**Fans Football Federation (FFF)**

You can reach the web APP via this link =>

**Description of the project**

**What is FFF?**

FFF is a web-app where users can upload their own news about football, follow other users and evaluate referees after every match.

**Why FFF?**

**FFF** is an user friendly and eye-pleasing alternative to the current web-site of Turkish Football Federation. With FFF fans can have followers and follow other users, TFF members can upload, delete and update news and fans can contact with responding people easily.

**USERS**

* User can register to FFF by clicking the “register” button at the top of the web-site.
* To register succesfully users have to enter unique and valid username (only accept alphabetic chracters) and email adress. Besides they have to set a password which is at least 6 chracter length.
* After register user can login with their username and passwords to use the feautures of the web-site.

**User Documentation**

**How to install the software?**

To install the software users can clone our repository to a local directory by opening a terminal and copying this command:

$ git clone https://github.com/SU-CS308-22FA/TEAM-38.git

**How to run the software?**

Users can click ….. to run the software.

Users can also start the software at their http://localhost:3000

**How to report a bug?**

We would appreciate if you follow the steps below to report a bug:

1. First of all show the exact location of the bug.
2. Explain why you think that is a bug.
3. Describe the expected behaviour and explain if you have an idea how to fix that bug.
4. Send us an email: [fansfootballfederation@gmail.com](mailto:fansfootballfederation@gmail.com)

**Known bugs?**

The option of deleting and updating an account seems to be problematic. Other than that there is no bug that we found currently but of course it is possible to detect all the bugs after releasing our app to users.

**Developer Documentation**

The FFF project is made using:

* [Mongo DB](https://www.mongodb.com/)
* [Express JS](https://expressjs.com/)
* [Node JS](https://nodejs.org/)
* [Cloudinary](https://cloudinary.com/)

**How to obtain the source code?**

All source files can be found on our github repository pages. You can click the branch button and reach out all the branches to find the source codes.

If you want to obtain this source codes:

* Click the code button on the right upper side of the repository
* Download Zip
* Extract zip file

If you download github desktop append the repository and chose the option clone.

If you created a folder on code IDE then use this command to clone the repository with all source codes:

$ git clone <https://github.com/SU-CS308-22FA/TEAM-38.git>

**The layout of project folder/file structure**

**Frontend side**

* **public – Public holds css, fonts, icons and images, i.e. static files**
* **views- This holds all the frontend ejs files of our project**
  + **partials - This folder has the \_header, \_menu and \_footer.ejs which we add to almost every other ejs files on our project so that we do not need to write them again.**
  + **index.ejs – This use the html templates fort he initial page of our project**
  + **dashboard.ejs - This file open the individual pages fort he each user who login.**
* **app.js - This import the neccessary modules and renders all of our browser routes**

**Backend side**

* **controller - This holds all the functions at the backend which help us to develop the feautures in our app.**
* **models - This is where we create schemas for MongoDB**
* **routes - This holds all URL path associations for each unique url**
* **package.json - Defines npm behaviors and dependencies with packages for the backend. Also includes first scripts to run the app**
* **tmp – Temporary images that user upload to our app occur in that folder, after this images uploaded at cloudinary they are deleted from folder temp so that user does not have to worry about local memory.**

**README – This gives the basic information about our app**

**How to build and deploy the software?**

**To build software**

1. Follow the instructions above to get source code
2. Install npm packages that you see on app.js file
3. Set the env variables
   * DB\_URI = the uri string for your mongo-database
   * PORT= If you want to determine a specific port write it and then you can work on that port
   * JWT\_SECRET = to hide the password create a jason web token [here](https://jwt.io/)
   * CLOUD\_NAME,API\_KEY = to not to worry about local memory connect to cloudinary from [here](https://cloudinary.com/)
4. To start the application

$ npm run dev

**To deploy to Heroku**

* Open a free account
* Link your github repository to this Heroku account
* Set environment variables on Heroku
* Press deploy