

!ReadyTech

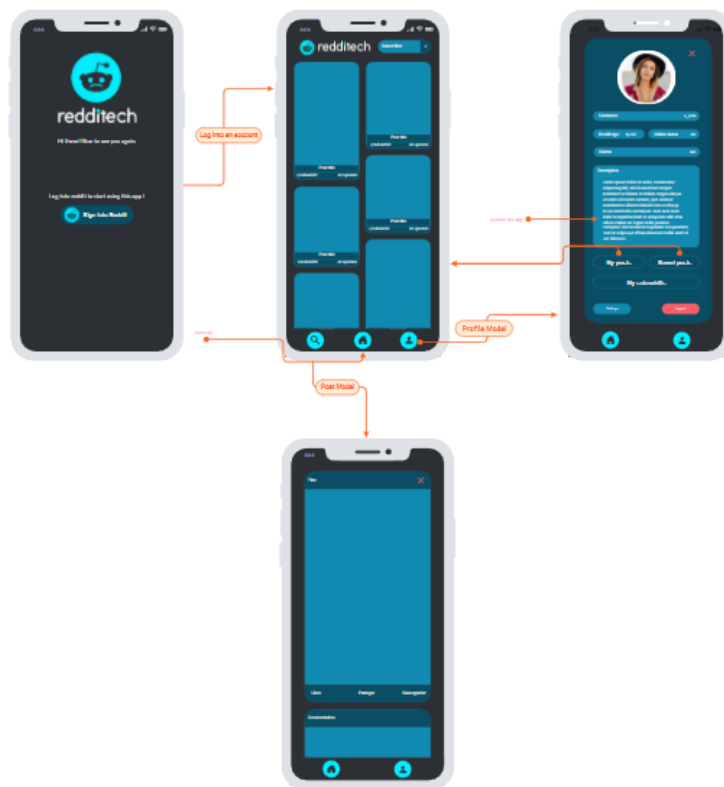
Bastien Pardini
Florent Dreux
Florian André
[NOM DE LA SOCIETE]

Technical documentation:

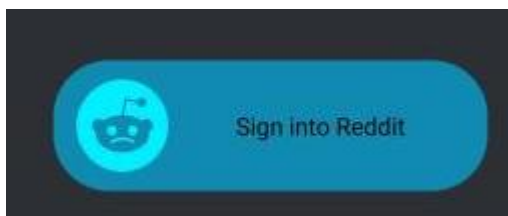
Summary

Sketch presentation:	2
Page presentation:	4
Login page :	4
The nav bar:	5
Home page :	6
Profile page:	7
Installation:	9

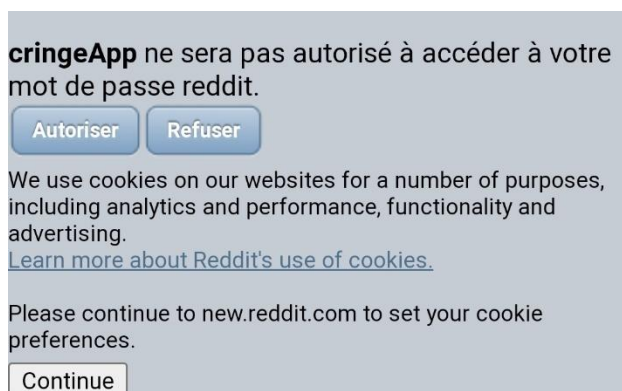
Sketch presentation:



We created this sketch using the moqups.com website. We can see a first page that contains a button "Sign into reddit".



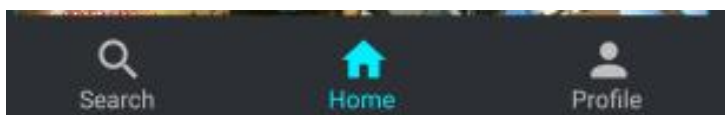
This page redirects us to an Oauths2 authorization.



When we authorize the connection, it displays our reddit feed.



We can see three buttons on the nav bar: a search button / a home button / a profile button.



The search button would offer the possibility to search a feed

The home button offers the possibility to go back to the top of his feed (see above)

The profile button offers to display his reddit profile.



Page presentation:

Code explanation

Login page :

To display our first login page, we will use this code:

```
export default function Login({ navigation }) {

  const [request, response, promptAsync] = useAuthRequest(
    {
      responseType: ResponseType.Token,
      clientId: id,
      scopes: ['*'],
      redirectUri: makeRedirectUri({
        scheme: 'com.redditech://'
      }),
    },
    discovery
  );

  const saveToken = async (newToken) => {
    await SecureStore.setItemAsync("token", newToken);
  }

  React.useEffect(() => {
    if (response?.type === 'success') {

      const newToken = response.authentication.accessToken;

      saveToken(newToken).then(navigation.navigate("Home"))
    }
  }, [response]);

  return (
    <View style={styles.login}>

      <View style={styles.loginItems}>
        <Image
source={require('.././components/Images/Round_reddit_white_title.png')}
style={styles.logo}/>

        <TouchableOpacity
          onPress={()=>{promptAsync()}}
          style={styles.button} >
          <Image
source={require('.././components/Images/Round_reddit.png')}
style={styles.buttonImage}/>
          <Text style={styles.buttonText}>Sign into Reddit</Text>
        </TouchableOpacity>
      </View>
    </View>
  );
};
```

This page displays an image with the button. Which calls the OAuths when the button is pressed. And redirects to the main page.

The nav bar:

```
export default function Nav() {
  return (
    <Tab.Navigator
      independent={true}
      initialRouteName={"Home"}
      activeColor="#00eeff"
      inactiveColor="#c0c0c0"
      barStyle={{ backgroundColor: '#2c2f33' }}
    >
      <Tab.Screen
        name="Search"
        component={Search}
        options={{
          tabBarLabel: 'Search',
          tabBarIcon: ({ color }) => (
            <MaterialCommunityIcons name="magnify"
              color={color} size={26} />
          ),
        }}
      />
      <Tab.Screen
        name="Home"
        component={Home}
        options={{
          tabBarLabel: 'Home',
          tabBarIcon: ({ color }) => (
            <MaterialCommunityIcons name="home" color={color}
              size={26} />
          ),
        }}
      />
      <Tab.Screen
        name="Profile"
        component={Profile}
        options={{
          tabBarLabel: 'Profile',
          tabBarIcon: ({ color }) => (
            <MaterialCommunityIcons name="account"
              color={color} size={26} />
          ),
        }}
      />
    </Tab.Navigator>
  );
}
```

These few lines of code allow you to redirect to the "Search"/"Home"/"Profile" pages. By calling the page with the same name.

Home page :

This page allows the display of the user's feed.

```
<ScrollView>
{
  posts.map((post) => {
    return (

      <Post
        key={Math.random() * ( numberOfPosts - 1 ) + 1}
        title={post.data.title}
        author={post.data.author}
        subreddit={post.data.subreddit_name_prefixed}
        contentType={post.data.post_hint}
        content={post.data.selftext}
        imageSrcUri={post.data.url_overridden_by_dest}/>

    )
  })
}
</ScrollView>
```

The scrollview retrieves the user's data and displays a post. The scrollview randomly displays this post by retrieving all the necessary data like:

- The author
- The subreddit of the post
- The content
- The url of content

And this for all the posts that follow.

Profile page:

```
const getData = () => {  
  
  let headers = {  
    'Authorization': `bearer ${token}`,  
    'User-Agent': USER_AGENT  
  }  
  
  axios.get(REDDIT_API + "/me", {  
    headers : headers  
  }).then((response) => {  
    setUsername(response.data.name);  
    setCreatedAt(response.data.created_utc);  
    setFriends(response.data.num_friends);  
    setAvatar(response.data.snoovatar_img);  
    setKarma(response.data.total_karma);  
    setDescription(response.data.subreddit.public_description)  
  
  }).catch((error) => {  
    console.log(error)  
  });  
}
```

This page works in the same way as the main page. It retrieves the necessary data from the user to display them. The following information can be found here:

- The username
- The date of creation
- The number of friends
- The avatar of reddit created by the user
- The number of karmas
- The description

The settings:

```
const bool_datas = () =>{
  let headers = {
    'Authorization': `bearer ${token}`,
    'User-Agent': USER_AGENT
  }
  axios.get(REDDIT_API + "/me/prefs", {
    headers : headers,

  }).then((response) => {
    setBool_values({
      'over_18':response.data.over_18,
      'video_autoplay':response.data.video_autoplay,
      'show_snoovatar':response.data.show_snoovatar,
      'show_presence':response.data.show_presence,
      'email_chat_request':response.data.email_chat_request,
      'enable_followers':response.data.enable_followers,
    })
  }).catch((error) => {
    console.log(error)
  });
}

const updateData = () =>{
  let headers = new Headers()
  headers.append('Authorization', `bearer ${token}`)

  let requete = {
    method:"patch",
    headers:headers,
    body:JSON.stringify(bool_values)
  }
  fetch("https://oauth.reddit.com/api/v1/me/prefs", requete).catch(error
=> console.log(error))
}
```

The settings are opened in modal. That is to say that a window opens on top of the active one. The settings can be modified by the user. We find there:

- Age (+18)
- Enabling followers
- Show presence
- Show snoovatar
- Video autoplay
- Email chat request

Installation:

To proceed with the installation, you will need to enter two simple commands.

The first one is to launch an install node module:

Npm install

Once the installation is finished, this command must be entered:

Npm run web

If the command fails, this one will work:

expo start

Then a QrCode appears, and you could test your application.

ENJOY!