

Deployment Document

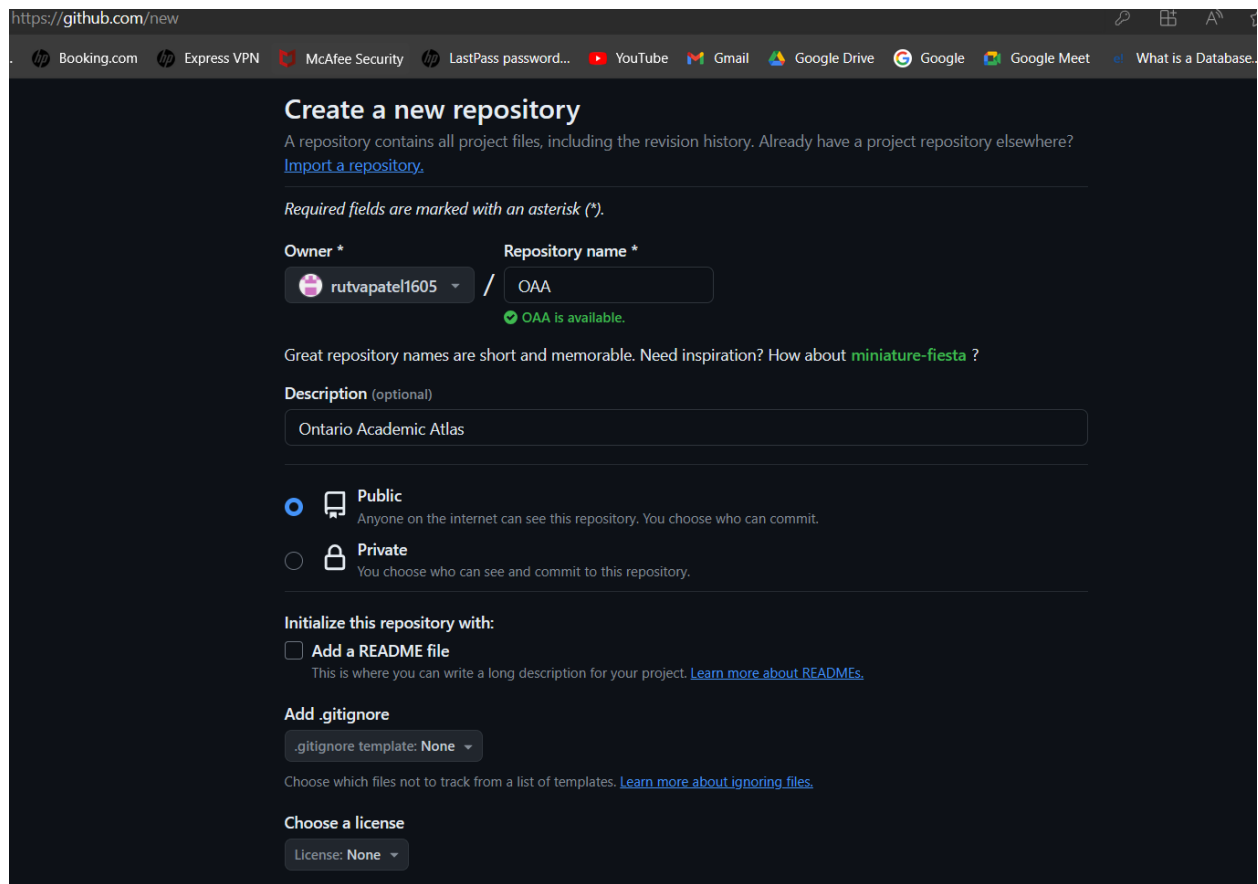
Ontario Academic Atlas

Introduction:

The deployment of the Ontario Academic Atlas project involves several steps to ensure a smooth transition from development to production. This document provides detailed instructions on how to push the Django project to a Git repository, set up a remote repository, and deploy the project on a server. Additionally, it includes information on how to download the project code from the Git repository and run it on a local machine.

Step 1: Pushing the Code to Git Repository:

We created a repository in GitHub by logging into our GitHub account and navigating to the main page. From there, we clicked on the "+" icon in the top right corner and selected "New repository". We then filled in the repository name, description, and other settings as needed, and optionally added a README file, a .gitignore file, and a license. Finally, we clicked on the "Create repository" button to create the repository on GitHub.



https://github.com/new

Create a new repository

A repository contains all project files, including the revision history. Already have a project repository elsewhere? [Import a repository.](#)

Required fields are marked with an asterisk (*).

Owner * rutvapatel1605 / Repository name * OAA

Great repository names are short and memorable. Need inspiration? How about [miniature-fiesta](#) ?

Description (optional)
Ontario Academic Atlas

☒ Public
Anyone on the internet can see this repository. You choose who can commit.

☐ Private
You choose who can see and commit to this repository.

Initialize this repository with:

☒ Add a README file
This is where you can write a long description for your project. [Learn more about READMEs.](#)

Add .gitignore
.gitignore template: None

Choose which files not to track from a list of templates. [Learn more about ignoring files.](#)

Choose a license
License: None

We initialize the Git repository by navigating to the root directory of the Django project using the command line.

We initialize a Git repository with the command: `git init`.

```
C:\Windows\System32\cmd.e  X  +  v
Microsoft Windows [Version 10.0.22631.3296]
(c) Microsoft Corporation. All rights reserved.

C:\BDAT-Sem2\Data Analytics Project\OAA>git init
Initialized empty Git repository in C:/BDAT-Sem2/Data Analytics Project/OAA/.git/

C:\BDAT-Sem2\Data Analytics Project\OAA>
```

We add all files to the Git repository by executing: `git add ...`

```
C:\Windows\System32\cmd.e  X  +  v
Microsoft Windows [Version 10.0.22631.3296]
(c) Microsoft Corporation. All rights reserved.

C:\BDAT-Sem2\Data Analytics Project\OAA>git init
Initialized empty Git repository in C:/BDAT-Sem2/Data Analytics Project/OAA/.git/

C:\BDAT-Sem2\Data Analytics Project\OAA>git add README.md
fatal: pathspec 'README.md' did not match any files

C:\BDAT-Sem2\Data Analytics Project\OAA>echo "# OAA" >> README.md

C:\BDAT-Sem2\Data Analytics Project\OAA>git add README.md

C:\BDAT-Sem2\Data Analytics Project\OAA>
```

After making the necessary changes, we commit them with a meaningful message: `git commit -m "Initial commit"`.

```
C:\Windows\System32\cmd.e  X  +  v
Microsoft Windows [Version 10.0.22631.3296]
(c) Microsoft Corporation. All rights reserved.

C:\BDAT-Sem2\Data Analytics Project\OAA>git init
Initialized empty Git repository in C:/BDAT-Sem2/Data Analytics Project/OAA/.git/

C:\BDAT-Sem2\Data Analytics Project\OAA>git add README.md
fatal: pathspec 'README.md' did not match any files

C:\BDAT-Sem2\Data Analytics Project\OAA>echo "# OAA" >> README.md

C:\BDAT-Sem2\Data Analytics Project\OAA>git add README.md

C:\BDAT-Sem2\Data Analytics Project\OAA>git commit -m "Adding OAA Code"
[master (root-commit) 4bfa04b] Adding OAA Code
 1 file changed, 1 insertion(+)
 create mode 100644 README.md

C:\BDAT-Sem2\Data Analytics Project\OAA>
```

Next, we create a new repository on our preferred Git hosting service, such as GitHub.

```
C:\Windows\System32\cmd.e  X  +  v
Microsoft Windows [Version 10.0.22631.3296]
(c) Microsoft Corporation. All rights reserved.

C:\BDAT-Sem2\Data Analytics Project\OAA>git init
Initialized empty Git repository in C:/BDAT-Sem2/Data Analytics Project/OAA/.git/

C:\BDAT-Sem2\Data Analytics Project\OAA>git add README.md
fatal: pathspec 'README.md' did not match any files

C:\BDAT-Sem2\Data Analytics Project\OAA>echo "# OAA" >> README.md

C:\BDAT-Sem2\Data Analytics Project\OAA>git add README.md

C:\BDAT-Sem2\Data Analytics Project\OAA>git commit -m "Adding OAA Code"
[master (root-commit) 4bfa04b] Adding OAA Code
1 file changed, 1 insertion(+)
create mode 100644 README.md

C:\BDAT-Sem2\Data Analytics Project\OAA>git branch -M main

C:\BDAT-Sem2\Data Analytics Project\OAA>
```

Following the hosting service's instructions, we add the remote repository URL to our local repository.

```
C:\Windows\System32\cmd.e  X  +  v
Microsoft Windows [Version 10.0.22631.3296]
(c) Microsoft Corporation. All rights reserved.

C:\BDAT-Sem2\Data Analytics Project\OAA>git init
Initialized empty Git repository in C:/BDAT-Sem2/Data Analytics Project/OAA/.git/

C:\BDAT-Sem2\Data Analytics Project\OAA>git add README.md
fatal: pathspec 'README.md' did not match any files

C:\BDAT-Sem2\Data Analytics Project\OAA>echo "# OAA" >> README.md

C:\BDAT-Sem2\Data Analytics Project\OAA>git add README.md

C:\BDAT-Sem2\Data Analytics Project\OAA>git commit -m "Adding OAA Code"
[master (root-commit) 4bfa04b] Adding OAA Code
1 file changed, 1 insertion(+)
create mode 100644 README.md

C:\BDAT-Sem2\Data Analytics Project\OAA>git branch -M main

C:\BDAT-Sem2\Data Analytics Project\OAA>git remote add origin https://github.com/rutvapatel1605/OAA.git

C:\BDAT-Sem2\Data Analytics Project\OAA>
```

Finally, we push the code to the remote repository using the command: `git push origin main`.

```
C:\Windows\System32\cmd.e x + v
C:\BDAT-Sem2\Data Analytics Project\OAA>git init
Initialized empty Git repository in C:/BDAT-Sem2/Data Analytics Project/OAA/.git/

C:\BDAT-Sem2\Data Analytics Project\OAA>git add README.md
fatal: pathspec 'README.md' did not match any files

C:\BDAT-Sem2\Data Analytics Project\OAA>echo "# OAA" >> README.md

C:\BDAT-Sem2\Data Analytics Project\OAA>git add README.md

C:\BDAT-Sem2\Data Analytics Project\OAA>git commit -m "Adding OAA Code"
[master (root-commit) 4bfa04b] Adding OAA Code
1 file changed, 1 insertion(+)
create mode 100644 README.md

C:\BDAT-Sem2\Data Analytics Project\OAA>git branch -M main

C:\BDAT-Sem2\Data Analytics Project\OAA>git remote add origin https://github.com/rutvapatel1605/OAA.git

C:\BDAT-Sem2\Data Analytics Project\OAA>git push -u origin main
Enumerating objects: 3, done.
Counting objects: 100% (3/3), done.
Writing objects: 100% (3/3), 227 bytes | 227.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To https://github.com/rutvapatel1605/OAA.git
 * [new branch]      main -> main
branch 'main' set up to track 'origin/main'.

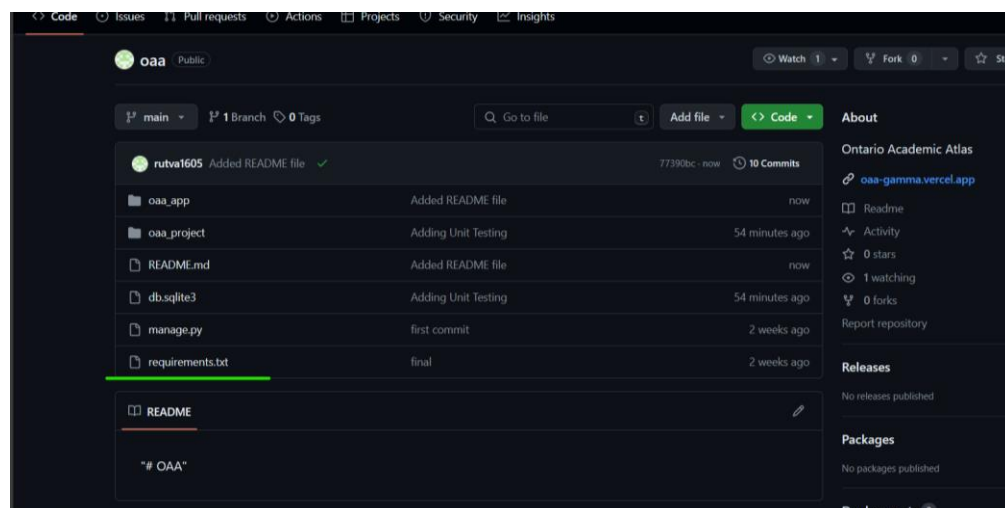
C:\BDAT-Sem2\Data Analytics Project\OAA>
```

Step 2: Deployment Process:

To deploy the project, we set up a server environment with all the necessary dependencies for running Django projects, including Python, Django, and a database server.

We clone the Git repository onto the server using the command: `git clone <repository_url>`.

Navigating to the project directory on the server, we install project dependencies via the command: `pip install -r requirements.txt`.

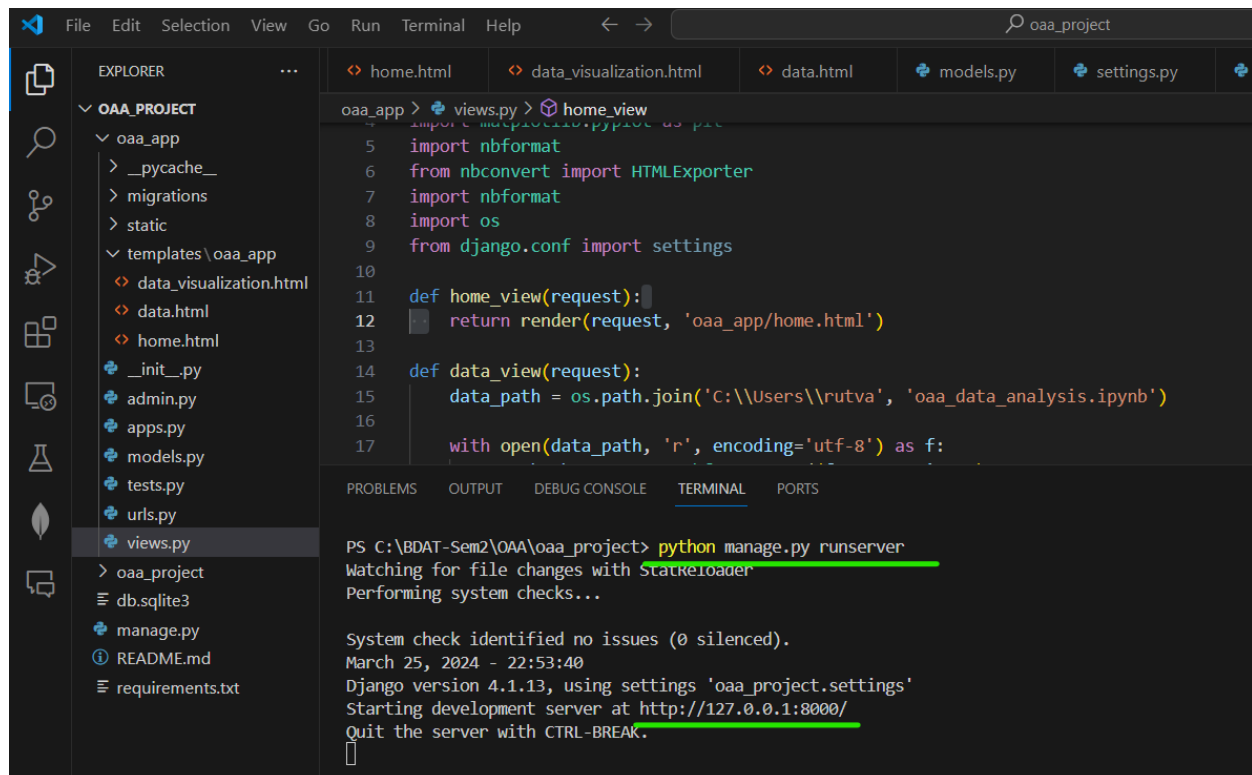


We configure the Django project settings for the production environment, adjusting database settings and security settings as necessary.

To collect static files, we execute: `python manage.py collectstatic`.

Migrating the database to reflect any changes, we use the command: `python manage.py migrate`.

Finally, we start the Django development server by running: `python manage.py runserver`.

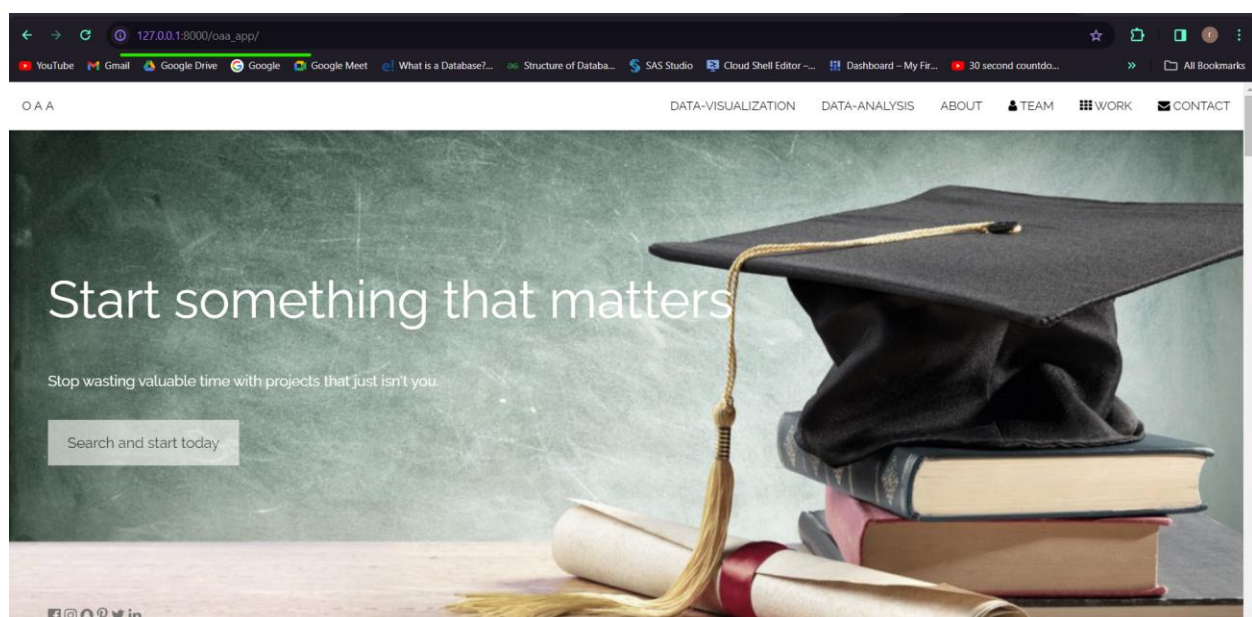


The screenshot shows a VS Code editor with the Django project structure on the left. The `oaa_app` directory is expanded, showing files like `__init__.py`, `admin.py`, `apps.py`, `models.py`, `tests.py`, `urls.py`, and `views.py`. The `views.py` file is open in the editor, showing the `home_view` and `data_view` functions. The terminal at the bottom shows the output of the `python manage.py runserver` command, indicating that the development server is running at `http://127.0.0.1:8000/`.

```
oaa_app > views.py > home_view
5 import nbformat
6 from nbconvert import HTMLExporter
7 import nbformat
8 import os
9 from django.conf import settings
10
11 def home_view(request):
12     return render(request, 'oaa_app/home.html')
13
14 def data_view(request):
15     data_path = os.path.join('C:\\Users\\rutva', 'oaa_data_analysis.ipynb')
16
17     with open(data_path, 'r', encoding='utf-8') as f:
```

```
PS C:\BDAT-Sem2\OAA\oaa_project> python manage.py runserver
Watching for file changes with StatReloader
Performing system checks...

System check identified no issues (0 silenced).
March 25, 2024 - 22:53:40
Django version 4.1.13, using settings 'oaa_project.settings'
Starting development server at http://127.0.0.1:8000/
Quit the server with CTRL-BREAK.
█
```



Step 3: Downloading and Running the Code:

- Visit the Git hosting service where the Ontario Academic Atlas project repository is hosted.
- Locate the "Clone or Download" button and click on it.
- Copy the repository URL provided.
- Open a terminal or command prompt on your local machine.
- Navigate to the directory where you want to download the project code.
- Clone the Git repository using the command: `git clone <repository_url>`.
- Navigate to the project directory.
- Install project dependencies using the command: `pip install -r requirements.txt`.
- Configure the Django project settings as needed for the local environment.
- Migrate the database using the command: `python manage.py migrate`.
- Start the Django development server using the command: `python manage.py runserver`.

Conclusion:

The deployment process outlined in this document provides a systematic approach to deploying the Ontario Academic Atlas project on a server environment and running it locally on a user's machine. By following these instructions, users can effectively deploy the project and access its functionalities either on a remote server or locally for development and testing purposes.