

# PLS Documentation

Namen van de studenten uit het team:

- Giovanni Frederikz : 0978427
- Illias (I don't know tbh)
- Nils :c Lindrob : 0989357

*Requirements en diagrammen staan op de volgende pagina's*

# Requirements Document

## Public library system

### Doel en doelgroep

De doelgroep is een werknemer in de bibliotheek. Het doel is om een stukje software te schrijven wat een database met boeken en gebruikers bijhoud.

### Productomschrijving

Wij maken een product voor een bibliotheek. In ons product staat een database met boeken en gebruikers. In die database staat ook aangegeven of een klant een boek heeft geleend, welk boek de klant heeft geleend en wie het boek heeft geleend. Een klant kan ook een boek terugbrengen.

### Requirements

#### User Requirements

- De beheerder kan het systeem vullen met boeken.
- De beheerder kan klanten toevoegen.
- Een boek kan opgezocht worden.
- Een klant kan een boek uitlenen
- Boeken terug kunnen brengen
- Backup van de database maken
- Backup van de database terugzetten

#### Functional Requirements

- **Er moet een books class komen**
- **Een boek kan niet twee keer tegelijk uitgeleend worden**
- **Er moet een Backup class komen**
- **Er moet een User class komen**
- **De persoon die het boek uitleend moet bekend zijn bij het systeem**

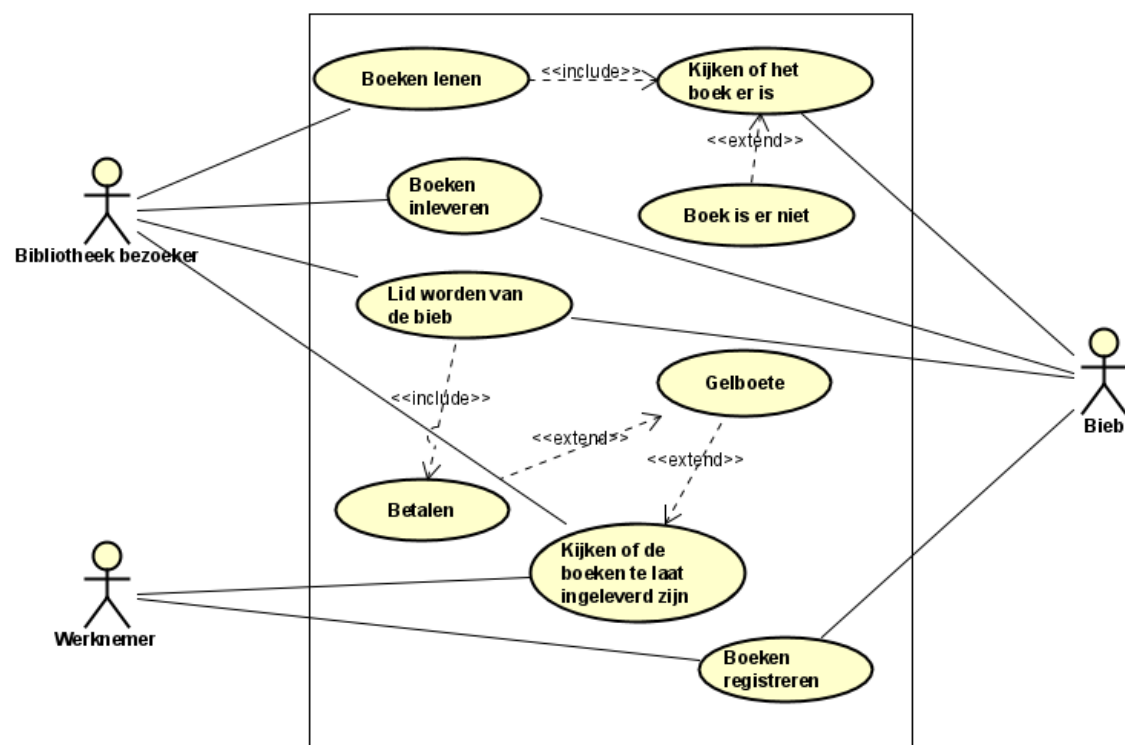
#### Non-Functional Requirements

- Er moet een handleiding komen die uitlegd wat alle functies doen
- Er moet een class diagram komen die de relaties tussten de classes uitlegd

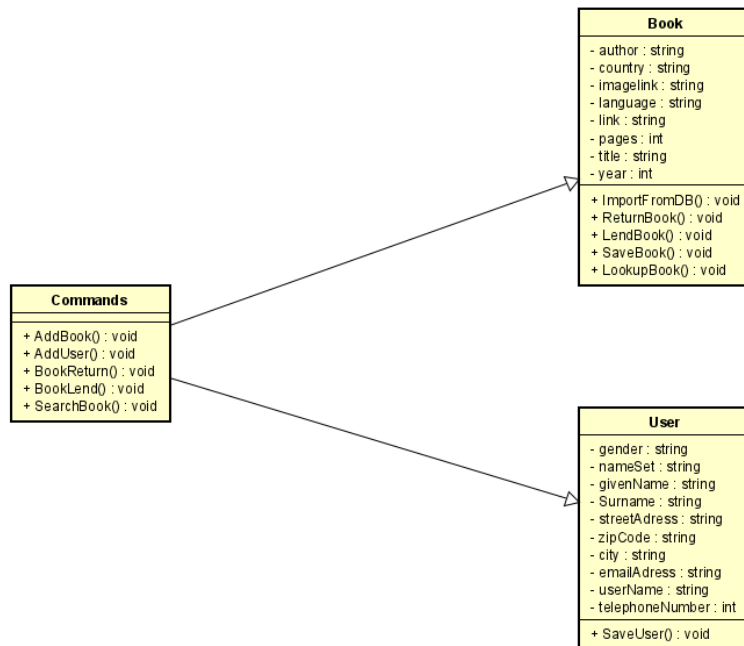
- Er moet een use case diagram komen die de relaties tussen de gebruikers en use cases uitlegd
- Er moet een commands class komen voor de commands die uitgevoerd kunnen worden

## Use case diagram

uc



## Class Diagram



# Pls analysis user manual

## Functions and what they do.

### Commands.AddUser(0)

Creates a user account with the input that give too it.

```
Whats is your gender:> a
Whats is your nameSet:> b
Whats is your givenName:> c
Whats is your Surname:> d
Whats is your streetAddress:> e
Whats is your zipCode:> f
Where do you live:> g
Whats is your emailAddress:>
```

### Commands.AddBook(0)

Creates a book and saves it.

```
Who is the author:>
What is the country:>
What is the imagelink:>
What is the language:>
What is the link:>
How many pages has the book:>
what's the name of the book:>
What is the year of realease of the book:>
```

### Commands.BookReturn(0)

If a book is lend away you can return it so that other can get it again.s

```
Input the name of the person who lend the book:>
```

### Commands.BookLend(0)

Gives a book away and puts it as not available for others.

```
Input the name of the person that wants to lend the book:>
```

### Commands.SearchBook(0)

Looks of the book with the given input.

```
]do you want to search via:
- title
- author
Type the word for the search key
```

### Backup.BackUp()

Creates a backup from all the save files.

**Backup.retrieve\_BackUp()**

Uses the saved back up as.

**Book.ImportFromDB(0)**

It sets the books set in book database.