

D8.2 Semester of Code guide for Organisations

Authors: RayCom Bv, Edwin Veenendaal, Raymond Elferink

Project Partners:

1. Universidad de Salamanca
2. OpenDirective Ltd.
3. Raycom B.V
4. University of Bolton
5. Università di Udine.
6. University of Cyprus
7. Mindshock S. L.
8. University of Oxford – OSS Watch

Contents

1. Introduction.....	3
2. Experiences from the organisations.....	4
3. Running the pilots.....	4
4. Process of Semester of Code.....	6
5. Roles in Semester of Code	7
6. The Implementation of Semester of Code.....	7
7. References.....	8

I. Introduction

The original purpose of this report as stated in the project proposal was the following: “...describing the experiences carried out, specially focused in how business and foundations can obtain benefits from the exchange of knowledge between them and with HE institutions through the virtual placements. It includes a description of the virtual placements carried out from the point of view of the business and foundation, taking into account the problems addressed, the innovation open source projects selected, how the virtual placements solve those problems, how to find students interested on this kind of problems, how to find future candidates to work in the companies, how make possible to evolve open source software projects, etc. In addition a guide to solve the most common issues is included in order it can be taken into account for future Semesters of Code.”.

But as extensively described in the document “Understanding the barriers to virtual student placements in the Semester of Code”, we found lots of difficulties on the road for achieving a convincing number of student placements during the various pilot runs we did. So instead, after describing the experiences from the company point of view, we will lay out the process in general and how to implement Semester of Code, building upon the recommendations done in the above mentioned document.

2. Experiences from the organisations

For organisations there was not so much new to the Semester of Code concept except that there is a third party involved (the university as a legal party to sign a contract with and the tutor)¹. As the name suggests, Semester of Code was derived from the idea of Google Summer of Code that is well known by organisations for years now.

Since the pilots ran for a longer time and were extended and/or used as input for the next pilot, projects were longer exposed to students than would be the case for Google Summer of Code. For that reason –we assume- some mentors lost interest or even changed jobs during the run of the various pilots and were sometimes difficult to reach.

3. Running the pilots

The first pilot started the 15th of september in 2014. To have the ability for universities to judge projects, possibly filter out spam projects and so on, we created a buffer period of 1 month between the start of the student exploration period and the final date to deliver projects by the organisations.

In the summer of 2014 we had the system more or less ready and organisations could register themselves in the system. They did that with great enthusiasm and in great numbers (64). They registered 238 different projects.

The universities were not entirely ready to engage their students because of their exams and the curriculum planning. So it lasted until the 15th of November before the first students arrived in the system and started to write proposals. Then since there were only a few students (7), we extended the period for writing proposals until the 15th of December. We came to 12 proposals from which 7 were accepted.

With the announcement on the front page of the VPS system and an email to all the participating mentors for permission, we transferred the mentors, tutors, students, and different organisations and of course the projects from the first pilot to a second one and started that from the 15h of January.

In that pilot instance we had 24 proposals from which 10 were accepted. The period for student registration and the period for writing the proposal was extended to make this possible. The promise of more universities being involved and the ever more complex task of finding a time schedule that matches all these different curricula, made us decide to let go of the Google Summer of Code idea of strict time slots for each phase.

¹ At the university of Salamanca it was obligatory to sign a contract with the participating organisation before the students could start with a project.

In the last pilot, we made it possible for persons to register any time as student or mentor for example and enter a project or a project proposal. For mentors we changed the logic to make the availability of a project dependent on the availability of the mentor. This more flexible time scheme (a pilot only has a beginning and an end –the next year) has as a consequence that there is less competition between students. We noted no decrease of participation of organisations after this change. The projects were either not critical or seen more like an experiment with no strict deadlines.

In the VPS system there is the possibility to finalise the project. That is to say, just like at the beginning of the project a kind of agreement is signed between the mentor, the student and the tutor that the project will be carried out and under which conditions, the project should be officially ended in the system. We noted that almost nobody really filled in the enquiry and the finalisation form. We are not sure whether to take this as a loose of interest since there is no incentive in the process why people should return to the system to perform this step.

4. Process of Semester of Code

The Semester of Code concept is about matching the need for performing non critical project ideas to the need to have practical work experience for students during the academic year. To enable this process, students and their tutor register in an online system and enter a proposal for a project that was entered before by a mentor who offers to be a mentor for this project.

This matching process can either be guided by phases enforced by the system (like it was done by Melange in Google Summer of Code) or by a more loose matching approach around the projects which have an available status that depends on a preset time period –by the mentor. Both are possible; it is a matter of a setting. But of course one has to choose per Semester of Code instance which way to go.

Since we have experienced difficulties in getting enough students to participate in Semester of Code, we have to broaden the range of higher education institutes that we aim for. That could mean that we not only approach universities but also other computer science higher education institutes. By having a large number of institutes there is more chance of competition between students, but also the danger of incompatible curricula which makes it impossible to have a timeline that covers all. So a more flexible matching process is then desirable.

Basically we have then the following phases in the matching process for a company.

- Decide that there is a project idea that is not critical and doable in about 3 months
- Find a mentor that is willing to counsel the project
- Find an instance of Semester of Code that is active
- Contact the corresponding contact person to get a registration code to be able to register
- With that code, register the company, the company contact person and the mentor
- Register one or more projects
- Possibly change the project idea, based on the comments placed on line with the project. Comments can be placed by tutors and students and the moderator of the site.
- Judge the incoming proposals and mark them as either rejected, accepted or comment on the proposal.
- Once the mentor marks the project as accepted, he/she and the student and the tutor have to click an agreement for acceptance. In this agreement additional appointments can be written about contact frequency, time scheduling etc. But this is not obligatory.
- Then the actual work takes place.
- After the work is finished to satisfaction, the mentor contacts the tutor and the student to mark the work as done. This should also be done in the VPS-system.

The work done is open source so it stays visible in the online version management system of the community.

5. Roles in Semester of Code

There are a number of roles fulfilled in the matching process.

There are students, supervisors and institute administrators for the academic site. In the system there is only one independent role: the administrator. This is the technical person within the university or organisation setting up a Semester of Code instance. He/she installs the VPS and is able to monitor the database and solve issues. Then there is the Semester of Code moderator –also called soc- who has almost the same rights as the administrator, but has no direct access to the resources of the system.

On the organisation side of the roles, we have two more roles: the organisation administrator and the mentor. The organisation administrator registers an organisation and is able to distribute a registration key to the mentors of his organisation. In a lot of cases, the roles coincide. So in practice the organisation admin is also the mentor.

To organise students for tutors, we divide them into student groups or classes. There is no limit to the number of student groups and the number of members. It is just a way to organise students. Student groups have no influence at all on the matching process from the perspective of organisations.

6. The Implementation of Semester of Code

The most important is to find an instance of Semester of Code with enough students attached to it via the participating institutes, so that your project will be seen by enough bright students that will provide some solution to your problem. Since large numbers are important here, we try to bundle the initiatives into one organisation that tries to organise this Semester of Code event once a year. This initiative is currently being setup by the University of Salamanca. It gets its own board and enthusiastic members to run it.

So suppose you contact that organisation, they will likely use the same vps and so the same registration procedure. You will be handed a registration key to register your organisation with some description etc. by email. The person doing the registration will be Organisation admin automatically. Next, you can produce a registration key yourself and register yourself as a mentor or give the keys to members of your organisation to become mentor for your organisation.

All mentors and organisation administrators can act as creator and mentor of a project. The organisation administrator is though the only one who can change or withdraw the organisation.

A more detailed description of the registration procedure can be found in the document “*HowToVPS_SemesterofCode_VPS2*”². How the VPS functions, can be found at the user manual incorporated in the online VPS itself.

Currently all instances of the VPS, have the form:

[*instance name*].semesterofcode.com. On the website of the organising initiative, you will find more information on where to find the instance of this year. It will be likely something like vps[*current year*][*next year*] like “vps1516”.

Before you start participating with an instance, make sure to know whether it is a flexible or a timeline steered instance. Both have advantages and disadvantages for you. With the timeline steered instance you will be held to follow the timeline and have less freedom, but you will more likely have more students at the same time offering a choice in proposal. With the flexible approach, you can mark the project as being available for a certain period. It offers you maximum control, but when you are not so lucky no or only a few students will write proposals for your project.

7. References

García-Peñalvo, F. J., Griffiths, D., Cruz-Benito, J., Veenendaal, E., Achilleos, A. P., Wilson, S., & Kapitsaki, G. (2016). Understanding the barriers to virtual student placements in the Semester of Code. *Education in the Knowledge Society*, 17(1). *In press*

Veenendaal, E., (2015). *HowToVPS_SemesterofCode_VPS2*. An extract can be found in the online manual: http://vps1516.semesterofcode.com/sites/all/modules/vals_soc/help/index.php?id=20