Documentation of Gnokky

Sommario

[Introduction 2](#_Toc140500405)

[Analysis & planning 2](#_Toc140500406)

[Setup Environment 3](#_Toc140500407)

[Installation 3](#_Toc140500408)

[Expo CLI 3](#_Toc140500409)

[Use Expo CLI 4](#_Toc140500410)

[Install & Setup Android Studio 4](#_Toc140500411)

[Create your first project 4](#_Toc140500412)

[Implementation 5](#_Toc140500413)

[Folder structure 5](#_Toc140500414)

[Admin 5](#_Toc140500415)

[Auth 6](#_Toc140500416)

[Login Page 6](#_Toc140500417)

[Chat 7](#_Toc140500418)

[ChatPage 7](#_Toc140500419)

[ChatTemplate 7](#_Toc140500420)

[ListResearchChat 7](#_Toc140500421)

[Message 7](#_Toc140500422)

[Home 7](#_Toc140500423)

[Models 7](#_Toc140500424)

[Navigations 7](#_Toc140500425)

[Notifications 7](#_Toc140500426)

[Post 7](#_Toc140500427)

[Profile 7](#_Toc140500428)

[Repost 7](#_Toc140500429)

[Search 7](#_Toc140500430)

[Stories 7](#_Toc140500431)

# Introduction

Our goal was to create a social media (like Instagram) with the framework “expo”. Expo is framework of react-native, that is a language based on JavaScript and is mainly used for creating mobile application. We chose to work with expo because we would like to learn to use this framework.

For the database we used “firebase” that is a cloud database offered by google and we chose it because it can offer some useful function (for example the email confirmation when you create an account).

# Analysis & planning

Before we start with the implementation, we dedicated a few days to analyse our project. We made the assignment book, that contains all the functionality that our app must have, some mock-up, a use-case diagram, the test-cases, and a Trello board.

Trello is a website and we used it for arrange our project. In the board there are lists of activity and every morning we make a little meeting where we chose which activity we want to do during the day.

Immagine che contiene testo, schermata, software, Software multimediale

Descrizione generata automaticamente

Figure 1 Trello board

# Setup Environment

There is a little guide for setup the environment on Windows 10/11.

## Installation

To develop applications with Expo, you will want to start with two tools:

* Expo CLI
* Expo GO (Auto installed when you start your first project)

### Expo CLI

To use Expo CLI, you need to have the following tools installed:

* NodeJS
* Git

#### NodeJS

Download the installer here → [https://nodejs.org/en](https://nodejs.org/en%20)

Only Node.js LTS releases (even-numbered) are recommended.

Once the download has finished, run the installer, and click next leaving everything default except for this one where **I recommend you tick the checkbox:**

Immagine che contiene testo, schermata, software, Pagina Web

Descrizione generata automaticamente

Figure 2 NodeJS download

Next, the terminal will be opened, follow the instructions written on it and wait until the installation is finished.

#### Git

To install Git, click here → [https://git-scm.com/download/win](https://git-scm.com/download/win%20) and choose:

*“64-bit Git for Windows Setup”*

Simply run the exe and let everything default.

## Use Expo CLI

To check if Expo CLI is it working type in terminal:

**npx expo -h**

It will just show you useful information about npx expo command.

## Install & Setup Android Studio

## Create your first project

Open CMD and type

**npx create-expo-app my-app**

and once it has done

**cd my-app**

Install dependencies (only if you need to develop this application also for web)

**npx expo install react-dom react-native-web @expo/webpack-config**

To start the project type

**npx expo start**

# Implementation

## Folder structure

Four our application we tried to keep an ordered structure for the folders of the project. In fact, we decided to divide all in many sub-folders, one for each functionality of the application.

Immagine che contiene testo, schermata, design

Descrizione generata automaticamente

Figure 3 Folders Structure

## Admin

## Auth

In the Auth Folder there are the three GUI (Graphical User Interface) used for the auth part that are, the Login page, the Register page, and the Waiting page. Over these three files there is the last file (AuthUtils) that contains some function needed for check if the registration and the login worked correctly.

Immagine che contiene testo, schermata, Carattere, Blu elettrico

Descrizione generata automaticamente

Figure 4 Auth Files

### Login Page

The login page is simple, there is just the logo f the application, two fields for the email and the password, the button for sign in and the button for go to register a new account.

Immagine che contiene testo, schermata, Cellulare, Dispositivo mobile

Descrizione generata automaticamente

Figure 5 Log in Page

When you click on the “SIGN IN” button the function “handleLogin” from the file AuthUtils. Here, we are going to validate email and password with a trim (that remove space before and after the word)

## Chat

In the chat folder there is the chat page and other component related to the chat functionality.

Immagine che contiene testo, schermata, Carattere, Blu elettrico

Descrizione generata automaticamente

Figure 6 Chat Files

### ChatPage

The ChatPage file contain the list of the existing chat. In the top there is a search bar for searching in the chat that already exists or new user for start a new chat.

Immagine che contiene schermata, testo, design

Descrizione generata automaticamente

Figure 7 Chat Page

### ListResearchChat

When you click on the search bar it opens the ListResearchPage and filter the existing chat and other user by the keyword that you digit in the search bar.

Immagine che contiene testo, schermata

Descrizione generata automaticamente

Figure 8 ListResearchPage

### ChatTemplate

The ChatTemplate is used for render the chat

### Message

## Home

## Models

## Navigations

## Notifications

## Post

## Profile

## Repost

## Search

## Stories