

CarGram - Documentatie Tehnica

1. Prezentarea Aplicatiei

Scopul si Obiectivul

CarGram este o aplicatie social media specializata pentru pasionatii de masini, inspirata din Instagram, dar adaptata specific pentru comunitatea auto. Aplicatia permite utilizatorilor sa impartaseasca poze cu masinile lor, sa interactioneze cu alti pasionati si sa gestioneze profiluri detaliate pentru vehiculele lor.

Functionalitati Principale

- **Autentificare si Inregistrare:** Sistem complet de autentificare cu Firebase Auth
- **Profil de Vehicul:** Crearea si gestionarea profilurilor detaliate pentru masini
- **Decodare VIN:** Integrare cu API extern pentru decodarea automata a VIN-urilor
- **Feed Social:** Vizualizarea si interactiunea cu postarile altor utilizatori
- **Crearea Postarilor:** Incarcarea de imagini cu descrieri și hashtag-uri
- **Sistem de Like-uri si Favorite:** Interactiune sociala cu continutul
- **Profil de Utilizator:** Gestionarea informatiilor personale si a postarilor
- **Navigare Intuitiva:** Interfata moderna cu navigare prin tab-uri

2. Specificatii Tehnice

Versiuni și Compatibilitate

- **Android API Level:** 24+ (Android 7.0 Nougat)
- **Target SDK:** 35 (Android 15)
- **Compile SDK:** 35
- **Java Version:** 17
- **Kotlin Version:** 1.9.0+

Framework-uri si Librarii

UI si Navigation

- **Jetpack Compose:** UI toolkit modern pentru Android
- **Navigation Compose:** Navigare intre ecrane (v2.7.7)

- **Material 3:** Design system modern

Arhitectura si Dependency Injection

- **Dagger Hilt:** Dependency injection (v2.51)
- **ViewModel:** Gestionarea starii UI (v2.7.0)
- **Coroutines:** Programare asincrona (v1.7.3)

Rețea si API

- **Retrofit:** Client HTTP pentru API-uri (v2.9.0)
- **OkHttp:** Client HTTP cu logging (v4.12.0)
- **Gson:** Serializare JSON

Baza de Date

- **Room Database:** Bază de date locala (v2.6.1)
- **Type Converters:** Conversii pentru tipuri complexe

Firebase (In Dezvoltare)

- **Firebase Auth:** Autentificare utilizatori
- **Firestore:** Baza de date cloud
- **Firebase Storage:** Stocarea fisierelor

Utilitare

- **Coil:** Incarcarea imaginilor (v2.4.0)
- **Extended Icons:** Iconite suplimentare Material Design

3. Arhitectura Aplicatiei

Pattern-uri Arhitecturale

- **MVVM (Model-View-ViewModel):** Separarea logicii de business de UI
- **Repository Pattern:** Abstraierea surselor de date
- **Dependency Injection:** Gestionarea dependentelor cu Dagger Hilt
- **Clean Architecture:** Separarea in straturi (data, domain, presentation)

📁 CarGram/

```
└─ 📁 app/
|
| └─ 📁 src/main/java/com/proiect/cargram/
|
| | └─ 📁 data/
|
| | | └─ 📁 api/      # API interfaces si responses
|
| | | └─ 📁 local/    # Room database si DAOs
|
| | | └─ 📁 model/    # Entitati si modele de date
|
| | | └─ 📁 repository/ # Implementari repository
|
| | └─ 📁 di/         # Dagger Hilt modules
|
| | └─ 📁 ui/
|
| | | └─ 📁 components/ # Componente reutilizabile
|
| | | └─ 📁 navigation/ # Navigation si routing
|
| | | └─ 📁 screens/    # Ecrane principale
|
| | | └─ 📁 theme/     # Tema si stilizare
|
| | | └─ 📁 viewmodel/ # ViewModels pentru fiecare ecran
|
| | └─ MainActivity.kt # Punctul de intrare
|
| └─ 📁 res/          # Resurse (imagini, string-uri, etc.)
```

4. Baza de Date și Relatii

Entitati Principale

Entitate	Descriere	Campuri Principale
User	Utilizatorii aplicatiei	id, username, email, profilePicturePath
Vehicle	Profilurile vehiculelor	vin, userId, make, model, year, engine, fuelType, etc.
Post	Postarile utilizatorilor	Id, userId, username, imagePath, caption, timestamp, likes, comments, shares, vehicleId
FavoritePost	Postarile favorite ale utilizatorilor	userId, postId

Relatii intre Entitati

- **User ↔ Vehicle:** One-to-Many (un utilizator poate avea mai multe vehicule)
- **User ↔ Post:** One-to-Many (un utilizator poate avea mai multe postari)
- **Vehicle ↔ Post:** One-to-Many (un vehicul poate fi prezent in mai multe postari)
- **User ↔ FavoritePost:** Many-to-Many (utilizatorii pot avea postari favorite)

API Integrations

VIN Decoder API

- **Endpoint:** {apiKey}/{controlSum}/decode/{vin}.json
- **Metodă:** GET
- **Scop:** Decodarea automata a VIN-urilor pentru obtinerea informatiilor despre vehicul
- **Raspuns:** Informatii complete despre vehicul (marca, modelul, anul, motorul, etc.)

5. Functionalitati Principale

Autentificare si Inregistrare

- **Login Screen:** Autentificare cu email si parola
- **Register Screen:** Inregistrare cu validare campuri
- **Firebase Auth:** Gestionarea sesiunilor si securitatea
- **Persistenta Sesiunii:** Utilizatorul ramane conectat intre sesiuni

Profil de Vehicul

- **Decodare VIN:** Introducerea VIN-ului pentru decodare automata
- **Introducere Manuala:** Completarea manuala a informatiilor despre vehicul
- **Campuri Detaliate:** Marca, modelul, anul, motorul, tipul de combustibil, etc.
- **Validare:** Verificarea completitudinii informatiilor inainte de salvare

Feed Social

- **Postari Dinamice:** Afisarea postarilor in format card
- **Sistem de Like-uri:** Interactiune cu postarile
- **Favorite:** Salvarea postarilor preferate

- **Sortare:** Optiuni de sortare a postarilor
- **Tab-uri:** "For You" si "Favorites"
- **Loading States:** Indicatori de incarcare si gestionarea erorilor

Crearea Postarilor

- **Incarcarea Imaginilor:** Selectarea si incarcarea fotografiilor
- **Descrieri:** Adaugarea de text si hashtag-uri
- **Asocierea cu Vehiculul:** Legarea postarii de un vehicul specific
- **Preview:** Vizualizarea postarii inainte de publicare

Profil de Utilizator

- **Informatii Personale:** Editarea username-ului si email-ului
- **Poza de Profil:** Incarcarea si gestionarea imaginii de profil
- **Postarile Mele:** Vizualizarea propriilor postari
- **Vehiculele Mele:** Gestionarea profilurilor vehiculelor

6. Implementari Tehnice Avansate

Dependency Injection cu Dagger Hilt

- **@AndroidEntryPoint:** Anotarea activitatilor pentru DI
- **@Inject:** Injectarea dependentelor in clase
- **@Module:** Definirea modurilor pentru Firebase, Room, si Network
- **@Provides:** Furnizarea instantelor pentru injectare

State Management cu ViewModel

- **UI State:** Gestionarea starii UI prin sealed classes
- **Data Flow:** Fluxul de date unidirectional
- **Coroutines:** Operatii asincrone pentru API calls si database
- **Error Handling:** Gestionarea erorilor si loading states

Navigation cu Compose Navigation

- **Type-Safe Navigation:** Navigare cu argumente tipizate
- **Deep Linking:** Suport pentru deep links
- **Navigation State:** Gestionarea starii de navigare

- **Back Stack Management:** Gestionarea istoricului de navigare

Room Database cu Coroutines

- **@Dao:** Data Access Objects pentru operatii CRUD
- **@Query:** Queries SQL optimizate
- **@TypeConverters:** Conversii pentru tipuri complexe (Timestamp, List)
- **Flow:** Observarea schimbarilor in baza de date

7. Dezvoltari Viitoare

Functionalitati Planificate

Firebase Integration Completa

- Sincronizarea datelor cu Firestore
- Stocarea imaginilor in Firebase Storage
- Notificari push cu Firebase Cloud Messaging

Sistem de Comentarii

- Adaugarea comentariilor la postari
- Raspunsuri la comentarii
- Notificari pentru comentarii

Functionalitati Sociale Avansate

- Sistem de follow/unfollow
- Feed personalizat bazat pe preferinte
- Hashtag-uri si trending topics

Functionalitati pentru Vehicule

- Istoricul de intretinere
- Modificari si upgrade-uri
- Compararea vehiculelor

Functionalitati de Cautare

- Cautare utilizatori
- Cautare vehicule
- Filtrare dupa criterii multiple

Functionalitati de Mesagerie

- Chat privat intre utilizatori
- Grupuri de discutii
- Partajarea de imagini in chat

Functionalitati de Geografie

- Localizarea utilizatorilor
- Gasirea pasionatilor din apropiere
- Evenimente auto locale

Functionalitati de Analytics

- Statistici pentru postari
- Analytics pentru utilizatori
- Rapoarte de performanta

Imbunatatiri Tehnice

- **Performance:** Optimizarea incarcarii imaginilor si caching
- **Offline Support:** Functionarea offline cu sincronizare
- **Testing:** Unit tests, integration tests, si UI tests
- **CI/CD:** Pipeline de build si deployment automat
- **Security:** Implementarea de masuri de securitate suplimentare
- **Accessibility:** Imbunatatirea accesibilitatii pentru utilizatori cu dizabilitati

8. Concluzie

CarGram reprezinta o aplicatie social media moderna si inovatoare, specializata pentru comunitatea auto. Cu o arhitectura solida bazata pe cele mai recente tehnologii Android, aplicatia ofera o experienta de utilizare fluida si intuitiva. Utilizand Jetpack Compose pentru UI, Dagger Hilt pentru dependency injection, Room pentru baza de date locala, si Firebase pentru serviciile cloud, CarGram este pregatita pentru scalabilitate si dezvoltare continua. Viitoarele dezvoltari vor transforma CarGram intr-o platforma completa pentru pasionatii de masini, oferind functionalitati sociale avansate, sisteme de mesagerie, si integrare completa cu serviciile Firebase.