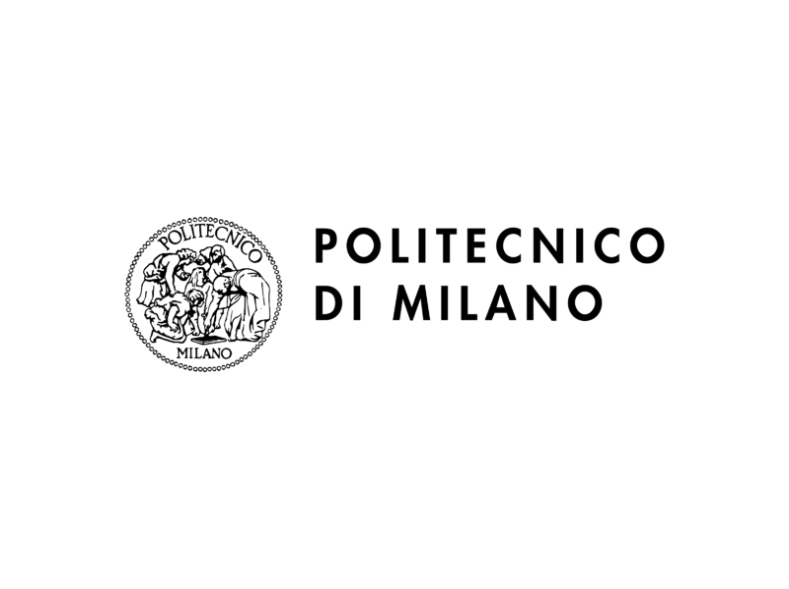
|  |
| --- |
| Petulicchio Lorenzo – Tallarico Mattia  [Data] |

|  |
| --- |
| Automation and Control Engineering |
| Requirements Analysis and Specification Document |
| Software Engineering Project |

Immagine che contiene Carattere, testo, Elementi grafici, design

Descrizione generata automaticamente 

# 1 - INTRODUCTION

This document constitutes the Requirement Analysis and Specification Document (RASD). Its purpose is to analyse the requirements that will lay the foundations of application services, to specify the application domain, the entities involved and their relationship, to clearly explain the objectives, the constraints and the features that are going to be implemented.

## 1.1 - Document structure

## 1.2 - Project purpose

The purpose of this project is to automatize the process of scheduling the exam calendar session in Politecnico di Milano. The main idea is to have a webapp that helps the university during the exam session scheduling. This project could not only help employees of the registrar's office in their job but it aims to facilitate students with their own session planning too. The project could also have secondary purposes, for example managing in the best way possible the presence of students in Politecnico facilities.

## 1.3 - Current situation and problems

Actually, at Politecnico di Milano the registrar’s office takes care to create the exams schedule for each exam session. This job is manually and not so efficient. Indeed, it takes a long time and it requires different reviews. The exams to schedule are split among personnel of the registrar's office, after this, the exams scheduling are put together and each one makes sure that there is no overlapping.

In this way a first draft is obtained, then it is sent to professors who reporting their requirements by requesting respective changes by a certain date beyond which changes can no longer be accepted

At this step, the modified draft is published to the students that analyse it and notify changes to do in order to meet their requirements.

Finally, the examination session schedule is definitively published.

With this situation the requirements taken into consideration are:

* for students:
  + exams of the same semester and same year must have two night of distance each other;
  + The exams of the same course must have at least 14 days of distance.
* for professors:
  + they can give their unavailability for what concerns days and time slot.
* for facilities:
  + the calendar is only predictive and only during the verifying phase the room are assigned;
  + It tries to do a equidistributional weighting with the number of students for each course in order to not have more than a certain number of students simultaneously.

The logic adopted to schedule the exams is: firstly are scheduled intercourses and courses with multiple staggers, secondly the mandatory exams and lastly are scheduled the elective courses exams.

Each exam is characterized only by a number (between 1 and 6) that identify the semester, the identification year and semester is not used.

**Semester enumeration**

|  |  |  |
| --- | --- | --- |
|  | January-February | June-July |
| 1st year | 1 | 2 |
| 2nd year | 3 | 4 |
| 3rd year | 5 | 6 |

This method brings several problems:

1. A lot of resources in terms of time is spent due to the flow adopted to obtain the final schedule.
2. The students most of the time receive a calendar that if difficult to fit with their necessity, for example the time to prepare well for an exam is not compliant with the distance between some exams.
3. Students receive the exam calendar not very early, so they will make their plan late without having the possibility to study during the semester in a very effective way.

## 1.4 - Targets

Our program aims to optimize student’s experience during exam session. Per questo ci poniamo questi obiettivi:

1. Massimizzare la distanza tra gli esami
   * Cosa vuol dire sessione ottima (risposte sondaggio?)
2. Soddisfare i requisiti
   * professori
   * studenti
   * aule (poli, strutture)
3. Automatizzare un processo lungo e dispendioso di forza-lavoro.
   * Renderlo più breve
   * Renderlo più robusto alle “variazioni “

## 

## 1.5 General description

The software can be used at the beginning of each semester in order to be able to publish the provisional calendar at times compatible with the personal organization of individual students with regard to their exam session.

The software is designed precisely from the needs of students who are struggling with preparing multiple exams at the same time resulting in high stress levels, etc.

In addition, our software aims to automate a process that is currently done by hand to avoid human error and optimize processing time. The users involved are:

- Active users: *secretariat*

- Passive users: *students, professors* and *Polimi*

The system will mainly be used by the student secretariat and dedicated committee members

## 1.6 Users characteristics

The people involved in the direct or indirect use of the software belong to three types: professors, students and Politecnico employees.

Professors are the PoliMi’s professors who are teacher of a course (vorrei dire che sono i “propritari” del corso) as written in the study plan card of the course, in this category we include also assistants, tutors or other collaborators of the main professor that help him during the exam.

Students, they are all students enrolled in PoliMi, from the first year to the fifth, without distinction if they are enrolled in Bachelor Degree, Master Degree or Single Courses. Everyone who have to take at least one exam, it doesn't care whether they will take an exam during the exam session or which exams they will take, and it doesn’t care if they are in course or not. Each student that can potentially take ad exam in the session in which is utilized the software is in this category. For example, students who are in Erasmus or similar program and can’t take any exam are not in this category as laureandi that have finished their exams.

PoliMi employees are firstly registrar's office personnel and commission personnel

These three classes are divided in this way

Direct user:

* PoliMi employees (registrar's office personnel and commission personnel)

Indirect user:

* Professors
* Students

Direct involved entities

* **Registrar's office employees:** they are the ones that will directly interact with the software, in particular with the GUI because they insert input data, they make the software perform the scheduling and they receive the output (calendar).
* **Professors of commission:** They communicate independently with their colleagues to gather information about unavailability of each professor of each course. Actually is required that professors need to communicate each own unavailability, otherwise it will not consider in scheduling.

Indirect involved entities

* **Students**: they receive the software output (a calendar with all scheduled exams). They can find the calendar on Polimi website, and their role is to decide which exams take and in which call take it
* **Professors**: They have to communicate their unavailability in terms of days or time slot or request about cable classroom necessity to registar’s office via email or form

- Referenti spazi ed aule

# 2 - FEASIBILITY STUDY

## 2.1 - Technical and technological Feasibility

All the technologies (better explained in the Design Document) are available on the market and largely diffused so the project is feasible in terms of realization and maintenance. The knowledge needed to realize the project are notions from different fields: web app, database, back-end and optimization algorithm. But being very used nowadays it is easy for other people to develop the software.

The critical point lays in the optimization algorithm because depending on the parameters such as number of exams, available days, constraints and weights parameters it is possible that a feasible solution can’t be provided, so having soft constraints and changing the weight parameters can be fundamental to obtain a feasible solution. Moreover, regarding the hardware there aren’t strict requirements, the only things to take care about is the ram of the processor that is necessary to solve the optimization algorithm in a reasonable time.

## 2.2 - Resource and Time Feasibility

## 2.3 – Student Survey Responses

To study the feasibility of the optimization algorithm, in particular the aspects regarding the weights parameters we provided students a survey which is divided into five sections: student profile, current situation, optimal exams session, required commitment of exams, exam planning.

In Student profile section it is asked the field of study to divide the responses study course,

CONTINUARE

## 2.4 - Domain Assumptions

* A solo scopo sperimentale, stiamo considerando il miglioramento e la conseguente ottimizzazione del calendario previsionale per la sede di Leonardo, per la scuola di 3I, che si tratta della più numerosa. In futuro potrà essere esteso anche alle altre scuole.
* At the moment only the calendar of Ingegneria dell’Automazione is taken into consideration;
* During the winter and summer exam sessions there are 2 calls per exams, in autumn it is one;
* Abbiamo considerato di avere un numero di appelli uguali a 2 per tutti gli esami. Poiché (purtroppo) la maggioranza delle scuole di Ingegneria ha questo limite sul numero di appelli di ogni esame. Il programma comunque può essere utilizzato anche per corsi di studio con numero di appelli superiori a 2
* Contiamo sempre di avere un problema feasible per il nostro algoritmo di ottimizzazione
* L’assegnazione delle aule è fatta in un secondo momento, al momento della schedulazione degli esami consideriamo un vincolo sul numero massimo di studenti contemporaneamente presenti al poli per gli esami. E’ probabile che poi non sia fattibile la gestione delle aule ma questa non è una cosa che possiamo sapere a priori in quanto l’iscrizione all’appello per lo studente chiude pochi giorni prima della data dell’esame (questa scadenza è temporalmente successiva alla pubblicazione del calendario quindi in caso di problemi verranno fatte modifiche a mano più avanti secondo le necessità come di fatto viene fatto tutt’ora dalla segreteria studenti).

# 3 - USE CASES AND SCENARIOS

## 3.1 - Use cases

- The students want to know when their exams are scheduled so they go to PoliMi website and search the link with the exam session calendar.

- Professors want to know when their exams are scheduled so they go to PoliMi website and search the link with the exam session calendar.

- Professor of course ‘*X*’ has to be abroad for a Researcher Meeting, so there will be a period in which he can’t do the exam for his courses

- One of the teaching assistant has a second job in a school and would like to express his unavailability for the mornings

## 3.2 - Scenarios

## 3.3 - Sequence diagram

# 4 - LIST OF REQUIREMENTS

## 4.1 - External Interface Requirements

The following section will give a more detailed description, in terms of hardware,

software and communication interfaces.

## 4.1.2 Hardware Interfaces

The hardware needed to use the software of this project is a pc with monitor (to visualise graphical user interface), keyboard and mouse to insert input data and a printer (not strictly necessary) to print the calendar.

## 4.1.3 Software Interfaces

The front-end (GUI) must communicate with back-end, this last one must communicate with database and should launch the optimization algorithm.

## 4.1.4 Communication Interfaces

## 4.3 Performance Requirements

The system does not have specific requirements on performances such as

response time. The optimization algorithm can take a long time to be solved but there aren't any problems.

## 4.4 Hardware Requirements

# 5 - DOMAIN MODEL

## 5.1 - Class diagram

## 5.2 - System scheme

# 6 – OPTIMIZATION ALGORITHM LOGICS

3.6 Software System Attributes

3.6.1 Reliability

The only aspect is that the optimization algorithm can find a solution.

3.6.2 Availability

The system should be ready to be used only during the exam schedulation period, but it is not an hard constraint because perchè se anche il calendario viene fatto una settimana dopo non è un problema

3.6.3 Security

The system doesn’t need any particular kind of security. It is accessible only to registrar’s office. A login page is enough, but not necessary.

3.6.4 Maintainability and scalability

These are important aspects of the system because it must be scalable in case of a new course, new rules regarding the exams scheduling (for example 3 calls instead of 2) or course deleting. It is also a good maintainability, the System should be organised in modules and be well documented, in order to make maintenance, upgrades and integration of new features easy.

3.6.5 Portability

The software must run on Windows.