**Project Bibliotheek-beheer – Duo**

*Eenvoudig beheer, volledige controle: Ontdek de Bibliotheekbeheerapplicatie!*



​ **ADSD** 2022-2023

| **Teamlid** | **Studentnummer** |
| --- | --- |
| Yassine Messaoudi | s1188088 |
| Faissal Koulej | s1184579 |
| Versie 1.1 |  |

Inhoudsopgave

[**Project Definitie – Duo** 1](#_Toc126848476)

[Samenvatting 2](#_Toc126848477)

[Aanleiding & Missie 2](#_Toc126848478)

[Doelgroep 3](#_Toc126848479)

[Werkwijze 3](#_Toc126848480)

[Project grenzen 3](#_Toc126848481)

[Technische eisen 3](#_Toc126848482)

[Functionele eisen 4](#_Toc126848483)

[*Aanbevelingen:* 4](#_Toc126848484)

[Vraagstelling 4](#_Toc126848485)

[Probleemstelling 4](#_Toc126848486)

[Doelstelling 4](#_Toc126848487)

[**Resultaatomschrijving** 5](#_Toc126848488)

## Samenvatting

In deze projectdefinitie wordt beschreven hoe er een beheersysteem voor bibliotheekmedewerkers en beheerders zal worden ontwikkeld. Dit systeem moet hen in staat stellen om data te bewerken over verschillende items, zoals boeken, CD's en DVD's, en de auteurs die hiermee samenhangen

## Aanleiding & Missie

De aanleiding voor dit project is het verbeteren van de workflow en efficiëntie van de bibliotheekmedewerkers en beheerders. Ons doel is om hen te voorzien van een gebruiksvriendelijk en effectief beheersysteem dat hen in staat stelt om de data snel en eenvoudig te bewerken.

## Doelgroep

De hoofddoelgroep voor dit project zijn de bibliotheekmedewerkers en beheerders. Zij zullen het systeem dagelijks gebruiken en baat hebben bij een gemakkelijke en efficiënte manier van data-bewerking.

## Werkwijze

De applicatie zal worden gebouwd met behulp van .NET 5, WPF en Entity Framework Core 7. Hierdoor moeten de medewerkers en beheerders alle gegevens kunnen bewerken. De namen van variabelen, functies, methods en properties zullen in het Engels zijn, terwijl commentaar in het Nederlands wordt toegestaan. De bouw van de applicatie zal in sprints worden opgeslagen in de git repository.

## Project grenzen

Het project zal zich beperken tot het ontwikkelen van een beheersysteem voor de bibliotheek. Andere aspecten, zoals de verkoop van items, worden buiten beschouwing gelaten.

### Technische eisen

* Lazy- en/of Eager loading moeten correct toegepast worden om de juiste data op te halen.
* Local.ToObservableCollection() moet toegepast worden om de Views direct te binden aan de database-data.
* De Views moeten gebruik maken van Binding.
* Er moeten elementen in de interface zitten die bindings en INotifyPropertyChanged demonstreren (bijvoorbeeld invulvelden en knoppen).
* De collecties moeten gebruik maken van ObservableCollection.
* Migrations en seedings moeten aanwezig zijn.

### Functionele eisen



* CRUD-functies voor items en auteurs.
* Master-detail view voor items en auteurs
* Overzicht van alle items Mogelijkheid tot sorteren en filteren van de lijst
* Minimaal de mogelijkheid om op item en auteur te zoeken
* Gebruiksvriendelijke interface

### *Aanbevelingen:*

* Bestudeer de theorie- en praktijkvideo’s in Teams.
* Het is aanbevolen om de volgende boeken te lezen: "Pro C# 10 with .NET 6: Foundational Principles and Practices in Programming" en "C# 10 in a Nutshell".
* Bestudeer de coding conventions en DOT-taal.
* Bestudeer technieken en methoden, zoals CollectionViewSource voor filteren en sortering, en LINQ-queries voor het weergeven van gegevens.

### Vraagstelling

Welke vereisten zijn er nodig om een gebruiksvriendelijk beheersysteem voor bibliotheekmedewerkers en beheerders te ontwikkelen?

### Probleemstelling

Het huidige beheersysteem voor bibliotheken is inefficiënt en gebruikersvriendelijkheid kan verbeterd worden.

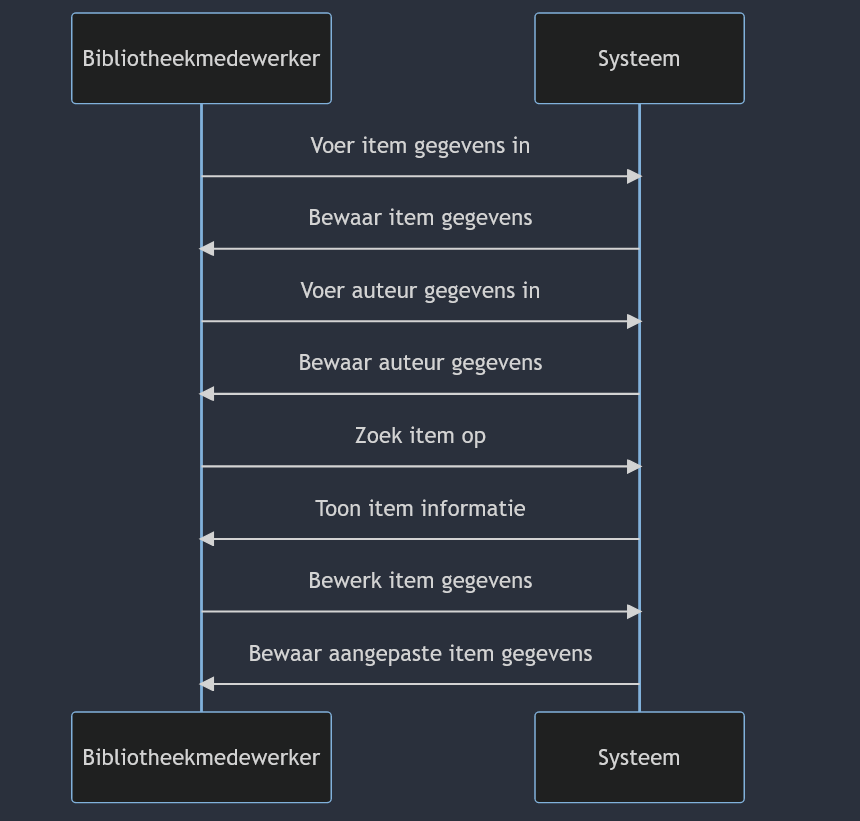
### Doelstelling

Het ontwikkelen van een gebruiksvriendelijk en efficiënt beheersysteem voor bibliotheekmedewerkers en beheerders.

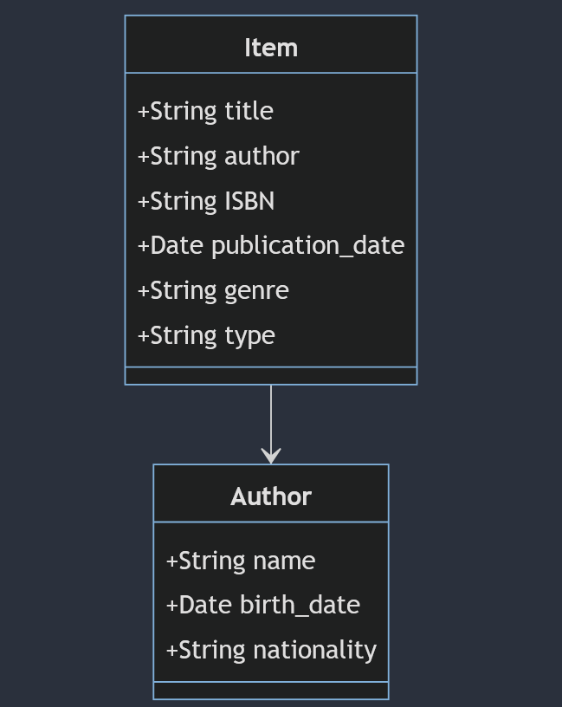
#### **Diagrammen**

**Voor het verduidelijken van het ontwikkelproces, zullen er verschillende diagrammen worden gebruikt, zoals data- en procesmodellen (we maken ze in mermaid code).**

Je zal hieronder meerdere diagrammen vinden dit betekent **niet** dat we ze ook daadwerkelijk op dezelfde manier gebouwd hebben zoals in de diagrammen weergeven is (het is een voorbeeld).



In dit procesmodel kun je zien hoe een bibliotheekmedewerker interacteert met het beheersysteem. Het proces begint wanneer de bibliotheekmedewerker item-gegevens invoert. Het systeem bewaart vervolgens deze gegevens. Hetzelfde proces gebeurt met auteur-gegevens. Daarna kan de bibliotheekmedewerker een item opzoeken en informatie over het item bekijken. Als laatste kan de bibliotheekmedewerker item-gegevens bewerken en het systeem bewaart de aangepaste gegevens.



*Dit is een class diagram dat de relatie weergeeft tussen twee classes: "Item" en "Auteur". De klasse "Item" heeft attributes zoals "title", "author", "ISBN", "publication\_date", "genre" en "type". De class "Auteur" heeft attributes zoals "naam", "geboortedatum" en "nationaliteit". De pijl tussen de twee classes vertegenwoordigt de relatie tussen een item en de bijbehorende auteur.*

## **Resultaatomschrijving**

Het eindresultaat van dit project zal een beheersysteem zijn waarmee bibliotheekmedewerkers en beheerders snel en eenvoudig data kunnen bewerken over items en auteurs. Dit zal leiden tot een verbetering van de workflow en efficiëntie van de bibliotheekmedewerkers en beheerders.

*Inleverproces:*

***Deadline: 14 april 2023***

Het project moet via de inlevermap op BrightSpace worden ingeleverd, samen met de Git repository. De .vs- en packages-mappen mogen niet in de repository worden opgenomen. Het project moet door twee personen worden ingeleverd en moet zijn gebaseerd op één git en één repository. Eventuele extra elementen die zijn toegevoegd op eigen initiatief, zijn voor eigen ris