Ceren Gazioglu Majoor (2000679)
Anton Eliens
Intelligent Multimedia Technologies
19/01/2010

VUUGLE REPORT

VUUGLE was supposed to be a search engine, an annotation tool, and a language learning application; and it became one; however, with a huge future to-do list!

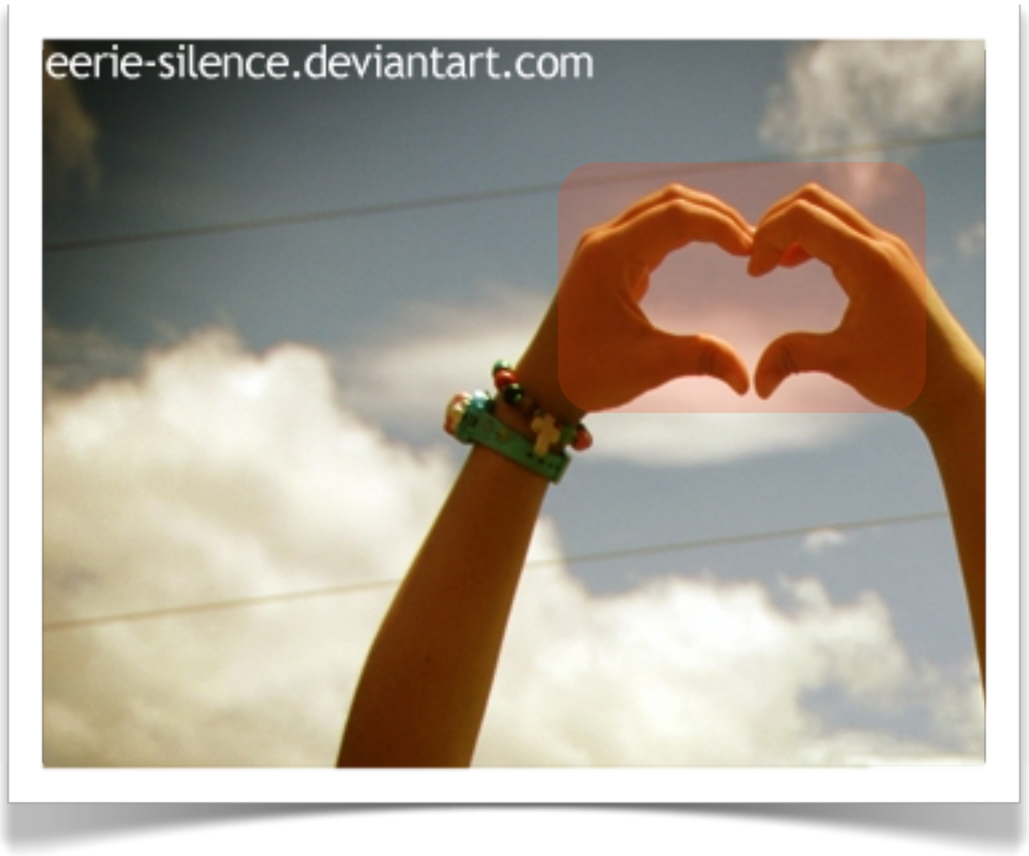
First of all, I realize that I have made my choices right, with choosing FLEX development environment and PHP. The commonness of these two technologies allowed me to deploy VUUGLE on my host and it became a real and live application, which already has almost 10 users now.

Coding with ActionScript 3, MXML was very easy and totally object oriented if you have already coded with Java, you will feel very comfortable and at home. The FLEX IDE which is also based on Eclipse provides intuitional functionalities like auto-completion of code and component explorer which gives you chance to use pre-made components like a TextArea or a Button. And there are a great bunch of libraries that come with flash and flex, which allows the developer to easily handle user inputs like mouse movements and events like clicking on a DataGrid item, etc.

Setting up AMFPHP and finding resources was hard (it is not being developed anymore), but not impossible. It comes with a services-config.xml file to be modified and once you create a service, you are good to go. To communicate with the back end side using AMFPHP allows the execution of queries to be faster than calling rest services and loading and reading JSON files for instance. Also the while using AMFPHP you don't deal with XML parsing but the result already comes as a ready object. While I appreciate the ease of use of AMFPHP, it lacks the multithreading of method calls. VUUGLE, being an online application, and a multi user application brings into play security issues. This means whenever I want to add or update a record in the database, things first have to be checked, if the data is already there, if everything is acting as it is supposed to be. And with AMFPHP you have to create 2 different remote objects it seems. Otherwise you don't know when one result event will return and the synchronization of back end logic cannot be made. So, I would use AMFPHP if I had basic data retrievals to make in projects where less security is needed.

Other than the programming languages review, I would like to comment on the architecture of this project. Since VUUGLE needs access to some images which users can annotate, one question was "where to find those images?". First thought was using Flickr, or Google. But there are a few drawbacks of using such an external system. One is the famouse "Security sandbox violation" where the page that contains the resources you will use need to explicitly state that you are allowed to use it using a crossdomain.xml file. The other is, what if the image is removed from that website. What will happen to the annotations that are already made on that image? So, I decided to let the users upload their own images if they are allowed to. This is possible by dividing users into categories like inactiveUser, approvedBasicUser, bannedUser, approvedVIPUser, admin etc. and giving them permissions later.

All in all, VUUGLE project has a concept which can be grown into something more secure, more stable and semantic. Some of the future development ideas are integration of semantic databases like DBPedia with the annotations and letting users match words in their mother language and second language which will let the users know that the two annotations made in two different languages have the same meaning. For now, let's say there is the following image with a red region over the hands. A user can annotate this region as “hands”, and another user can annotate this region as “heart”. So which one is the right one?



WORK CITED

Potter, Mike, http://blogs.adobe.com/mikepotter/2006/07/php_and_flex_js.html PHP and Flex - JSON, XML or AMFPHP? 12 July 2006

Obitko, Marek, http://www.obitko.com/tutorials/ontologies-semantic-web/, Ontologies and Semantic Web, 2007