Delegate Support System - Platform for Questionnaires on the Duration of Assessments

Filipe Rafael Soares
filipe.soares@tecnico.ulisboa.pt

Instituto Superior Técnico, Lisboa, Portugal

April 2016

Abstract

Fenix is the information management system - at Instituto Superior Técnico (IST) - that processes the information regarding the University degrees. It collects information about the courses being studied, students and teachers. Within this system, the element 'course' is represented by the object curricular unit that is subsequently composed by three domains: evaluation, students and teachers.

The organisation of the academic year requires that the number of hours associated with the evaluation of the different curricular units does not to exceed a certain limit. The Fenix does not have any tools to question about the number of hours used by the different elements. This requires the use of external mechanisms like google forms that allow to question and therefore get student feedback about overload, positive aspects or anything that needs to be improved regarding evaluation element's and or course.

The identification of this gap about data collection with a specific criterion led to the development of this dissertation. Its goal is the development of a platform able to question students about the duration of the courses evaluation elements. In order to make easy to expand, the platform was build on a modular fashion, using the Django e FenixEdu *Application Program Interface(API)* for the backend; bootstrap e jinja for the frontend. The platform was subsequently put to user tests which lead to the conclusion that it was fit for purpose and is overall user friendly apart of certain aspects at its onset. This platform is operational and running since November 2019

Keywords: Questionnaire; Evaluation Elements; Curricular Unit(CU); European Credit Transfer System(ECTS); Software Engineering;

1. Introduction

According to Martyn Denscombe [2, p. 166], a questionnaire should meet three criteria:

1. Should be designed to collect information which can be used subsequently as data for analysis;
2. Should consist of a written list of question. Occasionally images can be use.
3. Should gather information by asking directly questionnaire scope.

With this, we can define a questionnaire as a set of written questions, whose purpose is to gather information for further analysis, by asking directly about the desire scope. With the information gathered it is possible to create a conclusion about every single question and a overall picture about the focus of the questionnaire.

Martyn Denscombe [2, p. 167] says that a success of a questionnaire must be measure according with three aspects:

1. Ratio of answers – How many answers were obtained;
2. Ratio of complete answers – From the answers given, how many answered all the questions;
3. Validity of responses – How honest and accurate the answers are;

The information manager system at IST (Fenix) process the information regarding the University degrees. It manages information about courses, students, teachers and more. At IST, each course is associated with a level of work, that is measure according to the European Credit Transfer System(ECTS). This unit has some subjectivity behind it, for being complex to measure and by not being universal the conversation ratio inside the academic community. The mathematics behind this unit, counts with the frequency of the course, work done ("homework", projects, internships) as well with the study time.

At IST, 1 ECTS is converted to 28 hours of work[1]. This means that to end the 1st cycle, you need to invest 5040 hours to have the degree (28 hours $\times$ 180 ECTS = 5040 hours). To others universities and even for some teacher, the conversion ratio changes, fluctuating between 25 and 28 hours.

Having by example a course of 6 ECTS, this means that it will take a total of 168 hours (T) to complete this course. However, this amount of time is splitted into two groups. One related with the contact time (C – Time spend on theoretical and practical classes) and other to the autonomous work (TA – Time spend on research, study, work elaboration, and more). Normally, the value for C will round around 63 hours while TA will be 105 hours. The problem appears when the students spends more than 105 hours to TA, because this may make the course's work excessive.

The autonomous work is estimated by the teachers responsible by the course, however, this is a hard task because of the subjectivity on the measuring process. When doing this estimation, if the real value to the autonomous work end up being superior to the one expected, this may end up on putting the semester at risk, because the excess of time spend on a course, ends up stealing time from others courses. According to the Portuguese school system, the work time spent per week should not be superior to 40 hours, however this is something that happens in several weeks.

In order to track which course is stealing more time to the students, and better understand what can be different on the following year, it is necessary to question the students about the time spent on their work. With this purpose, this thesis consists on developing a tool capable of making questionnaires that directly asks about the time spent on evaluations like projects and labs.

1.1. Requirements

The main purpose of this tool is to question the students enrolled in CUs, about the duration of evaluation elements (like projects and labs), in the most automated way possible and later, process the answers. The questionnaire, in the case of being a project, can also question the students' satisfaction with the theme of the project.

The requirements raised for this tool can be grouped into 3 groups, which are:

- Information Collection;
- Questionnaire Manager;
- Information Processing.

[1] https://tecnico.ulisboa.pt/pt/ensino/estudar-no-tecnico/perguntas-frequentes/

1.1.1 Information Collection

These requirements refer to the information that must be collected when filling out a questionnaire, as well as guarantees that you must ensure (anonymity and authenticity). The requirements are:

1. The platform must collect the following information about the respondents, either automatically or manually:

   (a) Time invested by the group (manual and mandatory);

   (b) Individual invested time (manual and mandatory);

   (c) Degree, curricular unit and number of enrollments (preferably automatic and mandatory);

   (d) Grade that you expect to obtain in the work done in relation to the submitted (manual and mandatory);

   (e) Strengths, points to improve and comments/suggestions (manual and optional).

   The collection of time spent by the student and the group are the focus of this project. The degree, curricular unit and number of enrollments allows to better characterize the answer, as well as to make a more detailed analysis. The collection of the expected grade, when compared to the real grade, allows to perceive the students' perspective regarding the subject, being possible to comment if the curricular unit had too strict criteria or if the respondents were not aware of the subject. Strengths, points to improve and comments/suggestions are fields that later help the delegate to fill in the QUCs and also allow to transmit better information about what happened.

2. The platform must ensure that the respondents are students enrolled in the curricular unit of the questionnaire;

   It is important to ensure that the students who answer the questionnaires are actually the students who are enrolled in the course, as well as ensuring that they answer at most once.

3. The platform must guarantee the anonymity of the respondents:

   (a) In the event that the number of responses allows the identification of a respondent, responses should be grouped.

Anonymity of respondents is an essential aspect for obtaining honest comments and feedback.

4. The platform must indicate when the number of responses does not guarantee representativeness (for example, with the representativeness formula applied by the pedagogical council);

   Despite not having the necessary number to represent the curricular unit, the information is considered, although little, important and must be presented, indicating only that it has no representativeness.

1.1.2 Questionnaire Manager

These requirements concern the management of questionnaires, such as the states that one must have or the sending of notification about them. The requirements are:

1. A questionnaire can be in one of the three states:
   (a) Created - Status displayed after creation, where it is possible to change the release date;
   (b) Open - Status that allows users to respond;
      i. This state is active when:
         A. On the day of delivery delivery;
         B. In the week of delivery;
         C. One day stipulated by the delegate;
         D. Manually.
   (c) Closed - Users are unable to respond.
      i. This state is active when:
         A. Two weeks after opening;
         B. One day stipulated by the delegate;
         C. Manually.

   The passage of a questionnaire to the Open or Closed state are two very important aspects, since they give the platform a certain degree of automation, without removing the freedom of the delegate to manage the process in a more manual way.

2. The platform should contain information about the closing date of projects or laboratories, preferably automatically, but it should allow the delegate to insert this information if necessary. It should also allow the user to enter information about other possible assessment that should be the subject of questionnaires:
   (a) In the case of a project, the closing date corresponds to the delivery date;
   (b) In the case of laboratories, the end date corresponds to the end of the week in which the evaluation took place;
   (c) In the case of other elements, the end date must be one of the possible hypotheses mentioned above.

   Obtaining or introducing the dates of the evaluation elements that justify the execution of the questionnaires, makes the system more automated and removes the work that falls under the delegate. The existence of a manual part in this process is essential, as not all course units have dates on the Fenix and not all assessment elements have a date on the Fenix (laboratories, for example).

3. The platform must notify the target students of the questionnaire about its existence when in the open state to be filled out:
   (a) Sending an e-mail informing that the questionnaire for a given CU is already accessible. This e-mail can be modified by the delegate, but he must ensure that the link for filling in is present;
   (b) If possible, when logged into the Fenix, a message will be displayed on all questionnaires that this user has not done. When there are no questionnaires to complete, this message will not be displayed.

   Disclosure mechanisms are essential and a focus of the project, as they aim to reduce the number of abstentions and maintain a constant and high response rate throughout the semester.

4. The platform should allow the delegate to periodically resend an email as an incentive to fill out the questionnaire:
   (a) This reminder should be restricted only to people who have not yet filled it out;
   (b) The email can be free or based on a format, but it should always contain the questionnaire.

   Related to the previous topic, the intention is to reduce the number of responses abstained. There are times when the questionnaires are poorly filled, so it is believed that with some recall of the subject, the number of responses will be more constant throughout the semester.

5. The platform, when the status of the questionnaires is opened, should show users (delegates, students or teachers) only the number of responses, the closing date and how much time is left until the closing date.

   This point complements the previous one, providing the delegate with a dashboard. In this way, it allows the mentioned users to have a notion of how many questionnaires have already been filled out, and the delegate can send the email to reinforce the filling, or to the teacher during classes to reinforce the importance of filling out the questionnaires, or even the students themselves to ask the remaining students that fill them.

1.2. Information Processing

These requirements correspond to the processing of information that must be done when the collection of responses is completed, that is, when the status of the questionnaire is closed. These are:

1. The platform, when the status of the questionnaire is closed, should perform a summary of the data, showing:

   (a) Number of responses to the number of respondents, minimum, maximum, standard deviation, average (total and without 5 % of extremes).

   It allows a quick overview on the collected aspects.

2. The platform must allow the extraction of all the information collected when the status of the questionnaire is closed, in Excel format (.xls)

   It contains all the information collected and anonymized, allowing for further analysis.

3. The platform must notify all parties and allow viewing of responses when the status of the questionnaire is closed:

   (a) The coordinator must have access to the results of all questionnaires carried out in the course he coordinates, organized by year and semester;

   (b) The conductor and the teachers must have access to the results of the questionnaires carried out to the curricular units they teach;

   (c) The delegate must have access to the results of the questionnaires for the year he is responsible. The cycle delegate must have access to all results in the course, organized by year and semester;

   (d) The student must have access to the results of the questionnaires he has answered.

   The notification of the closing of a questionnaire is essential, as it allows an idea of what happened in the respective delivery and allows the delegate, conductor and coordinator to intervene even in class time, something that is not always possible due to delays.

2. Related work

In order to analyze characteristics of systems/platforms, we proceded to research similar tools, with emphasis on Google Forms and Moodle as they can be inserted in an institutional environment.

During this research other tools were considered, but were discarded since they did not meet the desired requirements, or they had a daily limit, with usage costs. In this chapter we intend to analyse to what extent each of the chosen tools fulfills the mentioned requirements, and present a vision for the potential of the tool, that produces the desired effect, inserted in Fenix. Finally, comments and an overview are presented on the subject addressed.

2.1. Google Forms

Google Forms[2] is a tool present in Google Drive, that allows users to build questionnaires through an intuitive interface. This, together with Fenix's functionality of sending emails to students according to the curricular unit (CU), is the current methodology followed by LEIC-T delegates, where they carry out the process described in subchapter 1.2 at [4].

Google Forms allows users to build questionnaires with a high degree of complexity, and a wide range of possibilities for the type of questions/answers that allows the flow of the questionnaire to be conditioned by the selected answers. The types of questions/answers are: Short answer, Paragraph, Multiple Choice, Checkbox, Dropdown, File Upload, Linear Scale, Multiple Choice grid, Checkbox grid, Date and Time.

As a Google Drive tool, Google Forms works together with others, namely Google Spreadsheets. This tool corresponds to an Excel sheet where the answers given in the questionnaires are stored. Based on this sheet a brief summary of the form is generated.

Another aspect of the described tool is the fact there is an active community to develop its plugins, offering all users a set of features on the results

[2] https://www.makeuseof.com/tag/5-awesome-reasons-to-use-google-forms/

and inquiries, such as sending documents, notification of submission, personalized mails when submitting surveys,etc.. thus expanding its potential.

With regard to authenticity, Google Forms also makes it possible to identify users, given that they have a Google account. However, to identify students from a given CU, it would be necessary to cross information from the Google account with the student's IST ID. In the absence of such an intersection, authenticity is not guaranteed. Finally, this tool creates forms reactive to any platform (PC or mobile), free of charge, without any limit to the number of questionnaires that can be created, although there may be some paid plugins.

2.2. Moodle

Moodle is an open-source software used in teaching as an online resource management platform. It allows the transmission and organization of programmatic content among other functionalities. It has three main user-types: Administrator, Teacher and Student. Each user is allowed to perform different operations to the system.

The most similar to what is intended with this document is the questionnaire tool present on the teacher's side. Considering that feature, we should keep in mind that the inquiry and the questionnaire are different objects. This difference influences the flow, something that is important in a survey and a positive aspect of Google Forms.

Regarding the anonymity of students in relation to their answers[3], there is a field in the creation of the questionnaire that allows to activate or deactivate the identification of target students. This information can also be collected and has a feature that allows you to obtain a brief summary.

Being an open-source software and having a strong community around it, there is good documentation on Moodle. However, this software as a solution to the problem would be quite complex, because in addition to having many features that would not be used, it involves an exhaustive replication of the information present in Fenix (on the CU's page). This replication from Fenix would need some changes, because since the functionality of the questionnaires is exclusive to the teachers, the delegates in this software would have to be profiled as teachers.

2.3. Fenix

With regard to the Fenix, it can be seen as a Moodle, but with less functionality and a less active community. Fenix, despite not allowing the construction of surveys such as Google Forms or Moodle, has a tool that behaves similarly to the desired one - questionnaire of "Qualidade das unidades Curriculares" (QUC) at IST.

QUC questionnaire fulfill part of the requirements desired in this project, guaranteeing, for example, authentication, anonymity, representativeness and adherence. This is scheduled to be accessible to students for a period of time after the end of the semester, which they are "obliged" to fill out in order to be able to enroll in the following semester. In addition to this "obligation", whenever a student accesses Fenix, he is reminded with a pop-up to answer the survey, encouraging students to complete it.

One of the problems of QUC surveys is the editing flexibility as it is not a survey like the ones created in Google Forms or Moodle, since the code is being written on the platform and presented to students. Changing the format or editing the survey would be quite complex. In addition, as a platform that covers the entire University of Lisbon (UL), there is a complex bureaucratic process that complicates editing features on the platform or adding new ones within it.

Like the other options, it fulfills some of the desired features, it is important to note that in terms of maintenance, reminder/incentive filling mechanisms, the collection of information for automation is the best option.

2.4. Summary

In short, none of the options fully meets the requirements mentioned in the requirements section and, as such, the need for a platform that meets the desired needs arises. It is intended, then, to collect the best characteristics of each option and replicate them (improving them if possible) in order to fulfill the project objective.

The Table 1 is evaluation, which focuses on some of the most relevant requirements in order to make it less exhaustive. With regard to automation, maintenance, authentication, reminder and disclosure mechanisms, Fenix is the best option, while building surveys, extracting and analyzing responses, Google Forms is superior. Moodle can be considered a middle ground, but considering all the features it has, it remains a "heavy" solution.

3. Implemention/Solution

Each module presents on this architecture is composed by its domain and services, which culminate in the realization of the use cases and fulfilling the requirements. Figure 1, shows the layered organization made to this tool.

In ascending order (starting from bottom to top), the layers are:

1. **Django Framework**[4] - Allows fullstack devel-

[3] https://moodle.fct.unl.pt/mod/page/view.php?id=106736

[4] Docoumentation: https://docs.djangoproject.com/en/2.2/

Table 1: Summary of the technologies pointed out crossed with relevant aspects to the implementation. 0 – Does not allowed it; 1 – Badly allow it; 2 – Almost allow it all; 3 – Allow it all

	Google Forms	Moodle	Fenix
Creation of quizzes	3	2	0
Authenticity	1	2	3
Anonymity	3	3	3
Easily to gather information automatically	1	2	3
Quiz spreading/sending	1	2	3
Reminder Mechanism	0	2	3
Easy to extract the answers	3	2	0
Summary of the answers	3	2	0
Maintenance	1	1	2

opment of web applications, using the python programming language. This is responsible for the persistence of information on the platform, as well as part of the authentication security (the rest being ensured by the CAS of IST) and the sanitization of the answers given, safeguarding the well-being of the database and the tool. It also has mechanisms for sending emails, forming groups with permissions to add, view, modify and delete elements from different domains and create dynamic presentation pages (through Jinja and the administrative page);

2. **Celery**[5] - Scheduler used in projects designed in Django, responsible for scheduling and executing asynchronous tasks;

3. **Fenix Python SDK** [6] - Responsible for accessing the endpoints of the IST information system (Fenix);

4. **GUC** - Module that allows users and content management;

5. **GRQ** - Module that allows the management of responses and questionnaires;

6. **WebApp** - Corresponds to the presentation layer, with which the user interacts. This is the combination of administrative pages with pages created from scratch in a dynamic way, using Jinja and bootstrap.

As mentioned at the beginning of the previous enumeration, the platform's authentication process results in a split of responsibilities between Django and CAS, with most of the responsibility falling on CAS. Since CAS is a mechanism that allows anyone with a valid IST-ID [7] to take advantage of IST services (such as email and Fenix), they will be responsible, through the Fenix Python SDK, ensure that it is an IST person invoking the authentication url. If the authentication process is successful, a User object is then created in the database, with information regarding the user [8]. The information collected in this process concerns the first and last name, your IST-ID and your role (student or teacher) within the faculty.

As the authentication process is ensured by the CAS and bacause it is an insecure and incorrect action to save the log in information (IST-ID and password), a "remote user" [9], being responsible for the user session on the platform, with the information previously mentioned. Depending on the user's role, a teacher or student is subsequently created, and then the rest of the information requested in the section (such as curriculum and course) is accessed/filled out.

Given that a student or a teacher can have changes in their curricular plan at any time, such as, for example, teaching a new CU (teacher or student-teacher), or enrolling in another CU (student), it is necessary to consult the information in the respective endpoints and update it if necessary. So that the information is not constantly updated whenever a user logs in, it was programmed so that every 15 days after the last log in, a comparison of the information in the endpoints is made, with what is contained in the database and if it is different, update it. This variable is found in the file "choices.py", with the name "UPDATE_DAYS".

3.1. WebAPP/CAS Registry

CAS is responsible for ensuring the authenticity of the user who accesses the application. In order to be able to take advantage of this service, it is necessary to register the web application with Fenix.

To register the application in Fenix, a set of steps are described at chapter 4.2 at [4].

After the registration, it is possible to see details of the registration. Information related to "client_id" and "client_secret" will be visible, which is crucial

[5] Documentation: https://docs.celeryproject.org/en/stable/
[6] https://github.com/FenixEdu/fenixedu-python-sdk
[7] Unique identifier assigned to students, faculty and staff
[8] Via the endpoint https://fenix.tecnico.ulisboa.pt/api/fenix/v1/person
[9] https://docs.djangoproject.com/en/3./howto/auth-remote-use/

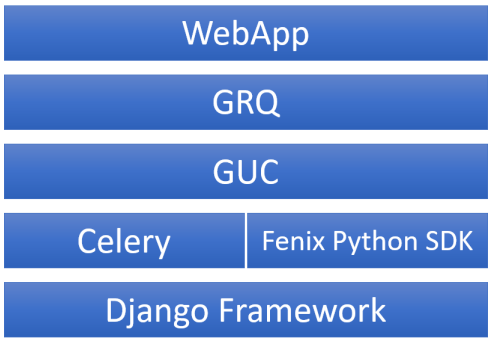

Figure 1: Layered architecture of the developed tool.

for the "Fenix Python SDK" layer and CAS to be used, once that the file "fenixedu.ini" contains that information.

The procedure for registering and using the CAS service for authenticity, are aspects common to other thesis of tool/platform development that require user recognition and access to Endpoints. An example of this is Jorge Simão Madeira Cordeiro de Aragão Goulart's thesis [1].

3.2. Observations

As mentioned in the section 4.1 at [4], Django has mechanisms for sending emails and forming groups with permissions to add, view, modify and delete elements of the domain. In order to be able to send emails, it was necessary to configure the backend to send emails using SMTP [10]. This configuration led to the creation of a gmail account (ist.delegados@gmail.com) for sending emails. However, the way the email is sent, means that if a student responds, he sends an email to the delegate and not to the email created. The email templates sent are found in the file "templates_mails.py".

Regarding the formation of groups and their permissions, it is important that when the tool is installed, the following groups are created:

1. Delegates
2. Teachers

Further information about the groups and the set of rules can be seen on the chapter 4.3 at [4].

As mentioned in the section 4.1.1 and reinforced in the section 4.1.2, at [4], the "Quiz" object is associated with a finite set of states (Figure 2). Depending on the state of the questionnaire, as mentioned in this document at section 2, there is an associated set of operations. This behavior is similar to the behavior associated with the "State" drawing pattern [3, p. 353], since the object has certain different behaviors, when its internal state is changed.

[10] https://docs.djangoproject.com/en/2.2/topics/email/

4. Validation/Results

In this section, we refer to the validation of the platform, with the process divided into: number of requirements fulfilled and the result of user tests. Despite the platform being in production, it was not possible to carry out a comparative analysis of the number of responses between the old method and the platform, since there was a weak dissemination of it by the student community resulting in an unfair analysis, given the platform's potential as a whole.

4.1. Requirements fulfilled

This subsection consists on the analysis of all the requirements that where listed at the start of this document. Due to page limitation of this document, the tables that show which requirements were or were not implemented and why, can be seen on the reference [4], chapter 5.1.

However, it is possible to affirm that the number of requirements fulfilled are much higher than those that were not. It is important to say that the reason requirements were not implemented or changed is because lack of resources or accessibility, and not because lack of knowledge or time.

4.2. User test results

A better description of the tasks performed on the User tests can be found at [4], chapter 5.2. Those tasks were performed to a random set of students from the 1st and 2nd cycle, with different number of enrollments and different degrees. The only requirement to perform the task were that the student had at least one course that was not the dissertation or the project. Also is important to say that neither of the users had an introduction to the tool, making the test without any previous input.

The results of each task can be seen on the Table 2. As is possible to see, it was monitored the number of clicks until the end of the task with success and the time (the time is rounded to the exact numbers). According to the results, the number of clicks it is not that far from the one expected, so in average the results are positive. During the test, verbally aspects were pointed as less intuitive and are presented in the thesis at the chapter already mentioned.

5. Conclusion

Given the organization at IST, the ambiguity in the academic community about the value in hours of 1 ECTS, the need to collect the information about the number of hours invested by the student on evaluation elements and the associated problems with the current solution, supports the need to create a tool like the one that was developed in this thesis, fulfilling with the requirements asked on the section 1

According to the related work shown on the sec-

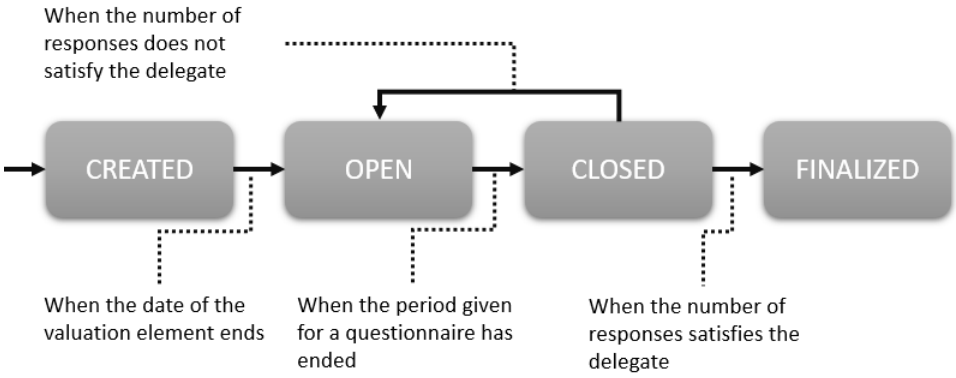

Figure 2: Finite state machine of the object "Quiz".

Table 2: Analysis of the user test results

	Num of clicks					Duration (seconds)			
	Max	**Min**	**Avg**	**Mod**	**Expected**	**Max**	**Min**	**Avg**	**Mod**
Task 1	16	5	8,53	8	4	106	30	62,09	70
Task 2	32	14	21,44	15	14	152	67	108,10	83
Task 3	25	8	11,91	12	7	77	10	40,03	45
Task 4	25	12	13,28	12	12	75	25	33,39	26
Task 5	14	5	9,34	6	5	47	12	26,69	30

tion 2, as it shown on the Table 1, none of the technologies can fulfill the overall of the requirements asked. Confirming what was mentioned on the previous paragraph, about the need to create a tool that can match the majority of the requirements.

The development of this tool was done using Django Framework and FenixAPI (in python) for the backend, with Jinja and Bootstrap for the frontend. Using UML diagrams to show the domain of the solution, its is possible to see how the objects communicate between them. It is also explained and shown the modular architecture followed on this development at the section 3. From a general point of view, the lower layers from the architecture are responsible for the authentication (using IST's CAS service) and login services (also with ITS's CAS and the user's manager session from Django). The persistence of the information, sanitation of the information saved on the database, is assure by Django Framework. The middle layers are responsible for building the domain for the problem and making the services available. Giving the need to schedule tasks to fulfill the requirements, was used Celery, that is an asynchronous schedular. Using Django SMTP backend, the messages are sent and using the group permission potentiality, groups for Teachers and Delegates were created with a specific set of rules. On the FenixAPI, bugs were found and fixed, but also during some experiences, duplicated information was received thru the endpoints.

When it comes to the tool validation, the heuristics followed to validate it were: From the total of requirements, how many were done with success; From the user's test, obtain good results that allow the conclusion that the interface is user friendly. At the start, one more heuristic was on the table, however, giving the fact that the tool had a low visibility between the students and the other extra features were not implemented, the heuristic was removed, because it would result on a fake comparation between the old and new tool when it comes to the ratio of answers. In conclusion, the tool was a success, giving the fact the majority of requirements were done with success and that, besides some details on the frontend being less intuitive, the feedback from the tool presentation was positive.

Although there were some problems during the development, the product meets the expectation, being the tool operational at RNL servers since November 2019. The ideology of making the tool more than just a simple questionnaire tool, makes it more interesting to the students. Also, the fact that with one single login the other services like, shuttle reservation, mail and scheduler generator are accessible, makes it more user friendly.

5.1. Future Work

One of the concerns when it comes to future work should be maintenance. Besides that, there is room to improve the interface experience and also make new modules and functionalities.

On the main page when a student accesses the tool, there are three tabs "Repositórios", "Opiniões" e "Procura Shuttle". The first one, is a service that consists on a single point to access all repositories that already exists thru the campi. These repositories would be grouped by Degrees for easy search, making the tool much more interesting for students.

The second one, is a “Trivago” look a like but for all courses that exists at IST. Only students would have access to this information, containing feedback about courses and teachers. The last one, it refers to functionality that would help track where the shuttle is, asking those who use the shuttle, where they are.

A more complex future work would be a creation of spaces for publicity on the tool. At first look, there is nothing that stops this idea on the RNL and IST Rules, neither for the usage of the thesis or server or the use of the rest of the content. The revenue from this adds could be used to create a fund to help on the rehabilitation of some buildings from de department, or to buy material for the labs or part of the revenue could be used as reward for those who fill more questionnaire.

References

[1] J. S. M. C. de Aragão Goulart. Sistema de gestão de informação do ipfn - gestão de recursos humanos e suporte informático. Master's thesis, Instituto Superior Técnico, 4 2018.

[2] M. Denscombe. *The Good Research Guide: For Small-Scale Social Research Projects*. Open University Press, 5 edition, 2014.

[3] A. Shvets. *Dive into Design Patterns*. Refactoring Guru, 1 edition, 2019.

[4] F. R. Soares. Sistema de apoio aos delegados - plataforma de questionários sobre duração dos elementos de avaliação. Master's thesis, Instituto Superior Técnico, 6 2020.