

Report week 9

Project name: Artfolio.

Name and surname: Greta Aquilina.

Week: : 2/05 up to 9/05.

Functionalities implemented during the week

During the 9th week, an in-depth test of the entire website, and its functionalities, was performed to verify that every part of the website worked correctly. The change that has been done:

- On the creator's side, when visualizing the messages received by companies, it's now possible to click on the meeting link.

(Commit link:

https://github.com/AquilinaGreta/Creating_Internet_Application_Project/commit/583ddfdeed15dfd388663399f6b3a7ca45bb19e7)

Functionalities implemented overall

The functionalities, from the specification, that have been implemented overall:

- Creation of an account or sign-up.
- Login into the system.
- Post a creation, as a creator.
- Post a job advertisement, as a company, and visualize suggested creators who have a better affinity for that job. So, the possibility, as a company: to visualize the creator's profile; to create an appliance for that creator and automatically accept the proposal and send him a message with meeting or contact information.
- Search for users.
- Visualize user's profile (by clicking on the user's name in the result of the search or on its name in the post displayed in the homepage)
- Homepage with: the possibility to browse the creations and job advertisements (i.e there are two sections in the homepage: creations and job advertisement); the possibility to filter the creations by order (ascending, descending) or by tag; same for the job advertisements.
- Visualize a post, user can click on the name of the user, who created that post, to be redirected to his profile and visualize that post in its profile.
- Apply to a job advertisement
- Visualize applications to a job advertisement in company's profile
- Visualize list of applied job advertisement in content creator's profile
- Visualize the content creators' portfolios that are suggested after the creation of a job advertisement or that are displayed in the list associated with a job advertisement.
- Sending meeting and contact informations to a candidate after accepting his/her proposal
- Modify or delete a job advertisement
- Modify or delete a creation

- Notifications in case: the creator's proposal has been accepted and so he/she has received a message; the creator's proposal has been rejected; the creator's profile, shown in the proposals list of the company, has been visited.
- On the creator's side, a section where the creator can visualize all the messages that he/she received from companies.
- Log out

Functionalities that changed or were added with respect to the specification

The functionalities that changed with respect to the specification are the following:

- In the specification, the job advertisements could be filtered according to: latest and popularity. In the actual implementation, the job advertisement can be filtered by order: ascending, descending (default order); by tag (i.e by job figure). The reason behind this change is due to the fact that a creator, looking for a specific job advertisement, might be interested in finding the job advertisement that is related to his/her job figure rather than a popular, but not related, job advertisement. Regarding the filtering according to the latest, the 'latest' filter corresponds to the 'descending' ordering of the job advertisement.

The additional functionality with respect to the specification is the following:

- On the creator's side, a section, where the creator can visualize all the messages that he/she has received from companies. The aim of this section is to increase the readability of some messages that might appear too long or difficult to be read in the notification section.

Functionalities that were not implemented

The functionalities that were not implemented in the project are the following:

- Hosting online of the project. I couldn't find a free or an affordable platform for online hosting.
- Implementation of prevention of the sql injections. This is due to the fact that the initial sql queries were written with a wrong structure with respect to the one required to perform the prevention of sql injections. Changing the structure of the sql queries during the last week, might have led to incorrect behaviors in the system, some of them might have been difficult to detect or fix in a small range of time.

Net time for the implementation of the project

The total net time required to implement the project is, approximately, 142 hours that corresponds to the original estimated hours, approximately 140 hours.

Issues while developing the project

During the website development, since I had no experience with AJAX, implementing functionalities that required the use of AJAX (i.e, filtering operations, checking the log-in and sign-up fields, etc.) was very difficult.

After developing the website, some things that could have been implemented differently are:

- Sql queries. They should have been implemented according to the correct structure to prevent sql injections. In this way, the website could have been more secure.
- Framework. A more popular framework could have been used, like React.js.

Despite the issues and parts of the website that could be reworked, I'm overall satisfied with the final version of the website.

Number of lines of the project

Total number of lines of the project with the framework:

```
github.com/AlDanial/cloc v 1.92 T=7.71 s (40.2 files/s, 39870.4 lines/s)
```

Language	files	blank	comment	code
JavaScript	171	9321	8057	180776
CSS	51	8302	496	57019
SVG	4	4	16	19776
HTML	1	5	0	6130
PHP	38	1241	244	4394
JSON	40	0	0	3635
TypeScript	1	0	0	3036
Markdown	3	434	0	2314
LESS	1	4	10	2305
SUM:	310	19311	8823	279385

Total number of lines of the project without the framework:

```
github.com/AlDanial/cloc v 1.92 T=0.06 s (628.9 files/s, 96640.1 lines/s)
```

Language	files	blank	comment	code
PHP	37	1234	228	4376
Markdown	1	0	0	1
SUM:	38	1234	228	4377