|  |  |
| --- | --- |
| Assignment 1 | Project Summary |
| Course | Multimedia Technologies with Angular and Typescript |

|  |  |  |
| --- | --- | --- |
| Project author | |  |
| № | Pseudonym | Face-to-face/ online |
| 1 | ddd | face-to-face |

|  |  |
| --- | --- |
| Project name | WorkoutCreator |

|  |
| --- |
| 1. Short project description (Business needs and system features) |
| There are numerous resources on the Internet about leading a healthy lifestyle and working out, but every fitness newbie gets overwhelmed by the massive amount of information, contradicting advice and the countless ready-made exercise routines. WorkoutCreator’s goal is to provide an easy to use tool for creating a custom made workout plan, based on the user’s level, goals, available equipment and time. The user would be able to save the created workout plan, make changes to it if needed and delete it.  The system will be developed as a *Single Page Application (SPA)* using ***Angular*** as front-end, and ***Node.js + express*** as backend technologies. Each view will have a distinct URL. The backend will be implemented as a ***REST/JSON API*** using JSON data serialization. The main user roles (actors in UML) are:   * Anonymous User – can view exercises and signup * Registered User – can view and filter exercises, add new exercises, save favorite exercises, create new workouts, save workouts, edit their own saved workouts, delete their own workouts, keep a diary log of workouts and measurements |

|  |  |  |
| --- | --- | --- |
| 2. Main Use Cases / Scenarios | |  |
| **Use case name** | **Brief Descriptions** | **Actors Involved** |
| **2.1. Create a workout plan** | The user is able to create a workout plan. | *Registered User* |
| **2.2. Save a created workout plan** | After creating a workout plan, the user has the option to save the workout to ‘My Workouts’. | *Registered User* |
| **2.3. View My Workouts** | The user can see a list of all of his saved workouts on the page ‘My Workouts’. | *Registered User* |
| **2.4. Edit a workout** | The user can edit (add or delete an exercise) his saved workouts. | *Registered User* |
| **2.5. Delete a workout** | The user can delete his saved workouts. | *Registered User* |
| **2.6. View exercises** | All users can view and filter the whole list of exercises. | *Anonymous* |
| **2.7. View exercise details** | The users can view details about each exercise. | *Anonymous* |
| **2.8. Add an exercise** | The user can add a custom exercise. | *Registered User* |
| **2.9. Delete an exercise** | The user can delete an exercise created by them. | *Registered User* |
| **2.10. Log workout / measurements** | The user can keep a diary log of done workouts and measurements by date. | *Registered User* |
| **2.11. Edit a log entry** | The user can edit their own log entries. | *Registered User* |
| **2.12. Delete a log entry** | The user can delete their own log entries. | *Registered User* |
| **2.13. View progress** | The user can view their own progress based on set goal and logged measurements through time. | *Registered User* |
| **2.14. Set a goal** | The user can set a goal – Lose Weight, Gain Weight, Maintain | *Registered User* |
| **2.15. Sign Up** | The users can create a new account in the system, if they haven’t already. | *Anonymous* |
| **2.16. Log in** | The users can log in by providing correct credentials. | *Anonymous* |

|  |  |  |
| --- | --- | --- |
| 3. Main Views (SPA Frontend) | |  |
| **View name** | **Brief Descriptions** | **URI** |
| **3.1. Home** | Introductory page | */* |
| **3.2. Log In** | View that allows users to enter their credentials to log in. | */login* |
| **3.3. Sign Up** | Presents a view that allows users to register | */signup* |
| **3.4. Exercises** | Presents a view that lists all exercises in the database. | */exercises/* |
| **3.5. Create Workout** | Presents a view that allows the user to create their own workout plan by choosing from existing exercises. | */create* |
| **3.6. My Workouts** | Presents a list of all workouts of the logged in user. | */workouts* |
| **3.7. Workout** | Allows the user to view and edit a particular workout of his saved workouts. | */workouts/{{wor koutId}}* |
| **3.7. Fitness Log** | Presents a view that shows the user’s log entries and progress. | */log* |
| **3.8. Account** | Allows the user to change his password. | */account* |

|  |  |  |
| --- | --- | --- |
| 4. API Resources (Node.js Backend) | |  |
| **View name** | **Brief Descriptions** | **URI** |
| **4.1. Users** | GET all users, POST new user data (register) | */api/users* |
| **4.2. User** | GET, PUT, POST user data | */api/users/{{userId}}* |
| **4.3. Login** | POST user credentials | */api/auth* |
| **4.4. Logs** | GET, POST, PUT, DELETE log data based on the provided logId | */api/logs/{{logId}}* |
| **4.5. Exercises** | GET, POST, PUT, DELETE data about a particular exercise, based on the provided exerciseId | */api/exercises/{{exerciseId}}* |
| **4.6. Workouts** | GET, POST, PUT, DELETE workout data based on the provided workoutId | */api/workouts/{{workoutId}}* |