**NER Web tool Installation Guide**

V1.0

(for Users)

|  |  |
| --- | --- |
|  | **Element** |
| By Team 4 |
| CHONG WEI WEN REUBEN (2001893) CHEN XIN XIN (2002169)  JEVAN NG JIA QI (2001135)  LEONG KAH EN (2002841)  WHITNEY TAN WENHUI (2002738) | |

**Table of Contents**

[**1** **Installing Web Tool** 2](#_Toc110331664)

[**1.1** **Clone the Repository** 2](#_Toc110331665)

[**1.2** **Include ‘.env’ file** 2](#_Toc110331666)

[**1.3 Install Web App dependencies** 3](#_Toc110331667)

[**2** **Installing Flask-microservice** 4](#_Toc110331669)

[**2.1** **Python virtual environment** 4](#_Toc110331670)

# **Installing Web Tool**

## **1.1 Clone the Repository**

Clone the repository from Azure DevOps environment.

## **1.2 Include ‘.env’ file**

**TLDR:**

|  |  |
| --- | --- |
| cd app | Go to app dir |
| touch .env | Make .env file |
| nano .env | Edit and include your db string |

As the web tool includes the use of mongo atlas cloud database, you will have to include the connection string to your own mongo atlas database, alternatively you can change the service provider to your preferred choice as long as it is document-based database.

The ‘.env’ file has to be in /app directory. Below shows a screenshot of the content of the env file.

Text

Description automatically generated

## **1.3 Install Web App dependencies**

**TLDR:**

|  |  |
| --- | --- |
| cd app | Go to app dir |
| npm install | Install dep backend |
| cd client | Go to client dir **from /app** |
| npm install | Install dep for frontend |
| cd .. | Go to app dir **from /app/client** |
| npm run dev | Run web tool |

Trouble shooting:

If there is an error running the “npm run dev” script, i.e. image shown below ensure that in /app/client/public, there are 3 folders called “archive”, “downloads” and “uploads”. The error happens because when the app runs, reactjs will query the server first to get the previously uploaded and unprocessed files from the server, so that in the frontend application, the user can see which files he/she has uploaded already, however if the folders does not exists, when server.js checks it’s file system to find the folders, it will face an error.

Text

Description automatically generated

Long version with images:

There are 2 directory where you will need to execute `npm install`. In the /app directory, run npm install, this will install dependencies for the backend web server. In /app/client, run npm install, this will install dependencies for the frontend react application. Example shown below.

Installing frontend dependencies



Installing backend dependencies



Running the web tool

Text

Description automatically generated

## Text Description automatically generated

Make sure there is “Server running on port: 5001”, “Successfully connected to MongoDB” and “Starting the development server…” messages. You can now visit localhost:3000 to run the web tool. (It is port 3000 as react frontend application is on 3000).

# **Installing Flask-microservice**

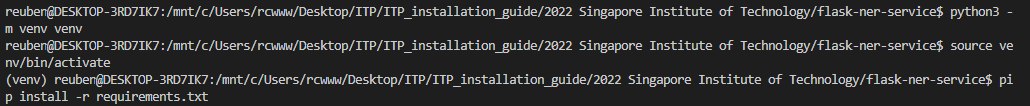
## **2.1 Python virtual environment**

**TLDR:**

|  |  |
| --- | --- |
| cd flask-ner-service | Go to flask ner dir |
| source {$venvFolderName}/bin/activate | Activate venv |
| pip install -r requirements.txt | Install dependencies |
| Python ner.py | Run microservice |

To contain the dependencies for the microservice, we will be using a virtual environment. Create a virtual environment with `python3 -m venv ${nameOfVenv}`. Activate the virtual environment, essentially running it. `source {$nameOfVenv}/bin/activate`. Install dependencies listed in “requirements.txt”. Steps shown as images below.

Create virtual environment, activate it and install dep



Run web microservice

Text

Description automatically generated