# Teamwork Project Assignment for the [JavaScript Applications Course @ SoftUni](https://softuni.bg/courses/javascript-applications/)

Design and implement a **Blog** / **Forum** / **Photo Album** / **Audio Album using JavaScript, jQuery, AJAX, REST**. Your project must meet all the requirements listed below.

## Requirements

* **Use JavaScript, jQuery and AJAX** – the major part of your work should be JavaScript written by your team
  + **Front-end**: use **HTML5, CSS3** to create the content and to style your Web application
  + **Back-end**: use REST API from **kinvey.com** / other **BaaS provider** / your **own back-end REST service**
  + Focus on the front-end, which should be written in JavaScript
* **Work in team** – all team members should contribute
  + **Use GitHub** or other source control systemas project collaboration platform
  + **Each team member should have commits in 5 different days**
* Publish your project **live in Internet** – your project should be public in Internet
  + You may **share your project** to get external feedback
  + Most shared and commented projects will get additional **bonus score**
* **Valid and high-quality JavaScript, jQuery, AJAX, HTML and CSS**
  + Follow the best practices for JavaScript Application development. Structure your code well
  + Validate (when possible) your HTML (<http://validator.w3.org>) and CSS code (<http://css-validator.org>)
  + Follow the best practices for **high-quality JavaScript, jQuery, HTML and CSS**
* **Usability, UX and browser support**
  + Your web application should be easy-to-use, with intuitive UI, with good usability
  + Ensure your Web application works correctly in the latest HTML5-compatible browsers: Chrome, Firefox, IE, Opera, Safari (latest versions, desktop and mobile versions)
  + You do not need to support old browsers like IE9

## Forbidden Techniques and Tools

You should **write your application from scratch**. You are **not allowed** to use:

* Existing projects and frameworks like CMS systems, forum systems, photo album systems, etc.
* Single Page Apps frameworks like AngularJS, Backbone, Ember, etc.
* JavaScript APIs for accessing the server-side data. **Use REST, JSON and AJAX calls**.

## Projects

Please choose one of the projects below.

### Blog

**Required** functionalities:

* **View** all posts (optionally with paging).
* Adding **new posts** by the blog owner (after login or password protected). Each post must have **tags**.
* Adding **comments** for every post by visitors – each visitor must fill out his name, email (optionally) and comment text.
* Implement a sidebar holding a **list of posts** sorted by month / year / etc. and a list of the **most popular tags**.
* **Counter of visits** for each post.
* Functionality for **searching** by tags.

**Optional** functionalities:

* **User registration** and user profiles.
* **Admin panel**: add / edit / delete posts, comments, tags, etc.

### Forum

**Required** functionalities:

* **View** all questions / categories (optionally with paging).
* Implement a simple **registration** for forum users.
* Adding **new question** by the forum users. Each question must have **tags** and **category**.
* Implement **categories** for the forum questions.
* Adding **answers** to the questions by the forum visitors – each visitor must fill out his name, email (optionally) and comment text.
* **Counter** for visits for each question.

**Optional** functionalities:

* **Admin panel**: add /edit /delete forum posts, tags, answers, categories.
* Functionality for **searching** by question, answer and tags.
* Implement **ranking** according to user activity.

### Photo Album

**Required** functionalities:

* **View** all albums / categories (optionally with paging). **Browse** album photos.
* **Creating** new album in a category.
* **Uploading photos** (validating pictures size and type) / **downloading** photos.
* Adding **comments** to photos and albums.
* Implement album's **ranking system** (e.g. vote from 1 to 10 or like / dislike).
* Show the most **highly ranked** albums in a special section at the main page.

**Optional** functionalities:

* Implement **user registration**.
* Functionality for **searching** by album name / category.
* **Admin panel** (if registration is implemented): add / edit /delete albums, photos, comments.

### Audio Album

**Required** functionalities:

* **View** all playlists / genres / songs (optionally with paging).
* **Listening** to songs online. **Downloading** songs.
* Creating **new playlist**.
* **Uploading songs** (validating file size and type).
* Adding **comments** to songs and playlists.
* Implement playlists' and songs' **ranking system**. Show the most **highly ranked** playlists in a special section at the main page.

**Optional** functionalities:

* Implement **user registration**.
* Functionality for **searching / filtering** by playlist name / song name / genre.
* **Admin panel** (if registration is implemented): add / edit /delete songs, playlists, genres, comments.

## Deliverables

Put the following in a **ZIP archive** and submit it (each team member submits the same file):

* The complete **source code** of your project (JavaScript, jQuery, HTML, CSS, images, scripts and other files).
* A **presentation** of your project (e.g. PowerPoint slides) of your project. It should provide the following information:
  + Project name and purpose – what you have created?
  + Team name, list of team members.
  + Contributions of each team member.
  + Technical description.
* Any other information (optionally).

## Public Project Defense

Each team will have to deliver a **public defense** of its work in front of the other students, trainers and assistants. Teams will have **only 10 minutes** for the following:

* **Demonstrate** the web application (very shortly).
* Show the **source code** and explain how it works.
* Explain how each team member has **contributed**: display the commit logs in the Source Control System you are using.
* Optionally you might prepare a **presentation** (3-4 slides).

Please be **strict in timing**! On the 10th minute you **will be interrupted**! It is good idea to leave **the last 2 minutes for questions** from the other students, trainers and assistants.

Be **well prepared** for presenting maximum of your work for minimum time. Bring your own laptop. Test it preliminary with the multimedia projector. Open the project assets beforehand to save time.

## Assessment Criteria

* **Functionality** (all the required functionalities according to the type of project you choose) – **0…7**
* Correct use of **AJAX, REST and JSON** for accessing the server-side data – **0…3**
* **Code quality** (well-structured code, split into modules and files, good naming, formatting, etc.) – **0…5**
* **Security** (correct data validation, absence of cross-site scripting / unescaped data in the UI, etc.) – **0…2**
* **Teamwork\*** (GitHub used; each team member contributed in 5 different days; distribution of tasks) – **0…3**
* **Bonus** (bonus point are given for implementing optional functionalities according to the type of project you choose, using of patterns) – **0..5**

\* If not all team members have contributed to the project, this **does not affect** the Teamwork points.

## Give Feedback about Your Teammates

You will be invited to **provide feedback** about all your teammates, their attitude to this project, their technical skills, their team working skills, their contribution to the project, etc. The feedback is important part of the project evaluation so **take it seriously** and be honest.