1. Betriebssystem auf SD-Karte flashen

2. Hochfahren, Locale, Netzwerk und SSH einrichten

3. Git installieren

```
sudo apt update
sudo apt upgrade
sudo apt install git -y
git --version
```

4. Git einrichten (ssh)

```
git config --global user.name [NAME]
git config --global user.email [EMAIL]
ssh-keygen -t ed25519 -C [EMAIL]
cat ~/.ssh/id_ed25519.pub
```

Schlüssel kopieren und bei Codeberg unter Profilbild → Settings → SSH- / GPG-Keys → Add Key

5. Projektverzeichnis erstellen:

```
mkdir sese_project
```

6. Klonen Git-Repository:

```
git clone ssh://git@codeberg.org/faeunin/serious-seeds.git
```

7. Virtuelle Umgebung

```
cd sese_project/serious-seeds
python3 -m venv env
source env/bin/activate
```

8. Requirements installieren

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

9. .env-file für Umgebungsvariablen anlegen

```
nano .env
```

mit folgendem Inhalt:

```
DJANGO_ENV=production

SECRET_KEY=[KEY]

DEBUG=False

POSTGRES_DB=[DB_NAME]
POSTGRES_USER=[DB_USER]
POSTGRES_PASSWORD=[DB_PW]
POSTGRES_HOST=localhost
POSTGRES_PORT=5432
```

10. Postgres installieren

```
sudo apt install postgresql postgresql-contrib libpq-dev -y
```

11. Datenbank erstellen

```
sudo -i -u postgres psql
```

```
CREATE DATABASE [DB_NAME];

CREATE USER [DB_USER] WITH PASSWORD '[DB_PW]';

ALTER ROLE [DB_USER] SET client_encoding TO 'utf8';
ALTER ROLE [DB_USER] SET default_transaction_isolation TO 'read committed';
ALTER ROLE [DB_USER] SET timezone TO 'UTC';

GRANT ALL PRIVILEGES ON DATABASE [DB_NAME] TO [DB_USER];

ALTER DATABASE [DB_NAME] OWNER TO [DB_USER];

\q
```

12. DB-Backup auf Raspberry kopieren

```
scp /pfad/zur/datei/auf/dem/laptop/dump.sql
benutzername@IP_DES_PI:/pfad/auf/dem/pi/
```

13. Dump laden

```
pg_restore -h localhost -p 5432 -U [DB_USER] -d [DB_NAME] --no-owner --no-
```

```
acl /path/to/dump.sql
```

14. Hinzufügen IP zu ALLOWED_HOSTS in settings.py

15. Static Files

```
python manage.py collectstatic
```

16. Nginx

```
sudo apt install nginx -y

sudo nano /etc/nginx/sites-available/seriousseeds
```

with following content:

```
server {
    listen 80;
    server_name <Raspberry_Pi_IP>;

    location / {
        proxy_pass http://unix:/run/gunicorn/gunicorn.sock:/;
        proxy_set_header Host $host;
        proxy_set_header X-Real-IP $remote_addr;
        proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;
    }

    location /static/ {
        alias </path/to/staticfiles/>;
    }
}
```

Next, enable the configuration:

```
sudo ln -s /etc/nginx/sites-available/seriousseeds /etc/nginx/sites-enabled/
sudo nginx -t
sudo systemctl restart nginx
```

Check:

```
sudo nginx -T | sed -n '1,200p' | grep -n 'proxy_pass\|upstream'
```

17. Gunicorn

```
sudo nano /etc/systemd/system/gunicorn.service
```

content:

```
[Unit]

Description=gunicorn daemon for Django
After=network.target

[Service]
User=[SYSTEMUSER]
Group=www-data
RuntimeDirectory=gunicorn
WorkingDirectory=[...]/sese_project/serious-seeds/app
ExecStart=[...]/sese_project/serious-seeds/env/bin/gunicorn --workers 5 --
threads 2 --timeout 60 --max-requests 1000 --max-requests-jitter 100 --bind
unix:/run/gunicorn/gunicorn.sock seriousseeds.wsgi:application

[Install]
WantedBy=multi-user.target
```

Start and enable the service:

```
sudo systemctl start gunicorn
sudo systemctl enable gunicorn
```