NTU Virtual Judge

System Manual

Revision	Date	Revised By	Notes
1.0	October 20, 2018	Ang Teng Huat	First Revision

Table of Contents

1. Introduction

NTU Virtual Judge is an application that serves as the platform for users to host and participate in mock competitive programming contests. This document will provide instructions for setting up, configuring, and maintenance of the application.

2. Installation

This section outlines the installation steps of the Virtual Judge application. Any hardware or software system environment requirements will be provided.

2.1.Requirements

The following table shows the minimum production machine specification required for the Virtual Judge system.

Operating System	Ubuntu 16.04 LTS
Memory	4GB
Hard Disk Capacity	60GB

2.2.Installation and Configuration

2.2.1.Project Installation and Configuration

The source code can be obtained in the GitHub repository located at: https://github.com/Teng-Huat/virtualjudge

2.2.2.Environment Configuration

2.2.2.1.Network Configuration

Network connectivity is required for the web project. To configure, edit the production file located at *config/prod.exs*

```
config :virtual_judge, VirtualJudge.Endpoint,
http: [port: {:system, "PORT"}],
url: [host: "172.21.148.174", port: 80],
cache_static_manifest: "priv/static/manifest.json"
```

2.2.2.2.Online Judge Credentials Configuration

Credentials used by the application to connect to external online judges have to be configured. To configure, create or edit file located at *config/prod.secret.exs*. Necessary fields are to be added accordingly.

```
config :virtual_judge, VirtualJudge.Mailer,
adapter: Bamboo.SendgridAdapter,
api_key: "YOUR SENDGID API KEY"
config :virtual_judge, codechef_username: "CODECHEF USERNAME",
codechef_password: "CODECHEF PASSWORD"
config :virtual_judge, codeforce_username: "CODEFORCE USERNAME
",
codeforce_password: " CODEFORCE PASSWORD "
config :virtual_judge, poj_username: "POJ USERNAME ",
poj_password: " POJ PASSWORD "
config :virtual_judge, timus_judge_id: "TIMUS USER ID"
config :virtual_judge, zoj_username: "ZOJ USERNAME",
zoj_password: "ZOJ PASSWORD"
config :virtual_judge, hust_username: "HUST USERNAME",
hust_password: "HUST PASSWORD"
config :virtual_judge, lydsy_username: "LYDSY USERNAME",
lydsy_password: "LYDSY PASSOWRD"
config :virtual_judge, fzu_username: "FZU USERNAME",
fzu_password: "FZU PASSWORD”
```

2.2.3.Dependency Installation and Configuration

2.2.3.1.Installing PostgreSQL

PostgreSQL is the database management system used by Virtual Judge system. To install, execute the following commands in the terminal.

```
> sudo apt-get update # update Ubuntu system
> sudo apt-get install postgresql postgresql-contrib #
install
postgres
```

2.2.3.2.Installing Redis

Redis is used to ensure job persistency of the Virtual Judge system even in an event of outage. To install, execute the following commands in the terminal.

```
$ cd /tmp # Do everything in temp folder so it gets deleted
later
$ sudo apt-get install build-essential tcl
$ curl -O http://download.redis.io/redis-stable.tar.gz
$ tar xzvf redis-stable.tar.gz
$ cd redis-stable
$ make
$ make test # see if everything is correctly extracted
$ sudo make install # this will install redis to the
application
```

2.2.3.3.Configuring Redis

Reais configuration file is to be edited for the use of the project. Execute the following commands in the terminal to open the configuration file in an editor.

```
$ sudo cp /tmp/redis-stable/redis.conf /etc/redis
$ sudo nano /etc/redis/redis.conf
```

The following changes are to be made to the configuration file.

1. Replace the line “supervised no” to “supervised systemd”
2. Replace the working directory to “dir /var/lib/redis”

In addition, the systemd file have to be configured. Execute the following command in the terminal to open the systemd configuration file.

```
$ sudo nano /etc/redis/redis.conf
```

The systemd file should look like the following.

```
[Unit]
Description=Redis In-Memory Data Store
After=network.target
[Service]
User=redis
Group=redis
ExecStart=/usr/local/bin/redis-server /etc/redis/redis.conf
ExecStop=/usr/local/bin/redis-cli shutdown
Restart=always
[Install]
WantedBy=multi-user.target
```

To ensure the proper installation of Redis, execute the following commands in the terminal.

```
$ redis-cli # this will run the redis CLI
127.0.0.1:6379 > ping # run the `ping` command to test
connectivity
```

3. Compilation and Deployment

To compile, build, and deploy the application, execute the following commands in the terminal.

```
# Initial setup

$ mix deps.get --only prod

$ MIX_ENV=prod mix compile

# Compile assets
$ brunch build --production
$ MIX_ENV=prod mix phoenix.digest

$ MIX_ENV=prod mix ecto.create
$ MIX_ENV=prod mix ecto.migrate

# Finally run the server
$ sudo PORT=80 MIX_ENV=prod mix phoenix.server
```

4. Maintenance

This section details the process of maintenance for the Virtual Judge system. Maintenance of the system is to be performed whenever there is a new update to the source code to be deployed to the production environment.

The following are the steps to be taken to maintain the server in the event of new update to be deployed.

1. Access hosting server
2. Pull changes from repository
3. Terminate existing instance
4. Compile new source code
5. Deploy new application

4.1.Accessing Hosting Server

As of revision 1.0, the Virtual Judge system is hosted on NTU SCSE virtual machine.

To access the hosting server, execute the following commands in the terminal.

```
$ ssh VMadmin@172.21.148.174
# enter the server's password
```

4.2.Pull Changes From Repository

As the source code is hosted on GitHub, new changes committed has to be pull to the hosting machine. To pull the new changes, execute the following commands in the terminal.

```
$ cd ~/Code/virtual_judge # the virtual judge is usually placed
in the code folder
$ git pull master origin # this will pull any new changes
```

4.3.Terminate Existing Instance

In the event that an instance of the application is running on the hosting machine, a termination of the existing instance is required. To terminate the existing instance, execute the following commands in the terminal.

```
$ sudo lsof -i :80 # this will tell you which PID to kill, for
example, the PID is 3242
$ kill 3242 # kill the process id 3242
```

4.4.Compile New Source Code

After the source code on the hosting machine is updated, a compilation procedure is required. To compile the new source code, execute the following command in the terminal.

```
$ MIX_ENV=prod mix compile
```

4.5.Deploy New Application

Upon compiling the new source code, the application is ready to be deployed. To deploy, execute the following command in the terminal.

```
$ sudo PORT=80 MIX_ENV=prod mix phoenix.server
```

5. Creation of Administrative User

This section details the steps to create an administrative user in the Virtual Judge system.

As of revision 1.0, the only way to create an administrative user is through the terminal. To create an administrative user, execute the following command in the terminal.

```
$ iex -S mix # this will open an elixir console

iex> VirtualJudge.Repo.insert!(%VirtualJudge.User{email: "[YOUR
EMAIL]", password_hash: Comeonin.Bcrypt.hashpwsalt("[YOUR
PASSWORD]"), bio: "admin of system", type: “admin”})
```

6. Further Help and Contact Information

For more information, you can contact us through the following personnel.

Title	Name	Email
System Owner	Kevin Anthony Jones	askajones@ntu.edu.sg
System Administrator	Ang Teng Huat	c150125@e.ntu.edu.sg