CyberBlock Getting Started

Setting Up the CyberBlock Webserver

Assumption:

1. The server will be run on a bare-bones Ubuntu machine running the current GA kernel version v20.04
2. The machine has the newest version of *apt* package manager

CyberBlock Instructions:

1. Install Node.js by using command:

```
$ sudo apt install nodejs
```

2. Check that the installation was successful by using command:

```
$ node –v
```

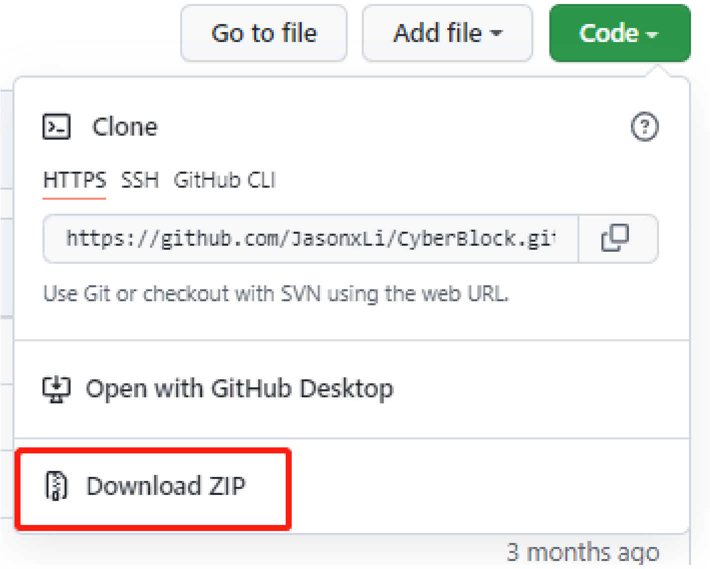
If it outputs something like v10.19.0, then the installation was successful

3. Install npm, the Node.js package manager by using command:

```
$ sudo apt install npm
```

4. Download code from https://github.com/brwnshoecrew/CyberBlock

Either via GitHub GUI:



```
https://github.com/JasonxLi/CyberBlock.gi
```

Use Git or checkout with SVN using the web URL.

Or by using command:

```
$ git clone https://github.com/brwnshoecrew/CyberBlock
```

Make sure you are on the right branch by using the command:

$ git checkout deploy

5. Unzip the file if it is downloaded via GitHub GUI.
6. Go into the zipped folder, and install required modules by using command:

$ npm install

7. Open two command line interfaces and traverse to client and server directory.
8. The commands above install all the libraries required for the application to compile.
9. The server is ready to be started by using command:

$ npm start

10. Run npm start on both directory the server will run on port 3001 and the client side on port 3000.

*Note if you want to change port number the app runs on:

For client side (React):

1. Locate *client/package.json*

2. Modify *"start"* key-value pair under *"scripts"* to specify port number, 3006 is used in the example below

```
"start": "PORT=3006 react-scripts start"
```

3. Locate *server/index.js, line 25*

Change *origin: http://localhost:3000* to the new port number you just specified.

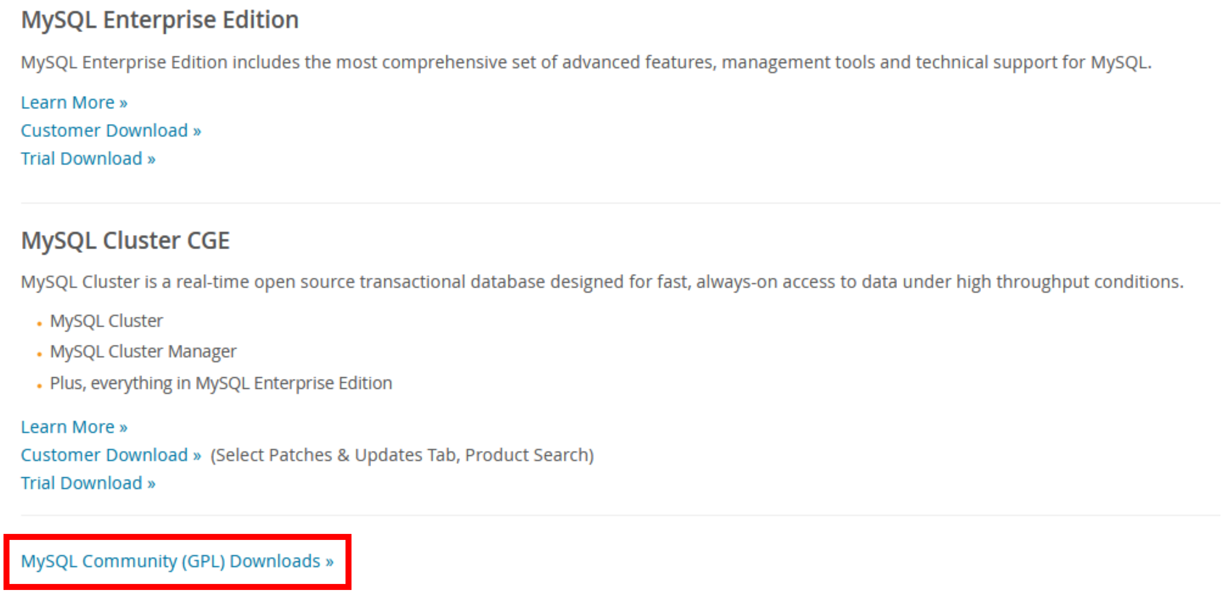
For server side (Node):

1. Locate *server/index.js, line 12*. And then change 3001 to the port number you want, such as 3007.

2. Locate *client/src/context/ContextProvider.js, line 5,* and change *const socket = io.connect("http://localhost:3001");* to the new port number you just specified, such as *const socket = io.connect("http://localhost:3007");*

Database Instructions:

1. Go to: https://www.mysql.com/downloads/
2. Scroll to the bottom of the page and click on "MySQL Community (GPL) Downloads"



MySQL Enterprise Edition

MySQL Enterprise Edition includes the most comprehensive set of advanced features, management tools and technical support for MySQL.

Learn More »
Customer Download »
Trial Download »

MySQL Cluster CGE

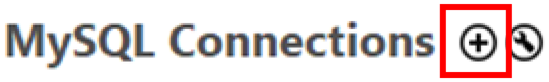
MySQL Cluster is a real-time open source transactional database designed for fast, always-on access to data under high throughput conditions.

- MySQL Cluster
- MySQL Cluster Manager
- Plus, everything in MySQL Enterprise Edition

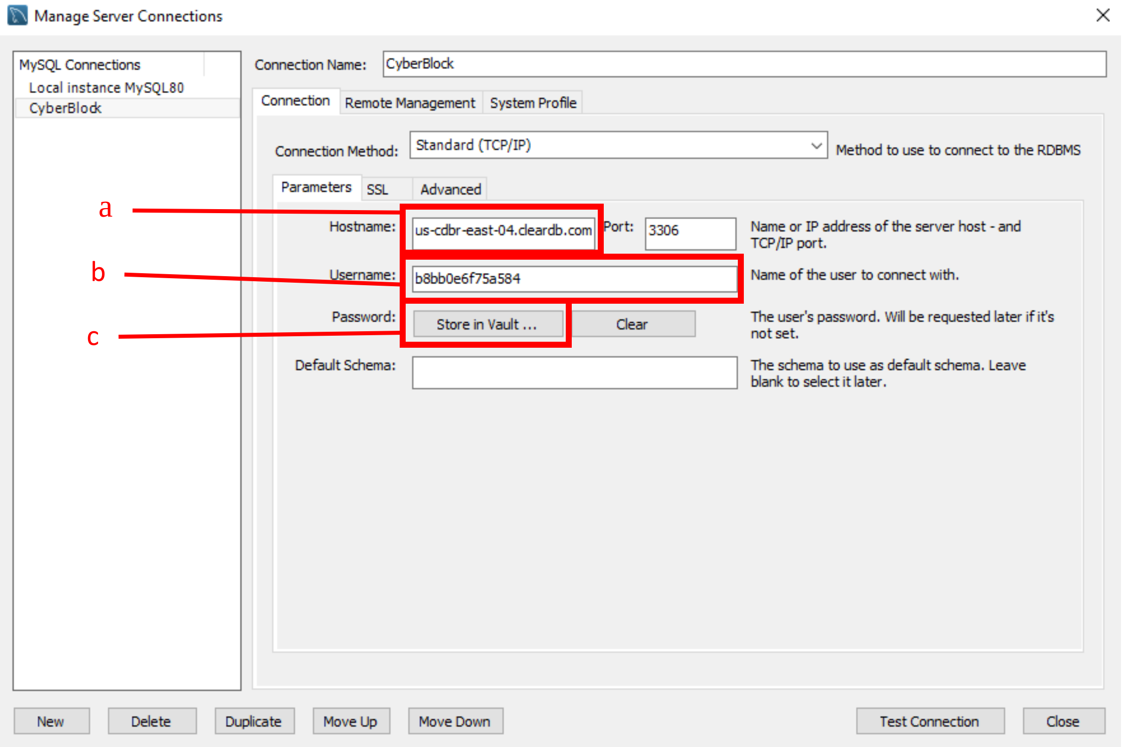
Learn More »
Customer Download » (Select Patches & Updates Tab, Product Search)
Trial Download »

MySQL Community (GPL) Downloads »

3. On the next page, click on “MySQL Community Server”
4. Download MySQL Community Server according to the operating system that you’re using
5. Once the download is complete, open MySQL Workbench
6. Click on the plus icon in MySQL Workbench



7. The following information will need to be put on the screen
   a. Host Name: us-cdbr-east-04.cleardb.com
   b. Username: b8bb0e6f75a584
   c. Password: 9d7d360a



Manage Server Connections

MySQL Connections
Local instance MySQL80
CyberBlock

Connection Name: CyberBlock

Connection Method: Standard (TCP/IP) — Method to use to connect to the RDBMS

Hostname: us-cdbr-east-04.cleardb.com Port: 3306 — Name or IP address of the server host - and TCP/IP port.

Username: b8bb0e6f75a584 — Name of the user to connect with.

Password: — The user's password. Will be requested later if it's not set.

Default Schema: — The schema to use as default schema. Leave blank to select it later.

8. Once the information has been input, click the “Test Connection” button
9. If the connection was successful, you now have access to the CyberBlock database

Read More on Clear dB: https://devcenter.heroku.com/articles/cleardb

Initialization

Data

The primary data matrix is primarily stored in an excel spreadsheet. The MySQL database relies on this matrix and is stored using MySQL workbench.

Constants/Settings

Package manager

- Node: To replicate the production and local environment. Refer to the instructions above.

GitHub Account

Account credentials for Heroku:

Username: benjamin.pope@medtronic.com

Password: @LEAP2021

Any cloud application for deployment

Change Set

The detailed commit history should reflect all the changes that we have made to the repository over time. However, some of the larger change sets are as follows:

1. Disconnection handling: User will be kicked out of the game in the event of any mishap.
2. Chat feature: Initially the game was set to only accommodate people physically present in the room, but the inclusion of chat feature allowed remote participation.
3. Leader Election: The leader was initially to be chosen by popular vote, but we opted to a round robin scheduling to allow fair participation.
4. Difficulty level: The application was created with scalability in mind and although there are three tiers of difficulty mode. The database is not large enough to support all three modes.