# Workshop: Todo’s App

For this workshop, you will be provided with a client app. You should only implement the required APIs.

## Part I – Authentication

### Creating the Auth app

Create an **todos\_auth** app that handles the authentication

### Views

Implement session authentication using Django **REST Framework**:

For this part, we will use the default user model, provided in Django.

We need to implement the Login/Register/Logout views.

* RegisterView implements CreateAPIView
* LoginView implements APIView and has a custom **post** to login the user
* LogoutView implements APIView and has a custom **post** to logout the user

RegisterView and LoginView should be public

LogoutView should be private

### URLS

Add the following urls:

* **'localhost:8000/api/auth/register/'** – registers an user
* **'localhost:8000/api/auth/login/'** – logs in an user
* **'localhost:8000/api/auth/logout/'** - logs out an user

## Part II – Todos

### Creating the Todos app

Create a new **app** called **'todos'** that handles the logic for todos

### Models

Create the following models:

#### Category

* Name - maximum 15 characters

#### Todo

* Title - maximum 30 characters
* Description - any text, with no limit of words/chars
* Category – a relation to category
* State – can be either “Done” or “Not done”

### URLs

Add the following **urls**:

* **POST 'localhost:8000/api/todos/'** – creates a todo for a logged-in user
* **GET 'localhost:8000/api/todos/'** – returns the todos for the logged-in user
  + Additional query parameters can be provided to further filter the result:
    - `category` - only todos to a category
    - `is\_done` - True/False
  + **GET 'localhost:8000/api/todos/?category={CATEGORY\_NAME}'**
  + **GET 'localhost:8000/api/todos/?is\_done={True/False}'**
  + **GET 'localhost:8000/api/todos/?is\_done={True/False}& category={CATEGORY\_NAME}'**Modifying Models

### Additional

* Try to add Profile for each user. A profile should have first and last names
* Try to add `assignees` to each todo, i.e. a user can assign a todo to another user