

UEK335_Dokumentation

Thema: Movie-Details-Checker

Autoreninformationen

Autoren: Kristijan Lisica, Maximilian Nöthe, Yara Cordero Ortiz

Inhaltsverzeichnis

1	Einleitung.....	3
1.1	Entwickler der Movie Details Checker App	3
1.2	Ausgangslage	3
1.3	Hauptziele der Applikation	3
1.4	Coding Guidelines.....	4
2	User Story.....	5
2.1	Homepage besuchen.....	5
2.2	Registrierung und Login.....	6
2.3	Random Movie und App stats sehen	7
2.4	Navigation.....	8
2.5	MoviePage besuchen und Movies durch Scrollen sehen.....	9
2.6	Filtern von A - Z und Z – A.....	10
2.7	Movie erstellen.....	11
2.8	Detail Ansicht eines Movie ansehen	12
2.9	Movie bearbeiten	13
2.10	Movie löschen	14
2.11	Aktuellen User sehen.....	15
3	Persona	16
4	Architektur und Design	17
4.1	Architektur.....	17
4.2	Design.....	18
5	Technischer Bericht	18
6	Testing.....	19

1 Einleitung

1.1 Entwickler der Movie Details Checker App



Yara Cordero Ortiz



Kristijan Lisica



Maximilian Nöthe

1.2 Ausgangslage

Unser Entwicklerteam wurde beauftragt, eine Mobile-App für die MovieCruxStudios zu realisieren. Diese App soll eine übersichtliche Darstellung der Filme des Studios bieten und detaillierte Informationen zu jedem Film anzeigen. Zusätzlich soll es möglich sein, neue Filme hinzuzufügen sowie bestehende Einträge zu aktualisieren oder zu löschen. Ziel ist eine benutzerfreundliche App, die die Filmdaten einfach verwaltbar macht und auf den gängigen mobilen Plattformen verfügbar ist.

Um diese Applikation zu entwerfen werden Technologien und Frameworks genutzt wie zum Beispiel:

- React Native
- React Native Paper
- Expo
- Expo Go
- Metro Bundler
- Yarn
- TypeScript

1.3 Hauptziele der Applikation

- **Filmdaten übersichtlich darstellen und verwalten:**
Die App bietet eine Übersicht aller Filme und ermöglicht das Hinzufügen, Bearbeiten und Löschen von Filmen.
- **Benutzerverwaltung:**
Users können sich registrieren, einloggen und ihr Profil verwalten.

- **Navigation und Benutzerfreundlichkeit:**
Einfache Navigation zwischen den Seiten und Filteroptionen für die Filmsuche.

1.4 Coding Guidelines

- **Namenskonventionen:**

Ordner: camelCase

Files: PascalCase

Variablen: camelCase

Funktionen: camelCase

- **Code-Struktur und Formatierung:** Einheitliche Einrückung Mithilfe von Prettier extension.
- **Dokumentation und Kommentare:** TSDocs für Kommentare im Code.
- **Branch-Naming & -Strategie:**

Naming:

Features: features/implementiertesFeature

Develop: develop

Main: main

Strategie:

Features, welche für die Applikation implementiert werden, werden in Form des Feature-Naming als Branch hochgeladen. Alle Veränderungen, die für die Implementation des Features benötigt werden, werden im erstellten Feature-Branch gemacht.

Diese Feature-Branche werden, sobald sie funktionsfähig und problemlos sind, in den Develop-Branch gemerged.

Wenn die komplette Applikation problemlos funktioniert wird der Develop-Branch in den Main-Branch gemerged für die finale Abgabe.

2 User Story

Folgenden User Stories widerspiegeln die wesentlichen Bedürfnisse und Anforderungen der Benutzer und teilen die Entwicklung unserer Applikation in sinnvolle Abschnitte.

2.1 Homepage besuchen

A	Homepage besuchen	3
<div><div>Ich als:</div><div>Mitarbeiter von MovieCruxStudios</div><div>möchte:</div><div>auf die Homepage zugreifen</div><div>um:</div><div>eine kurze Übersicht der App zu sehen (Text anpassen)</div></div>		
<div>Die User Story ist fertig wenn:</div> <ul style="list-style-type: none">• Die Installationsanleitung der App vorhanden ist• Der User sich registrieren kann• Der User sich anmelden kann		

2.2 Registrierung und Login

A

Registrierung und Login

4

Ich als: Mitarbeiter von MovieCruxStudios
möchte: einen Account erstellen und damit einloggen
um: die Applikation nutzen zu können

Die User Story ist fertig wenn:

- Die Loginpage ersichtlich ist
- Die Registrierungspage ersichtlich ist
- Die Registrierungsfunktion problemlos funktioniert
- Die Loginfunktion problemlos funktioniert

2.3 Random Movie und App stats sehen

A**Random Movie und App stats sehen****4**

Ich als: Mitarbeiter von MovieCruxStudios

möchte: Einen Random Movie und die Stats der App auf der Homepage sehen

um: eine ungefähre Ahnung zu haben was auf mich zukommt

Die User Story ist fertig wenn:

- Die Randomizer-Funktion problemlos funktioniert und bei Knopfdruck einen Random Movie der Liste anzeigt
- Die Stats (Anzahl Bücher) der gesamten App problemlos angezeigt werden können

2.4 Navigation

A

Navigation

2

Ich als: Mitarbeiter von MovieCruxStudios
möchte: Mithilfe der Navbar durch die verschiedenen Pages der App navigieren
um: weitere Funktionen der App zu benutzen und mehr Infos zu sehen

Die User Story ist fertig wenn:

- Die Navbar 3 Icons (Movie, Home & Profile) besitzt und am bottom gesetzt festgelegt ist
- Durch Klick auf ein Icon zur entsprechenden Page weitergeleitet wird

2.5 MoviePage besuchen und Movies durch Scrollen sehen

A**MoviePage besuchen und Movies durch scrollen
sehen****3**

Ich als: Mitarbeiter von MovieCruxStudios

möchte: Auf die MoviePage zugreifen und alle vorhandenen Movies durch scrollen sehen

um: zu wissen was für Movies alles auf der App zur verfügung gestellt werden

Die User Story ist fertig wenn:

- Die Verlinkung zur MoviePage problemlos funktioniert
- Alle Movies, welche zur verfügung stehen, auf der MoviePage angezeigt werden (Pagination?)
- Die einzelne Card eines Movies sehen, welche kurze Informationen zum Movie beinhaltet
- Durch scrollen die verschiedenen Optionen von Movies sehen

2.6 Filtern von A - Z und Z – A

A**Filtern von A - Z und Z - A****2**

Ich als: Mitarbeiter von MovieCruxStudios

möchte: Die Movies auf der MoviePage aufwärts und abwärts alphabetisch ordnen

um: Movies auf eine andere Art zu suchen

Die User Story ist fertig wenn:

- Die verfügbaren Movies auf der MoviePage angezeigt werden
- Der Filterbutton auf der MoviePage angezeigt wird
- Die Movies aufwärts alphabetisch gefiltert werden (Mit A beginnen)
- Die Movies abwärts alphabetisch gefiltert werden (Mit Z beginnen)

2.7 Movie erstellen

A**Movie erstellen****4**

Ich als: Mitarbeiter von MovieCruxStudios

möchte: Mithilfe des Addbuttons neue Movies erstellen damit diese in der Liste sichtbar sind

um: Neue Movies, die ich sehen möchte, zur Liste hinzuzufügen

Die User Story ist fertig wenn:

- Der Addbutton auf der MoviePage sichtbar ist
- Durch das Klicken des Addbutton zur CreateMoviePage navigiert wird
- Ich Informationen zum Movie eingeben kann
- Die Informationen abgespeichert werden
- Nach dem Abspeichern der Informationen zurück zur MoviePage navigiert wird

2.8 Detail Ansicht eines Movie ansehen

A**Detail Ansicht eines Movie ansehen****3**

Ich als: Mitarbeiter von MovieCruxStudios
möchte: Details zu einem spezifischen Movie sehen
um: eine Übersicht des Filmes zu haben

Die User Story ist fertig wenn:

- Ich auf eine Moviecard klicken kann
- Dies mich auf die MovieDetailPage eines Movies weiterleitet
- Die Informationen eines Movies vorhanden sind und übersichtlich angezeigt werden

2.9 Movie bearbeiten

A**Movie bearbeiten****4**

Ich als: Mitarbeiter von MovieCruxStudios
möchte: Einen bestehenden Movie bearbeiten
um: Fehler zu beheben oder Anpassungen vorzunehmen

Die User Story ist fertig wenn:

- Es einen Editbutton auf der DetailedMoviePage hat
- Der Editbutton auf die EditMoviePage navigiert
- Die bestehenden Daten des Movies vorhanden und bearbeitbar sind
- Die neuen Daten abgespeichert werden
- Nach dem Editieren eines Movies zurück zur MovieDetailedPage navigiert wird

2.10 Movie löschen

A**Movie löschen****3**

Ich als: Mitarbeiter von MovieCruxStudios
möchte: Einen bestehende Movie löschen
um: movies, die nicht in der Liste sein sollten zu entfernen

Die User Story ist fertig wenn:

- Ein Deletebutton auf der DetailedMoviePage vorhanden ist
- Durch Klicken des Deletebuttons der bestehende Movie gelöscht wird
- Nach dem Löschen des Movies zurück zur MoviePage navigiert wird
- Auf der MoviePage der gelöschte Movie nicht mehr zu sehen ist

2.11 Aktuellen User sehen

A**Aktueller User sehen****2**


Ich als: Mitarbeiter von MovieCruxStudios
möchte: Infos des aktuellen User der App sehen
um: zu wissen wer aktuell die App nutzt

Die User Story ist fertig wenn:

- Durch das User-Icon in der Navbar richtig zum UserProfilePage navigiert wird
- Die richtigen Informationen des aktuellen Users zu sehen sind

3 Persona

Um die Bedürfnisse unserer Zielgruppe besser zu verstehen haben wir eine Persona erstellt damit diese uns dabei hilft.

Lisa Fischer	
Alter	28 Jahre
Beruf	Marketing-Managerin und Film-Enthusiastin
Interessen	Lisa ist eine begeisterte Kinogängerin und interessiert sich für alles rund um Filme und Filmproduktion. Sie liebt es, neue Filme zu entdecken und ihre Kenntnisse über Regisseure und Genres zu erweitern. Da sie als Marketing-Managerin oft wenig Zeit hat, braucht sie eine App, die ihr eine schnelle und umfassende Übersicht über alle Filme bietet. Sie schätzt gut strukturierte, visuell ansprechende Apps.
Stärken	Lisa ist analytisch, kreativ und hat ein Interesse an Design und Benutzererfahrung. Ihre Fähigkeit, schnell relevante Informationen zu finden und zu verarbeiten, hilft ihr sowohl beruflich als auch privat. Sie erwartet, dass Apps einfach zu bedienen sind.
Ziele	Lisa möchte eine App, die ihr hilft, sich über die neuesten und beliebtesten Filme eines Studios zu informieren. Ihr Ziel ist es, eine Plattform zu nutzen, die aktuelle Informationen bietet und es ihr ermöglicht, Filme in ihrer Freizeit schnell zu finden und darüber mit Freunden zu sprechen.
Nationalität	Deutschland

4 Architektur und Design

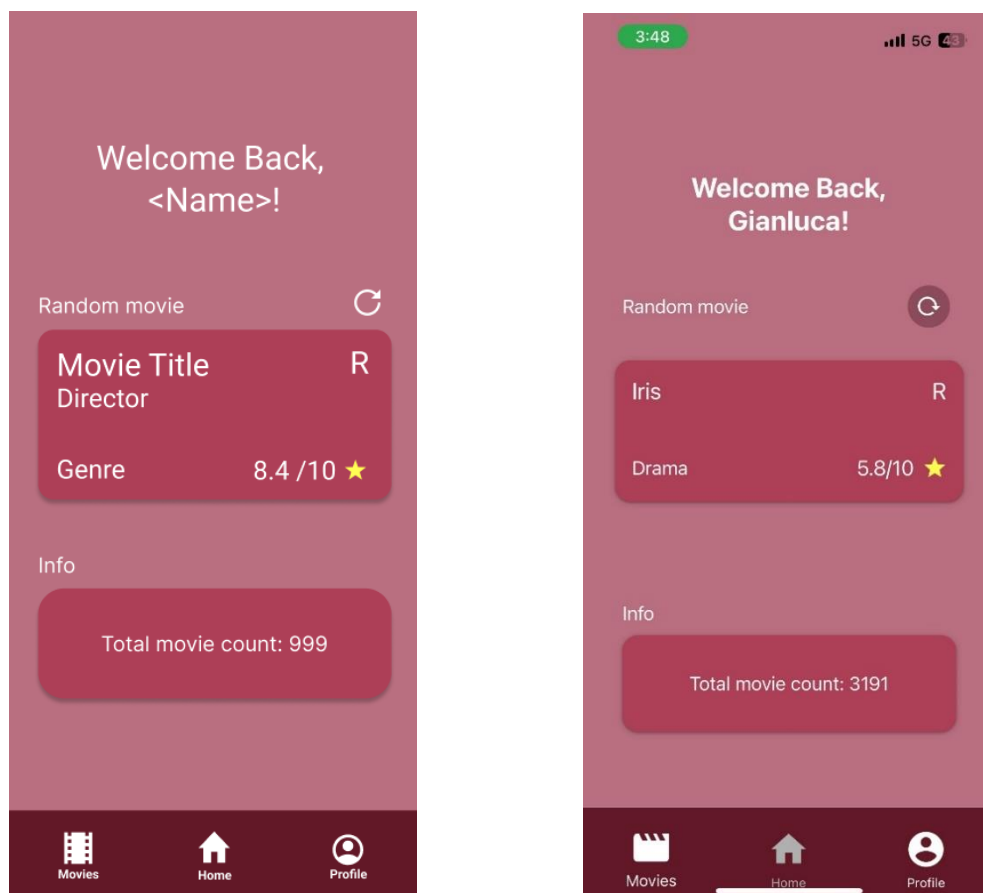
4.1 Architektur

Unsere Mobile App wird in diesen folgenden Seiten unterteilt:

- LoginPage
- RegisterPage
- HomePage
- MoviePage
- MovieDetailPage
- CreateMoviePage
- MovieEditPage
- UserProfile

Das Gestalten unserer Applikation haben wir mithilfe von Figma erarbeitet. Unter diesem Link kann man unser Wireframe wie auch unser Prototype ansehen: [Link](#)

Unser Ziel war es die Applikation anhand der erstellten Mockups nachzubilden. Im Beispiel sieht man die Mockupversion (links) wie auch die Offizielle Version (rechts) der Homepage:

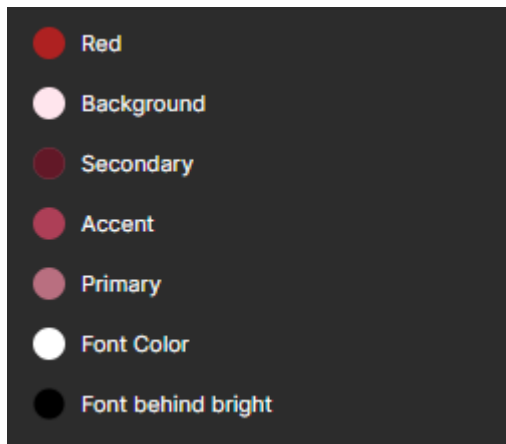


4.2 Design

4.2.1 Farben

Wir haben uns für diese Farben entschieden, da wir einen Netflix ähnlichen Look imitieren wollten. Die Farben entsprechen nicht zu 100% an Netflix's Design, dies ist aber mit Absicht so gewählt worden. Die Kombination von verschiedenen Rottönen vermittelt eine bestimmte Markenidentität die sowohl als kreativ aber auch minimalistic angesehen wird.

Die genauen Details zu den Farben sind im Figma zu finden.



5 Technischer Bericht

Das Testing für diese Applikation wird durch manuelle Tests erstellt. Diese Tests werden die CRUD-Funktionen des Backends testen und schauen, ob diese auch unsere erwarteten Ergebnisse zeigen.

Spezifisch für diese Applikation werden wir diese Komponenten im Frontend so ausführlich testen wie möglich. Zuerst testen wir das Authentifizieren des Users und die erwartete Page, die nachfolgend erscheinen sollte. Die Navbar unten sollte auch vorhanden sein welche den User in die verschiedenen Pages der Applikation navigiert. Die einzelnen Funktionen und Navigation der Pages sollten getestet werden.

Die Applikation kann in der Zukunft mit mehreren Funktionen und Pages weitergeführt werden. Eine Option wäre es die Daten in der Datenbank ausführlicher zu machen. Zurzeit haben nicht alle Objekte ausgefüllte Zeilen, welches zeigt das es an zu wenig Daten hat. Man könnte auch zu führend die Film Posters der respektiven Filme in der Datenbank hinterlegen, welches dann später auch in der Applikation aufgerufen werden kann. Im Frontend kann zukünftig mit dem Poster, die MoviePage verbessert werden. Die MovieCards können dann die Poster enthalten, dies macht es für den User viel einfacher einen Film zu erkennen als den Titel lesen zu müssen.

6 Testing

Test Protocol									
TestCaseld	Component	Priorit y	Description/ Test summary	Pre-requisite s	Test Steps	Expec ted Result	Actu al Res ult	Stat us	Test Exec uted By
Homepage_01	HomePage (Welcome Text, Random Movie, Movie Count)	P1	A user should be able to see the Homepage after successfully login in or registering.	Login or registratio n capability	1.Open the App and login or register themselves 2.Have the App open on the homepage, after login.	For the Homepage to be visible after registering or logging themselves in.	Loggin g in works , Home page is visible	PASS	Max
LoginUnd Registrierung_01	LoginPage (2_Textfield, Login-Button, Navigation Text)	P0	A user should be able to Register and Login using the new credentials	Login or registratio n capability	1. Open the app and register themselves 2.Login to the app using the new credentials	For the inputted user data to be saved to the database and for the user to be able to login themselves	Regis tering works , loggin g in too	PASS	Max
RandomMovie UndAppStats_01	HomePage (Randomizer Button , MovieCard, Statcard)	P2	A user should be able to randomize the Movie Card and see the current amount of saved Movies in the DB.	Login or registratio n capability	1.User must login or register. 2.The User should click on the random button. At the same time it should see the stats which are right below it.	The MovieC ard gets updated with a new Movie. Addition ally the stats below it should show a number.	Movie Card is updat ed and stats are show n	PASS	Yara
Navigation_01	Navbar (Navigation zu HomePage, MoviePage, UserProfilePage)	P1	A user should be free to navigate with the Navbar at the bottom and go to new Pages.	Login or registratio n capability	1.User must login or register themselves. 2. On the bottom there is a Navbar and the different Icons can be clicked to navigate to said pages.	The Navigati on should navigat e to a page with the same name as the bottom Navbar title has.	Navb ar is worki ng and user is navig ated to corres pondi ng page	PASS	Yara

Movie-Details-Checker

MoviePage BesuchenUnd MoviesScrollen_01			MoviePage (SortButton Multiple MovieCard)	P2	A user should go to the MoviePage and scroll through all available movies.	Login or registration capability	1. User must login or register themselves. 2. User navigates to the MoviePage. 3. The user can scroll up and down the page.	The user can scroll up and down the MoviePage, with all MovieData still present.	Movie Data appears but is not scrollable.	FAIL	Kristijan
MoviePageBesuchenUnd MoviesScrollen_01			MoviePage (SortButton Multiple MovieCard)	P2	A user should go to the MoviePage and scroll through all available movies.	Login or registration capability	1. User must login or register themselves. 2. User navigates to the MoviePage. 3. The user can scroll up and down the page.	The user can scroll up and down the MoviePage, with all MovieData still present.	Movie Page is scrollable, with all data present	PASS	Kristijan
FilternVonA-ZUndZ-A_01			MoviePage (SortButton Multiple MovieCard)	P3	A user should go to the MoviePage and press the sort button, which will sort the movies from a-z and z-a.	Login or registration capability	1. User must login or register themselves. 2. User navigates to the MoviePage. 3. The user can click the sort button.	The MovieCards should be sorted from either a-z or z-a, depending on the icon shown on the sort button.	No Movie Cards appear	FAIL	Kristijan
FilternVonA-ZUndZ-A_01	MoviePage (SortButton Multiple MovieCard)	P3	A user should go to the MoviePage and press the sort button, which will sort the movies from a-z and z-a.	Login or registration capability	1. User must login or register themselves. 2. User navigates to the MoviePage. 3. The user can click the sort button.	The MovieCards should be sorted from either a-z or z-a, depending on the icon shown on the sort button.	MovieCards are sorted a-z, z-a also working	PASS		Kristijan	

Movie-Details-Checker

MovieErstellen_01	CreateMoviePage (8_InputFields, CancelButton, CreateButton)	P2	A user should be able to create a new movie and add it into the database	Login or registration capability, Movie access capability	1. User must login or register themselves. 2. User navigates to the MoviePage. 3. The user must click on the 'plus' symbol and be navigated to the CreateMoviePage. 4. The user must fill in the required data in the page. 5. The user clicks on the 'create' button. 6. The user is sent back to the MoviePage. 7. The new movie is saved to the database and can be seen in the MoviePage.	Movie gets created and is shown in the list	Movie appears after creation	PASS	Yara
DetailAnsichtEinesMovieAnsehen_01	MoviePage (SortButton, Multiple MovieCard)	P1	A user should be able to open the details of a movie	Login or registration capability, Movie access capability	1. User must login or register themselves. 2. User navigates to the MoviePage. 3. User clicks on a Movie. 4. The Movie details are shown.	The Movie detail page is displayed after clicking	The info of the Movie does appear after user clicks on it	PASS	Max

Movie-Details-Checker

MovieBearbeiten_01	MovieDetailPage (6_Texts, DeleteButton, EditButton)	P3	A user should be able to edit the details of a movie	Login or registration capability, Movie access capability	1. User must login or register themselves. 2. User navigates to the MoviePage. 3. User clicks on a Movie. 4. User clicks on the edit button. 5. User makes changes to the TextInputs 6. User saves the changes 7. The changes are saved to the database and the user returns to the MoviePage	The changes are saved and the old data overwritten	The user edits the movie data and changes are saved	PASS	Max
MovieLöschen_01	MovieDetailPage (6_Texts, DeleteButton, EditButton)	P3	A user should be able to delete a movie	Login or registration capability, Movie access capability	1. User must login or register themselves. 2. User navigates to the MoviePage. 3. User clicks on a Movie. 4. User clicks on the delete button 5. The movie is deleted from the database and the user returns to the MoviePage	Movie gets deleted from the database and isn't visible in MoviePage	User able to delete data after clicking button	PASS	Kristijan
AktuellerUserSehen_01	UserProfilePage (LogoutButton, 4_TextFields)	P2	A user should be able to view the profile of the current logged in user	Login or registration capability	1. User must login or register themselves. 2. User navigates to the UserProfile. 3. The users data is loaded in and shown.	User is able to view their data in UserProfile	The data of the logged in user appears in UserProfile	PASS	Yara