React Madness

We decided to split the project into three individual components, as follows:

* client side interface,
* server side, which will call the web service,
* web-service, to build a project from scratch and allow the server to download it;

The persistence Web Service based interface follows the structure described in Fig.1 in the first project related document. This dictates the required operations that the web service must expose for possible clients. The main operations are described down below:

* /login authToken; (POST)

Params: @username;

@password;

* /register authToken; (POST)

Params:@username;

@password;

* /projects?token=value a list of projects that the user is owning; (GET)
* /createProject projectId; (POST)

Params:@name;

@ownerToken;

* /viewProject?projectId=value a list of pages and all the controls that belong to the project; (GET)
* /modifyProjectName; (POST)

Params:@projectId;

@name;

* /deleteProject; (DELETE)

Params:@projectId;

The rest of the database entities will have a similar behavior to the one described above. The method used for each operations is specifies inside the brackets.