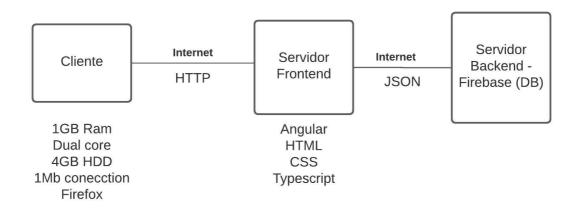
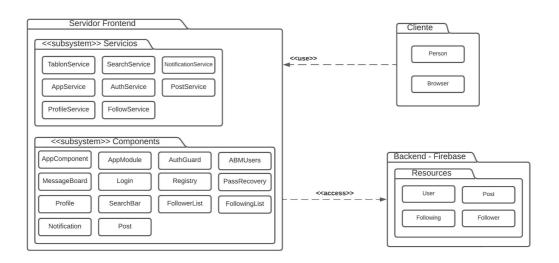


Arquitectura

1. Identificar los nodos y la configuración de la red (Diag. de Despliegue)



2. Identificar subsistemas y sus interfaces



3. Identificación de clases de diseño relevante para la arquitectura.

Cliente

<<entity>> Person

- name - lastname - email
- birth_day
- passwordbrowser

<<entity>> Browser

- name

Frontend

<<entidad>> AppComponent

<<entidad>> AuthGuard

<<interface>> MessageBoard

- messageList
- + displayPosts()

<<interface>> Login

- usernameField
- passwordField - loginButton
- + signIn(username,

<<interface>> Register

- usernameField - nameField
- lastnameField - emailField
- birthdayField
- passwordFieldregisterButtton
- + register(registryParams)

<<interface>> Followers

- followerList
- + removeFollower()

<<entidad>> AppModule

<<interface>> ABM Users

- usersId -usernames
- + editUser() + deleteUser()

<<interface>> Profile

- username
- user_messages
- + displayFollowing() + displayFollowers() + displayNotifications() + postMessage()

<<interface>> SearchBar

- promptField
- + searchElement(prompt) + displayElement()

<<interface>> Following

- followingList
- + unfollowUser()

<< interface>> Notification

- notificationTitle - notificationBody
- + displayNotification()

<< interface>> PasswordRecovery

- emailField - recoveryButton
- + recoverPassword()

<<interface>> MessageBoard

+ displayMessages() + displayAdverts()

<<interface>> Post

- postID
- userID - body
- likedBy
- repostedBy + likePost(token)
- + repostPost(token) + editPost(token)

<<control>> AdvertService

+ getAds()

<<control>> UserService

- + followUser(userID. userToken)
- + unfollowUser(userID, userToken)

<<control>> NotificationService

+ getUserNotifications(userID)

<<control>> SearchService

- + filterUsers(filterParams)
- + filterPosts(filterParams) + getUsers(searchParams)
- + getPosts(searchParams) + getUser(userId)
 - + getPost(postId)

<<control>> AppService

<<control>> AuthService

- + verifyUser(username,
- password) + generateHash(plainPassword)

- <<control>> PostService
- + likePost(postID, token)
- + getPost(postID) editPost(postID, userToken)
 - + deletePost(postID, userToken)

Backend

<<entity>> User

- id username - name
- lastname
- email - birthday
- password
- is_admin
- + getUsers(searchParams) + aetUser(userId)
- addUser(userParams)
- + deleteUser(userId) + modifyUser(modParams)

<<entity>> Post

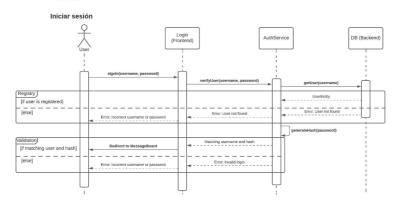
- id username
- message
- + getPosts(searchParams)
- + getPost(postId) addPost(postParams) + deletePost(postId) + modifyPost(modParams)

<<entity>> Follow

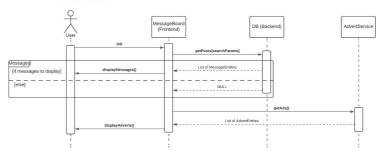
- followed_idfollower_id
- + getFollower(follower_id)
- + addFollower(followParams) + deleteFollower(followParams)

Diagrama de secuencia

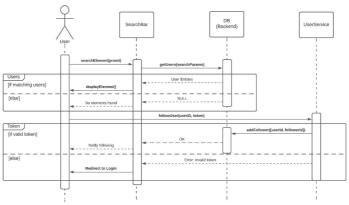
Punto 4



Acceder al tablón



Seguir usuario



Dejar de seguir usuario

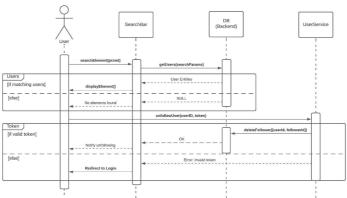
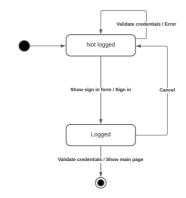
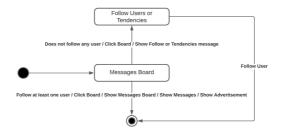


Diagrama de estado

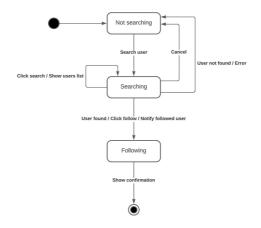
Iniciar sesion



Acceder al tablon de mensajes



Seguir Usuario



Dejar de Seguir Usuario

