**Team Clover Club**

Team Project – Video Manager

Project Description

Design and implement an object-oriented web SPA application by choice.

**Team Members:**

Андрей Божанков – Adrian.bozhankov

Ивайло Андонов – IvayloAndonov

Иван Василев – ivanvasilev\_90

Родотея Йорданова – Rodoteia\_Yordanova

**Project Explanation:**

Our project is a Video Manager that uses the YouTube Api, where every person can register and upload videos from YouTube in his account and group them in different categories. That way every person has a personal database with all his favorite videos, without the need to search them every time.

The web page starts with a greeting home page.

On the main page there are login and contacts fields.

When clicking on the Login button, the application shows a login form, where the user must enter his username and password if he is already registered. If the person doesn’t have an account, he must click on the Register button.

When clicking on the register button the client is redirected to a register form where he must enter his desired username and password(two times). If the username is not taken and the password is valid a message will pop-up, that the user has successfully registered.

Then the user can log in his account. When logging in, on the top of the screen there will appear two buttons – SHOW, that has a field with options of every category and ADD VIDEO.

When successfully logging in the selected category will always be ALL – to show every categories and videos in them, that the user has so far. When changing to a single category and clicking on the button SHOW, on the screen will appear only the selected category with all the videos in it.

When clicking on the ADD VIDEO button on the right will pop a form for adding videos. The form has two fields – in the first field the user must enter a YouTube url of the video he wants to upload, and in the second field he must enter the Category of the video. If the user enters only an url – the video will be uploaded to category Others.

Every video has a DELETE button, and if the user decides that don’t want to anymore some video, he must click on the button and the video will no longer show in his profile.

**Git repository:**

<https://github.com/Clover-Club/JS-Apps-Course-Project-Team-Clover-Club.git>

**Other Information about the project:**

# General Requirements:

Please define and implement the following assets in your project:

* DONE Use jQuery
* DONE Implement OOP design
  + Application logic using objects, modules and data hiding
    - Both Prototypal and classical inheritances are Ok
  + At least 3 modules
  + At least 7 types of objects
* DONE: Unit tests
  + Using Mocha, Chai and SinonJS
* DONE Implement a UI for your application
  + Use KendoUI, jQueryUI or implement your own UI logic
* DONE Use some kind of web data storage, one of the following is Ok
  + Your backend with Node.js, PHP, ASP.NET, Django, Ruby on Rails, Spring, etc...
  + Telerik Backend Services
  + Parse
  + Microsoft Mobile App Services
* DONE Use some kind of local storage, one of the following is Ok:
  + localStorage
  + sessionStorage
* DONE Use Twitter Bootstrap
  + Research and use Bootstrap for your application
  + Make the application responsive for different screens and resolutions
* DONE At least one third-party API to share something from your application
  + Samples: Share status to Facebook, Google+, Twitter, etc... Upload images to Facebook, Flickr, Instagram
* DONE The application must work in the latest versions of the browsers: Google Chrome, Mozilla Firefox, Internet Explorer 10/11, Opera and Apple Safari

# Additional Requirements:

Follow the best practices for producing high-quality code:

* DONE Correct naming
* DONE Data encapsulation
  + Use OOP and modules
* DONE Strong cohesion and loose coupling
* DONE Use GIT as a source control system
* DONE Host it on [http://github.com](http://github.com/)

# Optional Requirements:

If you have a chance, time and a suitable situation, you might add some of the following to your project:

* Backward compatibility (make the application usable on browsers like IE8, IE7 and IE6)
* Integration tests
* Usage of a structural JavaScript framework:
  + AngularJS, KendoUI, Knockout.js, Backbone.js, etc...