John Michael Jalinon

IT64 (IT36A)

# Introduction

The client-side functionality of the application is built using React and Ionic, while server-side functions like authentication and data storage are managed by Firebase services (Authentication and Firestore). The client and server interact regarding CRUD operations on ToDo items and authentication tasks.

# Features

**User Authentication:**

* **Sign Up:** Users can register for an account by providing their name, email, and password.
* **Login:** Users can log in to their accounts using their email and password.
* **Logout:** Users can log out of their accounts.

**Description of Your Works (Systems)**

# Login System

* Provides a login interface where users can input their email and password.
* Authenticates users using Firebase Authentication.
* Checks if the user exists in Firestore, and redirects to the home page upon successful login.

# Sign-Up System

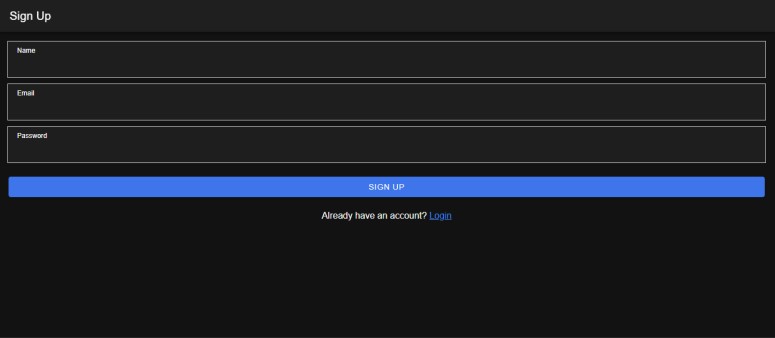
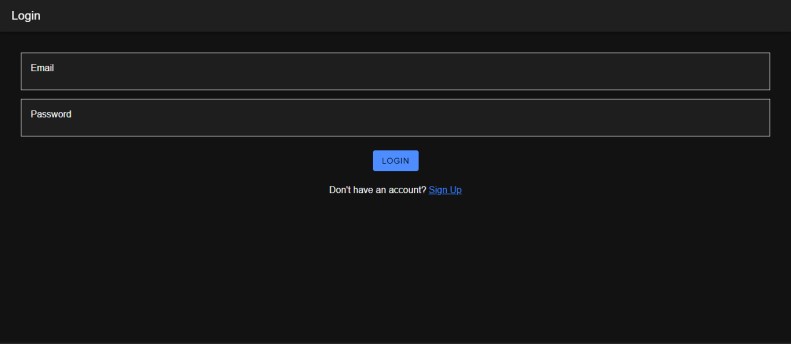
* Provides a sign-up interface where users can register by providing their name, email, and password.
* Validates user inputs and displays appropriate error messages.
* Stores user data in Firestore upon successful sign-up.

# Conceptual Framework

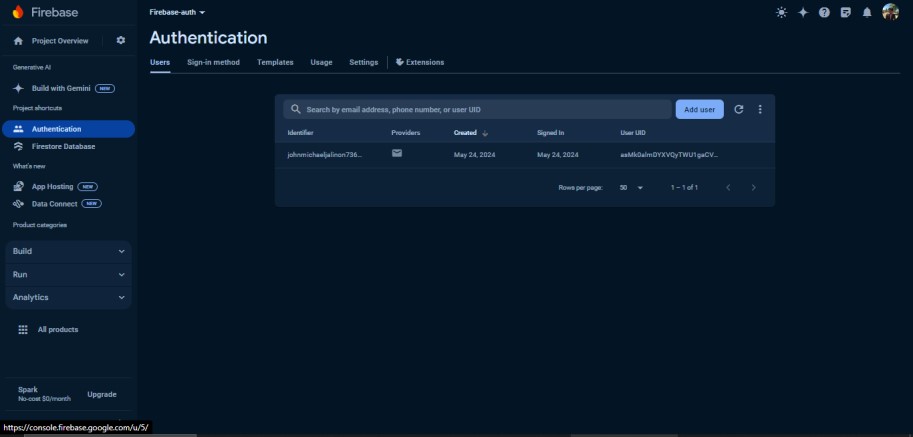
The program is designed using a client-server architecture. React and Ionic are used in the client-side development, and Firebase services (Authentication and Firestore) handle server-side functions like data storage and authentication. To perform CRUD operations on ToDo items and authentication tasks, the client and server communicate.

**The System (Screenshot of your Output)**

# Login Display Screen Sign-Up Display Screen



# Firebase Display Screen



**Reference:**

Jalinon, JM. (2024, May). ToDo List Application. Retrieved from https://github.com/Jalstzy/JALINON-FINALPT