INSTALL.md

# Installation Guide

Please read this guide **thoroughly** for the application to work as intended. If you have any questions regarding the installation process, please contact us.

#### Table of Contents

## Prerequisites

For this application to work, you will need to have Python and the following technologies installed on your system. Please make sure to download the appropriate version for your system:

> *Link to Python's download page*: Python 3.8.5
> *Link to Anaconda's download page*: Anaconda
> *Link to Git's download page*: Git

To check that you have successfully installed the said technologies, run the following command on the command line

For Python:

```
python --version
```

For Anaconda:

```
conda --version
```

For git:

```
git --version
```

## Virtual Environment

**Step 1** Create a virtual environment.

```
conda create -n [environment name] python=3.8.5

conda activate [environment name]
```

## Cloning the Repository

**Step 2** Clone the repository. Make sure you're in the directory that you intended to clone the project to.

```
git clone https://stgit.dcs.gla.ac.uk/tp3-2020-CS28/cs28-main.git
```

## Setup Django

**Step 3.1** Use the package manager pip (installed by default in Python 3.4 and up) to install Django 3.1.3 and other dependencies through the requirements.txt file:

```
pip install -r requirements.txt
```

**Step 3.2** Set up the database.

**Step 3.2.1** If you are in the root directory, navigate to the directory with manage.py

```
cd cs28_project
```

**Step 3.2.2** Run `makemigrations` and `migrate`.

```
python manage.py makemigrations

python manage.py migrate
```

If you have made any changes to the models after a database already exists, you will have to run `migrate` with `--run-syncdb`

```
python manage.py migrate --run-syncdb
```

## Setup Application

**Step 4.1** Create a superuser account

```
python manage.py createsuperuser
```

*You will be prompted to enter your username, email and password.*

**Step 4.2** Run the application

### For deployment purposes

**Step 4.2.1** Collect static files for deployment

```
python manage.py collectstatic
```

*A file named `assets` will be created in `cs28_project` containing all static files used in the project*

**Step 4.2.2** Run server

```
python manage.py runserver
```

### For local usage

**Step 4.2.1** Run server with the `--insecure` tag

```
python manage.py runserver --insecure
```

> WARNING: *This is only intended for local development purposes as it is **grossly inefficient and probably insecure** as stated by Django's documentation*

## Deployment

### SMTP

Password reset email is currently printed on the terminal. To send as an actual email, a SMTP service is required. For more information on setting up SMTP, please read Django's documentation:

> https://docs.djangoproject.com/en/3.1/topics/email/
>
> *The current implementation uses Sendinblue. For more information, visit https://www.sendinblue.com/*

### Live server

For more information on deploying on a live server, please read the following guide:

> https://developer.mozilla.org/en-US/docs/Learn/Server-side/Django/Deployment

### PythonAnywhere

The deployment process to Python Anywhere should be largely similar to the local deployment process. For more information on deploying to Python Anywhere, please watch the following guide:

> https://www.youtube.com/watch?v=Y4c4ickks2A

and Python Anywhere's documentation:

> https://help.pythonanywhere.com/pages/DjangoTutorial/

### Heroku

The application is ready for deployment on Heroku. However, there may be some changes to settings as with the previous option (live server). For more information on deploying to Heroku, please read Heroku's documentation:

> https://devcenter.heroku.com/articles/deploying-python

### PythonAnywhere vs Heroku

The reason Heroku was used for Continuous Deployment during development process was due to GitLab not being able to clone directly into the bash shell in PythonAnywhere for free plan users. Both are viable options for deployment depending on usage purposes (such as specific databases, pricing or free plan features). For more information on which to choose, please read the following blog:

> https://blog.pythonanywhere.com/65/#:~:text=Conclusions%3F-,%C2%B6,more%20like%20a%20development%20server.