|  |  |  |  |
| --- | --- | --- | --- |
| Assignment 1 | | Project Summary | |
| Course | | Fullstack Application Development with Node.js + Express.js + React.js - 2019 | |
|  | | | |
| Project author | | | |
| № | Pseudonym | | Face-to-face/ online |
| 1 | Antoni Dobrenov | | face-to-face |

|  |  |
| --- | --- |
| Project name | VolleyStats |

|  |
| --- |
| 1. Short project description (Business needs and system features) |
| As technologies grow, sports are getting coupled with them. Every professional sport team has its own statistics and becomes better when know what is wrong and what must be better for the next match. The project is for volleyball statistics, users can be registered (they may take the role of a team statist), teams and players, will be implemented using ***React.js*** as front-end, and ***Node.js + express*** as backend technologies. Тhe routing between pages will be done client side using ***React Router***. The backend will be implemented as a ***REST/JSON API*** using JSON data serialization. For database will be used Firebase or MongoDB.  The main user roles (actors in UML) are:  • *Anonymous User* – can only view results of public matches, but not their statistics.  • *Statist* (extends *Registered User*) – can manage (create, edit user data and delete) teams, players, matches, statistics.  • *Player* (extends *Registered User*) – can view own statistics and team statistics.  • *Fan* (extends *Registered User*) – can subscribe to teams and view their statistics. |

|  |  |  |
| --- | --- | --- |
| 1. Main Use Cases / Scenarios | | |
| **Use case name** | **Brief Descriptions** | **Actors Involved** |
| * 1. **Browse match results** | *The User can browse the results for public matches.* | All users |
| * 1. **Register** | *Anonymous User can register in the system by providing a valid e-mail address, first and last name, and choosing password. By default, all new registered users have Statist role.*  *Statists can register new Players by entering Player Data.* | *Anonymous User* |
| * 1. **Change Player Data** | *Registered User can view and edit her personal User Data.*  *Statists can view and edit Player Data of all Users.* | *Statist* |
| * 1. **Manage Players** | *Statists* can browse and filter players based on different criteria: first and last name, team, position, etc.  *Statist can choose a Player to manage, and can manage the chosen User - edit (using Change User Data UC) or delete.*  *Statists can create a new player using Register UC.* | *Statist* |
| * 1. **Manage Teams** | *Statist can browse and filter Teams based on different criteria: name of Team, player, etc.*  *Statist can create new Team.*  *Statist can choose a Team to manage, and can add or remove Players to the Team, or delete it if necessary.* | *Statist* |
| * 1. **Manage Matches** | *Statist* can browse her/his *Matches*, start new Match.  *Statist can enter statistics for started match.* | *Statist* |
| **2.7. Browse Statistics** | *Player can browse statistics of his team matches and each player personal statistics.*  *Statist can browse statistics of his team matches and each player personal statistics.*  *Fan can browse statistics of his team matches and each player personal statistics.* | *Player, Statist, Fan* |
| **2.8. Delete Statistics** | *Statist can delete statistics if necessary.* | *Statist* |
| **2.9. Delete Player** | *Statist can remove a Player from team if necessary.* | *Statist* |
| **2.10. Delete team** | *Statist can remove Team if necessary.* | *Statist* |
| **2.11. Browse Teams** | *Fan can browse teams.* | *Fan* |
| **2.12. Subscribe to Team/s** | *Fan can subscribe to team/s and to see matches and statistics of that team/s.* | *Fan* |

|  |  |  |
| --- | --- | --- |
| 1. Main Views (SPA Frontend) | | |
| **View name** | **Brief Descriptions** | **URI** |
| * 1. **Home** | Home page where user can choose either to login or register. | / |
| * 1. **Teams** | Teams page where user can browse teams. | */teams* |
| * 1. **Players** | Players page where Statist can browse players. | */players* |
| * 1. **Statistics** | Statistics page where user can browse his matches statistics. | */stats* |
| * 1. **Player Registration** | Presents a form for player registration. | */registerPlayer* |
| * 1. **Login** | Presents a view allowing the users to login. | */login* |
| * 1. **User Registration** | Presents a form for user registration. | */register* |
| * 1. **Edit Player** | Presents ability to edit Player data. | */editTeam* |
| * 1. **Edit Team** | Presents ability to edit Team data. | */editTeam* |
| **3.10. Start Match** | Presents ability to Start match with a team. | */startMatch* |
| **3.11. Match** | Presents ability to enter statistics during a match. | */match* |

|  |  |  |
| --- | --- | --- |
| 1. API Resources (Node.js Backend) | | |
| **View name** | **Brief Descriptions** | **URI** |
| * 1. **Players** | GET *User Data* for all users, and POST new *User Data* (Id is auto-filled modified entity is returned as result from POST request). Available only for *Statists*. | */api/players* |
| * 1. **Player** | GET, PUT, DELETE *Player Data* for *Player* with specified *userId*. | */api/players/{playerId}* |
| * 1. **Login** | POST *User Credentials* (e-mail address and password) and receive a valid *Security Token* to use in subsequent API requests. | */api/login* |
| * 1. **Logout** | POST a logout request for ending the active session with *OKTS,* and invalidating the issued *Security Token*. | */api/logout* |
| * 1. **Teams** | GET *Teams*, and POST new *Teams* (Id is auto-filled modified entity is returned as result from POST request), according to *User's Role*. | */api/teams* |
| * 1. **Teams modification** | GET, PUT, DELETE *Team* (including players in team) for *Teams* with specified *teamId*. | */api/team/{teamId}* |
| * 1. **Start Match** | GET team (chosen team for match start) and POST new *Match* (Id is auto-filled modified entity is returned as result from POST request). | */api/match* |
| * 1. **Match** | GET, PUT *Match Data* with specified matchId | */api/match/{matchId}* |
| * 1. **Statistics** | GET *Statistics* (according to *User's Role*) for *Statistics* with specified *matchId*. | */api/stats/{matchId}* |
| **4.10. Statistics** | GET *Statistics* (according to *User's Role*). | */api/stats* |