# Teamwork Project Assignment for the [Web Services and Cloud Course @ SoftUni](https://softuni.bg/courses/web-services-and-cloud/)

This teamwork project assignment is designed to develop skills for creating web services (ASP.NET Web API) and to deploy web services in cloud like (Azure, AppHarbor).

## Project Description

Design and implement a web services. Deploy the ASP.NET Web API application in the cloud. Implement a JavaScript client application which consumes the written services from the cloud.

You could implement project like:

* **Web Chat application**
  + You have users, messages, notifications…
  + Users can send messages between each other
  + Users can send files
  + Users receive notification when another user send them a message
  + …
* **LinkedIn-like application**
  + You have users, groups, skills, endorsements…
  + Users can have connections
  + Users can have skills
  + Users can endorse their connections for skill
  + Users can create groups
  + Users can join groups
  + …
* **Facebook-like application**
  + You have users, posts, comments…
  + Users can have friends
  + Users have wall
  + Users can write a post on his or on his friend wall
  + Users can create groups
  + Users can join groups
  + Users can comment his friends posts
  + …
* **Twitter-like application**
  + You have users, tweets…
  + Users can follow users and can be followed by another users
  + Users can tweet
  + Users can reply to tweets
  + …
* **Image gallery application**
  + You have users, galleries, albums, comments, notifications…
  + Users can create gallery
  + The gallery can have many albums
  + Users can subscribe for gallery
  + Users can upload images in albums in their own galleries
  + Images have title
  + Users can leave a comment about an image
  + Users receive notifications when somebody comments an image of theirs
  + …
* **Tic-tac-toe game**
  + You have users, games…
  + Users can create game
  + Users can join a random game
  + Users can perform moves in a started game
  + Users receive notifications when a user in a game of theirs has made their move
  + …
* **Bulls and Cows game**
  + You have users, games…
  + Users can create game
  + Users can join a random game
  + Users can perform moves in a started game
  + Users receive notifications when a user in a game of theirs has made their move
  + …
* **Application by choice**
  + You can use given example application
  + You can create your own application
  + You can improve one of the given example applications

## General Requirements

### Server Application

* Your application must be implemented using **ASP.NET WebAPI**
* Create **RESTful web services**
* Host the application in **AppHarbor** or **Azure**
* Use a **file storage cloud API**
  + **Dropbox**, **Google Drive** or other
* Use a **cloud database**
  + **MS SQL**, **MySQL**, **MongoDB**, **Redis** or other
* You can use push notifications for the notifications (PubNub, SignalR…)

### Client Application (JavaScript)

* Implement a UI for your application (you can use **Bootstrap** or implement your own **UI logic**)
  + Don’t put too much time on a beautiful UI
  + Use UI libraries to save time
* The client application should communicate with the services application using HTTP requests
* The application must work in last versions of the browsers

## Additional Requirements

* Follow the best practices for OO design:
  + Use data encapsulation.
  + Use exception handling properly.
  + Use inheritance, abstraction and polymorphism properly.
  + Follow the principles of strong cohesion and loose coupling.
* Use a source control system by choice.
* Submit a link to your repository.

## 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** how the application works (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

* **Authentication – 0…10**
* **Functionality** – **0…25**
* **Code quality** (well-structured code, split into classes and files, good naming, formatting, etc.) – **0…5**
* **Teamwork\*** (GitHub used; each team member contributed in 5 different days; distribution of tasks) – **0…5**
* **Bonus** (bonus points are given for implementing optional functionalities / original approach) – **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.