Project Phase III

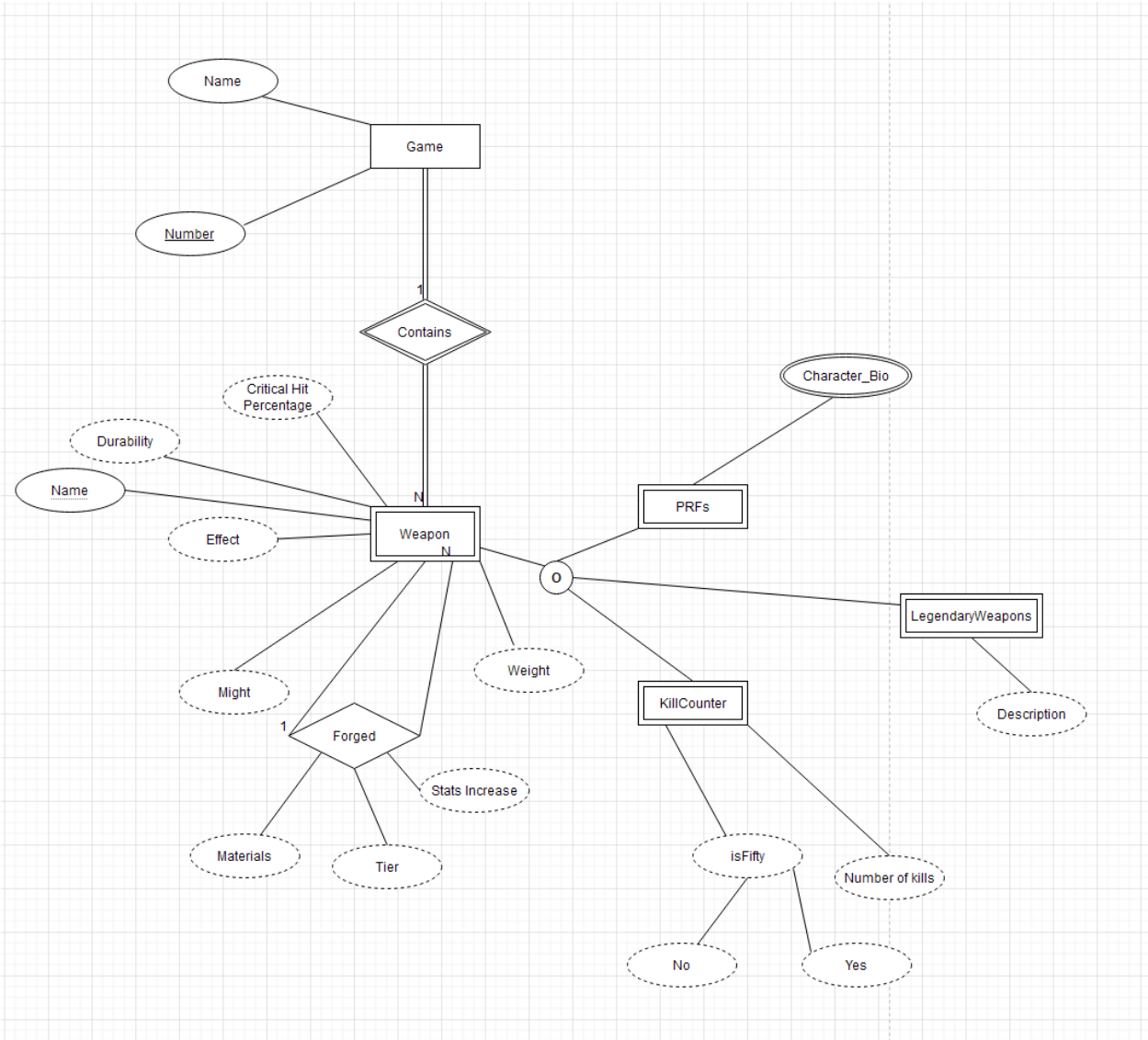
Members: Dominick Dickerson, Andrew Kalous, Alex Rushing.

Group Name: Armads

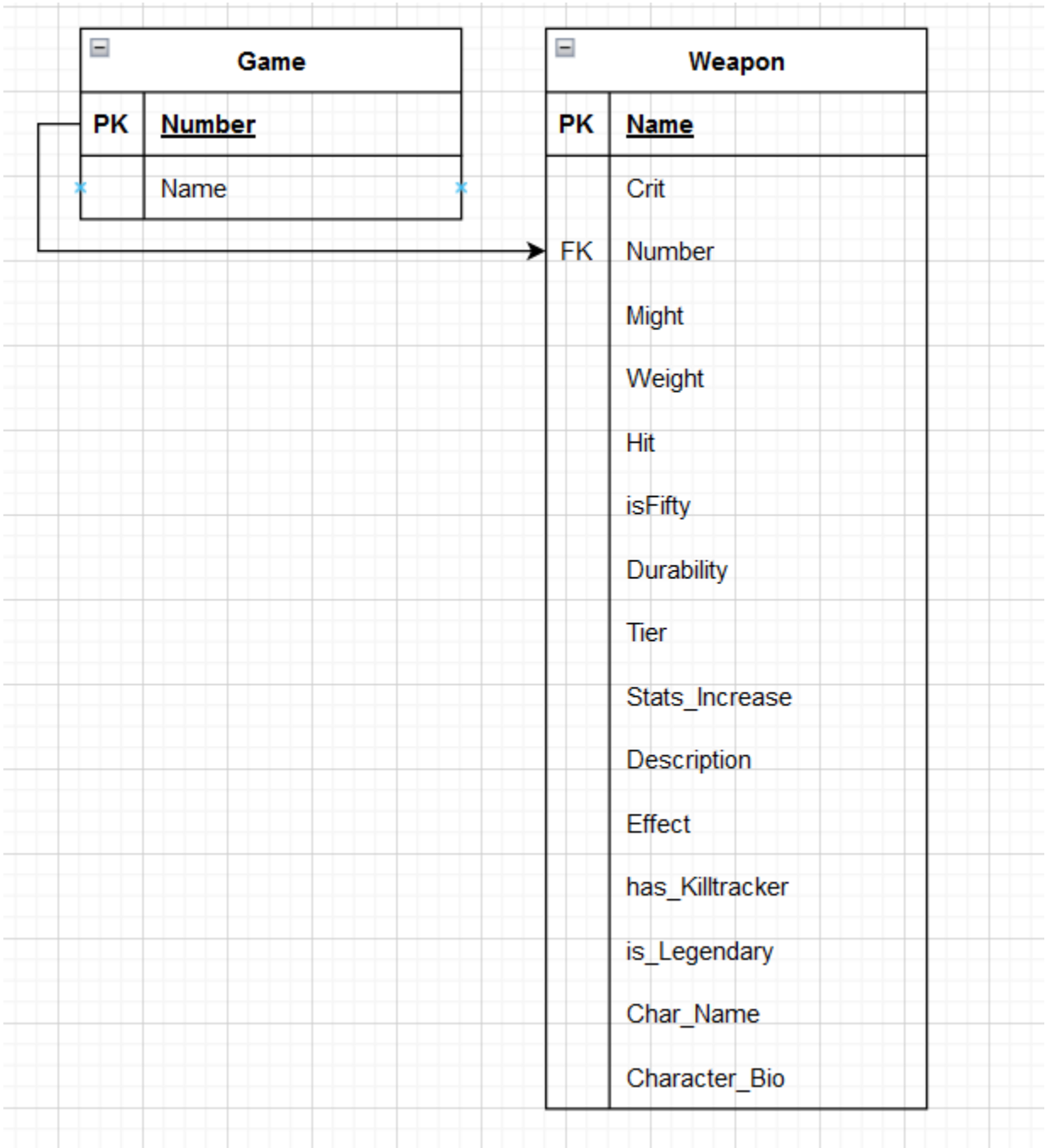
Project Idea/Problem Statement (Revised):

Our project idea was to create a library of weapons from every game in the video game series, Fire Emblem. It will be a web app developed using ReactJS that will allow the user to view basic stats for each weapon such as name, weight, might, and effect. The goal is to create a user-friendly library for players that allows them to view weapons and interact with some portions of the app, this will include sorting and filtering by game and other categories, and a damage calculator that calculates damage of a weapon from a formula. The weapons are sortable by game along with other attributes. The initial state of the database includes only weapons from Fire Emblem 7 for convenience.

ER Diagram (Revised)



Relational Model (Revised):



Attribute Table (Revised):

Table	Attribute	Type	Constraint
Game	Number	int	Primary Key
Game	Name	varchar(255)	NOT NULL
Weapon	Name	varchar(255)	Primary Key NOT NULL
Weapon	Crit	float	
Weapon	Hit	float	
Weapon	Number	int	
Weapon	Might	int	
Weapon	Weight	int	
Weapon	isFifty	bool	
Weapon	Durability	int	
Weapon	Tier	int	
Weapon	Stats_increase	varchar(255)	
Weapon	Effect	varchar(255)	
Weapon	has_Killtracker	bool	
Weapon	is_Legendary	bool	
Weapon	Char_Name	varchar(255)	
Weapon	Character_Bio	mediumtext	

Application Program Design (Pseudocode):

```
Login()
        name = prompt for username
        password = prompt for password
        if(username= query table for match && password matches)
                logged_in = true
        else
                display error message

Add_weapon()
        //Initialize all values to null (or 0 where appropriate)
        if(weapon has effect)
                effect = weapon effect
        //repeat above if statement for all attributes (e.g. might, effect, crit rate, etc.). If false,
then attribute set to NULL or 0
        //All of the above implemented into a pop-up GUI with fillable fields, some required, that
the user can fill in in order to add weapon.

Modify_weapon()
        //Bring up GUI with fields for all attributes. Users can then modify/type in those fields to
modify the weapon attributes.

Delete_weapon()
        specify weapon object
        execute delete operation, setting corresponding FK's to NULL

Query_weapon()
        name = prompt for weapon name
        game_num = prompt for game number
        if(name= query weapon table && game_num = query game table)
                return weapon result
        else
                return error message

Calc_dmg()
        specify weapon object
        defense = prompt user for defense val
        strength = prompt user for strength val
        Return strength + might(from weapon object) - defense
```

Aggregation Functions:

Sort by max/min in GUI for might

Sort by names, alphabetically

Filter by game number

Intended Operating System: Windows
Intended Browser for web app GUI: Chrome (or other Chromium based browsers, like Edge)
Dependencies List:

- react: ^18.2.0
- react-dom: ^18.2.0
- react-router-dom: ^6.10.0
- react-bootstrap: ^2.7.4
- styled-components: ^5.3.9
- axios: ^1.4.0
- localforage: ^1.10.0
- mysql2: ^3.2.4
- sort-by: ^1.2.0
- match-sorter: ^6.3.1
- reactjs-popup: ^2.0.5
- web-vitals: ^2.1.4
- @testing-library/jest-dom: ^5.16.5
- @testing-library/react: ^13.4.0
- @testing-library/user-event: ^13.5.0
- react-scripts: 5.0.1

Installation Instructions are discussed in detail in the following User Manual:

Armads User Manual

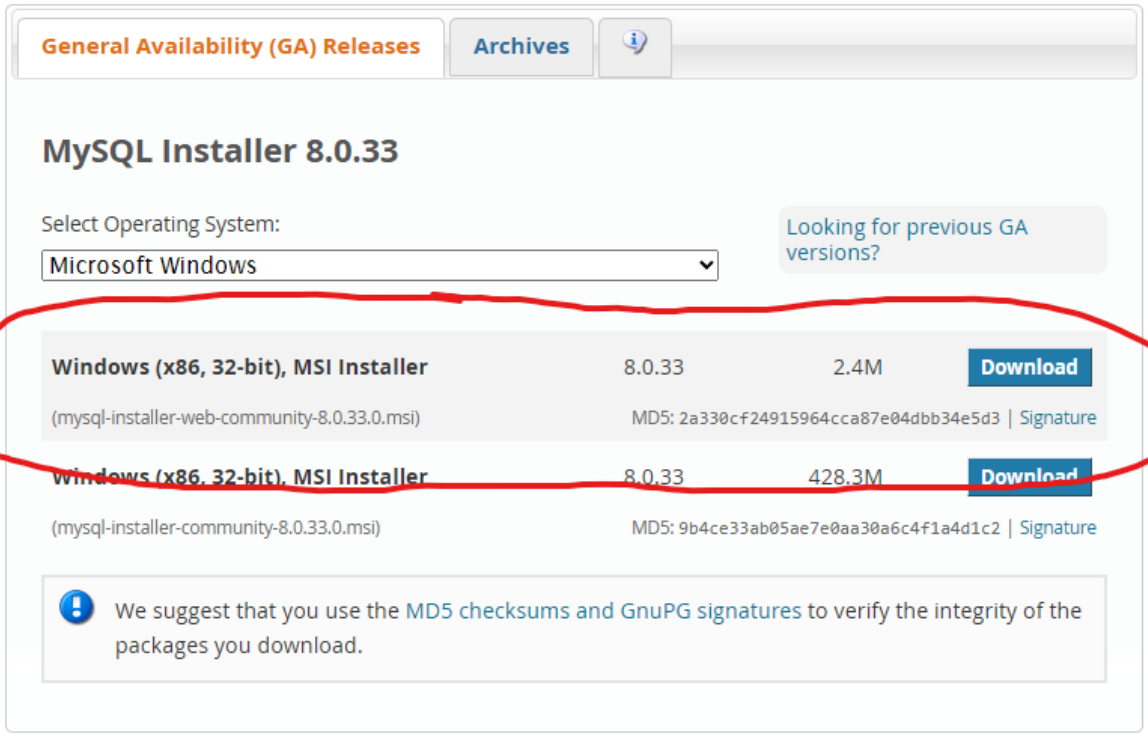
Project done by: **Dominick Dickerson**, **Alex Rushing,** and **Andrew Kalous**

Table of Contents

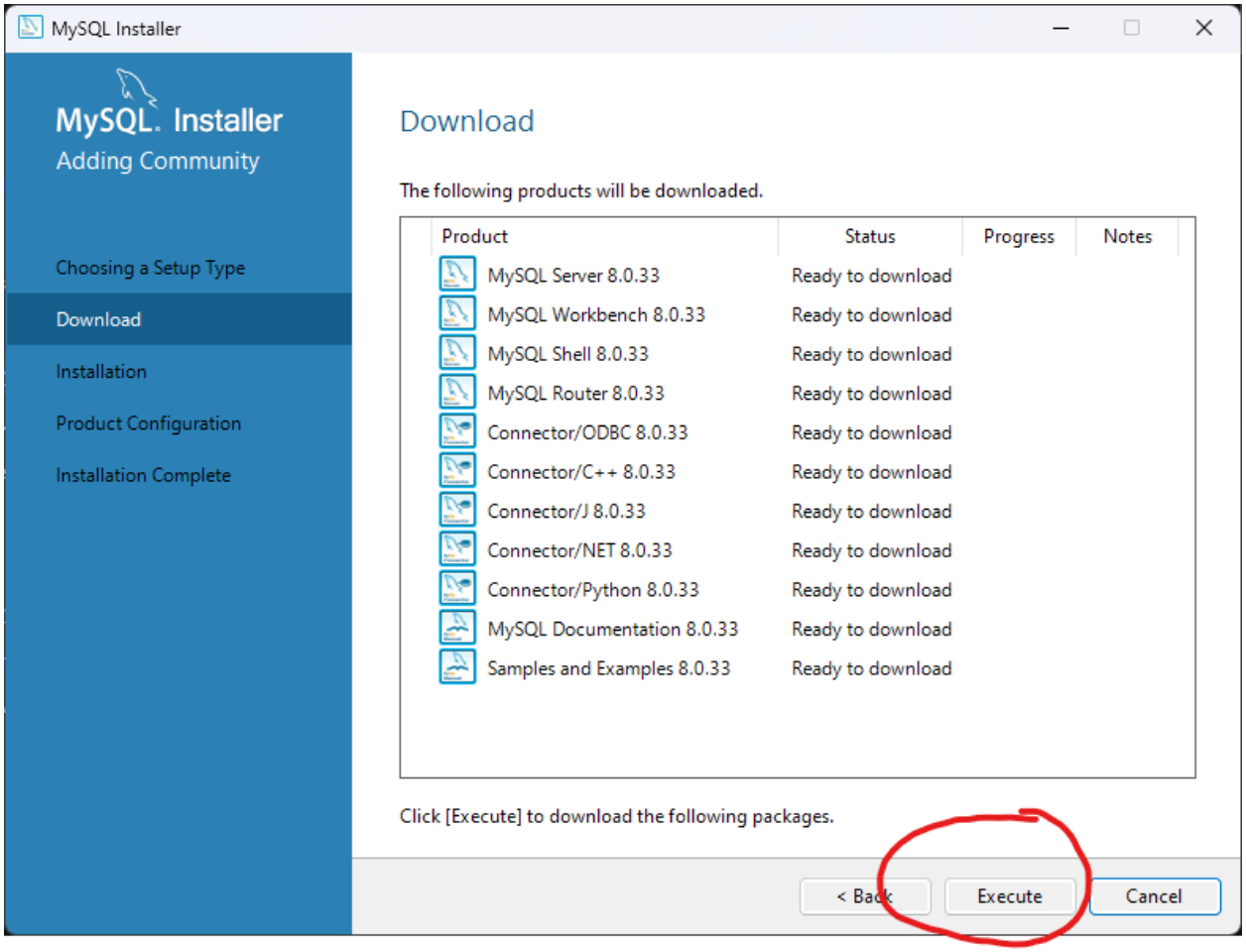
I. Installation

A: MySQL Installation

To install the project and all dependencies, begin by downloading the MySQL installer found at this link: [MySQL :: Download MySQL Installer]



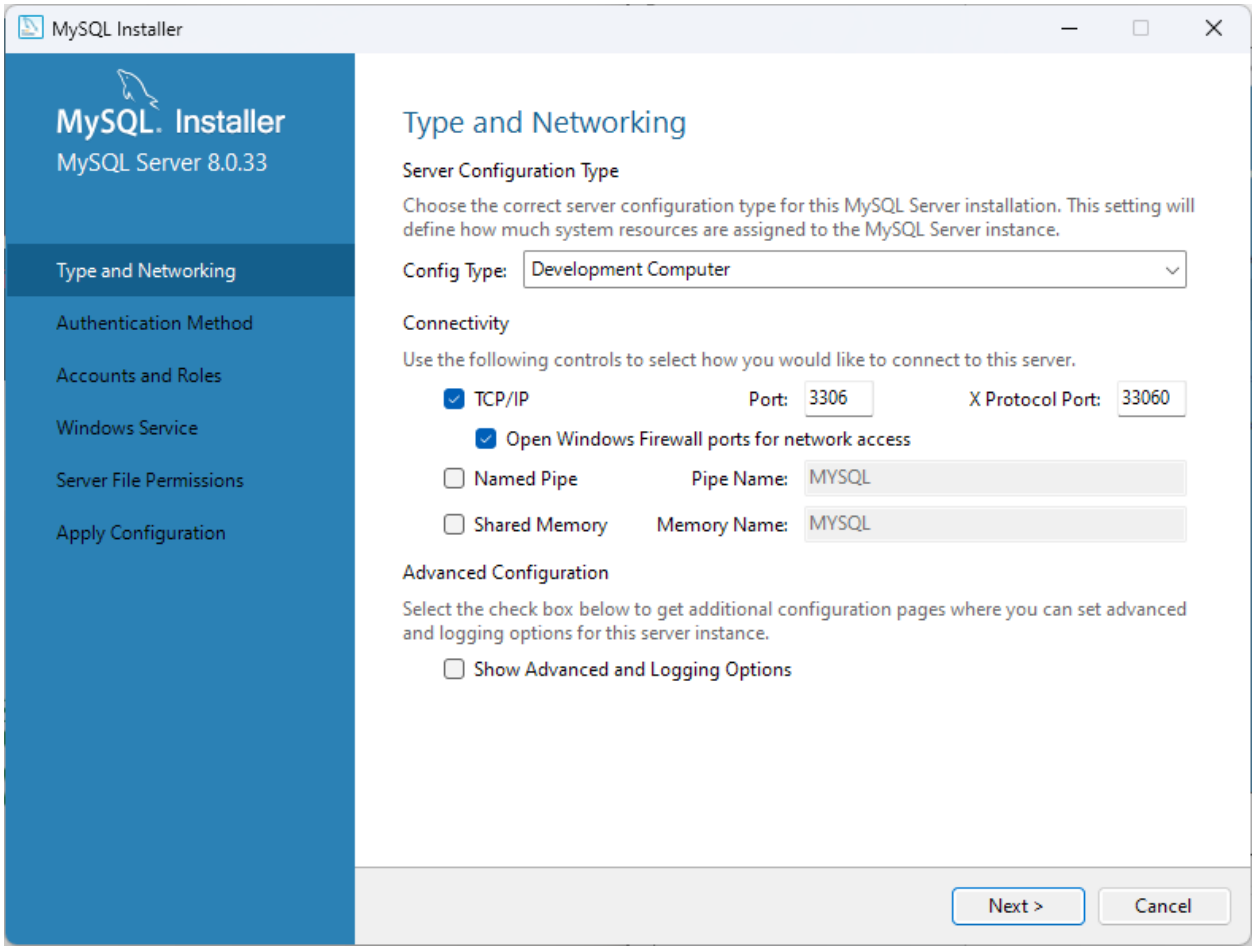
Once downloaded, open the .msi file that was downloaded and select “Developer Default” setup type. You will then click the “Execute” button to download all MySQL dependencies.



Once all of the packages are downloaded, click “Next” and then click “Execute” again to install all of the packages.

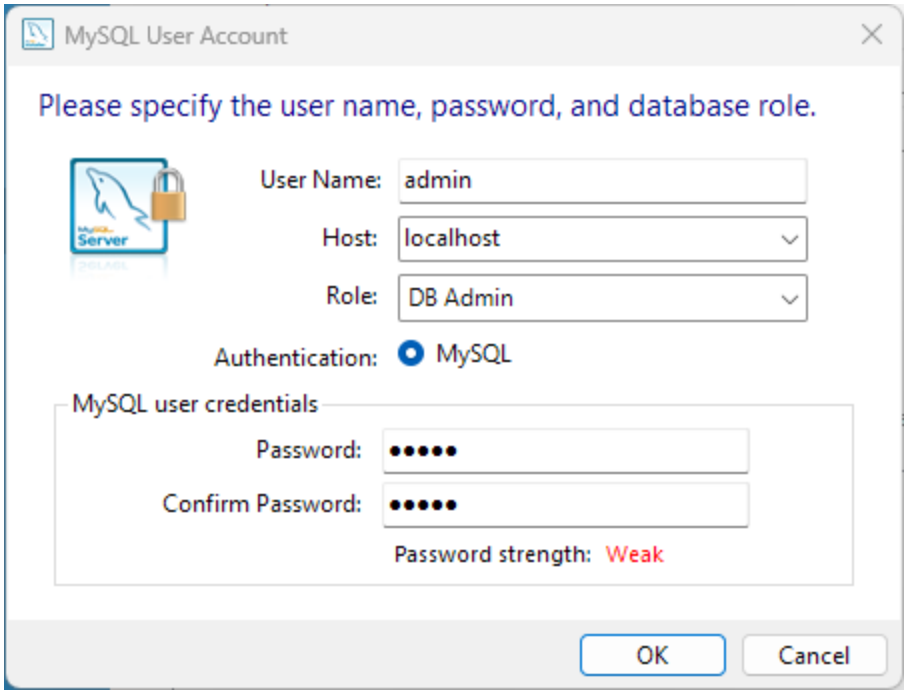
It may say that Python connector failed to install, that’s okay, don’t worry about it.

After the packages are installed, you will be put into a configuration page. Make sure you use the default settings for it. The default settings should look like this:

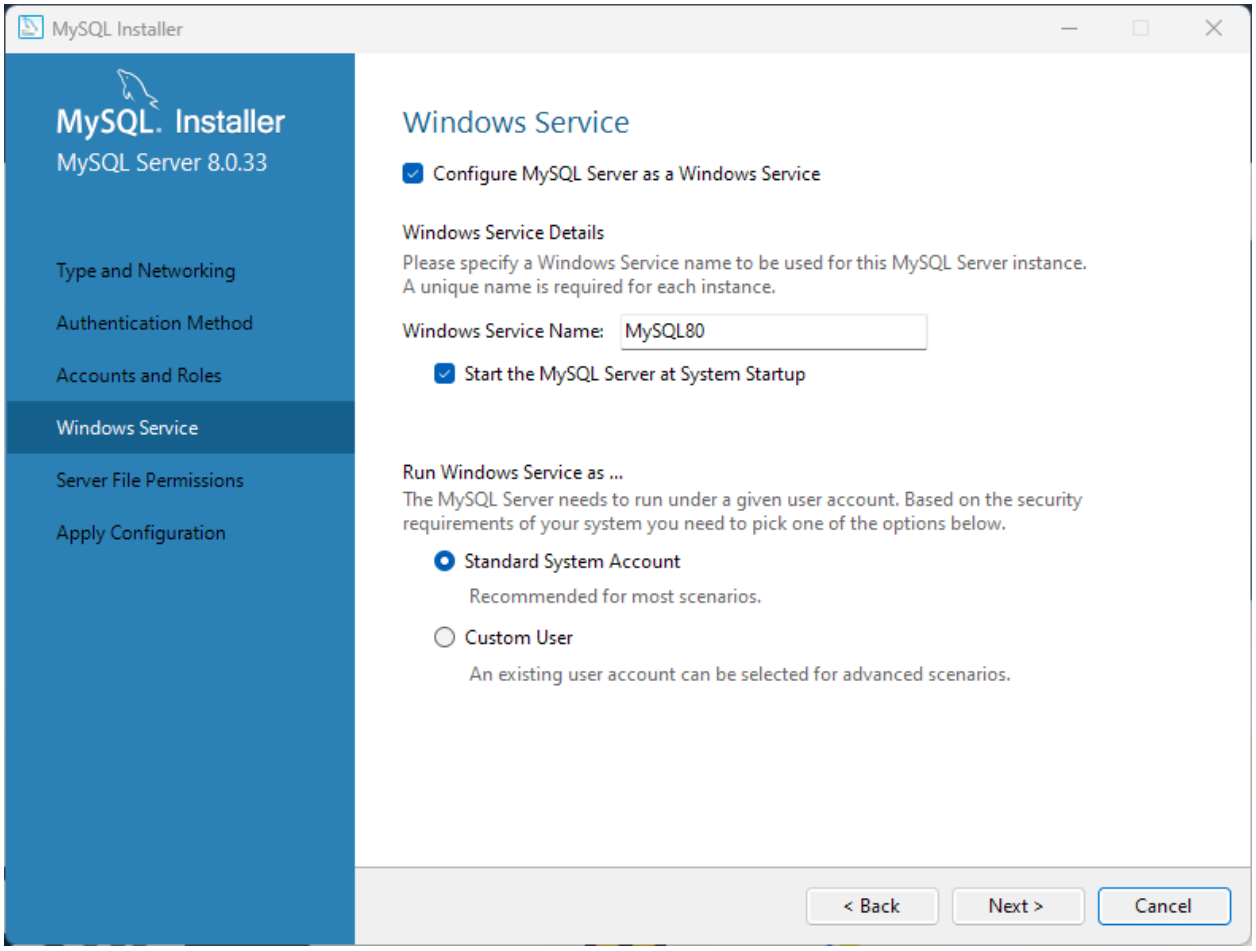


On the next page make sure the option selected is: “Use Strong Password Encryption for Authentication (RECOMMENDED)” and if it is, click the next button.

For the MySQL root password, enter “admin” as the password since the application uses that to login, once that is set, click “Add User”, make sure username is “admin”, host is “localhost” (if it is not, select it in the dropdown menu), and the password is “admin.” The Add User window should look like this:

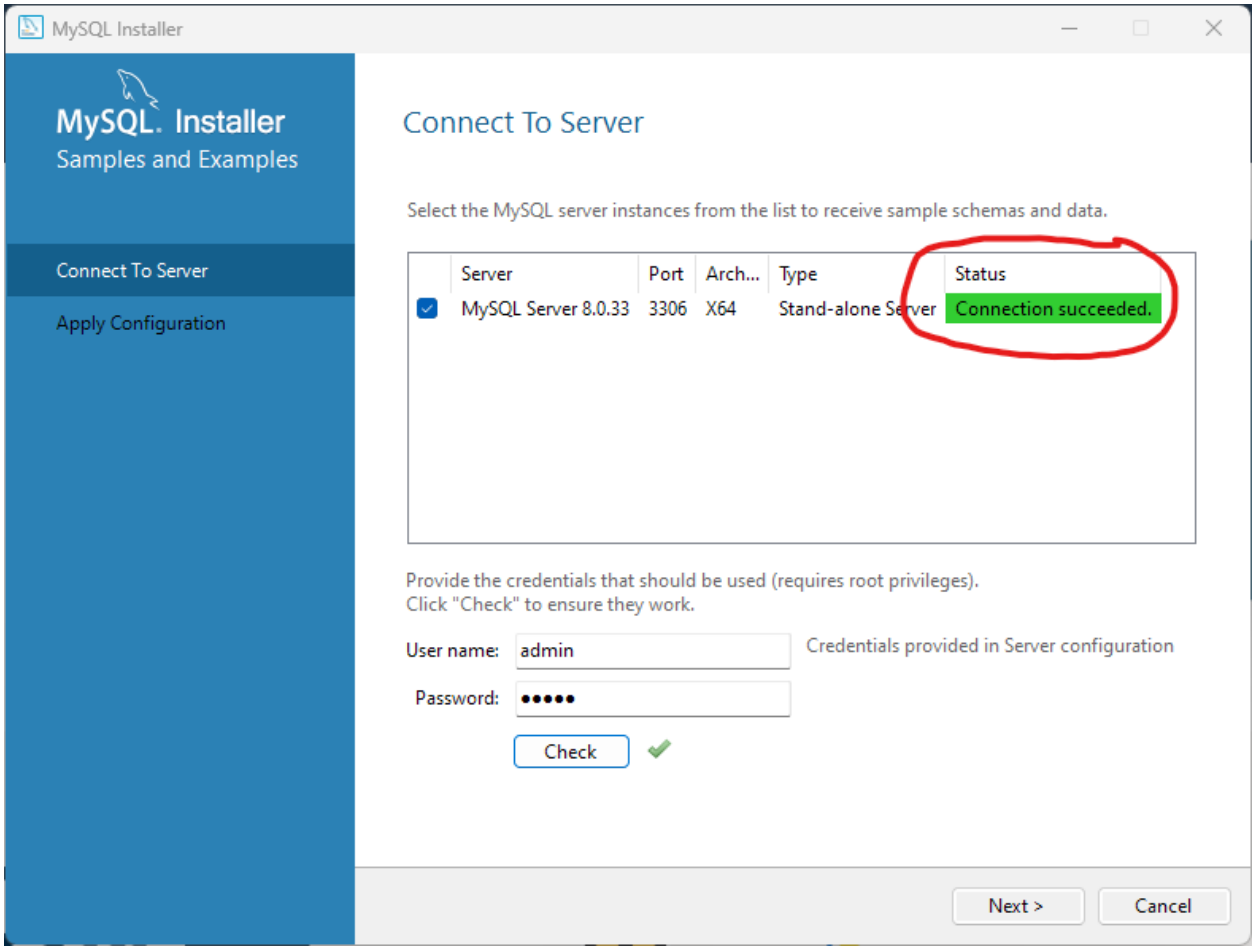


Hit “Next” and on the next page, leave everything as the default settings, it should look like this:



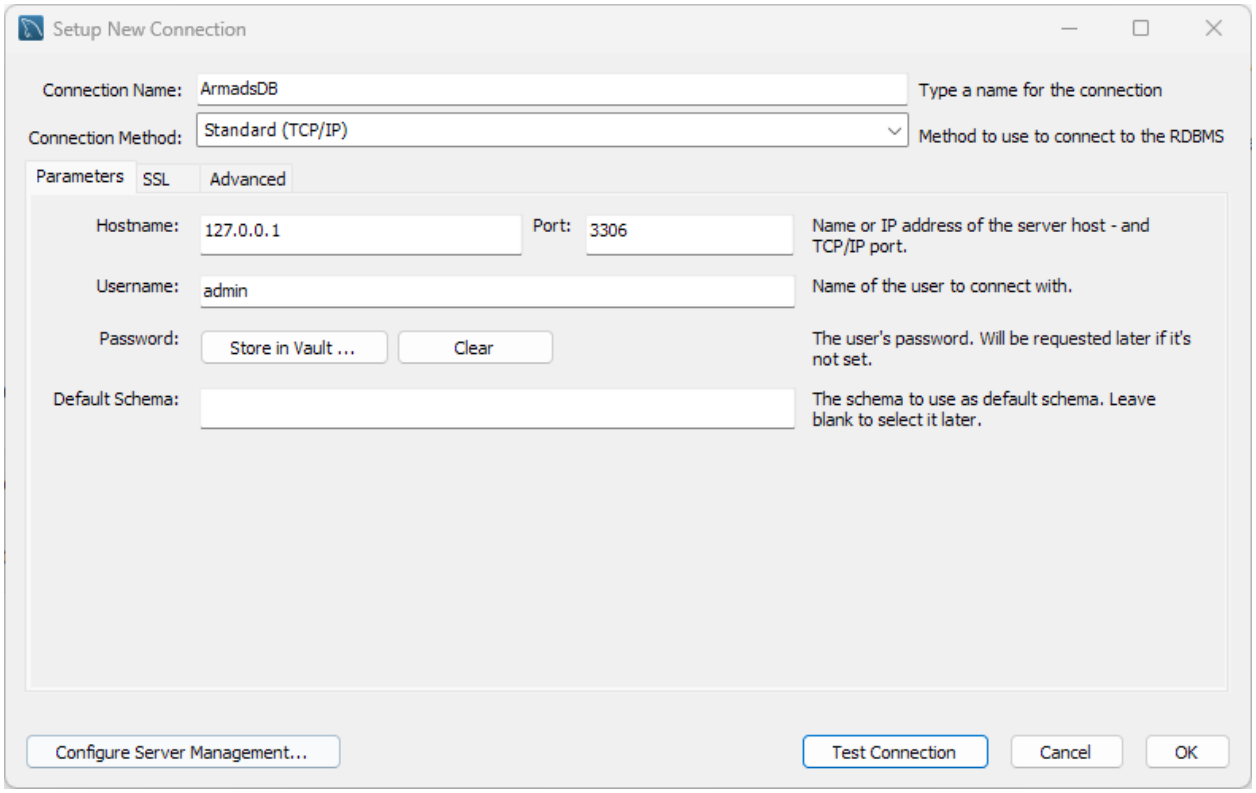
Again on the next page, make sure the option that is selected is “Yes, grant full access…” once that has been selected click next, and on the next page select “Execute”, this should now configure MySQL.

Keep hitting next until you come to a page that says “Connect To Server” on this page you will connect to the SQL server that you have installed. Type in “admin” for username and password, click check, and under status it should say “Connection Succeeded.”

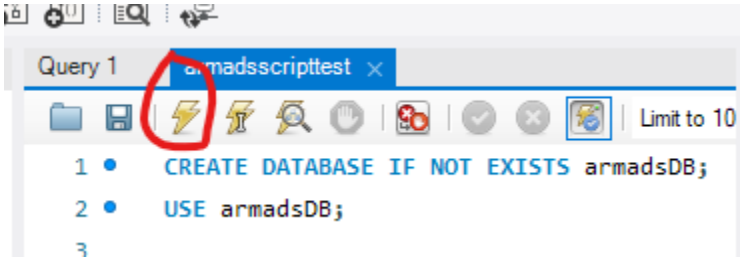


Click next and “Execute”, and keep clicking to the next page until the end, you will see a “Finish” button, after this has been done you can close the shell that pops up.

Navigate to the MySQL workbench and next to MySQL connections (ignore the default created connection), click the plus button, create a new connection with whatever name, default hostname and port, username as “admin”, click on the “Store in Vault…” and type in “admin” as a password. When finished click test connection and it should say “Successfully made the MySQL connection”, here’s what the window should look like:



When finished, open the newly created connection. When in the connection navigate to the top bar and click File > Open SQL Script > Navigate to the FEDatabase Folder > Select the armadsscripttest.sql. Once selected and opened in the MySQL workbench, click the first lightning button to make the database:



Once that is finished, the database is created and now we can move onto the next step.

B: Installing all dependencies

To install all the dependencies we need, you can follow this guide, or you can follow the steps in “RUN ME TO SET UP THE APP.txt” (minus the installing MySQL part)

First off, we need to open a terminal or command prompt and point it to the directory with the folder, this commands to do this are (assuming that the “FEDatabase” folder is in the Downloads directory):

cd %HOMEPATH%
cd Downloads
cd FEDatabase
cd FEDatabase (This is only needed if the folder is inside the originally extracted folder.)

Once the terminal is pointed at your database folder (to verify this you can run “dir” in the command prompt and it should show you all the files inside of it).

```
PS C:\Users\domin\Documents\FEDatabase> dir


    Directory: C:\Users\domin\Documents\FEDatabase


Mode                 LastWriteTime         Length Name
----                 -------------         ------ ----
d-----          5/2/2023   6:10 PM                build
d-----          5/2/2023   8:19 PM                FEDatabase
d-----          5/2/2023  10:09 PM                node_modules
d-----         4/22/2023   6:03 PM                public
d-----          5/1/2023   2:16 PM                Server
d-----          5/2/2023   4:41 PM                src
-a----         4/20/2023   8:39 PM            310 .gitignore
-a----          5/2/2023   4:35 PM           5795 armadsscripttest.sql
-a----          5/2/2023   7:43 PM            168 install.ps1
-a----          5/2/2023   7:43 PM             76 installDependencies.ps1
-a----          5/3/2023   3:39 PM       31539200 node.msi
-a----          5/2/2023  10:09 PM         697017 package-lock.json
-a----          5/2/2023  10:09 PM           1077 package.json
-a----         4/20/2023   8:39 PM           3359 README.md
-a----          5/2/2023  10:34 PM           2645 RUN ME TO SET UP THE APP.txt
-a----          5/2/2023   7:43 PM             60 startApp.ps1
-a----          5/2/2023   6:28 PM             75 startNode.ps1
```

You should run the following command:
“powershell -NoExit -ExecutionPolicy Bypass -File install.ps1”
which will download the node.js installer as a .msi installer called “node.msi” (if for some reason this fails, you can go to “https://nodejs.org/dist/v18.16.0/node-v18.16.0-x64.msi” to download it.

Run the node.msi installer (you do not need to check any boxes besides the license agreement.) Once it is installed you should close the command prompt and open a new one, point it at the same folder, the commands are below:

cd %HOMEPATH%
cd Downloads
cd FEDatabase
cd FEDatabase (This is only needed if the folder is inside the originally extracted folder.)

Now you should run “npm -v” this should return a version number, if it does you have successfully installed node.js.

```
C:\Users\domin\Documents\FEDatabase>npm -v
9.5.1
```

Now, in the same terminal, run the following command:
“powershell -NoExit -ExecutionPolicy Bypass -File installDependencies.ps1”
this will install all dependencies for the application.

Now, run the following command in the same command prompt window:
“powershell -NoExit -ExecutionPolicy Bypass -File startNode.ps1”
this will start node.js. If this is successful, you will get a message that says “MySQL Connected…”

```
Starting Node.JS...
Server running on port 5000
MySQL Connected...
```

Start a new terminal without closing the MySQL Connected window, point it to the same folder as before, using the same commands before, and run the following command:
“powershell -NoExit -ExecutionPolicy Bypass -File startApp.ps1”
this will start the actual application in a browser window. If all goes well, the application should be running with a server.

IMPORTANT NOTE, THIS APPLICATION RUNS THE BEST IN A CHROMIUM BROWSER (EDGE, CHROME), IF THIS IS STARTED IN FIREFOX, NAVIGATE TO “http://localhost:3000/” IN A CHROMIUM BROWSER

*If you get an error about upgrading MySQL, run this command: ALTER USER 'root'@'localhost' IDENTIFIED WITH mysql_native_password BY 'password'; as a SQL query, if that does not work, remove the @'localhost', replace the 'password' and 'root' with your password and

username respectively, then run the command: flush privileges; as a SQL query, both of them in the MySQL Workbench*

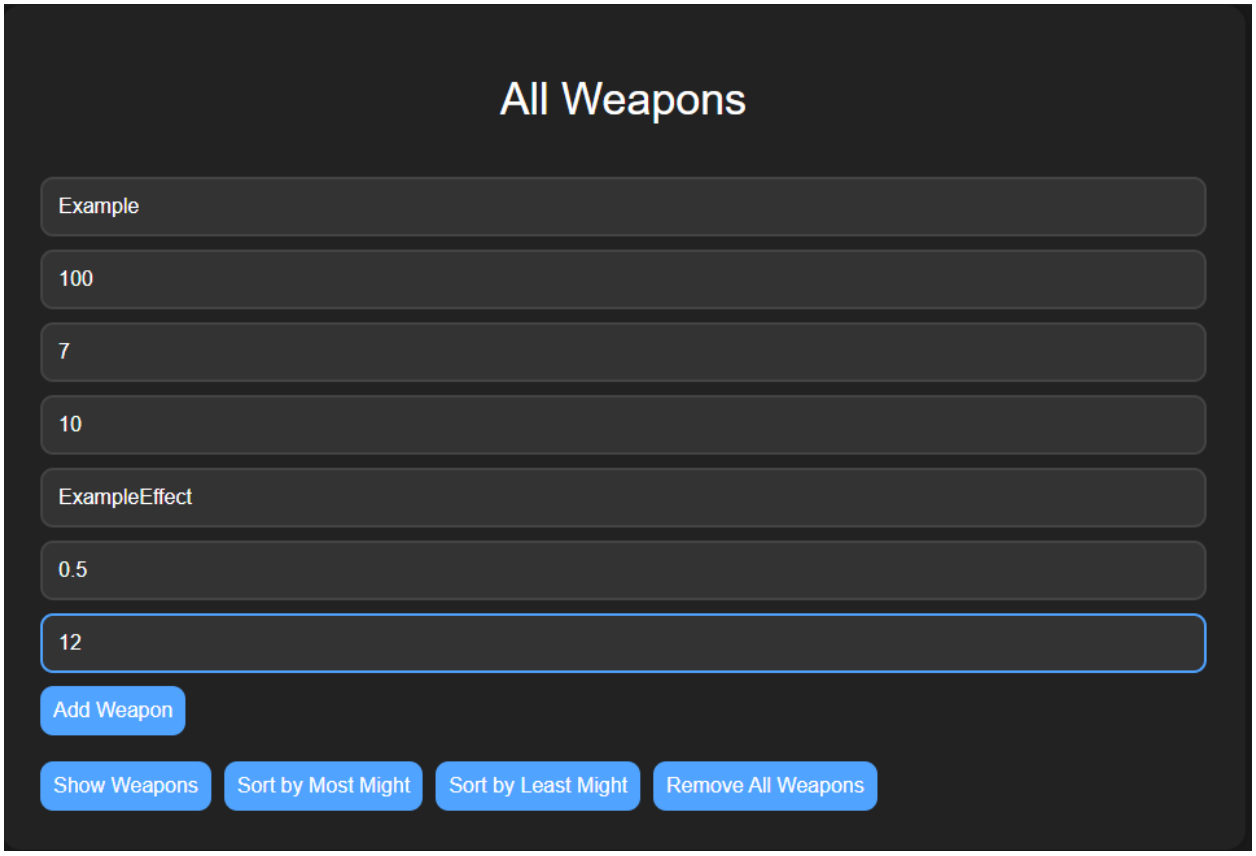
```
1 •  ALTER USER 'root'@'localhost' IDENTIFIED WITH mysql_native_password BY 'password';
2 •  flush privileges;
```

Next we will take a look at the features of the application.

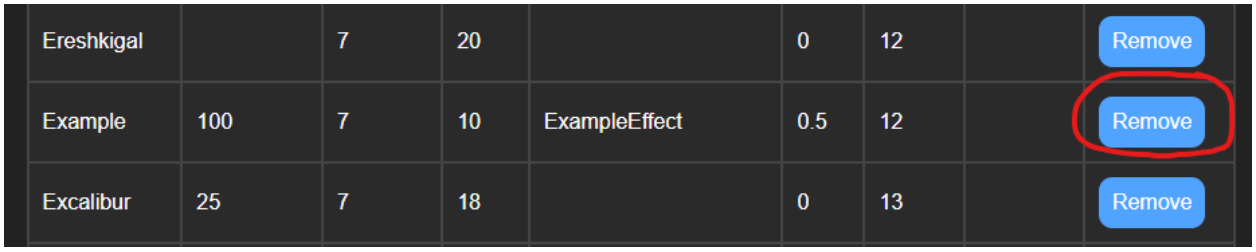
II: Application Features

A: Weapons Page

The weapons page is the default page you will be entered into, to add a weapon into the database, enter the details in the form provided:



Then click “Add Weapon”, in order to view all weapons, including the weapon you just added, click the “Show Weapons” button (this button also acts as a refresh). In the table that drops down you can see all weapons and certain stats, in order to remove a single weapon, click the “Remove” in the weapon table, in the column of whatever weapon you want to add.

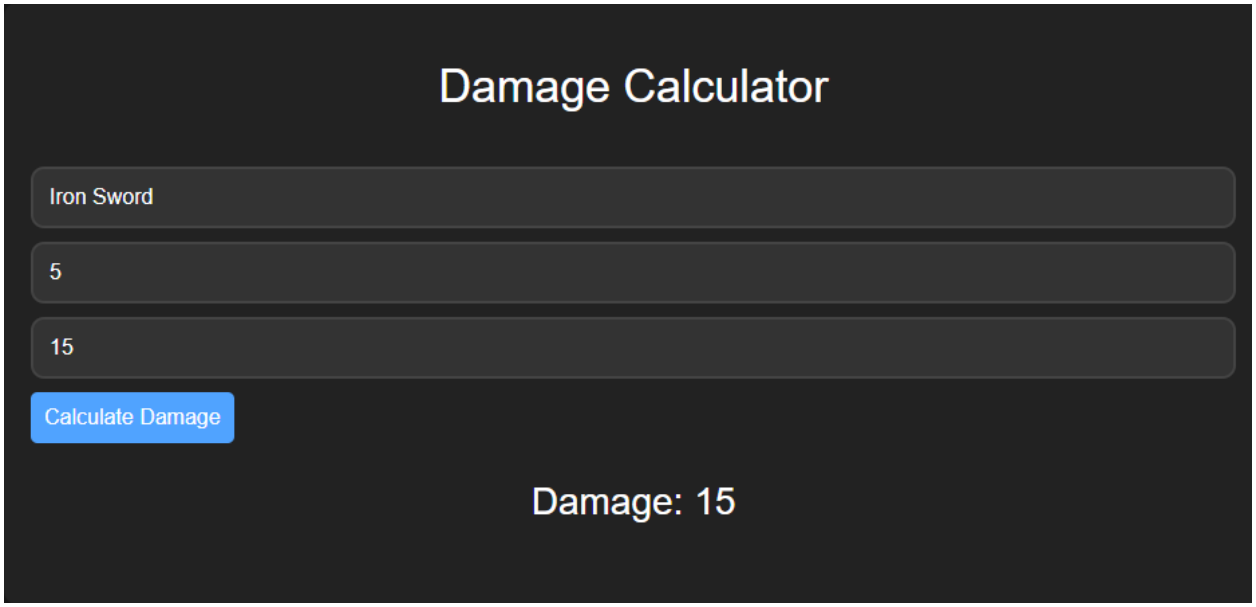


To remove all weapons, you can click the “Remove All Weapons” button at the bottom. Going back to the top, you can sort the weapons by most and least might by clicking their respective

buttons. To return to this page anytime, simply click the “Weapons” button on the top bar. Let’s navigate to the “Damage Calculator” page by clicking the button at the top.

B: Damage Calculator

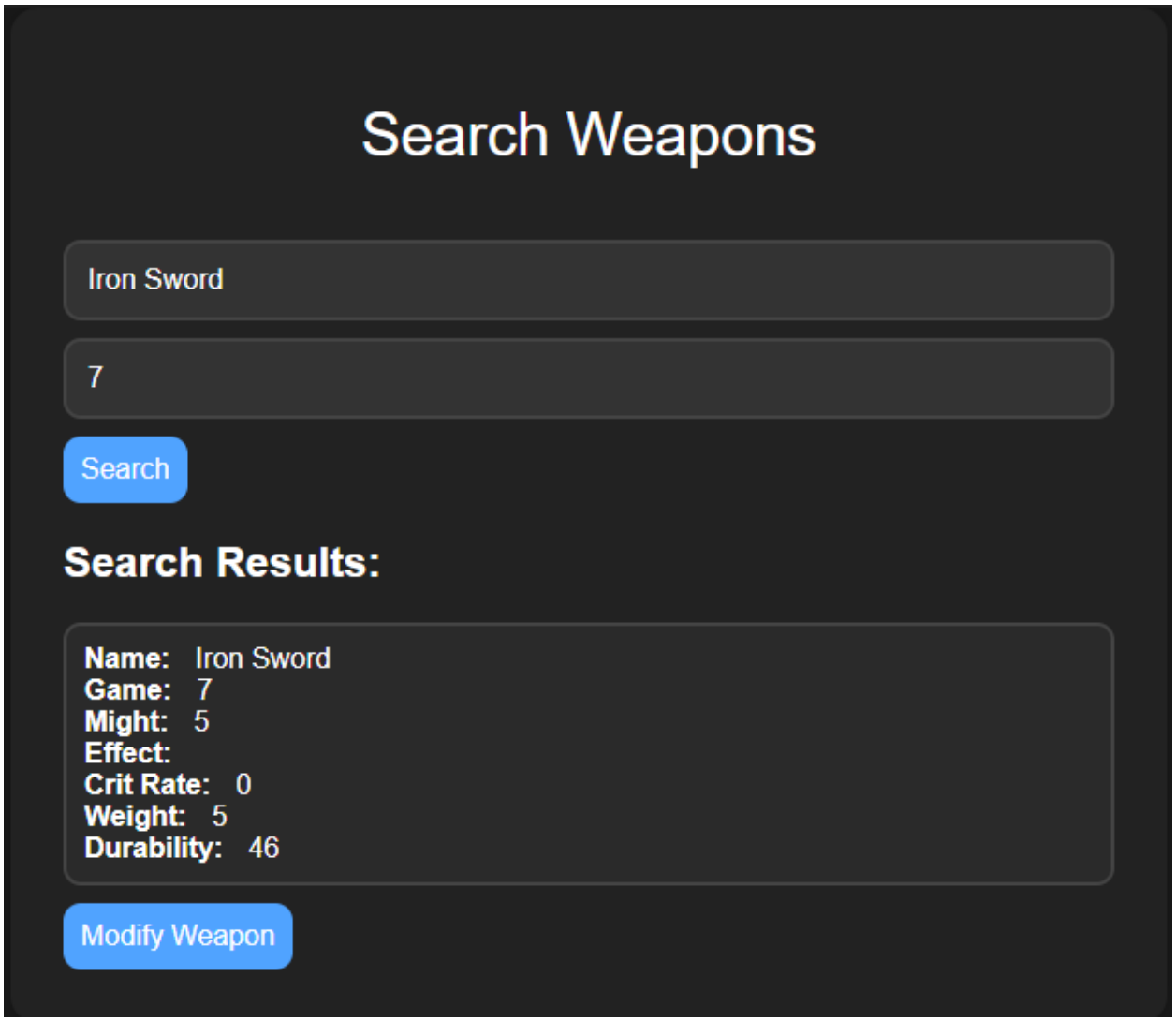
Next, on the damage calculator page, to calculate the damage of a weapon, input the name of the weapon, defense, and strength. Note that the defense and strength will automatically update but the weapon might, will not until you click the “Calculate Damage” button. The damage will show under the table.



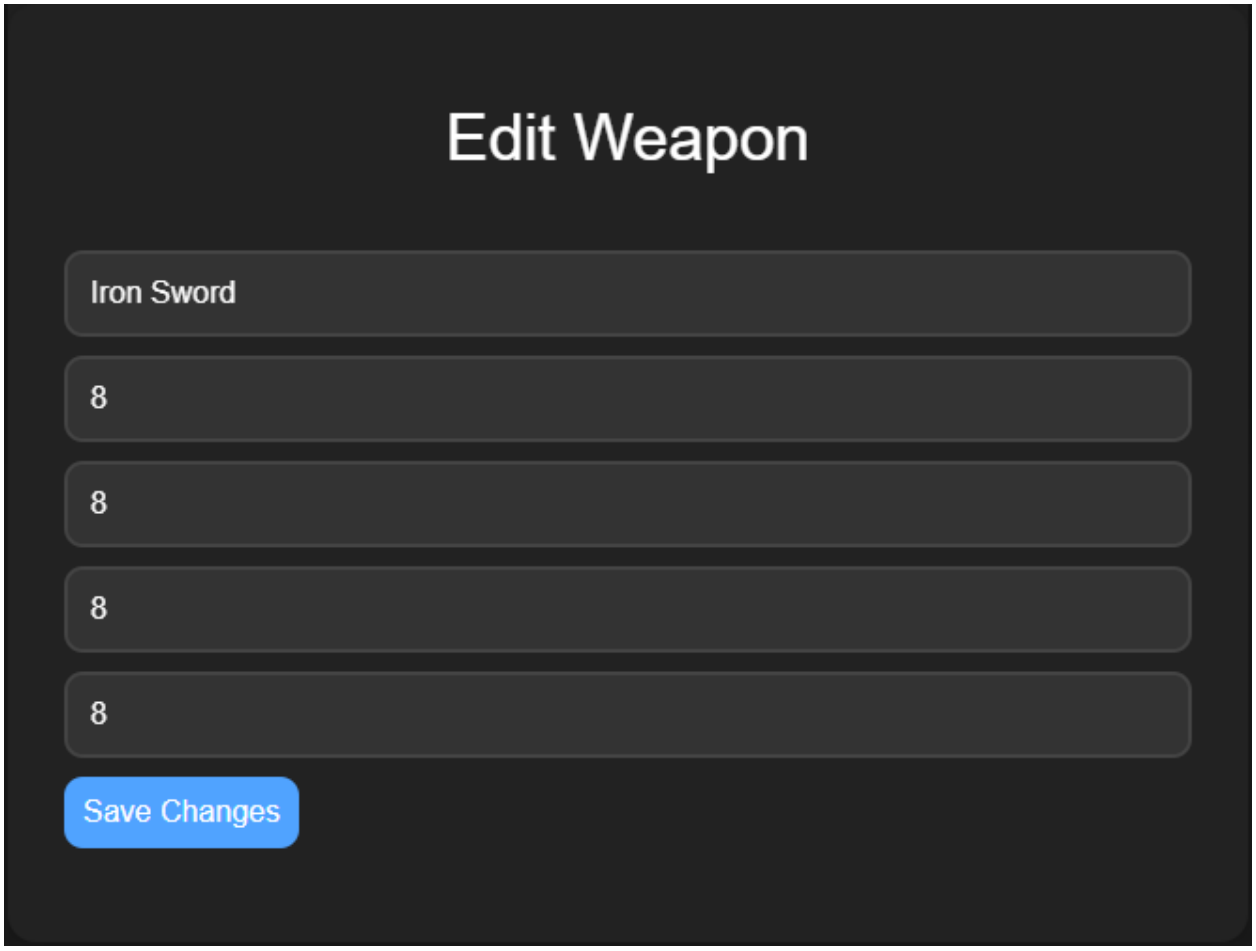
You can navigate back to this page by clicking “Damage Calculator” at the top of the page. Next we will navigate to the final page, the search page by clicking on the “Search” button at the top.

C: Search

Finally, the last page, the search page. In order to search for a weapon, input the name and game number of the weapon you want to search for and click “Search”, if the query returns any results they will be printed below, if not nothing will happen.



You might have noticed that a “Modify Weapon” button, when clicked this will pop up a new box that allows you to select a weapon by name, and modify certain stats of it, in order to modify the weapon simply input the name of the weapon to modify and the stats that you want to modify, and click “Save Changes”



To confirm that the changes have taken place, you can search for the weapon that you modified. And thats it, thats the final page. Thank you for the time and patience.

III: Credits

This web application was made by Dominick Dickerson, Alex Rushing, and Andrew Kalous. The website was written in JavaScript using ReactJS, is connected to the database with NodeJS, and the database used is MySQL. We hope you enjoy our website, we worked really hard on it. Thank you for your time!