

# NÁVRH ŘEŠENÍ

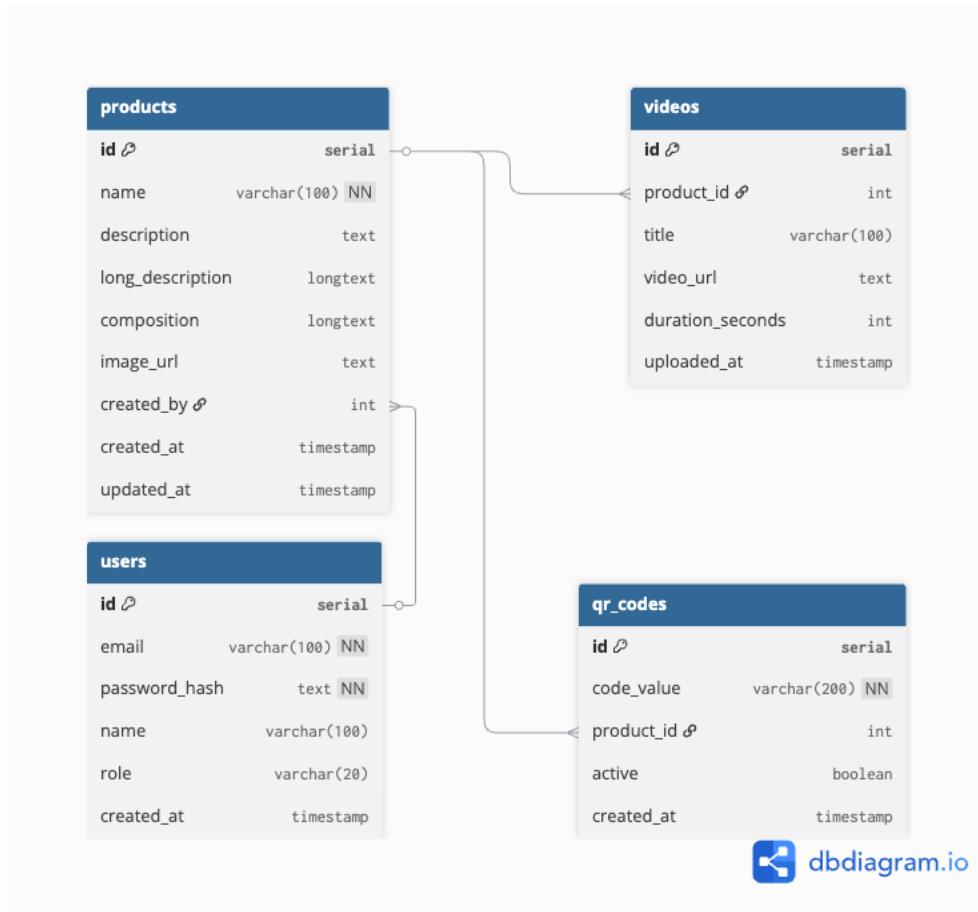
**Autor:** Matěj Straka

**Název práce:** Mobilní aplikace pro prezentaci produktů pomocí QR kódů

**Oponent:** Mgr. Karel Jindra

**Datum zpracování návrhu:** 27. 10. 2025

## Popis modelu dat



Model vytvořen pomocí služby DBdiagram.io

# **Popis klíčových procesů v systému**

## **A. Skenování QR a získání dat (mobilní klient)**

- Kód: integrace skenovací knihovny ve Flutteru: mobile\_scanner
- Proces: kamera → detekce QR → volání API GET /api/qr/{code} → server vrátí JSON s produktem, videem a složením → klient zahájí přehrání.
- Náročné body: rychlosť detekcie, odolnosť vo špatnom osvetlení, Šriešení: používať osvedčenou knihovnu, vizuálna pomoc, umožniť zapnúť svítidlo

## **B. Přehrávání videa (mobilní klient)**

- Kód: napojenie na stabilný prehrávač ve Flutteru: video\_player
- Proces: načtení video\_url → prehrávanie → po konci videa sa spustí UI pro Složenie/Více informací.

## **C. Backend – Serverpod API a databázové operace**

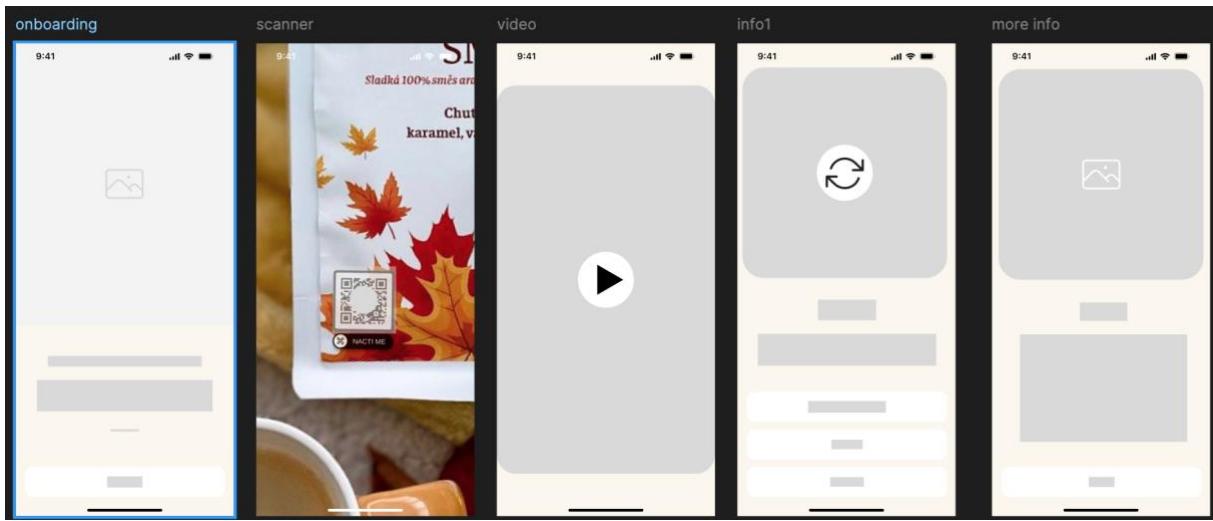
- Kód: Serverpod generuje API a modely; implementace endpointov: GET /api/qr/{code}, CRUD pre products, videos, compositions, users, qr\_codes
- Proces: ověření tokenu (admin časti), validace vstupu, nahrávání metadat videí, zpracování požadavků adminu.
- Náročné body: bezpečnosť (auth, role-based access), validace uploadu videí

## **D. Admin rozhraní (web) – CRUD a správa uživatelů**

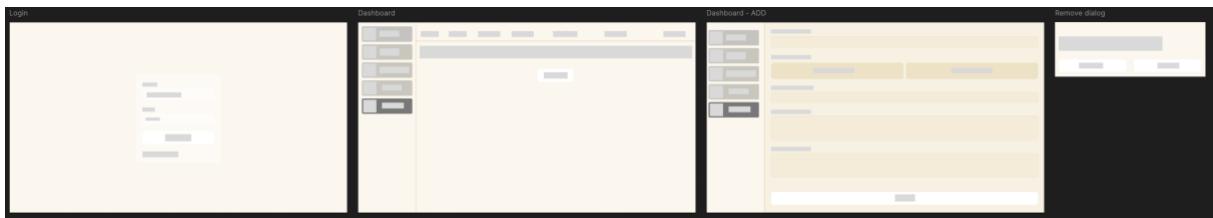
- Kód: Flutter Web (sdílený kód) nebo samostatný web ve Flutteru; formuláre pro produkty, nahrávaní/odkazování videí, generování QR záznamů, správa rolí.
  - Náročné body: UX pre ne-technický personál, bezpečné nahrávanie súborov, správa práv.
-

# Wireframe

- Aplikace



- Webu

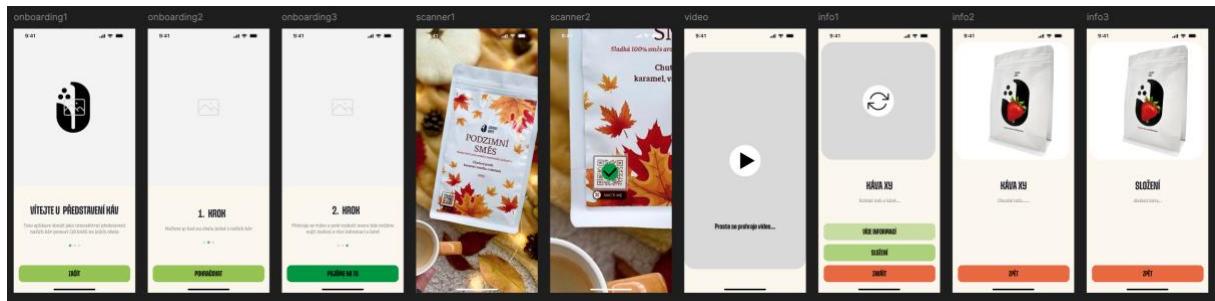


Wireframe vytvořen v aplikaci Figma

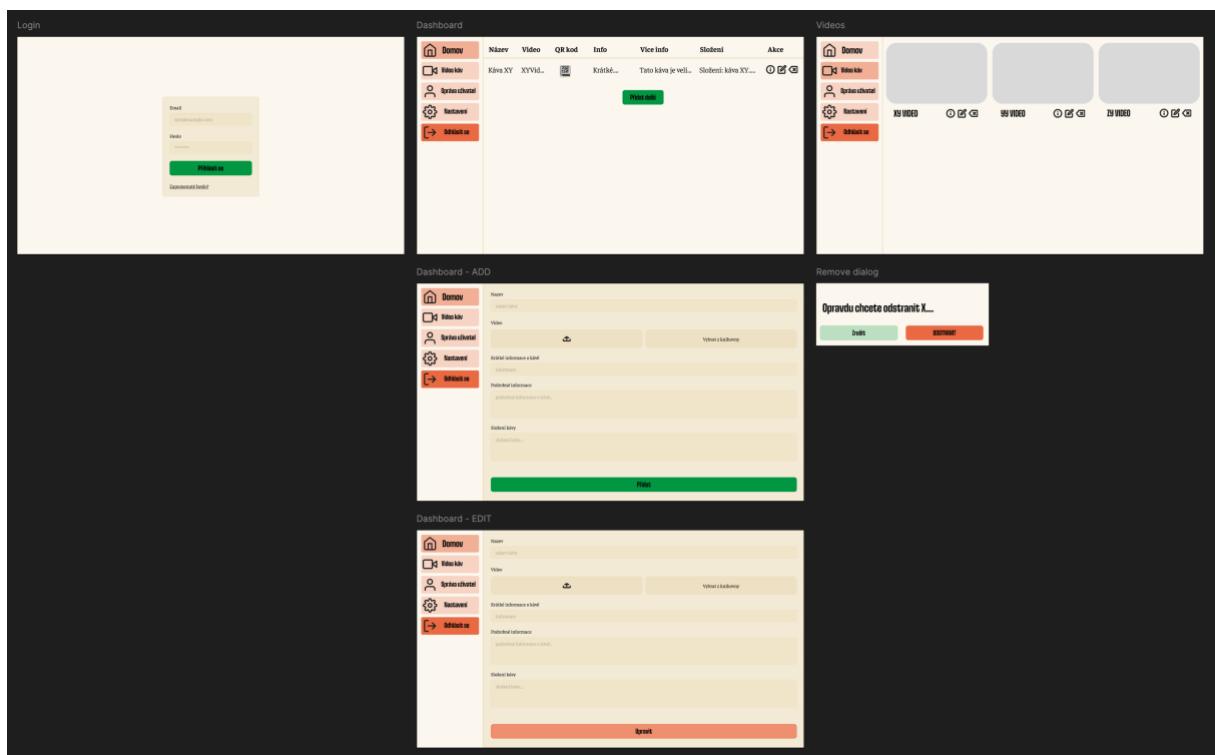
---

# Grafický návrh

- Aplikace



- Webu



Návrh vytvořen v aplikaci Figma

## Použité nástroje / frameworky / řešení

### A. Flutter (klient – mobil a web admin)

- Proč zvoleno: multiplatformní vývoj z jedné codebase, dobrá podpora UI, bohatý ekosystém pluginů (kamera, video). Umožní sdílet část logiky/UI mezi mobilem a webem.
- Varianty zvažované: React Native, nativní vývoj (Kotlin/Swift).
- Výhody Flutter: jednotný kód pro Android/iOS/Web, rychlý prototyp, stabilní výkon.
- Nevýhody Flutter: větší velikost výsledné aplikace, menší množství hotových nativních balíčků než u nativu.

### B. Serverpod (backend v Dartu)

- Proč zvoleno: přímá integrace s DART/Flutter ekosystémem, generování modelů a API, podpora PostgreSQL. Umožní sdílení modelů mezi klientem a serverem a urychlí vývoj.
- Varianty zvažované: Node.js (Express), Django REST, Firebase/Supabase.
- Výhody Serverpod: jeden jazyk pro celý stack, snadné API generování, dobrá integrace s Flutterem.
- Nevýhody Serverpod: menší komunita, méně hotových pluginů/integrací.

### C. PostgreSQL (databáze)

- Proč zvoleno: relační DB s transakčními vlastnostmi, oficiálně podporovaná Serverpodem.
- Varianty zvažované: MySQL, Firestore (NoSQL).
- Výhody Postgres: silná konzistence, SQL, indexování, podpora pro analytické dotazy.
- Nevýhody: administrace serveru, zálohování.

---

## Odkaz na video

<https://youtube.com/shorts/UlWSDVfdw3s>