

# CySim

FINAL PROJECT REPORT

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 - Full Stack Developer

Matthew Daoud - Full Stack Developer

Email: [sdmay23-11@iastate.edu](mailto:sdmay23-11@iastate.edu)

Website: <https://sdmay23-11.sd.ece.iastate.edu>

Revised: April 29 2023 / Version 1.3

## **Revised Project Design:**

In 491, we sought to specify and implement our rough designs for CySim's network architecture, scoring systems, and game automation. We were originally going to start development with a server to host services like Serilog to store logs, Nagios to power our scoring functionality, and an email server to connect with users. Due to the logistics of acquiring said server, there would have been delays resulting in less time to develop actual functionality.

We worked with our advisor on defining the scope of our work, and we agreed to narrow down the deliverables this semester to match with the time and resource constraints, as we only had two semesters and no budget for the largest portions of CySim's specification. We decided that it would be best to choose one component that we could begin work on without delay, and to flesh it out as much as possible. Therefore, our primary focus this semester has been the CySim web application, which is the primary user interface of the project.

Its functional requirements are to connect users and teams with CySim's infrastructure, and to store and present data like scores and user/game information. Its non-functional requirements include visual appeal and informative content, like tutorials that we designed ourselves. There aren't any particular standards that are relevant to this stage of the project, except for those already utilized by the .NET Core Framework. In future iterations of the project, developers will need to conform with networking protocols including IEEE 802.2, the layers of the OSI model that incorporate TCP/IP. CySim may also need to conform with IEEE 802.11, the standard for WiFi and LAN networking, when the physical location's network is set up.

There are also no relevant physical security threats at this stage of CySim's implementation, and we have addressed cybersecurity threats by incorporating strict error-handling and identity management to prevent any disallowed or exploitative actions in the web-app.

While we don't have CySim's full network assets put together, we have delivered a key component of the final implementation to streamline the process for future teams to get CySim online.

## **Implementation:**

For our web application, we decided to use the .Net Core Framework as it is able to be used on Mac, Windows and Linux. We have team members that are using all of these OSes, so it was crucial for us to find a Framework that would work for us all. Another plus to using this Framework is it is roughly 4x faster than the .Net Framework and there is an abundant amount of documentation that can help along the development process. We used MS SQL Server as our

database to be able to effectively communicate with our website. We used Azure Data Studio and SQL Management Studio to be able to view, edit and delete our data in a centralized platform. We also choose to use the MVC pattern to develop our application. That way we could easily control our endpoints, views and models in a controlled manner.

We needed many packages to work with our application like Microsoft Identity to be able to ensure that our users would have a secure way to register and login to our application. As this library uses AES encryption giving user confidentiality on their data as well as gives you a great start to control user operations. Users have the ability to change their profile pictures, passwords, usernames, and add other personal information. We needed the Entity Framework to be able to more effectively communicate with the data. This allowed us to not need to write much data access code, instead we could just inject our context into all of our classes and access the database through the DbContext.

```
public class TeamRegistrationController : Controller
{
    private readonly ILogger<TeamRegistrationController> _logger;
    private readonly ApplicationDbContext _context;
    private readonly SignInManager<IdentityUser> _signInManager;
    private readonly RoleManager<IdentityRole> _roleManager;
    private readonly UserManager<IdentityUser> _userManager;

    public TeamRegistrationController(ILogger<TeamRegistrationController> logger, ApplicationDbContext context, SignInManager<IdentityUser> signInManager,
        RoleManager<IdentityRole> roleManager, UserManager<IdentityUser> userManager)
    {
        _logger = logger;
        _context = context;
        _signInManager = signInManager;
        _roleManager = roleManager;
        _userManager = userManager;
    }

    [HttpGet]
    public IActionResult Index()
    {
        return View(_context.TeamRegistrations.ToList());
    }
}
```

From the image above you can see that we are passing a ApplicationDbContext into our TeamRegistration class and having our own local context to be able to read from that. As in the index get method we are reading TeamRegistrations from our context to be able to return it in a list so that when you access the index page you would be able to see all of the teams that already exist in our app. We are also using other managers like SignInManager, UserManager, and Logger. We are using the sign in manager to be able to control users' sign ins to the application. This is useful so we are able to refresh the users views when they join a team so that they will not be able to see the Join button after they have registered to a team. We are using the user manager to be able to control the users of our application and this is useful to be able to add roles to an identity user through code. The logger is used to log events that happen in the application. It can be useful to see output in the application to understand the behavior of the running process without needing to set breakpoints.

## Sign In Manager

```

IdentityUser identityUser = _context.Users.Where(x => x.Email == User.Identity.Name).First();
await _signInManager.RefreshSignInAsync(identityUser);
_context.Add(registration);
_context.SaveChanges();

```

## User Manager

```

if (User.IsInRole("Blue Team") || User.IsInRole("Red Team"))
{
    IdentityUser identity = GetIdentityUser(registration.TeamCreator);
    await _userManager.AddToRoleAsync(identity, "Team User");
}

```

## Logger

```

if (file == null)
{
    _logger.LogError("Scenario Create: No file was uploaded");
    TempData["errors"] = "No file was uploaded";
    return RedirectToAction(nameof(Create));
}
if (Description == null)
{
    _logger.LogError("Scenario Create: No Description was entered");
    TempData["errors"] = "No description was provided";
    return RedirectToAction(nameof(Create));
}

```

Throughout development we have also been cautious about security measures by using data attributes to guarantee that a model entity can only be a certain number of characters long and endpoints can only be accessed by signed in users or specific role types. We have also been parameterized queries anywhere we are reading data directly from the database by SQL connections.

## Data Attributes

```

[MaxLength(512)]
[DisplayName("Profile Picture")]
public string FilePath { get; set; }

```

```

[HttpPost]
[Authorize(Roles = "Red Team,Blue Team")]
public async Task<IActionResult> Join(TeamRegistration teamRegistration)

```

## Parameterized Query

```
private static string FindUser(string Email)
{
    string connectionString = "Server=localhost;Database=CySim;User Id=sa;Password=cpre491p@ssword;Trusted_Connection=True;MultipleActiveResultSets=true;Integrated !
    string returnUser = null;
    if (Email != null)
    {
        using (SqlConnection connection = new SqlConnection(connectionString))
        {
            connection.Open();
            using (SqlCommand command = new SqlCommand("", connection))
            {
                command.CommandText = "SELECT * FROM [CySim].[dbo].[AspNetUsers] WHERE Email=@Email";
                command.Parameters.AddWithValue("@Email", Email);
                SqlDataReader reader = command.ExecuteReader();
                while (reader.Read())
                {
                    returnUser = reader["Email"].ToString();
                }
            }
            connection.Close();
        }
    }
}
```

Lastly we used Github to control all of our code so that we could all work simultaneously on different branches and be able to easily merge all of our code after we tested it to make sure that it worked as expected.

## Testing Process and Results:

For unit testing the CySim web application, only the controllers were tested by writing test code using xUnit test framework and Mockito to create false versions of modules used in the controller. Only the controllers are tested, as they are the only part of our code that handles backend logic in our application. xUnit tests allows automating using multiple inputs and testing if certain conditions are met, typically the output from our controller. In development, these tests have been used as part of regression testing as they run when we push code to our git repo to ensure that key controller functionality is not broken.

For user interface testing, the key pieces we tested are the views of the web application. We manually handled ensuring the application's front end page was rendered correctly, had the correct information, and displayed errors as appropriate.

For integration and system testing, this was mostly done manually by each of our team, testing that the .NET web server modules and MSSQL database were working together. For the integration tests, we started a version of our web server locally using an in-memory database loaded with testing data instead of the actual MSSQL server. We then could ensure that all requests and modules to the web server functioned correctly and retrieved the correct data. For system testing, each developer did this for their own parts of the application. They would test the

key functionality of the pages they developed and ensure that values from the database are being pulled or updated. This would mean that you may add a new entry through a web form in the app and then view the database entries to ensure that all data was loaded correctly, and no other data was invalidated. This was also used for regression testing as when we merged code, we would run through these tests to ensure no broken changes.

For acceptance testing, we demoed many versions of our web application with our client, Doug Jacobson. We showed key features and design elements of each page of the website. This served as a key point of updating and modifying the requirements of the application. One key feature added was modifying our admin view, so they can differentiate red and blue team elements of the website. Other than adding functionality, this was to confirm that the website handled the key needed that Doug Jacobson had for the application.

Finally, for security testing, very high level and basic tests were run that analyzed the code we have written or the application while it was running. To analyze our code, we ran it through CodeQL in GitHub. This only brought up false alarms in the jQuery JavaScript plugin that we had, as its text is all condensed into one line. To scan the web application, we ran it through a HAProxy web scan. This simply requests all the information from each page, finding issues with in JavaScript or HTML and how the web server responds to requests. This has a few mild issues being that certain HTML headers are not set such as anti-clickjacking, content security, and content type. These issues will be given to our client for future development of the web application, along with other security recommendations we have.

## **Related Products and Literature:**

Since CySim is going to be a next generation cyber sports complex, there are many existing examples of cyber events that we can look to for inspiration and guidance.

- Hack the Box
  - Hack the Box is a popular online cybersecurity course focused around allowing the user to use relevant tools in a safe environment. While it is very effective at allowing learners to gain some knowledge, it does not provide the in-person, hands-on experience that CySim is designed for. We take inspiration from its use of tutorials at all skill levels and its online interactivity.
- Cyber Defense Competitions, or CDCs
  - Like many other universities and organizations with a focus on technology, Iowa State has their own CDCs that they run throughout the year. CySim is designed

with these events in mind, and intends to expand the level of immersion and teamwork that players can take part in. While many CDCs in Iowa State take place in the Coover TLA or in disparate home-bases, CySim seeks to unify the experience into a venue that emulates the physical and technical environment that players would most likely find themselves in during a real cybersecurity attack.

- DEF CON
  - DEF CON is a convention that occurs once a year in Las Vegas that is built around a variety of different hacking competitions.
  - Some of the competitions at this convention are modeled on real world threats, which teams will have to address in the least amount of time possible. CySim is also based on real-world threats, and teams will be scored on how they can maneuver and respond to a given scenario.
- Communications-Specific Tabletop Exercise Methodology (US DHS)
  - CySim's core gameplay ethos is designed to expand on the principles of traditional tabletop exercises. These are interactive experiences that simulate the push and pull of dynamic threats in any sector, and in this case, cybersecurity. They also have a strong emphasis on communications among the team members and outward to a public audience.
  - The US Department of Homeland Security has published a guide that was passed to us by our client/advisor that specifies a gameplay process to facilitate this dynamic experience. It specifies activities like warm-up exercises, planning meetings, debriefing sessions, and conferences during and after the exercises to maximize the learning experience. It also suggests ways to engage and energize participants.
  - Our portion of CySim has been built with these activities in mind, which is why it was important for us to begin our web-app design by ensuring future teams have the room and capabilities to expand on these core principles as the design is passed on and iterated.

## **Appendices:**

### **Appendix I - Operation Manual:**

#### **Appendix I - Section 1: How to Setup and Deploy CySim:**

This guide will explain how to install and operate CySim locally on a Windows based desktop system. Mac and Linux may require different steps to properly setup and deploy an instance of CySim. Please keep in mind that as the required programs continue to be updated steps and commands listed here may change. Any commands listed in this document will be encased within quotes. Do not include these outer quotes unless explicitly stated. All figures for this section can be found in the section titled “Figures Relating to CySim Setup and Startup”.

#### **Required Installations and Downloads:**

This guide will expect for the following programs to be installed going forward. Please reference developer documentation when installing the following software:

- Microsoft Visual Studio (2022)
  - Installation Notes: Be sure to install the “.NET desktop development”, “Data storage and processing” and “ASP.NET and web development” packages.
  - Download: <https://visualstudio.microsoft.com/>
- Docker Desktop
  - Download: <https://www.docker.com/>
- SQL Server Management Studio 19
  - Download: <https://learn.microsoft.com/en-us/sql/ssms/download-sql-server-management-studio-ssms?view=sql-server-ver16>

This guide also expects the following downloads:

- A Copy of the CySim .NET source code
  - Download instructions can be found in Appendix IV - Code and How to Obtain

#### **Initial Database Setup:**

CySim requires an mssql server to host the database for the storage of information such as users, uploaded documents, team registration data, and other information required for CySim to



properly function. The steps below explain what will be needed in order to properly setup this mssql server.

This section assumes:

- The user understands the basic usage of Windows Powershell
- The user has installed Docker Desktop
- The user has installed MSSQL Server Management Studio 19

Steps:

1. Begin by opening a Windows Powershell terminal
  - 1.1. Enter and run the following command in order to have Docker download the necessary mssql files: `"docker pull mcr.microsoft.com/mssql/server:2019-latest"`
    - 1.1.1. Once installation is complete as indicated by powershell, open the Docker Desktop application and ensure under the "images" tab (Fig. 1.1) that the proper mssql server has been downloaded.
  - 1.2. Enter and run the following command in order to create a mssql 2019 Docker container with the proper credentials: `"docker run --name CySim -e "ACCEPT_EULA=Y" -e "SA_PASSWORD=cpre491p@ssword" -e "MSSQL_PID=Developer" -p 1433:1433 -d mcr.microsoft.com/mssql/server:2019-latest"` Explanation can be found in section 1.2.2.
    - 1.2.1. Once the command has finished running, navigate to the "Containers" tab (Fig. 1.1) in the Docker Desktop application. Ensure a new container named "CySim" has been created.
    - 1.2.2. This command will create an instance of a mssql container in Docker. It will also accept the EULA, set the proper credentials, set the proper PID value, and finally bind it to the proper port.
  - 1.3. To start or stop the mssql server container in the Docker Desktop click the start/stop container button shown in figure 1.1. Ensure it is running before navigating to the next part of this guide.
2. With the creation of the server complete, the security settings for the mssql server must be updated to allow the SQL Server Authentication. To begin, open the SQL Server Management Studio 19 application.
  - 2.1. A popup window should appear like that shown in figure 1.2. Ensure authentication is set to "Windows Authentication" and click the connect button. figure 1.2 illustrates what these settings will look like.
  - 2.2. Right click on the SQL Server Header (Fig 1.3). Navigate to the properties option and click here. Navigate to "Security" (Fig 1.5) and ensure that the "SQL Server and Windows Authentication Mode" is selected. Click the "Ok" button.
  - 2.3. In order to ensure these changes took effect, right click on the SQL Server Header in the "Object Explorer" (Fig 1.3) and click disconnect. In the main SQL Server Management window, navigate to "File" and then click "Connect Object Explorer". A window like that in figure 1.2 will appear. Ensure authentication is set to "SQL Server Authentication". The login username and password is shown below in section 4. Click the connect button.

If the database connects, proceed to section 3. Otherwise this issue may be related to troubleshooting section 1.

3. The next part explains the initial CySim database creation. Ensure connection to the database is established similar to what is described in section 2.3 above.
  - 3.1. Once properly connected to the server a database will need to be created. In the “Object Explorer” on the left side of the SQL Server Management Studio main window right click “Databases” as shown in figure 1.3. Click the “New Database” option and a new small window should appear.
    - 3.1.1. In this Window shown in figure 1.4, enter the database name as “CySim” and click the ok button to create the database.
      - 3.1.1.1. Ensure this database has been created by clicking to expand the “Databases” drop down menu and checking for “CySim” (Fig 1.3).
4. This concludes the necessary setup for the CySim database. The database should now be ready to have tables added and store CySim data. The default credentials of this server are a username of “sa” and password of “cpre491p@ssword”. Proceed to the next section.

## **Visual Studio CySim Setup and Database Connection:**

Visual Studio is required to run CySim locally. The steps below explain what will be needed in order to properly setup CySim in visual studio and how to connect the database for demo purposes.

This section assumes:

- The user has installed Microsoft Visual Studio (2022)
- The user has setup the MSSQL server through the steps in the previous section
- The user has obtained a copy of the CySim .NET source code
  - Download instructions can be found in Appendix IV - Code and How to Obtain

Steps:

1. Begin by opening Microsoft Visual Studio (2022).
  - 1.1. Ensure that the CySim .NET source code has been unzipped and it where the user wants it to be. Make note of this location.
  - 1.2. In the Microsoft Visual Studio landing page (fig 1.6) navigate to “Open a project or solution”. Inside the file explorer navigate to where the CySim .NET source code is located and enter the CySim folder. Select the “CySim.sln” folder and click open.
2. With the project now open in Visual Studio, new database migrations will need to be created to ensure proper operation.
  - 2.1. Once opened in Visual Studio, navigate to the “Solution Explorer” (fig 1.7). Find the “Migrations” folder and expand this. Ensure that all existing files are deleted.
  - 2.2. Navigate to the search bar near the top of the Visual Studio project window and search for “Package Manager Console”. Select this option.

- 2.2.1. A console should appear near the bottom of the Visual Studio Window shown in figure 1.8.
  - 2.3. In this console enter the following command to create the migrations necessary for the database and web application: “Add-Migration CySimMigrations”.
  - 2.4. Once the migrations have been successfully created enter the following command to update the database: “Update-Database”.
    - 2.4.1. If errors occur ensure that the database is active or this will fail. It may also be necessary to visit troubleshooting section 1.
3. Congratulations. You are now ready to run CySim and begin using the application. In order to run CySim, navigate to the top of the Visual Studio main project page and click the button shown in figure 1.7 to start CySim. A window in your default browser will open and you will now be able to explore what CySim has to offer. If an error occurs be sure to look at the troubleshooting section.

### **Non First Time Startup:**

Once CySim has been properly set up the process for starting it up again after system shutdown or application closure is simple. This portion of the guide explains how to bring CySim back online for local usage.

This section assumes:

- The user has previously setup CySim as shown in sections above
  - Initial Database Setup
  - Visual Studio CySim Setup and Database Connection

Steps:

1. Begin by opening up the Docker Desktop application.
  - 1.1. Navigate to the CySim container (fig 1.1) and start the container using the start/stop button.
    - 1.1.1. This will bring the server online for the web application.
2. Next, navigate to SQL Server Management application and ensure login via SQL Server Authentication is possible (fig 1.2). If an issue occurs, please see troubleshooting section 1.
3. Once login to the mssql server is confirmed as possible, open the Visual Studio application and open the CySim project.
  - 3.1. Navigate to the top of the main project window (fig 1.7) and click the start button.
    - 3.1.1. After some time CySim should appear in a browser window.

### **Shutting Down of CySim:**

CySim is very simple to shut down and can be done in a few easy steps. This guide explains how to safely and quickly end the application.

This section assumes:

- The user has and active instance of CySim running

Steps:

1. The CySim web application can be shut down two different ways:
  - 1.1. The first method is to simply exit the browser tab that CySim is open. This will automatically stop the application.
  - 1.2. The other option is to navigate to the Visual Studio main project window (fig 1.7) and navigate to the same button used to start CySim. Click here to end the process.
2. Once the web application is shut down, the database will also need to be shut down. First, navigate to the Docker Desktop Application (fig 1.1).
  - 2.1. Click the start/stop button for the CySim container
3. CySim and all of its components will now be shut down. To restart the application view the “Non First Time Start” section above.

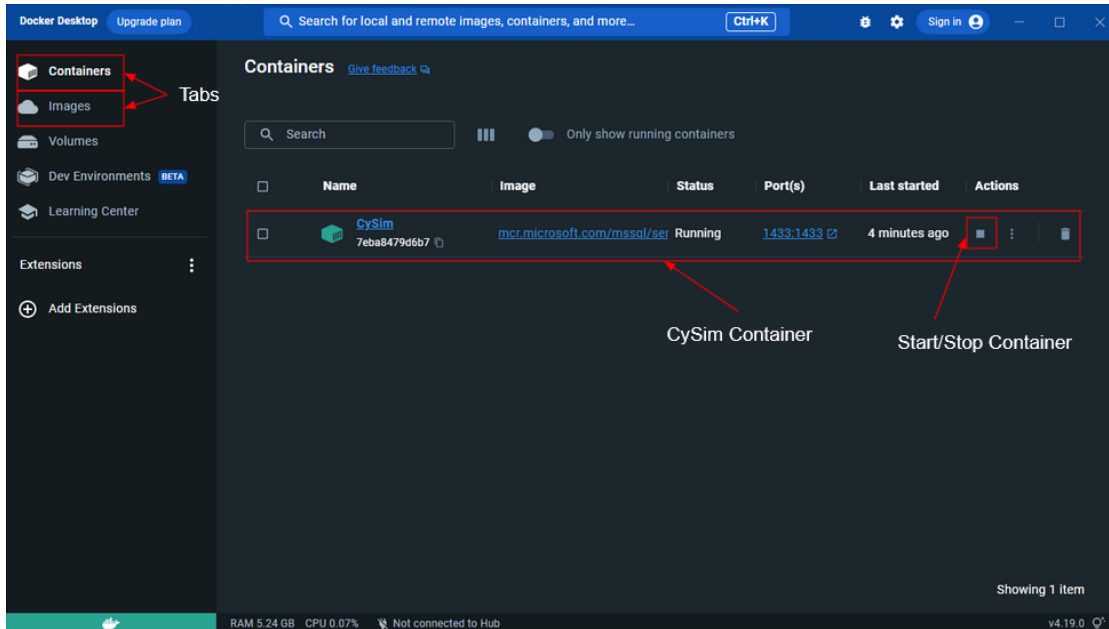
## Troubleshooting:

This section contains a few issues a user may encounter while working with CySim. Please follow the instructions carefully for each error mentioned. If the problem persists after going through these steps it may be necessary to completely delete all elements relating to CySim and restart the installation. These elements include the mssql server container in Docker Desktop containing the CySim database and the CySim project file.

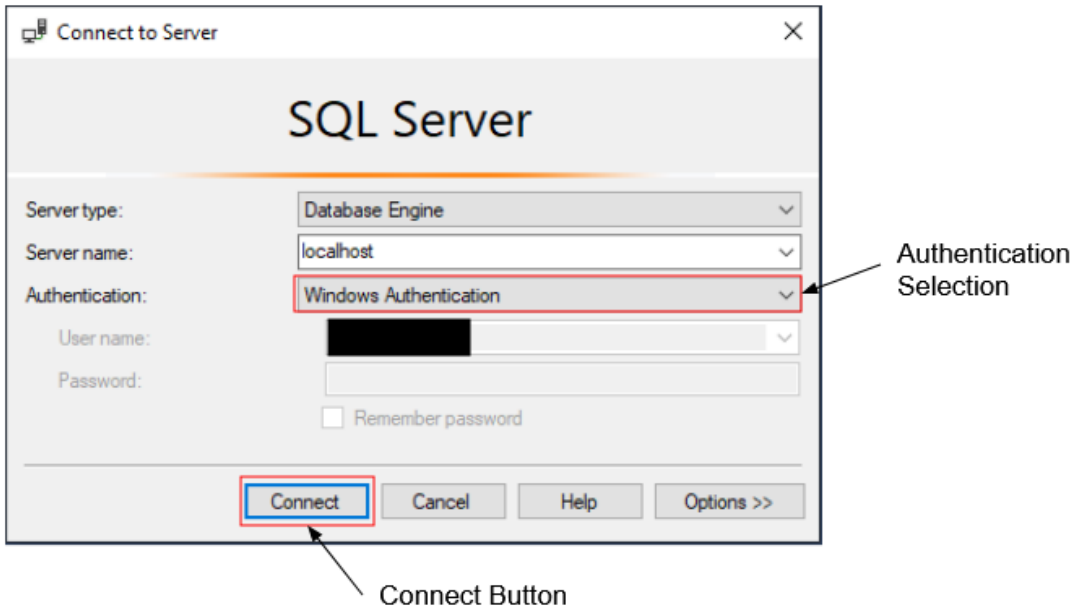
1. Failure to connect using SQL Server Authentication after changing security settings.
  - 1.1. First, ensure that the mssql server is running through the Docker Desktop application.
  - 1.2. Once running, open the Windows “Services” application (figure.1.9).
  - 1.3. Individually select each service relating to SQL and select the stop service button.
    - 1.3.1. Make sure to do this for each SQL related service.
  - 1.4. Once done, navigate to the Docker Desktop application (fig 1.1) and stop the CySim container. After stopping the container allows a bit of time to pass before starting it up again.
  - 1.5. The issues relating to this error should now be fixed. It may be necessary in some cases to do this each time CySim is started locally. Failure to do so will result in a failure of the link between the web application and database.
2. Migrations Related Errors
  - 2.1. Open CySim in Visual Studio and navigate to the “Solution Explorer” (fig 1.7). Find the “Migrations” folder and expand this. Ensure that all existing files are deleted.
  - 2.2. Navigate to the search bar near the top of the Visual Studio project window and search for “Package Manage Console”. Select this option.
    - 2.2.1. A console should appear near the bottom of the Visual Studio Window shown in figure 1.8.

- 2.3. In this console enter the following command to create the migrations necessary for the database and web application: “Add-Migration CySimMigrations”.
- 2.4. Once the migrations have been successfully created enter the following command to update the database: “Update-Database”.
- 2.4.1. If errors occur ensure that the database is active or this will fail. It may also be necessary to visit troubleshooting section 1.

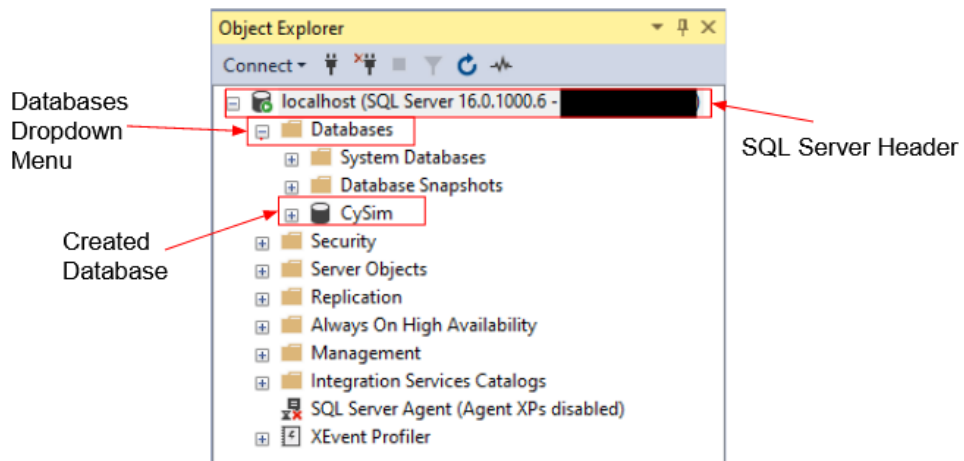
## Figures Relating to CySim Setup and Startup:



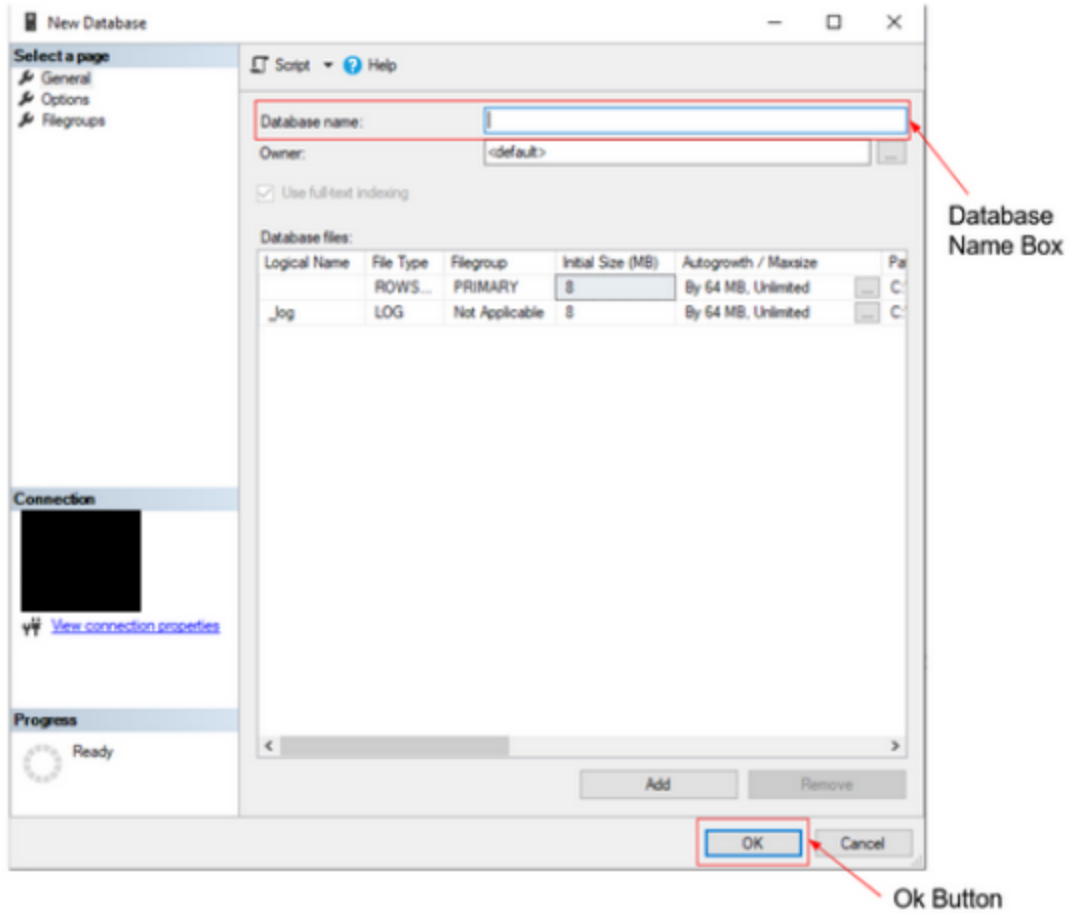
**Fig 1.1 - Docker Desktop**



**Fig 1.2 - SQL Server Management Studio 19 Database Connection Popup**



**Fig 1.3 - SQL Server Management Studio 19 Object Explorer**



**Fig 1.4 - SQL Server Management Studio 19 Database Creator**

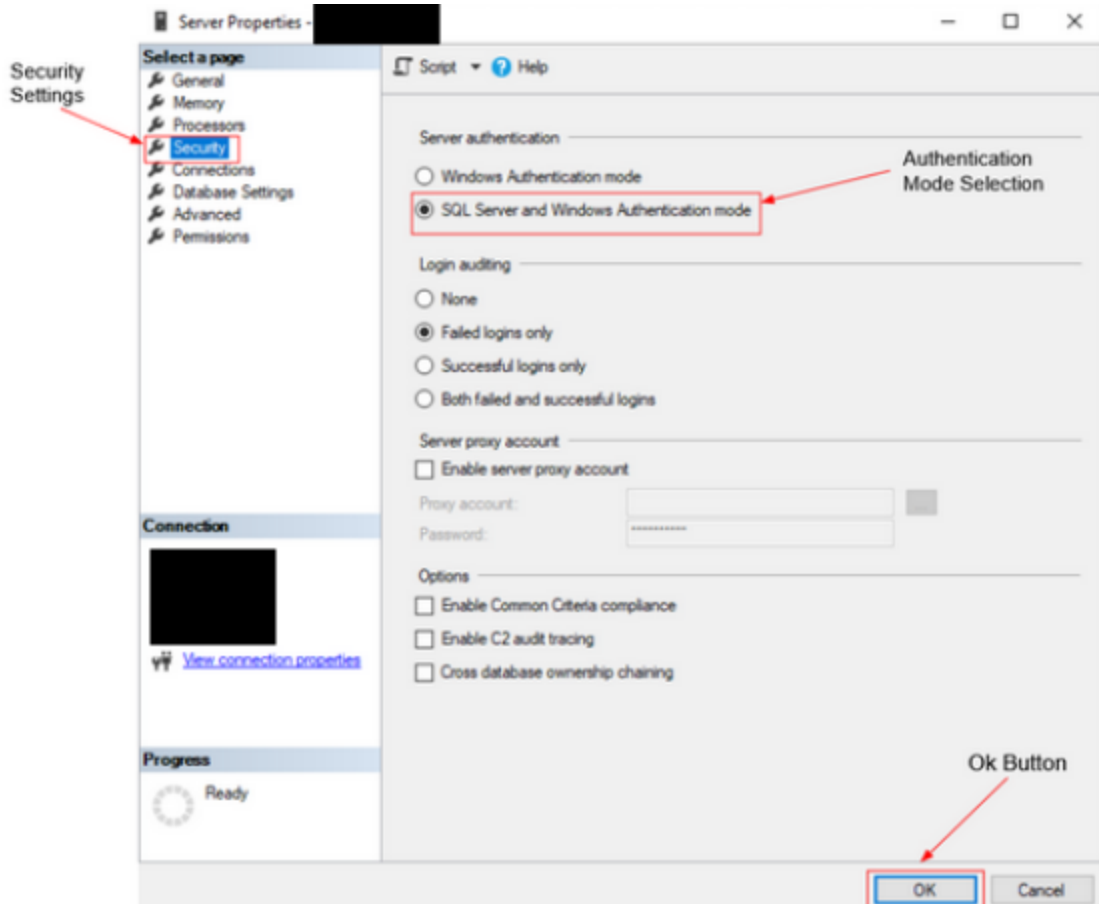


Fig 1.5 - SQL Server Management Studio 19 Database Security Settings

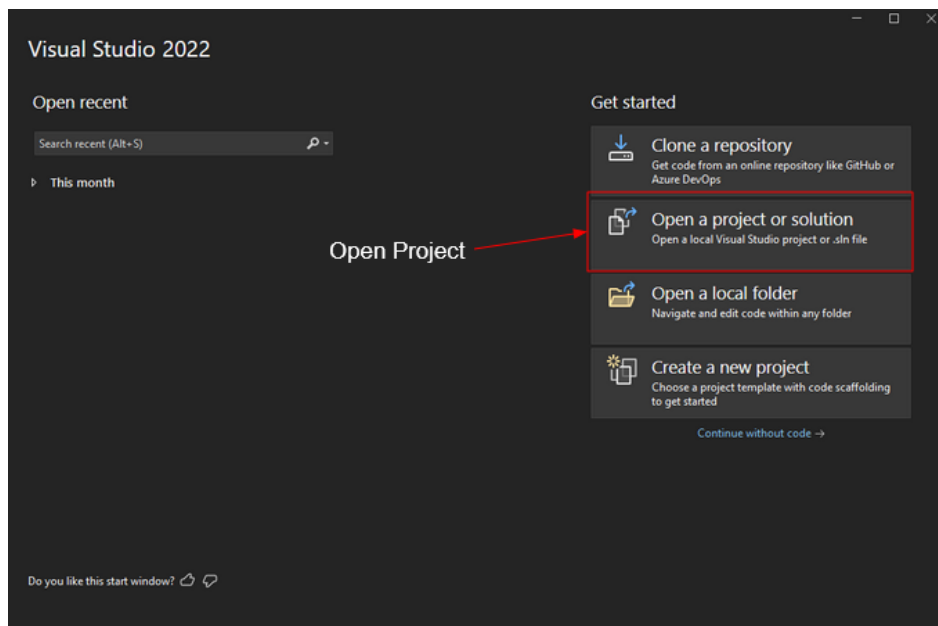


Fig 1.6 - Visual Studio Landing Page



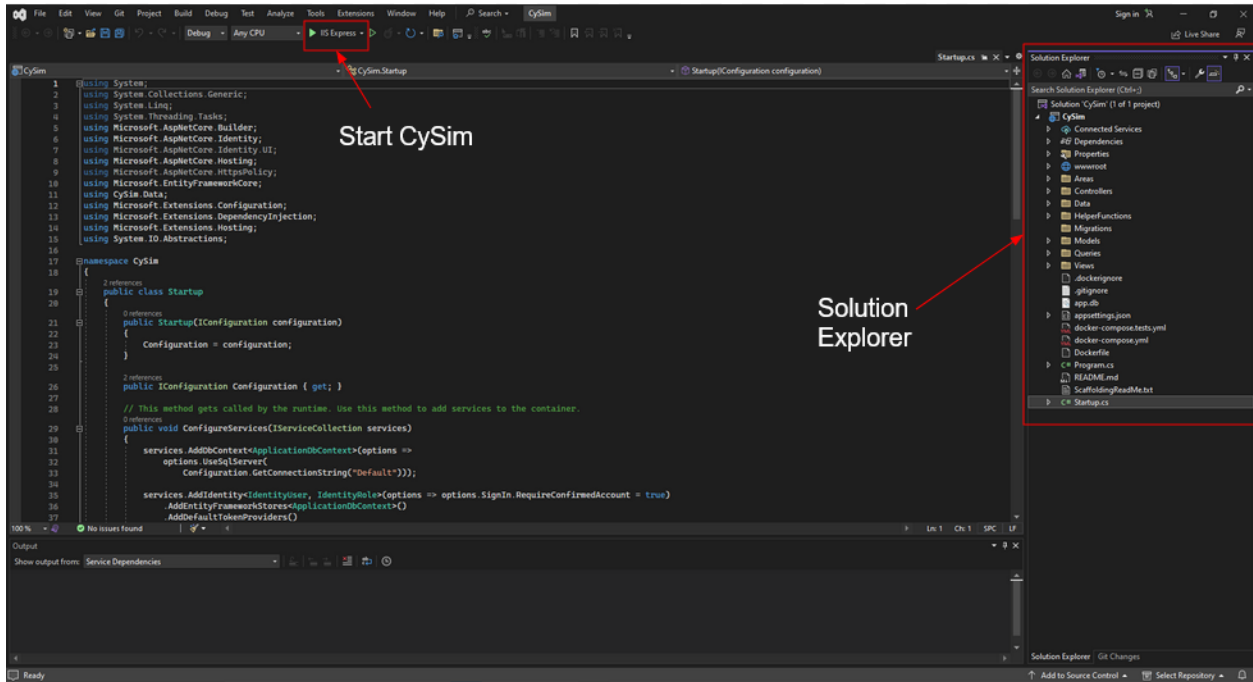


Fig 1.7 - Visual Studio Main Project Window

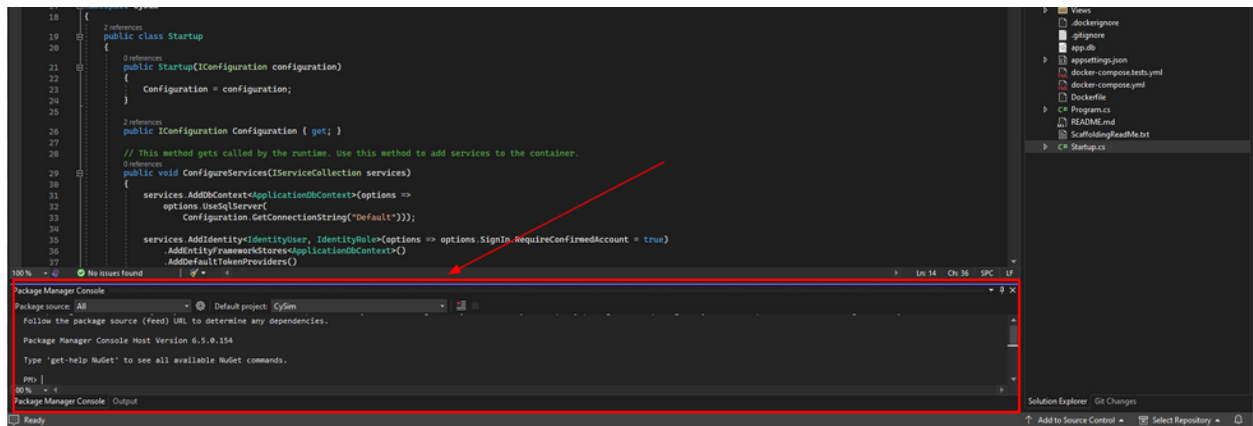
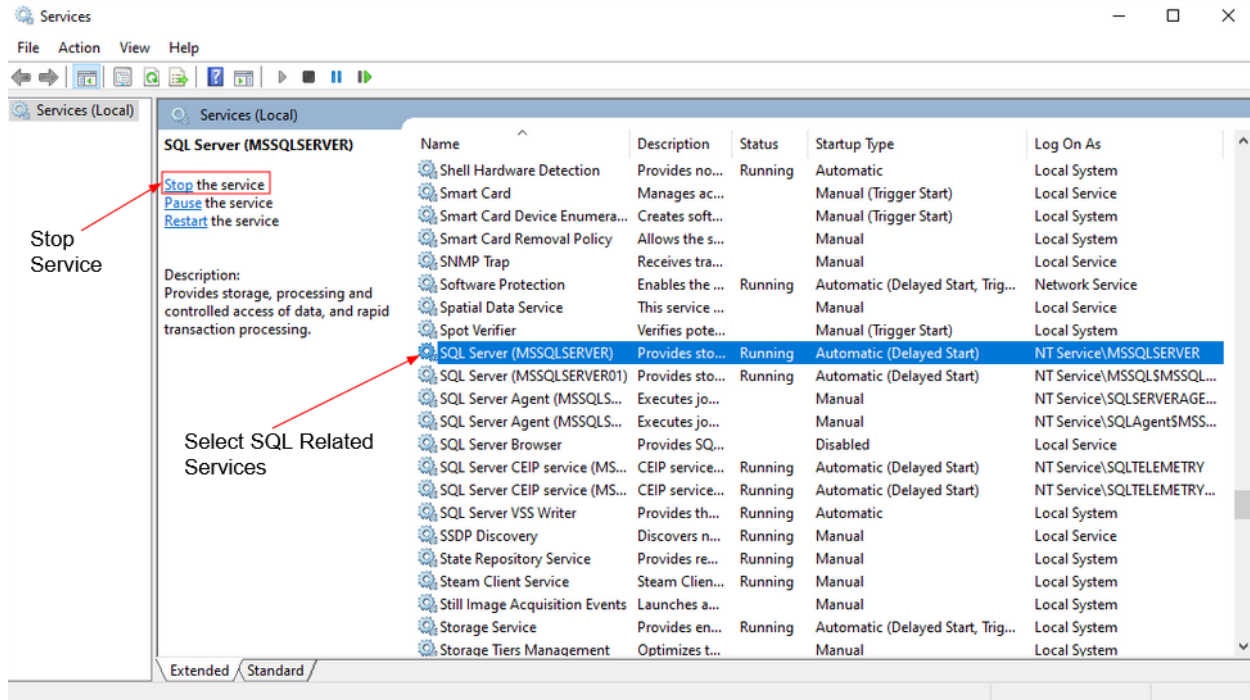


Fig 1.8 - Visual Studio Package Manager Console



**Fig 1.9 - Windows Services Application Window**

## Appendix I - Section 2: How to Use CySim:

CySim has many features and pages to explore. This guide will focus on user actions but also take the time to point out and explain the many different pages the CySim web application has to offer. Please keep in mind that the final edition of CySim may differ slightly when compared to what is contained within this document.

This section assumes:

- The user has properly setup CySim using Appendix I - Section 2
- The user has an active instance of CySim running

Required Knowledge:

- The CySim web application has three different user types with the following definitions:
  - Admin Users: The individuals responsible for managing cyber games.
  - Red Team Users: Participants tasked with completing red team goals such as performing target reconnaissance, infiltrating a system, establishing persistence, and obtaining target resources.

- Blue Team Users: Participants tasked with completing blue team goals such as scanning for suspicious activity, patching vulnerable systems, and responding to instances of red team activities such as system infiltration.
- The CySim web application may also be different depending on login status:
  - Pre-login user refers to a user who is currently not signed into a valid account within the web application.
  - Post-login user refers to a user who is currently signed into a valid account within the web application.
- Keep in mind certain pages may appear differently for different user types and login status. Affiliation is used to refer to Red/Blue team membership.

The following guide contains sections pertaining to:

1. A Short Introduction to the CySim Header
2. The CySim Home Page
3. Accessing the Privacy Policy
4. Login, Logout, and Account Registration
5. Account Settings
6. Team Registration
7. The Scoreboard Page
8. The Tutorials Scenario Page
9. The Machines Dropdown Menu

## **1. A Short Introduction to the CySim Header:**

- 1.1. The CySim headers main purpose is for easy navigation throughout the web application. It can be found at the top of all pages and is the same for all user types. However, it does differ depending on login status. Section 1.2 covers the pre-login header while section 1.3 covers the post-login header.
- 1.2. The pre-login user CySim header can be seen in figure 1.1 with numbers relating to each element contained within it. This section will go over each of these elements 1 to 3.
  - 1.2.1. Element 1: The CySim Header logo. At any point within the web application the user is able to click this in order to return to the CySim Home Page addressed in section 2 of this guide.
  - 1.2.2. Element 2: The Register button. Clicking here will bring the user to a CySim account registration page addressed in section 4 of this guide.
  - 1.2.3. Element 3: The Login button. Clicking here will bring the user to a CySim login page addressed in section 4 of this guide.
- 1.3. The post-login user CySim header can be seen in figure 1.2 with numbers relating to each element contained within it. This section will go over each of these elements 1 to 8.
  - 1.3.1. Element 1: The CySim Header logo. Please view section 1.2.1 for more information.

- 1.3.2. Element 2: The Team Registration page. Clicking here will bring the user to the team registration page addressed in section 6 of this guide.
- 1.3.3. Element 3: The Scoreboard page. Clicking here will bring the user to the scoreboard page addressed in section 7 of this guide.
- 1.3.4. Element 4: The tutorial page. Clicking here will bring the user to the tutorial page addressed in section 8 of this guide.
- 1.3.5. Element 5: The Scenarios page. Clicking here will bring the user to the scenarios page addressed in section 8 of this guide.
- 1.3.6. Element 6: The Machine Dropdown Menu. Clicking here will reveal a dropdown menu addressed in section 9 of this guide.
- 1.3.7. Element 7: The Account Settings page. Clicking here will bring the user to the account settings page addressed in section 5 of this guide.
- 1.3.8. Element 8: The Logout button. Clicking here will log the user out of the CySim web application. This is addressed more in depth in section 4 of this guide.

## **2. The CySim Home Page:**

- 2.1. The CySim home page is a small landing page for users to view some preliminary information about CySim along with an informational video. It can be accessed from any part of the application by clicking the logo located in the header addressed in section 1.2.1 and 1.3.1. Figure 1.3 shows a labeled homepage for a pre-login user. For more information regarding the header view section 1. For the footer view section 3.

## **3. Accessing the Privacy Policy:**

- 3.1. At this current time the CySim privacy policy is incomplete and more of a template for a future update. It is recommended this be updated with legal consultation. This privacy policy can be accessed from any point in the web application by navigating to the footer of the page the user is currently viewing. This footer is labeled in figure 1.3 and will appear there not only for the home page but all other pages. Clicking will bring the user to the privacy policy page shown in figure 1.4.

## **4. Login, Logout, and Account Registration:**

- 4.1. This section will cover the processes of logging into, logging out of, and creating an account within the CySim web application. Be aware that these actions require database connection and will fail if this is not the case.
- 4.2. Logging into Cysim:
  - 4.2.1. This section requires the user to have previously made an account as detailed in section 4.4. Please ensure this has been completed before proceeding with the following steps.
  - 4.2.2. In order to login to CySim, navigate and click the button described in section 1.2.3.
  - 4.2.3. Once the login button is clicked, the user will be redirected to a login page shown in figure 1.5.

- 4.2.4. To login simply enter a valid username, password and click the login button. All elements and options present in section 4.2.4 are labeled in figure 1.5. Other options are shown below:
  - 4.2.4.1. “Remember Me” check box but running the application locally means this has no effect.
  - 4.2.4.2. “Forgot Your Password?” for account recovery via email but this will not work due to lack of an email server.
  - 4.2.4.3. “Register as a new user” redirects the user to the account registration page covered in more detail in section 4.4.
- 4.3. Logging out of CySim:
  - 4.3.1. This section requires the user to have previously made an account as detailed in section 4.4 and be logged into the application as described in section 4.2.
  - 4.3.2. To log out of a user account simply navigate and click the button described in section 1.3.8. This will log the user out of CySim and send them back to the pre-login CySim home page described in section 2.
- 4.4. Creating a CySim Account:
  - 4.4.1. In order to begin the process of creating a CySim user account, navigate and click the button described in section 1.2.2.
  - 4.4.2. Once this button has been clicked the user will be redirected to an account registration page shown in figure 1.6.
  - 4.4.3. To register an account simply enter a username, email, password, and select the team to register for from the “Role Selection” dropdown menu. To finish creation simply click the “Register” button.
    - 4.4.3.1. Note: The email address does not have to be an actual address. Read section 4.4.4 for more information.
  - 4.4.4. After clicking the button you will be required to confirm your email address. However as of this time CySim does not have email functionality. The user will be redirected to an email confirmation page. Follow the instructions here to confirm the account.
  - 4.4.5. Once finished the user will be redirected to another page. To login to the newly created account follow the instructions laid out in section 4.2.

## **5. Account Settings:**

- 5.1. This section will cover many options found in the account settings pages. In order to access these settings, first navigate and click the button described in section 1.3.7.
- 5.2. Once clicked, the user will be redirected to the account settings profile page. This page and all other pages mentioned can be found under figure 1.7. A description of each of these different pages is also provided below. In order to navigate through these pages simply click on the account settings navigational links.
- 5.3. A Description of each page and their functions are provided below. Images of each page can be found in the order of this list in figure 1.7. The order follows that laid out by the account settings navigational links.

- 5.3.1. The Profile Page: Here a user is able to add a phone number to their account. At this time there is no way to confirm these numbers. To save simply click the “Save” button.
- 5.3.2. The Email Page: Here a user is able to change the email associated with their account. Enter a new email into the “New email” text box and click the “Change email” button. This will follow the procedures for confirmation layed out in section 4.4.4.
- 5.3.3. The Password Page: Here a user is able to change their password by first entering their current password and then their new password twice. To finish, click the “Update password” button.
- 5.3.4. The Two-Factor Authentication Page: Due to the web application being ran locally, two factor authentication is disabled. When fully implemented a user will be able to follow the instructions to add an authenticator application.
- 5.3.5. The Personal Data Page: Here a user is able to either delete their account by clicking the “Delete” button or download a json file containing all information CySim collects on the user. This json can be downloaded by clicking the “Download” button.

## **6. Team Registration:**

- 6.1. This section will cover all information related to the creation of teams, editing of team information, deletion of teams, and some information about how the teams are used within CySim. This page will appear different depending on if the user is an Admin user or part of the Red/Blue team. Section 6.2 will cover this page from the perspective of an Admin user while section 6.3 will cover it from the perspective of a Red or Blue team user.
- 6.2. The Admin user is able to see and edit both Blue affiliated teams and Red affiliated teams. They can also add and edit all teams. This view is shown and labeled in figure 1.8. This section will cover the fundamentals of the Team Registration page.
  - 6.2.1. General Page Description: The page contains some general information pertaining to the teams including how many can join a team at once, and who is able to create, edit or delete teams. The actual teams are listed under their respective color as shown in figure 1.8. The order of information about each team is as follows:
    - 6.2.1.1. Profile Picture: Teams can upload a profile image to be displayed both here and in the scoreboard page (section 7).
    - 6.2.1.2. Team Name: Teams can choose the name of their choice to be displayed both here and in the scoreboard page (section 7).
    - 6.2.1.3. Available Spots; Teams can have up to 6 members. This displays how many spots are still available.
    - 6.2.1.4. Users: This contains the email of each user that is apart of a team.
    - 6.2.1.5. There is also a section titled controls where a team may be deleted, edited, or joined..
  - 6.2.2. Creating a Team: In order to create a team, navigate to and click the “Add a New Team” button.

- 6.2.2.1. This will redirect the user to a team creation page shown in figure 1.9. Here a user creates a team by selecting an image to act as the team image and enter a team name.
  - 6.2.2.1.1. If no image is selected a default one will be put in place.
- 6.2.2.2. Admin users also have a checkbox to indicate whether this team is a Red team. If creating a Blue team leave this unchecked. If creating a Red team make sure to mark this.
- 6.2.2.3. Navigate and click the create button to complete the team creation process.
- 6.2.3. Deleting a Team: To delete a team, navigate to the controls section indicated by section 6.2.1.5 and click the red “Delete” button. This will prompt the user for confirmation and once given will remove the team.
- 6.2.4. Editing a Team: To edit a team, navigate to the controls section indicated by section 6.2.1.5 and click the blue “Edit” button. This will bring the user to an edit page shown and labeled in figure 1.10. Here a new image can be uploaded to replace the previous team image, a new team name can be entered, the team type can be changed, and members can be added or removed. The team type follows the same rules as those laid out in section 6.2.2.2. To save these changes, simply click the “Update” button.
- 6.2.5. Joining a Team: Admin users are unable to join teams. For more information on how to join one as a Red or Blue team user please view section 6.3.5.
- 6.3. Red and Blue Teams have a very similar view to that of the Admin user. Much of this information will refer back to section 6.2. The view is very similar to figure 1.8.
  - 6.3.1. General Page Description: A user will only be able to see the teams pertaining to their team color being either Red or Blue unlike the Admin who can see both. Otherwise the information is the same as covered in section 6.2.1.
  - 6.3.2. Creating a Team: Same process as that described in section 6.2.2 except that Red and Blue users will not be able to choose between creating a Blue or Red team. The team will be of the type for which they are affiliated with.
  - 6.3.3. Deleting a Team: This is the same process as 6.2.3 except a user can only delete teams that they have created.
  - 6.3.4. Editing a Team: This is the same process laid out in section 6.2.4 except that users will not be able to edit team affiliation. It will remain as part of the type of which they are affiliated with.
  - 6.3.5. Joining a Team: Simply navigate to and click the “Join” button within the controls section described in 6.2.1.5.

## **7. The Scoreboard Page:**

- 7.1. This section will cover all information related to the scoreboard. This includes the content of the scoreboard, and the ability for an Admin user to edit scores. Similar to the Team Registration page covered in section 6, this page will appear slightly different for Admins and Red/Blue team users. Section 7.3 will cover the page from an Admin users perspective while section 7.4 will cover the differences for a Red/Blue team user.

- 7.2. The scoreboard will auto scale and generate as many entries as needed depending on the amount of teams registered. No manual team addition is required to the scoreboard.
- 7.3. The Admin user is able to see both the scores of Red and Blue affiliated teams. This page is shown and labeled in figure 1.11. This section will cover the information displayed and how Admins can edit team scores.
  - 7.3.1. General Page Description: The page contains some general information pertaining to teams scores and rankings. Ranking is determined by a team's current score.
    - 7.3.1.1. Placement: Displays ranking of teams against other teams.
    - 7.3.1.2. Profile Picture: Displays the profile picture selected during team creation. If none was selected, it shows the default image.
    - 7.3.1.3. Team Name: Displays team name chosen during team creation.
    - 7.3.1.4. Score: Displays current team score.
    - 7.3.1.5. Flags: Displays current team flags
    - 7.3.1.6. Usability: Displays current team usability score
  - 7.3.2. Editing Team Scores: Admins are the only user group with the ability to see the edit button and change team scores. To begin, navigate to and click the blue “edit” button under the controls section shown in figure 1.11.
    - 7.3.2.1. This will redirect the user to a page like that shown in figure 1.12. Here an Admin can change various scoring metrics for a team. Click the “update” button to reflect this change in the scoreboard.
- 7.4. The Red/Blue teams are able to only see the teams which are affiliated with them. This and the lack of an edit option are the only differences between the two user views. For information on the different information displayed for each team see section 7.3.1.

## **8. Tutorials and Scenario Page:**

- 8.1. This section will cover the functionality of both the tutorials and scenario pages. They are quite similar in implementation and therefore will both be covered in this section. Once again, both of these pages are different depending on if the user is an Admin or a member of Red/Blue team. Section 8.2 covers the tutorials page while section 8.3 covers the scenarios page.
- 8.2. The tutorial page is where Admins are able to upload tutorial documents for use during a CySim event. This page has two different views depending on if the user is an Admin or member of the Red/Blue team. An example of the tutorial page is shown in figure 1.13 from the Admin perspective. There are two different document sections for each team. These are tutorials and gametypes. Section 8.2.1 covers this page from the Admin perspective while section 8.2.2 covers this page from a Red/Blue team perspective.
  - 8.2.1. Admins are able to view all the tutorials and gametypes that are uploaded for either team. This section will cover both what is included with each entry as well as how to add, delete, and edit the tutorials or gametypes.
    - 8.2.1.1. What is Visible: All users are able to view the filename and description of the file. Admins also have a separate category which contains the options for deleting and editing an entry.
    - 8.2.1.2. How to Add a Tutorial or Gametype: In order to add a tutorial navigate and click the “Add a Tutorial” button.



- 8.2.1.2.1. This will redirect the user to another page shown in figure 1.14. Here the user must upload a document. The user can also add a description, indicate this is for red team, and indicate this is a gametype. Once complete navigate and click the “Create” button.
      - 8.2.1.3. How to Delete a Tutorial or Gametype: In order to delete a tutorial, navigate and click the “Delete” button mentioned in section 8.2.1.1. This will prompt the user for confirmation which when accepted will remove the tutorial or gametype.
      - 8.2.1.4. How to Edit a Tutorial or Gametype: In order to edit a tutorial navigate to and click the “Edit” button mentioned in section 8.2.1.1.
        - 8.2.1.4.1. This will redirect the user to another page shown in figure 1.15. Here the user can upload a replacement document, add another name for the file, edit the description, change team affiliation, and change this to a gametype or tutorial. Once complete navigate and click the “Update” button.
    - 8.2.2. Red and Blue team users only differ from the Admins in one way. These users are not allowed to add, delete, or edit any of the documents added. They’re also only allowed to view the documents related to their specific affiliation.
  - 8.3. The scenarios page is where Admins are able to upload scenario documents for use during a CySim event. This page has two different views depending on if the user is an Admin or member of the Red/Blue team. An example of the scenarios page is shown in figure 1.16 from the Admin perspective. Section 8.3.1 covers this from an admin perspective while 8.3.2 covers it from a Red/Blue team perspective.
    - 8.3.1. Admins are able to view scenarios that are uploaded for either team. This section will cover both what is included with each entry as well as how add, delete, and edit the tutorials or gametypes.
      - 8.3.1.1. What is Visible: All users are able to view the filename and description of the file. Admins also have a separate category which contains the options for deleting and editing an entry.
      - 8.3.1.2. How to Add a Scenario: In order to add a scenario navigate and click the “Add a Scenario Document” button.
        - 8.3.1.2.1. This will redirect the user to another page shown in figure 1.17. Here the user must upload a document. The user can also add a description and indicate this is for the red team. Once complete navigate and click the “Create” button.
        - 8.3.1.3. How to Delete a Scenario: In order to delete a scenario, navigate and click the “Delete” button mentioned in section 8.3.1.1. This will prompt the user for confirmation which when accepted will remove the scenario.
        - 8.3.1.4. How to Edit a Scenario: In order to edit a scenario navigate to and click the “Edit” button mentioned in section 8.3.1.1.
          - 8.3.1.4.1. This will redirect the user to another page shown in figure 1.18. Here the user can upload a replacement document, add another name for the file, edit the description, and change team

affiliation. Once complete navigate and click the “Update” button.

- 8.3.2. Red and Blue team members differ only from the Admins in one way. These users are not allowed to add, delete, or edit any of the documents added. They’re also only allowed to view the documents related to their specific affiliation.

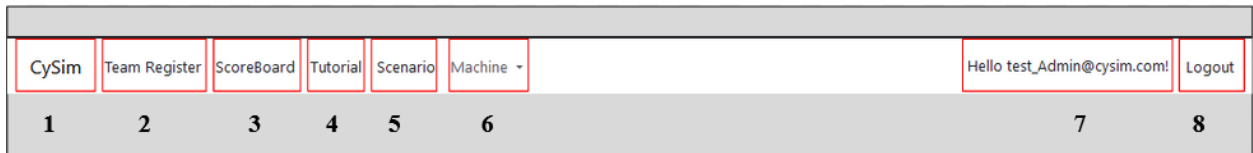
## **9. Machines Dropdown Menu:**

- 9.1. This section will briefly cover each of the options found within the machine's dropdown menu. The menu is meant to pertain to the machines issued to teams and players for use in cyber games. This menu is shown in figure 1.19. As of now they are incomplete due to time constraints. However, the use for each of these pages appear in the order in which they appear in the menu:
  - 9.1.1. Machines: This page will act as the access point for a user to gain access to their allotted virtual machines.
  - 9.1.2. Machine Info: This page will display a basic explanation and manual of how to access the users allotted machines as well as how to use them.
  - 9.1.3. Machine Status: This page will display the status of the machines currently deployed in the CySim environment.

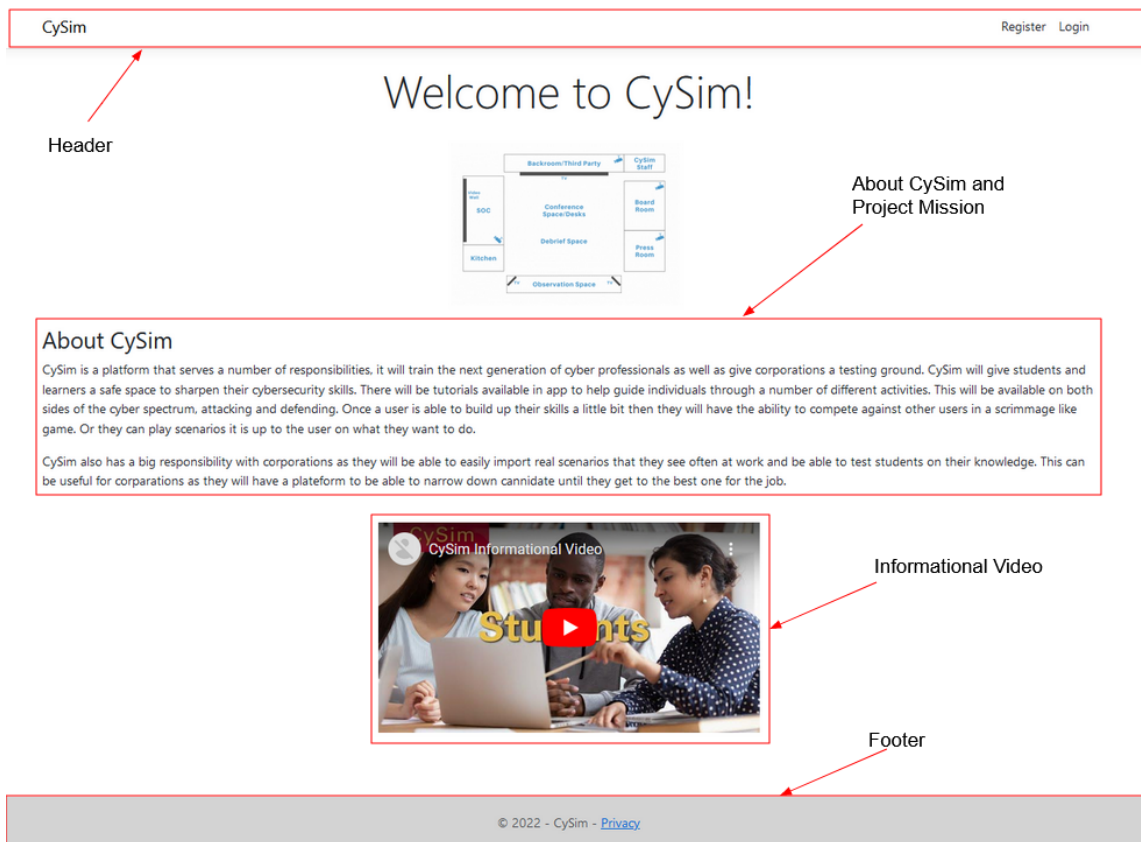
## Figures Relating to CySim Usage:



**Fig 1.1 - CySim Web Application Header Pre-Login**



**Fig 1.2 - CySim Web Application Header Post-Login**



**Fig 1.3 - CySim Labeled Homepage for Pre-Login User**

## Privacy Policy of CySim

CySim operates the [www.CySim.iastate.edu](http://www.CySim.iastate.edu), which provides the SERVICE.

This page is used to inform website visitors regarding our policies with the collection, use, and disclosure of Personal Information if anyone decided to use our Service, the [www.CySim.iastate.edu](http://www.CySim.iastate.edu) website.

If you choose to use our Service, then you agree to the collection and use of information in relation with this policy. The Personal Information that we collect are used for providing and improving the Service. We will not share your information with anyone except as described in this Privacy Policy.

The terms used in this Privacy Policy have the same meanings as in our Terms and Conditions, which is accessible at the [www.CySim.iastate.edu](http://www.CySim.iastate.edu) site, unless otherwise defined in this Privacy Policy.

### Information Collection and Use

For a better experience while using our Service, we may require you to provide us with certain personally identifiable information, including but not limited to your name and email. The information that we collect will be used to contact of identify you.

### Log Data

We want to inform you that whenever you visit our service, we collect information that your browser sends to us that is called Log Data. This Log Data may include information such as your computer Internet Protocol ("IP") address, browser version, pages of our Service that you will visit, the time and date of your visit, the time spent on those pages, and other statistics.

### Cookies

Cookies are files with small amount of data that is commonly used an anonymous unique identifier. These are sent to you browser from the website that you visit and are stored on your computer's hard drive.

Our website uses these "cookies" to collection information and to improve our Service. You have the option to either accept or refuse these cookies, and know when a cookie is being sent to your computer. If you choose to refuse these cookies, you may not be able to use some portions of our Service.

**Fig 1.4 - CySim Privacy Policy Page**

## Log in

Use a local account to log in.

The screenshot shows the login form with the following elements and annotations:

- Email Entry:** Points to the email input field.
- Password Entry:** Points to the password input field.
- Remember me Checkbox:** Points to the checkbox labeled "Remember me?".
- "Log in" Button:** Points to the blue "Log in" button.
- Forgot Password Link:** Points to the link "Forgot your password?".
- Register New Account Link:** Points to the link "Register as a new user".

**Fig 1.5 - CySim Login Page**

# Register

Create a new account.

The registration form contains the following elements:

- Username Entry Box:** A text input field for the user's name.
- Email Entry Box:** A text input field for the user's email address.
- Password and Password Confirmation Boxes:** Two text input fields for creating and confirming a password.
- Role Selection Dropdown Menu:** A dropdown menu with the text "--Select Role--".
- "Register" Button:** A blue button with the text "Register".

**Fig 1.6 - CySim Account Registration Page**

## Manage your account

Change your account settings [Account Settings Navigational Links](#)

The account settings page features a sidebar with the following options:

- Profile:** Selected and highlighted with a blue bar.
- Email**
- Password**
- Two-factor authentication**
- Personal data**

The main content area for the Profile section includes:

- Username:** test\_Admin@cysim.com
- Phone number:** (empty input field)
- Save:** A blue button.

## Manage your account

Change your account settings

The account settings page features a sidebar with the following options:

- Profile**
- Email:** Selected and highlighted with a blue bar.
- Password**
- Two-factor authentication**
- Personal data**

The main content area for the Manage Email section includes:

- Email:** test\_Admin@cysim.com (with a checkmark icon)
- New email:** test\_Admin@cysim.com (input field)
- Change email:** A blue button.

## Manage your account

Change your account settings

Profile

Email

Password

Two-factor authentication

Personal data

### Change password

Current password

New password

Confirm new password

Update password

## Manage your account

Change your account settings

Profile

Email

Password

Two-factor authentication

Personal data

### Two-factor authentication (2FA)

Authenticator app

Setup authenticator app

Reset authenticator app

## Manage your account

Change your account settings

Profile

Email

Password

Two-factor authentication

Personal data

### Personal Data

Your account contains personal data that you have given us. This page allows you to download or delete that data.

**Deleting this data will permanently remove your account, and this cannot be recovered.**

Download

Delete

**Fig 1.7 - CySim Account Settings Pages**

### Some things to know before you register for a team



- If you do not see a team that you wish to join then you will be able to create your own
  - The creator of a team is also able to edit and delete the team that they create
- If you see a team that you wish to join then you will be able to click join on that team
- Each team has the ability to have up to 6 players so choose wisely

General Team Registration Page Information



Add Team Button

Registered Red Teams

Add a New Team

Profile Picture	Team Name	Available Spots	Users	Controls
	Vipers	6		<a href="#">Edit</a> <a href="#">Delete</a>
	Vultures	6		<a href="#">Edit</a> <a href="#">Delete</a>

### Blue Teams

Profile Picture	Team Name	Available Spots	Users	Controls
	The Goats	6		<a href="#">Edit</a> <a href="#">Delete</a>
	Eagles	6		<a href="#">Edit</a> <a href="#">Delete</a>

Registered Blue Teams

Fig 1.8 - CySim Team Registration Page Admin View

CySim Team Register ScoreBoard Tutorial Scenario Machine ▾ Hello test\_Admin@cysim.com! Logout

### Add a New Team

Select a Profile Picture to Upload Upload a file to use as a profile image

Browse... No file selected. Team Name Entry Box

Team Name

Red Team?  Red Team Checkbox

[Create](#) [Back](#) Create Team and Back Button

Fig 1.9 - CySim Team Registration Page Admin View

### Edit a Team for Test Team

Upload a file to use as a profile image

Select file to replace profile picture (Optional):  
 Browse... No file selected.

Rename Team Entry Box

Team Name: Test Team

User1  
User2  
User3  
User4  
User5  
User6

Red Team?

Update and Back Button



Red Team Checkbox

User List to Add/Remove Users



**Fig 1.10 - CySim Team Registration Page Admin View**

### Red Affiliated Teams Scoreboard

#### Red Team Scoreboard

Placement	Profile Picture	Team Name	Score	Flags	Usability	Edit
1		Vulture	4	4	4	<a href="#">Edit</a>
2		Vipers	2	2	2	<a href="#">Edit</a>

#### Blue Team Scoreboard

Placement	Profile Picture	Team Name	Score	Flags	Usability	Edit
1		Eagles	5	5	5	<a href="#">Edit</a>
2		The Goats	1	1	1	<a href="#">Edit</a>

### Blue Affiliated Teams Scoreboard

**Fig 1.11 - CySim Scoreboard Page Admin View**



Edit Team Overall Score
Edit Team Usability Score

**Edit the Score for Vulture**

Score:  Usability:

Flags:

Update
Back

**Fig 1.12 - CySim Scoreboard Admin Edit Score Page**

## Welcome to the Tutorials Page

Some general background on the Tutorial Page:

- Below you will see the respective tutorials for your team. If you are red you will have access to red tutorials, if blue you will have access to blue tutorials.
  - There is four total tables for tutorials. Two red and two blue, the two red are for tutorials as well as gametype, and blue is for tutorials and gametype too.
  - Admin is able to create/edit and delete tutorials as well as assign them to respective teams.
- You will be able to click the hyperlinked lines for each tutorial, this will allow you to download the pages and view them to better understand CySim.

Admin Add Tutorial Button

Access Tutorial Documents

[Add a Tutorial](#)

General information on the Tutorial Page

Red Team Tutorials and Game Types

### Red Team Tutorials

File Name	Description	Actions
<a href="#">NMAP.pdf</a>	Tutorial on Nmap	<span>Edit</span> <span>Delete</span>
<a href="#">Meta-Tutorial.docx</a>	Metasploit Tutorial	<span>Edit</span> <span>Delete</span>

### Red Team GameTypes

File Name	Description	Actions
<a href="#">RedTeamGameType1.pdf</a>	Red Team GameType 1	<span>Edit</span> <span>Delete</span>
<a href="#">RedTeamGameType2.pdf</a>	Red Team GameType 2	<span>Edit</span> <span>Delete</span>

### Blue Team Tutorials

File Name	Description	Actions
<a href="#">LDAP.docx</a>	Tutorial on Setting up LDAP	<span>Edit</span> <span>Delete</span>

### Blue Team GameTypes

File Name	Description	Actions
<a href="#">BlueTeamGameType1.pdf</a>	Blue Team GameType 1	<span>Edit</span> <span>Delete</span>
<a href="#">BlueTeamGameType2.pdf</a>	Blue Team GameType 2	<span>Edit</span> <span>Delete</span>
<a href="#">BlueTeamGameType3.pdf</a>	Blue Team GameType 3	<span>Edit</span> <span>Delete</span>

Blue Team Tutorials and Game Types

**Fig 1.13 - CySim Tutorial Page Admin View**

## Add a New Tutorial

Select a Tutorial to Upload  
Browse... No file selected. Upload Tutorial or Game Type Document

Description: Tutorial/Game Type Description Entry Box

Red Team? Mark as Red Team Tutorial  
 Game Type? Mark as Game Type Document

Create Back Create/Back Buttons

**Fig 1.14 - CySim Add Tutorial Page Admin View**

CySim Team Register ScoreBoard Tutorial Scenario Machine - Hello test\_Admin@cysim.com! Logout

### Edit the LDAP.docx Tutorial Document Below

Tutorial document to replace LDAP.docx contents (Optional):  
Browse... No file selected. Upload Replacement Tutorial or Game Type Document

Provide the file name for the document:  
LDAP.docx Edit Title Entry Box

Provide the description for the document:  
Tutorial on Setting up LDAP Edit Document Description

Is this a red team document? Mark as Red Team Tutorial  
 Is this a Game Type? Mark as Game Type Document

Update Back Update/Back Buttons

**Fig 1.15 - CySim Edit Tutorial Page Admin View**

CySim Team Register ScoreBoard Tutorial Scenario Machine - Hello test\_Admin@cysim.com! Logout

### Access CySim's Scenario Documents

[Add a Scenario Document](#) Add Scenario Button Red Team Scenarios

Red Team Scenario Documents		
FileName	Description	Actions
<a href="#">RedScenario1.docx</a>	Red Team Scenario 1	<a href="#">Edit</a> <a href="#">Delete</a>
<a href="#">RedScenario2.pdf</a>	Red Team Scenario 2	<a href="#">Edit</a> <a href="#">Delete</a>

Blue Team Scenario Documents		
FileName	Description	Actions
<a href="#">BlueScenario1.docx</a>	Blue Team Scenario 1	<a href="#">Edit</a> <a href="#">Delete</a>
<a href="#">BlueScenario2.pdf</a>	Blue Team Scenario 2	<a href="#">Edit</a> <a href="#">Delete</a>

Blue Team Scenarios

**Fig 1.16 - CySim Scenarios Page Admin View**

### Add a New Scenario Document

Select Scenario Document to Upload  
Browse... No file selected.

Provide the description for the document:

Red Team?

Create Back

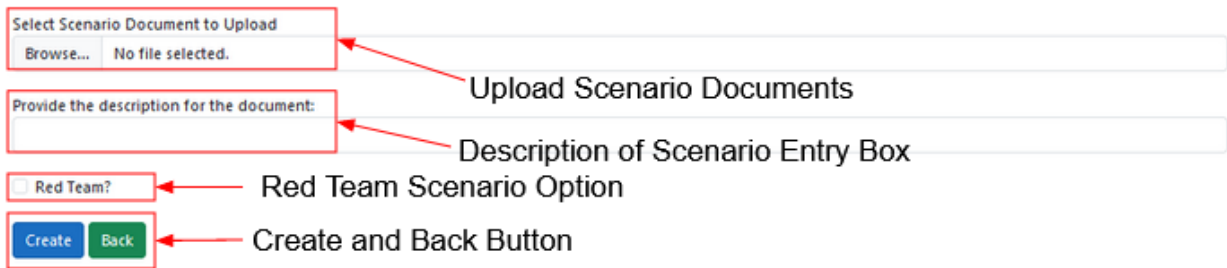


Fig 1.17 - CySim Add Scenarios Page Admin View

### Edit the TestScenario.pdf Scenario Document Below

Scenario document to replace TestScenario.pdf contents (Optional)  
Browse... No file selected.

Provide the file name for the document:  
TestScenario.pdf

Provide the description for the document:  
Test Scenario for CySim

Is this a red team document?

Update Back

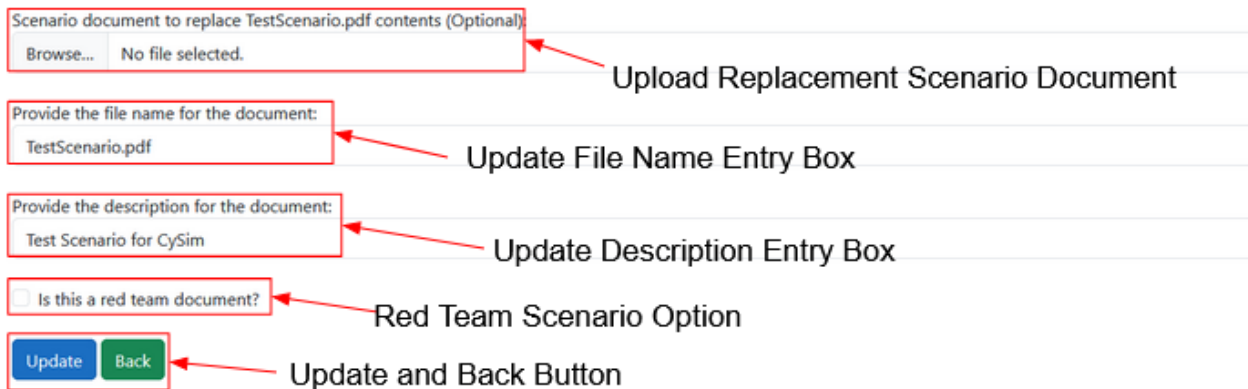


Fig 1.18 - CySim Edit Scenarios Page Admin View

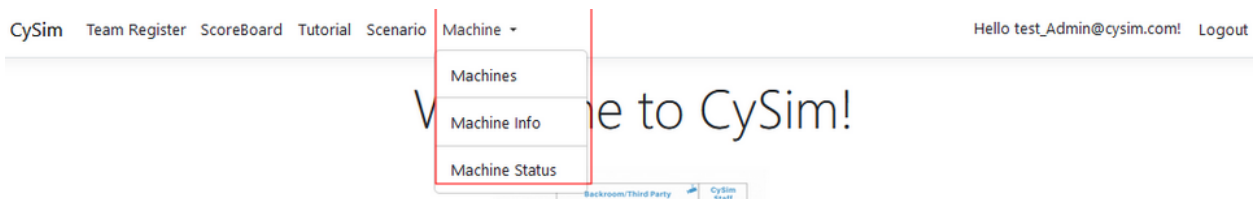


Fig 1.19 - CySim Machine Drop Down Menu

## Appendix II - Other Considerations:

Below are some of the other considerations that have occurred throughout the process of creating and developing the CySim web application.

- The OS Hurdle
  - When we started to form a game plan for the development of CySim we ran into an interesting problem.
  - We had team members utilizing 3 different operating systems. Apple, Windows and Linux. This made for a rather challenging start to the project because we had to immediately find a way to accommodate all of the different OS.
  - **SOLVED:** Through a lot of testing and different research, we were able to figure out a way for each OS to be able to load and work on the CySim web application.
- Server Situation
  - Another thing that came up throughout the first and a portion of the second semester was CySim running on a server.
  - There was a lot of contemplation on whether we were going to design CySim a specific way based on what type of server we were getting.
  - We ended up not using a server yet so we fully built CySim using the .NET framework and from there the server will work around it in the future.
- Team Tidbits
  - This section is completely built for our experiences working in a team. Below you will find some better than average advice from our members about collaboration as well as some fun parts working together.
  - **TIP:** By far the easiest way to have issues and just all around confusion is through lack of communication. Almost every single time we had a case of this, something went wrong or had to be redone.
  - One thing that was really cool to see was members adaptations off of eachother, you see this throughout the web application where one team member developed a concept and then another member took that concept and made it look and perform even better.
  - Throughout the semester we found that our team meetings were some of the most productive time settings that we could have. This is where we would discover what needed to be done moving forward and what we could edit or change to make more efficient in regards to the design of the CySim web application.
- Final Considerations
  - The team as a whole went through and debugged the rest of the web application towards the end of semester to look for any small bugs that may be hidden or not found.

- Due to this we were able to find bugs that needed to be fixed and helped CySim look the way it does now.

### **Appendix III - Source Code and How to Obtain:**

At this moment the CySim source code is hosted on GitHub. That being said, it is currently not publicly available. However, authorized individuals can be added to the repository in order to download the source code. Once downloaded from the option of your choice, please see Appendix I for installation instructions. It is recommended that these be opened via Microsoft Visual Studio as this is a .NET based project.