**Analyse:**

* Klantenbestand bestaande uit naam, adres, klantnummer (geen dubbele waarden), telefoonnummer, e-mailadres, klanten login en wachtwoord.
* Producten in 1 tabel en verschillende tabellen bestaande uit fysieke boeken, e readers en audioboeken
* Titel, ISBN-nummer, Genre, prijs, auteur, taal, reeks/series, pagina’s en lengte
* Sorteerfunctie op prijs, auteur, genre, type product, ISBN-nummer, taal, reeks/series
* wishlist linken met klantenbestand
* klantenlogin
* beheer login
* login/wachtwoord reset

**Wat gaan we gebruiken?:**

* een GIT repository
* WTForms ([WTForms — WTForms Documentation (3.0.x)](https://wtforms.readthedocs.io/en/3.0.x/))

**Wat moet waar?:**

Databank(tabellen):

* product
* klant
* (verlanglijstje?)
* fysiek\_boek
* audio\_boek
* e\_boek

Python code:

* klant
* (verschillende boek klassen of niet?)
* filters/sorteerfunctie

Extra:

* use case diagram maken van tabellen en klassen
* use case diagram om alles te linken
* eventueel een DFD

**Vragen**:

In welke taal moet het programma staan? Engels of Nederlands?

* Maakt niet uit. Zorg wel dat alles in dezelfde taal is.

Moeten klanten login en beheer login ook in de databank gestoken worden?

* Daar zijn meerdere manieren voor. Je kan ze afhankelijk maken van je klantenbestand of ook niet. Dat wordt een klasse ‘user’. Meer wordt hierover gezien komende maandag (31 Januari 2022).

Moeten er aankopen geprogrammeerd worden?

* Nee

Geen shopping cart?

* Nee. De shopping cart fungeert eigenlijk een beetje als wishlist.

Verschillende rollen voor medewerker en manager?

* 1 profiel voor shopeigenaar is genoeg. Verschil tussen klant en beheerder is goed.

Moeten we iets van JSON voorzien?

* Nee

Hoeveel punten staan op analyse? Wat moet daar precies in?

* Als de code onduidelijk is, moet de analyse het wel verduidelijken. Daar mag een woordje uitleg bij.

De inhoud van de databank kunnen we dus niet delen. Dit zal dus enkel een db-design worden?

* Ja. De inhoud maakt niet uit, de werking en design wel.

Gaat de beheerder klanten kunnen verwijderen? Bijvoorbeeld klanten die inactief zijn, etc.

* Die functie mag, maar is niet per se nodig aangezien het ingewikkeld is.

Aangezien de deadline voor het project tegen eind mei is, is het de bedoeling dat we dan ons project constant gaan updaten met de nieuwe kennis en dingen die we nog gaan zien de komende maanden/weken? Of enkel met wat we ongeveer tot nu toe hebben gezien?

* Eigenlijk moet je het project kunnen maken met alles wat we tot nu toe hebben gezien.

**Extra toewijzingen:**

Als klant moet je kunnen inloggen en browsen, filteren, opzoeken, boeken toevoegen aan je wishlist. Als beheerder moet je kunnen inloggen, boeken toevoegen, verwijderen, etc.

Er gaan klanten zijn die niet in staat zijn zelf hun wachtwoord te resetten, dus stel de optie in om zelf via de commandline als admin het wachtwoord te veranderen/resetten.

‘@with\_appcontext’-command is super belangrijk!

Cookies duration van 30 dagen is meer dan genoeg.

**Agenda:**

28 Januari 2022:

Eerste plan van aanpak besproken, document met korte analyse gemaakt van wat we nodig hebben en gaan maken. Dat hebben we dan opgesplitst in wat we gaan coderen in Python en wat we gaan steken in de databank. Ook zijn er vragen bedacht voor het vragenuurtje en hebben we andere interessante vragen van andere groepjes met de antwoorden toegevoegd aan de analyse.

Taak tegen 31 Januari 2022:

We gaan allebei use-case diagrammen maken voor alle klassen en tabellen om die dan te vergelijken. Dit doen we om een goed overzicht te krijgen op wat we gaan doen.

31 Januari 2022:

Bespreken hoe we de login gaan aanpakken en verdere opvolging opdracht, begin pair-programming.

**Extra notities:**

*Gaan alle klanten geregistreerde klanten zijn? Of ook “gast”-klanten waarvoor geen registratie nodig is?*

*Gaan we in de producten databank een foreign key gebruiken? Het is gemakkelijker om dingen te verwijderen in de grote producten databank, want door die te linken met een foreign key met de kleinere producten databanken, worden ze daar dan ook automatisch verwijderd. het zorgt ook voor snellere queries.*