**📘 Asset Capture Application – Deployment Documentation**

**1. Server Overview**

* **OS:** Ubuntu 22.04 LTS (server)
* **Application:** Flask (asset\_capture\_app\_dev)
* **Application server:** Gunicorn
* **Web server / reverse proxy:** Nginx
* **Database:** SQLite (local file storage)
* **Static files:** Served directly by Nginx from /home/developer/asset\_capture\_app\_dev/static/
* **Domain:** appprod.assetcap.facilities.ubc.ca
* **IP Address:** 142.103.68.1
* **SSL/TLS:** Let’s Encrypt via Certbot
* **Systemd service name:** assetcap

**2. Directory Structure**

/home/developer/

└── asset\_capture\_app\_dev/ # Application source code

├── venv/ # Python virtual environment

├── app.py # Flask entry point

├── static/ # Static assets (CSS, JS, images)

├── templates/ # HTML templates

└── data/ # SQLite database

**3. Initial Setup**

**3.1 Install dependencies**

sudo apt update

sudo apt install -y git python3-venv python3-pip nginx certbot python3-certbot-nginx acl

**3.2 Clone the repository**

cd /home/developer

git clone https://github.com/gandradepa/asset\_capture\_app\_dev.git

cd asset\_capture\_app\_dev

**3.3 Create virtual environment and install Python packages**

python3 -m venv venv

source venv/bin/activate

pip install --upgrade pip

pip install -r requirements.txt

deactivate

**3.4 Database initialization**

mkdir -p data

python3 - <<'PY'

import sqlite3, os

db = "/home/developer/asset\_capture\_app\_dev/data/QR\_codes.db"

os.makedirs(os.path.dirname(db), exist\_ok=True)

conn = sqlite3.connect(db)

conn.execute("CREATE TABLE IF NOT EXISTS QR\_codes (QR\_code\_ID TEXT PRIMARY KEY);")

conn.commit()

conn.close()

print("SQLite DB initialized at", db)

PY

**4. Gunicorn via Systemd**

Create service file:

sudo tee /etc/systemd/system/assetcap.service > /dev/null <<'EOF'

[Unit]

Description=Gunicorn - Asset Capture App

After=network.target

[Service]

User=developer

Group=developer

WorkingDirectory=/home/developer/asset\_capture\_app\_dev

Environment="PATH=/home/developer/asset\_capture\_app\_dev/venv/bin"

ExecStart=/home/developer/asset\_capture\_app\_dev/venv/bin/gunicorn --bind 127.0.0.1:8000 --workers 3 app:app

ExecReload=/bin/kill -HUP $MAINPID

Restart=on-failure

[Install]

WantedBy=multi-user.target

EOF

Enable and start:

sudo systemctl daemon-reload

sudo systemctl enable assetcap

sudo systemctl start assetcap

sudo systemctl status assetcap --no-pager

**5. Nginx Configuration**

sudo tee /etc/nginx/sites-available/assetcap-app > /dev/null <<'EOF'

server {

listen 80;

server\_name appprod.assetcap.facilities.ubc.ca;

client\_max\_body\_size 50M;

location / {

proxy\_pass http://127.0.0.1:8000;

proxy\_set\_header Host $host;

proxy\_set\_header X-Real-IP $remote\_addr;

proxy\_set\_header X-Forwarded-For $proxy\_add\_x\_forwarded\_for;

proxy\_set\_header X-Forwarded-Proto $scheme;

}

location /static/ {

alias /home/developer/asset\_capture\_app\_dev/static/;

access\_log off;

expires 30d;

}

}

server {

listen 443 ssl;

server\_name appprod.assetcap.facilities.ubc.ca;

ssl\_certificate /etc/letsencrypt/live/appprod.assetcap.facilities.ubc.ca/fullchain.pem;

ssl\_certificate\_key /etc/letsencrypt/live/appprod.assetcap.facilities.ubc.ca/privkey.pem;

include /etc/letsencrypt/options-ssl-nginx.conf;

ssl\_dhparam /etc/letsencrypt/ssl-dhparams.pem;

client\_max\_body\_size 50M;

location / {

proxy\_pass http://127.0.0.1:8000;

proxy\_set\_header Host $host;

proxy\_set\_header X-Real-IP $remote\_addr;

proxy\_set\_header X-Forwarded-For $proxy\_add\_x\_forwarded\_for;

proxy\_set\_header X-Forwarded-Proto $scheme;

}

location /static/ {

alias /home/developer/asset\_capture\_app\_dev/static/;

access\_log off;

expires 30d;

}

}

EOF

sudo ln -sf /etc/nginx/sites-available/assetcap-app /etc/nginx/sites-enabled/

sudo nginx -t && sudo systemctl reload nginx

**6. SSL Certificate with Certbot**

sudo certbot --nginx -d appprod.assetcap.facilities.ubc.ca

Certificates are stored under:

/etc/letsencrypt/live/appprod.assetcap.facilities.ubc.ca/

Automatic renewal is handled by a systemd timer (certbot.timer).

**7. Permissions for Static Files**

Allow www-data (Nginx) to traverse /home and read static files:

sudo setfacl -m u:www-data:rx /home

sudo setfacl -m u:www-data:rx /home/developer

sudo setfacl -m u:www-data:rx /home/developer/asset\_capture\_app\_dev

sudo setfacl -R -m u:www-data:rX /home/developer/asset\_capture\_app\_dev/static

sudo setfacl -dR -m u:www-data:rX /home/developer/asset\_capture\_app\_dev/static

**8. Deployment Workflow (branch Dev\_environment)**

cd /home/developer/asset\_capture\_app\_dev

git fetch --all

git reset --hard origin/Dev\_environment

source venv/bin/activate

pip install -r requirements.txt

deactivate

sudo systemctl restart assetcap

Check logs:

sudo systemctl status assetcap --no-pager

sudo journalctl -u assetcap -f

**9. Verification**

* Gunicorn directly:

curl -I http://127.0.0.1:8000

* Via Nginx HTTP:

curl -I http://appprod.assetcap.facilities.ubc.ca

* Via Nginx HTTPS:

curl -I https://appprod.assetcap.facilities.ubc.ca

* Static asset (logo):

curl -I https://appprod.assetcap.facilities.ubc.ca/static/img/ubc\_facilities\_logo.png

Expected: HTTP/1.1 200 OK.

**10. Maintenance**

* **Restart app:**
* sudo systemctl restart assetcap
* **Check app logs:**
* sudo journalctl -u assetcap -f
* **Reload Nginx after config changes:**
* sudo nginx -t && sudo systemctl reload nginx
* **Renew SSL manually (if needed):**

sudo certbot renew --dry-run