Deployment/Installation Instructions

Eric DiGioacchino

Jay Kerbelis

Claudia Usis

Alan Benitez

**Download links**

NodeJS: https://nodejs.org/en/

IntelliJ IDEA: https://www.jetbrains.com/idea/download/

MySQL Suite: https://dev.mysql.com/downloads/installer/

**A Note on Necessary Software:**

**NodeJS** – This project uses Node as the server platform. It’s simple to setup, easy to extend and extremely scalable, with minimal pitfalls.

**IntelliJ IDEA** – A far as powerful IDEs go, you don’t get much better than an IntelliJ IDE. This system handles our Java runtime for the project, as it is in development.

**MySQL** – Just about any SQL server would suffice for this project, but others require payment or encounter troublesome setup procedures. MySQL installer allows us to collect the three elements needed to get this project off the ground.

**How to Install:**

**NodeJS**:

**Windows**

1. Open the NodeJS URL in the ‘Download Links’ section
2. Click the link with the qualifier “Recommended for most users”
3. Open the downloaded msi file and follow the installation instructions in the prompt
4. Once complete, open a command terminal and enter “node” press enter and you should see the prompt waiting for input – this is a new node terminal. To exit the node terminal enter “.exit” and press enter.
5. Run ‘sudo apt install npm’ this will help us install packages later

**Linux**

1. Open a terminal
2. Enter the command string “sudo apt install nodejs” and press enter
3. You will be prompted in the terminal to select Y/n – select Y to continue
4. Let the terminal complete setup then enter “node” into the terminal and press enter and you should see the prompt waiting for input – this is a new node terminal. To exit enter “.exit” and press enter

**IntelliJ IDEA**:

**Windows:**

1. Open the IntelliJ IDEA URL in the ‘Download Links’ section
2. Click the Link marked “Community Edition” and allow the installer to download
3. Open the installer – and allow the program to make changes
4. In the installation options page check all of the check boxes
5. Continue following the prompt allowing the software to install
6. You should now have an IntelliJ icon on your desktop

**Linux**

1. Open the IntelliJ IDEA URL in the ‘Download Links’ section
2. Click the Link marked “Community Edition” and allow the installer to download
3. Extract the folder to your desired location - make sure it’s a place you’ll remember
4. Once the files are extracted open right-click the bin folder and select ‘Open in Terminal’ then type “./idea.sh” and press enter – this should open the Idea UI. – You can close this now
5. Open a new terminal –and follow the instructions supplied by https://gist.github.com/rob-murray/6828864

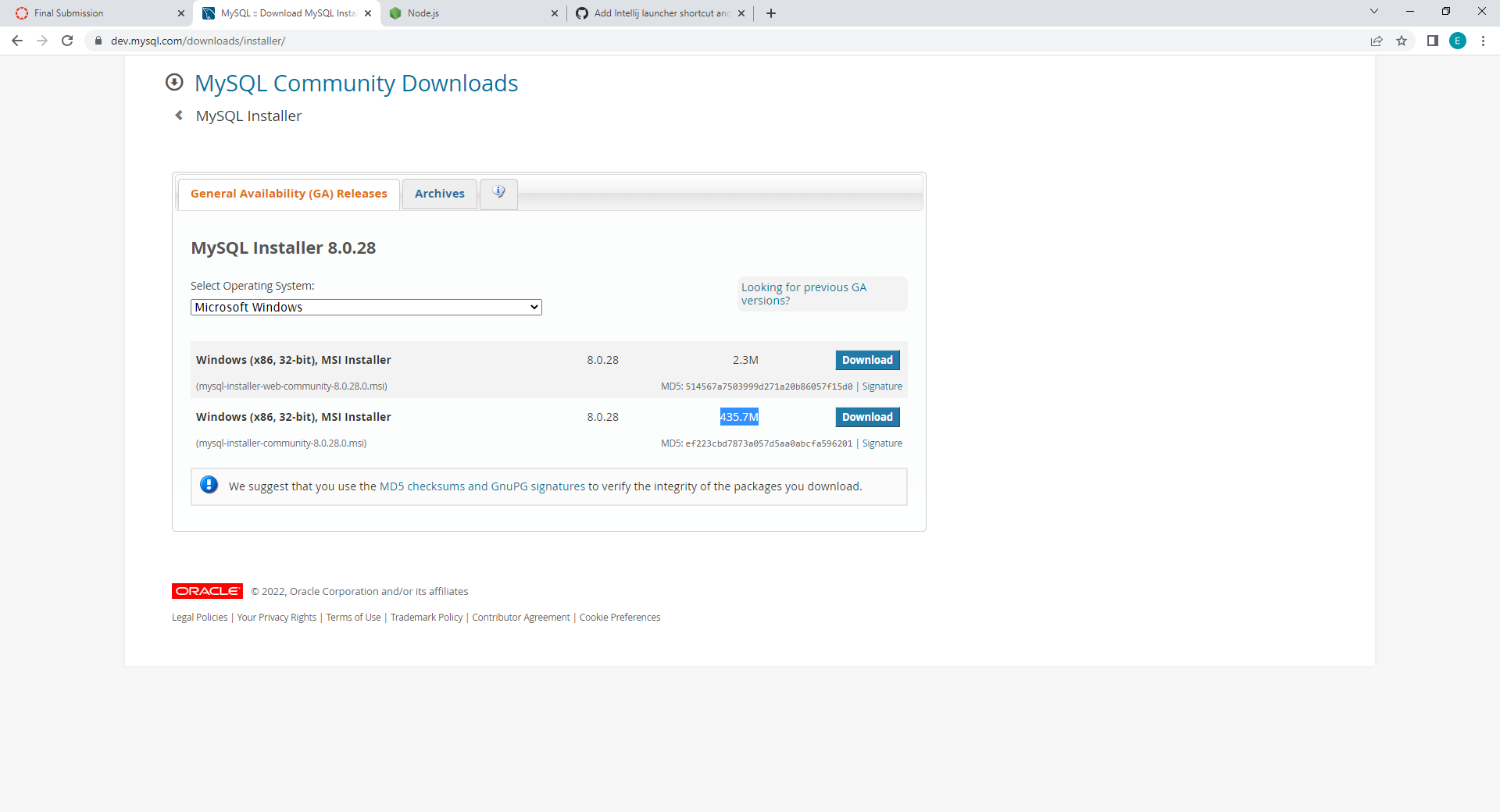
Making sure to replace the version and directory paths according to the current location of IntelliJ. You can find the version of IntelliJ by opening product-info.json

1. You will find the application shortcut in your apps, from here you can add it to favorites.

**MySQL:**

**Windows:**

1. Open the MySQL URL in the ‘Download Links’ section
2. Select the option that requires the most memory – this will ensure you don’t get any undesirable prompts.



1. When downloaded open the msi and follow the prompt
2. You will reach a page where you can select the products to install:

You will need: MySQL Workbench, MySQL Server, and MySQL Command Line Shell at the minimum, attempt to make the versions match

If given the chance – Download/Install as much of MySQL as you can

1. Follow the remaining prompts in the installer and install
2. Verify you have the Workbench shortcut in your applications by pressing the windows key and entering “mysql” in the search bar.

**Linux**

1. Locate and open Software Installer in your apps
2. Search ‘mysql’ in the search bar
3. Select “mysql-shell” and follow instructions to install
4. Select “Mysql Workbench Community” and follow instructions to install
5. Search “mysql” in your applications to verify installation
6. If you encounter an error connecting to your MySQL server instance run the following:

# If your server isn’t running

sudo apt update

sudo apt install mysql-server

sudo systemctl start mysql.service

#For errors with creating a root user

sudo mysqlSHOW VARIABLES LIKE 'validate\_password%';

install plugin validate\_password soname 'validate\_password.so';

SET GLOBAL validate\_password\_policy=0;

SET GLOBAL validate\_password.policy=0;

alter user 'root'@'localhost' identified with mysql\_native\_password by 'APIadmin';

flush privileges;

# For Ubuntu 20 only

sudo snap connect mysql-workbench-community:password-manager-service :password-manager-service

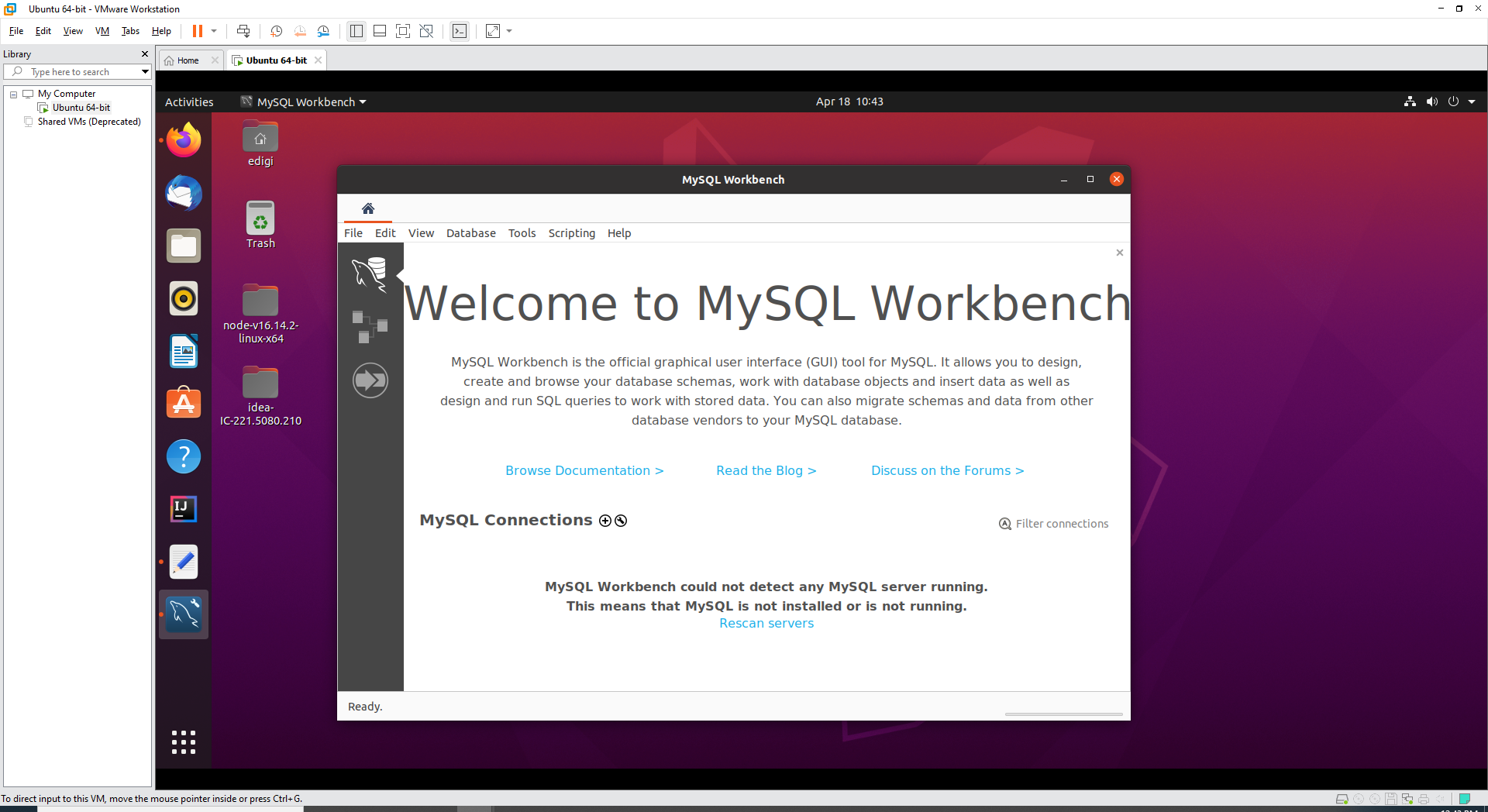
**Setup**

**Collect GitHub Project:**

1. Open a terminal in the directory where the project can reside ( I made a capstone folder inside my documents folder)
2. Run “git clone https://github.com/EricDiGi/ArgoNotes.git”

**MySQL:**

1. Open MySQL Workbench
2. Locate the ‘MySQL Connection’ section and click the ‘+’ button



1. Now we start building the connection by naming it “ArgoNotesDB” – test the connection
2. If all is running correctly, double-clicking the ArgoNotesDB connection should open the Database manager.
3. In the newly opened manager, on the left-hand side of the screen you will see an administration panel – select the ‘Users and Privileges’ option
4. Create a new user named ‘service’ with the password ‘service\_engine’ and update the Administrative Roles by selecting the DBA option then click apply.
5. If no errors occur, close the Users and Privileges tab.

**Seeding**

1. Open the master\_seed.sql file found in the SQL seeds folder of the project
2. Run it
3. In the Schemas Pane on the left side of the window show the tables in the argonotes schema
4. Mouse over the ‘users’ table and select the settings button
5. At the bottom of the newly opened page select the Triggers tab
6. Mouse over “AFTER INSERT” and click the plus sign.
7. In the editor open the file named ‘user\_activity\_init\_proc.sql’ and click Apply in the bottom right. And follow the prompt to apply.
8. Now mouse over “BEFORE DELETE” and click the plus sign.
9. In the editor open the file named ‘user\_activity\_del\_proc.sql’ and click Apply in the bottom right. And follow the prompt to apply.

**NodeJS Server**:

1. Open a terminal in the ‘Node Server’ directory of the project
2. In the terminal enter ‘node simpleServer.js’ and press enter – it should indicate a connection to MySQL.

**Java Program:**

1. Launch IntelliJ
2. Open the ArgoNotesApp directory in the IDE and give it a moment to resolve dependencies
3. Expand directories until the ‘UserApp’ class is visible
4. Right-click the class and select “Run ‘UserApp.main()’”

**Scaling Architecture**

**Server**

Instructions regarding scaling the NodeJS application with minimal downtime, please refer to:

https://www.freecodecamp.org/news/scaling-node-js-applications-8492bd8afadc/

This page outlines the process required to prepare the server for scaling. This feature is currently not built into the current version of this distribution.

**MySQL Database**

Instructions regarding the scaling of a MySQL service can be found here:

https://docs.rackspace.com/support/how-to/configure-mysql-source-replica-replication

This page outlines the process required to extend the functionality of a MySQL instance. This distribution is capable of being scaled horizontally from the parent instance.