

Full-stack Angular + Django

The goal of this exercise is to build a little application composed of an input form and a video view (player or empty square if there is nothing to display) below.

- The input form takes a YouTube video URL (not embed url) as entry and displays the YouTube video inside the video view.**
- Additionally, each request should create an entry in a local history list shown on the left side of the page. If an item of the history is clicked, the related video should be shown again.**
- A bookmark button will be added. When clicking on it, the URL of the playing video will be added to bookmarks. Another button will give access to the list of all the bookmarks. The position of this button is up to the candidate.**
- The list of bookmarks and history will be stored through the Django backend application. When the page is loaded, an indicator (label or other) should show the number of bookmarks saved.**

I. Front-end Angular

The candidate will develop 4 components and these components should be added in the app.component.html file like this:

```
...  
<searchBar ... ></searchBar>
```

```
<videoView ... ><videoView>  
<History ... ><History>  
<Bookmarks ... ><Bookmarks>  
...
```

The candidate will use the framework Angular version 8 or newer (with TypeScript) and *not* AngularJS.

The application should run without errors when the command "*ng build --prod*" is run in the project directory. The candidate should test the installation of their application before returning it.

The candidate can use Bootstrap, Font Awesome or any other libraries for rendering purposes.

Before linking this application to the API (in II.) the candidate can use the JavaScript *localStorage* to store and get the history and bookmarks.

II. Back-end Django

The candidate will develop a simple Django API to store the user's history and bookmarks through two entities: History and Bookmark.

The API will provide 4 routes to:

- List all videos in the history
- Add a video to the history

- List all videos in the bookmarks
- Add a video to the bookmarks

There is no need to build any authentication mechanism on this API.

III. Full-stack

The candidate will link both applications to create a fully working video player application.

Some requirements/clues:

- The candidate should build a service in the Angular application that will handle calls the Django API.
- The service will be used to get the history and bookmarks and store them, as well as adding new items to the history and bookmarks.
- Be careful to handle CORS issues.
- The application should run on a single computer, using the port 8000 for the Django API and 4200 for the Angular app.

Please provide all necessary information to run your application.