# Social Network – AngularJS Practical Project

You are assigned to design and implement a Social Network as **single page application (SPA)** using **HTML5** and **AngularJS**. The app has **users**. Each user has a **wall** and **friends**. **Friends** can see each other's **walls** and add post on them. Posts can be **commented**. Users can send **friend requests** to other users, **approve** or **reject** requests. Anonymous site visitors can only **login** and **register**. Logged in users can **logout**. You are given the server-side REST services to be called by your app with AJAX requests so you do not need to develop back-end.

## Important: Commit Every Day in GitHub

Please **use GitHub** for your project development!

* Commit several times a day.
  + You need to show many small commits that indicate your constant work on the project.
  + Avoid committing large blocks of code at once.
* You should prove that you have worked at least 3-4 days over your project!

Note that Git does not require Internet connection in order to commit changes. You can commit locally and push your changes to GitHub at once. Please commit many times to show your work progress step by step.

Up to 100 score

## Social Network REST Services

You are given the following REST services for your Social Network SPA application.

* Services base URL (in the Windows Azure cloud): <http://softuni-social-network.azurewebsites.net/api>
* Services help page (online documentation): <http://softuni-social-network.azurewebsites.net/Help>

The documentation below provide additional details for the Social Network REST services.

## Social Network Web Design

You аre given the **Web design** and **UI prototype** of the Social Network SPA application as PDF document.

You need to convert the design to HTML + CSS. Pixel-perfect layout is not required. You do not need to match exactly the sizes, fonts and colors of the elements. A responsive design is highly desirable. You can use responsive CSS frameworks like **Bootstrap**.

10 score

## Social Network SPA Application

Design and implement a client-side SPA application based on **AngularJS**.

### AngularJS Project Structure

* Prepare an AngularJS project structure following the industry best practices.
* You should have separate folders for controllers, directives, filters, services, views, etc.

30 score

# Project Requirements

### Public Screens

Public screens are accessible for site visitors without login.

* **Login Screen**
  + Route: **#/**
  + **Logins an existing user**. Shows notification for success or error message.
  + After login, the user is automatically redirected to the user home screen.

5 score

* **Register User Screen**
  + Route: **#/**
  + **Registers a new user**. Shows notification for success or error message.
  + After registration, the user is automatically logged in and is redirected to the user home screen.

10 score

### User Screens

User screens are accessible for authorized users only (after login).

* **User Home Screen (news feed page)**
  + Route: #/
  + Includes **user header** (**name**, **buttons**, **search bar**)

5 score

* + **Search** users by name. Display matched users **profile image** and **name**.

10 score

* + Display **pending** friendship **requests**.

3 score

* + Display requests' details when clicked on the pending requests button.

7 score

* + Display own friends count, as well as **image** and **name** about 6 of your friends (retrieved from "**api/me/friends/preview**").

10 score

* + Display **posts made by friends** (retrieved from "**api/me/feed"**).

5 score

* + Display posts data (**author profile image**, **name**, post **date**, **content**, **likes count**, total **comments count**, **comments** (by default only 3, if clicked on "More…" button, display all comments)).

10 score

* + Dropdown **text area** for commenting a post on clicked [Comment] button and submitting comment to post.

10 score

* + Implement liking and unliking **posts**.

5 score

* + Popup when hovering over other users - displays user image, name and one of the three statuses - **friend**, **pending** (meaning there is currently a request) or **invite** button (for sending a friend request).

5 score

* **User wall**
  + Route: #/users/:username
  + **Display** the user's wall

10 score

* + Post box with **[Submit]** button for **adding new post** on wall.

10 score

* **Friends**
  + Route: #/users/:username/friends
  + Display all **user friends** in a table (showing **name** and **profile image** for each user).

10 score

* **Edit Post**
  + Users should be able to **edit only their own posts**. Show notification for success or error message.

5 score

* **Delete Post**
  + Users should be able to **delete their own posts**,as well as **any posts on their own wall** (regardless of who the author is). Show notification for success or error message.

5 score

* **Edit User Profile**
  + Route: #/profile
  + Users should be able to **edit their profile data** (**name**, **email**, **gender**,upload **profile picture**,upload **cover picture**). Show notification for success or error message.

10 score

* **Change User Password**
  + Route: #/profile/password
  + Users should be able to **change their password** from form (contains **old password**, **new password** and **confirm new password**). Show notification for success or error message.

5 score

* **Logout**
  + Route: #/logout
  + Successfully logged in users should be able to **logout** from the app.
  + Logout shows a notification message and redirects to the Home screen.

5 score

* **Guest Authorization Checks**
  + Anonymous site visitors (without login) should be able to access only Login and Register screens.
  + An attempt to access anonymously these screens should redirect the user to the Home screen.

5 score

* **User Authorization Checks**
  + Users should be able to access **everybody's wall**, but only their **own** **news feed**.
  + Users should **NOT** be able to **post** on **non-friend walls**.
  + Users should **NOT** be able to **like/unlike** **posts** when **neither the author, nor wall owner is a friend**.
  + Users should **NOT** be able to **see non-friends' friends preview** or **full friends list**.

10 score

Total 300 score

# Project Bonuses

### User Screens

* **User Home Screen (news feed page)**
  + Implement liking and unliking **comments**.

5 score

* + **Dynamic scrolling** with **pagination.**

10 score

* **User wall**
  + **Dynamic scrolling** with **pagination.**

10 score

* **Friends**
  + Dynamic **client-side name filter.**

5 score

* **Edit Comment**
  + Users should be able to **edit only their own comments**. Show notification for success or error message.

5 score

* **Delete Comment**
  + Users should be able to **delete their own comments**, as well as **any comments of their own posts**. Show notification for success or error message.

5 score

* **User Authorization Checks**
  + Users should **NOT** be able to **comment** on posts when **neither the author, nor wall owner is a friend**.
  + Users should **NOT** be able to **like/unlike** **comments** when **neither the post author, nor wall owner is a friend**.

20 score

Total 60 score