**Introduction**

In this reading, you will explore the solution steps to the capstone project.

**Step 1 – Setting up the project**

1. Sign up and login to [http://github.com](http://github.com/). Create a new repository named YourRepositoryName.
2. Clone your directory called workspace in your local machine.
3. Create a Python virtual environment in the workspace directory.
4. Install the Django framework in this virtual environment.
5. Create a Django project called YourRepositoryName.
6. Create a new Django app called **yourApp**.
7. Define the **index()** view that renders the home page of your application – index.html. Configure the URL routes.
8. Put the **index.html** in the templates folder under the **BASE\_DIR**.
9. Run the server and check if the home page is displayed.
10. Clone your GitHub repository in the outer YourRepositoryName folder.
11. Add and commit the project files.
12. Push the changes to the remote repository.

**Step 2 – Setting up the MySQL database and users**

1. Install the PostgreSQL library.
2. Configure your project for the PostgreSQL database.
3. Declare the Menu and Reservation models.
4. Migrate the models.
5. Register the models with the admin app.
6. Create a superuser.
7. Add and commit the changes to your project.
8. Push the changes to the remote repository.

**Step 3 – Implementing the Django REST Framework**

1. Install the Django REST Framework.
2. Add restframework to **INSTALLED\_APPS** and run migrations
3. Declare serializer classes for the **menu** and **reservation** models
4. Declare **ModelViewSet** classes for the **menu** and **reservation** models, and set the **serializer\_class** attribute of each.
5. Use **DefaultRouter** to register views.
6. Add the **routers.urls** to URLConf.
7. With DRF’s browsable API, perform CRUD operations on menu and reservation models.
8. Add and commit the changes to your project.
9. Push the changes to the remote repository.

**Step 4 – Installing Djoser and testing with Insomnia**

1. Install the djoser library
2. Add **djoser** to INSTALLED\_APPS and run migrations
3. Configure the **settings.py** file for token authentication.
4. Set the **permission\_classes** attribute of the ViewSets to **IsAuthenticated**
5. Obtain the authorization token from the admin interface.
6. Use the token to authenticate the API endpoints.
7. Use ThunderClient to test the API endpoints.
8. Add and commit the changes to your project.
9. Push the changes to the remote repository.

**Step 5 – Creating the unit tests**

1. Create a **tests** folder in the YourRepositoryName project package folder.
2. Use Django’s testing framework to write unit tests for testing the models and views.
3. Run the tests using manage.py.
4. Add and commit the changes to your project.
5. Push the changes to the remote repository.

In this reading, you explored the solution steps to the capstone project.