

Developer's Manual for HiveSociety System

This manual provides detailed instructions on installing, configuring, running, testing, and deploying the application. Follow the steps below to set up both local and production environments.

Table of Contents

Developer's Manual for Django Application.....	1
Table of Contents	Error! Bookmark not defined.
1. Required Software	1
2. Installation	1
2.1 Cloning the Repository	1
2.2 Setting Up the Backend	1
3. Running and Testing the Application.....	3
Running the Backend Locally	3
Collecting Static Files (for frontend assets)	3
Running Tests	3
4. Deployment.....	3
4.1 Vercel Deployment.....	3
4.2 Supabase Database Sync (Production Only)	3
4.3 Environment Variables on Vercel.....	3
5. Additional Notes and Best Practices.....	4

1. Required Software

Ensure that the following software and tools are installed before proceeding:

- **Python 3.10+:** [Installation Guide](#)
- **pip** (comes with Python)
- **virtualenv** (for creating isolated development environments)
- **PostgreSQL** (for production; SQLite is used in development)
- **Git:** [Installation Guide](#)
- **Vercel CLI:** [Installation Guide](#)
- **Supabase CLI:** [Installation Guide](#)

2. Installation

2.1 Cloning the Repository

Clone the project from GitHub and navigate to the project directory:

```
$ git clone https://github.com/dinasaurs_project.git
$ cd dinasaurs_project
```

2.2 Setting Up the Backend

Create and Activate a Virtual Environment

On Linux/Mac:

```
$ python3 -m venv venv
$ source venv/bin/activate
```

On Windows:

```
$ python -m venv venv
$ venv\Scripts\activate
```

Install Dependencies

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

Database and Storage Setup

Development Environment: The default SQLite database is used, requiring no additional setup.

Production Environment: Supabase PostgreSQL is used, but this setup is not necessary for local development.

AWS S3 is used for media file storage in both environments.

1. **Configure the .env File:**

Create a .env file in the root of your project with the following content:

```
# Development database
DB_ENGINE=django.db.backends.sqlite3
DB_NAME=db.sqlite3

# Production database (used only on Vercel)
# Supabase credentials
DB_NAME=postgres
DB_USER=postgres.kjzcbbisaprhgzxhuqot
DB_PASSWORD=8n6EmBEsVlMRD6R1
DB_HOST=aws-0-eu-west-2.pooler.supabase.com
DB_PORT=6543

# AWS S3 Configuration (for media storage)
AWS_ACCESS_KEY_ID=AKIASBP32A24YRTVXNEA
AWS_SECRET_ACCESS_KEY= zulYm84/J0LPkpDqr9uy9NYXh6oWmsS+ixRytdhN
AWS_STORAGE_BUCKET_NAME=hive-society
AWS_S3_REGION_NAME=<your-aws-region>

# Django Secret Key
SECRET_KEY=django-insecure048$05y)ar037ha8w&3!(6%@264f+(y4f@zo*v%#bbya(kj5a0
```

2. **Run Migrations:**

Apply database migrations:

```
$ python manage.py migrate
```

3. **Seed the Database**

```
$ python manage.py seed
```

3. Running and Testing the Application

Running the Backend Locally

```
$ python manage.py runserver
```

Collecting Static Files (for frontend assets)

```
$ python manage.py collectstatic
```

Running Tests

```
$ python manage.py test
```

4. Deployment

4.1 Vercel Deployment

1. **Install Vercel CLI:**

```
$ npm install -g vercel
```

2. **Login to Vercel:**

```
$ vercel login
```

3. **Deploy the Application:**

```
$ vercel --prod
```

4.2 Supabase Database Sync (Production Only)

To sync local database changes with the production database:

```
$ supabase db push
```

4.3 Environment Variables on Vercel

Configure the following environment variables on Vercel for production:

- DB_NAME
- DB_USER
- DB_PASSWORD
- DB_HOST
- DB_PORT
- AWS_ACCESS_KEY_ID
- AWS_SECRET_ACCESS_KEY
- AWS_STORAGE_BUCKET_NAME
- AWS_S3_REGION_NAME

You can set these variables using the Vercel CLI:

```
$ vercel env add DB_NAME postgres
$ vercel env add DB_USER postgres.kjzcbbisaprhzgqxhuqot
$ vercel env add DB_PASSWORD 8n6EmBEsVlMRD6Rl
$ vercel env add DB_HOST aws-0-eu-west-2.pooler.supabase.com
$ vercel env add DB_PORT 6543
$ vercel env add AWS_ACCESS_KEY_ID AKIASBP32A24YRTVXNEA
$ vercel env add AWS_SECRET_ACCESS_KEY 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
$ vercel env add AWS_STORAGE_BUCKET_NAME hive-society
$ vercel env add AWS_S3_REGION_NAME eu-west-2
```

5. Additional Notes and Best Practices

- **Regularly Backup Data:** For production environments, set up regular backups of the PostgreSQL database
- **Keep Dependencies Updated:** Periodically run `pip list --outdated` and update packages as needed.
- **Static Files:** Always run `collectstatic` after making changes to static files (CSS, JS, images) to ensure they are properly collected.
- **Environment Variables:** Ensure that the `.env` file contains all necessary environment variables before running the application including the `SECRET_KEY`.