

AREA Technical Documentation

Name : AREA

Start: 19/09/2022

End : 13/11/2022

Goal: The AREA is a web and mobile application that can allow its user to choose reactions made by a platform according to an action that has been made with another platform (for example : receiving an email with the number of likes from a video as soon as it is liked, reaction action between YouTube and Gmail). As soon as there is an action coming from a platform, this one will generate the reaction of another platform which in turn will make an action

The user should be able to :

- ✓ 1. Create an account.
- ✓ 2. Log in via your Google, Facebook or Twitter account.
- ✓ 3. Be able to connect to different platforms (Discord, Youtube, Twitch, Google, Twitter, Spotify).
- ✓ 4. Being able to choose reaction actions according to the connected platforms.
- ✓ 5. Being able to deactivate reaction actions.
- ✓ 6. Being able to disconnect from a platform.
- ✓ 7. Being able to disconnect from the AREA.
- ✓ 8. Have access to their account on the web and mobile app with the same saved settings.

Languages used :

Web Application : Java Script

Mobile Application : Progressive web app React PWA

Applications used for the project :



Docker : Launch applications in software containers.



Notion : Follow-up of the project, meetings as well as tasks throughout its development.



GitHub : Project hosting.

GitHub

ssh : [git@github.com:EpitechPromo2025/B-DEV-500-LIL-5-1-area-eliott.ayetand.git](ssh:git@github.com:EpitechPromo2025/B-DEV-500-LIL-5-1-area-eliott.ayetand.git)

https : <https://github.com/EpitechPromo2025/B-DEV-500-LIL-5-1-area-eliott.ayetand.git>



Git flow : Git branch management.

Organization of the deposit « [B-DEV-500-LIL-5-1-area-eliott.ayetand](#) » :

We use Git flow to organize our different branches. This allows team development while limiting conflicts. Everyone has the opportunity to work in their corner to develop features and merge them at any time in the project. We can also check the proper functioning of our project step by step in order to have on our master branch, which is our main branch of notation, a functional project.

Git flow documentation : <https://git-flow.readthedocs.io/fr/latest/presentation.html>

ARCHITECTURE

Server :

- actions/ : All the actions' functions, one for each action
- config/ : Where the connection with passport are
- controllers/ : The functions that handle the CRUD system
- middlewares/ : Contains the middlewares functions
- models/ : Control the models (database representation), according to CRUD system
- routes/ : Every routes defined in our API

Client :

- actions/ ; reducers/ ; services/ : Implement the React Redux system
- components/ : Components that can be reused across the whole application
- config/ : Handle the API requests'
- views/ : Every page of the application
- assets/ : Contains every image of the web app