**User Documentation – FilmFókusz**

This user documentation provides detailed guidance on using the system and helps users make the most of its features. The purpose of the documentation is to simplify the user experience and ensure that all functionalities are accessible and easy to use.

**1. Introduction**

Welcome to the FilmFókusz application! This platform enables users to search for, browse, and access detailed information about their favorite movies and series. The system offers a simple, user-friendly interface that ensures quick access to the most popular movies and series.

**2. Key Features of the System**

The FilmFókusz application offers several useful features to help users quickly and easily find their desired movies and series:

**2.1 Searching for Movies and Series**

**Why is it important?**The search feature allows users to quickly find the movies or series they are looking for.

**How to use it:**

1. Begin typing the title or genre of the desired movie or series into the search bar at the top of the page.
2. As you type, the app will automatically start filtering available titles.
3. Results can be narrowed down quickly based on titles and genres.

**2.2 Browsing Movies and Series**

**Why is it important?**The system allows users to effortlessly browse different genres and titles.

**How to use it:**

1. Movie and series cards are displayed in the middle of the page. Each card includes the title, genre, and a thumbnail preview.
2. Clicking on any card reveals detailed information about the selected movie or series, such as its description, director, cast, and other useful details.

**2.3 Filtering by Genre**

**Why is it important?**Filtering helps users quickly find movies and series of interest without having to scroll through all available titles.

**How to use it:**

1. Users can filter content by genre using options located next to the search bar.
2. Select a genre to automatically refresh the results, displaying only titles within the selected category.

**3. User Interface (UI) and Navigation**

The FilmFókusz application features a simple and intuitive user interface to facilitate navigation and usability. Below is an overview of the main components and their functions:

**3.1 Header**

The header contains the application name and navigation menu:

* **Navigation Menu:** Links to various pages, such as movies, series, and additional features.
* **Search Bar:** Located on the left side of the header, it allows quick searches for movies and series.

**3.2 Navigation Bar**

The navigation bar, fixed at the top of the screen, remains accessible at all times. It includes:

* **Logo:** Clicking on the logo returns the user to the homepage.
* **Navigation Links:** Links to pages like "Movies," "Series," and "Account."
* **Search Functionality:** The search bar aids in locating desired content quickly.

**3.3 Movie and Series Cards**

The central area displays movie and series cards that include:

* **Thumbnail Preview:** The cover image of the movie or series.
* **Title:** The name of the movie or series.
* **Genre:** The genre of the movie or series.
* **Description:** Brief details about the movie or series.

**3.4 Footer**

The footer, located at the bottom of the page, provides information about copyright, privacy policies, and other essential details.

**4. Interactive Features**

**4.1 Search Functionality**

The search functionality is a key tool for quickly finding desired movies and series. Steps to use it:

* **Inputting a Search Query:** Type the desired movie or series into the search bar. The app filters data in real-time.
* **Filtering by Titles and Genres:** The system displays movies and series that match the entered criteria.
* **Displaying Search Results:** Results are presented as cards containing covers, titles, and genres.

**4.2 Clicking on Cards**

**Why is it important?**Clicking on cards provides users with more detailed information about the selected movie or series.

**How to use it:**

1. Click on the card of the desired movie or series.
2. The system displays detailed information, such as a description, cast, director, and more.

**5. Technology Stack (HTML, CSS, JavaScript, Git, Trello)**

This section introduces the technologies used in the project and their roles, including HTML, CSS, JavaScript, Git, and Trello, which were utilized for project tracking.

**5.1 HTML (HyperText Markup Language)**

**What is HTML?**HTML defines the basic structure of web pages. It is used to arrange content such as text, images, videos, tables, links, and other elements.

**Role in the project:**The front end of the FilmFókusz app is built using HTML to define the structure of the user interface, including elements like movie and series cards, search fields, and navigation menus.

**5.2 CSS (Cascading Style Sheets)**

**What is CSS?**CSS is a stylesheet language used to control the appearance of HTML elements, such as colors, layouts, fonts, animations, and other visual properties.

**Role in the project:**CSS ensures the visual design of the FilmFókusz app, including its layout, color scheme, and responsive design for various devices.

**5.3 JavaScript**

**What is JavaScript?**JavaScript is a programming language that enables dynamic functionality on websites.

**Role in the project:**The app's interactive features, such as search and filtering, were implemented using JavaScript.

**5.4 Git**

**What is Git?**Git is a version control system that tracks changes in source code and supports collaborative development.

**Role in the project:**Git was used to manage versions and facilitate teamwork.

**5.5 Trello**

**What is Trello?**Trello is a visual project management tool that helps organize and track tasks using boards and cards.

**Role in the project:**Trello was employed to organize tasks and track progress during the development of the FilmFókusz app.

**Developer Documentation – FilmFókusz**

**1. Introduction**

In today's world, the entertainment industry offers an overwhelming amount of content, which can be both a blessing and a challenge for viewers. The aim of this project is to create a website that provides personalized recommendations to help users decide what to watch. The platform seeks to meet the expectations of both technological innovation and user experience.

**1.1 Project Objectives and Problem Definition**

The FilmFókusz website is designed as a simple, user-friendly platform for recommending movies and series. Our goal is to assist users in discovering their new favorites through categories and a search function.

**Problem:**  
Due to information overload, users find it challenging to locate relevant content. Our aim is to simplify searches and increase viewer satisfaction.

**Functionality:**

* Dynamic movie and series recommendations displayed as cards.
* Search functionality for movies and series by title or genre.

**1.2 Core Features**

* **Movie and Series Display:** The system presents movies and series in two separate sections.
* **Search:** Allows users to filter by titles and genres.

**2. System Requirements**

**2.1 Hardware Requirements**

No special hardware is required to run the project, but the minimum recommended configuration includes:

* **Processor:** 1 GHz or better.
* **RAM:** At least 2 GB.
* **Storage:** 50 MB free space.

**2.2 Software Requirements**

The following software is required to run the project:

* **Operating System:** Windows 7/8/10, macOS, or Linux.
* **Browser:** Google Chrome, Mozilla Firefox, Safari, or Edge (modern versions).
* **Development Environment:** Visual Studio Code or any text editor.
* **Version Control:** Git is recommended for team collaboration.

**3. Folder and File Structure**

The simple structure ensures easy navigation:

* **index.html:** The main HTML file of the site.
* **css/style.css:** Defines the website's design.
* **js/script.js:** JavaScript logic for dynamic functionality.

**4. HTML Code Detailed Explanation**

**4.1 Structure**

The **film.html** consists of three main parts:

**Header**

* **Navigation Bar:**  
  Includes a logo, a search field, and a dropdown menu with icons.

**Main**

* **Movie Section:**  
  Dynamically generated cards displaying the title, genre, and image of movies.
* **Series Section:**  
  Similar to the movie section but displays series instead.

**Footer**

* Located at the bottom of the page, containing legal information and additional links.

**4.2 Core Features**

* **Search Field:** An input field and button enabling content filtering.
* **Cards:** Each movie/series is displayed as a card containing the title, genre, and an image.

**5. CSS Detailed Description**

**5.1 Base Colors and Design Principles**

The project’s visual harmony is achieved through the following colors:

* **Background Color:** #0C0F0A – A dark shade for a modern look while preserving contrast.
* **Text Color:** #E8E8E8 – A light shade for readability against the dark background.
* **Highlights:** #800020 – A burgundy color for emphasizing navigation elements, buttons, and interactive zones.

**Typography:**  
The fonts provide a lighthearted tone. The main font is *Comic Sans MS*, chosen for its clean yet decorative style. Text sizes establish hierarchy:

* **Headings:** Larger and bold.
* **Descriptions:** Smaller and simple.

**5.2 Responsive Solutions**

Responsive design is implemented using **CSS Grid** and **flexbox**, ensuring dynamic layouts on various screen sizes.

**Responsive Elements:**

* **Navigation Bar:**  
  Menu elements automatically adjust on smaller screens. The search field shrinks but retains functionality.
* **Card Layout:**  
  Movie and series cards adjust based on screen width:
  + **Large Screens:** 3-4 cards per row.
  + **Smaller Screens:** 1-2 cards per row.
* **Buttons and Fields:**  
  Sizes are reduced on mobile devices while maintaining visual balance.

**6. JavaScript Functionality**

**6.1 Data Management**

The project uses two main datasets stored as static JavaScript objects:

1. **movies:** A list of movies with title, genre, and image URLs.
2. **series:** Similar structure for series data.

**Example Data Structure:**

Dynamic elements are displayed on the page using the renderList function, which accepts a container ID and a dataset as parameters.

**6.2 Interactions and Event Handling**

* **Search Functionality:**  
  Users can enter the name or genre of the desired movie/series into the search bar. The filtering logic works as follows:
  1. The user clicks the "Search" button.
  2. JavaScript collects the content of the search field.
  3. The filter method filters movies and series by the title and genre fields.
  4. The results are displayed on the page.

**Search Code Example:**

* **Dynamic List Generation:**  
  The renderList function creates and displays a dynamic card list on the page.

**Code Example:**

**7. Development Guidelines**

**7.1 Coding Standards**

**Readability:**  
Identifiers, variables, and functions are named meaningfully.

**Comments:**  
All JavaScript functions include brief explanations to assist future developers.

**Modularity:**  
Each feature is organized into separate functions, ensuring the code is more readable and maintainable.

**7.2 Version Control Principles**

**Git:**  
Code is tracked using Git version control. The following branch structure is used:

* **main:** Stable, published versions.
* **dev:** Features under active development.
* **feature/[feature-name]:** Development of specific features.

**7.3 Project Management and Task Tracking with Trello**

Trello, a visual project management tool, was used to manage tasks and coordinate teamwork during development. It helped ensure that every team member was aware of their responsibilities and deadlines.

**8. Future Development Opportunities**

* **User Account Management:**  
  Allow users to register and receive personalized recommendations.
* **Interactive Content:**  
  Features for rating and commenting.
* **Backend Integration:**  
  Adding a database to dynamically manage movie and series data.