

.NET Framework and C# - Project work - Group 1

Table of Contents

1. [About The Project](#)
 - [Contributors](#)
 - [Built With](#)
2. [Setup Process](#)
3. [Description](#)
 - [Mockup](#)
 - [ER Model](#)
 - [Dump File](#)
 - [Postman Collection](#)

About The Project

This repository contains the group project for the course **.Net Framework and C#** at the university of applied sciences in Darmstadt, Germany.

The application consists of a backend [WebAPI](#), a frontend [WPF](#) client and a [postgresql](#) database for data storage.

Contributors

Student members of project group:

- [Toubeyas \(Abel, Tobias\)](#)
- [Iron-Mike-Tyson \(de Riese-Meyer, Kevin\)](#)
- [T-hai \(Quang Thai Vu\)](#)

Lecturer:

- [alaluuk \(Alaluukas, Pekka\)](#)

Built With

This chapter contains a list of all technologies that are being used in this project.

Frameworks

- [.NET 6.0](#) for class library and frontend
- [ASP.NET Core 6.0](#) for backend
- [MSTest](#) for unit testing

Security

- Authorization of WebAPI with [JWT](#)

Services

- [Central postgresql server](#) from [h_da](#) as a database (german documentation)
- [The Movie Database API](#) is used to retrieve initial Media and Series objects and their Streaming Availability

Setup Process

(how to install the application, so that a new developer can start to work) These are the Steps you need to take to develop with this application:

1. Clone repository
2. Open project in Visual Studio 2022 and Build the solution or run "dotnet build solution.sln -c Release" in a shell
3. Start WebAPI backend by going into the StreamKing.Web folder and running "dotnet run -c Release --no-build" (--no-build is recommended so this process does not block any dll files from being used in other builds). You will be asked to login to the API (Username: streamking, Password: 7dyAvoCqTkmVqKNu5c9eX3wg) and then login to the database (Username: dotnet, Password:{ask us for the password privately, this is our live database, so we do not want to give out the password in the ReadMe.md}). The backend then is started on <https://localhost:9595> and hosts the swagger files at <https://localhost:9595/swagger/index.html>.
4. Start Frontend by executing the previously built exe File, which can be found at (StreamKing\bin\Release\net6.0-windows\streamking.exe).
5. Admin account credentials are Username: admin, Password: admin9876
6. Have fun!

Description

The StreamKing Application is used to get Streaming Availability Informations for many Movies & Series in the Regions USA, Germany and Finland. You are also able to save these to your own personal Watchlist which is linked to your Account and is then being saved in our central Database.

Mockup

Login View

Stream King

 USERNAME

 PASSWORD

LOGIN

[CREATE ACCOUNT](#)

Create Account View

Stream King

 FIRST NAME

 LAST NAME

 E-MAIL

 USERNAME

 PASSWORD

CREATE ACCOUNT

[Already have an Account? LOGIN](#)

Captcha View

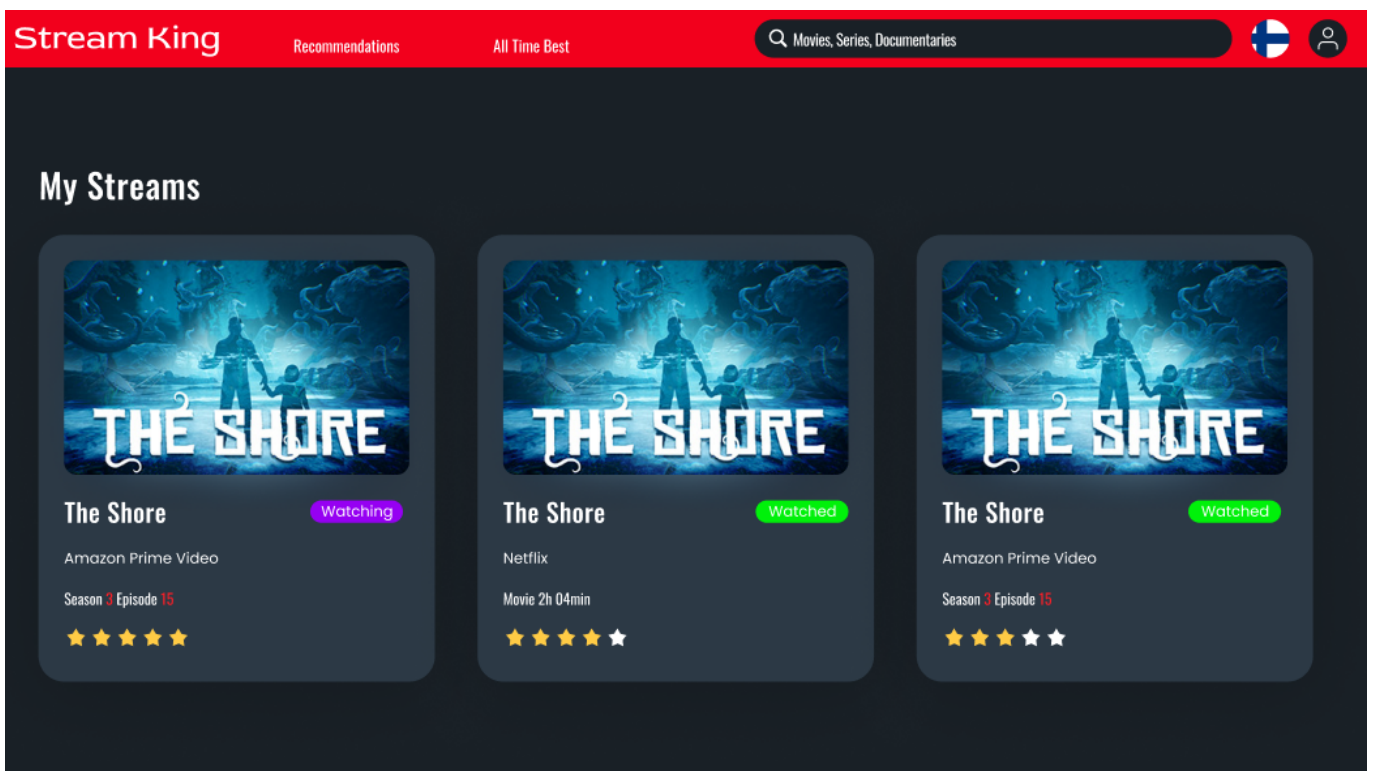
Stream King

CAPTCHA



Verify

Landing Page View




Search results View

Stream King

RecommendationsAll Time Best

The Shore

Results for The Shore




The Shore

Watching

Amazon Prime Video

Seasons 1 Episodes 121

★★★★★




The Shore

Watched

3+ Providers

Movie 2h 04min

★★★★★




The Shore

Unwatched

Amazon Prime Video

Season 3 Episode 15

★★★★★



Detail View Series



Drama Thriller Supernatural

Stranger Things

Seasons 7 Episodes 121

In 1980s Indiana, a group of young friends witness supernatural forces and secret government exploits. As they search for answers, the children unravel a series of extraordinary mysteries.

Community Rating Your Rating 

Season 1



1. Name of Episode goes here



1. Name of Episode goes here



1. Name of Episode goes here



1. Name of Episode goes here



1. Name of Episode goes here



1. Name of Episode goes here



1. Name of Episode goes here



Season 2



Season 2



Season 2



Season 2



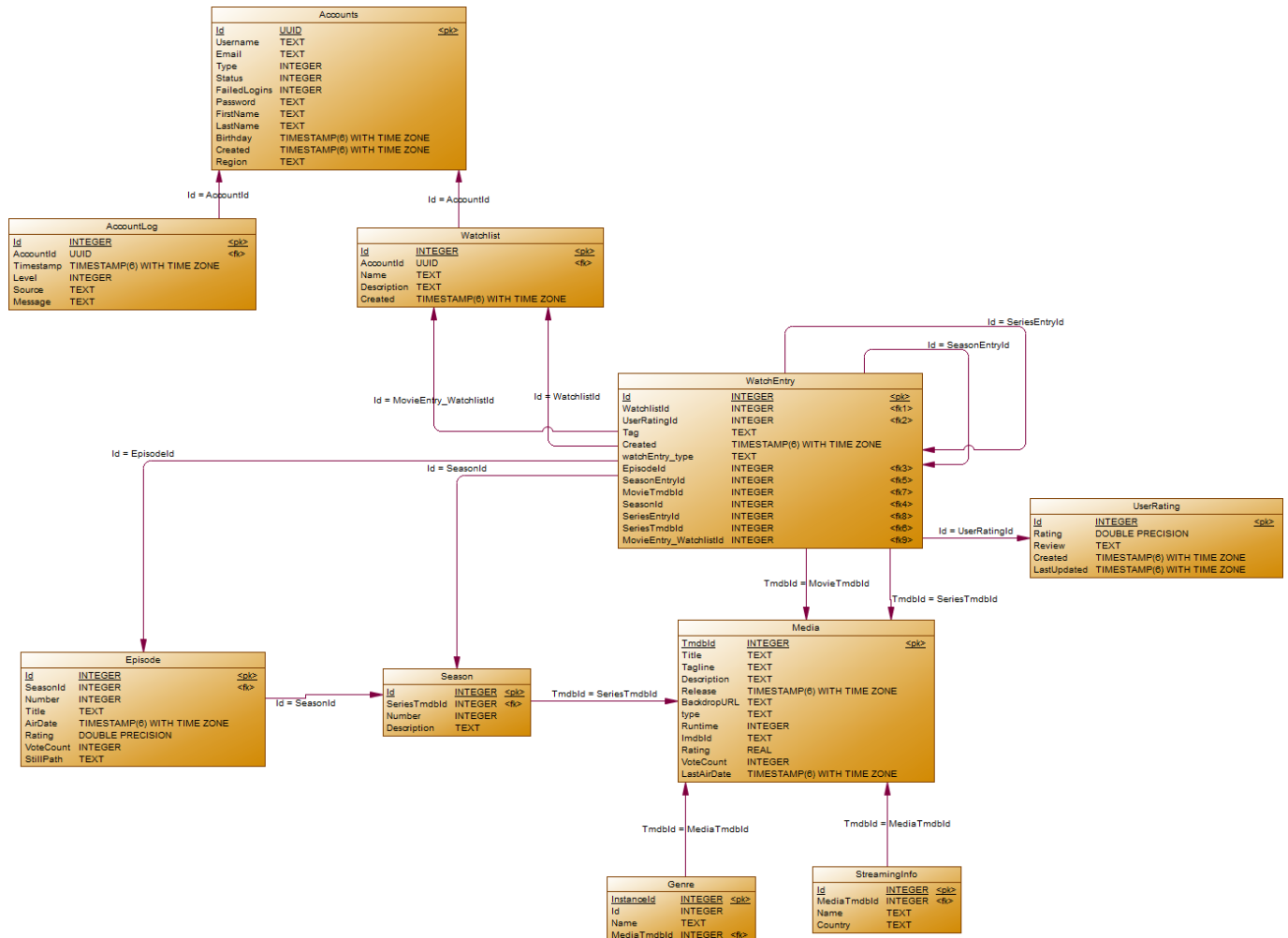
Season 2





ER Model

This is the ER-Model for the PSQL Database, which was created with the help of the SAP PowerDesigner. We didn't use the automatically created versions, e.g. from pgAdmin, because they weren't really readable and we had more freedom with this tool.



Dump File

[SQL Dump File](#) (From: 09.09.2022 00:10 AM) which was created with the pgAdmin PSQL Tool.

Postman Collection

[Postman Collection](#) (From: 09.09.2022 09:143 AM) which was created with Postman and used for API testing.