

Modul : MIB 14 -Softwareentwicklungsprojekt
Pflichtenheft

Software-Projekt
“MovieRent”

Laura Salazar López 5279483
Samira Joulai Vakili 5301788
Fabius Pudlich 5319932
Timo Trauschold 5299641
Dahun Lee 5277155
Philipp Luis Elgert 5219650

Wintersemester 2021/2022

Inhaltsverzeichnis

1 Einführung	3
1.1 Problembeschreibung	3
1.2 Vorstellung der Webseite und gewähltes Vorgehen	3
1.3 Systemkontext	4
1.4 Stakeholder	5
1.5 Verwendete Technologien	5
1.6 Rahmenbedingungen	5
2 Anforderungen	6
2.1 Akteure	6
2.2 Funktionale Anforderungen	8
2.3 Qualitätsanforderungen	8
3 Fachlichkeiten	10
3.1 Use-Case Modell	10
3.2 Anwendungsfall (Use-Cases)	11
4 Details	18
4.1 Gui-Beschreibung und Mock-Ups	18
4.2 Datenbank	24
5 Glossar	25
6 Quellen	26

1 Einführung

1.1 Problembeschreibung

Nachdem wir uns als Gruppe mit verschiedenen Online Videotheken auseinandergesetzt haben, sind folgende Aspekte aufgefallen:

- Simplizität (1)
- Schnelligkeit (2)
- kostenlose Zugänglichkeit (3)
- Registrierung von Konten (4)

Anhand diesen Punkten haben wir beschlossen ein webbasiertes Programm zu entwickeln, mit dem es möglich ist, Kunden anzumelden (4) und die gewünschten Filme möglichst einfach (1) und schnell (2) zu verleihen. Eine kostenlose und freie Zugänglichkeit (3) ist bei unserer Software nicht gegeben, da der Anwendungsbereich intern bei einem Mitarbeiter einer Filmbibliothek liegt.

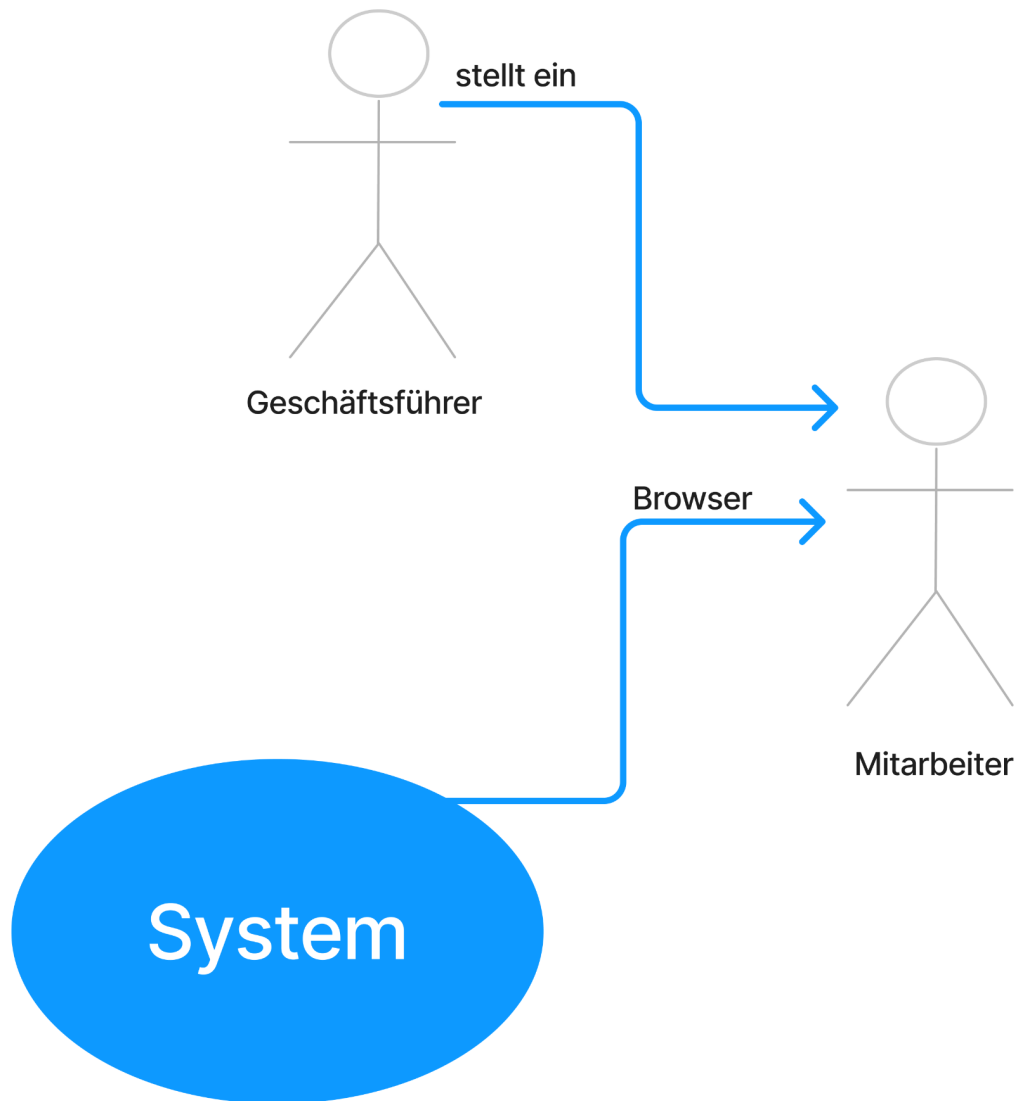
1.2 Vorstellung der Webseite und gewähltes Vorgehen

Wie bereits festgestellt, handelt es sich bei unserem Projekt, um ein webbasiertes Programm. Dabei ist der Verleih aus der Perspektive eines Kassierers bei einem Videoverleih zu verstehen. Mit dem Programm soll es möglich sein, Filme zu suchen und diese dann anzuzeigen, ob sie verfügbar sind. Darüber hinaus soll auswählbar sein, ob der Film verliehen oder zurückgegeben werden soll. Bei unserem Programm gibt es zwei Bereiche - Filmbereich und Kundenbereich. Im Filmbereich werden anhand von Vorschlägen Filme angezeigt, die ebenfalls ausgeliehen werden können. Diese erscheinen dann im Warenkorb (mehr in 4.1 Gui-Beschreibung und Mock-Ups). Andererseits gibt es den Kundenbereich. In diesem Bereich kann nach Kunden gesucht werden, die dann eine Liste der ausgeliehenen Filme anzeigt. Dies ist relevant für die spätere Rückgabe des Films.

Beim Filmverleih ist dann die Länge des Verleihs und der zu zahlende Preis ersichtlich. Des Weiteren soll bei der Filmrückgabe überprüft werden, ob der Film pünktlich zurückgegeben wurde.

Optional soll ebenso möglich sein, Filmempfehlungen nach Genre wiederzugeben und Informationen, des zu verleihenden Films, anzuzeigen (wie z.B. FSK Rating, Filmlänge, Bewertungen, etc.). Weitere Optionen wie Rabattaktionen für Kunden oder Kunden, die regelmäßig ihre Ausleihzeit überziehen, entsprechend in eine Blacklist zu setzen, sind ebenfalls geplant.

1.3 Systemkontext



1.4 Stakeholder

Für dieses Projekt haben wir unser Team so organisiert

- Philipp Luis Elgert : Projekt Manager
- Dahun Lee: Projektassistenz
- Laura Salazar López: Software Architect
- Fabius Pudlich: Entwickler, (Server-Administrator)
- Samira Joulaei Vakili: Entwicklerin
- Timo Trauschold: Entwickler

Zudem sind zwei weitere Stakeholder zu benennen - der **Endanwender** und der **Geschäftsführer**. Der Endanwender, in dem Fall der Mitarbeiter, hat die Rolle des Benutzers, d.h. er ist derjenige, der die Seite aufruft und letztendlich verwendet. Der Geschäftsführer besitzt die Rolle des Admins. Dies bedeutet, dass der Geschäftsführer von der Architektur überzeugt werden muss und ist somit ebenfalls der Arbeitgeber des Projekts.

1.5 Verwendete Technologien

Für das Projekt benutzten wir folgende Technologien:

- HTML (Hypertext Markup Language)
- Javascript
- CSS (Cascading Style Sheets)
- Node JS

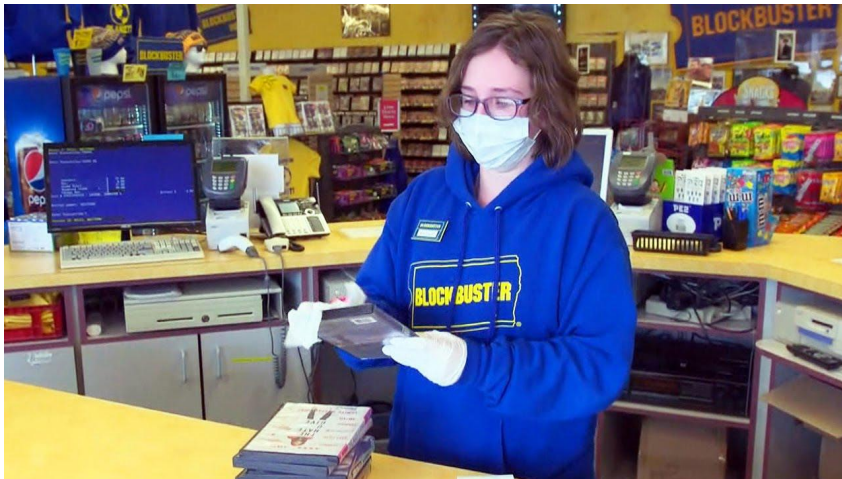
1.6 Rahmenbedingungen

Das Projekt muss bis zum 17.04.2022 fertiggestellt sein. Aus technischer Sicht wurde auf eine englische Namenskonvention vereinbart, um einen einheitlichen Code zu gewährleisten.

2 Anforderungen

2.1 Akteure

User Persona - Mitarbeiter



(1)

Quelle : <https://i.ytimg.com/vi/jJMEPHNjHE0/maxresdefault.jpg>

Eigenschaften

Zwischen 18-40 Jahren alt

Möchten zuverlässig Filme ausleihen

Möchten den Kunden nicht zu lang warten lassen

Frustrationen

Möchten nicht zu oft ins Lager gehen um Manuelle zu schauen ob die Filme da sind

Möchten gerne nicht so viel Rechnen

Sie wollen gerne aufhören ein Papier formular ausfüllen

Nutzung Überwindung

Bedienungsfreundlich

Abbau von extra Arbeit

Verbesserung des IST-Zustands

Spezifische Ziele

Verlässliche Informationen

Einfache Bedienung der Software

Allgemeine Ziele

Möchte Kunden schneller abfertigen

Möchte gerne weniger machen

5-7/
10

Tech-savvy

2.2 Funktionale Anforderungen

Funktion	Beschreibung	Priorität
/ LF 10 /	Filmsuche	Hoch
/ LF 20 /	Kunden-Konten erstellen	Hoch
/ LF 30 /	Kunden-Konten History einsehbar	Hoch
/ LF 40 /	Preis berechnen (Bußgeld bei überziehen)	Hoch
/ LF 50 /	Verleih aufträge erstellen	Hoch
/ LF 60 /	Kundensuche	Hoch
/LF 70/	Warenkorb	Hoch

2.3 Qualitätsanforderungen

Für die Qualitätsanforderungen wurde eine Tabelle angefertigt, die darstellt welche Leistungen von besonderer Relevanz ist und wie wichtig die jeweiligen Funktionen sind:

Leistung	Sehr Gut	Gut	Normal	Irrelevant
Funktionalität	X			
Zuverlässigkeit	X			
Benutzbarkeit		X		
Effizienz		X		
Änderbarkeit			X	
Sicherheit				X
Portierbarkeit				X

Funktionalität:

Das Programm soll eine sehr gute Funktionalität beinhalten. Sollten eventuelle Fehler auftreten, würde die Schnelligkeit und die generelle Aufgabe des Programms behindern. Wie wir herausgestellt haben, ist die Schnelligkeit (siehe 1.1) eines der wichtigsten Aspekte, die wir umsetzen wollen. In dem Aspekt wollen wir sicherstellen, dass die Seite mit einer Ladezeit von 200 Millisekunden zur Verfügung steht.

Zuverlässigkeit:

Mit der gleichen Begründung soll ebenfalls eine sehr gute Zuverlässigkeit gegeben sein. Der Mitarbeiter einer Filmbibliothek soll zuverlässig die gewünschten Daten erhalten, um dann diese dem Kunden bereitzustellen. Dabei sollten so wenig Fehler wie möglich entstehen. Beispielsweise soll der Server höchstens einmal im Monat ausfallen, um eventuelle Aktualisierungen durchzuführen.

Benutzbarkeit:

Der Grund, warum wir uns für eine gute Benutzbarkeit entschieden haben ist, dass das Programm allein von dem Mitarbeiter verwendet wird. Allerdings ist trotzdem eine gute Bedienfreundlichkeit substanziell, um die Simplizität (siehe 1.1) zu gewährleisten.

Effizienz:

Aus den Aspekten *Simplizität* und *Schnelligkeit* ergeben sich ebenso eine gute Effizienz.

Änderbarkeit:

Wie in 1.2 beschrieben, besitzen wir eine Reihe von optionalen Use Cases, die wir im Projekt vorgenommen haben. Deshalb ist die Änderbarkeit unseres Programms als "normal" einzustufen

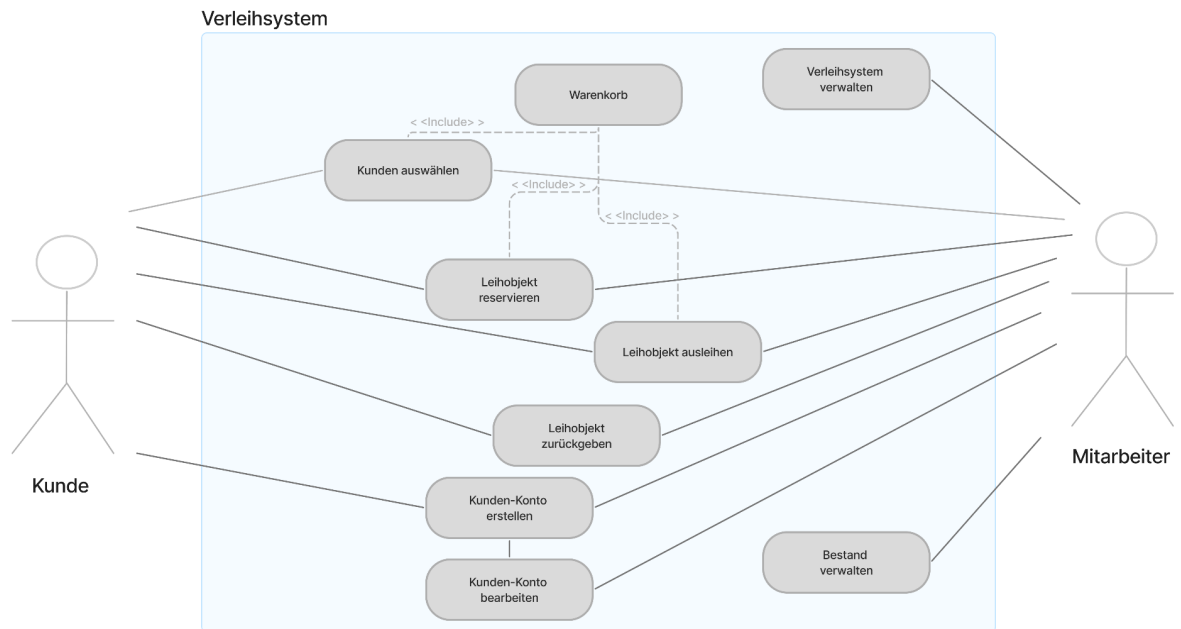
Sicherheit:

Die Sicherheit in unserem Programm ist weniger relevant als die weiteren Leistungen, da das Programm intern von den Mitarbeitern verwendet wird. Sicherheitslücken werden demnach vorhanden sein, ungeachtet dessen werden diese als optional betrachtet.

Portierbarkeit:

Die Portierbarkeit ist in unserem Fall irrelevant, da unsere Anwendung am Kassenschalter fest integriert und somit der Endanwender im Grunde nur der Mitarbeiter ist.

3.1 Use-Case Modell



3.2 Anwendungsfall (Use-Cases)

Use-Case-Spezifikation 1

Bezeichner	/ LF 10 / Filmsuche
Autoren	Philipp Elgert
Priorität	Hoch
Kritikalität	Sehr Kritisch
Beschreibung	Funktion, wo ein Name eingeben wird und dann die Datenbank nach Übereinstimmung abgesucht wird
Auslösendes Ereignis	Eingabe im Suchfeld
Akteure	Mitarbeiter
Vorbedingung	Das Suchfeld muss ausgewählt sein und der Mitarbeiter muss im Filmbereich sein
Ergebnis	Ergebnisse werden in einem Drop-down Menü angezeigt

Use-Case-Spezifikation 2

Bezeichner	/ LF 20 / Kunden-Konten erstellen
Autoren	Philipp Elgert
Priorität	Hoch
Kritikalität	Kritisch
Beschreibung	<p>Dem Mitarbeiter werden zwei Eingabefelder gegeben, wo die Daten des Kunden eingetragen werden</p> <p>Zum Beenden des Erstellens gibt es ein "Fertig" Button</p>
Auslösendes Ereignis	Der Benutzer drückt auf den "Kunden anlegen" Button
Akteure	Mitarbeiter
Vorbedingung	Der Mitarbeiter muss im Kundenbereich sein
Ergebnis	Ein neuer "Kunde" mit den Werten "Name", "Vorname" und "Kundennummer" wird in die Datenbank eingefügt

Use-Case-Spezifikation 3

Bezeichner	/ LF 30 / Kunden-Konten History einsehbar
Autoren	Philipp Elgert
Priorität	Hoch
Kritikalität	Kritisch
Beschreibung	<p>Der Mitarbeiter kann die History von einem Kunden einsehen</p> <p>Es wird angezeigt, welchen Film sie hatten, das Ausleihdatum und die Rückgabe</p>
Auslösendes Ereignis	Der Kunde wurde gesucht und ausgewählt
Akteure	Mitarbeiter
Vorbedingung	Der Kunde muss angelegt sein und der Mitarbeiter muss im Kundenbereich sein
Ergebnis	Die Kundendaten werden angezeigt mit der History was sie ausgeliehen haben

Use-Case-Spezifikation 4

Bezeichner	/ LF 40 / Preis berechnen
Autoren	Dahun Lee
Priorität	Hoch
Kritikalität	Sehr Kritisch
Beschreibung	Im Warenkorb wird der Preis berechnet und angezeigt
Auslösendes Ereignis	Wenn das gewählte Produkt ausgeliehen wird (bzw. in den Warenkorb gelegt wird)
Akteure	Mitarbeiter
Vorbedingung	Ein Film, Kunde und Ausleihdauer müssen gewählt und eingetragen werden
Ergebnis	Der Preis wird berechnet

Use-Case-Spezifikation 5

Bezeichner	/ LF 50 / Verleih Aufträge erstellen
Autoren	Philipp Elgert
Priorität	Hoch
Kritikalität	Sehr Kritisch
Beschreibung	Nach dem Checkout wird dem Kunden dem Film zugeteilt mit dem Ausleihdatum
Auslösendes Ereignis	Ein Button mit der Beschriftung "Checkout" wurde gedrückt
Akteure	Mitarbeiter
Vorbedingung	Es müssen alle Parameter des Warenkorbs ausgefüllt sein
Ergebnis	Dem Kunden wird der Film zugeteilt

Use-Case-Spezifikation 6

Bezeichner	/ LF 60 / Kundensuche
Autoren	Philipp Elgert
Priorität	Hoch
Kritikalität	Kritisch
Beschreibung	<p>Der Mitarbeiter kann Kunden im Kundenbereich suchen und auswählen</p> <p>Die Kunden können auch im Warenkorb gesucht und ausgewählt werden</p>
Auslösendes Ereignis	Der Mitarbeiter
Vorbedingung	Der Warenkorb muss offen sein
Ergebnis	Der Kunde wird angezeigt und kann ausgewählt werden

Use-Case-Spezifikation 7

Bezeichner	/ LF 70 / Warenkorb
Autoren	Philipp Elgert
Priorität	Hoch
Kritikalität	Kritisch
Beschreibung	In den Warenkorb werden Filme eingefügt und dem Kunden zugeordnet
Auslösendes Ereignis	Der Mitarbeiter drückt den "Warenkorb" Button
Nachbedingung	Es gibt eine Meldung und die Start-Seite wird geladen
Ergebnis	Der Warenkorb kann bearbeitet (bspw. Film hinzufügen) werden

4 Details

4.1 Gui-Beschreibung und Mock-Ups

Logo	Suchleiste	neuer Kunde	Waren- korb	Wechsel zum Kundenbereich									
<div>"genre": Action</div> <table border="1"><tr><td>Vorschlag 1 "title"</td><td>Vorschlag 2 "title"</td><td>Vorschlag 3 "title"</td></tr></table> <div>"genre": Drama</div> <table border="1"><tr><td>Vorschlag 1 "title"</td><td>Vorschlag 2 "title"</td><td>Vorschlag 3 "title"</td></tr></table> <div>"genre": Science Fiction</div> <table border="1"><tr><td>Vorschlag 1 "title"</td><td>Vorschlag 2 "title"</td><td>Vorschlag 3 "title"</td></tr></table>					Vorschlag 1 "title"	Vorschlag 2 "title"	Vorschlag 3 "title"	Vorschlag 1 "title"	Vorschlag 2 "title"	Vorschlag 3 "title"	Vorschlag 1 "title"	Vorschlag 2 "title"	Vorschlag 3 "title"
Vorschlag 1 "title"	Vorschlag 2 "title"	Vorschlag 3 "title"											
Vorschlag 1 "title"	Vorschlag 2 "title"	Vorschlag 3 "title"											
Vorschlag 1 "title"	Vorschlag 2 "title"	Vorschlag 3 "title"											

Abb. 1 - Mock-Up Startseite

Beginnend mit dem Logo, der sich links oben auf der Seite befindet, ist anklickbar und führt zurück auf die Startseite. Daneben befindet sich die Suchleiste, in der Filme eingegeben werden kann, um dann schließlich Filme auszuleihen. Trotzdem bietet die Seite Vorschläge an, die nach einem bestimmten Genre sortiert sind, und zeigt diese an. Die Vorschläge sind jeweils anklickbar.

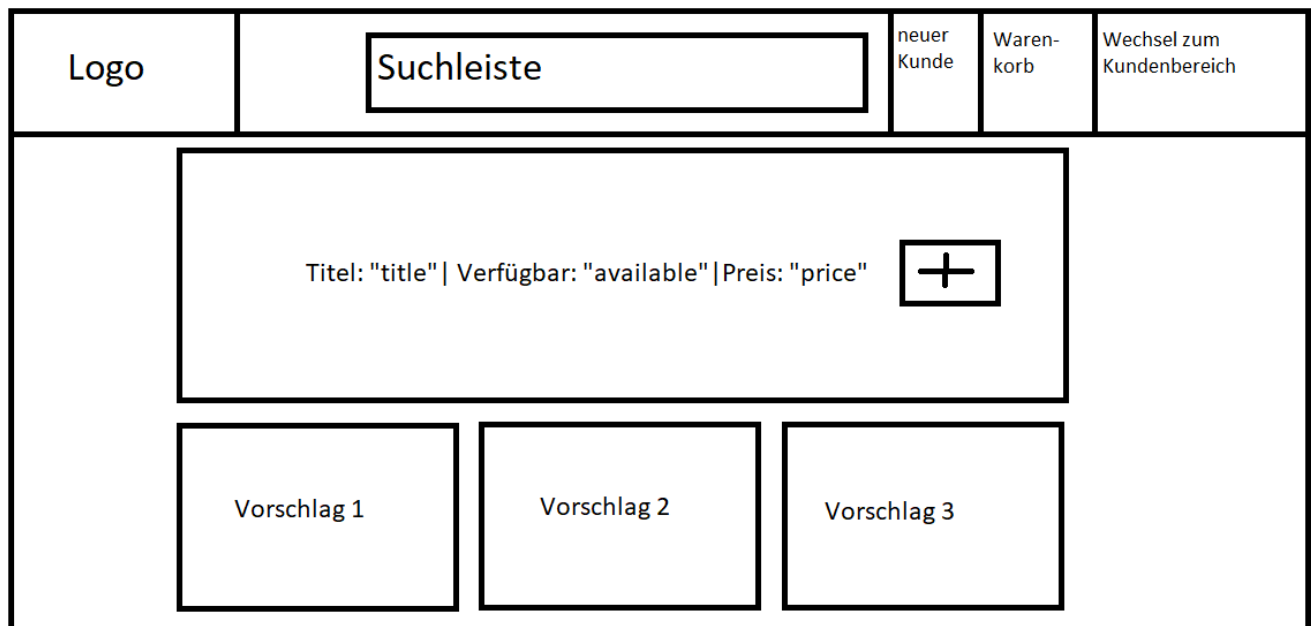


Abb. 2 - MockUp Filmverleih

Die verfügbaren Filme werden in der Mitte der Startseite angezeigt, wenn beispielsweise ein Vorschlag (von Abb. 1) angeklickt wird. Diese haben jeweils einen “+”-Button, um die gewünschten Filme in den Warenkorb zu legen. Des Weiteren werden unter dem ausleihenden Film weitere Vorschläge präsentiert, die ebenso anklickbar sind.

Logo	Suchleiste	neuer Kunde	Waren-korb	Wechsel zum Kundenbereich
				<div>Kundensuche</div> <div>"surname" "cNumber"</div> <div>"title" <input type="text"/></div> <div>Checkout</div>

Abb. 3 - MockUp Warenkorb

Der Warenkorb, gekennzeichnet durch ein klassisches Warenkorb-Logo, ist dabei aufklappbar (rechts auf der Seite). Im aufgeklappten Zustand befinden sich dort die gewünschten Filme, die jeweils durch den “-”-Button vom Warenkorb entfernt werden können. Oberhalb kann man den jeweiligen Kunden auswählen. Der Checkout-Button finalisiert den Einkauf.

Logo	Suchleiste	neuer Kunde	Waren- korb	Wechsel zum Kundenbereich
<div> Meldung : Film ausgeliehen Preis : <div>Weiter</div> </div>				<div>Kundensuche</div> <div>"surname" "cNumber"</div> <div>"title" <input type="text"/></div> <div>Checkout</div>

Abb. 4 - MockUp Meldung (Film ausgeliehen)

Wird nun der Checkout-Button gedrückt, wird eine Meldung in der Mitte der Seite angezeigt. Dabei wird einem benachrichtigt, dass die ausgewählten Filme endgültig ausgeliehen sind. Mit einem weiteren Button "Weiter" wird die Seite neu geladen und es wird zu der Startseite gewechselt.

Logo	Suchleiste	Neuer Kunde	Warenkorb	Wechsel zum Kundenbereich
------	------------	-------------	-----------	---------------------------

<div> <div>"forename" "surname""cNumber"</div> <table> <tr> <td>Filme</td> <td>ID</td> <td>Ausleihdatum</td> <td>Rückgabe</td> </tr> <tr> <td>"title"</td> <td>"movieID"</td> <td>"rentTime"</td> <td>checkbox</td> </tr> </table> <div>Rückgabe</div> </div>					Filme	ID	Ausleihdatum	Rückgabe	"title"	"movieID"	"rentTime"	checkbox
Filme	ID	Ausleihdatum	Rückgabe									
"title"	"movieID"	"rentTime"	checkbox									

Abb. 5 - MockUp Kundenbereich

Rechts oben auf der Startseite ist ein klickbarer Textfeld "Kunde", der zu einem Kundenbereich wechselt. In diesem Bereich kann man in der Suchleiste nach einem Kunden suchen. Wählt man nun den jeweiligen Kunden aus, werden die Filme, die diese Person bereits ausgeliehen hat, angezeigt. In der Rückgabe-Spalte kann man nun, einen Haken ankreuzen, um den Film zurückzugeben.

Logo	Suchleiste	neuer Kunde	Warenkorb	Wechsel zum Kundenbereich
------	------------	-------------	-----------	---------------------------

Neuer Kunde

Vorname:

Nachname:

Submit

Abb. 6 - MockUp Kundenerstellung

Wenn man nun den Button “Neuer Kunde” anklickt, kommt man zu dieser Abbildung. Dabei kann man einen neuen Kunden in das System registrieren. Man gibt den Vor- und Nachnamen in das Textfeld ein und beendet die Registrierung mit dem “Submit”-Button.

4.2 Datenbank

Für die Datenbank (JSON) sollten folgende Daten aufgelistet sein:

Tabelle	Inhalt
Filmdaten	<ul style="list-style-type: none">• Titel (Name des Films)• Menge (Wie viele Filme verfügbar sind)• Preis (Ausleihpreis)
Kundendaten	<ul style="list-style-type: none">• Vorname• Nachname• Kundennummer

Dabei erhält die Datenbank von den jeweiligen Skripte Daten, die in der Datenbank gespeichert werden, wie z.B. nach einer Ausleihe eines bestimmten Films, die Menge der verfügbaren Filme um die ausgeliehene Anzahl reduzieren.

5 Glossar

Begriff	Definition
Button	Eine englische Übersetzung für das deutsche Wort Knopf (oder Schaltfläche). Der Button ist ein Bedienelement in grafischen Benutzeroberflächen, das dem Benutzer ermöglicht, eine zugeordnete Funktion auszulösen
Dropdown-(Menü)	Das Dropdown-Menü ist eine spezielle Form des Auswahlmenüs. Es erscheint eine Auswahlliste auf dem Bildschirm.
Element	Grundbestandteil oder Komponenten, aus dem mit anderen zusammen etwas aufgebaut wird.
Frustrationen	Was die Arbeiter an ihrem System stört
History	Mit der History ist eine Chronik oder ein Verlauf gemeint: Eine Liste von zuletzt durchgeführten Aktionen.
Main-Page	Die Main-Page ist die Startseite des Programms.
Pop-Up (Fenster)	Pop-Up Fenster sind Bedienfelder, die sich automatisch öffnen, und der Benutzer einen Text (oder ähnliche Elemente) gezeigt bekommt.
Skript	Ein Skript ist eine kurze Abfolge von Befehlen oder Funktionen, die sich in einer Webseite integrieren lässt.
Tech-savvy	Die englische Übersetzung für "technisch versiert", d.h. großes Wissen über die Technologie besitzen
webbasiert	Aus dem World Wide Web beruhend bzw. mit ihm arbeitend

6 Quellen

Informationsquellen:

Softwarespezifikation

Link:

https://moodle.thm.de/pluginfile.php/737201/mod_resource/content/0/06_Anforderungsspezifikation.pdf

W3Schools

Link:

<https://www.w3schools.com/>

Bildquellen :

(1) World's Last Blockbuster Video Is Still Going Strong

Videolink:

<https://www.youtube.com/watch?v=jJMEPHNjHE0>

Bildlink:

<https://i.ytimg.com/vi/jJMEPHNjHE0/maxresdefault.jpg>