

Systeemontwerp

Architectuur

Domeinbeschrijving

Een bioscoop is een bedrijf dat films op grote schermen aanbiedt aan zijn klanten, en bestaat uit filmzalen met elk een verschillend aantal zitplaatsen en type projectiescherm. Daarnaast voorziet de bioscoop een uitgebreide shop waar klanten versnaperingen, drank en merchandising kunnen aankopen. De bioscoop heeft zijn eigen parking, voorbehouden aan klanten gedurende de uren van de film. Om dit alles te kunnen verwezenlijken moet de bioscoop ook voorzien in verschillende soorten personeel, dit omvat onder meer de kuisdienst, security dat alles in de gaten houdt, vakkenvullers, kassamedewerkers etc. Als laatste speelt publiciteit ook een rol, de bioscoop kan acties organiseren die gepaard gaan met kortingen, er moet publiciteit voorzien worden voor aankomende films (bv in bushokjes), trailers moeten afgespeeld worden voor de films.

Domeinanalyse

Business use case 1: Films inplannen

De programmatie verandert met de tijd. Films die net werden uitgebracht, worden doorgaans meerdere keren afgespeeld per week, in tegenstelling tot films die reeds een bepaalde tijd op de programmatie staan. Afhankelijk van het doelpubliek van de film worden een bepaalde filmzaal en tijdstip toegekend. Zo zullen films die vooral bedoeld zijn voor kinderen vroeger op de dag worden ingepland dan de films voor volwassenen.

De populariteit van een film speelt ook belangrijke rol, aangezien elke zaal een verschillende capaciteit (aantal zetels) heeft. De populariteit van films wordt gekenmerkt door de IMDB-score en bezoekersscore. Wanneer er veel interesse is voor een voorstelling, tracht men hieraan te beantwoorden door een film meerdere keren na elkaar in te plannen.

Business use case 2: Ticketverkoop

De klant kan online of ter plaatse de programmatie van alle films in een bepaalde bioscoop zien, en selecteert de gewenste film, de plaats(en), en het tijdstip naar keuze. Indien de klant over een kortingsvoucher of een bepaald lidmaatschap beschikt, kan dit tijdens deze aankoopprocedure worden aangegeven om het ticket aan te kopen aan verlaagde prijs.

Klanten krijgen toegang tot de filmzaal wanneer ze in het bezit zijn van een filmticket.

Business use case 3: Aanvullen stock van de shops

Om ervoor te zorgen dat alle klanten kunnen kiezen uit een rijkelijk aanbod aan versnaperingen en drank voor bij de film is het noodzakelijk dat de stock van de shop goed wordt aangevuld en beheerd.

Welke en hoeveel producten er verkocht worden wordt automatisch geregistreerd via het kassasysteem. Op basis van deze registratie en met een nader te bepalen drempelwaarde worden dan de stock voor de verschillende producten aangevuld. Voorbeeld hierbij: als er bijvoorbeeld een normale stock van 500 flesjes Coca Cola Zero 50cl zijn kan de drempelwaarde voor het aanvullen van de stock ingesteld worden op 30% (zijnde 150 flesjes) om deze aan te vullen.

Éénmaal per maand wordt de stock manueel geteld en vergeleken met wat de situatie van de stock normaal zou moeten zijn, dit om na te gaan hoeveel er gestolen is of kapot gegaan is die maand.

Business use case 4: Parking beheren

Opdat bezoekers van de cinema die met de auto gekomen zijn deze zouden kwijt kunnen, voorziet de cinema een parking. De bezoekers rijden de parking op en passeren hierbij de slagboom. Bij een druk op de knop krijgen de bezoekers een parkeerticket dat later gevalideerd moet worden.

De parking is gratis indien de bezoekers bij het valideren van het parkeerticket de barcode van hun filmticket laten scannen. Op die manier wordt de parking enkel gebruikt voor auto's van bezoekers die effectief een film gaan bekijken. Indien er bij de validatie van het ticket geen geldig filmticket kan voorgelegd worden, moet de bezoeker een bedrag betalen afhankelijk van de tijd bij het binnenrijden van de parking.

Het uitrijticket blijft geldig tot een kwartier na validatie. Op die manier kan vermeden worden dat de parking langdurig gebruikt wordt na het bekijken van een film.

Bij het binnenrijden van de parking is er een elektronisch bord aanwezig dat het aantal vrije parkeerplaatsen aangeeft, zo kan de bezoeker zien of hij/zij zijn/haar auto nog kwijt kan op de parking.

Business use case 5: Personeel beheren

Personeel wordt ingezet om de klanten van dienst te zijn. Naargelang er meer of minder films op de planning staan, wordt er meer of minder personeel aan de kassa geplaatst van de ticketverkoop en de shop. Er wordt een inschatting gemaakt van het nodige personeel voor de verwachte drukte.

Wanneer een werknemer zich ziek meldt, moet de werklust verdeeld worden onder het aanwezige personeel of wordt er gekeken naar extra werkrachten zoals jobstudenten. Een werknemer wordt dus dag per dag op de hoogte gehouden van zijn specifieke taken.

Bij het maken van de planning wordt niet alleen gekeken naar de programmatie van films en verwachte drukte maar ook naar de wensen van het personeel zoals bijvoorbeeld aangevraagd verlof.

De kuisploeg hoort tussen 2 vertoningen van een zaal de zaal te reinigen. Wanneer de aftiteling van de film start, kan de kuisploeg in de zaal. Pas nadat de kuisploeg klaar is, mag de zaal weer opengesteld worden voor nieuwe bezoekers.

Na de laatste vertoning of voor de eerste vertoning wordt elke zaal ook grondiger gepoetst. Ook de centrale hal wordt één tot meerdere keren per dag gereinigd, dit gebeurt doorgaans op vaste tijdstippen. De kuisploeg verzorgt ook de reiniging van de sanitaire voorzieningen en de parking.

Business use case 6: Publiciteit

De bioscoop maakt reclame voor zijn films en andere vertoningen. Door de ticketverkoop in de gaten te houden, kan ingespeeld worden op een onderboeking door binnen- en buitenshuis extra reclame te maken. Binnenshuis wordt filmreclame afgespeeld voor de film begint, maar ook in de centrale hal worden trailers afgespeeld en hangen posters op. Buitenshuis gaat het vooral om het afspelen van trailers in de media, reclame op posters en op digitale borden.

Elke trailer (en film) is bestemd voor een bepaald publiek. Op basis van doelgroepen worden de trailers geselecteerd bij het maken van een reclamefilm. In de reclamefilm worden ook advertenties van sponsors en bedrijven opgenomen, deze worden ook geselecteerd op basis van de doelgroep (bv. geen reclame voor bier in een kinderfilm). Als laatste is er in de reclamefilm plaats voor algemene boodschappen van de bioscoop, zoals reclame voor de shop, kortingsacties en info over de technologie. Reclamefilms worden op geregelde tijdstippen opnieuw gegenereerd.

Op basis van trefwoorden gelinkt aan films en reclames, worden zij optimaal gematcht en worden reclames gemedien waar mogelijk. Ieder bedrijf die reclame wil afspelen, kan online bestellen hoeveel kijkers het wil die zijn reclame zien. Hierbij dient het externe bedrijf dan ook zijn reclame mee te geven.

Business use case 7: Movie flow

Wanneer een film is ingepland voor een bepaalde zaal, moet die zaal de juiste trailers en film ophalen. Welke trailers er worden afgespeeld, wanneer de pauze wordt gestart, en de duurtijd van een pauze is afhankelijk van de film.

System operations

Gebaseerd op de domeinanalyse hebben we de systeemoperaties geïdentificeerd, een overzicht wordt hieronder weergegeven:

Ticket service:	Beschrijving
sellTicket()	Verkoop een ticket
cancelTicket()	Annuleer ticket en geef het zitje terug vrij
addActionDiscount()	Pas een korting toe op het ticket
validateParkingTicket()	Valideert een parkeerticket op basis van het cinematicket
Event: Ticket_sold	Dit event wordt uitgestuurd als het ticket verkocht is

HallPlanning service:	Beschrijving
getCinemaLayout(...)	Geeft terug hoe de cinema eruit ziet.
getDay(s)AfterDate(...)	Geef alle ingeplande dagen (voor een bepaalde cinema) terug na een bepaalde datum.
planMovie(...)	Geeft aan dat een bepaalde film ingepland moet worden.
unPlanMovie()	Geeft aan dat de film niet meer ingepland moet worden
getPlannedMovies()	Call dat alle ingeplande films teruggeeft.
Event: PLANNING_MADE	Event dat aangeeft dat de planning is aangepast.
Event: NEW_MOVIE	Event dat aangeeft dat er een nieuwe film zal gespeeld worden

Shop service	Beschrijving
sellItems(...)	Controleer dat er nog zoveel items in stock zijn, en verkoop ze indien mogelijk
changeTresholdForItem()	Veranderd de hoeveelheid, vooraleer er bijbesteld moeten worden.
changeStockCountInterval()	Veranderd het interval dat het commando uitgestuurd wordt.
updateStock	Geeft een verandering in de stock aan. (Diefstal of glitch is het systeem).
Command: askForStockCount	Geeft aan dat er best nog eens een stocktelling gehouden wordt. Is niet tijdsafhankelijk, dus is geen point of failure.

Parking Service	Beschrijving
getParkingTicket()	Ontvang ticket.
validateTicket()	Valideer het parkeerticket
exitParking()	Bij het verlaten van de parking dient de parkeerplaats vrij te komen
getNumberOfAvailableSpots()	Krijg het aantal vrije parkeerplaatsen

Movie Flow Service	Beschrijving
getMoviePlanning(cinemald, hallId)	Deze planning omvat informatie m.b.t. een individuele voorstelling, onder meer bestaande uit: trailers, tijdstip waarop pauze wordt aangevangen, duurtijd pauze.
Events: <ul style="list-style-type: none"> • MOVIE_[STARTED PAUSED RESUMED FINISHED] • MOVIE_BREAK_[STARTED FINISHED] 	Event die aangeeft dat een film (of pauze) werd gestart, hervat of gestopt.
Publicity service	Beschrijving
generateAdMovies(...)	Een expliciete vraag om meer reclame te maken voor bepaalde acties of films.
getAdByMovie(movie)	Geeft een adMovie terug voor een bepaalde film op basis van zijn categorie.
Staff service	Beschrijving
getDaySchedule(day)	Geeft een overzicht van de timeslots voor die dag. Timeslots bevatten informatie over de beschikbaarheid van een werknemer, alsook de locatie (e.g. filmzaal) waaraan ze eventueel toegekend zijn.
C/U/D AvailabilityTimeslot(day, timeslot)	Werknemers kunnen per timeslot aangeven of ze al dan niet beschikbaar zijn door een timeslot aan te maken. Het personeelsmanagement kan de beschikbare timeslots van werknemers al dan niet bevestigen. Bovendien is het mogelijk om via de timeslots te registreren of de werknemers effectief aanwezig waren.
API gateway	Beschrijving
getTicket()	Ontvang filmticket.
cancelticket()	Annuleer filmticket
getParkingTicket()	Ontvang parkeerticket.
validateTicket()	Valideer het parkeerticket.
exitParking()	Verlaat de parking.
C/U/D AvailabilityTimeslot(day, timeslot)	Voor het beheer van beschikbaarheids timeslots.
changeTresholdForItemInShop()	Veranderd de drempelwaarde van een item in de shop.

changeStockCountInterval()

Veranderd het interval waarin een stock count wordt uitgevoerd in shop.

Service decompositie

Gebaseerd op de systeem operaties en de domeinanalyse hebben we volgende services geïdentificeerd:

Services

TicketService

Beschrijving

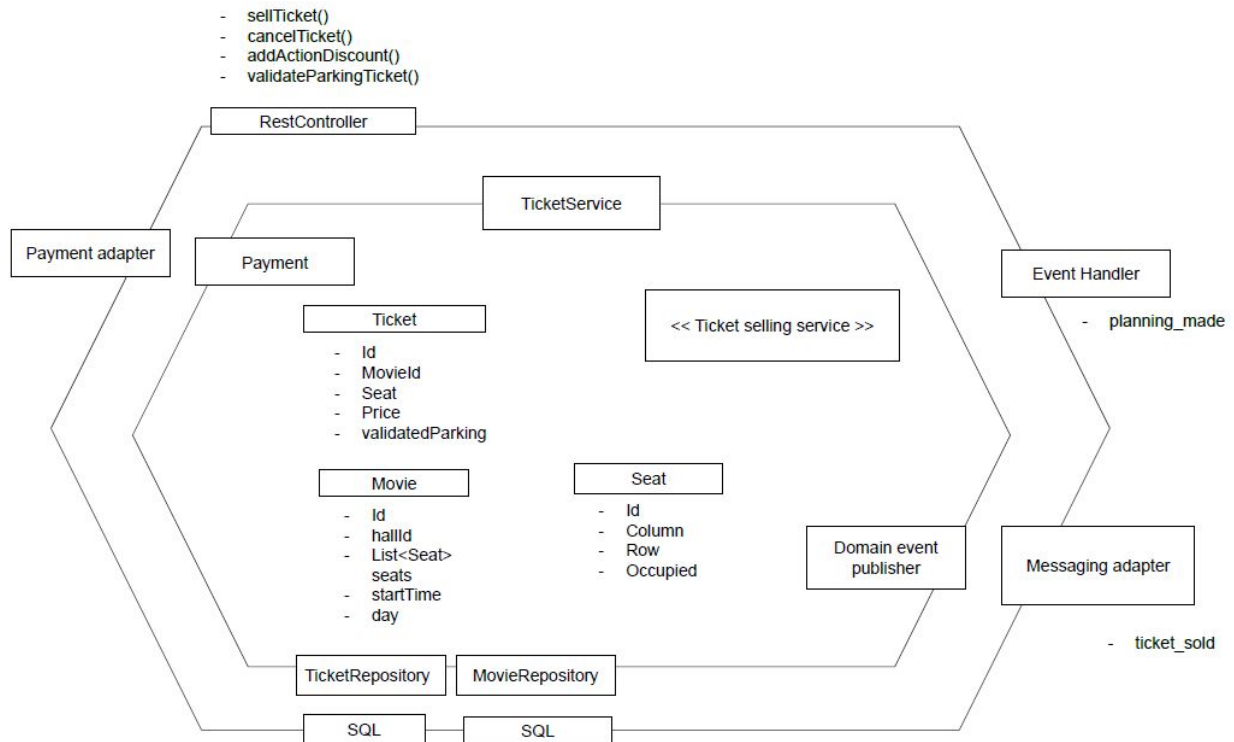
TicketService is verantwoordelijk voor het beheer en verkoop van filmtickets. De service beheert alle tickets die in omloop zijn en weet te allen tijde van alle tickets in omloop welke verkocht zijn, nog te koop zijn, nog niet zijn beschikbaar gemaakt, etc... Het staat in voor de ticketverkoop alsook het annuleren van gekochte tickets. Tickets kunnen aangemaakt worden en later beschikbaar gesteld worden voor verkoop. Daarnaast kan er een actiekorting toegepast worden op een ticket. Bij het gebruik van de parking van de cinema dient het parkeerticket gevalideerd te worden, dit kan door het cinematicket in te scannen bij validatie.

API

Naam	Beschrijving
sellTicket()	Verkoop een ticket
cancelTicket()	Annuleer ticket en geef het zitje terug vrij
addActionDiscount()	Pas een korting toe op het ticket
validateParkingTicket()	Valideert een parkeerticket op basis van het cinematicket
Event: TICKET_SOLD	Dit event wordt uitgestuurd als het ticket verkocht is

Collaborators

- HallPlanningService: getMovieSchedule()



HallPlanningService

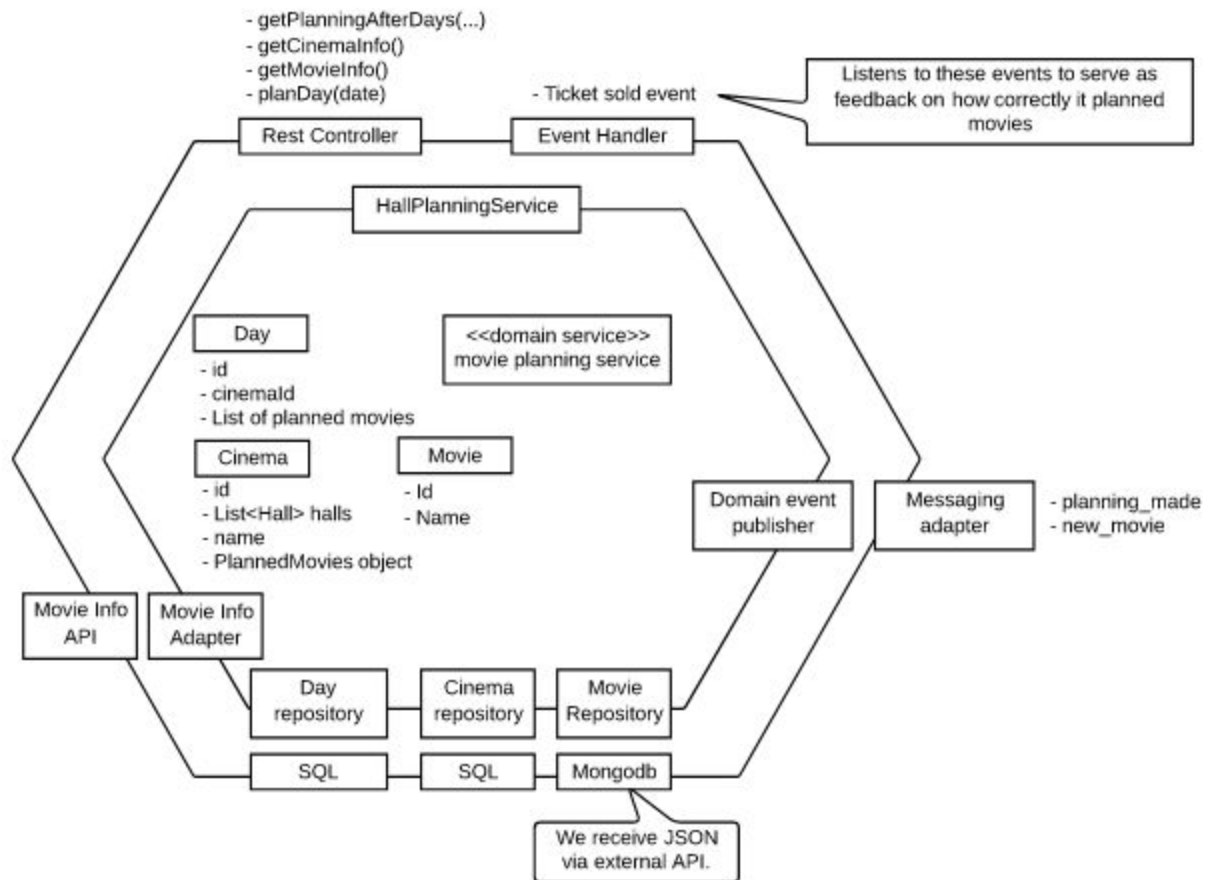
Beschrijving

De HallPlanningService beheert alle gespeelde films in de cinema. Films worden ingepland op bepaalde tijdstippen in de zaal. Per zaal is er een bepaald aantal beschikbare plaatsen. Deze plaatsen worden geboekt bij aankoop van een ticket en vrijgegeven bij annulatie van een ticket. De filmgegevens worden verkregen via een externe API. Deze service vangt de events van hoeveel verkochte tickets op, en kan als er veel of weinig tickets zijn, ingrijpen. De HallPlanningService kan echter volledig autonoom de planning aanmaken, zonder de informatie omtrent de verkochte tickets. Dit impliceert dat TicketService geen collaborator is.

Name	Description
getCinemaLayout(...)	Geeft terug hoe de cinema eruit ziet.
getDay(s)AfterDate(...)	Geef alle ingeplande dagen (voor een bepaalde cinema) terug na een bepaalde datum.
planMovie(...)	Geeft aan dat een bepaalde film ingepland moet worden.
unPlanMovie()	Geeft aan dat de film niet meer ingepland moet worden
getPlannedMovies()	Call dat alle ingeplande films teruggeeft.
Event: PLANNING_MADE	Event dat aangeeft dat de planning is aangepast.
Event: NEW_MOVIE	Event dat aangeeft dat er een nieuwe film zal gespeeld worden

Collaborators

- Geen



Shop Service

De shop service staat in voor het beheer van de versnaperingen, voordeelkaarten, cadeaubons en merchandise. Telkens een item wordt verkocht, wordt dat element als verkocht gemarkeerd. Verder is het ook handig om te weten hoeveel er al verkocht is van elk item. Regelmatig worden dan ook stocktellingen gedaan van items in voorraad, om te zien of niets verdwenen is. Ook kan er manueel bijbesteld worden, of aangepast wanneer een extra event is.

API

Name	Description
sellItems(...)	Controleer dat er nog zoveel items in stock zijn, en verkoop ze indien mogelijk
changeTresholdForItem()	Veranderd de hoeveelheid, vooraleer er bijbesteld moeten worden.
changeStockCountInterval()	Veranderd het interval dat het commando uitgestuurd wordt.
updateStock	Geeft een verandering in de stock aan. (Diefstal of glitch is het systeem).
Command: askForStockCount	Geeft aan dat er best nog eens een stocktelling gehouden wordt. Is niet tijdsafhankelijk, dus is geen point of failure.

Collaborators

- geen

Parking Service

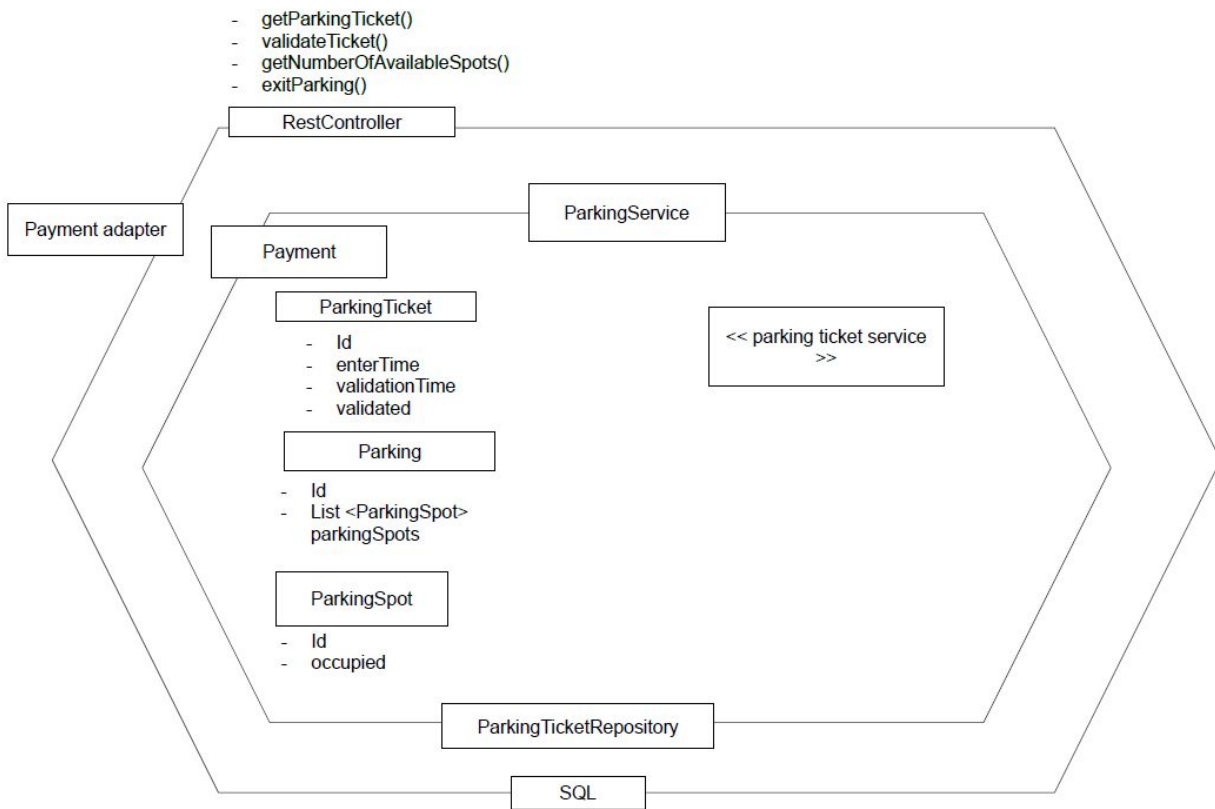
De parking service beheert alles wat te maken heeft met de parking van de cinema. Als eerste houdt het alle bezette/vrije plaatsen bij. Daarnaast beheert de service alle parkeertickets, bij het binnenrijden krijgt de klant een parkeerticket die hij/zij na de film dan kan valideren. Bij het valideren kan het cinematicket gebruikt worden om gratis de parking buiten te kunnen rijden. Zoniet dient het gebruik van de parking betaald te worden.

API

Naam	Beschrijving
getParkingTicket()	Ontvang ticket.
validateTicket()	Valideer het parkeerticket
exitParking()	Bij het verlaten van de parking dient de parkeerplaats vrij te komen
getNumberOfAvailableSpots()	Krijg het aantal vrije parkeerplaatsen

Collaborators

- TicketService: validateParkingTicket()



Movie flow Service

Deze service staat in voor het correct verloop van de film. De service haalt de reclamefilm en de film zelf op. De service zorgt dat deze op het juiste tijdstip aanvangen (reclamefilm speelt, film speelt, pauze, eindgeneriek speelt, einde).

API

Naam	Beschrijving
getMoviePlanning(cinemald, hallId)	Deze planning omvat informatie m.b.t. een individuele voorstelling, onder meer bestaande uit: trailers, tijdstip waarop pauze wordt aangevangen, duurtijd pauze.
Events: <ul style="list-style-type: none">• MOVIE_[STARTED PAUSED RESUMED FINISHED]• MOVIE_BREAK_[STARTED FINISHED]	Event die aangeeft dat een film (of pauze) werd gestart, hervat of gestopt.

Collaborators

- Publicity Service: getAdByMovie(movie)
- HallPlanning Service: getPlannedMovies()

Publicity Service

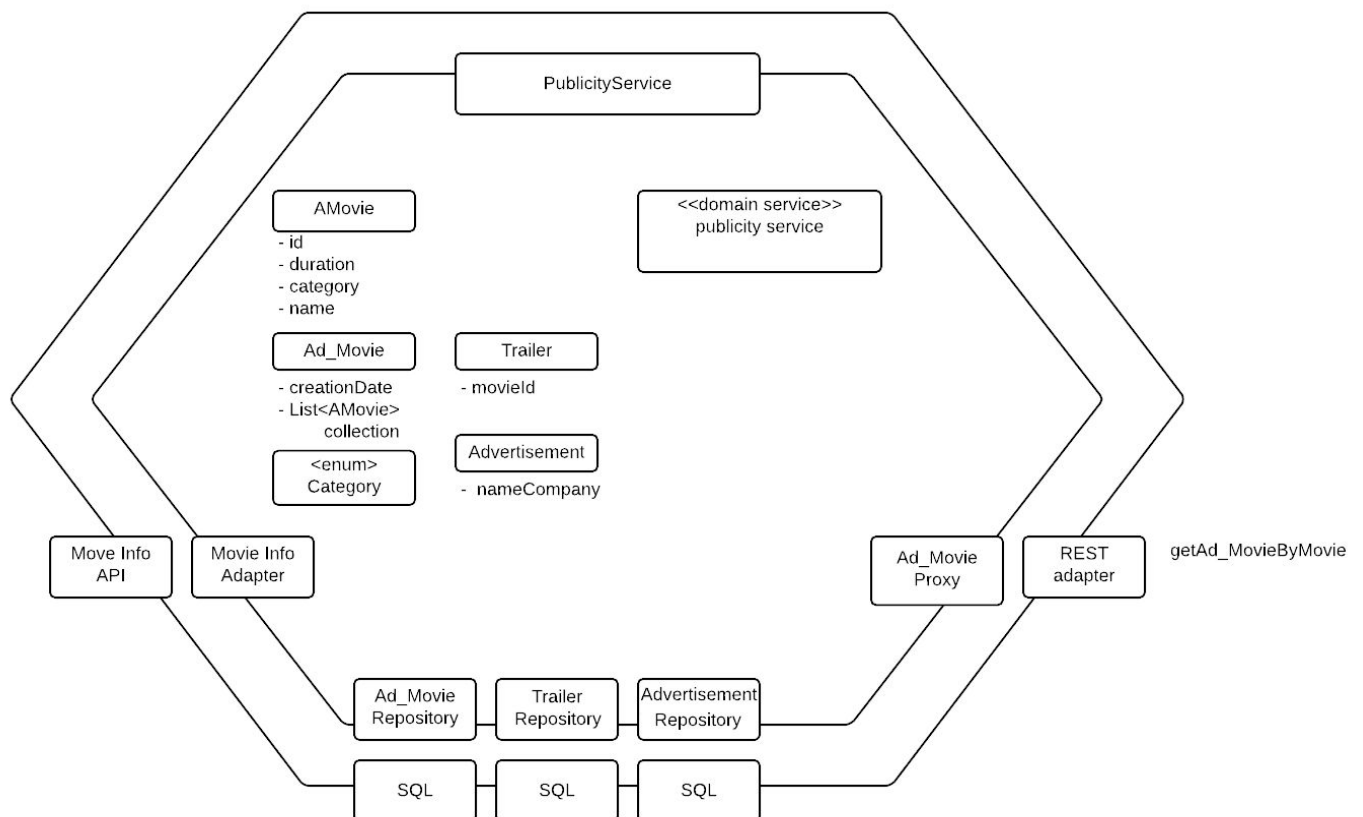
De Publicity service staat in voor het berekenen van de reclamefilms. De service genereert op geregelde tijdstippen nieuwe reclamefilms op basis van de nieuwe trailers en nieuwe advertenties. Oude stukken worden hierbij niet meer opgenomen. De service maakt per categorie één reclamefilm en labelt deze aan de hand van een tag. Er wordt op elk moment ook maar één reclamefilm per categorie ter beschikking gesteld. De service levert op basis van een meegegeven film, een gepaste reclamefilm.

API

Naam	Beschrijving
generateAdMovies(...)	Een expliciete vraag om meer reclame te maken voor bepaalde acties of films.
getAdByMovie(movie)	Geeft een adMovie terug voor een bepaalde film op basis van zijn categorie.

Collaborators

- geen



Staff Service

Personeel wordt ingezet in de kuisploeg (verzorgt de parking, zalen e.a.), aan de shop (aan de kassa en als teller), aan de ingang van de zalen.

De Staff Service staat in voor het inplannen van personeel. De service laat toe dat personeel hun beschikbaarheden kunnen doorgeven, zich ziek kunnen melden en verlof kan aanvragen. Het personeel kan de beschikbaarheden van de werknemers al dan niet bevestigen. Wanneer de staff een timeslot uitvoert, kan zowel de functie/taak als locatie (balie, zaal1, ..., etc...) worden gespecificeerd. Bovendien is het ook mogelijk om de effectieve aanwezigheden registreren.

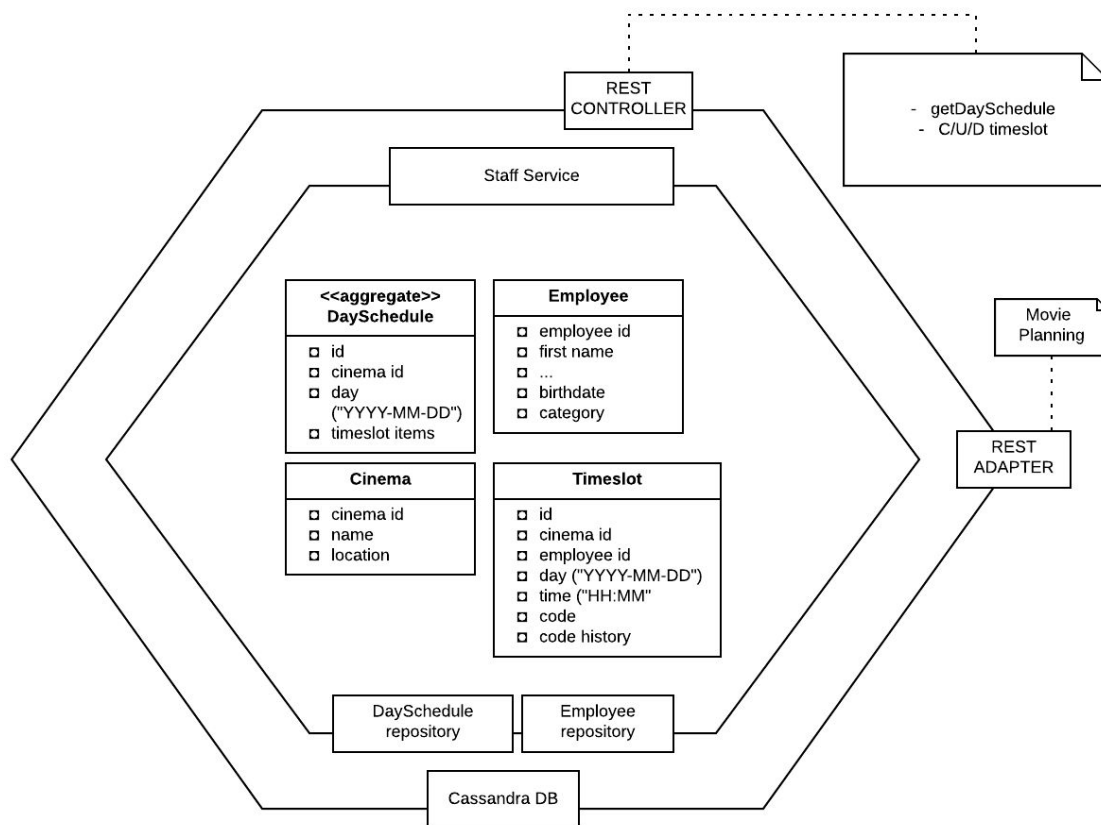
Als database wordt gekozen voor de *column-family structure* (Cassandra), omdat het aantal geboekte werknemers kan sterk variëren (e.g. populariteit film, grootte van de cinema verschilt per locatie). Dit brengt met zich mee dat ook de momenten waarop werknemers zich beschikbaar stellen sterk kan variëren. De tabel met planningen kan gepartitioneerd worden per cinema en dag. Bovendien kunnen de tijdsintervallen (*timeslots*) gesorteerd worden opgeslagen, zodat efficiënte range queries (e.g. tussen 14:00 en 18:00) kunnen worden uitgevoerd wanneer men moet nagaan welke werknemers al dan niet werden geboekt of zich als beschikbaar hebben ingesteld.

API

Naam	Beschrijving
getDaySchedule(day)	Geeft een overzicht van de timeslots voor die dag. Timeslots bevatten informatie over de beschikbaarheid van een werknemer, alsook de locatie (e.g. filmzaal) waaraan ze eventueel toegekend zijn.
C/U/D AvailabilityTimeslot (day, timeslot)	Werknemers kunnen per timeslot aangeven of ze al dan niet beschikbaar zijn door een timeslot aan te maken. Het personeelsmanagement kan de beschikbare timeslots van werknemers al dan niet bevestigen. Bovendien is het mogelijk om via de timeslots te registreren of de werknemers effectief aanwezig waren.

Collaborators

- HallPlanningService: de HallPlanningService bevat informatie over welke zalen er staan ingepland en het verwachte aantal toeschouwers. Deze informatie wordt door de StaffService gebruikt om de planning te maken.



ApiGateway

Beschrijving

De API gateway behandelt alle aanvragen van buitenaf en routeert ze naar de juiste service. Op die manier is het eenvoudig om de interne architectuur aan te passen zonder dat dit een grote invloed heeft op de manier hoe externe partners moeten interageren met het systeem.

API

Naam	Beschrijving
getTicket()	Ontvang filmticket.
cancelTicket()	Annuleer filmticket
getParkingTicket()	Ontvang parkeerticket.
validateTicket()	Valideer het parkeerticket.
exitParking()	Verlaat de parking.
C/U/D AvailabilityTimeslot(day, timeslot)	Timeslots beheren door werknemer.
changeTresholdForItemInShop()	Veranderd de drempelwaarde van een item in de shop.

changeStockCountInterval()

Veranderd het interval waarin een stock count wordt uitgevoerd in shop.

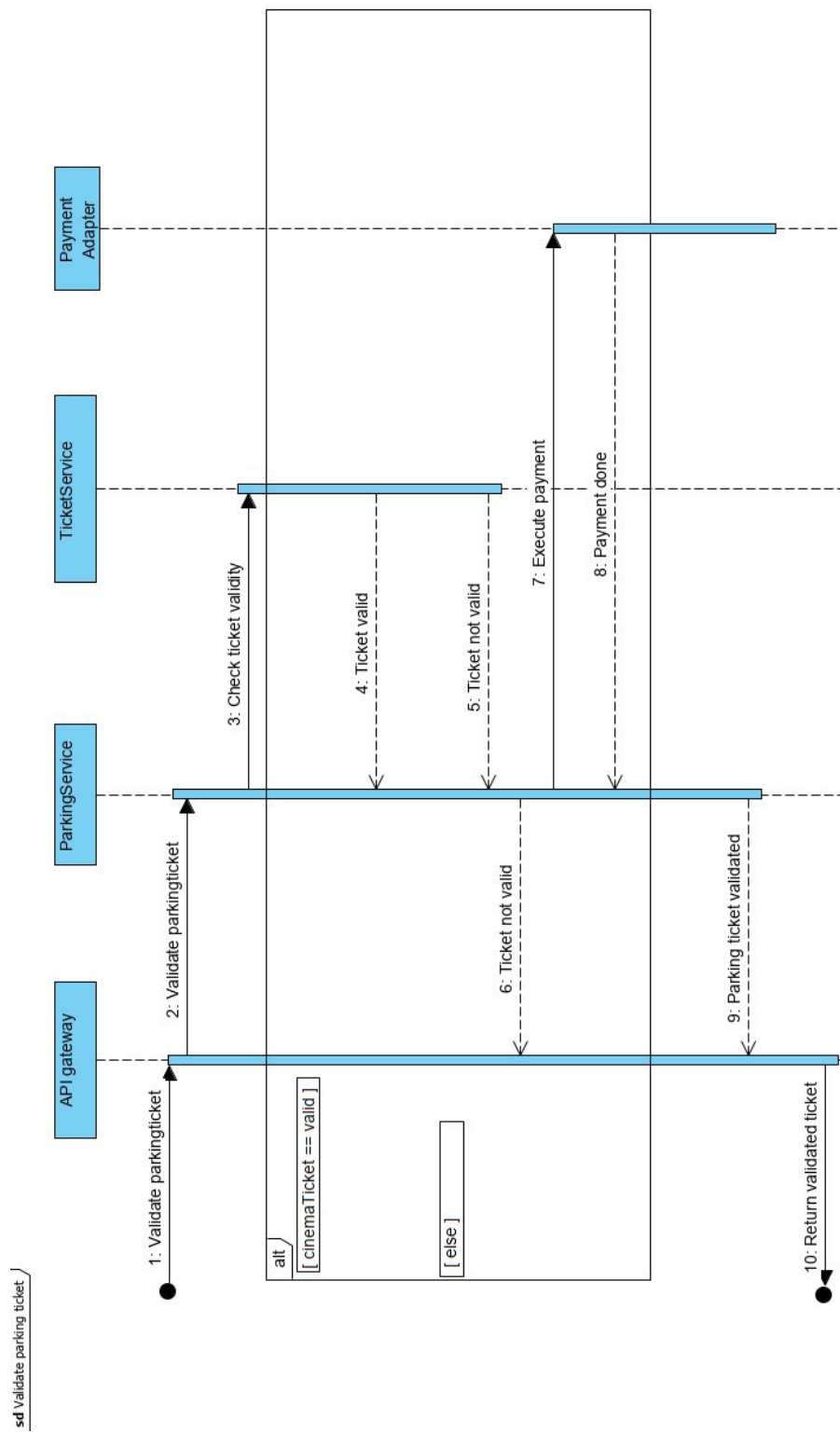
Collaborators

- TicketService: overeenkomstige ticket-functionaliteit doorgeven
- ParkingService: overeenkomstige parking-functionaliteit doorgeven
- StaffService: overeenkomstige staff-functionaliteit doorgeven
- ShopService: overeenkomstige shop-functionaliteit doorgeven, dit om toe te laten acties te ondernemen via een webclient

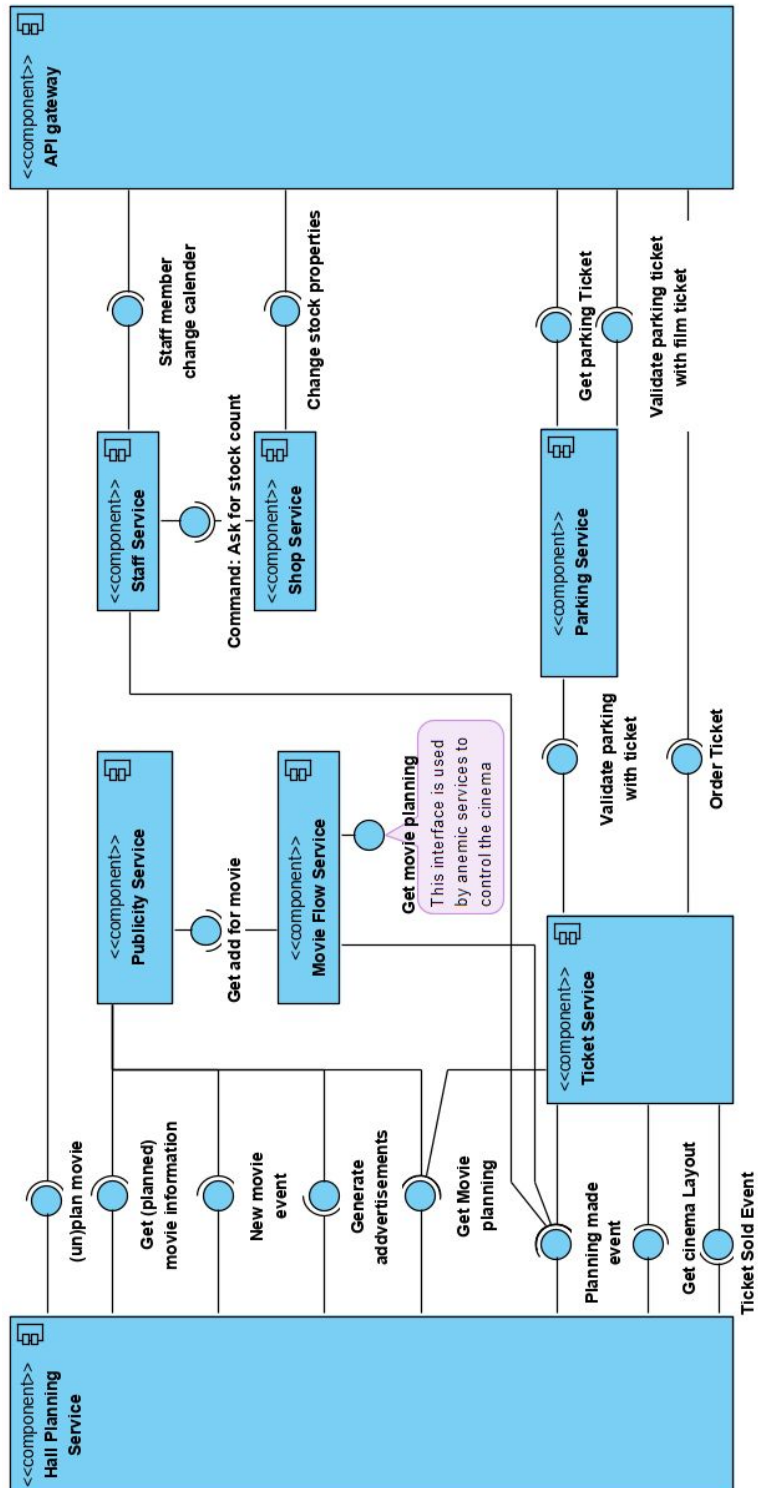
Sequentie diagrammen

Valideren van een parkingticket

Voordat een klant de parking kan buitenrijden dient zijn/haar parkeerticket gevalideerd te worden. Dit gebeurt door het ticket in de validatieautomaat te steken. Daarbij dient ook het cinematicket gescand te worden om zo gratis de parking buiten te kunnen rijden. Indien het cinematicket niet geldig is voor de geparkeerde tijd moet de klant betalen voor het gebruik van de parking. Een betaalsessie wordt gestart en wanneer die voltooid is, krijgt de klant het gevalideerde parkeerticket terug. Deze kan dan gebruikt worden bij het buitenrijden van de parking.



Component diagram



Addendum

Hieronder zijn per service eventuele wijzigingen aan de API of architectuur te vinden.

Parking Service

API

- `getNumberOfAvailableSpots()` is hernoemd naar `getNumberOfFreeSpots()`
- `getParkings()` toegevoegd (eerder voor testdoeleinden maar kan interessant zijn bij uitbreiden van het systeem)
- `getTickets()` toegevoegd (geeft aanwezige parkingtickets in de database terug)

Domain

- Klasse `ParkingSpot` is weggelaten, bleek overbodig te zijn.

Persistence

- `ParkingRepository` is toegevoegd (interageert met de database die de parking bevat)

Ticket service

API

- `cancelTicket()` is weggelaten
- `addActionDiscount()` is weggelaten
- event `TICKET_SOLD` werd niet geïmplementeerd
- `getAllTickets()` werd toegevoegd (om de aanwezige tickets te zien in de database)
- `getTicket()` werd toegevoegd (om een ticket te bekijken, gegeven een id)
- `getScreenings()` werd toegevoegd
- `getScreeningsByDate()` werd toegevoegd

Domain

- Klasse `Seat` is weggelaten
- Klassen `CinemaProxy`, `Hall`, `ScreeningProxy` werden toegevoegd
- Klasse `Movie` werd hernoemd naar `Screening`

Persistence

- `MovieRepository` werd hernoemd naar `ScreeningRepository`

Services

- `PaymentService` werd toegevoegd

Publicity service

De `Publicity_service` luistert nu ook naar het event 'update-planning' van de `HallPlanning_service`. Hierdoor weet de service welke dagen al ingepland zijn en welke niet. De service hoort naar dit event te luisteren om ervoor te zorgen dat als de businesslogica wijzigt i.v.m. de lengte van een `AdMovie`: Aangezien `HallPlanning` nu ook telkens vraagt hoelang de `AdMovie` zal duren (en hiervoor dus tijd moet inplannen), is het belangrijk dat een wijziging in de lengte hiervan pas mag ingevoerd worden voor de dagen die nog niet ingepland zijn. In tegenstelling tot de films, worden de `AdMovies` pas de dag van spelen opgevraagd en niet op de dag van planning.

API

- `/MaxDuration: getMaxDurationInSeconds()`: Geeft de maximale duur weer van een `AdMovie`. `HallPlanning` gebruikt deze waarde om in de plannen voor de start van de film
- `/maxDuration/{minutes}: setMaximalDuration(int minutes)`: Instellen van de businesslogica in een call. Zorgt ervoor dat de nieuwe duur van een `AdMovie` correct wordt aangepast.
- `/delay/{minutes}: setDelay(int minutes)`: Instellen van de delay: het verschil tussen de minimale en maximale duur van een `AdMovie`.
- De expliciete call voor `generateAdMovies` is niet geïmplementeerd.

hall-planning service

De structuur van de hall-planning is licht aangepast.

Er wordt geen gebruik gemaakt van de kafka stream `ticket_sold`. Ook wordt er geen gebruik gemaakt van `generate_ad's`, aangezien dit uit zichzelf gebeurt. Wat er wel gebeurt, is een restcall naar `publicity` die de maximum duration van de planning opvraagt.

Shop-service

De structuur van de shop service is licht aangepast.

Er wordt geen gebruik gemaakt van het commando `ask for stock recount`.

Staff service

De functionaliteit van de staff service voorziet de functionaliteit zoals werd aangegeven, met uitzondering op het verwijderen van timeslots (Create & Update operaties zijn echter wél geïmplementeerd).

Movie Flow Service

De restcall werd achterwege gelaten.

Api-gateway

De api-gateway is vervangen door een kubernetes ingress. Er is dus geen spring boot applicatie, maar wel een ingress. Alle rest calls worden doorgestuurd aan de hand van een ingress controller.