

# GNU social Summer of Code 2020

## Transcript for Eliseu Amaro using Portuguese ECTS scale and a student provided grading method

Joshua Judson Rosen      Phablulo Joel      Diogo Peralta Cordeiro      Daniel Supernault

This document is a custom transcript done as per request using the Portuguese ECTS scale in addition to the one adopted during the GNU social Summer of Code 2020 as well as a student-provided method (made available in the appendix of this document).

### Document Structure

- Assessments: We describe the reasoning behind the given assessment and the method
- Appendix: Hours Rules
- Appendix: Milestones
- Appendix: Mentoring Staff functions
- Appendix: GNU social Summer of Code 2020 Grading System
- Appendix: Student Provided Grading Method
- Appendix: Funding Note
- Appendix: Portuguese numerical scale and ECTS grades
- Appendix: Final Report Instructions given to our students

### Attached Deliverables

- A tarball with a copy of the small reports the student had to write daily while enrolled in the programme: `daily_reports.tar.gz`
- Major final tech report written by Eliseu Amaro documenting the work done: `technical_report.pdf`
- GSSoC 2020 Certificate of Completion for Eliseu Amaro: `certificate.pdf`
- GSSoC 2020 Completion Verification Letter for Eliseu Amaro: `verification_letter.pdf`

### Verifying the authenticity

We have provided detached GnuPG signatures for each attached deliverable.

Import the public keys:

- Diogo Peralta Cordeiro: `curl -sSL https://www.diogo.site/public.asc | gpg --import -`
- Joshua Judson Rosen: `curl -sSL https://www.hackerposse.com/~rozzin/gpg-public-key.asc | gpg --import -`

Verify the document:

```
gpg --verify "doc_mentor.sig" "doc.ext"
```

Where `mentor` stands for the first name of the mentor, `doc` stands for the doc filename you want to verify, and `ext` to the extension of the doc being verified (i.e. `pdf` or `tar.gz`).

## Contacts

Should you require any further information, please do not hesitate to contact via email:

Diogo Peralta Cordeiro <mail@diogo.site> and Joshua Judson Rosen <rozzin.gsoc@hackersposse.com>

## Workload

Eliseu was accepted into the program by, and developed free source software for, GNU Project beginning March 31, 2020 and ending on August 24, 2020. Their contributions consisting of  $146+438=584$  hours of workload are in the field of Web Technologies, Accessibility and Design.

Over that period, they passed each of the three evaluations conducted by their mentors with the final grade of 42, 1337, and 1337, respectively. The assessment method is available in the appendix of this document.

A Portuguese ECTS is 28h of student work. Hence, we suggest  $5+16=21$ ECTS, proposal + internship.

## Assessments

### Proposal (April)

Each year many students apply to participate with only a portion of them being accepted into the program, and not every student finishes the program successfully. Eliseu was approved in the proposal phase, meaning he was able to:

- work on an idea that solves real problems and is realistic;
- show understanding of the project;
- propose a well sustained roadmap;
- engage with the community and discuss their ideas.

Grade in this proposal phase: 42

### Using the student provided method:

Autonomy was High, Objectives were High, and Difficulty was Low. Using the ECTS scale we have now to choose between a grade in C(14-15) and B(16). Given the good presentation and the initiative to make a commit fixing some bugs in the existing themes, it seems fair to classify this as “Good”, which is the whole idea behind a 42 grade in the originally employed method.

Grade: 15 out of 20.

**N.B.:** The grade of the proposal phase did not accumulate with the internship phase – in our assessment.

## Internship/Project

### May + June

The workload decided for June was actually started in May. So a one-month workload was spread in two months to better accommodate the parallel student college exams season. This was done following Google Summer of Code recommendations, which this program derives from.

## Formal Hours Requirement

The student dedicated 137h.

Status: Met.

## Objectives satisfaction

Met:

- Further design work
- Develop at least the skell of the core template
- Port at least the static pages

Started:

- Implement the dynamical components of the core views
- Port core actions to controllers and implement their views

Pending:

- Implement visual customization plugins

Merge Request: <https://notabug.org/diogo/gnu-social/pulls/162>

Result: 42

**Using the student provided method:** Average

## Intrinsic difficulty level of their work

The student wasn't familiar with Web Development nor with the Symfony framework and its components. On the other hand, these tasks can be demanding for new blood although not complex.

Result: 42

**Using the student provided method:** Low

## Autonomy with which the work was done (Team working Skills)

- Some autonomy but with blocking situations that required the intervention of the mentor.
- Average dedicated weekly mentor time: 6h
- Submitted quality daily reports regularly.

Result: 42

**Using the student provided method:** Average

## Grading

Result: 42 (Pass)

**Using the student provided method:** 12-14 range, using the ECTS scale we have now to choose between a grade in D(12-13) and C(14). Our 42 grade was meant to always imply a "Good" work, thus 14.

## July

### Formal Hours Requirement

The student dedicated 132h. This is over the minimum allowed 128h/month.

Status: Met.

## Objectives satisfaction

The [] means optional task / wasn't demanded to be done just suggested in case the others were done

Met:

- Finish new design
- Port core UI actions (but feed)

Pending:

- [Implement visual customization plugins]

Merged: <https://notabug.org/diogo/gnu-social/src/v3>

Therefore, the objectives were reached up fully.

Result: 1337

**Using the student provided method:** High

## Intrinsic difficulty level of their work

The CSS written reveals that the student has acquired a high level of proficiency in web frontend development. Given that the student started studying the technological stack used to accomplish this in this summer, the intrinsic difficulty was high.

Reference: <https://www.youtube.com/watch?v=JrsFWSMHzeY>

Result: 42

**Using the student provided method:** Average

## Autonomy with which the work was done (Team working Skills)

- Average dedicated weekly mentor time: 3h
- Submitted quality daily reports regularly.

Result: 1337

**Using the student provided method:** High

## Grading

Result: 1337 (Pass)

**Using the student provided method:** 16-18 range, using the ECTS scale we have now to choose between a grade in B(16-17) and A(18). Our 1337 grade was meant to always imply a "Very Good" work, thus 17 – as per the student provided method, "grades above 16 should be given to very good or excellent work, supported by a well structured, objective and complete report, and performed with great autonomy".

## August

### Formal Hours Requirement

The student dedicated 143h. This is over the minimum allowed 128h/month.

Status: Met.

## Objectives satisfaction

Met:

- Implement every core action (but APIs)
- Implement activity interaction buttons

Started:

- Port feed controllers
- [Port Directory]

Merged: <https://notabug.org/diogo/gnu-social/src/v3>

Therefore, the objectives were reached up fully.

Result: 1337

**Using the student provided method:** High

## Intrinsic difficulty level of their work

The CSS written reveals that the student has acquired a high level of proficiency in web frontend development. The student had to write backend queries and learn about Symfony controllers and so on.

Result: 1337

**Using the student provided method:** High

## Autonomy with which the work was done (Team working Skills)

- Average dedicated weekly mentor time: 3h
- Submitted quality daily reports regularly.

Result: 31337

**Using the student provided method:** High

## Grading

Eliseu has continuously improved over time and has done all the requested tasks with quality and minimal help.

Result: 1337 (Pass)

**Using the student provided method:** 18-20 range, using the ECTS scale that's an A. Our 1337 grade was meant to always imply a "Very Good" work, thus 18.

## Final Grade

- The proposal phase grade was 42 -> 15 out of 20. The proposal phase corresponds to the first 146h of workload and does not accumulate with the internship/project work.
- The final internship/project work grade was 42, 1337, and 1337. Which results in 1337. Using the student provided mechanism, the final internship/project grade was: 14, 17, 18. Thus 16.

## Appendices

### GNU Summer of Code Hours rules

Note: GSoC = Google Summer of Code

With respect to the time you will have to dedicate, GSoC demands 30h to 40h of work per week. GNU social's Summer of Code expects you to work between [32, 36.5]h/week, you can organize that time as you please, but you must be sure to dedicate that in your weekly work or to be overly productive.

We suggest you to do a four-day work week with 6h of work/day + 2h to document, review and test and report the progress you've done. As breaks are important, we recommend a 1h lunch break, 15min break after 4h of continuous work and a further 15mins break after 6h of work. These breaks won't be considered as part of your work time.

In some places, GSoC starts in the middle of college's last exam season, if that's your case, we accept that you start doing some hours of work in May (bonding month) and debit those in June (first coding month) weeks.

In average, you will work 146h/month, ideally 128h/month will be enough (maybe even only 96h/month, if you're special). We do not accept that you transfer expected work time from a month to another. An under-performing week will make us request more hours from you either in the same week or the one immediately after.

### Agreed roadmap with Eliseu

Defined goals:

June

----

- Develop the skeleton of the core template
- Port every static pages

July

----

- Finish new design
- Port core UI actions (but feed)

August

-----

- Implement feed controllers
  - |- /home
  - |- [/groups]
  - |- [/people]
- Implement every remaining core action (but API)
  - |- Reply
  - |- Favourite
  - |- Recycle

Given as optional tasks

----

- [- Implement visual customization plugins]
- [- Port plugins to v3]

### Mentoring Staff

This programme promotes a high individuality. Being the mentors usually only expected to dedicate about 4h of assistance weekly. Thus the high entry barrier to only let students ready for such a demanding experience in.

**Joshua Judson Rosen (Main Mentor):** Provided the requirements for an accessible interface and the principles of a lightweight interface inside what's to be expected in GNU social.

**Phablulo Joel (Technical Mentor):** Provided support regarding CSS and JS issues and challenges that the student would sometimes face.

**Diogo Peralta Cordeiro (Programme Organizer):** Set the milestones, assigned tasks, ensured the deadlines were respected and the daily reports were accurate, and reviewed the work periodically. Has also provided the necessary insight about the backend and architecture to be followed.

**Daniel Supernault (Technical Mentor):** As the maintainer of Pixelfed, provided external feedback on the presented designs.

## Grading system adopted in GNU social Summer of Code

### Effective Grading

Either pass or fail.

### Qualitative Grading

Grade	Definition
31337	Outstanding
1337	Very Good
42	Competent
0	Failed

### Method

Graded every month, the ceiled average is the final.

No Quantitative Grading system will be used.

The contributions will be evaluated according to the following directives:

### Autonomy with which the work was done

- Low (was unable to progress autonomously);
- Competent (some autonomy but with blocking situations that required the intervention of the mentor);
- Very Good (very good ability to solve problems independently in useful time with the mentor mostly focused on defining the next steps of the work);
- Outstanding (is sometimes able to suggest surprising and valid alternatives to what was originally planned by the mentor).

### Objectives satisfaction

- Low (haven't reached to the minimum objectives admissible for the proposed work);
- Competent (the objectives were reached though not entirely);
- Very Good (reached up fully);
- Outstanding (exceeded up the objectives set).

### Intrinsic difficulty level of their work

- Low (relatively easy work, both from a scientific or technical point of view, based on widespread knowledge);
- Competent (work with some high complexity of details requiring more advanced
- knowledge/expertise, either technical or scientific);

- Very Good (relatively complex work, requiring a substantial knowledge and technical skills, or resulting in some innovative contribution);
- Outstanding (InSaNe).

### Grade formula per month

Satisfaction of objectives	Autonomy level			
	Low	Competent	Very Good	Outstanding
Low	0	0	0	0
Competent	42	42	42	1337
Very Good	42	1337	1337	31337
Outstanding	42	1337	1337	31337

Difficulty level	Grade from matrix above			
	0	42	1337	31337
Low	0	0	42	1337
Competent	0	42	1337	1337
Very Good	0	1337	1337	1337
Outstanding	0	1337	1337	31337

### Modules in GNU social Summer of Code

**N.B.:** The following are the minimum averages in GNU social's Summer of Code, we will come up with a custom "transcript" for any interested student. The mentorship time is derived from Google Summer of Code 2020 guidelines with average mentor expected time to be 4h/week.

### Web Technologies

#### Amount of time allocated to each module unit

Designation	Time (hours)
Autonomous study	80
Mentorship	20
Project work	46
Total	146

#### Assessment Components

Designation	Weight (%)
Proposal	80
Proof of Competence	20

#### Proposed Credits:

- 1 Carnegie Unit
- 5 ECTS



## Internship | Training

### Amount of time allocated to each module unit

Designation	Time (hours)
Internship	276.5
Autonomous Study	93.5
Final Report	24
Mentorship	44
Total	438

### Assessment Components

Designation	Weight (%)
Practical or project work	100

### Proposed Credits

- 4 Carnegie Unit
- 18 Austria, Italy, and Spain ECTS
- 16 Finland, The Netherlands, Portugal, and Russia ECTS
- 15 Germany, Belgium, Romania, and Hungary ECTS

### Student Provided Grading Method

We were given the following method as being the student's university criteria for internships assessment.

### Calculation formula of final grade

The internship will be evaluated according to the following directives:

### Autonomy with which the work was done

(Information to be provided by the external supervisor)

- Low - difficult to progress autonomously;
- Average - some autonomy but with blocking situations that required the intervention of the advisor;
- High - very good ability to solve problems independently eat intervention advisor focused on defining the next steps of the work.

### Satisfaction of objectives

(Assessed by the report and the external supervisor's information )

- Low - reached to the minimum objectives admissible for the proposed work;
- Average - the objectives were reached though not entirely;
- High - reached up fully, or even exceeded up, the objectives set.

### Intrinsic difficulty level of their work

(Assessed by the report and presentation/discussion)

- Low - relatively easy work, both from a scientific or technical point of view, based on widespread knowledge;
- Average - work with some high complexity of details requiring more advanced knowledge/expertise, either technical or scientific;
- High - relatively complex work, requiring a substantial knowledge and technical skills, or resulting in some innovative contribution.

Satisfaction of objectives	Autonomy level		
	Low	Average	High
Low	E	D	C
Average	D	C	B
High	C	B	A

Difficulty level	Grade from matrix above				
	E	D	C	B	A
Low	10-12	11-13	12-14	13-15	14-16
Average	11-13	12-14	13-15	14-16	16-18
High	12-14	13-15	14-16	16-18	18-20

Grades above 16 should be given to very good or excellent work, supported by a well structured, objective and complete report, and performed with great autonomy.

### Eliseu Funding note

They received a stipend of 3000 USD for the project in the form of charitable donations from The Freaks Club. Eliseu was not directly employed by GNU or The Freaks Club during GNU social Summer of Code 2020. Eliseu worked from home on their project and was not required to visit either GNU Project or The Freaks Club's offices.

### Portuguese numerical scale and ECTS grades

1. The final assessment of a course unit is expressed in a numerical grading from 0 to 20, with 10 as the minimum passing grade, as well as in its equivalent ECTS grading scale (from A to E).
2. The final classification of a degree programme is expressed in a numerical grading from 10 to 20 within the whole numerical scale from 0 to 20, as well as in its equivalent ECTS grading scale (from A to E). The final classification is calculated using the weighted average number of ECTS credits earned in each approved course unit, which is eligible for the completion of the degree programme.
3. The final numerical classification of the degree programme can be accompanied by a qualitative classification, expressed by the following terms: Sufficient (10-13), Good (14-15), Very Good (16-17), and Excellent (18-20).
4. The ECTS grading scale is based on a percentile statistical distribution, which splits into five percentage segments also called A, B, C, D, E according to the following table:

ECTS Grades	% of successful students normally achieving the pass grade
A	10
B	25
C	30
D	25
E	10

The following table (provided by the Portuguese Directorate-General for Higher Education, Order 11196) can be applied.

ECTS Grades	E	D	C	B	A
Qualitative definition	Sufficient	Satisfactory	Good	Very Good	Excellent
Portuguese numerical scale (10-20)	10-11	12-13	14-15	16-17	18-20

According to the Portuguese legislation (DL 42/2005 of 22 February), the grades below 10, in the 0-20 numerical scale, are failing grades and do not have a corresponding ECTS grade.

### **GNU social Summer of Code Final Report Instructions**

The report must have A4 paper geometry with 12pt font-size. The pages, sections and figures must be enumerated. Maximum of 5000 words. These limits don't include cover, indexes nor appendices. The report must be well structured.

One example of a suitable format is:

- Cover
  - Title / Author / Date
- Mentors Page
  - Page for the mentors to sign and make the necessary comments (namely a declaration confirming the student has evidenced certain knowledge throughout the summer and worked the declared hours). Note that this is mandatory.
- Abstract
  - A brief summary of approximately 250 words. Note that this is mandatory.
- Preface
  - Explaining when, where, and why the project was carried out, and thanking those who helped.
- Table of contents
- Introduction
  - Introduction to the done work, and description of the objectives during the summer.
  - Short background, issue and aim, as well as the structure of the report.
- Methodology
  - Theories, tools, etc., that were applied in the project.
  - Description of used technologies as well as discarded alternatives.
- Results
- Discussion and conclusion
  - Reflection on whether the aim was fulfilled, and on the opportunities for further development. Critical review and discussion of the results.
- References
  - In order to enable the reader to review the project and go to the original sources on which the material is based, the report needs a good reference list.
  - Continuously filling in the reference list can save you a lot of time compared to doing everything at the end.
- Appendices
  - Extensive data material that is relevant to the work, but too large to incorporate into the text, can be included in an appendix.