

Il ruolo dei LLM nello sviluppo di applicazioni mobile



Prova finale di Santin Simone – Matricola 886116

Relatore: Prof.ssa Micucci Daniela

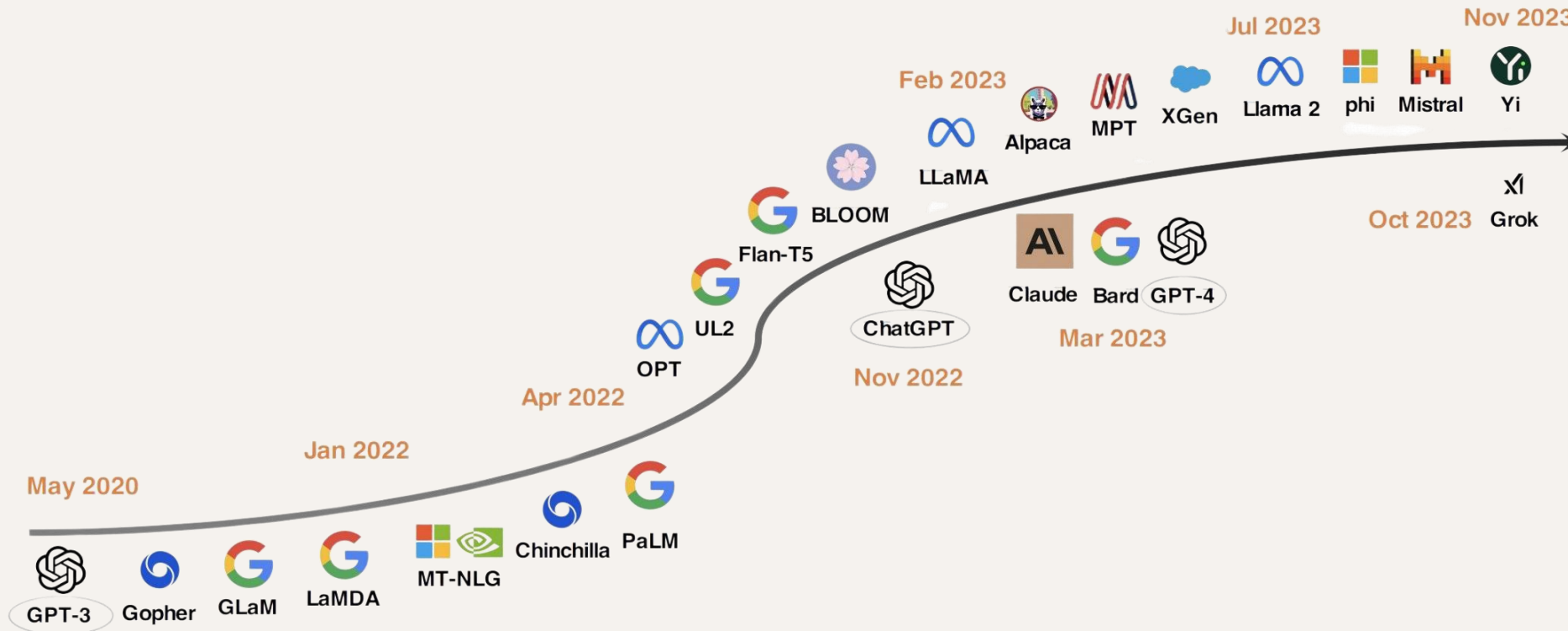
Co-Relatore: Dott.ssa Rossi Maria Teresa

Anno accademico: 2023-2024

Introduzione

Negli ultimi anni, gli LLM hanno conosciuto un'ampia diffusione in svariati contesti. Sono utilizzati in chatbot e assistenti virtuali, traduttori automatici e per il supporto nello sviluppo software.

Esistono molteplici strumenti basati su LLM, ciascuno progettato per specifici obiettivi. Noi, in quanto informatici, ci concentriamo su quelli che supportano il progettista.



Obiettivo

Verificare l'utilità degli LLM nello sviluppo mobile attraverso:

- Sviluppo di un'applicazione nel campo ambientale
- Utilizzo di un LLM durante tutte le fasi del ciclo di sviluppo
- Registrazione di ogni scambio di messaggi in un diario
- Analisi dei benefici e delle limitazioni riscontrati nell'uso dell'LLM



Flutter



EcoSwap

Requisiti dell'applicazione

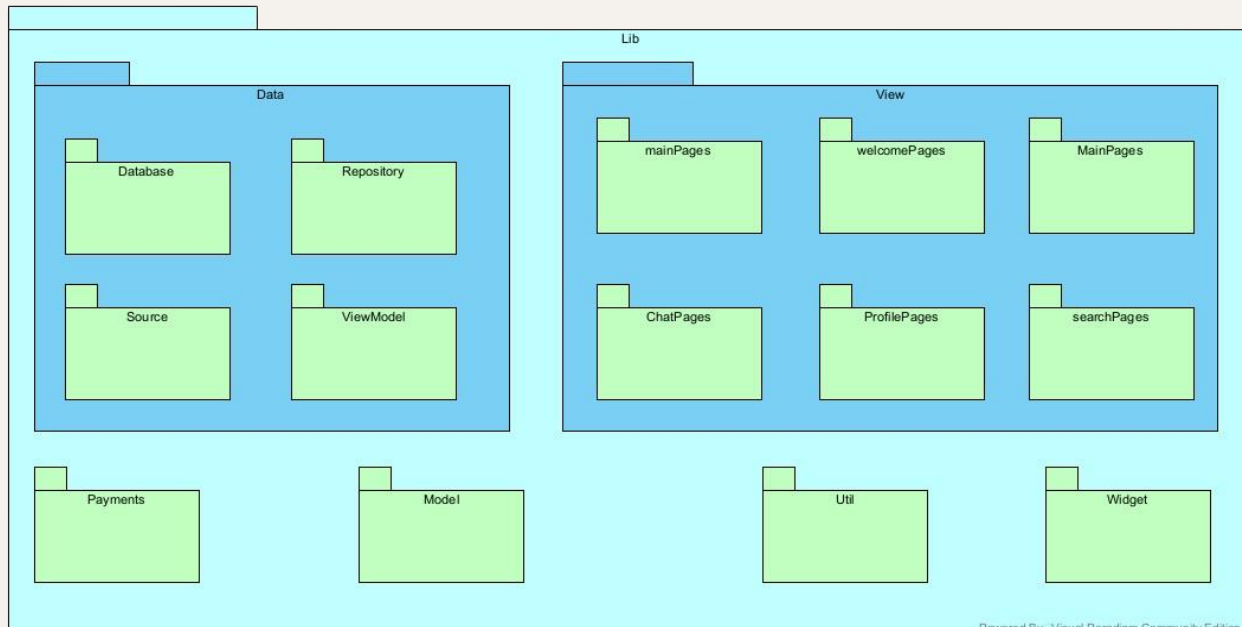
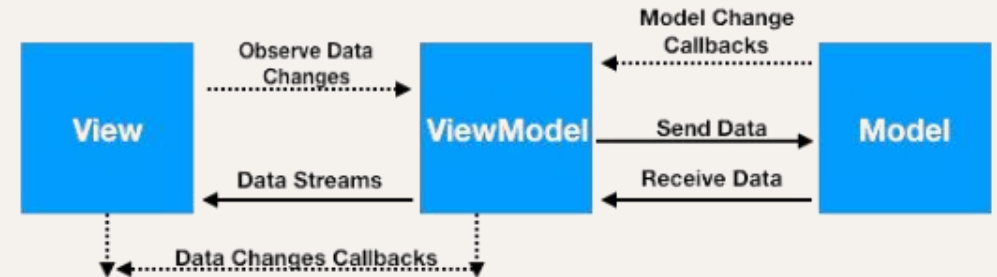
Principali requisiti dell'applicazione:

- Registrazione e autenticazione degli utenti
- Gestione dati personali
- Pubblicazione di annunci
- Gestione annunci
- Funzionalità di ricerca
- Avvio di noleggi
- Implementazione di una chat tra utenti
- Feed utente
- Recensioni e valutazioni



Architettura dell'applicazione

L'architettura dell'applicazione è basata sull'architettura Model-View-ViewModel in modo da mantenere una chiara separazione delle responsabilità.



Package principali:

- **Data:** accesso e manipolazione dei dati
- **View:** pagine dell'applicazione
- **Model:** strutture dei dati
- **Payments:** funzionalità di pagamento
- **Util:** utility comuni
- **Widget:** componenti UI

Testing

Strumenti e framework utilizzati:

- Flutter test Framework
- Mockito
- Firebase Auth Mocks

Test effettuati:

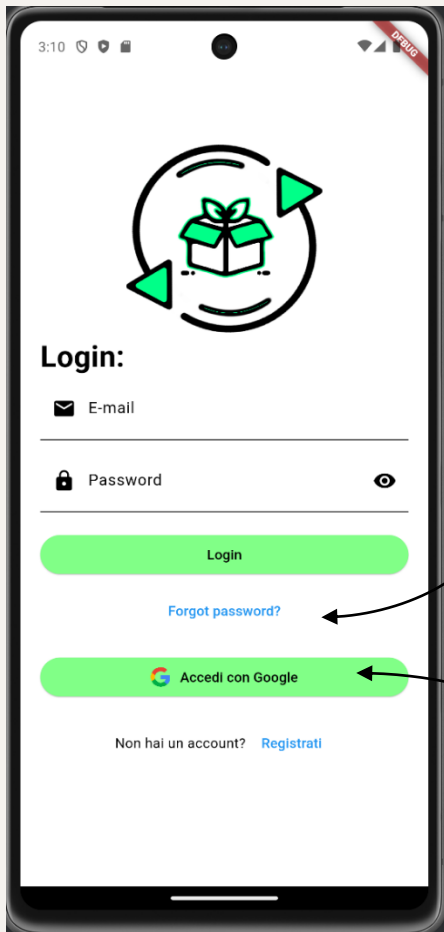
- Test di registrazione
- Test di login

```
test('Login with valid credential', () async {  
  when(mockFirebaseAuth.signInWithEmailAndPassword(  
    email: "1@2.com",  
    password: "123456",  
  )).thenAnswer((_) async => userCredential);  
  final result = await userAuthDataSource.login(  
    email: '1@2.com',  
    password: '123456',  
  );  
  expect(result, 'Success');  
});
```

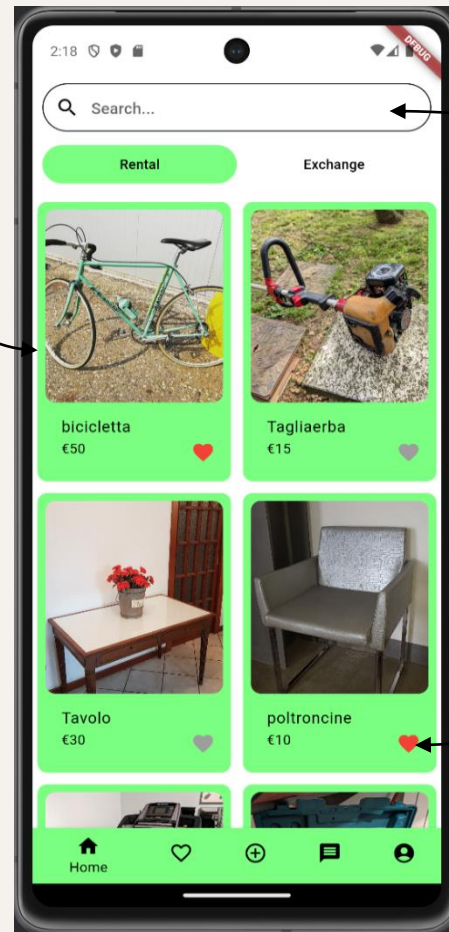
```
test('Login with wrong credential', () async {  
  when(mockFirebaseAuth.signInWithEmailAndPassword(  
    email: "invalid@example.com",  
    password: "invalidPassword",  
  )).thenThrow(FirebaseAuthException(code: 'wrong-password'));  
  final result = await userAuthDataSource.login(  
    email: 'invalid@example.com',  
    password: 'invalidPassword',  
  );  
  expect(result, 'Wrong password provided for that user.');
```

Risultato

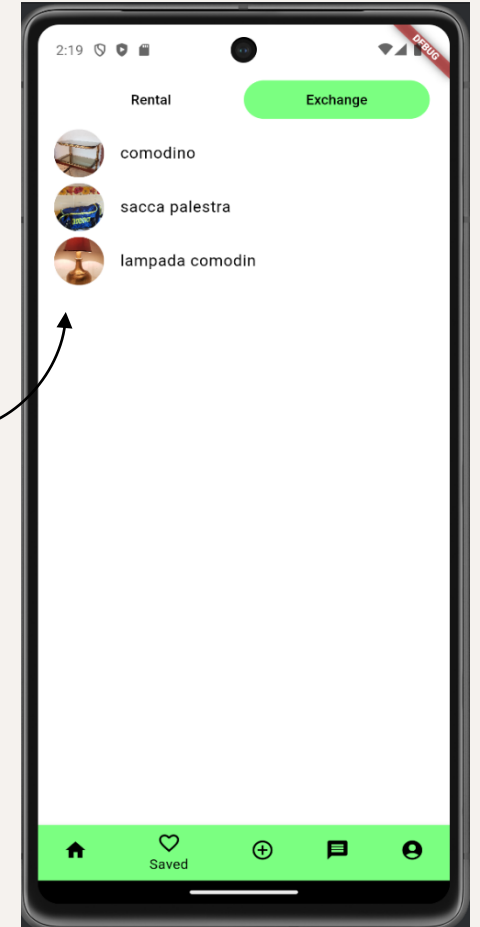
Login Page



Home Page



Favourites Page



Articoli per distanza

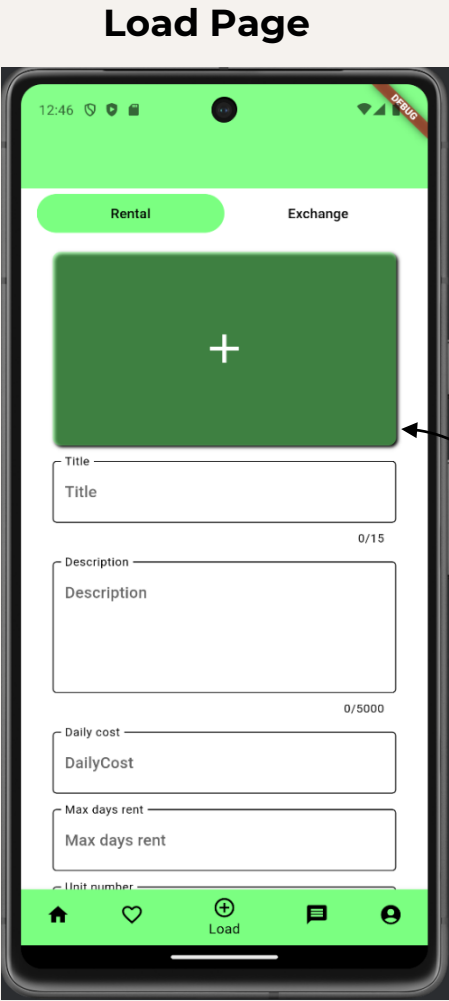
Gestione password

Accesso con google

Barra di ricerca

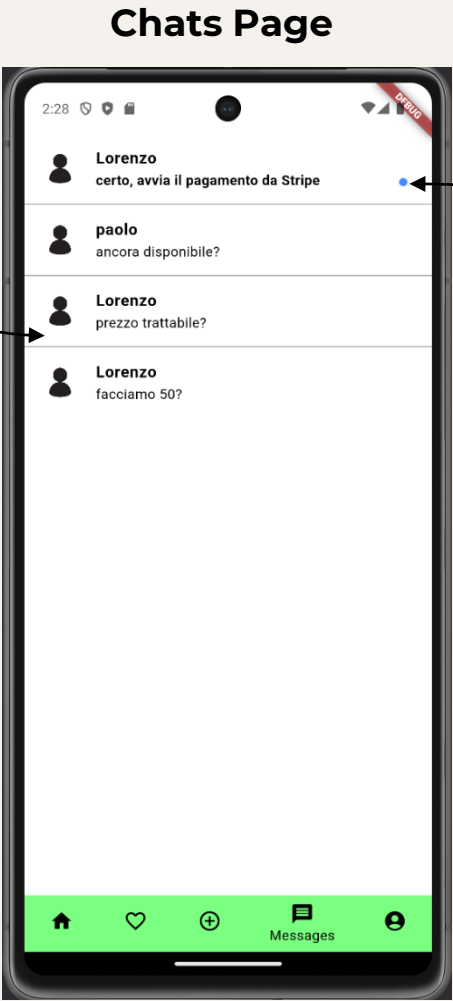
Articoli salvati

Risultato



Chat per articolo

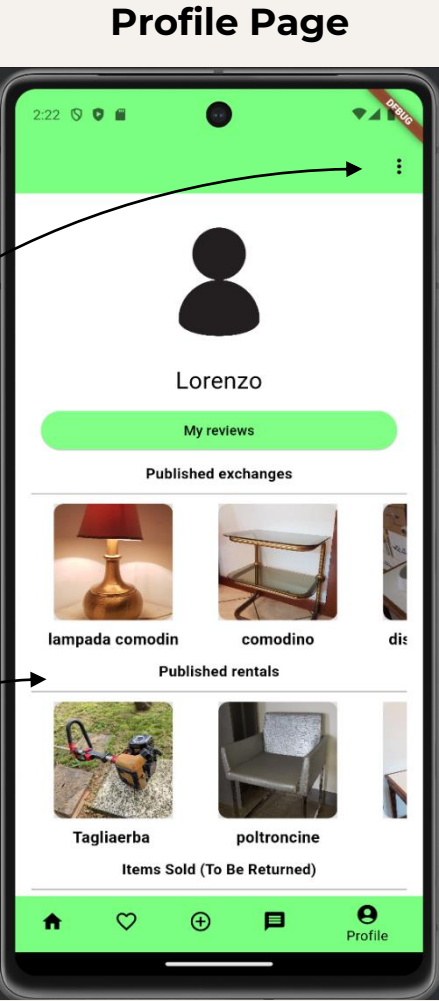
Selettore di immagini



Messaggio non letto

Gestione profilo

Gestione articoli



Analisi del supporto di ChatGPT

Fase	Benefici	Limitazioni
Analisi dei requisiti	<ul style="list-style-type: none">Suggerimenti utili per definire funzionalitàIdentificazione dei requisiti	<ul style="list-style-type: none">Necessità di supervisioneCapacità di innovazione
Implementazione	<ul style="list-style-type: none">Ottimizzazione del codiceRisoluzione di erroriGenerazione di codice ripetitivoCreazione rapida dello scheletro delle pagine	<ul style="list-style-type: none">Errori nelle soluzioni proposteRisoluzione di errori complessiDesign delle pagine carente
Testing	<ul style="list-style-type: none">Identificazione dei test di baseDocumentazione per testStrutturazione dei casi di test	<ul style="list-style-type: none">Comprensione dei test avanzatiErrori nell'implementazione di test

Considerazioni finali

Vantaggi dell'integrazione di ChatGPT:

- Riduzione dei tempi di sviluppo
- Supporto nella risoluzione di problemi comuni
- Supporto creativo per requisiti e design
- Consolidamento delle best practice di sviluppo
- Maggiore conoscenza del framework Flutter

In conclusione, l'integrazione di ChatGPT nel progetto ha significativamente migliorato l'efficienza e la produttività, ottimizzando il flusso di lavoro e potenziando le competenze tecniche.

