

Projektaufgabe: SteaM-System

Zweiter Teil: Umsetzung des Websystems mit JSF, JavaScript und Angular

1. Rahmenbedingungen / Organisatorisches

- Die Pflichtaufgaben sind – neben der Klausur – Teil der Prüfungsleistung und müssen somit erfolgreich absolviert werden, um erfolgreich die Lehrveranstaltung Webtechnologien zu bestehen.
- Für die Pflichtaufgabe sollen die Gruppen aus dem ersten Aufgabenteil bestehen bleiben.
- Abgabemodus: Laden Sie Ihre Lösung im Rahmen der WT-Übungen **bis spätestens** einschließlich **13. Juni 2022** in Moodle hoch (gruppenweise). Jede **Gruppe** muss die **Ergebnisse** live **vorstellen**. Bei der Vorstellung müssen alle Gruppenmitglieder anwesend sein. Entsprechende Slots werden ich noch einrichten und mich melden, sobald Sie sich registrieren können.

2. Ziel: Entwicklung des Steam-Websystems

Mit verschiedenen Web-Technologien (JSF, JavaScript) soll das bereits in Pflichtaufgabe 1 verwendete SteaM©-System (Streaming Series Memory – SteaM©) komplett umgesetzt werden. Wie schon dort beschrieben bietet SteaM© die folgenden Möglichkeiten:

- Eine Benutzerin kann sich am System mit ihrem Namen anmelden (mit Passwortüberprüfung). Nach erfolgter Anmeldung wird ihre persönliche Seriensammlung angezeigt.
- Eine Benutzerin kann neue Serien anlegen. Dabei werden Titel, Anzahl der Folgen (seasons), Genre, Streaming-Plattform einer Serie erfasst.
- Darüber hinaus kann eine Benutzerin für jede Serie ein Rating (good, bad,..) und eine Bemerkung vergeben.
- Serien können nach User, Genre, Plattform sowie Rating (des eingeloggten Users) gesucht werden.

Sie sollen dabei auf die SteaM©-Fassade, d.h. die Klasse SteamServcies aufsetzen. Ein UML-Diagramm von SteaM© finden Sie am Ende dieses Dokuments.

3. Aufgaben:

Aufgabe 1: (JSF) Setzen Sie Ihren Wireframe-Protoyp von SteaM© **mithilfe von JSF** um! Dabei sollen folgende Technologien verwendet werden:

- Session-Tracking
- Konvertierung und Validierung

Software-Design (wichtig): Erstellen Sie eigene Pakete bspw. web.data und web.control, für Ihre ManagedBeans. Insbesondere dürfen die SteaM-Backend-Klassen nicht in ManagedBeans umwandelt werden!)

Aufgabe 2: (JavaScript) Setzen Sie Ihren Wireframe-Protoyp mithilfe von **JavaScript** um!

Software-Design: Nutzen Sie Ihre im ersten Teil der Pflichtaufgabe implementierte REST-Schnittstelle!

Aufgabe 3: (Angular) Setzen Sie mindestens einen Dialog **mit Angular** um! (Die Minimalvariante ist hier die Login-Seite, die per REST die Gültigkeit des Passworts abfragt und die Antwort (Login richtig bzw. falsch) auf der selben Seite ausgibt.

Abzugebende Ergebnisse:

A. SOURCECODE:

1. **JSF:** Ein lauffähiges NetBeans-Projekt mit einer **JSF-Umsetzung** des Steam-Projekts.
2. **JavaScript:** Eine **JavaScript-Umsetzung** von Steam (=gezippter Ordner) und die zugehörigen REST-Variante von Steam (=normalerweise das NetBean-Projekt aus Ihrer ersten Pflichtaufgabe).
3. **Angular:** Fall Ihr Angular-Projekte zu groß ist, zippen Sie den src-Ordner.

B. DOKUMENTATION:

1. **Architekturbilder Ihrer drei Systeme (JSF, JavaScript, Angular – als pdf)**

Für die einzelnen Technologien müssen auf den Architekturbildern folgende Bausteine gezeigt werden:

- JSF:
 - alle xhtml-Seiten (mit sprechenden Namen)
 - alle ManagedBeans (sowie ggf. sonstige Java-Klassen)
 - für jede ManagedBean sollen die Attribute und die Event-Handler, sowie der Scope gezeigt werden.
- JavaScript:
 - alle html-Seiten
 - all verwendeten JavaScript-Dateien mit den enthaltenen JavaScript-Methoden (Event-Handler).
 - der JavaScript-Code sollte in mehrere Dateien mit passenden Namen ausgelagert sein. Jede Datei sollte einen Themenkomplex mit passenden JavaScript-Funktionen behandeln.
- Angular:
 - UML-Bild mit den verwendeten Typescript-Klassen, das die relevanten Attribute und Methoden zeigt.

2. **README-Datei.**

Darin erläutern Sie bitte, wer an der Gruppe beteiligt ist (Name, Vorname, Matrikelnummer) und wie Ihre Programme zum Ablauf gebracht werden können. Beschreiben Sie kurz den Beitrag jedes einzelnen Mitglieds zur Lösung der Aufgabe.

Bitte melden Sie sich, falls es noch Fragen oder Probleme gibt.

UML-Diagramm von SteaM©

