**Theft Alert System for Supermarkets Deployment Manual­­**

**Secure View Theft Alert System**

# Introduction

## Purpose of this Manual

This manual provides step-by-step guidance for deploying the CCTV Security Platform on a production server. It is designed for system administrators responsible for ensuring the platform is securely and efficiently set up.

## System Overview

The CCTV Security Platform enables real-time video streaming, storage, and monitoring of surveillance footage. Key features include:

Scalable and robust architecture to handle multiple video sources. User management through a secure web interface.

Integrated logging and monitoring for system health and security. SSL/TLS encryption for secure communication.

# Prerequisites

## Hardware Requirements

Ensure the following hardware specifications for optimal performance:

* **Server/Host Device**: PythonAnywhere provides the hosting environment, so no local server hardware is needed.
* **Client Devices**: Any device capable of running a modern web browser (e.g., Chrome, Firefox).
* **Storage**: At least 500MB of disk space for the application and logs (included in PythonAnywhere plans).
* **Memory**: PythonAnywhere plans include sufficient RAM for most Django applications. For heavy traffic, ensure your plan supports additional worker processes.

A screenshot of a computer

Description automatically generated

## Software Requirements

PythonAnywhere already includes most necessary software. Con rm the following versions:

* **Python**: Python 3.8 or later (PythonAnywhere supports multiple versions).
* **Database**: MySQL (if required; PythonAnywhere provides hosted MySQL).
* **Django**: Ensure compatibility with the version specified in requirements.txt.

To install additional dependencies:

pip install -r requirements.txt

## Network Requirements

Ensure the following network conditions:

* **Internet Access**: Stable connection for accessing the PythonAnywhere dashboard and transferring les.
* **Ports**:
* PythonAnywhere handles standard web ports (e.g., HTTP: 80, HTTPS: 443).
* No additional port configuration is required as PythonAnywhere handles web traffic routing.

# Preparing the Environment

## Setting Up the Server

PythonAnywhere provides a hosted environment, so there's no need to configure a physical or virtual server. Follow these steps to set up your PythonAnywhere account:

## Create an Account

* + - Visit [PythonAnywhere](https://www.pythonanywhere.com/) and sign up for an account.
    - Choose a plan that your project requires.
    - Free plans support basic setups, while paid plans allow for more traffic and resource use.

## Access the Dashboard

* Log in and navigate to the dashboard, where you can manage your applications, and databases.

## Installing Required Software

PythonAnywhere provides pre-installed software for Python and Django development. However, you'll need to ensure your application's dependencies are installed:

## Upload Your Project

 Use PythonAnywhere’s le manager or a secure le transfer protocol (SFTP) client like FileZilla to upload your project les.

## Set Up a Virtual Environment

 Create a virtual environment to isolate your application dependencies:

mkvirtualenv --python=/usr/bin/python3.x venv

Replace 3.x with the Python version your application requires.

## Install Dependencies

* Activate the virtual environment:

workon venv

* Install dependencies:

pip install -r requirements.txt

## Configuring the Database

**Using SQLite**

If your project uses SQLite (default for Django), no additional setup is needed. The database will be included in the uploaded project.

## Using MySQL

If your project uses MySQL, configure a database on PythonAnywhere:

Go to the **Databases** tab on the PythonAnywhere dashboard. Create a new MySQL database and note the credentials.

A screenshot of a computer

Description automatically generated

Update your settings.py with the database connection details:

DATABASES = {

'default': {

'ENGINE': 'django.db.backends.mysql',

'NAME': 'cctvsecurity $cctvdb',

'USER': 'admin11’,

'PASSWORD': '\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*',

'HOST': 'your\_username.mysql.pythonanywhere-services.com', 'PORT': '3306',

}

}

# Deploying the Application

## Obtaining the Source Code

**Upload the Project Files**

 Use PythonAnywhere’s le manager or an SFTP client to upload your project les to a directory, such as ~/mysite.

## Cloning from a Repository

 If your project is hosted on GitHub or another Git hosting service, you can clone it directly:

git clone https://github.com/KipkorirVictor/Supermaket-Survaillance-System.git

## Configure Django Settings

**Modify settings.py**

 Update the ALLOWED\_HOSTS in settings.py:

ALLOWED\_HOSTS = [‘cctvsecurity.pythonanywhere.com’, ‘127.0.0.1’]

Set DEBUG to False for a production environment:

DEBUG = False

## Update Database Settings

 Configure the database settings if you're using MySQL (refer to the database section above).

## Set Up Environment Variables

 Create a. env le to store sensitive information, such as SECRET\_KEY.

* Update settings.py to read from this le (using a library like python-decouple or dotenv).

## Setting Up Static and Media Files

**Collect Static Files**

 Run the following command to collect all static files into the STATIC\_ROOT:

python manage.py collectstatic

 Confirm that static files are in the correct directory.

## Media Files

 Ensure that media files are uploaded and accessible. Specify a directory for media files in your settings.py:

MEDIA\_URL = '/media/'

MEDIA\_ROOT = '/home/cctvsecurity/mysite/media'

## Configuring the Web Server

PythonAnywhere uses a WSGI server instead of consulting Nginx or Apache directly.

## Set Up the Web App

Go to the **Web** tab on the PythonAnywhere dashboard.

Click **Add a new web app** and follow the prompts:

 Choose **Manual configuration**.

 Select the Python version that matches your project.

## Configure Static and Media Files

 Under the **Static les** section in the Web tab, add entries like:

 URL /static/ → Directory /home/cctvsecurity/mysite/static

URL /media/ → Directory /home/cctvsecurity/mysite/media



## Setting Up WSGI

**Edit the WSGI File**

* Open the WSGI configuration in the Web tab.
* Update it to point to your Django project:

**import** os

**import** sys

*# Adjust paths*

path = '/home/cctvsecurity/mysite'

**if** path **not in** sys.path:

sys.path.append(path)

os.environ['DJANGO\_SETTINGS\_MODULE'] = 'cctv\_project.settings'

**from** django.core.wsgi **import** get\_wsgi\_application

application = get\_wsgi\_application()

## Reload the Web App

Click **Reload** in the Web tab to apply the changes.

A screenshot of a web page

Description automatically generated

# Database Migration and Initial Setup

## Running Database Migrations

**Activate the Virtual Environment**

 Ensure you are in your project directory and activate your virtual environment:

## Run Migrations

 Apply migrations to set up the database:

This command creates the required database tables for Django’s built-in apps and any custom apps you’ve added.

python manage.py migrate

## Creating a Superuser

**Run the create superuser Command**

* Create an admin user to access the Django Admin interface:

python manage.py createsuperuser

## Provide Required Information

* Enter the following details when prompted:
* Username
* Email address
* Password

## Verify the Superuser

* Ensure the user is created successfully. You will use this account to log in to the admin panel.

## Initial Data Setup

**Load Initial Data (if applicable)**

* If your project requires pre-defined data, use Django’s load data command to import textures.

python manage.py load data initial\_data.json

## Verify the Data

* Access the Django Admin interface by visiting:

Log in with the superuser credentials and confirm that the necessary data is loaded.

https://cctvsecurity.pythonanywhere.com/admin