**GitFit**

**Introduzione**

GitFit è un'applicazione Android che vuole fornire ai propri utenti la libertà di organizzare esercizi e gestire le proprie schede di allenamento. L’applicazione presenterà una lista di esercizi già compresi. Nel calendario è possibile segnare in quali giorni ci si è allenati e aggiungere quali esercizi sono stati svolti.

**Scopo e motivazioni**

Lo scopo di questa applicazione è quello di semplificare gli allenamenti, che siano a casa o in palestra. Per fare ciò l’applicazione permette all’utente di creare, modificare e cancellare in maniera semplice liste di esercizi personalizzate e favorisce la creazione di routine che possano soddisfare i bisogni dell’utente. Viene implementato un sistema che supporta l’utente prima, durante e in seguito agli allenamenti, così da permettere a quest’ultimo di consultarli con facilità e analizzarli a piacere.

**Architettura del Sistema**

L’applicazione comunica con il database per mantenere i dati relativi all’utente e comunica con il telefono per quanto riguarda salvare i dati relativi al calendario.

Database: Nella nostra architettura il database gestisce i dati che la nostra applicazione salva ed utilizza al suo interno. Al momento il database viene popolato al momento della registrazione di un utente con i dati che questo utente inserisce e viene consultato al momento del login per connettersi.

App: Il calendario in cui è possibile inserire in locale i giorni in cui ci alleniamo e gli esercizi fatti tramite gli input degli extra.

**Materiale e Tecnologie Utilizzate**

Draw.io: è stato utilizzato per creare i primi veri prototipi dell’app prima di utilizzare figma.

Figma: è stato utilizzato dal nostro gruppo per proseguire sui primi design fatti su draw.io. Tramite questo sito siamo stati in grado di delineare i diversi Wireframe e Storyboard. Abbiamo utilizzato le diverse funzioni di esportazione per potere uploadare facilmente i nostri lavori. E’ stato molto utile inoltre a creare lo schema ER che ci ha permesso di definire le entità della nostra applicazione, i loro attributi e appunto la loro relazione.

Android Studio: questo è il nostro punto di riferimento per quanto riguarda la scrittura del codice per l’applicazione Android e per testare che l’applicazione funzioni correttamente, sia grazie all’emulatore che permette di emulare diversi sistemi operativi e diversi dispositivi ma anche tramite la funzione che permette di collegare il proprio telefono ad Android Studio per testare una applicazione direttamente su un dispositivo fisico.

MySQL: utilizzato per immagazzinare dati relativi al profilo dell’utente

Retrofit e Flask: non pienamente implementati

**Linguaggi Utilizzati**

Data l’ampiezza del progetto, per potere comunicare con tutti i sistemi in luogo avremo bisogno di utilizzare diversi linguaggi, verranno elencati e verrà descritto in che modo li abbiamo utilizzati.

Java: è il nostro linguaggio di programmazione base con il quale scriviamo il codice della nostra applicazione, tramite il codice java la nostra applicazione è in grado di comunicare con l’utente e di svolgere calcoli in background per assicurare al nostro utente la miglior possibile esperienza. Grazie ai servizi internet che Java offre siamo in grado di collegarci facilmente e comunicare con il telefono le nostre necessità. Usando Android Studio siamo anche in grado di definire gli aspetti visivi dell’applicazione e gestire i diversi modi in cui agiscono quando l’utente ci interagisce con Java.

SQL: è il linguaggio utilizzato per la gestione dei dati e per la capacità di analizzarli. E’ molto importante per noi che il codice sia scritto bene per evitare che si presentino dei memory leak e che funzioni tutto correttamente.

HTML e Javascript: utilizziamo questi linguaggi per programmare le pagine web per gli utenti e la pagina del back office che gli admin possono utilizzare. E’ importante assicurarci che l’app web sia in grado di comunicare con l’applicazione e vice versa, inoltre vogliamo utilizzare questi linguaggi per creare delle interfacce che siano coerenti con quelle dell’applicazione Android.

**Struttura dei dati: ER e modello dei dati** Immagine che contiene diagramma, design

Il contenuto generato dall'IA potrebbe non essere corretto.

Lo schema ER esplica le relazioni che le diverse entità hanno tra di loro. In primo luogo, l’utente è in grado di svolgere degli esercizi che verranno contenuti dentro delle schede, i dati di questi esercizi sono associati alla data in cui sono stati eseguiti e salvati sul calendario; grazie a ciò abbiamo i progressi che, tramite un sistema di misurazione degli esercizi a sfinimento dell’utente, sarà in grado di ridare informazioni riguardo la relazione tra le sue performance presenti e le sue performance passate. Quando l’utente avrà dei progressi notevoli essi verranno registrati come obiettivi raggiunti.

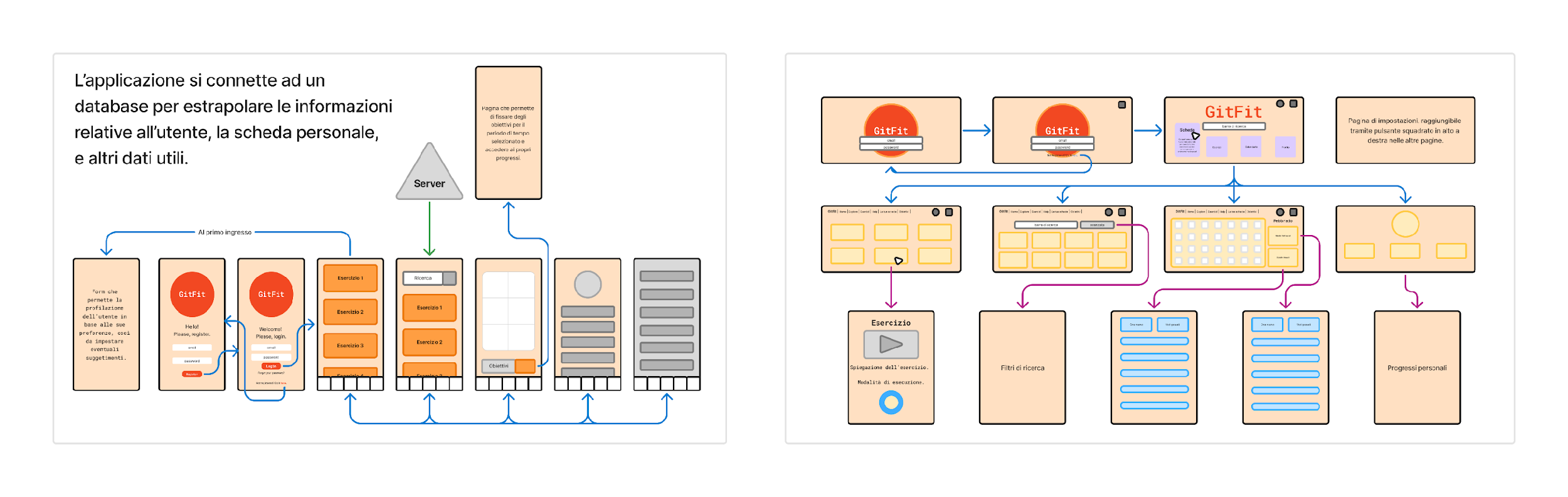
**Wireframe/Storyboard dell’applicazione Android**

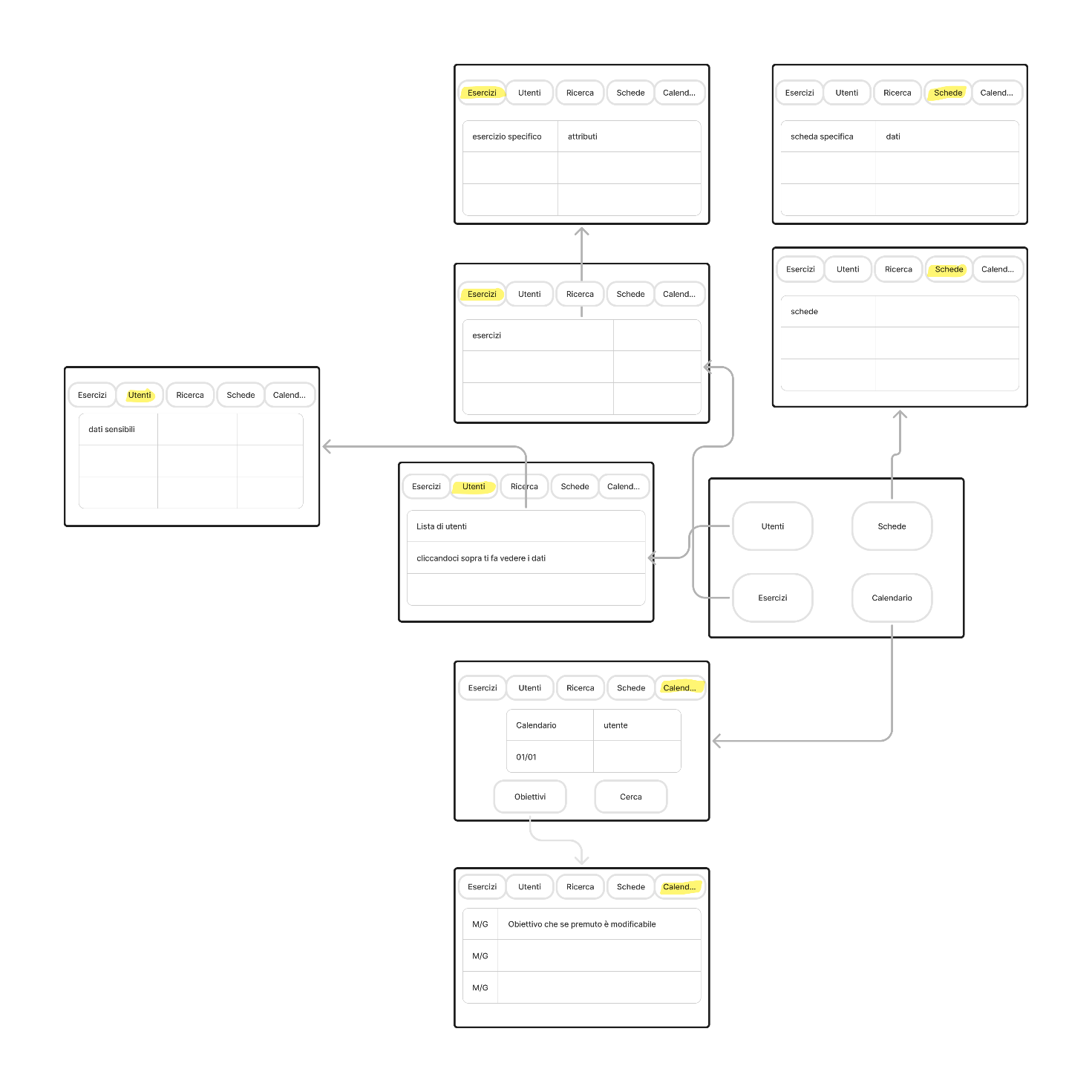
Immagine che contiene testo, schermata, diagramma, Piano

Il contenuto generato dall'IA potrebbe non essere corretto.

Dopo le schermate di login ed eventuale registrazione, l’applicazione si apre sulla schermata principale. Al primo avvio dell’applicazione, in caso non sia stato creato un profilo, l’applicazione fa delle domande per migliorare l’esperienza dell’utente. L’applicazione. E’ possibile accedere in qualsiasi momento al calendario per controllare i progressi che si stanno svolgendo.

**Wireframe/Storyboard del Sito Web**

**Wireframe e Back Office**



**Sviluppi Futuri**

In quanto l’applicazione non è pienamente conclusa questi sono gli sviluppi futuri che avremmo voluto implementare:

* Tramite l’implementazione di Flask e Retrofit permettere a ogni parte dell’applicazione di comunicare con un server per interfacciarsi con un database, permettendo a ogni parte dell’app di autocompilarsi così da rendere più facile il ruolo del backend e rendere più facile all’amministratore inserire nuovi esercizi e schede.
* Al momento i dati dell’applicazione non legati al login vengono salvati in locale, bisogna implementare il salvataggio su database
* Implementare la logica nella scheda così che sia possibile utilizzarla per creare un gruppo di esercizi da svolgere in sequenza.
* Il sito web deve essere sviluppato e collegato al database per permettere l’accesso ai dati anche senza l’utilizzo dell’applicazione mobile.
* Il Back office deve essere implementato per permettere una facile gestione dei dati (schede, esercizi ecc.)