CORSO MOBILE PROGRAMMING 2024/2025 04/06/2025

Gruppo 04:
ADINOLFI GIOVANNI 0612708352
DI CRESCENZO FRANCESCO 0612708640
DI MARINO DOMENICO 0612707421
SCANDONE ALESSANDRO 0612707955



OBIETTIVI DELL'APPLICAZIONE

Schermata principale

Elenco aggiornato delle serie in corso

Schermata dettaglio serie

Visualizzazione dettagliata di ogni serie (trama, episodi, stato, piattaforma)

Schermata aggiunta/modifica serie
Inserimento e modifica delle informazioni sulle serie

Schermata categorie

Visualizzazione per genere o piattaforma, categorie personalizzabili

Schermata ricerca

Ricerca per titolo, genere o piattaforma

Schermata analisi

Statistiche: serie seguite, completate, in pausa

(iii) Vincoli tecnici:

L'app deve essere sviluppata utilizzando

React Native o Flutter

ARCHITETTURA GENERALE - DOMFLIX



1. Stack tecnologico

Flutter (Dart)

SQLite (via sqflite) per storage locale UI responsive, tema dark personalizzato

2. Componenti principali

- SplashScreen: Avvio app, mostra logo
- ★ Home: Sezioni (preferiti, in corso, suggeriti, ecc.), navigazione
- SeriesScreen: Dettaglio serie, stato, stagioni/episodi
- **SearchScreen**: Ricerca avanzata (testo, filtri)
- **CategoriesScreen**: Serie per genere/ piattaforma (anche custom)
- **AnalyticsScreen**: Statistiche avanzate
- Add/Edit/SeasonEpisode: Gestione serie stagioni, episodi
- **CustomFooter**: Navigazione principale persistente

```
Future<void> _onCreate(Database db, int vers
  // Tabella principale delle serie
  await db.execute('''
    CREATE TABLE series(
      id INTEGER PRIMARY KEY AUTOINCREMENT,
      imageUrl TEXT NOT NULL,
      title TEXT NOT NULL,
      trama TEXT NOT NULL,
      genere TEXT NOT NULL,
      stato TEXT NOT NULL,
      piattaforma TEXT NOT NULL,
      isFavorite INTEGER NOT NULL DEFAULT 0,
      seasons TEXT NOT NULL DEFAULT '[]',
      dateAdded INTEGER,
      dateCompleted INTEGER
```

Creazione tabella principale nel DataBase



ARCHITETTURA **GENERALE**

3. Persistenza & Modello Dati

- DatabaseHelper:
 - Singleton
 - CRŬD Serie
 - Tabelle: serie, categorie personalizzate, statistiche

Model:

- Series → Seasons → Episodes
 Campi: stato, date, preferito, ecc.

Initial Data:

• Popolamento automatico all'avvio

4. State Management

- setState nei principali StatefulWidget
 Aggiornamento automatico UI dopo modifiche

5. Extra

Immagini locali/remoti (copertine serie)
Palette custom Domflix UI adattiva (responsive)

```
void main() => runApp(const LetterboxdApp());
class LetterboxdApp extends StatelessWidget {
 const LetterboxdApp({Key? key}) : super(key: key);
 @override
  Widget build(BuildContext context) {
    return MaterialApp(
      title: 'Domflix',
      debugShowCheckedModeBanner: false,
      theme: ThemeData.dark().copyWith(
        scaffoldBackgroundColor: □ const Color(0xFF181c2)
        appBarTheme: const AppBarTheme(
          color: ■Color(0xFFB71C1C),
         elevation: 0,
          surfaceTintColor: □Colors.transparent,
        ), // AppBarTheme
      initialRoute: SplashScreen.routeName, // Cambia qu
      routes: {
       SplashScreen.routeName: (context) => const Splash
       DomflixHomePage.routeName: (context) => const Do
```

Navigazione e struttura iniziale dell'app

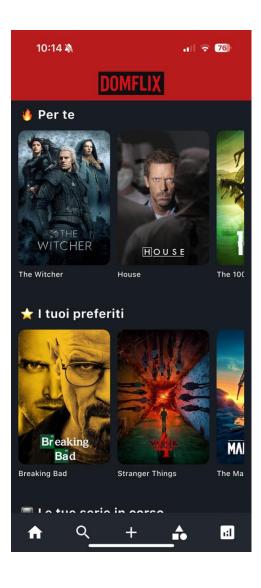
10:14 🦄



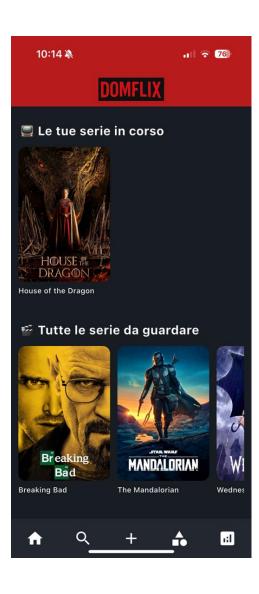
DOMFLIX

FUNZIONALITA' PRINCIPALI DOMFLIX

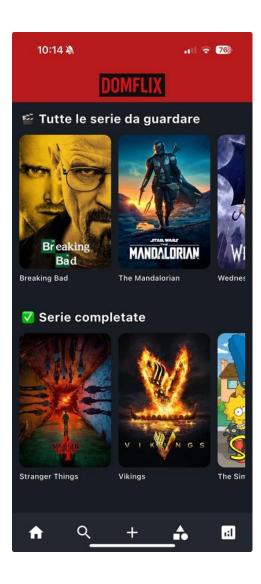
Schermata di avvio



Homepage (serie consigliate per l'utente e sezione preferiti)



Homepage (Serie in corso e serie da guardare)



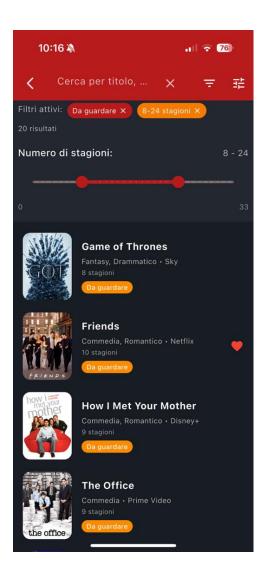
Homepage (Serie completate)



Aggiunta nuova serie



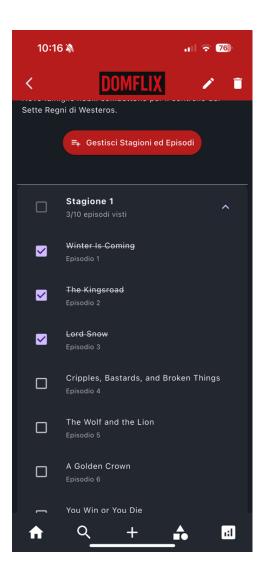
Gestione stagioni ed episodi



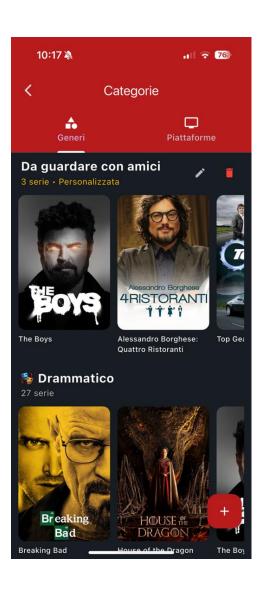
Ricerca con filtri per stato e numero di stagioni



Dettaglio serie e aggiunta ai preferiti

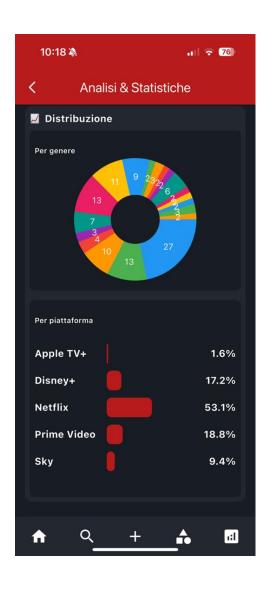


Dettaglio stagioni ed episodi

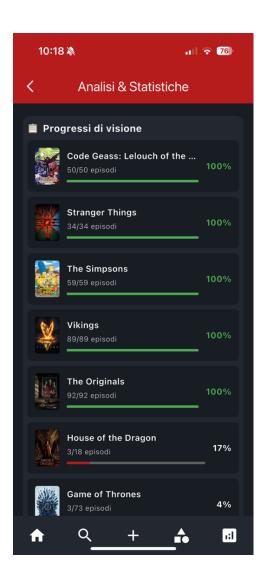


Gestione categorie e personalizzazione









Flutter (Dart)	Sviluppo rapido, UI moderna, cross-platform
SQLite (sqflite)	Dati offline, veloce, affidabile
path_provider & path	Gestione percorsi, immagini locali
e collection	Utility avanzate per liste
🦚 Material Design	UI nativa, accessibile, personalizzabile

SCELTE PROGETTUALI - DOMFLIX

Tecnologie e librerie utilizzate

SCELTE PROGETTUALI

Motivazioni delle scelte

- Flutter: sviluppo rapido, UI personalizzabile, community ampia, deploy facile Android/iOS
- **SQLite**: storage locale leggero, nessun cloud necessario, perfetto per app offline
- **path_provider / path**: gestione immagini e dati locali dell'utente
- collection: operazioni avanzate su liste (utile per stagioni/episodi
- Material Design: UI professionale, responsive, adattabile al tema "Domflix"
- Gestione locale: tutte le funzionalità sono disponibili offline, nessun server esterno
- **Struttura modulare**: schermate indipendenti, manutenzione e aggiunte future semplificate

```
class DatabaseHelper {
   static final DatabaseHelper instance = DatabaseHelper._internal();
   DatabaseHelper._internal();

Future<int> insertSeries(Series series) async {
    final db = await database;
    return await db.insert('series', series.toMap());
   }
}
```

Esempio pratico di SQLite

CONCLUSIONI



Gestione Serie TV: aggiunta, modifica, avanzamento episodi/stagioni



Esperienza Utente: interfaccia intuitiva, navigazione veloce, tema "Domflix"



Personalizzazione: categorie custom, preferiti, filtri avanzati



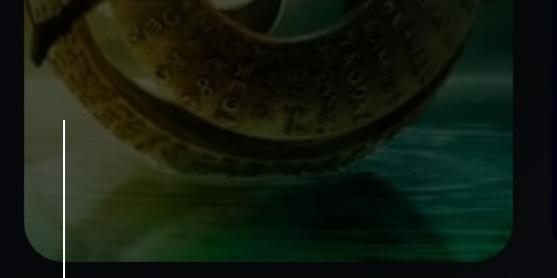
Statistiche: analisi andamento, suggerimenti smart, grafici



Robustezza: dati offline, nessuna dipendenza server, affidabilità



Scalabilità: struttura modulare, facile estensione







Sense8

Brooklyn Nine-Nine

Peaky Blinders

I tuoi preferiti







