

Guia d'estil

Index

INTRO	3
DEPENDÈNCIES:	4
EQUIP	5
REQUISITS	7
FUNCIONALS	7
REGLES DE NEGOCI:	8

Introducció

Per al desenvolupament de la nostra aplicació mòbil, hem triat utilitzar **el framework React Native**. Aquesta plataforma ens permet construir una base de codi única en el llenguatge de programació **JavaScript**, que pot ser desplegada tant en dispositius Android com en dispositius iOS. La flexibilitat de React Native ens permet crear una experiència d'usuari responsiva i fluida.

Pel que fa al backend, emprem **Firebase**, una plataforma de desenvolupament d'aplicacions mòbils i web proporcionada per Google. Firebase ens ofereix eines com una base de dades en temps real, autenticació d'usuaris i emmagatzematge en el núvol, entre altres funcionalitats importants, sense necessitat de gestionar directament el backend.

A més d'això, estem utilitzant diverses APIs per millorar la funcionalitat de l'aplicació. Hem seleccionat les APIs de **Mapbox** i **Google Maps** per a la integració de mapes i geolocalització. Aquestes APIs ens permeten mostrar mapes interactius i oferir indicacions de rutes en temps real.

En resum, estem combinant les capacitats de React Native, l'eficàcia de Firebase com a plataforma de backend, i la funcionalitat de diverses APIs, com ara les de Mapbox i Google Maps, per a crear una aplicació mòbil completa i funcional per als nostres usuaris.

Dependències:

Per implementar el projecte informàtic que utilitza React Native, cal seguir aquestes passes per a la configuració de l'entorn de desenvolupament en diferents sistemes operatius. El següent resum destaca els passos principals per a la configuració i execució de l'aplicació React Native:

1. Instal·lació de dependències:

- Cal Node.js i el React Native Command Line Interface (CLI).
- S'ha de tenir instal·lat un Java SE Development Kit (JDK).
- Pots utilitzar Chocolatey per instal·lar Node.js i el JDK.

2. Entorn de desenvolupament Android:

- Cal instal·lar Android Studio, Android SDK (Tiramisu), i les eines relacionades.
- Cal configurar la variable d'entorn ANDROID_HOME per a les eines de React Native.

3. Preparació del dispositiu Android:

- Es pot utilitzar un dispositiu Android físic o un Android Virtual Device (AVD) per emular un dispositiu Android al teu ordinador.
- Si utilitzes un AVD, cal assegurar-se que tinguis instal·lat Intel HAXM per a un rendiment òptim.

4. Preparació del dispositiu iOS:

- S'haurà de instal·lar una màquina virtual per simular el macOS.
- S'haurà de instal·lar node i watchman.
- El simulador serà xcode.

5. Execució de l'aplicació React Native:

- Iniciar el Metro Bundler amb npm start en una terminal.
- Iniciar l'aplicació amb npm run android en una altra terminal.
- L'aplicació hauria de començar a córrer en l'emulador o en un dispositiu Android físic.

Aquest és un resum dels passos principals per a la configuració i execució d'un projecte React Native.

Segueix les instruccions d'instal·lació aquí: <https://reactnative.dev/docs/environment-setup>
Selecciona React Native Cli.

Equip

Per a garantir un desenvolupament col·laborat i organitzat del nostre projecte, hi ha algunes consideracions i eines que serà important tenir en compte.

En primer lloc, per gestionar el codi font i les versions del nostre projecte, farem servir GitHub. Això implica que caldrà tenir instal·lat **Git** al teu sistema. **Git** és una eina de control de versions que ens permetrà col·laborar eficaçment, mantenir un historial del codi i gestionar canvis de manera coherent. El repositori utilitzat pel moment serà <https://github.com/Pilouco26/app-mapa-react.git>.

Pel que fa a la convenció de noms en JavaScript, és essencial seguir unes pràctiques coherents per mantenir un codi llegible i organitzat. Utilitzarem la notació **camelCase** per als noms d'arxius, carpetes i variables. A més, per als noms de components en React, farem servir la notació **PascalCase**. Aquestes pràctiques faciliten la comprensió del codi i fomenten una estructura uniforme.

Un altre aspecte clau és l'ús de **noms descriptius i significatius**. Triar noms que reflecteixin amb precisió la funció o propòsit d'una variable, funció, arxiu o carpeta ajudarà a tot l'equip a comprendre millor el codi i a treballar de manera més eficaç. Per exemple, si una funció calcula l'àrea d'un cercle, li direm `getCircleArea` i no pas `funcio3`. Els noms de variables i funcions hauran d'estar amb **angles**, per facilitar la recerca a internet i mantenir l'harmonia del idioma en el projecte.

Finalment, per a la comunicació i la col·laboració en l'equip, utilitzarem **Microsoft Teams** com a canal principal. Aquesta plataforma ens permetrà mantenir converses, compartir fitxers i coordinar les diferents tasques de manera organitzada.

Amb aquestes eines i pràctiques, estem assegurant un enfocament ordenat i eficient en el desenvolupament del nostre projecte, amb l'objectiu de produir un codi de qualitat i una comunicació efectiva en tot moment.

Per al desenvolupament de la nostra aplicació, recomanem utilitzar Visual Studio Code (VSCode) com a entorn de desenvolupament. VSCode és un editor de codi lleuger i potent amb una gran quantitat d'extensions que ens poden ajudar a millorar la nostra productivitat i eficiència. Aquí tenim algunes extensions específiques que ens poden ser útils:

1. **Colorize**: Aquesta extensió ressalta els colors en codi CSS i LESS, fet que fa més fàcil identificar i entendre els esquemes de colors.
2. **CSS Peek**: Aquesta extensió ens permet fer un seguiment dels selectors CSS directament al seu origen, fet que fa més fàcil veure com s'aplica el CSS als elements HTML.
3. **Import Cost**: Aquesta extensió ens mostra la mida dels imports de JavaScript/TypeScript directament a la línia de codi. Això ens pot ajudar a identificar quins imports poden afectar el rendiment.
4. **GitLens**: Aquesta extensió millora la integració amb Git, proporcionant informació detallada sobre cada línia del nostre codi, com ara quan va ser modificada i per qui.
5. **Indent Rainbow**: Aquesta extensió ressalta els nivells d'indentació amb colors, fet que facilita la visualització de les estructures de codi.

6. **Prettier:** Prettier és una eina de formateig de codi que ens permet mantenir un estil de codi consistent. L'extensió integra Prettier amb VSCode per a un formateig automàtic i coherent del nostre codi.
7. **Simple React Snippets:** Aquesta extensió ofereix una col·lecció d'esquemes de codi predefinits per a React. Això pot accelerar el nostre desenvolupament proporcionant fragments de codi comuns amb només uns pocs clics.
8. **ES7+ React/Redux/GraphQL/React-Native snippets:** Aquesta extensió proporciona una sèrie de snippets per a les tecnologies més comunes en el desenvolupament amb React. Aquests snippets ens permeten generar ràpidament porcions de codi comuns.

En resum, utilitzar Visual Studio Code com a entorn de desenvolupament juntament amb aquestes extensions ens proporcionarà eines valioses per a la millora de la nostra productivitat i la qualitat del nostre codi en el desenvolupament de l'aplicació.

Requisits

Funcionals

L'aplicació es dividirà en tres pantalles principals: "Amics", "Mapa" i "Tiquets". A més d'aquestes, també inclourà altres pantalles com "Entrada", "Tutorial", "Inici de sessió" i la visualització detallada de "Bar/Restaurant". En tots les pantalles hi serà present el *"footer"*, un component compost d'ícones que ens portarà a les 3 pantalles principals.

Amics serà la pantalla per poder afegir persones a la teva xarxa, gràcies a un **barra de cerca**.

Les funcionalitats estaran limitades ja que només podràs cercar, afegir, consultar i esborrar amics, no s'espera incloure un xat de text. Metode **CRUD**. Amb la col·laboració s'espera que els usuaris puguin aconseguir millors codis.

Mapa es la gràcia d'aquesta aplicació, s'espera crear un mapa d'Andorra. En aquest mapa s'ha de veure la teva localització i la dels teus amics, la localització s'ha de visionar amb un avatar/stickman. **Els bars seran interactuaves**, quan cliquis sobre un sortirà un **desplegat**, on donarà **poca informació** sobre els teus tiquets i miscel·lània, per a més detalls es redimirà a la pantalla del bar on sortirà més aviat informació sobre el bar. Serà la **pantalla de primer visionat**.

A **tiquets** s'espera en primer lloc, un modal per **bescanviar codis**. Després, es veurà la **llista de tiquets disponibles** per usar i al final una **sèrie de reptes**. Un possible repte seria aconseguir tiquets a 5 bars diferents, un sistema de *"grind"*.

Queda com a incògnita **com aplicarem aquest sistema de codis** i com seran capaços d'adoptar aquests codis les factures del bars.

Footer serà un component permanent a totes les pantalles de redirecció a les tres pàgines principals. Es constituirà de tres icones principals, no totes les icones estan decidides però està clar que el redireccionament al mapa serà una icona de la silueta d'Andorra.

Petita entrada serà una pantalla per amagar el temps de render del mapa, només s'executarà a la inicialització de l'aplicació. Qualsevol cosa serveix, un logotip i un nom.

Pestanya **Restaurant** es la última pantalla de l'aplicació, ha de donar informació sobre contacte, que tenen i **vibes de "reseñas de google"**.

Inici de sessió es la pantalla per poder accedir a totes les funcionalitats. Es podrà accedir a través de la aplicació i de google.

Regles de Negoci:

1. Un usuari no pot afegir dos cops al mateix amic.
2. Un usuari no pot bescanviar el mateix codi dos cops.
3. Els usuaris han de ser únics.
4. Un usuari ha de estar iniciat per poder accedir als tiquets. (prerequisit)
5. Un usuari no pot esborrar a un usuari no amic.
6. Un usuari no pot veure la localització d'altres usuaris que nos siguin amics seus.
7. Un usuari no pot utilitzar el mateix tiquet dos cops.
8. No pot haver un codi amb la mateixa id.
9. No pot haver dos restaurants amb la mateixa id.
10. L'idioma de l'aplicació és el català.
11. Els reptes són únics i només completaves un cop.

