Smart Invigilator Allocation System

Installation Guide

National University of Singapore – ISS

Master of Technology

Intelligent Reasoning Systems – Group Project

Term: Jan'2021 to May'2021

SIAS-GRP Project Team Members

Narendernath Baskar	A0230120J
Yusuf Pranggonoh	A0229966J
Neoh Shi Kang	A0229965L
Tan Wee Han	A0125244N

Table of Contents

1. Objective

Objective of this document is to provide details on the pre-requisites and sequence of steps involved to install the “Smart Invigilator Allocation System” application and the dependent packages.

2. Environment Requirement

2.1 Hardware Requirement

To install and run SIAS software, a Linux environment with at least 4GB of RAM and 20GB of hard disk storage space is recommended.

2.2 Software Requirement

For best results, the operating system to use is Ubuntu-18.04 or Ubuntu-20.04, and the web browser to use is Mozilla Firefox (version 88.0).

3. Installation of Pre-Requisites

The SIAS project comprises of 3 components: Frontend, Backend, and Scheduler. All the commands will be run in terminal, unless otherwise stated.

Step-1: For Frontend, install Node Package Manager using the below terminal commands

```
sudo apt update
sudo apt install npm
```

Step-2: For Backend, install Anaconda, then create a conda environment.

a. Install “curl” if not available

```
sudo apt install curl
```

b. Download and install Anaconda for Linux

```
cd /tmp
curl -O https://repo.anaconda.com/archive/Anaconda3-2020.02-Linux-x86_64.sh
bash Anaconda3-2020.02-Linux-x86_64.sh
```

c. Use "Enter" key to review the license agreement and type "yes" at the bottom to agree the terms.

d. Press "Enter" key to confirm the installation location.

e. Press "yes" when prompted for "Do you wish the installer to initialize Anaconda3 by running conda init? [yes|no]"

f. Activate the installation by typing following command

```
source ~/.bashrc
```

g. Create clean "sias" conda environment (First time only)

```
conda create -n sias python=3.7
```

Step-3: For Scheduler, install JDK 1.8+

```
sudo apt install default-jdk
```

4. Download the SIAS Package

One of the below options can be followed to download and copy the application code into local drive.

4.1 Option-1: Download from GitHub using terminal command

This step explains how to download and copy the code from GitHub project repository.

Step-1: Extract the Package from GitHub and move into "/sias" directory using the below terminal commands

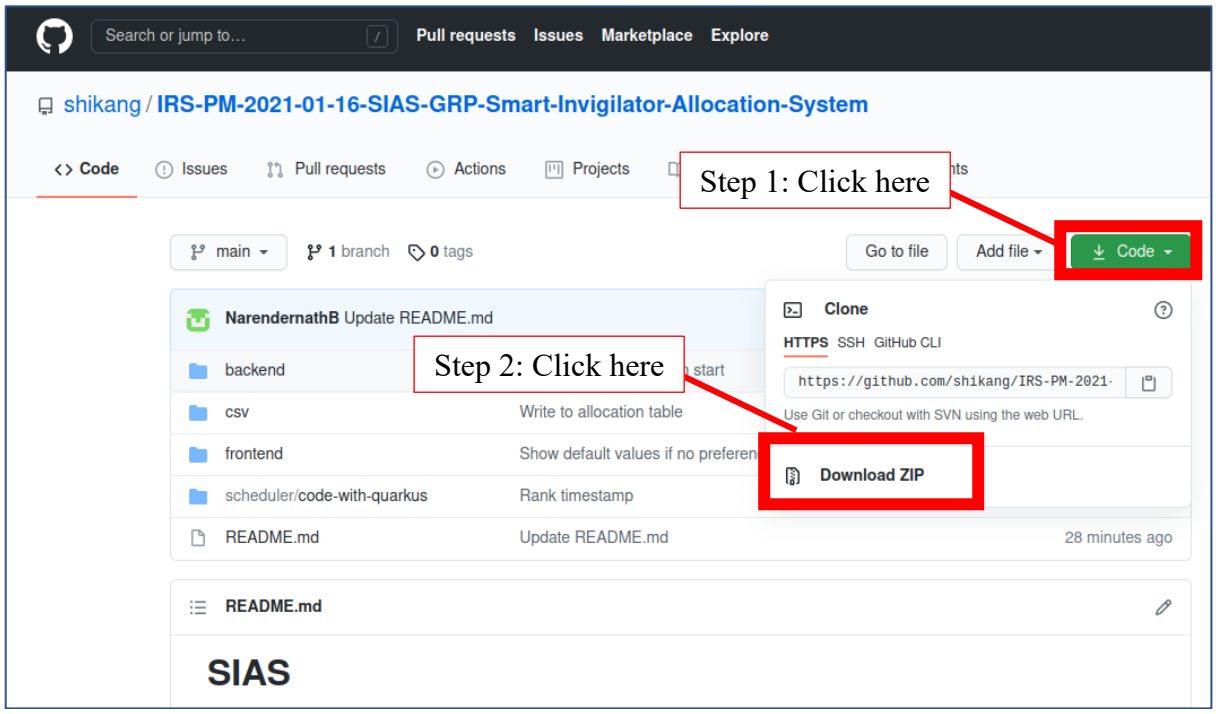
```
git clone https://github.com/shikang/IRS-PM-2021-01-16-SIAS-GRP-Smart-Invigilator-Allocation-System
sudo mv IRS-PM-2021-01-16-SIAS-GRP-Smart-Invigilator-Allocation-System /sias
cd /sias
```

4.2 Option-2: Download from GitHub using web browser

SIAS project is located on GitHub in the following link:

https://github.com/shikang/IRS-PM-2021-01-16-SIAS-GRP-Smart-Invigilator-Allocation-System

Step-1: Visit the link and download the zip package to local “Downloads” folder



Step-2: Create "/sias" folder using the below terminal command

```
sudo mkdir /sias
```

Step-3: Extract all the files and folders under "IRS-PM-2021-01-16-SIAS-GRP-Smart-Invigilator-Allocation-System" folder from the zip file "IRS-PM-2021-01-16-SIAS-GRP-Smart-Invigilator-Allocation-System.zip" that was downloaded in "Downloads" folder (Step-1) into "/sias" directory.

4.3 Option-2: Copy from submitted project package

This step explains how to copy the code from project package submitted in "LumiNUS" portal under "IRS-PM: Practice Module".

Step-1: Download "IRS-PM-2021-01-16-SIAS-GRP-Smart-Invigilator-Allocation-System.zip" from "LumiNUS" portal under "Files → IRS-PM: Practice Module" into "Downloads" folder.

Step-2: Create "sias" folder using the below terminal command

```
sudo mkdir /sias
```

Step-3: Extract all the files and folders under "IRS-PM-2021-01-16-SIAS-GRP-Smart-Invigilator-Allocation-System" folder from the zip file "IRS-PM-2021-01-16-SIAS-GRP-Smart-Invigilator-Allocation-System.zip" that was downloaded in "Downloads" folder (Step-1) into "/sias" directory.

5. Installation of SIAS Application

The SIAS project comprises of 3 components: Frontend, Backend, and Scheduler. All the commands will be run in terminal, unless otherwise stated.

Step-1: For Frontend, install dependencies

```
cd /sias/frontend
npm install
```

Step-2: For Backend, activate "sias" conda environment and install dependencies

```
cd /sias/backend
conda activate sias
pip install -r requirements.txt
```

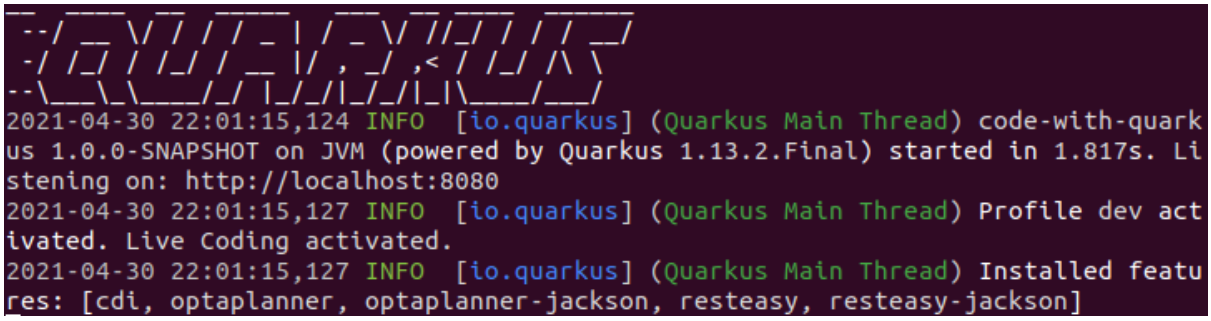
Note: In case if "flask-cors" is not installed from the above command, run the below command

```
pip install flask_cors
```

Step-3: Install and start the Scheduler

```
cd /sias/scheduler/code-with-quarkus
./mvnw compile quarkus:dev
```

The installation may take a few minutes. After installation, you should see the following in the terminal:



6. Start Services for SIAS Application

Services for SIAS application should be started in this sequence: Scheduler -> Backend -> Frontend

Step-1: Start the Scheduler
Open a new Terminal and execute the following commands

```
cd /sias/scheduler/code-with-quarkus
./mvnw compile quarkus:dev
```

After starting the Scheduler the text similar to below screenshot will be seen in the terminal.

```
__  ____  __  _____   ___  __ ____  ______
 --/ __ \/ / / / _ | / _ \/ //_/ / / / __/
 -/ /_/ / /_/ / __ |/ , _/ ,< / /_/ /\ \
--\___\_\____/_/ |_/_/|_/_/|_|\____/___/
2021-04-30 22:01:15,124 INFO  [io.quarkus] (Quarkus Main Thread) code-with-quark
us 1.0.0-SNAPSHOT on JVM (powered by Quarkus 1.13.2.Final) started in 1.817s. Li
stening on: http://localhost:8080
2021-04-30 22:01:15,127 INFO  [io.quarkus] (Quarkus Main Thread) Profile dev act
ivated. Live Coding activated.
2021-04-30 22:01:15,127 INFO  [io.quarkus] (Quarkus Main Thread) Installed featu
res: [cdi, optaplanner, optaplanner-jackson, resteasy, resteasy-jackson]
```

Step-2: Start the Backend
Open a new Terminal and execute the following commands

```
cd /sias/backend
conda activate sias
python app.py
```

After starting the backend application, the below text will be seen in the terminal.

```
 * Serving Flask app "app" (lazy loading)
 * Environment: production
   WARNING: This is a development server. Do not use it in a production deploym
ent.
   Use a production WSGI server instead.
 * Debug mode: off
 * Running on http://0.0.0.0:5000/ (Press CTRL+C to quit)
```

Step-3: Start the Frontend
Open a new Terminal and execute the following commands

```
cd /sias/frontend
npm start
```

After starting the frontend application, the below text will be seen in the terminal.

```
Compiled successfully!

You can now view my-app in the browser.

  Local:            http://localhost:3000
  On Your Network:  http://10.0.2.15:3000

Note that the development build is not optimized.
To create a production build, use npm run build.
```

Upon starting the Frontend, the default browser will open up the SIAS application. If not opened, you can open your browser and go to the link: http://localhost:3000

The below page will be displayed.