

OOSE DEA - Spotitube

v1.0

Rowan Paul Flynn 637588

Michel Portier

26 Maart 2021

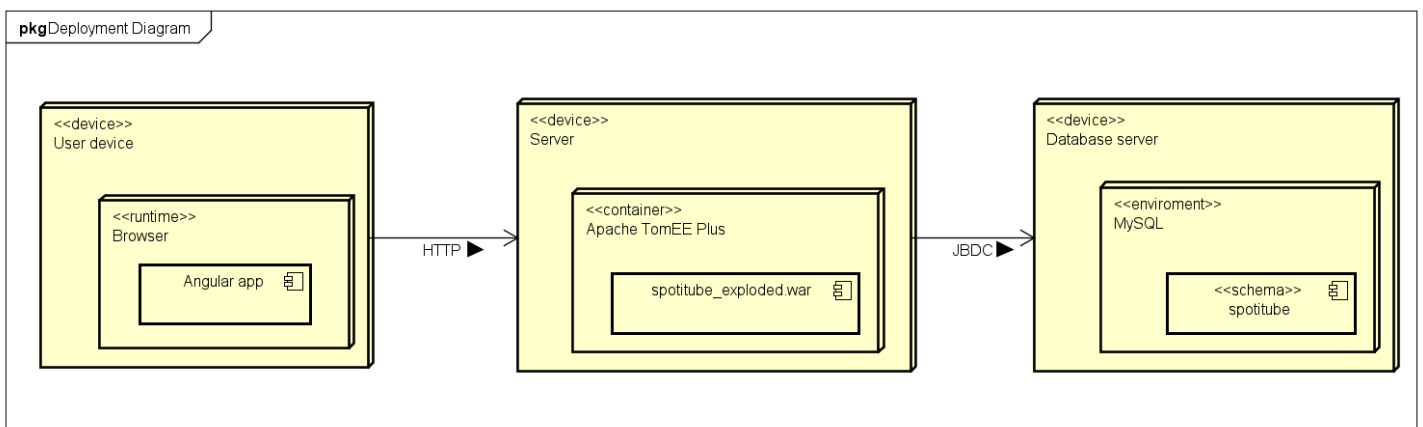
Inhoudsopgave

- [Inhoudsopgave](#)
- [Voorwoord](#)
- [Deployment diagram](#)
- [Package diagram](#)
- [Ontwerp keuzes](#)
- [Conclusie](#)

Voorwoord

Voor de OOSE DEA course heb ik een JAX-RS api gemaakt voor de reeds ontwikkelde front-end Spotitube app. De casus is Spotify en Youtube hebben hun krachten gebundeld en werken samen aan een app (Spotitube) waarmee een klant een overzicht krijgt van playlists met daarin audio- en videostreams. Ze willen eerst een deel van de back-end ontwikkelen en deze testen via een bestaande webapplicatie voordat ze verder gaan met de ontwikkeling van de app.

Deployment diagram



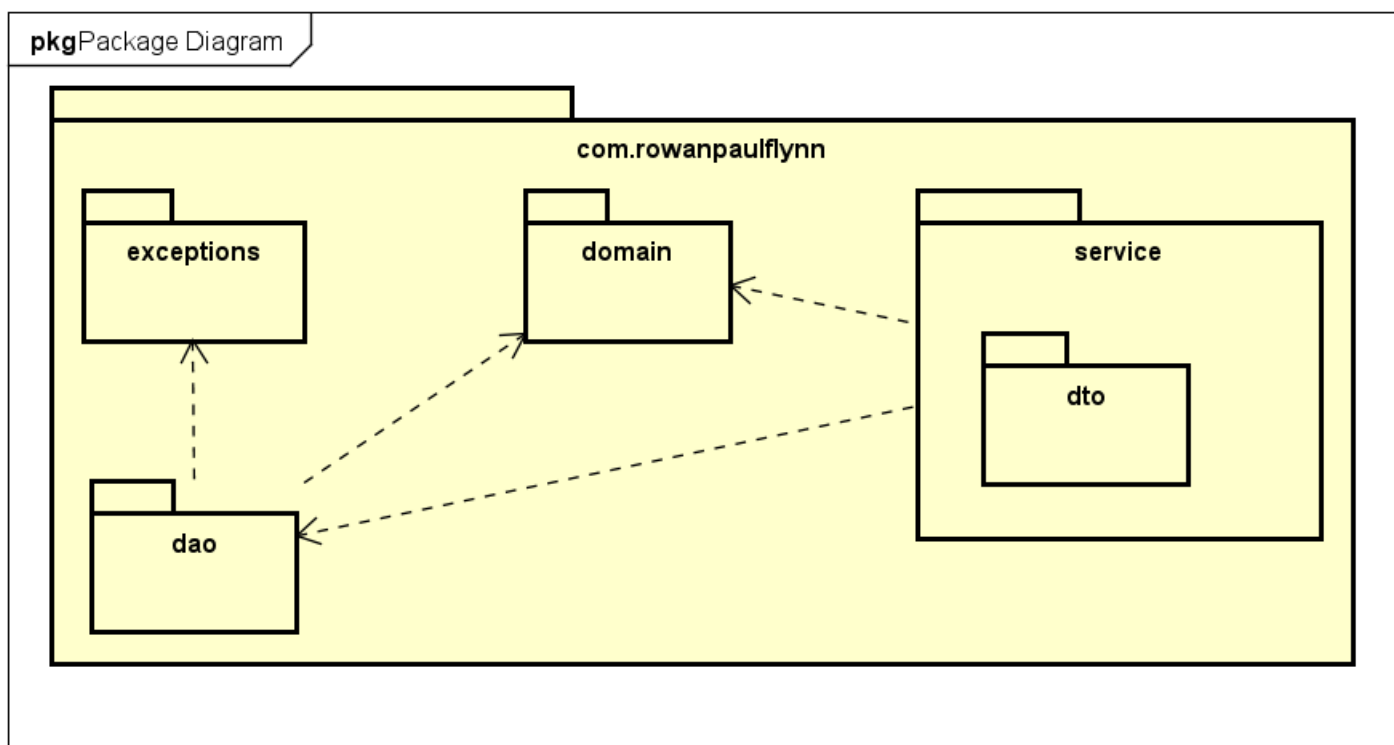
De applicatie heeft 3 verschillende devices:

- User device, waar de frontend op benaderd wordt door de gebruiker

- Server voor de JavaEE applicatie
- Database server voor de MySQL server

Om de database in te stellen kan gebruik worden gemaakt van het [database.sql](#) bestand die de database vult met wat voorbeeld data en alle tabellen goed opzet. De connectie naar de databse is geregeld via JDBC door de database toe te voegen aan de config bestand van TomEE.

Package diagram



Er zijn 3 lagen: de service laag, de domain laag en de dao laag. Al deze lagen zijn onafhankelijk van elkaar en kunnen uitgewisseld worden door andere implementaties. Dit is onder andere gedaan door dependency injection.

Service laag

De service laag bevat de routes en de DTO's, deze laag spreekt de client dus aan. De variabelen in de dto zijn direct aanspreekbaar en bedoeld om naar en van de client te versturen/ontvangen.

Domain laag

De domain laag bevat de classes die gebruikt worden in de dao. Deze classes hebben methoden en bevatten soms meer of andere variabelen dan de dto laag.

DAO laag

De DAO laag bevat de connectie naar de database en haalt als het ware alle gegevens daaruit en stuurt ze naar de service laag. De dao laag maakt op dit moment gebruik van MySQL, maar zou gemakkelijk door een NoSQL DAO kunnen worden vervangen.

Ontwerp keuzes

De gebruiker krijgt nadat hij/zij is ingelogd een token waarmee routes kunnen worden aangesproken. De token wordt willekeurig gegenereerd door een UUID aan te maken en die, gekoppeld aan de gebruikersnaam in de database te zetten.

Er zijn verschillende custom exceptions in de applicatie te vinden, bijvoorbeeld om aan te geven dat de gebruiker niet correct is. Er wordt gebruik gemaakt van HTTP status codes om te communiceren naar de server of iets fout is gegaan.

Conclusie

Met behulp van dit bestand kan de code van spotitube beter begrepen worden en kan de applicatie ook ingesteld worden op een eigen enviroment.