UNIVERSITÄT DUISBURG ESSEN

Open-Minded

Lab for Software Engineering
Winter term 2022/2023

Lab for Software Engineering

Cinema Management Application

Ifrat Jahan (3098878)
Jenniffer Maxisch (3106694)
Georgios Adamos (3093306)
Thomas Klimek (3067855)
Melvin van der Linde (3106762)

October 24, 2022

University of Duisburg-Essen · Faculty of Engineering Working group Software Engineering – Prof. Dr. M. Heisel
Date: Do. 12-14- Supervisor: Marcel Schweikert

Contents

List of Figures

1 Analysis

1.1 A1

1.1.1 Requirements & Domain-Knowledge

Requirements

R1 Customers can create an account by providing an e-mail address and a password.

R2 If an e-mail address which is already associated with an account is provided during registration, account creation fails.

R3 Customers can log in by providing their e-mail address and their password.

R4 A logged in customer can log out.

R5 A customer can browse available showings, ascendingly sorted by date.

R6 A logged in customer can book tickets by selecting the showing from the browsing list and selecting the desired seats.

R7 A showing can only be booked up to 15 minutes before it starts.

R8 Staff can add new showings to the database by providing the required data.

R9 Once a showing starts it is marked as "archived".

R10 Archived showings are visible to staff, but not to customers.

R11 Staff can cancel showings.

R12 When a showing is cancelled it is deleted.

R13 When a showing is cancelled the customers who booked tickets for it are notified via e-mail.

R14 Showings which took place a year ago or longer are automatically removed from the database.

R15 When a showing is deleted its associated bookings are also deleted

Facts

F1 A showing consists of the title of the movie, its duration, the date date, the hall number and unique ID.

F2 A hall consists of a number of rows, a number of seats per row and a unique hall number.

F3 Only one person at a time can sit in a seat.

Assumptions

A1 A web application is a good choice for implementing the desired functionality and all customers are able to use it.

A2 Customers only provide e-mail addresses they can access.

A3 Customers will stay up to date with the list of available showings.

A4 The cinema operator has provied a predifned list of cinema halls.

A5 Every booking is paid via an external service.

A6 Staff will only add showings which take place in the future.

1.1.2 Contextdiagram

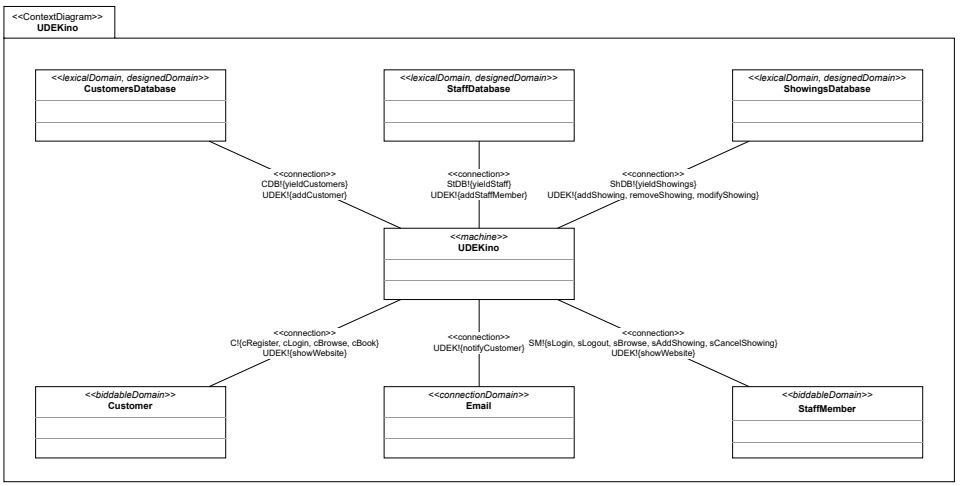

Figure 1.1: Contextdiagram

1.2 A2

We can derive the following problem diagrams

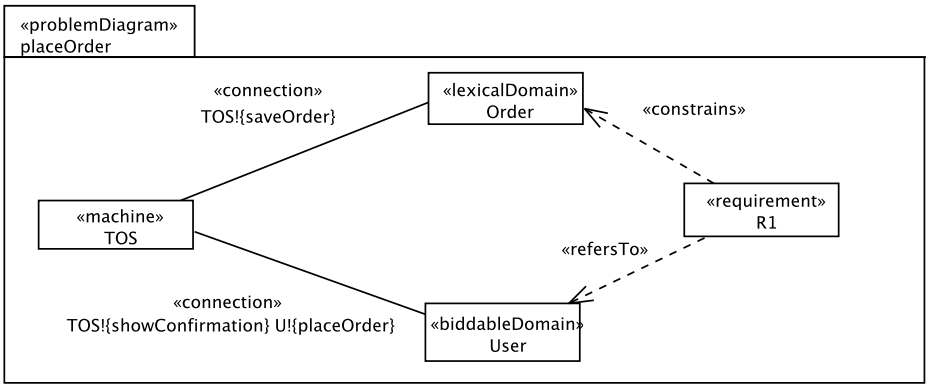

Figure 1.2: Problemdiagram for R1

1.3 A3

1.4 A4

1.5 A5

A short OCL example:

```
1 context Person inv: self.alter >=0
2
3 pre alter>30
4 post alter=alter@pre+1
```

1.6 A6

Examples of a life-cycle using the math-environment:
$LC_{guest} = (Browse^{+}; [Book])^{*}$

2 Design

2.1 D1

2.2 D2

2.3 D3

2.4 D4

State diagrams with tikZ:

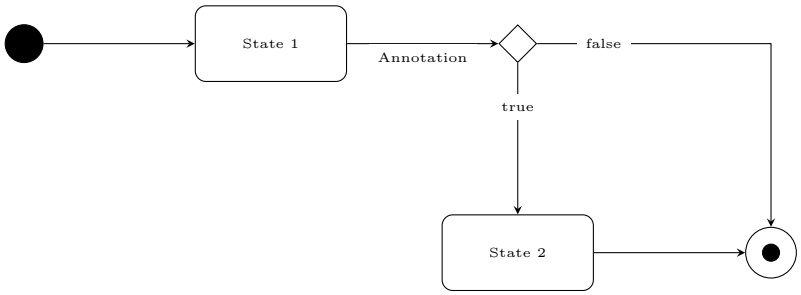

Figure 2.1: Zustandsdiagramm Person 1

3 Implementation & Testing

3.1 I

3.2 T1

3.3 T2

3.4 T3

4 Glossary

Table 4.1: Glossary

Name	Type	Description	Source
A			
Anton	biddable Domain	User of the system	Contextdiagram
B			
C			
D			
E			
F			
G			
H			
I			
J			
K			
L			
M			
N			
O			
P			
Q			
R			
S			
Stakeholder	biddable Domain	User of the system	contextdiagram
T			
TOS	machine Domain	Software	contextdiagram
U			

Table 4.1: Glossar

Name	Type	Description	Source
User	biddable Domain	User of the system	contextdiagram
V			
W			
X			
Y			
Z			