

Muhammed Kaplan

mmuhammedkaplan0@gmail.com | muhammedkaplan.com | github.com/muhammedkpln

Profil

Självlärd mjukvaruutvecklare med över 4 års erfarenhet som specialiserat sig på full-stack-utveckling. Kunnig i att skapa responsiva webb- och mobilapplikationer med hjälp av moderna ramverk och tekniker. Demonstrerad förmåga att optimera prestanda, implementera robust tillståndshantering och leverera användarvänliga lösningar. Passionerad för kontinuerligt lärande och anpassning till nya utmaningar inom mjukvaruutveckling.

Kunskaper

Programmeringsspråk: JavaScript/TypeScript, HTML/CSS, SASS, PHP, Python, Dart

Bibliotek & ramverk: React, React Native, Flutter, Django, Tailwind, GraphQL, WebRTC, Socket.IO

Verktyg: Git, GitHub, Netlify, Vercel, Fly, Docker, Webpack, Figma

Arbetslivserfarenhet

Sätra Trafikskola | App för körkortsteori | *PHP/Laravel, React/Next.js, TypeScript, Electron*

- Utvecklade ett responsivt användargränssnitt med React & NextJS och säkerställde typsäkerhet med TypeScript.
- Byggde en Electron-applikation för inbyggd Windows-kompatibilitet, vilket gör den enklare att använda för vanliga användare och ledningen.
- Implementerade i18n för internationalisering, vilket möjliggjorde stöd för flera språk.
- Använde Redux för tillståndshantering, effektiv hantering av autentiseringstokens och relaterade objekt.
- Använde PostgreSQL för datalagring, optimerade API-prestanda med SQL-frågeoptimering och Redis-cachelagring.

Projekt

Kinnema | Filmapplikation | *Typescript, Next.js, Django*

github.com/kinnema/frontend

- Utvecklade frontend med hjälp av React, Next.js, Tailwind CSS och TypeScript, vilket säkerställde hög kodkvalitet och minskade antalet buggar.
- Använde React Query för API-hämtning och TMDB API för data, vilket förbättrade datahämtning och tillståndshantering.
- Byggde med Django-backend, vilket gjorde att vi kunde fokusera på funktionskvalitet och enkel underhållning.
- Använde PostgreSQL för datalagring och Redis för caching, vilket resulterade i färre API-anrop och bättre resultat för användarbaserade förslag.
- Implementerade Docker för produktionsdistribution, vilket gjorde det enkelt att skala och underhålla applikationen.

Scan & GO | Kommunikation med QR | *React Native, TypeScript, Capacitor*

github.com/scan-go/mobile-ionic

- Utvecklade en mobilapplikation med plattformsöverskridande kompatibilitet i åtanke med hjälp av React Ionic och Capacitor. Gjorde applikationen tillgänglig på både webb- och mobilplattformar.
- Använde Supabase för datalagring och synkronisering, React Query för API-datahämtning, Zustand för tillståndshantering och Firebase FCM för notiser, vilket resulterade i en sömlös användarupplevelse med realtidsuppdateringar.
- Implementerade Supabase Realtime för meddelanden mellan användare, vilket uppnådde låglatenskommunikation.

Socium | Anonym social chattapp | *React Native, TypeScript, WebRTC, GraphQL*

github.com/muhammedkpln/socium

- Utvecklade ett API med Nest.js och GraphQL som effektivt cachar svar, vilket minskar serverbelastning och bandbredd för att förbättra prestandan.
- Implementerade Socket.IO för låglatens peer-to-peer-kommunikation och WebRTC för att hantera ljud- och videosamtal, vilket resulterade i sömlösa videosamtal med minimal fördröjning mellan användare.

Flyea | Filöverföringsprogram | *Flutter, WebRTC*

github.com/muhammedkpln/fileshare

- Utvecklade en applikation för sömlös filöverföring mellan enheter med hjälp av WebRTC, vilket gjorde den säker och förbättrade användarupplevelsen mellan användare.
- Implementerade Supabase Realtime för att lokalisera användare på samma nätverk, vilket ökade anslutningspålitligheten.