3.4 is where our Gantt Chart is that breaks down our project schedule with specific tasks and milestones that we have.
- Section 4.1.3 is where two diagrams are located that breakdown the technical complexity of our project.
- Section 4.2.2 is where our Lotus Blossom diagram is located for CySim.
- Section 4.3.1 is where our CySim design is located with regards to user interaction. Along with this is another design that displays the CySim server and its regards to user interaction.
- Section 4.3.2 is where our in depth prototype design images are located for CySim. This is a part of the detailed design portion of the project.
- Section 5.8 is displaying images showing the results of our web application for what CySim currently looks like at this time.

Tables:

- Section 3.6 is where our table is located that shows the projected team contributions for CySim.
- Section 4.1.1 is where our table is located that shows the broader context to our project, this breaks down what communities and organizations our project relates and works with.
- Section 4.2.3 is where our Decision-Making Matrix is located for CySim.
- Sections 7.1 and 7.2 are where our tables full of our results of the IEEE protocols and breaking down each and ranking them on our level of involvement.

1 Team

1.1 Team Members

- Bailey Heinen
- Ethan Swan
- Brady Schlotfeldt
- Jacob Boicken
- Matthew Daoud

1.2 Required Skill Sets for Your Project

- **Web Application / .NET**
- **Software Development**
- **Version Control (git)**
- **Networking**
- **Virtualization**
- **General Cybersecurity Tools**

1.3 Skill Sets covered by the Team

- Web Application / .NET - Bailey Heinen, Ethan Swan
- Software Development - Bailey Heinen, Ethan Swan, Brady Schlotfelt, Jacob Boicken, Matthew Daoud
- Version Control (git) - Bailey Heinen, Ethan Swan, Brady Schlotfelt, Jacob Boicken, Matthew Daoud
- Networking - Jacob Boicken, Matthew Daoud, Brady Schlotfelt
- Virtualization - Bailey Heinen, Ethan Swan, Brady Schlotfelt, Jacob Boicken, Matthew Daoud
- General Cybersecurity Tools - Bailey Heinen, Ethan Swan, Brady Schlotfelt, Jacob Boicken, Matthew Daoud

1.4 Project Management Style Adopted by the team

Each team member has equal say in decisions. A single individual will be responsible for ensuring the project is pushed forward. This includes scheduling of meetings, contact with the client, and maintaining team communication. Overall the management style is best described as laissez-faire.

1.5 Initial Project Management Roles

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

2 Introduction

2.1 Problem Statement

What problem is your project trying to solve? Use non-technical jargon as much as possible. You may find the Problem Statement Worksheet helpful.

Who - Students or cybersecurity or enterprise professionals that want to improve their cybersecurity readiness. Corporations that want to scout the next generation of cybersecurity professionals.

What - Cybersecurity offense and defense practice. The cybersecurity and business fields need a way for people to learn and improve abilities in defending computer systems, exploiting vulnerabilities, and responding to cyberattack situations.

Where - Situation based, in workspaces when incidents occur and employees don't know how to accurately respond, students in their everyday life trying to prepare for after college, and they lack the real life attack prevention and response experience needed to fully be prepared.

When - A continuous and ongoing issue for corporations, employees, and students who need to keep and develop talent prepared for attacks.

Why - To help students and employees practice and learn techniques in response to cyberattacks. This helps prepare students and companies' abilities in the many fields of incident response.

How - Through the creation of a cybersecurity arena where individuals can learn, practice, and develop their skills for the cybersecurity industry.

Problem Statement: What is a good way to decide whether a candidate is fit for cybersecurity roles, and to train the next generation of cybersecurity professionals.

2.2 Intended Users and Uses

Who will use the product you create? Who benefits from or will be affected by the results of your project? Who cares that it exists? List as many users or user groups as are relevant to your project. For each user or user group, describe (1) key characteristics (e.g., a persona), (2) need(s) related to the project (e.g., a POV/needs statement), and (3) how they might use or benefit from the product you create. Please include any user research documentation, empathy maps, or other artifacts as appendices.

Students:

- Demographics: Likely a younger individual. Attempting to gain an undergraduate degree or maybe graduate degree. Might not have a lot of experience in real world cybersecurity applications.
- Hobbies/Interests: Probably quite busy with school work, but still enjoys computers and computer technology. An interest in offensive or defensive cybersecurity. May tinker, but lacks a place to put skills to use.

- Motivation: To earn a living doing something that they're interested in. To also gain valuable skills and expertise in their field.
- Personality/emotions: Likely overwhelmed and stressed with their schoolwork. Also stressed due to having to somehow get an internship or work after college. Might be more of an introvert who prefers to tinker with their devices.
- Values: Putting forth a good effort on schoolwork and succeeding in their field.

Students need a place to practice their cybersecurity skills as well as learn new ones because they want to have a successful career in cybersecurity.

Students will have a place to practice and learn new cybersecurity skills. They can also potentially prove their capabilities to professors and employers, giving an even greater benefit.

Technical Corporation Employees:

- Demographics: Likely middle-aged or older with some form of degree and certifications. Probably new to cybersecurity or already has gained quite a bit of experience.
- Hobbies/Interests: Probably busy with work, but still enjoys computers and computer technology. An interest in offensive or defensive cybersecurity. May tinker, but like students may lack a place to put their full range of skills to use. Likely tries to keep up with the evolving landscape the best they can by reading articles/news about cybersecurity.
- Motivation: To earn a living doing something that they're interested in and support which they've likely started.
- Personality/emotions: Likely overwhelmed and stressed with their work. Also stressed due to the shifting cybersecurity environment.
- Values: Keeping the company safe from cyberattacks. Probably starting a family and values their side projects. Likely values their free time and family time. Ensuring they do the best job possible to continue advancing.

Technical corporation employees need a way to efficiently practice their skills because it may have been a while since they've encountered certain scenarios or may not be prepared for a new cybersecurity threat.

The benefits besides improving their skills and gaining experience against new threats would be potentially showing their employer they deserve extra compensation or some sort of promotion.

Non-Technical Corporation Employees:

- Demographics: Likely older with very little experience in cybersecurity. Likely doesn't have much of an idea of the technical aspects and is more focused on running the company. May have a degree like an MBA. Probably quite wealthy overall as they take on more of a management role.
- Hobbies/Interests: Probably enjoys more free time outside of work. Might go on weekend trips or collect expensive items.
- Motivation: To make their company earn as much money as possible. Find the best talent and keep the company secure in the cyber realm.
- Personality/emotions: Likely outgoing and friendly. Maybe a bit boastful or may brag about their position.

- Values: Likely values company success and employee motivation. Likely values their free time and families.

Non-Technical employees need a way to be able to reach recruitable students and practice non-technical skills such as media management during a cybersecurity attack. This is because in order to ensure their company remains profitable, they have to maintain a high level of company security.

The benefits of this would be making it easier for recruiters to find new talent at student events. It would also allow for them to practice those skills outside of the technical ones that may be needed during a cyberattack.

Professors:

- Demographics: Likely a middle-aged or older individual. They likely have a higher level degree, such as a PhD. Probably has a lot of experience and knowledge relating to the field of cybersecurity.
- Hobbies/Interests: Likely spends a lot of their time writing new lessons, exams, and lab assignments. Likely runs CDC's and enjoys discussing cybersecurity related matters with students.
- Motivation: To teach and help students learn new cybersecurity skills while also helping them to practice old ones. To create the next generation of cybersecurity experts.
- Personality/emotions: Likely stressed with having to put together a lesson plan and teach. Likely enjoys putting on CDC's and seeing their students gain knowledge related to cybersecurity.
- Values: Challenging their students. Making sure that they gain knowledge related to cybersecurity.

Professors need a simple-to-use program or space to efficiently teach and help students to gain skills and experience, since they are motivated by training the next generation of cybersecurity experts.

Professors can use CySim as a new way to engage with students in the field that they are teaching, and can treat it as a new way of administering curriculum or other academic, skills-based challenges. The other benefit is making this easier than now and giving them a good way to let students gain real world experience in a controlled environment.

2.3 Requirements & Constraints

List all requirements for your project. Separate your requirements by type, which may include functional requirements (specification), resource requirements, physical requirements, aesthetic requirements, user experiential requirements, economic/market requirements, environmental requirements, UI requirements, and any others relevant to your project. When a requirement is also a quantitative constraint, either separate it into a list of constraints, or annotate at the end of the requirement as "(constraint)." Ensure your requirements are realistic, specific, reflective or in support of user needs, and comprehensive.

Functional Requirements:

- The CySim Field needs to be an open space for each type of participant, such as red teams, blue teams, conference speakers, employees and spectators.
- The Backend Logic needs situational logic for different types of scenarios being played through, along with logic to track the progress being made as well as the points being scored.
- The Scoreboard Design needs to display both teams scores to the room and also display progress and other details about a simulation.

Resource Requirements:

- We are being provided a server, this will be used to create the Backend Logic for the CySim Field as well as create the UI for the CySim Field itself.
- We will utilize .NET libraries for our backend, limiting us to functionality offered by .NET libraries (constraint)

User Experiential Requirements:

- Should provide simulated feedback to user actions and inputs that may take the form of simulated social media and news.
- Should allow technical and non-technical members to interact in the scenario, such as managing public relations or employee meetings for non-technical team members.
- Users should be able to tailor scenarios to their specific needs and allow for practice against scenarios that may occur more often in their sectors, such as hospitals and ransomware attacks.

Environmental Requirements:

- CySim should allow users to increase their skills and preparedness for real-world threats.
- The facility can offer students real-world experience before and after they enter careers in cybersecurity.
- CySim can easily stay updated on emerging technologies and threats as the security field evolves.

2.4 Engineering Standards

What Engineering standards are likely to apply to your project? Some standards might be built into your requirements (Use 802.11 ac Wi-Fi standard) and many others might fall out of design. For each standard listed, also provide a brief justification.

IPV4

- We plan on using IPv4 for communications between systems we create to keep network design simplistic for blue team management and logging and for red team reconnaissance

NIST Cybersecurity Framework - Blue team

- The NIST CF goes over the five functions of blue teams
  - Identify, Protect, Detect, Respond, and Recover
- We will be designing the scoring and systems for the blue teams around these functions

MITRE ATT&CK Framework - Red Team

- ATT&CK framework goes over the many stages of a cyberattack that a red team will follow
- We will be using these stages for designing scoring and backend system for the red teams to use

OSI 7 Layer Model

- We plan on using the OSI 7 Layer Model in order to display and simplify security events for the sake of scoring and explanation to the players

Microsoft Identity Authentication.

- This ensures login accuracy
- Hash's user's password
- Email verification is easy to implement so no spam user

.Net core software security standards:

- Parameterized stored procedures talking to database
- Cleanse file paths, logs and any other place our app is passing untrusted data.

3 Project Plan

3.1 Project Management/Tracking Procedures

Which of agile, waterfall or waterfall+agile project management style are you adopting. Justify it with respect to the project goals.

For this project, our team has a management style that is a combination of both waterfall and agile. The main reasoning behind this is we input styles from both within our project. Our timeline is adaptive as we can change/switch things around as time goes on, our client involvement is more on the waterfall side as we have check-ins but not as common as agile, for iteration we follow more closely with waterfall as we operate in phases more than likely, lastly, we have high flexibility which relates to agile and high planning required which relates to waterfall.

What will your group use to track progress throughout the course of this and the next semester. This could include Git, GitHub, Trello, Slack or any other tools helpful in project management.

Below are the different ways we will track progress throughout this course and the following semester:

Discord - We have used discord throughout the semester already as a form of communication, however, we also have channels that are filled with important dates, information and other goals that we have for this project.

GitHub - When we start working on the backend logic of the CySim games it will make a lot of sense to have a centralized location where all of our code can be stored and accessed, GitHub is one of the easiest to use, and we all have previous experience with it. We can have milestones, share issues, and manage merge conflicts.

Google Drive - Our group has a shared Google Drive where we input all the important documents that we may need and anything else that is important to the project.

3.2 Task Decomposition

In order to solve the problem at hand, it helps to decompose it into multiple tasks and subtasks and to understand interdependence among tasks. This step might be useful even if you adopt agile methodology. If you are agile, you can also provide a linear progression of completed requirements aligned with your sprints for the entire project.

For the task decomposition, we are taking more of a waterfall design approach:

- Requirements
    - Developing the CySim field, this is the main interface for where the games will be played.
    - For the backend logic, we will code the different games and scenarios being played so that CySim can actually function properly at the current time.
    - Developing and implementing that Web Application that will be used to access CySim.

    - Start .net core project.
    - Connect the created Web Application to a database.
- Design
    - SOC/Scoreboard, we are tasked with creating an efficient and good-looking design for the scoreboard so that it displays all of the information that the users, spectators and anyone else in the facility may need.
    - Designing the main interface for the field, this will be in connection with the scoreboard.
    - Creating a way to access the blue/red team's machines.
    - Create a tutorial module as well as any modules that have to do with user type.
- Implementation
    - Work on the backend logic of the game so that it can be uploaded to the server once functional.
    - Work on the CySim field that will hopefully display the results and other aspects of the backend logic.
    - Implement a way to combine the user interface field with the backend logic so that it all functions properly.
    - Pull our designs for the scoreboard and then implement them into the scoreboard layout to see if it fits properly.
    - Implement a team registration page so that users can join or create a team.
    - Implement a scoreboard that displays the active score for both red and blue teams.
- Testing
    - Testing/debugging the backend logic for the CySim functionality to make sure that all of the scenarios are working correctly.
    - Making sure that the user interface as well as the scoreboard layout looks correct and also functions in the way that it should.
    - Making sure that both the logging in and creating user functionality continues to work properly.
- Maintenance
    - As time goes on, look for potential problems or errors, or even things that can be adjusted for efficiency's sake.
    - Along with this, we then change/edit code or design features to make sure that the issue no longer occurs or that the design is more efficient.

3.3 Project Proposed Milestones, Metrics, and Evaluation Criteria

What are some key milestones in your proposed project? It may be helpful to develop these milestones for each task and subtask from 2.2. How do you measure progress on a given task? These metrics, preferably quantifiable, should be developed for each task. The milestones should be stated in terms of these metrics: Machine learning algorithm XYZ will classify with 80% accuracy; the pattern recognition logic on FPGA will recognize a pattern every 1 ms (at 1K patterns/sec throughput). ML accuracy target might go up to 90% from 80%.

In an agile development process, these milestones can be refined with successive iterations/sprints (perhaps a subset of your requirements applicable to those sprint).

Below are some key milestones that have been assigned to us within the project:

Creating the Web Application that will serve as the way to access CySim.

Have the Web Application connect to a database so that proper information can be stored.

Classification of Red team, Blue team and admin accounts when logging in or creating an account within the website.

Merge the Web Application with the backend so that it is all coordinated and functioning properly.

Successfully create and implement tutorial pages for all teams within the web application.

Backend Logic will be fully functional for all scenarios that the client wants implemented for CySim.

The scoreboard will have a layout that's representative of the state of the current scenario, and updates to keep accurate information. It will also be readable and accessible to any and all players.

Mitigate users to a join table so that they can join a team, whether its red team or blue team, allow the users to be able to register.

Creation and implementation of at least one game to try and utilize and simulate for CySim.

Implement a mail service that functions hand in hand with the web service.

3.4 Project Timeline/Schedule

• A realistic, well-planned schedule is an essential component of every well-planned project

• Most scheduling errors occur as the result of either not properly identifying all of the necessary activities (tasks and/or subtasks) or not properly estimating the amount of effort required to correctly complete the activity

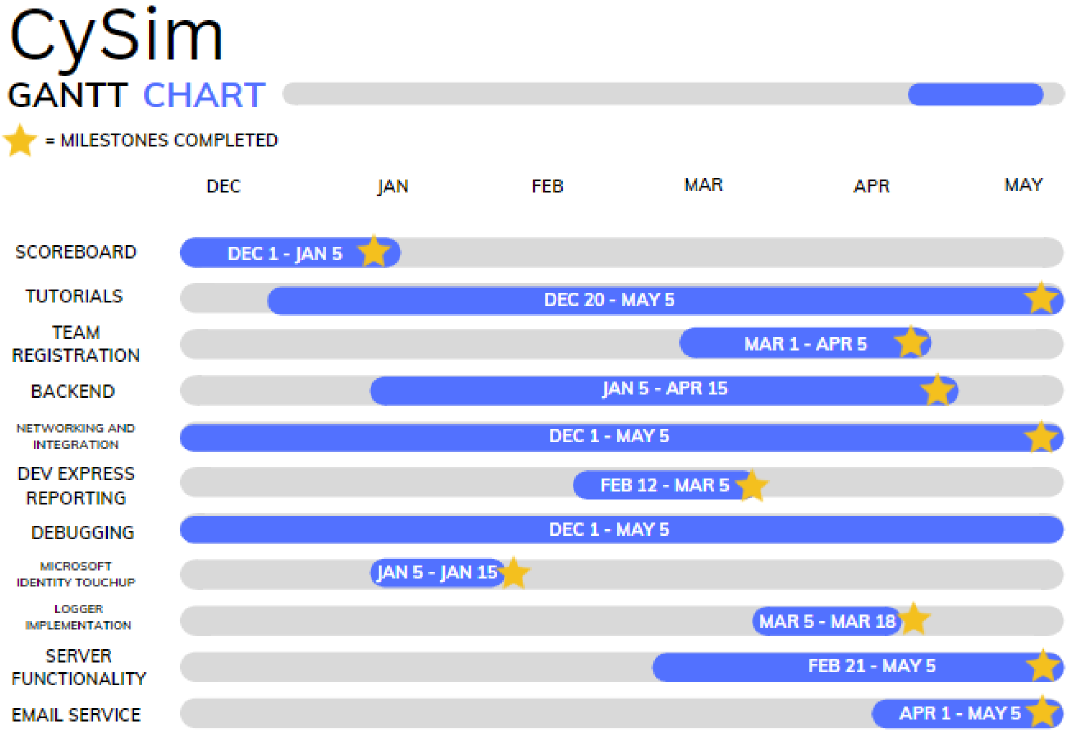
• A detailed schedule is needed as a part of the plan:

– Start with a Gantt chart showing the tasks (that you developed in 2.2) and associated subtasks versus the proposed project calendar. The Gantt chart shall be referenced and summarized in the text.

– Annotate the Gantt chart with when each project deliverable will be delivered

• Project schedule/Gantt chart can be adapted to Agile or Waterfall development model. For agile, a sprint schedule with specific technical milestones/requirements/targets will work.

Below is our Gantt chart showing our Project Timeline and schedule:



3.5 Risks And Risk Management/Mitigation

Consider for each task what risks exist (certain performance target may not be met; certain tool may not work as expected) and assign an educated guess of probability for that risk. For any risk factor with a probability exceeding 0.5, develop a risk mitigation plan. Can you eliminate that task and add another task or set of tasks that might cost more? Can you buy something off-the-shelf from the market to achieve that functionality? Can you try an alternative tool, technology, algorithm, or board?

Agile project can associate risks and risk mitigation with each sprint.

Task 1: Creating the Backend Logic to the scenarios within CySim (Make it playable/functional)

- Risks
  - Bugs within the logic we develop that we don't find/see
    - Risk probability: 0.6
    - This could lead to the scenarios failing for no reason due to various bugs
- Mitigation Plan
  - In depth debugging by each group member to ensure that everything has been tested to the fullest, and with that we can confirm confidently that we don't have any errors.

Task 2: Developing the scoreboard design so that it can show everything the client wants it to show, as well as be visible to everyone.

- Risks
    - The ratio of our design does not accurately represent the scoreboard and because of that everything we designed doesn't work.
        - Risk Probability: 0.2
        - Easy fix, clarify dimensions with client ahead of time

Task 3: Creating and implementing the CySim field which will display the user interface for playing the different scenarios

- Risks
    - The user interface has trouble relating to the backend logic and doesn't work properly because of that.
        - Risk Probability: 0.3
    - Other issues such as items not displaying or loading in properly as well could be an issue.
        - Risk Probability: 0.2

Task 4: Creation and implementation of the Web Application as well as adding it to a database

- Risks
    - While attempting to establish a connection between the database and web application, we could run into issues where the database gets more information than we wanted to give it, or it causes an issue that potentially affects the current web application.
        - Risk Probability: 0.2
- Mitigation Plan
    - I think with strong debugging and following through and double-checking with each step of the merging process, we as a team will be able to catch these issues ahead of time and get in front of it if need be.

Task 5: Debugging and testing all of the previous tasks to make sure that no existing errors occur and everything works properly.

- Risks
    - While going through and testing we find something major that could cause a major change in design and if we take too long we could risk running out of time.
        - Risk Probability: 0.5
- Mitigation Plan
    - I would say the best way to prevent this would be to implement debugging and testing even more than normal and do it as we move forward in the project, this helps limit big issues getting pushed to the end as well as allowing us to finish with plenty of time.

3.6 Personnel Effort Requirements

Include a detailed estimate in the form of a table accompanied by a textual reference and explanation. This estimate shall be done on a task-by-task basis and should be the projected effort in total number of person-hours required to perform the task.

Below is our personnel effort requirements:

Team Members	Tasks to be worked on	Projected Person Hours
Brady Schlotfeldt	Systems Development, Networking Tasks, Testing and Debugging	80 Hours
Matthew Daoud	Team Management, Networking Tasks, Testing and Debugging	80 Hours
Bailey Heinen	Web Application Development and Integration	80 Hours
Ethan Swan	Web Application Development and Integration	80 Hours
Jacob Boicken	Systems Development, Networking Tasks, Web Application Integration	80 Hours

3.7 Other Resource Requirements

Identify the other resources aside from financial (such as parts and materials) required to complete the project.

Below is a list of resources that are provided to us:

- Server
  - We are provided from the client with a server for which we will use to implement our backend logic
  - Physical server will be in Coover.
- A floor layout/map of projected facility
  - A design provided by the client that displays how they want the facility to be laid out.
  - This helps us know where people will be so that we can take that and input it into our scoreboard design.
- Domain for the Web Application
  - Iastate domain name is being provided for our web application.

4 Design

4.1 Design Context

4.1.1 Broader Context

Describe the broader context in which your design problem is situated. What communities are you designing for? What communities are affected by your design? What societal needs does your project address?

List relevant considerations related to your project in each of the following areas:

Area	Description
Public health, safety, and welfare	**Cybersecurity Training** - Increases security team's abilities in protecting corporate/personal information and critical infrastructure, and students are able to use CySim for training experiences and increase job opportunities
Global, cultural, and social	**Accurate/Similar to real life scenario** - Organizations are allowed to create scenarios for students that will be tailored to situations that happen in their work environment. These need to reflect their work environment's culture and show how they can improve their security posture.
Environmental	**Power Consumption** - Servers will be running on electricity, likely generated by some nonrenewable resources. **Server / Equipment Materials** - Servers and computing resources utilize rare earth metals
Economic	**Executive Response Training** - Executives will have the space to freely examine situations and be able to practice accurately reporting on the situations. In real life, these actions affect the company's trust and stock evaluation. **Cybersecurity Training** - Participants will be able to practice defending specific systems that are tailored to real situations. So, we will have people prepared in incident response able to lessen financial damages in cyber breaches.

4.1.2 Prior Work/Solutions

Include relevant background/literature review for the project

– **If similar products exist in the market, describe what has already been done**

- **CDC competitions** - Students are able to secure a network of systems to the best of their ability. Red teams will attack these spaces and plant flags. The students are then required to look at logs and try and figure out how the red team got in and submit how. As this is going on, there are also mini challenges like decrypting a cipher text, buffer overflow attacks, and updating server software to get their team some more points.

- **CTF** - Participants all around the world can participate. This is different from the CDC because it is more individual based rather than working in a team. It is also only a challenge, and you are not securing systems. You will be doing a task to get a flag, and then you can submit the flag to get yourself some points. A lot of these challenges will include buffer overflow, integer overflow and network problems
- **Tabletop** - There is a multi-stage scenario on the system, and it is up to the participants to explain the actions they would take to be able to resolve the situation each step of the way.

– If you are following previous work, cite that and discuss the advantages/shortcomings

- **CDCs** have a lot of what we need, like space for teams to be able to secure systems. Also, it has live scoring, and mini challenge space. What it does lack is the ability for corporations to get involved.
- **CTFs** are great to practice utilizing exploits in order to capture data, and help give hands-on experience in identifying vulnerabilities. Much like the CDC, however, it's still not as accessible to team members without a basic level of vulnerability exploitation knowledge.
- **Tabletops** are very accessible to all members, but require a lot more discussion time and debriefing in order to fully understand the impact of the exercise.

– Note that while you are not expected to "compete" with other existing products / research groups, you should be able to differentiate your project from what is available. Thus, provide a list of pros and cons of your target solution compared to all other related products/systems.

- Pros of CySim:
    - CySim focuses on relating game scenarios to real world industry operations
    - CySim allow for corporate management to see incident response operations and understand a feel for time and impact of security breaches
    - CySim simulates financial impacts of security breaches, showing technical teams the importance of their positions
    - Depending on the players available, different styles of games are able to be played. These games incorporate many aspects from CDCs and Tabletops
    - CySim allows the ability to remotely connect where the CDC you have to physically be there
- Cons of CySim
    - CySim, like a CDC, is a full day competition whereas CTFs and Tabletops are able to only be an hour to a few hour events
    - In order to maintain CySim you will need to rely on external sources to ensure funding

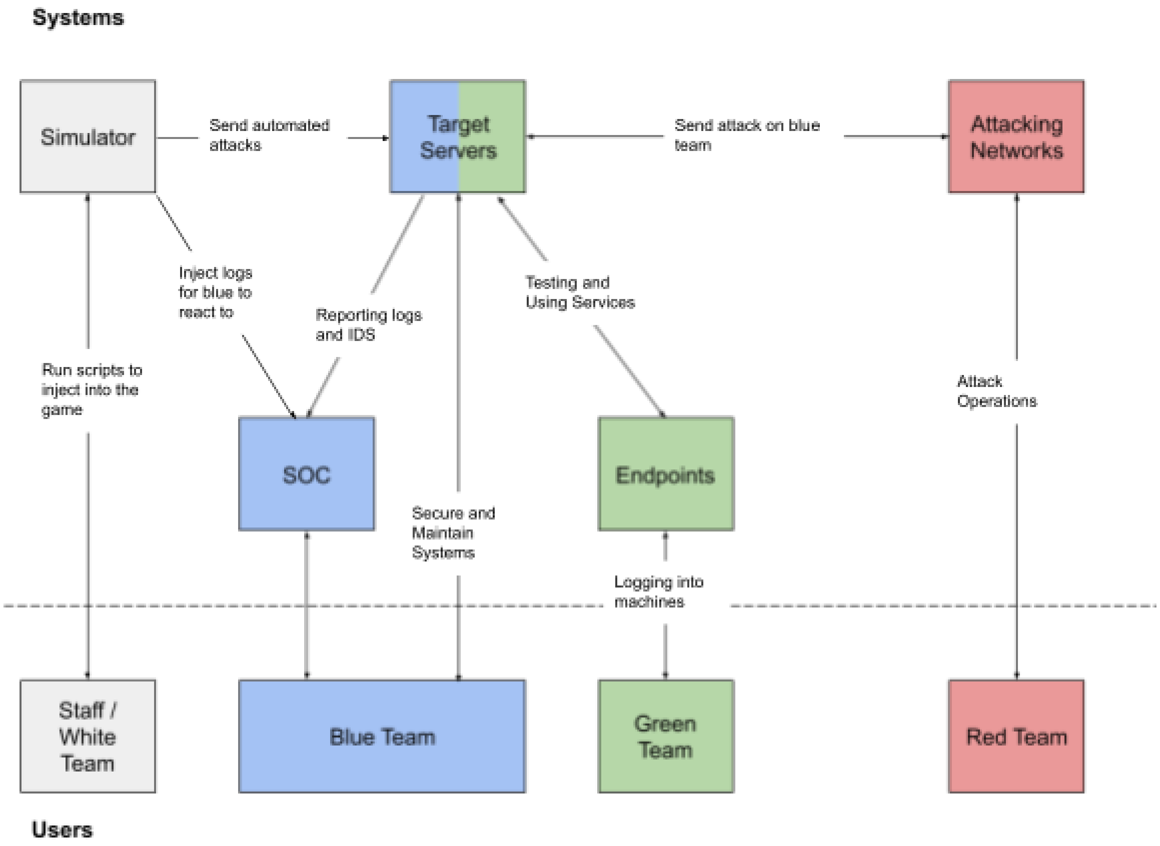
4.1.3 Technical Complexity

Provide evidence that your project is of sufficient technical complexity. Use the following metric or argue for one of your own. Justify your statements (e.g., list the components/subsystems and describe the applicable scientific, mathematical, or engineering principles)

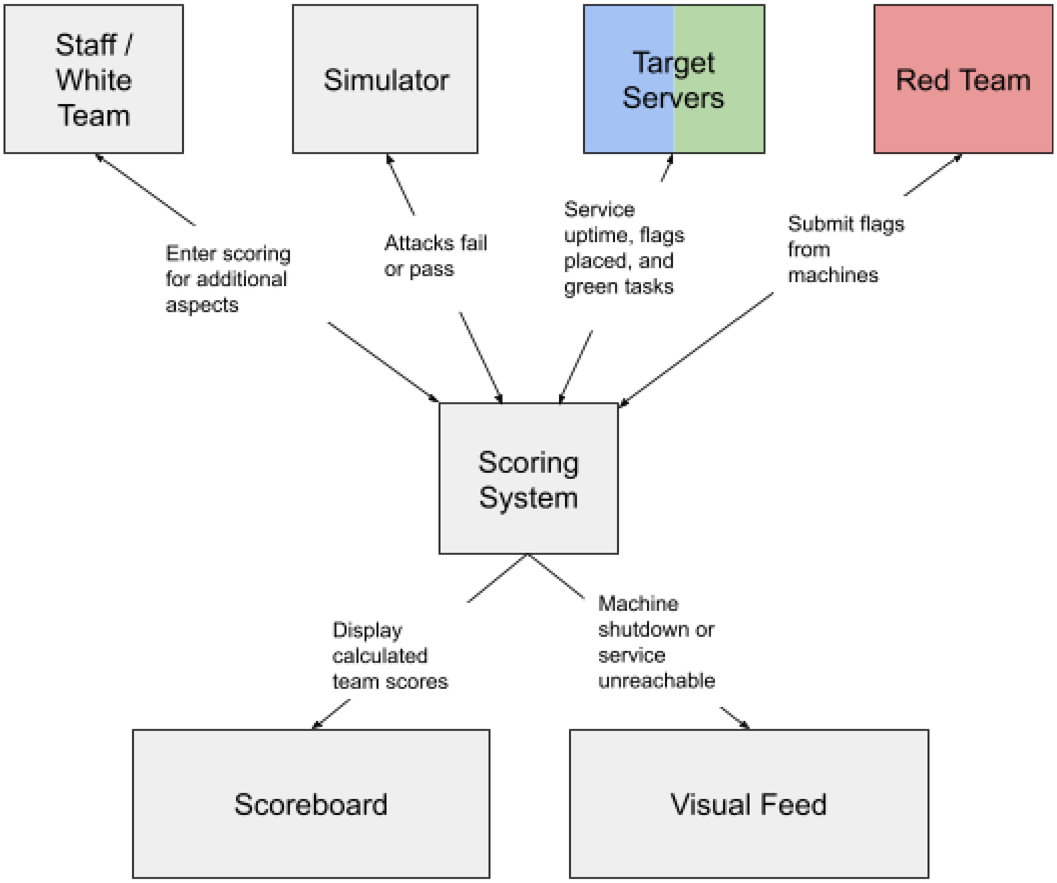
1. **The design consists of multiple components/subsystems that each utilize distinct scientific, mathematical, or engineering principles –AND–**
2. **The problem scope contains multiple challenging requirements that match or exceed current solutions or industry standards.**

As shown in our system diagrams below, our design involves multiple subcomponents across various interacting with each other to provide the desired functionality.

Systems:



Scoreboard Logic:



Staff / White Team
Simulator
Target Servers
Red Team
Enter scoring for additional aspects
Attacks fail or pass
Service uptime, flags placed, and green tasks
Submit flags from machines
Scoring System
Display calculated team scores
Machine shutdown or service unreachable
Scoreboard
Visual Feed

4.2 Design Exploration

4.2.1 Design Decisions

List key design decisions (at least three) that you have made or will need to make in relation to your proposed solution. These can include, but are not limited to, materials, subsystems, physical components, sensors/chips/devices, physical layout, features, etc. Describe why these decisions are important to project success.

How are we going to automate (script) the games?

One game mode that will be a part of CySim is the scripted games and events that will occur throughout the simulation. Because of this, we need to figure out how these will be scripted. This includes automating red team attacks and attack log generation, as well as scripted events based off of blue team actions. Scripted scenarios is likely to be the main game type conducted in CySim.

How/when will simulation feedback to the participants be triggered?

Simulated feedback will give participants a feeling of real world effect based on events and actions that occur during the gameplay. We need to research and decide how and when these will take place in order to provide this experience to participants.

Will scripted feedback be part of only the CDC's, only scripted games, or both?

Along with the previous question, we need to decide whether this simulated feedback system will be universal among all game types or reserved for specific ones such as scripted events. This is important, as simulated feedback is a large part of what makes CySim special as a cybersecurity training tool.

How will scenario control be handled?

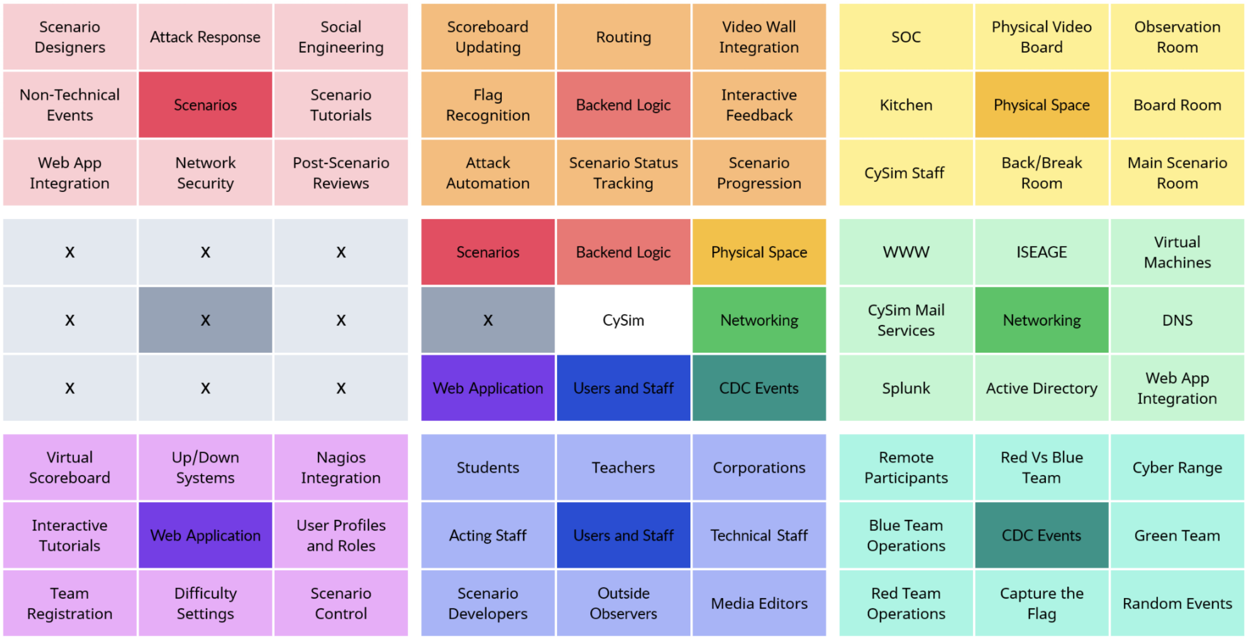
One important requirement is for the game master to have complete control over the cyber games, especially for scripted events. Because of this, we need to decide how they will interact with the game and control its flow in order to properly run the games. For this we have decided on the use of a web application.

What will be within the scope of the web Application?

Since deciding that the web application will be used to handle scenario control, another question to be considered is if we want this application to also handle other parts of the system. This may include scoreboard or team registration. Whether we want each of these elements to be controlled separately or as part of the same system needs to be decided. How we came to our decision is part of section 4.2.2.

4.2.2 Ideation

For at least one design decision, describe how you ideated or identified potential options (e.g., lotus blossom technique). Describe at least five options that you considered.



Scenario Designers	Attack Response	Social Engineering	Scoreboard Updating	Routing	Video Wall Integration	SOC	Physical Video Board	Observation Room
Non-Technical Events	Scenarios	Scenario Tutorials	Flag Recognition	Backend Logic	Interactive Feedback	Kitchen	Physical Space	Board Room
Web App Integration	Network Security	Post-Scenario Reviews	Attack Automation	Scenario Status Tracking	Scenario Progression	CySim Staff	Back/Break Room	Main Scenario Room
x	x	x	Scenarios	Backend Logic	Physical Space	WWW	ISEAGE	Virtual Machines
x	x	x	x	CySim	Networking	CySim Mail Services	Networking	DNS
x	x	x	Web Application	Users and Staff	CDC Events	Splunk	Active Directory	Web App Integration
Virtual Scoreboard	Up/Down Systems	Nagios Integration	Students	Teachers	Corporations	Remote Participants	Red Vs Blue Team	Cyber Range
Interactive Tutorials	Web Application	User Profiles and Roles	Acting Staff	Users and Staff	Technical Staff	Blue Team Operations	CDC Events	Green Team
Team Registration	Difficulty Settings	Scenario Control	Scenario Developers	Outside Observers	Media Editors	Red Team Operations	Capture the Flag	Random Events

Looking at CySim as a whole, we created a lotus blossom in order to answer some of the design decisions that we had encountered. Focusing on what the scope of the web application will be we created a web application section of our lotus blossom diagram. The potential options considered are listed below:

The first thing considered was how users will be registered and assigned to teams. This was found to be well handled by a web application with an attached database.

The second consideration was the hosting of tutorials. Since CySim may be used for novice cyber security students the inclusion of tutorials is something that may be useful. Including these in the web application would make them easy to access and use.

Third was the method of displaying system status. Nagios could likely be integrated with the already existing web application to allow easy lookup of system status depending on the scenario.

Fourth is a virtual scoreboard for CDC style events. While there will be a physical scoreboard, remote users will not be able to view this. Therefore including this in the web application will be important for these users.

Finally we have the main topic of game control. In order to give a large and easily accessible volume of control to the game master we decided to include it as part of the web application. This will allow individuals with less technical knowledge to have some control over a scenario. It will also allow for more advanced administrator controls.

4.2.3 Decision-Making and Trade-Off

Demonstrate the process you used to identify the pros and cons or trade-offs between each of your ideated options. You may wish you include a weighted decision matrix or other relevant tool. Describe the option you chose and why you chose it.

For our Decision-Making and Trade-Off portion, we decided to use a weighted decision matrix due to it being an efficient way to help us as a team make decisions that can be considered more complex or difficult than normal. Along with this, the weighted decision matrix also gives each of our team members an opportunity to make arguments and also collaborate together and solve problems that we may have within the project. This is the main reason why we decided to go with creating a weighted decision matrix.

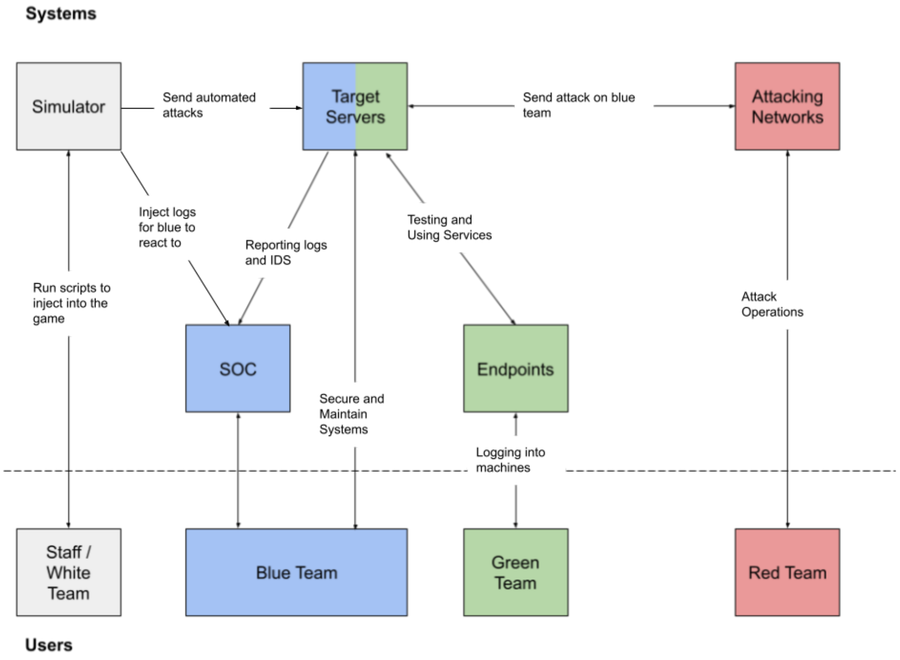
		Network Structure Options							
Criteria	Weighting (Less 1-5 More)	ISEAGE Instance		Regular VMs		Physical Hosts		Participants Just Roleplay the Whole Time	
		Score	Total	Score	Total	Score	Total	Score	Total
Ease of Implementation	2	4	8	3	6	2	4	1	2
Resources Required	4	4	16	3	12	2	8	2	8
Maintainability	4	4	16	4	16	4	16	5	20
Ease of Automation	3	4	12	2	6	2	6	1	3
Scalability	2	5	10	2	4	1	2	5	10
Avoids Obsolescence	4	4	16	4	16	2	8	5	20
	TOTAL:	78		60		44		63	

4.3 Proposed Design

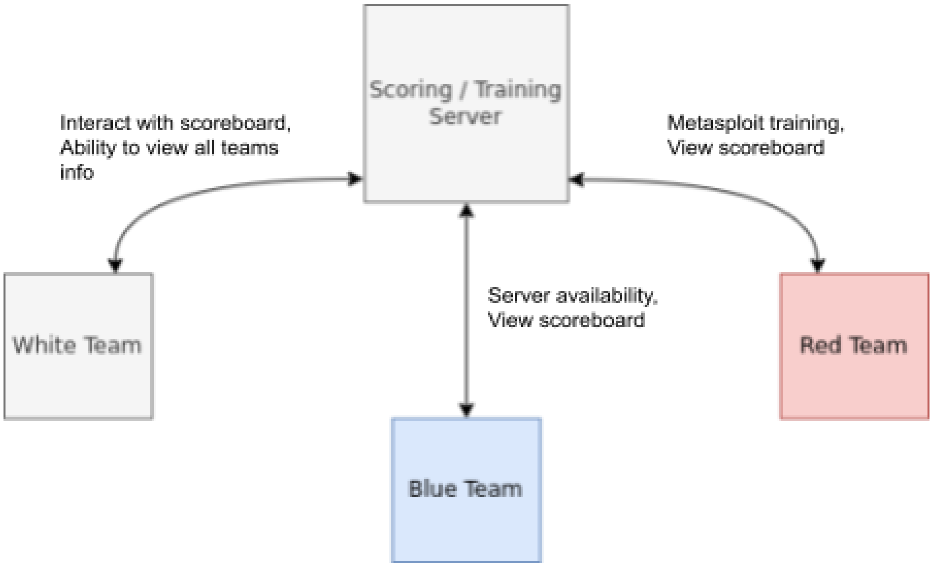
4.3.1 Overview

Provide a high-level description of your current design. This description should be understandable to non-engineers (i.e., the general public). Describe key components or sub-systems and how they contribute to the overall design. You may wish to include a basic block diagram, infographic, or other visual to help communicate the overall design.

CySim Design With User Interaction:



CySim Scoring Web Server User Interactions:


Scoring / Training Server
Interact with scoreboard, Ability to view all teams info
Metasploit training, View scoreboard
Server availability, View scoreboard
White Team
Red Team
Blue Team

Describe key components or subsystems and how they contribute to the overall design.

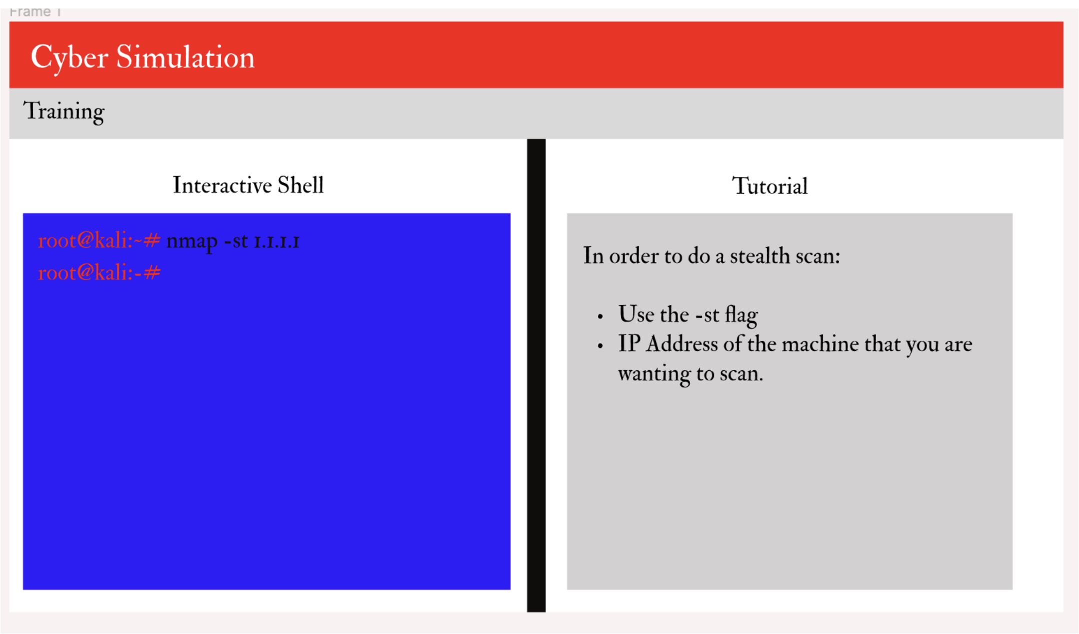
Our current design for CySim as a whole is a mock blue/green team enterprise network and red team attack network for teams to utilize for the games. For our implementable design at this early stage of CySim, we are designing a web server that all teams in the game can interact with. This server will show the current score of the scenario, be used by teams for training in how to utilize the systems given to them, and allow for submissions to impact team scores. This facilitates the operations of game scenarios and allows for interactive teaching as teams are able to see how their actions impact their score.

4.3.2 Detailed Design and Visual(s)

Provide a detailed, technical description of your design, aided by visualizations. This description should be understandable to peer engineers. In other words, it should be clearly written and sufficiently detailed such that another senior design team can look through it and implement it.

The description should include a high-level overview written for peer engineers. This should list all sub-systems or components, their role in the whole system, and how they will be integrated or interconnected. A visual should accompany this description. Typically, a detailed block diagram will suffice, but other visual forms can be acceptable.

The description should also include more specific descriptions of sub-systems and components (e.g., their internal operations). Once again, a good rule of thumb is: could another engineer with similar expertise build the component/sub-system based on your description? Use visualizations to support your descriptions. Different visual types may be relevant to different types of projects, components, or subsystems. You may include, but are not limited to: block diagrams, circuit diagrams, sketches/pictures of physical components and their operation, wireframes, etc.



Frame 1

Cyber Simulation

Training

Interactive Shell

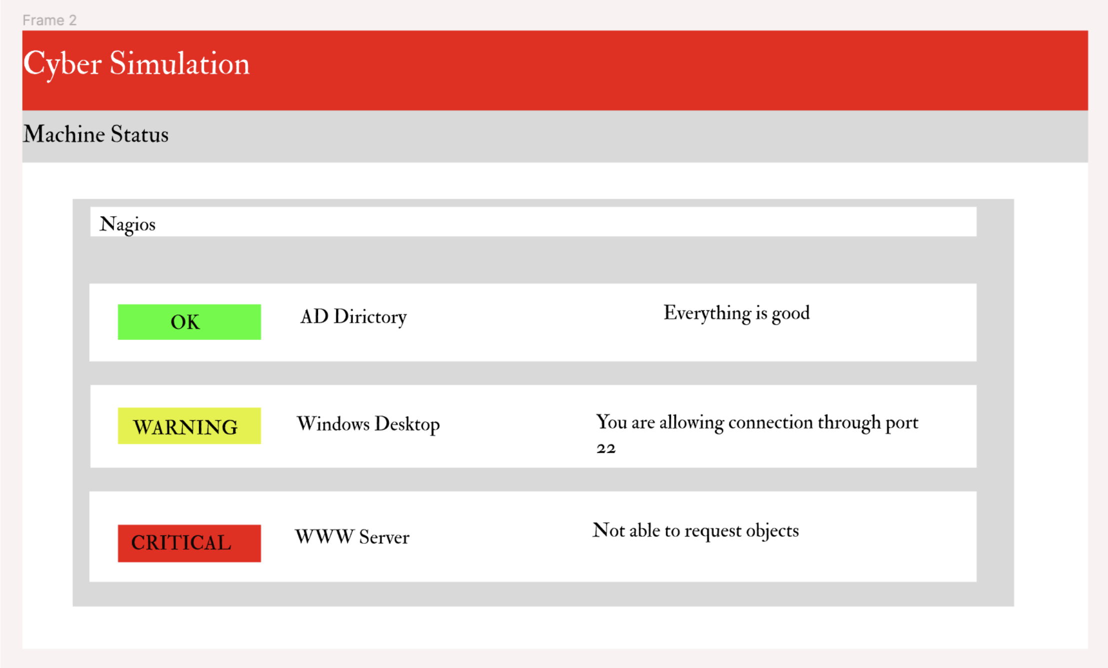
```
root@kali:-# nmap -st 1.1.1.1
root@kali:-#
```

Tutorial

In order to do a stealth scan:

- Use the -st flag
- IP Address of the machine that you are wanting to scan.

- There will be a training module so that users can use CySim as a practice environment. You will be able to learn both offensive and defensive security practices in this module.



Frame 2

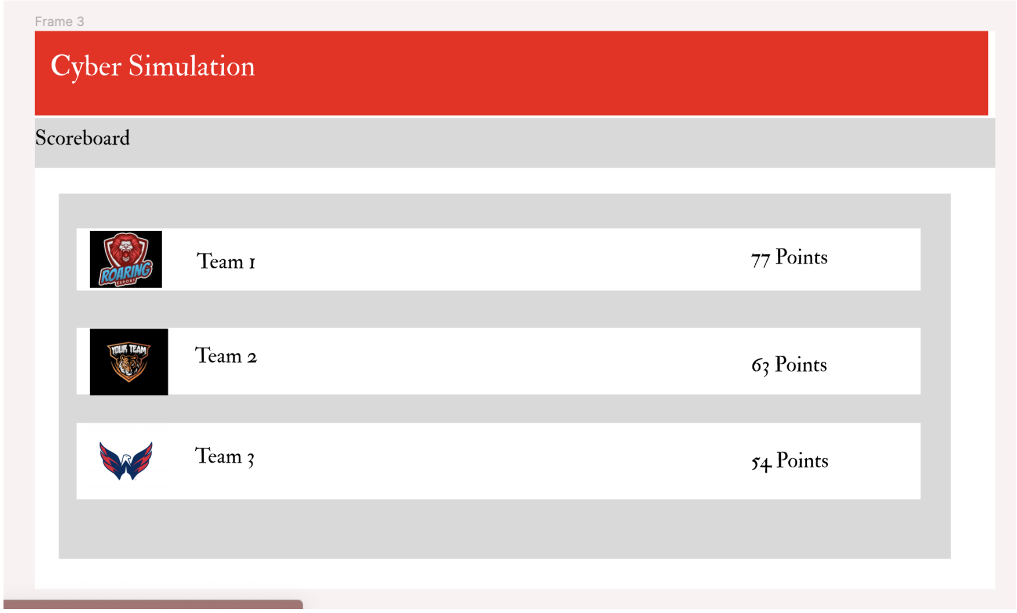
Cyber Simulation

Machine Status

Nagios

OK	AD Dirictory	Everything is good
WARNING	Windows Desktop	You are allowing connection through port 22
CRITICAL	WWW Server	Not able to request objects

- There will also be a status page to be able to see the status of the blue teams machines in one central location. The blue team will use this to know that their services are up and running, and the red team will use it to know which of the blue team's machines are remaining. Status scans will happen every ~15 mins.



Cyber Simulation

Scoreboard

Team 1	77 Points
Team 2	63 Points
Team 3	54 Points

- This is the leader board for all of the teams that are competing in the competition. The teams will be able to come to this module to see where they are standing in the competition.

4.3.3 Functionality

Describe how your design is intended to operate in its user and/or real-world context. What would a user do? How would the device/system/etc. respond? This description can be supplemented by a visual, such as a timeline, storyboard, or sketch.

CySim was created for a variety of purposes, these are shown below:

Provide mock real world experience for students, this means that students will have the ability to use CySim to get thrown into real world situations, from there these students will be able to learn how to prepare, respond and all around adapt based on the circumstances at hand.

Provide companies with the opportunity to train their employees, this means that if a bank is having an issue with accounts and data being stolen, they can send their employees to CySim and practice encountering these issues in a hypothetical experience. This allows these employees to better understand how to react and prevent these potential threats from occuring in the first place. All in all, this can be a big advantage and bonus to companies that utilize CySim.

CySim will function as an arena for cybersecurity competitions as well. This can vary from “Red, Blue, Green” to other situations, inside there will be spectators, a scoreboard displaying information and live results, as well as a press room.

4.3.4 Areas of Concern and Development

How well does/will the current design satisfy requirements and meet user needs?

Currently, our design for CySim meets the needs of our users because we have thought of the needs of all of the user types which includes Blue Team, Red Team-, Red Team+, and staff. If a user logs in as the blue team member, then they will be presented with everything they need. Like all of the machines' status, score, and availability.

If they were to log in as the Red Team-, these are the attackers that are using CySim as a training platform, and they will have a desktop with a Kali Linux to be able to access a terminal in metasploit. Their machine will be placed somewhere on the network to be able to run scans to find all the available machines that they have to attack and start their process at the reconnaissance stage.

The Red Team+ are the organization user's that can pass their scripted events. These types of attacks will not be man'd by a terminal. When a user is logging in as this, they will be able to see the scoreboard, blue team machine status page, and all the scripted events that they have available.

Based on your current design, what are your primary concerns for delivering a product/system that addresses requirements and meets user and client needs?

At this stage in the design process, we're most concerned with ensuring the stability of the services we implement so that as we build more modular game types on top of our foundation, we are able to grow CySim without having to start over from scratch any time a change comes down the line. This design will also have to be modular enough to support design changes over the next couple of years before CySim is launched.

What are your immediate plans for developing the solution to address those concerns? What questions do you have for clients, TAs, and faculty advisers?

To address these concerns, we are building the most basic components first and starting with what we know must absolutely be included. Infrastructure to support the VMs and hypervisor machines is an example of one of these critical components, and the work we achieve on the control server/central web server will pave the way for every game type and user interaction to come.

4.4 Technology Considerations

Describe the distinct technologies you are using in your design. Highlight the strengths, weakness, and trade-offs made in technology available. Discuss possible solutions and design alternatives.

As part of CySims development, we have decided or been required to use the following technologies:

- **Splunk:** Splunk will be used as another form of network monitoring in order to train participants on real-world network management practices during the exercises. Since it's an industry standard tool, it will be an effective inclusion in CySim.
- **pfSense:** pfSense is considered a universal, basic firewall tool that comes built-in with many Linux distributions. It will be used in CySim to train participants on the basic

concepts surrounding firewalls, while more advanced groups might use a different firewall system.

- **Nagios:** Nagios is a system monitoring software which can be used to monitor system status during an event.
- **Windows 10:** The biggest strength is Windows 10 being one of the largest and most popular OS's with a wide range of compatibility with many different software products. It is also free to use the basic version, cutting down costs. Some weaknesses include lack of customization and advanced features that other OSes include. Some alternatives include macOS as well as Linux distributions such as Ubuntu, Kali, or Mint.
- **Ubuntu Linux:** Some strengths of Ubuntu are that it is the largest distribution of Linux. This means that it has good compatibility with a wide range of software, as well as the great customization features which are a hallmark of the Linux operating system. It is also free, allowing for costs to be cut down. Some weaknesses include the lack of advanced tools built into the OS that a distribution like Kali or Parrot OS would have. Some alternatives of course would be Kali, Parrot, or Mint distributions of Linux.
- **.NET Core:** Some strengths of .NET is that it is fast, diverse, scalable, and has a lot of documentation. Some weaknesses include library constraints and overall lack of experience with the technology in our team. Some alternatives are Mono and Ceylon, but these are not as popular and less likely to be supported outside this project.
- **Server:** The main server that we're given will be multipurpose: It will host the web application that the red and blue teams interface with to participate in the game, as well as any other number of necessary services to drive the gameplay and provision the CySim field.

4.5 Design Analysis

Discuss what you have done so far, i.e., what have you built, implemented, or tested? Did your proposed design from 4.3 work? Why or why not? Based on what has worked or not worked (e.g., what you have or haven't been able to build, what functioned as expected or not), what plans do you have for future design and implementation work? For example, are there implications for the overall feasibility of your design, or have you just experienced build issues?

Our design in 4.3 was approved by our client, and we moved forward with the web app development, geared towards hosting it on our client's server. We're feeling confident that our web app design is up to our client's specifications.

We have been able to begin our .Net Core project and connect it to the MsSql database. We have also configured our web app to load different views based on the user's role type. The blue/ red team users are able to see their respected tutorials, scoreboard, and registration page.

5 Testing

Testing is an extremely important component of most projects, whether it involves a circuit, a process, power system, or software.

The testing plan should connect the requirements and the design to the adopted test strategy and instruments. In this overarching introduction, given an overview of the testing strategy and your team's overall testing philosophy. Emphasize any unique challenges to testing for your system/design.

In the sections below, describe specific methods for testing. You may include additional types of testing, if applicable to your design. If a particular type of testing is not applicable to your project, you must justify why you are not including it.

When writing your testing planning, consider a few guidelines:

- **Is our testing plan unique to our project? (It should be)**
- **Are you testing related to all requirements? For requirements, you're not testing (e.g., cost related requirements) can you justify their exclusion?**
- **Is your testing plan comprehensive?**
- **When should you be testing? (In most cases, it's early and often, not at the end of the project)**

5.1 Unit Testing

What units are being tested? How? Tools?

For unit testing, we have a couple of units that we think fit best here:

The first unit we would test would be the backend code:

Testing the functionality of the CySim scoreboard, to do this, we plan to implement a variety of different test cases throughout the code in order to check if each specific piece is working the way that it should be. Doing this will allow us to confirm that it is working properly and is outputting the correct values as need be. An example of this is that we should have a live tracker constantly checking the percentage finished for each team, if we go in and implement test cases to print out the values of the percentage we will be able to compare it with our actual thoughts that we had for this and determine if it's all functioning correctly.

The other unit we would be testing is the database behind the web application:

Testing the database behind the web application is crucial because this is very important to helping our application work properly. For example, we will use "mock emails" to create new users and input them into the database, from there we can have them go through and use the web application, and we should hopefully be able to track their status and progress in the database as well as information that they may provide. The mock emails would be a nice tool that we can use in order to accurately test the database unit of the web application.

5.2 Interface Testing

What are the interfaces in your design? Discuss how the composition of two or more units (interfaces) are being tested. Tools?

Our design will incorporate at least two interfaces, the scoreboard and the web app interface. These interfaces will be tested for understandability and accuracy, with the scoreboard being tested to ensure up-to-date information is being shown to participants in a timely manner. Testing the web app will involve lots of test cases and input handling to ensure that users can interact with the simulation's backend smoothly, likely incorporating unit testing tools like Mockito.

5.3 Integration Testing

What are the critical integration paths in your design? Justification for criticality may come from your requirements. How will they be tested? Tools?

For our groups critical integration paths we have multiple, these are, connecting the different portions of the score calculation in the backend to produce a status percentage, combining all the different pieces of the scoreboard together to display everything needed to be displayed, combining the backend logic to have the game function as it should, develop the database features for the web application so that it can be added to the web app and used properly and also implement an add and create user section of the web application that allows us to collect information about the users accessing and using the web application.

Below is how they will be tested:

Connecting the different portions of the score calculation in the backend to produce a status percentage: For this, we will implement markers and test cases within the code in order to track outputs and make sure that everything's adding up overtime.

Combining different pieces of the scoreboard together to display everything needed to be displayed: Doing this we will output all the pieces together in order to have a mock display that shows how it all is displayed on the screen, from there we will be able to see if it looks the way we intended it to.

Combining the backend logic to have the game function as it should: Similar to the score calculation, we will be using test cases in order to track outputs for this portion and make sure that it all looks correct.

Develop the database features for the web application so that it can be added to the web app and used properly: We will use "mock emails" to create new users and input them into the database, from there we can have them go through and use the web application, and we should hopefully be able to track their status and progress in the database as well as information that they may provide.

Also implement an add and create user section of the web application that allows us to collect information about the users accessing and using the web application: We will be able to use the database to track this and make sure that information is being collected and distributed properly so that there are no issues and the add and create users works properly.

5.4 System Testing

Describe system level testing strategy. What set of unit tests, interface tests, and integration tests suffice for system level testing? This should be closely tied to the requirements. Tools?

Some critical features include:

- Web Application Frontend Login, User, and Features (submit flags and view information)
- Web Application CySim Backend Integration (scoreboard and scenario control)
- Nagios Web Application Integration
- Networking and Access to Specified Machines

As said before, we will be creating mock users in our database of multiple teams/colors. These can be utilized by C# / Dotnet unit tests to ensure users have appropriate team view of the website. We can then further utilize these users as other features of the website are operating correctly. They can be utilized to submit flags, view team specific information, and test controls.

To test our system's backend integrations, such as the scoreboard system, we can utilize unit tests to ensure the scoring of each measured part is calculated correctly. Then we can create a series of mock values for all scoring fields and compare what they are calculated to and weighted as. Additionally, we can test white scoring influence by allowing white team users to modify how scoring is weighted.

To test the Nagios integration, we will be creating a test network/scenario consisting of mock blue team systems with Nagios system monitoring. The server will connect to the Nagios instance for service uptime information. With this information, we can test the system networking capabilities and Nagios integration functionality by confirming information received is what Nagios has stored. We can control the status of blue team services and test the blue team uptime and scoring systems by seeing if they match the appropriate outputs for a series of cases.

5.5 Regression Testing

How are you ensuring that any new additions do not break the old functionality? What implemented critical features do you need to ensure they do not break? Is it driven by requirements? Tools?

In order to ensure that new features don’t disturb previously added ones, we will employ the use of git version control, allowing us to develop in independent branches for testing and implementation of new features. We will require that any new features be implemented and fully tested in a separate branch before, and for another team to merge the branch if ready. Along with this, we can utilize git to integrate with various testing frameworks to automatically test code upon updates to a branch.

In the case of virtual machines systems we are building, we can utilize snapshots between each major update, ensuring that at all times there is at least one working backup of the machine being configured for CySim. As well, during the time of updates/development to the machine, we can have multiple instances of a VM running to ensure that other members are not interrupted if a server update does break some functionality.

5.6 Acceptance Testing

How will you demonstrate that the design requirements, both functional and non-functional, are being met? How would you involve your client in the acceptance testing?

To demonstrate that the laid out requirements have been met, we have been tasked with not just setting up the necessary systems, but also building a test scenario which will properly test each element of the project. This scenario is to encompass all elements of the project and be usable by our client. While the test may not be exactly what a final scenario would look like, it will be geared towards stressing systems and features that have been implemented. During this phase, we will also be in constant contact with our client to ensure that the project meets their expectations and requirements.

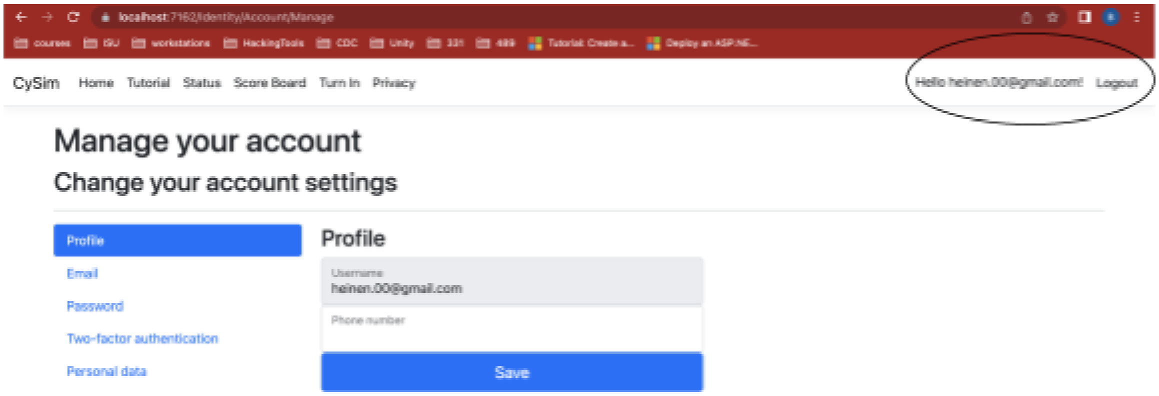
5.7 Security Testing

Test web applications to make sure that there are no sql injection, CRLF, and file path vulnerabilities. Also make sure that users passwords cannot be packet sniffed. We will use secure protocols for implementing the email server so that attackers are not able to see packets going across the network in plain sight.

5.8 Results

What are the results of your testing? How do they ensure compliance with the requirements? Include figures and tables to explain your testing process better. A summary narrative concluding that your design is as intended is useful.

For testing, we decided the best place for us to start would be to get the skeleton functionality of the website going. The first thing that we were able to do was create a login method mechanism for our website, we were able to use the .Net login. This gave us the ability to create a user, login as the user, and then we were able to view account settings. In account settings we were able to change password, change email, setup two-factor authentication, and download personal data.



Manage your account

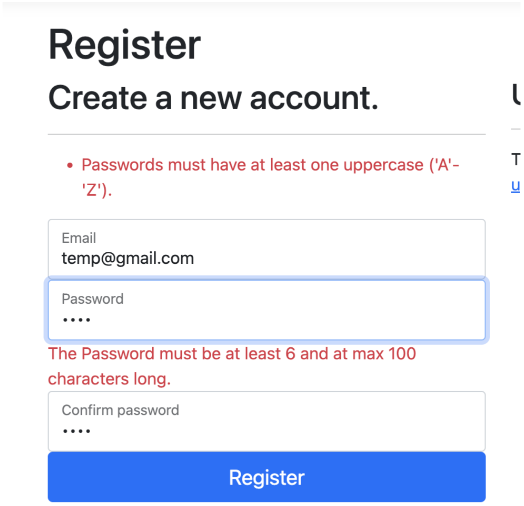
Change your account settings

Profile

Username: heinen.00@gmail.com

Phone number:

One neat feature that we tested completely is when you are creating a new user, the users password needs to be 6-100 characters long and you need to have one capital letter



Register

Create a new account.

- Passwords must have at least one uppercase ('A'-'Z').

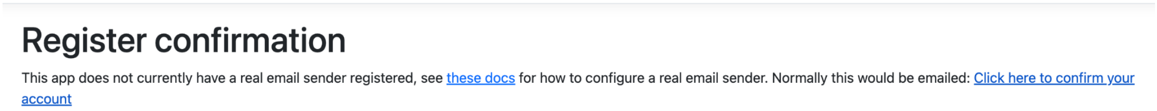
Email
temp@gmail.com

Password
••••

The Password must be at least 6 and at max 100 characters long.

Confirm password
••••

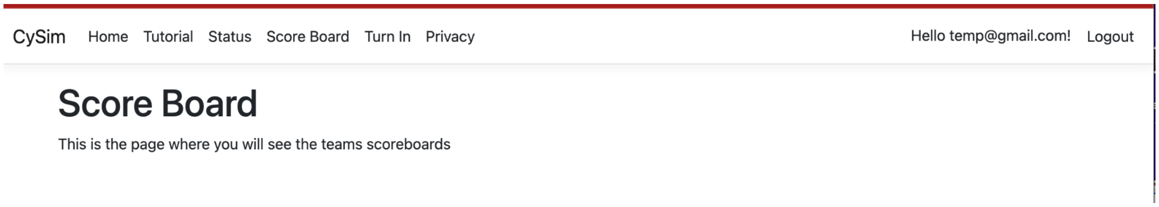
You will also be required to confirm your email account, right now you just click I accept, but in the future you will need to go to your own email and confirm.



Register confirmation

This app does not currently have a real email sender registered, see these docs for how to configure a real email sender. Normally this would be emailed: Click here to confirm your account

Another thing that we created and tested was the tab bar and a user is able to go to the following tabs home, tutorial, status, score board, turn in, and privacy. And currently, each of these tabs will just load dummy pages.



Score Board

This is the page where you will see the teams scoreboards

I also created a working scoreboard, but can not get a picture at the moment. The scoreboard logic will work off of query logic where you search your data and get the data and descending order with the highest score being the first element. And you can select the number of entries that you wish to show, like 10 or 15 for example.

This is useful for our design because this is essentially the skeleton of our project. Our entire project will be driven by the web app. This is how participants will be able to access their machines, see their status in the game, and much more. We now will just have to focus on more technical aspects and can focus less on the route.

6 Implementation

Describe any (preliminary) implementation plan for the next semester for your proposed design in 3.3. If your project has inseparable activities between design and implementation, you can list them either in the Design section or this section.

- Microsoft Identity Touch Up(January 5 - January 15)
  - Small touch up to make the username display after a user logs in instead of the whole email.
  - Also, verify user logs using email instead of username.
- Scoreboard (Dec 1 -Jan 5)
  - Complete the module to be able to read from the database and order by descending using the score entity.
  - Show scores for both Red and Blue teams.
- Tutorial (Dec 20 - May 5)
  - Complete the module for the red and blue to be able to view tutorials.
- Dev Express (Feb 12 - March 5)
  - This will be used to view our tutorials through a report manner.
- Team Registration (March 1 - Apr 5)
  - This is the module where users will be able to create or view available teams.
  - Blue and Red Team user's will have the ability to join these teams.
- Backend (Jan 5 - Apr 15)
  - We will be getting our services online on our server, and incorporating VMs
- Networking and Integration (Dec 1 - May 5)
  - Setting up the CySim network environment to connect VMs and our backend code
- Debugging (Dec 1 - May 5)
  - As we implement more technologies and scripts, we will be constantly debugging code and rooting out network issues as we find them
- Logger Implementation (March 5 - March 18)
  - Here is where we can build a real world application for CySim
  - CySim will have the ability to be able to print their log information to whatever file that they want.
- Server Functionality (Feb 21 - May 5)
  - We will host the web app and CySim network to the Coover server.
- Email Service (Apr 1 - May 5)
  - User verification to make sure that the user's registering to work with CySim are real people and can verify their account.
  - This will also be useful if the user happens to forget their password.

7 Professional Responsibility

This discussion is with respect to the paper titled " Contextualizing Professionalism in Capstone Projects Using the IDEALS Professional Responsibility Assessment", *International Journal of Engineering Education* Vol. 28, No. 2, pp. 416–424, 2012

7.1 Areas of Responsibility

Pick one of IEEE, ACM, or SE code of ethics. Add a column to Table 1 from the paper corresponding to the society-specific code of ethics selected above. State how it addresses each of the areas of seven professional responsibilities in the table. Briefly describe each entry added to the table in your own words. How does the IEEE, ACM, or SE code of ethics differ from the NSPE version for each area?

Area	How IEEE Addresses this Area
Work Competence	This is addressed in section 6 of the IEEE code of ethics. It explains the need to continue improving technical competence while only undertaking tasks or work if we are qualified or to disclose limitations. This differs from NSPE as it doesn't outright say to avoid deceptive acts, but the IEEE code implies it in saying to fully disclose your limitations.
Financial Responsibility	This is addressed in sections 4, 5, and 9. It states to avoid bribery, acknowledge and correct errors, be honest in claims based on available data, and avoid injuring an individual's property. Besides the bribery, the others state the responsibility of an engineer to deliver reasonable products and to not "injure" the client's property which I think can be extended to reasonable costs. Unlike the NSPE, I feel the IEEE code of ethics doesn't cover Financial Responsibility as directly. It's much more dispersed and implied.
Communication Honesty	This is addressed in multiple sections, including 1, 2, 3, 5, and 6. The code of ethics explains that we are to promptly disclose factors that may be dangerous, inform society about the capabilities and implications of technologies, honestly disclose conflicts of interest, be honest in criticisms, and finally, fully disclose limitations in work competency. Compared to the NSPE, the IEEE code of ethics, in my opinion, does a better job of giving many examples of honest communication instead of the more general warning to avoid deception as in the NSPE.
Health, Safety, Well-Being	This is addressed in section 1 of the IEEE code of ethics. It explains that as engineers, we are to hold paramount the public's safety, health, and welfare. It is also mentioned in section 9 to avoid injuring others. This is addressed here pretty much exactly as it is in the NSPE. The NSPE goes into more

	detail addressing situations where one may have to report a situation to the proper authority if the Engineer believes the situation may endanger people.
Property Ownership	This is addressed in section 9 of the IEEE code of ethics. Here it states that we are to avoid injuring others, their property, etc. Section 5 also states to credit properly the contributions of others. The NSPE, in my opinion, puts a bit more into respecting property and ideas than the IEEE code. But the IEEE code of ethics does mention these two factors. The IEEE also doesn't mention specifically the employer's property.
Sustainability	This is also addressed in section 1 of the IEEE code of ethics. The IEEE code of ethics states that we must strive to comply with sustainable development practices and promptly disclose factors that may affect said environments. Compared to the NSPE, the IEEE makes it a bit more of a must by stating to strive to comply with these practices than the NSPE which says engineers are encouraged to adhere to these principles. However, unlike the IEEE, NSPE defines sustainability.
Social Responsibility	This is addressed in section 2 of the IEEE code of ethics. Here it states that we should strive to improve society's understanding of the capabilities and implications of new technologies. In section 4, the code also explains to avoid unlawful conduct. Sections 7, 8, and 9 also address societal issues such as discrimination and harassment. Compared to the NSPE, it does say many of the same things but doesn't get into as much depth about the different forms of discrimination. Overall it is very similar.

7.2 Project Specific Professional Responsibility Areas

For each of the professional responsibility area in Table 1, discuss whether it applies in your project's professional context. Why yes or why not? How well is your team performing (High, Medium, Low, N/A) in each of the seven areas of professional responsibility, again in the context of your project. Justify.

Below is our evaluation of how much each area applies to our project in professional context:

Area	Level of Importance
Work Competence	I would say this is a high level of importance. This is because, for this project to succeed, we need a high level of competence in a wide range of Cyber Security and Software development skills. This project is heavy on these aspects, and we need a strong understanding to deliver the best product possible.
Financial Responsibility	I would say this is a low level of importance. This is marked as low because we don't have a budget or funding for this project. While this project may receive future funding, it will not be necessary for our portion, which only needs a server already in existence.
Communication Honesty	I would say this is a high level of importance. This is because we need to communicate honestly with our clients and ensure that we are truthfully reporting what we have completed. This is important for this project as it gives them an idea of what to expect as a final product.
Health, Safety, and Well-Being	I would say this is a low level of importance. The reason for this is that due to the nature of what we are doing, there is very little risk to the health, safety, and well beings of individuals.
Property Ownership	I would say this is of medium importance. This is because we are working with a technology that our client has developed. We are not to disclose any specific information about its inner workings and are to respect the client's property and designs.
Sustainability	I would say this is of low importance. While this project has the potential to take a large number of resources in terms of energy for running the needed system, there is little that we can do to decrease this need. While we plan to consider it, I don't see it as a chief concern due to the singular resource we will be using.
Social Responsibility	I would say this is of medium importance. The reason for this is that due to the field we are working in as a part of this project, namely cybersecurity, there is the potential for unlawful actions. Also, there is the aspect of this being in a possible physical

	location and employing individuals. Because of this, many social factors, such as sexual harassment, unfair treatment, or discrimination, will need to be addressed.

Below this is our evaluation of how our team is performing in each of the seven areas:

Area	Current Performance
Work Competence	I would say our current level of performance is medium. This is because there are systems we must work with, such as virtualization and web applications, that we are not fully confident in using.
Financial Responsibility	N/A - As mentioned before, this project has no funding.
Communication Honesty	I would say our current level of performance is high. This is because we submit truthful weekly reports to our clients and meet with them regularly to discuss design, implementation, and issues we have encountered.
Health, Safety, and Well-Being	N/A - As mentioned before, there is very little chance of affecting an individual's health, safety, or well-being.
Property Ownership	I would say our current level of performance is high. This is because we have continued to respect our client's ideas and intellectual property relating to the systems that are currently in use by our team.
Sustainability	N/A - As mentioned earlier, this is out of our control due to the nature of the project.
Social Responsibility	I would say our current level of performance is low. This is because we haven't considered methods of preventing the illegal use of this project nor informing and ensuring that our client and society know the possible risks of such a project.

7.3 Most Applicable Professional Responsibility Area

One important responsibility is work competence. This means being versed in cybersecurity applications and software development. While it currently sits at medium due to a lack of competence with some tools and skills, such as virtualization and management of virtual machines. However, throughout our discussions with our client, we have shown a large degree of competence in other areas, such as general cybersecurity and tools used in the industry. This has helped us understand the project's goal and some features our client wants to include in the final product. We also have a high degree of competence with things such as Linux and scripting. Because of this, we have had a smoother workflow with fewer issues occurring and more control over the project. Many team members also have networking experience, which has helped us develop an overall network design and networking maps. Because of this, we have been able to communicate network-related requirements and features more effectively with our client. We have also been able to better identify and set up the most efficient network for this project.

8 Closing Material

8.1 Discussion

Discuss the main results of your project – for a product, discuss if the requirements are met, for experiments oriented project – what are the results of the experiment, if you were validating a hypothesis – did it work?

- We were able to successfully get the identity working and allow a user to get into our application.
- We were also able to successfully get our views to load based off of user roles.
- We were able to get a scoreboard working in the web application
- We were able to get the route done for the tutorial, scoreboard, team registration, and home modules.

8.2 Conclusion

Summarize the work you have done so far. Briefly re-iterate your goals. Then, re-iterate the best plan of action (or solution) to achieving your goals. What constrained you from achieving these goals (if something did)? What could be done differently in a future design/implementation iteration to achieve these goals?

- Getting the database to start working with the project
    - We are using MsSql Server
- Customizing the user registration to assign user role to the account to load the correct web app views
    - identity is very complex and touchy so very little things wrong will cause the whole project to not work.
- Getting the scoreboard was tricky because we needed to read all of the data from our database and then display it by descending order.

8.3 References

List technical references and related work / market survey references. Do professional citation style (ex. IEEE).

[1] *Communications Specific Tabletop Exercise Methodology.* Department of Homeland Security, 2012.https://www.cisa.gov/sites/default/files/publications/CommunicationsSpecificTabletopExerciseMethodology_0.pdf.

[2] *Cybersecurity Simulation (CySim) Training Center: A Cyber Sports Complex.* Iowa State University. https://www.cyio.iastate.edu/cysim.

8.4.1 Team Contract

Team Name ____sdmay23-11____________________

Team Members:
1) _Matthew Daoud______________ 2) _Ethan Swan____________________
3) _Jacob Boicken______________ 4) _Brady Schlotfeldt________________
5) _Bailey Heinen_______________

Team Procedures

1. **Day, time, and location for regular team meetings:** Virtual, Tuesday@ 6:30 (Depending on availability)
2. **Preferred method of communication updates, reminders, issues, and scheduling:** Discord
3. **Decision-making policy :** Consensus and Compromise if necessary
4. **Procedures for record keeping:** Ethan Swan as main minute taker. Document in shared drive.

Participation Expectations

1. **Expected individual attendance, punctuality, and participation at all team meetings:** The team should be made aware ahead of time if someone is not going to be at a meeting or on time to one, and it is generally expected that each team member will give some form of input during meetings, especially on team decisions.
2. **Expected level of responsibility for fulfilling team assignments, timelines, and deadlines:** If you commit to a portion make sure you are meeting deadlines and asking questions if stuck. Timelines can be flexible, just need to communicate.
3. **Expected level of communication with other team members:** Update team during weekly meetings, if you have questions or concerns, reach out in discord as soon as possible.
4. **Expected level of commitment to team decisions and tasks:** Expected that decisions are at least recognized even if not fully accepted by an individual. Expected to perform the best they can on a task.

Leadership

1. **Leadership roles for each team member (e.g., team organization, client interaction, individual component design, testing, etc.):** Each team member gets equal say in decisions, and tasks will be distributed fairly and according to each member's ability. Individual work should be done according to what the team agreed upon ahead of time, or at least be consistent with what is discussed.
2. **Strategies for supporting and guiding the work of all team members:** Ask if help is needed if a member seems to be struggling. Don't force opinions or methods down on someone. Be kind and provide constructive criticism. If help is needed it can become a point in a team meeting or have a meeting to resolve it. Something should be put in the team chat.
3. **Strategies for recognizing the contributions of all team members:** Recognition in team meetings. The goal would be for the group to acknowledge the progress and make the team member aware that the group is happy with how much progress is being made.

Collaboration and Inclusion

1. **Describe the skills, expertise, and unique perspectives each team member brings to the team.**
    a. **Ethan**: As a Cyb E major, I have academic experience with network infrastructure and security, programming, web development, and group projects. I also have hands-on experience with developing security products from my internship, and am used to long-term project goals and development cycles.
    b. **Jacob**: As a Cybersecurity engineer, I have academic experience working with all layers of networking, system administration, low and high level programming, and team projects. I have hands-on experience working within a virtual SOC during my internship. I also have hands-on experience configuring VLANs, virtualization/containerization management, and Unix-like operating systems (to a kernel level).
    c. **Matthew**: Computer Engineering major with a cyber security minor. I have experience with networking, basic cyber security tools, linux, virtualization, and programming.I also have group project experience.
    d. **Bailey**: As a Cybersecurity Engineer, most of my experience is software security and software development. I have 1 year experience in the workplace and have raised corporations' security scores and have completed numerous developing projects. For cybersecurity networking I have Iowa state class experience and have done 3 CDC's.
    e. **Brady**: As a computer engineering major, I have a lot of experience with working frontend and backend of projects as well as operating within a team, good communication and work ethic. I also have a minor in cyber security so I have a solid background on security as well.

2. **Strategies for encouraging and supporting contributions and ideas from all team members:** Hear out everyone's ideas and brainstorm future steps together, provide constructive feedback, and take note of each other's thoughts.
3. **Procedures for identifying and resolving collaboration or inclusion issues (e.g., how will a team member inform the team that the team environment is obstructing their opportunity or ability to contribute?):** Close out every meeting to make sure everyone is doing good, and they are happy with what they are doing. If there is an inclusion or collaboration issue we will want to be on top of it fast. If worse comes to worse we will be able to move people around.

Goal-Setting, Planning, and Execution

1. **Team goals for this semester:**
    a. To form a solid basis for a project we can all be proud of
    b. To each contribute ideas and work that satisfy the client
    c. To generate at least a "proof of concept" that is detailed and sophisticated enough to set us on track to complete our design project next semester.
2. **Strategies for planning and assigning individual and team work:**
    a. Everyone will take on/receive tasks fairly according to their current workload and skills
3. **Strategies for keeping on task:**
    a. Prioritizing quick/less complicated tasks first
    b. Updating each other on progress at each team meeting
    c. Sending progress reports in Discord when tasks are completed

d. Pushing to Git if code is written

Consequences for Not Adhering to Team Contract

1. **How will you handle infractions of any of the obligations of this team contract?**
   Infractions will be handled during meetings by attempting to find compromise to the issue at hand.
2. **What will your team do if the infractions continue?**
   If possible, it will be handled by the team. If not, the relevant professor or contact will be alerted.

****

a) *I participated in formulating the standards, roles, and procedures as stated in this contract.*
b) *I understand that I am obligated to abide by these terms and conditions.*
c) *I understand that if I do not abide by these terms and conditions, I will suffer the consequences as stated in this contract.*
1) Brady Schlotfeldt DATE 09/19/2022
2) Bailey J Heinen DATE 09/19/2022
3) Ethan Swan DATE 09/19/2022
4) Jacob Boicken DATE 09/19/2022
5) Matthew Daoud DATE 09/19/2022