

ACI Rental



Datum: 17-01-2022

Klas: S3-DB01T

**Namen: Daan Brouwer, Michael Bednarek, Cyrion van Dongen, Bart Pelsma,
Xander Pluimert**

Wat is onze opdracht?

Wij gaan het bestaande systeem uitbreiden voor een echte klant. Voor Fontys ACI heeft een groep semester 6 studenten een eerste versie van een uitleensysteem ontwikkeld. Met dit systeem kunnen studenten AV-materialen lenen. Medewerkers kunnen bijhouden welke student wat heeft geleend en wanneer dit terug komt. Dit systeem mist nog functionaliteiten en het is onze opdracht om deze functionaliteiten te bedenken en uit te voeren.

Problemen die wij zijn tegengekomen.

Probleem 1	Opzetten lokale/online database en dan verbinden met back-end d.m.v. omgevingsvariabelen.
Probleem 2	Migration tables die de back-end moet aanmaken in de database werken niet/niet volledig.
Probleem 3	Front-end samen met back-end draaien, sommige services werken niet andere services half.
Probleem 4	Gebruik maken van Docker lost database problemen niet op.
Probleem 5	Uitleg van omgevingsvariabelen is kort en wordt slecht toegelicht.
Probleem 6	Het toevoegen van NuGet Packages gaat erg moeilijk of werkt niet.
Probleem 7	Reserveren lijkt niet te werken. We weten niet precies waar het probleem in zit. Beste gok is dat het alleen werkt met accounts alleen accounts werkt nog niet.
Probleem 8	Accounts. Dit is een van de dingen die we kregen van het andere team als een punt om aan te werken. Sommige functies lijken alleen te werken met accounts.
Probleem 9	Opdelen van rechten. Momenteel kunnen alle gebruikers overal bij. We moeten later als we accounts hebben hun kunnen opdelen in leerlingen en admins.
Probleem 10	Omdat reserveren nog niet werkt is het moeilijk om te testen of de reservering systeem werkt.
Probleem 11	Als we accounts hebben moeten we ook nog een login scherm maken. Misschien moeten wij ook nog een "maak account" pagina maken.
Probleem 12	We moeten iets regelen dat als producten verwijderd worden dat reserveringen en winkelmandjes ook geleegd worden van dat product.

Problemen oplossen

Oplossing 1	Ga naar "omgevingsvariablen van systeem bewerken", click bij "Geavanceerd" op "omgevingsvariablen". Maak een nieuwe omgevingsvariable met de naam "aci_db_string" en voer daar de connectionstring van een onlinedatabase. Als je dat gedaan hebt restart je pc.
Oplossing 2	Een standaard functie van Entity framework is het uitvoeren van migrations als je op YouTube opzoekt hoe je migrations uitvoert kan je makkelijk deze oplossing vinden. Hieronder staat ook de oplossing > dotnet ef migrations add InitialCreate > dotnet ef database update
Oplossing 3	Gebruik als browser Microsoft Edge of Chrome. Andere browsers lijken problemen te veroorzaken voor de front-end.
Oplossing 4	Zorg ervoor dat het een onlinedatabase is. Lokale databases lijken niet te werken.
Oplossing 5	Uitleg is aangepast waardoor het hopelijk wat beter te begrijpen is.
Oplossing 6	Hieronder staat de verdere uitleg voor het werkend krijgen van NuGet Packages in de front-end
Oplossing 7	Het probleem was in glitch in de datumprikker dit is nu opgelost.
Oplossing 8	Accounts onderdeel is momenteel zo gemaakt om eruit gehaald te kunnen worden dit was een eis van onze opdrachtgever.
Oplossing 9	De rechten per gebruiker moeten nog gemaakt worden, momenteel hebben we alleen een student, medewerker en een admin.
Oplossing 10	Reserveren onderdeel werkt kan getest worden en er zijn meer testen toegevoegd.
Oplossing 11	Login scherm is aangemaakt momenteel kunnen alleen accounts toegevoegd worden in de database.
Oplossing 12	Dit probleem is opgelost, er wordt nu getest in de winkelmand op deze mogelijkheden.

NuGet Packages verder werkend krijgen

install "*Microsoft.EntityFrameworkCore.Design*" in alle Services (versie 5.0.4)

In de terminal, werk in de folder van het project

```
> dotnet tool install --global dotnet-ef
```

```
> dotnet tool update --global dotnet-ef
```

Hierna voor iedere service:

Cd naar de service

```
> dotnet ef migrations add InitialCreate
```

```
> dotnet ef database update
```

Uitbreidingen en verbeteringen.

- De catalogus en de winkelmand werken nog erg onhandig, probeer het probleem te vinden en deze functies te optimaliseren. 1.
- Het maken van een rollensysteem denk hierbij aan een medewerker van de ACI rental, en de student die het product leent. 2. (geen entity framework, kan wel maar moet los te koppelen zijn)
- Aangeven of een lening al is teruggebracht, om te kijken welke nog niet terug zijn. 3.
- Mailtjes versturen naar studenten als hun lening afloopt, of een week van tevoren een mailtje sturen om de student te informeren.
- Beschikbaarheid van product. Wanneer een product terug hoort te zijn om weer uitgeleend te worden. 4.
- Ophaaldatum voor alle producten samen. 5. lcm met 1
- Medewerker kan leningen weigeren en accepteren. (bij speciale producten)
- Inloggen met Fontys PCN account. (eventueel aparte inlog repo pattern,
- Leengeschiedenis.
- Overzicht van een specifiek product. 6.

Swagger API

API staat voor Application Programming Interface dat is een software-tussenpersoon waarmee twee applicaties met elkaar kunnen praten.

Swagger is een krachtige en makkelijk te gebruiken API, wat tijdens het designen, bouwen, testen en het afmaken van je systeem gebruikt kan worden.

Swagger maakt zelf ook gebruik van verschillende functies die je kunnen helpen tijdens het maken van het systeem. Het is schaalbaar in grootte van kleine functie van een microservice, van grootte functies in microservices.

Waarom Swagger in dit project gebruikt is niet toegelicht maar het zal te maken hebben met aanmaken van een IP-adres voor het aanmaken van een beveiligde pagina binnen de catalogus.

REST

...

'REST' staat voor **RE**presentational **State** **T**ransfer en werd in 2000 geïntroduceerd door Roy Thomas Fielding.

REST is een techniek om verschillende nodes in een client-server architectuur met elkaar te laten communiceren. REST is hierbij niet de enige optie, andere mogelijke manieren zijn bijvoorbeeld SOAP of GraphQL. De reden dat REST in dit project gebruikt gaat worden, is voornamelijk gezien het al in de opgegeven structuur zit. Waarom er voor REST gekozen was, is verder niet aangegeven in vorige documentatie.

Verder werkt REST met zes basisprincipes. Zodra er niet volledig aan alle zes de principes wordt voldaan is het geen REST API, maar een RESTful API.

Docker

[What Is Docker?](#) | [What Is Docker And How It Works?](#) | [Docker Tutorial For Beginners](#) | [Simplilearn](#)



API Gateway/JSON



[What is API gateway really all about? Java Brains - Brain Bytes](#)

Entity Framework

<https://www.entityframeworktutorial.net/basics/how-entity-framework-works.aspx>



Definition of done:

- De issue moet gereviewd worden met een ander groepslid
- Wanneer Deze gereviewd is moet de front/back-end getest worden
- De issue moet gemerged worden met de dev branch
- Check met een docent of de story gemerged kan worden vanuit dev naar originele dev

Issues (die er nu nog zijn)

- Als gebruiker wil ik inloggen met mijn Fontys PCN, om gebruik te maken van de applicatie. #1
- Als gebruiker wil ik uitloggen van mijn Fontys PCN account, om de applicatie te verlaten. #1
- Als gebruiker wil ik mijn reservering kunnen wijzigen, zodat ik flexibiliteit heb in mijn lening. #2
- Als gebruiker wil ik mijn leengeschiedenis inzien, zodat ik een overzicht heb van mijn leningen. #3
- Als gebruiker wil ik mijn leengeschiedenis inzien, zodat ik een overzicht heb van mijn leningen. #4
- Als gebruiker wil ik kunnen filteren op mijn leengeschiedenislijst, zodat ik specifieke mijn leningen kan vinden. #5
- Als gebruiker wil ik een specifiek product kunnen bekijken, zodat ik een overzicht krijg van het product. #6
- Als medewerker wil ik een product kunnen wijzigen, zodat ik de producteninventaris kan aanpassen. #8
- Als medewerker wil ik een reserveeraanvraag kunnen goed-afkeuren, zodat sommige producten niet zomaar geleend kunnen worden. #9
- Als medewerker wil ik de leengeschiedenis in kunnen zien, zodat ik een overzicht heb van alle leningen. #11
- Als medewerker wil ik kunnen filteren op de leengeschiedenis, zodat ik specifieke leningen kan vinden. #12
- Als medewerker wil ik een printbon kunnen genereren, zodat ik dingen klaar kan leggen. #14
- Als medewerker wil ik een rol kunnen aanmaken, zodat ik deze kan toewijzen aan een gebruiker. #16
- Als medewerker wil ik een rol kunnen wijzigen, zodat ik de permissies van een rol aan kan passen. #17
- Als medewerker wil ik een rol kunnen archiveren, zodat ik permissies gemakkelijk kan verwijderen van een groep gebruikers. #19
- Als medewerker wil ik aangeven of een lening is opgehaald of teruggebracht, zodat ik kan inzien wat er nog teruggebracht moet worden. #23
- Als gebruiker wil ik dat het menu standaard word ingesteld op basis van welk device ik gebruik zodat de applicatie gebruikersvriendelijk is. #30

User Stories (voor de volgende groep)

UC	Issues	Omschrijving
UC1	#8	Medewerker van de uitleen service moet informatie van een product aan kunnen passen.
UC2	/	Als gebruiker wil ik in mijn catalogus een centrale datumpicker, waardoor ik kan zien welke producten beschikbaar zijn op mijn gekozen datum.
UC3		