Wemeet

Temptation Javalsland

Davide Trucci	899628
Marco Carpio Herreros	899802
Alessandro Nappi	899650
Kevin Davide Dextre Gonzales	899926

Indice

INTRODUZIONE

WeMeet è un'app Android progettata per aiutarti a scoprire eventi pubblici e connetterti facilmente con nuove persone. Pensata per chi ama socializzare e vivere nuove esperienze, l'app offre diverse funzionalità:

- Esplora eventi in base ai tuoi interessi o cerca eventi specifici per nome.
- Visualizza gli eventi più vicini alla tua posizione
- Salva gli eventi tra i preferiti per non perderne nemmeno uno.
- Accedi al tuo profilo personale e gestisci le impostazioni in modo semplice.

TECNOLOGIE UTILIZZATE

- **Android Studio:** IDE utilizzato per sviluppare applicazioni Android in Java
- **GitHub:** Software utilizzato per gestire la collaborazione tra gli sviluppatori nel-l'implementazione dell'app.
- **Firebase:** è una piattaforma di Google che offre strumenti backend per lo sviluppo di app, come database in tempo reale e autenticazione.
- **Ticketmaster Discovery API:** consente di cercare eventi, artisti, sedi e biglietti in tempo reale. È pensata per integrare facilmente nei propri servizi un sistema di esplorazione eventi, filtrando per posizione, data, genere musicale e altro.
- **Retrofit:** Libreria utilizzata per effettuare le chiamate API.
- **Room:** Libreria utilizzata per implementare il database offline.
- **Glide:** Libreria utilizzata per la gestione delle immagini.
- **Geocoder (Android API):** Utilizzato in locale per ottenere il nome della città a partire dalle coordinate GPS.
- **Google Maps SDK for Android:** Utilizzato per visualizzare la posizione dell'evento su mappa interattiva.
- **FusedLocationProviderClient**: Servizio di Google per ottenere in modo efficiente la posizione dell'utente

ARCHITETTURA

L'applicazione adotta un'architettura basata sul modello MVVM (Model - View - ViewModel), che favorisce la separazione delle responsabilità, la testabilità e la scalabilità del codice.

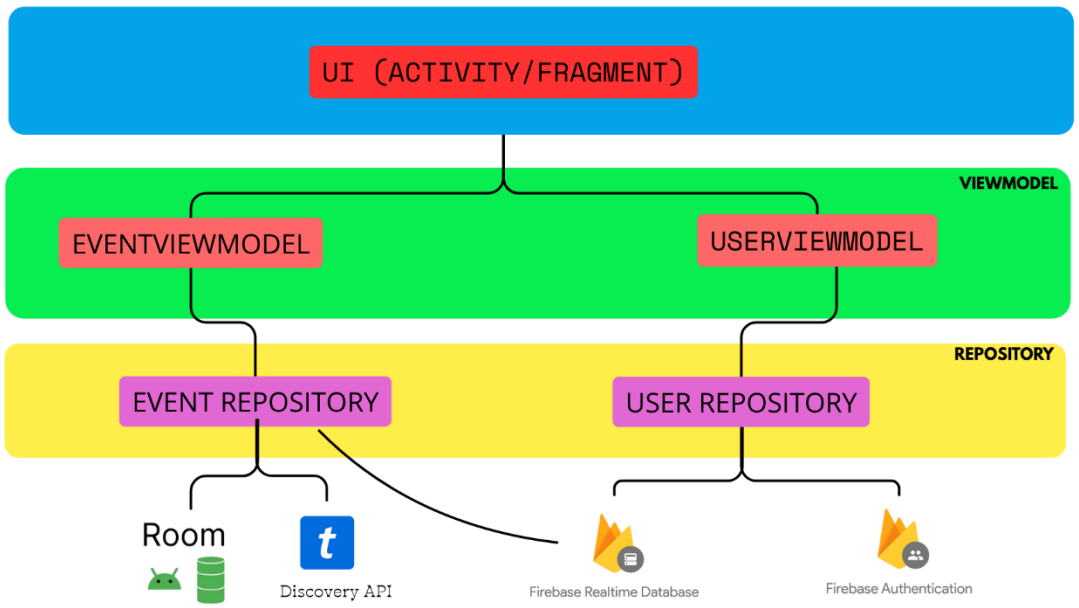
Strati principali dell'architettura:

View (UI): Composta da Activity e Fragment (es. HomePageActivity, LoginFragment, EventPageFragment), rappresenta l'interfaccia utente. La View osserva i dati esposti dai ViewModel e reagisce ai cambiamenti per aggiornare la UI.

ViewModel: Contiene la logica di presentazione. Classi come EventViewModel e UserViewModel gestiscono l'interazione tra la View e i Repository, esponendo i dati sotto forma di LiveData o State.

Repository: Funziona da intermediario tra ViewModel e sorgenti dati. I repository (EventRepository, UserRepository) decidono se recuperare i dati dalla rete o dal database locale, nascondendo la complessità di queste operazioni al ViewModel.

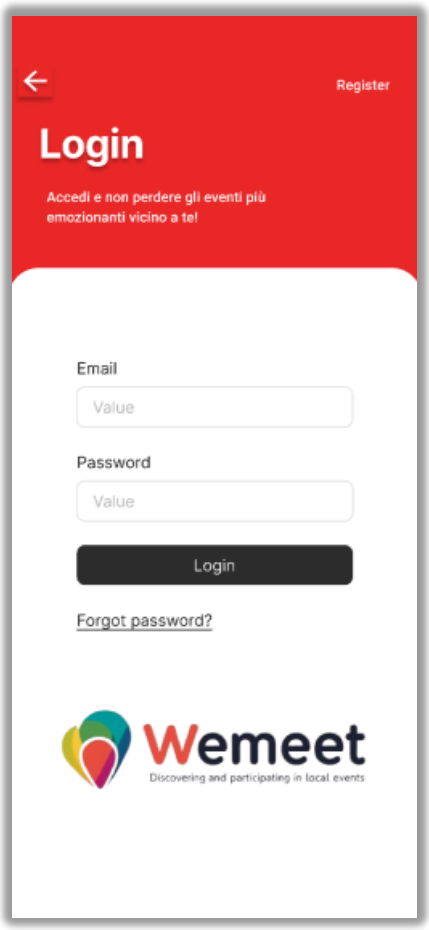
Data Sources (Local e Remote): Strato che si occupa di gestire l'accesso ai dati da fonti locali o remote.



DESIGN

PROGETTAZIONE UI

Per la progettazione dell'interfaccia utente ci siamo affidati a Figma, uno strumento online molto utilizzato per la progettazione grafica e la prototipazione di interfacce. Questo tool ci ha permesso di realizzare prototipi interattivi delle principali schermate dell'app, fornendo così una guida visiva chiara in modo da rendere lo sviluppo definitivo su Android Studio più semplice.



La tua posizione

Bicocca, MI

Gate Milano

Via valtellina 21,Milano
Sabato,30 novembre

Gate Milano

Via valtellina 21,Milano
Sabato,30 novembre

I tuoi eventi

All | Discoteca | Concerti | Svago | Sport

Partecipi a 4 eventi

Duomo dal 1952
Piazza Sempione, Milano
Sabato, 30 Novembre

Duomo dal 1952
Piazza Sempione, Milano
Sabato, 30 Novembre

Duomo dal 1952
Piazza Sempione, Milano
Sabato, 30 Novembre

Duomo dal 1952
Piazza Sempione, Milano

MATERIAL DESIGN

L’app è stata sviluppata ispirandosi ai principi di **Material Design 3**, cercando di mantenere coerenza visiva e usabilità. Sono stati utilizzati e adattati i **Material Components**, come pulsanti e card, per garantire un’interfaccia moderna e intuitiva. Inoltre, grazie alla palette cromatica offerta da Material Design, abbiamo introdotto due modalità grafiche: **tema chiaro** e **tema scuro**, offrendo all’utente un’esperienza visiva personalizzabile.

SCHERMATE E FUNZIONALITÀ

AUTENTICAZIONE

Queste schermate consentono all'utente di autenticarsi tramite tre modalità: login classico, registrazione con e-mail e password e accesso rapido con l'account Google. L'obiettivo è offrire un'esperienza di accesso semplice e flessibile, adattabile alle preferenze dell'utente. Inoltre, l'integrazione con Firebase Authentication garantisce una gestione sicura e affidabile delle credenziali. L'interfaccia è progettata per essere chiara e intuitiva, guidando l'utente in ogni fase dell'accesso all'app.


11:03
Login
Accedi e non perdere gli eventi più emozionanti vicino a te!
E-mail
Password
La password deve contenere almeno 8 caratteri
Accedi
or
Login with Google

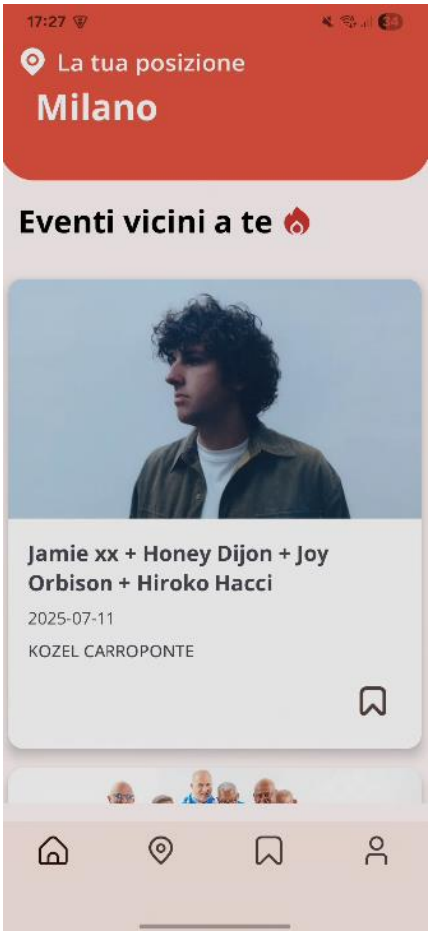

11:05
Crea account
Registrati per non perdere nessun evento!
E-mail
example@email.com
Password
Mario!1234
La password deve contenere almeno 8 caratteri
Conferma password
Mario!1234
Registrati

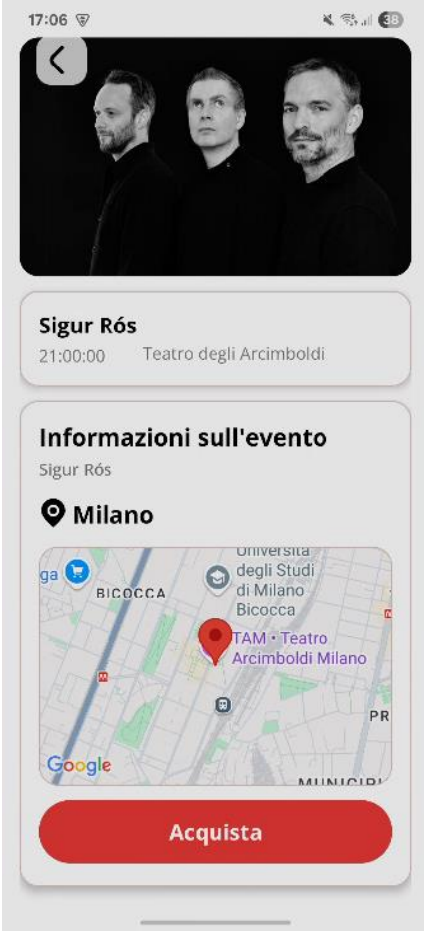
HOMEPAGE

La schermata Home mostra all'utente gli eventi più vicini, basandosi sulla posizione attuale. Se l'utente non concede il permesso di accesso alla localizzazione, per comodità viene impostata come predefinita la città di Milano.
Oltre agli eventi suggeriti, è possibile esplorare liberamente l'intero catalogo e toccando un evento, l'utente può visualizzare una scheda dettagliata contenente:

- Luogo dell'evento
- Immagine di copertina
- Descrizione
- Data e ora

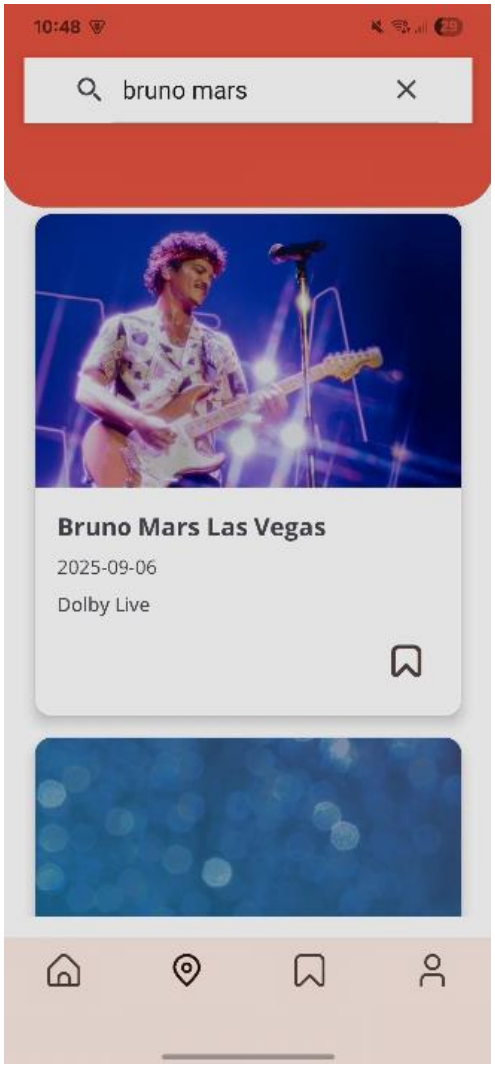
Inoltre, dalla Home è possibile salvare tra i preferiti gli eventi di maggiore interesse, in modo da ritrovarli facilmente in un secondo momento.
L'interfaccia è pensata per essere intuitiva e immediata, rendendo la scoperta di nuovi eventi semplice e coinvolgente.





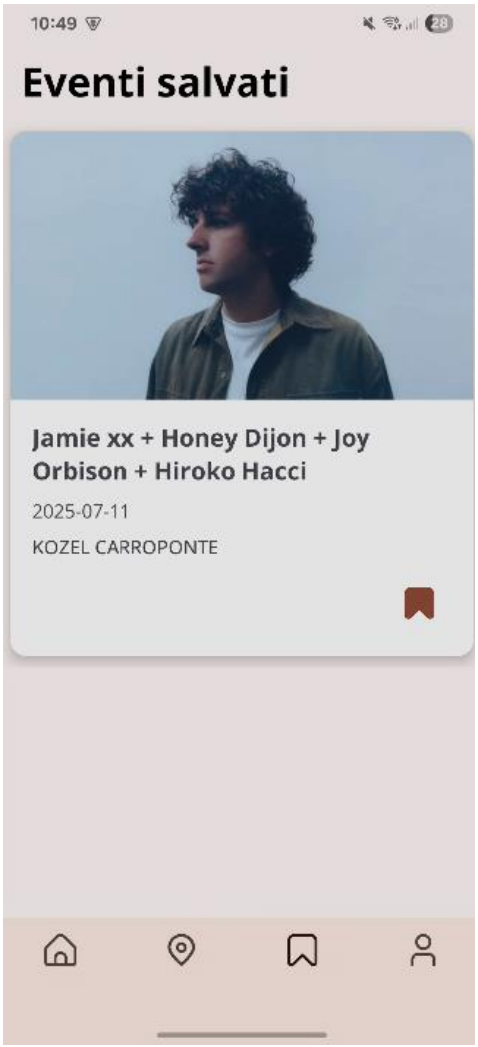
RICERCA EVENTI

Questa schermata offre all’utente la possibilità di cercare eventi di interesse utilizzando parole chiave personalizzate. Grazie a questa funzione, è semplice trovare eventi specifici in base ad artisti, località o keyword. Una volta visualizzati i risultati, l’utente può cliccare un evento per accedere alla pagina dettagliata dell’evento, proprio come in Home. Inoltre, sempre come nella Home, è possibile salvare gli eventi preferiti.



PREFERITI

In questa schermata l’utente può visualizzare tutti gli eventi che ha salvato come preferiti. Se un evento non è più di interesse, è possibile rimuoverlo facilmente dalla lista. La schermata si aggiorna immediatamente andando a rimuovere l’evento dai preferiti.



IMPOSTAZIONI PROFILO UTENTE

In questa schermata l'utente può visualizzare l'e-mail con cui si è loggato. Inoltre, può impostare la modalità notte e giorno, fare il logout oppure decidere di eliminare il profilo permanentemente.

ACTIVITY E FRAGMENT

Di seguito è riportata la lista delle activity e dei fragment che compongono l'applicazione.

WELCOME

WelcomeActivity

La WelcomeActivity rappresenta la schermata di benvenuto dell'applicazione e viene visualizzata al primo avvio dell'app. In particolare è composta dalla WelcomeFragment, LoginFragment e SignUpFragment.

Operazioni effettuate:

- Viene abilitata la modalità EdgeToEdge per una visualizzazione a schermo intero, adattando l'interfaccia ai bordi del dispositivo.
- Viene impostato il layout grafico tramite activity_welcome.xml.

WelcomeFragment

Il WelcomeFragment è la schermata introduttiva dell'applicazione, mostrata all'avvio per permettere all'utente di scegliere tra l'accesso o la registrazione.

Operazioni effettuate:

- Se l'utente risulta già autenticato tramite Firebase Authentication, viene automaticamente reindirizzato alla HomePageActivity, saltando la schermata di benvenuto.
- In caso contrario, vengono visualizzati due pulsanti:
    - Accedi: porta l'utente al LoginFragment per effettuare l'accesso.
    - Registrati: reindirizza l'utente al SignUpFragment per creare un nuovo account.

SignUpFragment

Il SignUpFragment gestisce la registrazione di un nuovo utente all'interno dell'applicazione, includendo validazioni, creazione account e feedback per l'utente.

Operazioni effettuate:

- Imposta limiti di lunghezza sui campi input per nome, cognome, e-mail e password.
- Gestisce il comportamento dei pulsanti:
    - Torna indietro: riporta al WelcomeFragment.
    - Registrati: esegue il flusso di registrazione:
        - Verifica che tutti i campi siano validi (nome, cognome, e-mail con formato corretto, password ≥ 8 caratteri, corrispondenza tra password e conferma).
        - In caso di validazione corretta, tenta la registrazione tramite il UserViewModel, che comunica con Firebase Authentication.
        - Gestisce gli errori più comuni con Snackbar (password debole, utente già registrato, errori generici).
        - In caso di successo, naviga verso HomePageActivity.

Validazioni specifiche:

- E-mail: controllata tramite EmailValidator della libreria Apache Commons.
- Password: lunghezza minima di 8 caratteri e corrispondenza tra i due campi.

LoginFragment

Il LoginFragment gestisce l'autenticazione degli utenti tramite e-mail/password o tramite Google One Tap, fornendo validazioni e feedback per migliorare l'esperienza utente.

Funzionalità principali:

- Inizializzazione:
    - Configura il client Google One Tap per il login con Google.
    - Prepara l'ActivityResultLauncher per gestire il risultato dell'intento di accesso Google.
- Login automatico:
    - Se l'utente è già autenticato (getLoggedUser() restituisce un valore), viene reindirizzato automaticamente alla HomePageActivity.
- Login via e-mail/password:

    - Verifica che e-mail e password rispettino i criteri minimi (e-mail valida, password ≥ 8 caratteri).
    - In caso di successo, effettua il login tramite Firebase Authentication.
    - In caso di fallimento (e-mail errata o password errata), mostra uno Snackbar con messaggio d'errore.
- Login via Google:
    - Avvia il flusso Google One Tap con l'intento preparato in signInRequest.
    - Se l'utente seleziona un account e il login ha successo:
        - Autentica l'utente in Firebase usando le credenziali Google.
        - Reindirizza alla HomePageActivity.
    - Se fallisce, mostra errore tramite Snackbar.

Validazioni:

- E-mail: verificata con EmailValidator di Apache Commons.
- Password: lunghezza minima 8 caratteri.

Navigazione:

- Pulsante Torna indietro: ritorna al WelcomeFragment.
- Pulsante Login: avvia login classico con Firebase.
- Pulsante Login con Google: avvia login tramite Google One Tap.

MAIN

HomePageActivity

La HomePageActivity rappresenta l'attività principale dell'app dopo il login, dalla quale l'utente può accedere alle diverse sezioni tramite la barra di navigazione.
In particolare, è composta da HomeFragment, LocationFragment, PreferedFragment, ProfileFragment, EventPageFragment e MapsFragment.
Operazioni effettuate:

- Viene abilitata la modalità EdgeToEdge per una UI a schermo intero.
- Viene caricato il layout activity_home_page.xml, che include un NavHostFragment per la navigazione tra fragment e una BottomNavigationView per il passaggio tra le schermate principali dell'app.
- Viene configurato il NavController per gestire la navigazione tra i fragment: HomeFragment, LocationFragment, PreferedFragment e ProfileFragment.

- Viene inizializzato il servizio di localizzazione tramite FusedLocationProviderClient, con richiesta dinamica del permesso di accesso alla posizione:
    - Se il permesso è concesso, viene recuperata la posizione attuale dell'utente.
    - Latitudine e longitudine ottenute vengono passate al homeFragment sotto forma di Bundle.
    - In caso di rifiuto del permesso, viene mostrato un messaggio all'utente.

HomeFragment

La HomeFragment consente all'utente di visualizzare gli eventi disponibili nella propria area.

Funzionalità principali:

- Visualizzazione eventi: Una volta ottenuta la posizione dell'utente, l'app effettua una chiamata API per recuperare gli eventi vicini e li mostra in una lista.
- Preferiti: L'utente può aggiungere o rimuovere eventi dai preferiti cliccando sull'icona dedicata. Le modifiche vengono salvate nel backend associato al profilo dell'utente.
- Dettagli evento: Cliccando su un evento, l'utente viene reindirizzato alla pagina dei dettagli corrispondente.
- Localizzazione: Il fragment converte automaticamente le coordinate geografiche in un nome di città leggibile, mostrato nell'interfaccia.
- Gestione connessione: Se non è disponibile una connessione a Internet, viene mostrato un messaggio d'errore al posto della lista degli eventi.

LocationFragment

Il LocationFragment consente all'utente di cercare eventi in base a una parola chiave, artista o località, con un'interfaccia focalizzata sulla ricerca.

Funzionalità principali:

- Ricerca eventi: L'utente può inserire una parola chiave nella barra di ricerca per trovare gli eventi desiderati. I risultati vengono aggiornati in tempo reale durante la digitazione.
- Preferiti: Ogni evento nei risultati può essere aggiunto o rimosso dai preferiti. Le modifiche vengono salvate localmente e nel backend associato all'utente.
- Dettagli evento: Cliccando su un evento, si accede alla relativa pagina dei dettagli.

- Stato della vista: Quando non viene effettuata alcuna ricerca, viene mostrato uno stato "vuoto".
- Persistenza preferiti: Gli eventi già salvati come preferiti vengono riconosciuti anche nei risultati di ricerca grazie a un confronto tra ID.

PreferedFragment

Il PreferedFragment mostra la lista degli eventi salvati dall'utente. Questa sezione funge da raccolta personalizzata e sincronizzata tra locale e remoto.

Funzionalità principali:

- Visualizzazione eventi preferiti: Viene inizialmente popolata con gli eventi salvati localmente, e aggiornata automaticamente quando i dati remoti vengono recuperati dal backend.
- Navigazione ai dettagli: Selezionare un evento porta alla pagina con i dettagli completi.
- Rimozione dai preferiti: L'utente può togliere un evento dai preferiti. La rimozione è immediata sia a livello locale che remoto.
- Sincronizzazione locale e remoto:
    - I preferiti remoti vengono scaricati dal backend e salvati nel database locale.
    - Ogni evento remoto sincronizzato aggiorna anche lo stato locale.
- Persistenza automatica: Tutte le modifiche sono persistite utilizzando Room e aggiornate in tempo reale nella UI attraverso LiveData.

ProfileFragment

Il ProfileFragment è la schermata dell'utente dedicata alla gestione del profilo personale, con funzionalità che includono visualizzazione dell'e-mail, logout, cambio tema (Dark/Light Mode) ed eliminazione dell'account utente.

Funzionalità principali:

- Visualizzazione utente:
    - Mostra l'indirizzo e-mail dell'utente autenticato.
    - Supporta autenticazione sia con e-mail/password sia tramite Google.
- Cambio tema (Dark/Light Mode):
    - Mostra lo stato corrente (Tema: Giorno/Notte).
    - Consente di invertire il tema dell'app dinamicamente.

- Logout:
    - Mostra un dialogo di conferma.
    - Pulisce i dati locali, esegue il logout da Firebase e ritorna alla WelcomeActivity.
- Eliminazione profilo:
    - Chiede conferma prima della cancellazione.
    - In base al provider (google o password):
        - Riautenticazione con Google Sign-In.
        - Riautenticazione tramite password utente.
    - Dopo la riautenticazione:
        - Elimina i dati utente dal database.
        - Rimuove l'account Firebase e torna alla WelcomeActivity.

EventPageFragment

EventPageFragment è la schermata dedicata alla visualizzazione dettagliata di un evento selezionato. Mostra immagini, data, luogo, descrizione, mappa e consente anche l'accesso diretto alla pagina d'acquisto dei biglietti.

Funzionalità principali:

- Visualizzazione evento completo:
    - Nome, data, descrizione e immagine dell'evento.
    - Luogo e città.
    - Mappa interattiva con marker e zoom automatico sulla location dell'evento.
- Integrazione con Google Maps (MapView):
    - Viene mostrata una mappa centrata sulla location dell'evento.
    - Coordinate ottenute dalla chiamata API.
- Acquisto biglietti:
    - Se disponibile, viene aperto il browser con il link dell'evento.
    - In caso contrario, viene mostrato un Toast con un messaggio di errore.

MapsFragment

Il MapsFragment è una vista che incorpora Google Maps per visualizzare una posizione sulla mappa.

Funzionalità principali:

- Integrazione con Google Maps:
    - Utilizza un MapView per visualizzare la mappa direttamente nel Fragment.
- Gestione ciclo di vita della mappa:

- Implementa correttamente i metodi onResume, onPause, onDestroy e onLowMemory per mantenere la mappa sincronizzata con il ciclo di vita del Fragment.

OFFLINE-FIRST

L’applicazione implementa un approccio **Offline First** per la gestione degli eventi. Quando non è disponibile una connessione Internet, recupera e mostra gli eventi salvati nel database locale del dispositivo, permettendo all’utente di consultare e interagire con essi anche offline. Se non ci sono dati locali, viene mostrato un messaggio che segnala l’assenza di connessione e di eventi disponibili. Inoltre, eventuali modifiche come il salvataggio di eventi durante la modalità offline vengono registrate localmente e sincronizzate automaticamente con il database remoto una volta ripristinata la connessione. Quando la connessione è presente, l’app acquisisce la posizione GPS dell’utente (o usa una posizione di default se non disponibile) e carica gli eventi aggiornati da un servizio remoto, mostrando la lista aggiornata all’utente. Inoltre, l’utente può salvare o rimuovere eventi preferiti, con sincronizzazione che avviene sia localmente che su Firebase.

In questo modo, l’esperienza utente rimane fluida e continua, sfruttando prima i dati locali e aggiornandoli automaticamente quando possibile.

SVILUPPI FUTURI

Per migliorare ulteriormente l'esperienza utente e ampliare le funzionalità dell'applicazione, sono previste alcune evoluzioni future che arricchiranno l'interazione e la sicurezza all'interno dell'app. Di seguito sono elencate le principali funzionalità in fase di valutazione per le prossime versioni:

- **Autenticazione a due fattori (2FA):** Per aumentare il livello di sicurezza degli account utente, si prevede l'integrazione di un sistema di autenticazione a due fattori. Questo permetterà di proteggere maggiormente i dati personali e le

attività dell'utente, richiedendo un secondo livello di verifica oltre alla password.

- **Visualizzazione dei partecipanti agli eventi:** Una futura implementazione includerà la possibilità per gli utenti di visualizzare la lista dei partecipanti a un evento. Ciò favorirà una maggiore trasparenza e potrà contribuire a creare una community più coesa.
- **Contatto tra partecipanti:** Collegata alla funzionalità precedente, sarà introdotta la possibilità di mettersi in contatto con altri partecipanti agli eventi (nel rispetto della privacy e con il consenso degli utenti). Questa funzione ha lo scopo di facilitare la socializzazione e la creazione di nuovi legami, sia prima che dopo l'evento.
- **Supporto multilingua (inglese):** Per ampliare la base utenti e rendere l'app accessibile anche a persone non italiane, si prevede l'aggiunta del supporto alla lingua inglese. Gli utenti potranno selezionare la lingua dell'interfaccia secondo le proprie preferenze.
- **Possibilità di acquistare i biglietti in-app:** In futuro vogliamo permettere agli utenti di comprare i biglietti per gli eventi direttamente dall'app, senza dover uscire o usare altri siti. Questo renderà tutto più semplice e veloce, migliorando l'esperienza e facilitando la partecipazione agli eventi.

Questi sviluppi rappresentano un passo importante verso la creazione di un'app più completa, sicura e inclusiva, in grado di soddisfare esigenze sempre più ampie degli utenti.