SDGinUJI

Final technical report

Authors:

Erik Artigas Reverter

Laura Llorens Angulo

Silvia García Gómez

Group: LA1ArtigasGomezLlorens

Version: Final

Date: 22/12/2022

Index

1. Introduction

This document explains different aspects of the prototype of the SDGinUJI project made by the students Erik Artigas, Silvia Garcia and Laura Llorens in the subject EI1023 Fundamentals of Software Engineering.

The objective of the document is to define some aspects of the project that will be developed in the subject EI1027 Design and Implementation of Information Systems. The aspects discussed in this document are the project context, the preliminary planning, the requirements, the analysis and the design.

2. Project context

2.1. Context

The Universitat Jaume I (UJI) wants to have a record of all the initiatives regarding the Sustainable Development Goals[1] currently being carried out, or likely to be launched in the future, by the members of the university community. These Sustainable Development Goals are an action by all countries to promote prosperity while protecting the planet.

The proposal for the register of initiatives, is promoted by the administrative staff responsible for the Oficina de Cooperació al Desenvolupament i Solidaritat (OCDS-office) at UJI[2].

The software product to be developed will be a web application, which will be mainly for users of the OCDS office and for other members of the university community. With this product it will be possible to manage all the information on the initiatives related to any of the SDGs defined by the United Nations[3]. This product will allow registering, managing and obtaining information on the different SDG initiatives.

All the initiatives ODS must be classified based on [4], as: Learning and teaching, Governance, Research, and External leadership. Moreover each of the initiatives must deal with one or many actions, defined regarding to the targets of the SDGs.

SDGinUJI, it is important to provide a public area where the initiatives and their results can be disseminated as a part of the SDGs policies of the university.

2.2. Goals and scope of the product

We will define the objectives of our product, SDGinUJI, and define its scope in different areas.

First, we will analyze what are the objectives of our product. To do this, we will analyze which are all its benefits. The first benefit of our product is that it makes it easier to share the different SDG initiatives with other people. The second benefit of our product is that it makes it easier to manage initiatives (consult them, deny them...). The last benefit is that it increases the participation of the members of the university community in the SDG initiatives and that increases the possibility that the members of the university community have new ideas.

Finally, we will define some scopes of our product. The first scope we will study is the functional scope. In summary, the functional scope of our product is to store and manage information about all the initiatives that are being done or that will be done in the future. The second scope is the organizational scope. Our organizational scope includes the OCDS-office staff, the members and non-members of the university community and, as developers, the 'Servei d'Informàtica de l'UJI'. The last scope is the technological scope. Respect to this scope, we will need: a web page able to be runned in all kinds of devices (mobile phones, PCs, tablet) and a good authentication system. This authentication system will be done easily due to all the authentication tools of the 'Servei d'Informàtica de l'UJI'.

3. Preliminary planning

In this section, we will define the project context and set up preliminary planning. It includes the definition of the project scope and objectives, the identification and allocation of resources, establishment of restrictions and risks, and the proposal of a preliminary planning of the project.

3.1. Objective and scope of the project

The objective of the project is to develop the product explained in the previous section. Consequently, the goals of the projects are all the functionalities that our products must achieve, described in the context section.

Finally, we will define the scope of our product.The first scope we will study is the functional scope. In summary, the functional scope of our product is to store and manage information about all the initiatives that are being done or that will be done in the future. The second scope is the organizational scope. Our organizational scope involves the OCDS-office staff and the members and non-members of the university community. The last scope is the technological scope. Respect to this scope, we will need a good authentication system. This authentication system will be done easily due to all the authentication tools of the 'Servei d'Informàtica de l'UJI'.

3.2. Risks and restrictions identification

On the one hand, we will analyze the risks of our project. In the team, we find a risk of not complying with the aforementioned time restriction because there is a high probability that we will not have enough time to develop the project. Finally, we also find the risk at the user level that the product may not be used as much as we want due to the lack of interest in its.

On the other hand we will see the restrictions of the project. The main restrictions we have are: the temporary restriction associated with the subject ending at any given time, though by having all the facilities that the UJI offers us, we will not find great economic, spatial and technological restrictions.

3.3. Temporal estimation

We will divide the tasks to do in 5 supertasks that have to be done one after the other one. That is, a task that contains inside other tasks (subtasks). The 5 supertasks are the following:

- Beginning of the project: It is divided into 3 subtasks and its purpose is to have an initial idea of the product that we want to develop.
    1. Definition of project context: Analyze the necessities of our potential clients (UJI members and non-members) in order to define what our product must bring them.
    2. Prototype: Think and transform our initial idea into a minimalist real product
    3. Goals and scope of the product: Define what our product must bring to the potential clients
- Preliminary Planning: It is divided into 4 subtasks and its purpose is to create and organize a project whose unique purpose will be developing the product according to the specification explained in the previous supertask.
    1. Definition project team: Optimize responsibilities dividing tasks according to a reduced set of possible roles.
    2. Risks and restrictions: Evaluate the events that could make our project fail and the probability of them.
    3. Estimation: Evaluate and calculate the amount of time required in order to develop our product. This part is also important because time is also a cost that must be considered.
    4. Final Schedule: Define as exactly as possible what each member of our team will do and which deadline we will have. This schedule is not a rigid rule but an aproximation.
- Requirements: It is divided into 5 subtasks and its purpose is to specify the requirements of our product. That is, specify what our product must do and, even more importanly, what our product must not do.
    1. Use case: Define exactly the functionalities that our product must bring to the client attending to the “Goals and scope of the product” task. We will make, for example, interviews with the clients in order to have a precise idea of what we want to achieve.
    2. UCD: Describe the relation between the UCs and the clients of the product (that include UJI members, OCDS-Staff…).
    3. Requirements: List and explain all the elements (data, technologies…) that must be available for implementing the UCs succesfully.
    4. UCD and goals and scope
    5. Other requirements
- Analysis: It is divided into 2 subtasks and its purpose is to establish a global vision of how the different elements of our product (classes) will interact and which relations will be between them.
    1. Class diagram: We will describe the relations between all our classes (actors, hardware…)
    2. CD/UCD verification and validation: We will verify that our previous work (CD/UCD) is consistent with the scope and objectives (previously defined).
- Design: It is divided into 2 subtasks and its purpose is to specify as much as possible how the product will be developed. That is, specify all the general concepts or ideas of

the previous subtasks. That way, we will be able to implement the product, which is the objective of our project.

1. Class diagram: We will define all the details that have not been added to the class diagram of the Analysis supertask. For example, attributes, specific methods…
2. Graphical user interface (GUI) design: Decide how the different actors will see our product (the interface of our product). This will be decided according to the type of actor (novel actor, expert actor…).

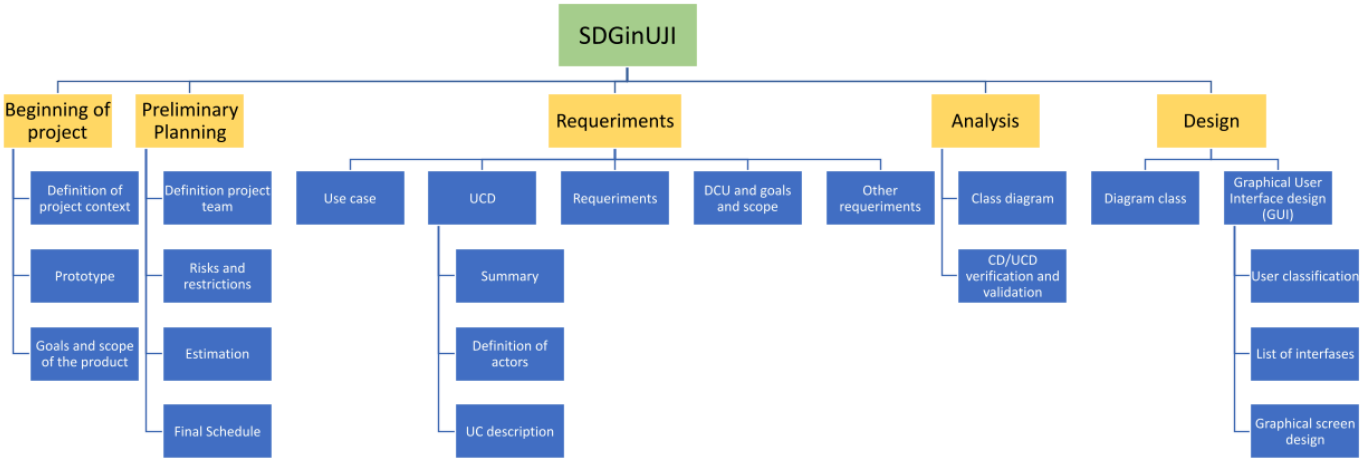


Figure 1: WBS

3.4. Resources allocation

In this section we will focus on human resources allocation. That is, we will define the most important roles between the members of our group.

The first one is the Project Manager [5]. We have decided to assign the role of Project manager to Silvia Garcia Gomez because we believe that she has the necessary skills to organize and plan project tasks because she has work experience.

The second one is the Quality Assurance Manager/Quality Manager [6]. We have assigned this role to *Laura Llorens Angulo* because she is a very organized and responsible person so she will be able to organize the team effectively.

The third one is the System Analyst [7], we have decided to assign this role to *Erik Artigas Reverter* because he is skilled at problem solving.

The last one is responsible for submitting the report to the "Virtual Classroom". We have decided that, as we have very different schedules, we will assign this role to a different member of the team each week.

3.5. Final schedule and Gantt diagram

In this section we will make a time schedule that will result in a calendar with the estimated activities, times and resources for the project. To represent all this we will make a Gantt diagram.

Based on the WBS and the deadlines we estimate the time needed for each task. Once the time is estimated we assign resources to the task depending on the workload involved.

For more detailed information about the schedule the subject has a time schedule which this group has followed rigorously.

	Modo de	Nombre de tarea	Duración	Comienzo	Fin	Predecesoras	Nombres de los recursos
1		•Beginning of project	**14 días**	**lun 19/09/22**	**jue 06/10/22**		
2		•Definition of project context	4 días	lun 19/09/22	jue 22/09/22		Silvia
3		•Prototype	4 días	vie 23/09/22	mié 28/09/22	2	Erik;Laura
4		•Goals and scope of the product	14 días	lun 19/09/22	jue 06/10/22		Erik;Laura;Silvia
5		•Preliminary Planning	**10 días**	**vie 07/10/22**	**jue 20/10/22**		
6		•Definition project team	5 días	vie 07/10/22	jue 13/10/22		Laura;Erik;Silvia
7		•Risks and restrictions	5 días	vie 07/10/22	jue 13/10/22		Erik;Laura;Silvia
8		•Estimation	2 días	vie 14/10/22	lun 17/10/22		Silvia;Laura
9		•Final Schedule	3 días	mar 18/10/22	jue 20/10/22	8	Erik
10		•Requeriments	**14 días**	vie 21/10/22	mié 09/11/22		
11		•Use case	1 día	vie 21/10/22	vie 21/10/22		Laura;Erik;Silvia
12		•UCD	**14 días**	vie 21/10/22	mié 09/11/22		
13		•Summary	1 día	vie 21/10/22	vie 21/10/22		Laura;Erik;Silvia
14		•Definition of actors	1 día	vie 21/10/22	vie 21/10/22		Laura;Erik;Silvia
15		•UC description	1 día	vie 28/10/22	vie 28/10/22		Laura;Erik;Silvia
16		•Requeriments	1 día	vie 04/11/22	vie 04/11/22	12	Laura;Erik;Silvia
17		•DCU and goals and scope	1 día	sáb 05/11/22	sáb 05/11/22	12	Laura;Erik;Silvia
18		•Other requeriments	4 días	dom 06/11/22	mié 09/11/22	12	Laura;Erik;Silvia
19		•Analysis	**14 días**	**vie 11/11/22**	**mié 30/11/22**		
20		•Class diagram	2 días	vie 11/11/22	sáb 12/11/22		Laura;Erik;Silvia
21		•CD/UCD verification and v	9 días	lun 14/11/22	jue 24/11/22	20	Laura;Erik;Silvia
22		•Design	**10 días**	**vie 02/12/22**	**mié 14/12/22**		
23		•Diagram class	2 días	vie 02/12/22	dom 04/12/22		Laura;Erik;Silvia
24		•Graphical User Interface design (GUI)	**10 días**	**vie 02/12/22**	**mié 14/12/22**		
25		•User classification	2 días	vie 02/12/22	dom 04/12/22	23	Laura;Erik;Silvia
26		•List of interfases	2 días	vie 09/12/22	sáb 10/12/22	23	Laura;Erik;Silvia
27		•Graphical screen desigr	2 días	dom 11/12/22	lun 12/12/22	26	Laura;Erik;Silvia
28		•Entrega final	5 días	**vie 16/12/22**	**jue 22/12/22**		Laura;Erik;Silvia

Figure 2: Gantt diagram

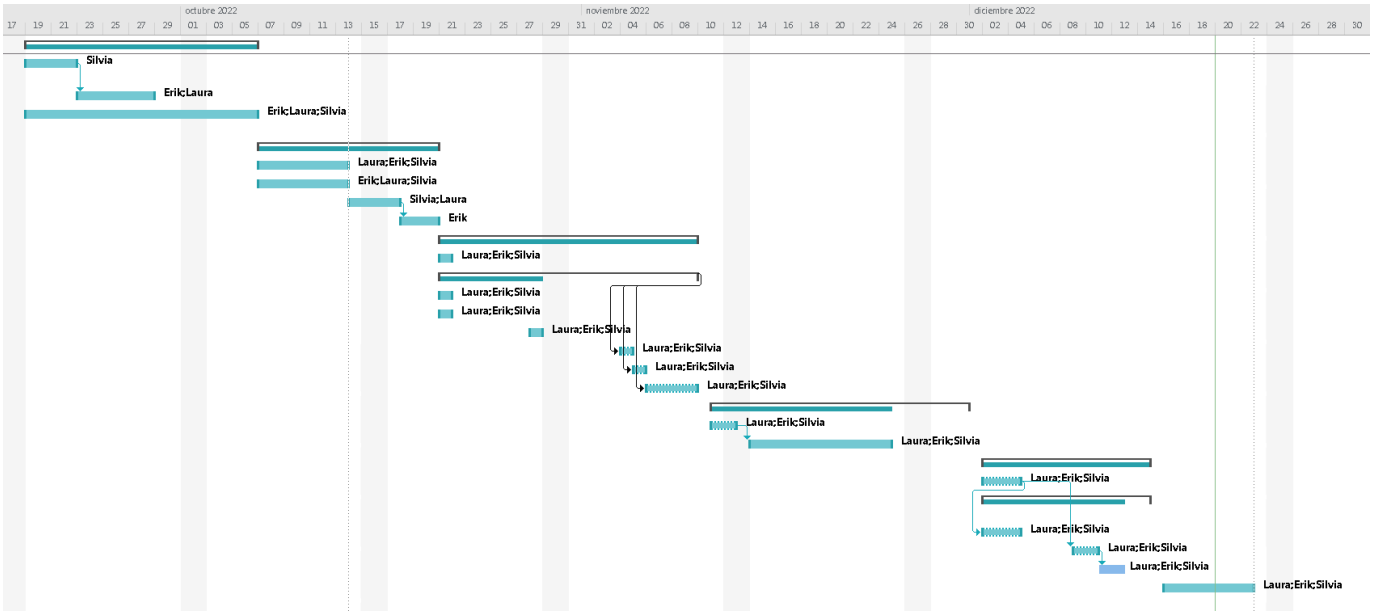


Figure 3: Sequential Gantt Diagram

4. Requirements definition

In this section we are going to talk about the requirements definition. The objective of the requirements definition is to identify the functionality and the information that the software processes to cover the needs of the users.

The definition of requirements is developed in three fundamental stages, on the one hand the research, in this stage, techniques are used to search for information where meetings will be held and the ideas that will allow determining the system requirements will be collected.

On the other hand, analysis and documentation, where facts are analyzed and documented. Finally, the requirements are validated and checked.
At this stage, the results obtained will be displayed and the viability will be validated and reviewed to verify that the requirements are correctly defined.

In the first place we will show the use case diagram made with MagicDraw.

In the second place we will show a summary of the use cases and the actors that affect each use case. In addition, we will also see a short description of the actors involved in the use case diagram.

In the third place we will show a list of templates with each use case.

Fourthly we have the list of requirements and the validations of the requirements.

Finally we will explain other requirements that may affect the use case and the details and their templates.

4.1. Use case diagram

The use case diagram is one of the most representative UML diagrams.[9] that is used to model the behavior of the system. The behavior model represents how the system is going to behave at all times, and there may be Use Case diagrams and other UML diagrams.

The elements of the use case diagram are the actors, use cases and relationships. The use case diagram is a visual representation of the actors and cases with definitions and complementary specifications.

The actors represent a role that can be assumed by a person, a group of people or an external device of the system. These actors are external to the system and have to interact with the system directly.

The use cases represent a sequence of actions and describe what the behavior of the system will be under conditions that will be defined in the system. In other words, it shows what the programmer has to do from the point of view of the users and represents the different situations depending on the conditions of the interaction of the different actors, independently of the technology to be used.

Below we show the use case diagram that we have defined in our project:

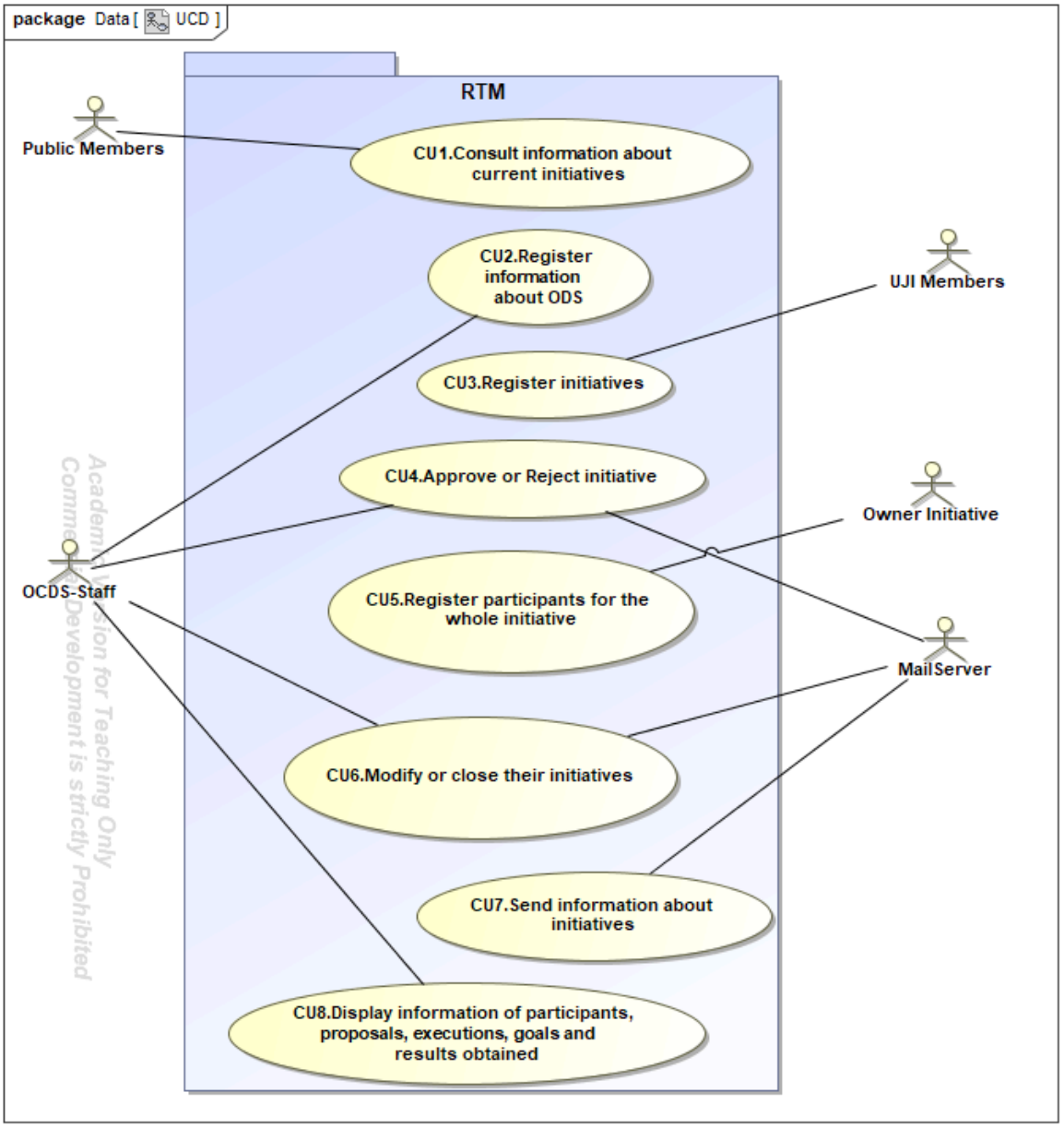


Figure 4: Use case diagram

4.2. Documentació del UCD

4.2.1. Summary of use cases and actors

In this section we show a table where the main actors related to each use case are.

Primary Actor	Use Cases
MailServer	CU3.Register initiatives CU4.Approve or Reject initiative CU6.Modify or close their initiatives CU7.Send information about initiatives
OCDS-Staff	CU2.Register information about ODS CU4.Approve or Reject initiative CU6.Modify or close their initiatives CU8.Display information of participants, proposals, executions, goals and results obtained
Initiative Owner	CU5.Register participants for the whole initiative
Public Members	CU1.Consult information about current initiatives CU3.Register initiatives
UJI Members	CU3.Register initiatives

Table X:

4.2.2. Actors description

The main actors of our system are initiative Owner, OCDS-Staff, public Members and Mail Server.

The initiative owner actor refers to any member of the uji that can create an initiative. This actor will have the function of interacting with the system to create initiatives.

El actor principal OCDS-Staff is a group of people in charge of the office dedicated to the objectives of sustainable development (ODS).

The actor public members, hace referencia a las personas de la comunidad de la universidad Jaime I y las personas que están fuera de comunidad que están interesadas en las iniciativas propuestas por los usuarios de la uji.

The actor Mail server es un dispositivo fisico externo al sistema. Server in charge of sending the answers to the different requests of the clients. For example, if a person wants to join an initiative, they will send that request to the server. The server will then respond by accepting or denying the initiative.

4.2.3. UC description (templates)

Use Case Name	Author	Review
CU1. Consult information about current initiatives	Erik	Laura
CU2.Register information about ODS	Silvia	Erik
CU3.Register initiatives	Laura	Silvia
CU4.Approve or Reject initiative	Erik	Silvia
CU5.Register participants for the whole initiative	Silvia	Laura
CU6.Modify or close their initiatives	Laura	Erik
CU7.Send information about initiatives	Silvia	Laura
CU8.Display information of participants, proposals, executions, goals and results obtained	Laura	Erik

Table 1: Use case Name

Use Case Name	**Consult information about current initiatives**	**ID**	CU1
Complexity	Average Complexity		
Description	It shows a list of the current initiatives to the public members. The user can see detailed information about every initiative of the list. The user can search for some initiatives.		
Actors	● Public Members		
Goal	All the members will be able to see the information about current initiatives.		
Assumption	The user selects Consult initiatives		
Scenarios			
Basic Flow of Events	**Steps normal sequence:** 1. The system shows the searcher with the filters. 2. The user selects some fields from the filter. 3. The system shows a list of the initiatives with the requirements. 4. The user selects any initiative. 5. The system shows the details of the initiative.		
Exceptional Flow of Events	The user goes back without viewing any initiative.		

Use Case Name	**Register information about ODS**	ID	CU2
Complexity	Average Complexity		
Description	It shows a list of the ODS to the OCDS-Staff and allows them to register information about ODS.		
Actors	● OCDS-Staff		
Goal	The OCDS-Staff will be able to register information about some ODS		
Assumption	The OCDS-Staff selects Register information about ODS.		
Scenarios			
Basic Flow of Events	**Steps normal sequence:** 1. The system shows a list of the ODS. 2. The OCDS-Staff selects an ODS. 3. The system shows all the details of the ODS selected.		

	4. The OCDS-Staff makes some changes. 5. The system asks to confirm the changes. 6. The OCDS-Staff confirm the changes.
Alternative Flow of Events	The OCDS-Staff doesn't confirm the changes.
Exceptional Flow of Events	The OCDS-Staff goes back without confirming the changes.

Use Case Name	**Register initiatives**	**ID**	CU3
Complexity	Average Complexity		
Description	The system will allow UJI members to record information on new initiatives. Once the applications are registered, they will have to be accepted or rejected by the OCDS-Staff		
Actors	● MailServer ● Public Members ● UJI Members		
Goal	Uji members will be able to register new initiatives		
Assumption	Uji members register new initiatives		
Scenarios			
Basic Flow of Events	**Steps normal sequence:** 1. The system asks for the title and the ODS related to the new initiative 2. The system checks that there is no initiative with the same title and ODS. 3. If it does not exist, the system asks for the necessary information to complete the initiative. 4. The system checks all data required to complete the initiative and asks for confirmation of the initiative. 5. If the user confirms the creation of the new initiative, the system registers the new initiative in the system.		
Alternative Flow of Events	Dismiss the new initiative.		
Exceptional Flow of Events	1. If there is an initiative with the same title and ODS, information about the existing initiative is shown. 2. If you have not completed all the necessary fields to fill in the initiative, the system will ask you to add the required fields.		

Use Case Name	**Approve or Reject initiative**	**ID**	CU4
Complexity	Average Complexity		
Description	The OCDS-Staff will ask the system to see all the initiatives pending to respond. Once the initiatives are displayed, one will be selected and the initiative will be accepted or rejected. The owner of the initiative will then receive an email with the resolution.		
Actors	● MailServer ● OCDS-Staff		
Goal	OCDS-Staff will be able to see the initiatives created and that are pending to respond. To do this, it will accept or reject the initiative and the MailServer will send an email to the person responsible for the initiative		
Assumption	The OCDS-Staff select Approve initiatives		
Scenarios			

Basic Flow of Events	**Steps normal sequence:** 1. The system shows a list of initiatives pending approval. 2. The user selects one initiative of the list. 3. The system shows the details of the selected initiative. 4. OCDS-Staff accept the initiative. 5. The system asks if you are sure to accept the initiative 6. The system registers the information. 7. The system sends an email to the person who has registered the initiative and to the people who are subscribed to the related SDGs.
Alternative Flow of Events	If the OCDS-Staff do not confirm the accept of the initiative, the OCDS-Staff returns to the list of initiatives
Exceptional Flow of Events	

Scenarios	
Basic Flow of Events	**Steps normal sequence:** 1. The system shows a list of initiatives pending approval. 2. The user selects one initiative of the list. 3. The system shows the details of the selected initiative. 4. OCDS-Staff reject the initiative. 5. The system asks if you are sure to reject the initiative 6. The system registers the information. 7. The system sends the order to send an email to the Initiative Owner.
Alternative Flow of Events	If you do not confirm the reject of the initiative, you return to the list of initiatives

Use Case Name	**Register participants for the whole initiative**	**ID**	CU5
Complexity	Average Complexity		
Description	The system will show all the accepted initiatives that members have created. The system will allow to add new participants to the initiative.		
Actors	● Initiative Owner		
Goal	Being able to add participants to an approved initiative.		
Assumption	The system will allow initiative owner to add participants to the approved initiative.		

Scenarios	
Basic Flow of Events	**Steps normal sequence:** 1. The member of the UJI will access the list of initiatives created by him and that are approved. 2. The system will show a list and a filter of initiatives. 3. The user will select an initiative. 4. The user will add participants to the initiative. 5. The system will verify that the participants added to the initiative exist. 6. The system will display a confirmation message of the added participants. 7. Initiative owner will confirm that the added participants are correct.
Alternative Flow of Events	Back to the list of initiatives
Exceptional Flow of Events	The participant does not have a valid email and cannot be added to the initiative.

Use Case Name	**Modify or close their initiatives**	**ID**	CU6
Complexity	Average Complexity		
Description	The system shows a list of aspects of the initiative that can be changed and an option that allows to close the initiative		
Actors	● MailServer		

	● OCDS-Staff
Goal	Allowing the creator of an initiative to change different aspects such as detalines if it is needed or even delete it
Assumption	The system will allow initiative owner to modify or close their initiatives

Scenarios	
Basic Flow of Events	**Steps normal sequence:** 1. The system shows a list of aspects of the initiative that can be changed. 2. The user selects one of the aspects and change its value. 3. The system asks if you are sure to change that aspects 4. The system registers the changes. 5. The system sends the order to send an email to the users who are registered to the changed initiative.
Basic Flow of Events Diagrams	**Steps normal sequence:** 1. The system shows a list of the initiatives approved that can be closed. 2. Closed initiative 3. The system asks if you are sure to close the initiative. 4. The system sends the order to send an email to the persons who are registered to the removed initiative.
Alternative Flow of Events	Back to the list of initiatives
Exceptional Flow of Events	The user doesn't confirm the changes or the removal of an initiative

Use Case Name	**Send information about initiatives**	**ID**	CU7
Complexity	Average Complexity		
Description	Every day the system will check the list of uji members who have subscribed to notifications of new initiatives related to an ODS topic.		
Actors	● MailServer		
Goal	You want to send a message with the information requested by the members of the uji on certain initiatives on the requested ODS issues.		
Assumption	The mail server will send a message with the information requested by the uji members.		

Scenarios	
Basic Flow of Events	**Steps normal sequence:** 1. The system will access the list of uji members who have subscribed to an ODS issue. 2. The system will search for initiatives with the ODS topics related to the subscription of each uji member. 3. For each initiative, an email will be sent with the desired information.
Alternative Flow of Events	There are no approved initiatives with the requested topic. Then no mail will be sent.

Use Case Name	**Display information of participants, proposals, executions, goals and results obtained**	**ID**	CU8
Complexity	Average Complexity		
Description	The system saves the data of all the initiatives, even if their are closed, and allows OCDS-Staff to access relevant information.		
Actors	● OCDS-Staff		
Goal	Allowing OCDS-Staff to track easily the initiatives that are being developed . That way initiatives can be changed (upgraded) or closed based on the performed tracking.		
Assumption	The OCDS-Suff introduce a collection of data from which the system filters the initiatives to be displayed to the OCDS-Stuff		

Scenarios	
Basic Flow of Events	**Steps normal sequence:** 1. The user fill all the possible data. 2. The user confirms the information given. 3. The system displays the information filtered based on the data received from the user
Alternative Flow of Events	1. The user doesn't fill in any of the data. 2. The user confirms. 3. The system displays information of all the initiatives without filters.
Alternative Flow of Events Diagrams	Back to the initial menu
Alternative Flow of Events Diagrams	1. The user introduces data. 2. The user confirms. 3. The system doesn't find any initiative that matches with the information received. 4. The system notifications the user about that.

4.3. Data requirements

In this section we are going to define data requirements. This section includes a list of all the data requirements and detailed data requirement templates.

List of all the data requirements

Data Requeriments	Author	Review
DR01. SGDs	Laura	Silvia
DR02. Initiatives	Silvia	Laura

Table 2: Data requirements

Detailed DR templates

Requirement	**DR01. SGDs**		
Author	Laura Llorens Angulo	**Created**	07/11/2022
Reviewer	Silvia Garcia Gomez	**Approved**	09/11/2022
Origen/ Source	OCDS-Staff	**Version:**	v1
		Data revised	09/11/2022
Data	SDG's data: Name, Targets, Details.		
Detail	Details provide some details of the social, environmental and economic challenges covered.		
Documents	List of the 17 SDG defined by the UN. [1]		

Requirement	**DR02. Initiatives**		
Author	Silvia Garcia Gomez	**Created**	07/11/2022
Reviewer	Laura Llorens Angulo	**Approved**	09/11/2022
Origen/ Source	Public Members	**Version:**	v1

		Data revised	09/11/2022
Data	Initiatives: title, motivation, ODS considered regarding the initiative, textual description, url that can be accessed to look for more information about the initiative, short and long term actions, which target and indicators are connected to the actions, initial and expected final date for the initiative as well as for each of the actions included, and expected results (textual) regarding each action		
Detail	All the initiatives will be shown, although only the initiatives that are already accepted can be selected. On the other hand, the initiatives that are pending or rejected can be viewed but not selected for viewing.		

4.4. Requirements V&V (DCU and goals and scope)

In this section we will describe the way we have validated and verified our requisite in an efficient and rigorous way.

First of all, we will have a set of different requisites, already validated and verified (this set can be empty). Then, we see which functionality has not been attended to yet. After that, we create a requisite for that functionality. This way, we ensure this requisite is rigorously needed in order to achieve our objectives. Then we see if that requisite clashes with the previous ones. Finally, we check that the requisite is not out of our scope. We do this until all functionalities have been considered.

Parallelly, (for each requisite) we will verify them one by one. That is, we will focus on the possibility to simplify or documentate that requisite better (abstraction level, complexity level…).

4.5. Other requirements

In this section we are going to define other requirements. This section includes a list of all the other requirements and detailed other requirements templates.

List of all the other requirements

Other Requirements	Author	Review
OR01. The system must be able to generate reports with information of participants, proposals, executions, goals and results obtained.	Laura	Erik
OR02. The system must store information about the initiatives	Silvia	Erik
OR03. The system must be able to publish a initiative	Erik	Silvia
OR04. The system must be able to send an email when a initiative is approved/rejected	Erik	Laura

Table 3: Other Requirements

Detailed OR templates

Requirement	**OR01.** The system must be able to generate reports with information of participants, proposals, executions, goals and results obtained.		
Author	*Laura Llorens Angulo*	**Created**	07/11/2022
Reviewer	*Erik Artigas Reverter*	**Approved**	09/11/2022
Origen/ Source	OCDS-Staff	**Version:**	v1
		Data revised	09/11/2022
Description	The system must generate reports with all the information of participants, proposals, executions, goals and results obtained.		

Requirement	**OR02. The system must store information about the initiatives**		
Author	Silvia Garcia Gomez	**Created**	07/11/2022
Reviewer	*Erik Artigas Reverter*	**Approved**	09/11/2022
Origen/ Source	The system and OCDS-Staff	**Version:**	v1
		Data revised	09/11/2022
Description	The system must store all the information given by the user about the initiative. The minimal required information is explained in DR02		
Notes/	All required fields of the initiative must contain a valid value.		

Requirement	**OR03. The system must be able to publish a initiative**		
Author	*Erik Artigas Reverter*	**Created**	07/11/2022
Reviewer	Silvia Garcia Gomez	**Approved**	09/11/2022
Origen/ Source	*OCDS-Staff*	**Version:**	v1
		Data revised	09/11/2022
Description	When an initiative is created and accepted, the system must publish it, that is, the system must make the new initiative accessible to the public members when they try to consult accepted initiatives.		
Notes	The system could also send an email to the public members that are “potential viewers”.		

Requirement	**OR04. The system must be able to send an email when a initiative is approved/rejected**		
Author	*Erik Artigas Reverter*	**Created**	07/11/2022
Reviewer	*Laura Llorens Angulo*	**Approved**	09/11/2022
Origen/ Source	OCDS-Stuff	**Version:**	v1
		Data revised	09/11/2022
Description	The system must be able to obtain the emails of the users that are subscribed to the initiative that has been recently approved or rejected and send a short description about what has been changed (acceptance or rejection) and a link in order to allow the user to easily obtain more detailed information.		
Documents			
Notes/	The emails are supplied by IGLU.		

5. Analysis

The main objective of system analysis is to transform the definition of user requirements into a software specification, in addition, the initially defined objectives, the restrictions and the particular needs of the system must be taken into account.

To perform and document the analysis correctly it is necessary to use methods and techniques that allow representing the software and the rest of the components of a computerized system.

To define this section we will divide the analysis into two parts, the class diagram and its documentation.

The class diagram serves both to show the structure of the system information, as well as to show its behavior through the operations of the classes. In addition, it is also used to show the main elements that the system will contain and the representation of the functionality represented in the use cases through operations.

5.1. Class diagram

We will make a class diagram to represent some aspects of the system. The aspects to represent are the structure of the system information and the system behavior. The system behavior can be seen through the operations.

The class diagram is composed of classes, associations and operations.
For our project, we used the following diagram:

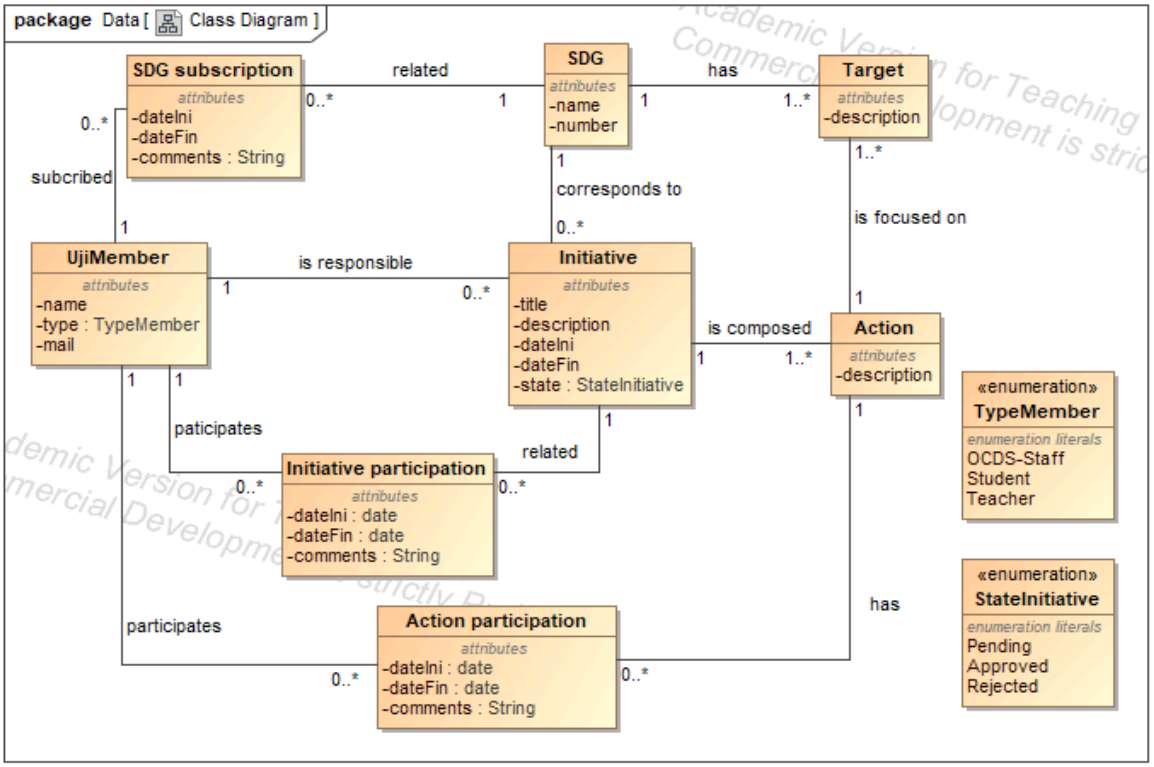


Figure 5: Class diagram

In figure X we show the most important main classes UjiMember, SDG subscription, SDG, Initiative, Initiative participation, Action participation, Target and Action. Moreover, we have also extracted two classes that are enumerations to be able to name the types of UjiMember and the status of Initiative.

On the other hand, we also show the relationships between each class with its multiplicities.

5.2. Class diagram documentation

5.2.1. Classes

In this section we show the tables related to each class and the description of their attributes and some examples for each case.

Class	**SDG**		
Author	*Silvia Garcia Gomez*	**Date of creation**	**18 / 11 / 2022**
Revisor	*Laura Llorens Angulo*	**Date of approval**	18 / 11 / 2022
Source	Initiative Owner	**Version**	v1
		Date of review	18 / 11 / 2022
Description	The SDG class [1] represents the objectives and goals of sustainable development, in these it will show the information of each one of the objectives classified by name and objective number.		

Attribute details	Name, number.
Object example	Goal 1: End poverty in all its forms everywhere
Validation	
Comments	

Class	**Target**		
Author	*Laura Llorens Angulo*	**Date of creation**	**18 / 11 / 2022**
Revisor	*Erik Artigas Reverter*	**Date of approval**	18 / 11 / 2022
Source	OCDS-Staff	**Version**	v1
		Date of review	18 / 11 / 2022
Description	The class Target represents the targets of an SDG.		
Attribute details	Description.		
Object example	Target 1.1[8]: By 2030, eradicate extreme poverty for all people everywhere, currently measured as people living on less than $1.25 a day.		
Validation			
Comments			

Class	**Initiative**		
Author	Laura, Silvia and Erik	**Date of creation**	**18 / 11 / 2022**
Revisor	Laura, Silvia and Erik	**Date of approval**	18 / 11 / 2022
Source	Product owner	**Version**	v1
		Date of review	18 / 11 / 2022
Description	The class Initiative represents all proposed initiatives. An initiative is a set of actions related to an SDG.		
Attribute details	Title, description, start date, end date, state. The attribute state is of the type StateInitiative which is an enumerated attribute that can be pending, approved or rejected.		
Object example	Title: Donate objects that we do not use Description: There are more and more millions of people in the world living in extreme poverty. In developed countries we tend to accumulate a lot of things we don't use, such as clothes, books, toys, etc. We could start sorting out everything we don't use and take it to an NGO where it can go to people who need it more than we do. Start date: 30/11/2022 State: Pending		
Validation			
Comments			

Class	**Action**		
Author	*Erik Artigas Reverter*	**Date of creation**	**18 / 11 / 2022**
Revisor	*Silvia Garcia Gomez*	**Date of approval**	18 / 11 / 2022
Source	Initiative Owner	**Version**	v1
		Date of review	18 / 11 / 2022
Description	This class represents all the actions in which an initiative is composed, that is, the set of ordered actions that must be done in order to accomplish the created initiative.		
Attribute details	Description		
Object example	*Buy solar panels*		
Validation			
Comments			

Class	UjiMember		
Author	Laura, Silvia and Erik	Date of creation	18 / 11 / 2022
Revisor	Laura, Silvia and Erik	Date of approval	18 / 11 / 2022
Source	UJI Member	Version	v1
		Date of review	18 / 11 / 2022
Description	The class UjiMember represents all the persons who have a Uji account.		
Attribute details	Name, Type, Mail. The attribute Type is of the type TypeMember which is an enumerated attribute that can be OCDS-Staff, Student, Teacher.		
Object example	Elena Perez Perez is a UjiMember of type Student with the mail al123456@uji.es.		
Validation			
Comments			

Class	SDG subscription		
Author	*Silvia Garcia Gomez*	Date of creation	25 / 11 / 2022
Revisor	*Erik Artigas Reverter*	Date of approval	25 / 11 / 2022
Source	Uji member	Version	v1
		Date of review	25 / 11 / 2022
Description	The SDG subscription class represents the relationship between the uji member identifier and the SDG identifier as well as the comment related to the two tables.		
Attribute details	name UjiMember, name SDG, start date, end date and comments.		
Object example	The uji member silvia wants to receive notifications of the SDG end poverty.		
Validation			

Class	Initiative participation		
Author	Laura Llorens Angulo	Date of creation	25 / 11 / 2022
Revisor	Erik Artigas Reverter	Date of approval	25 / 11 / 2022
Source	UjiMember	Version	v1
		Date of review	25 / 11 / 2022
Description	The Initiative participation class represents the relationship between the Ujimembers and the initiatives.		
Attribute details	Start date, end date and comments.		
Object example	The UjiMember Elena Perez Perez participate in the initiative Donate objects that we do not use		
Validation			
Comments			

Class	Action participation		
Author	*Erik Artigas Reverter*	Date of creation	25 / 11 / 2022
Revisor	*Laura Llorens Angulo*	Date of approval	25 / 11 / 2022
Source	Initiative Owner	Version	v1
		Date of review	25 / 11 / 2022
Description	This class represents the relationship between an action and the UjiMemeber participating in this action.		

Attribute details	DateIni, dateFin, comments
Object example	*The UjiMemeber Elena Perez Perez participates in the action: Buy solar panels*
Validation	
Comments	

5.2.2. Associations

As for associations, next we are going to show the tables related to the associations, where you can see the multiplicity between each class according to the use cases of our system.

Association	**UjiMember is subscribed SDG subscription**		
Author	*Silvia Garcia Gomez*	**Date of creation**	25/11/2022
Revisor	*Erik Artigas Reverter*	**Date of approval**	30/11/2022
Source	Uji Member	**Version**	v1
		Date of review	30/11/2022
Description	This association represents the connection between an UJi Member and the SDG because an UjiMember can ask to be subscribed to one or more SDG in order to receive information.		
Multiplicity	The multiplicity of UjiMember is 1 and the multiplicity of SDG subscription is 0..*		
Validation	It is necessary to have this association in the model because it is necessary to be able to connect the UjiMembers with the ODS through a subscription since the ujiMembers can request information from the different ODS.		

Association	**SDG Subscription related SDG**		
Author	*Silvia Garcia Gomez*	**Date of creation**	25/11/2022
Revisor	*Erik Artigas Reverter*	**Date of approval**	30/11/2022
Source	Uji Member	**Version**	v1
		Date of review	30/11/2022
Description	This association represents that each SDG subscription is related to a specific SDG. This association represents the connection between an UJi Member and the SDG because an UjiMember can ask to be subscribed to one or more ODS in order to receive information.		
Multiplicity	Subscription ODS has multiplicity 0..* and ODS its multiplicity is 1		

Association	**UjiMember is responsible Initiative**		
Author	*Silvia Garcia Gomez*	**Date of creation**	25/11/2022
Revisor	Laura Llorens Angulo	**Date of approval**	30/11/2022
Source	Initiative Owner	**Version**	v1
		Date of review	30/11/2022
Description	This association represents that an uji member will be responsible for one or more initiatives. The creators of the initiative can only be a member of the uji.		
Multiplicity	The Uji member has multiplicity 1 and the Initiative has multiplicity one to many.		

Association	**UjiMember participe Initiative participation**		
Author	*Silvia Garcia Gomez*	**Date of creation**	25/11/2022
Revisor	Laura Llorens Angulo	**Date of approval**	30/11/2022

Source	Uji member	**Version**	v1
		Date of review	30/11/2022
Description	This association means that one Ujimember can participate in many initiatives and many Ujimembers can participate in one initiative. This association will have a start and end date and comments that will refer to the initiative.		
Multiplicity	The UjiMember has multiplicity one to many and the initiative participation has multiplicity one to many.		

Association	**Initiative Participation related Initiative**		
Author	Laura Llorens Angulo	**Date of creation**	25/11/2022
Revisor	Erik Artigas Reverter	**Date of approval**	30/11/2022
Source	UjiMember	**Version**	v1
		Date of review	30/11/2022
Description	This association represents that each Initiative participation is related to a specific initiative and each initiative will be related to zero, one or many participation.		
Multiplicity	The Initiative has multiplicity 1 and the Initiative participation has multiplicity zero to many.		

Association	**UjiMember is subscribed participates Action participation**		
Author	Laura Llorens Angulo	**Date of creation**	25/11/2022
Revisor	Erik Artigas Reverter	**Date of approval**	30/11/2022
Source	UjiMember	**Version**	v1
		Date of review	30/11/2022
Description	This association represents that each action participation involves one Uji Member and each Uji member participates in zero, one or many action participations.		
Multiplicity	The UjiMember has multiplicity 1 and the Action participation has multiplicity zero to many.		

Association	**Action Participation has action**		
Author	Laura Llorens Angulo	**Date of creation**	25/11/2022
Revisor	Silvia Garcia Gomez	**Date of approval**	30/11/2022
Source	UjiMember	**Version**	v1
		Date of review	30/11/2022
Description	This association represents that each action participation has a specific action and each action has zero, one or many participations.		
Multiplicity	The Action has multiplicity 1 and the action participation has multiplicity zero to many.		

Association	**Initiative is composed Action**

Author	Laura Llorens Angulo	**Date of creation**	25/11/2022
Revisor	Silvia Garcia Gomez	**Date of approval**	30/11/2022
Source	Initiative owner	**Version**	v1
		Date of review	30/11/2022
Description	This association represents the connection between an initiative and an action because an initiative is composed of one or more actions.		
Multiplicity	The Initiative has multiplicity 1 and the action has multiplicity one to many.		

Association	**SDG has Target**		
Author	*Erik Artigas Reverter*	**Date of creation**	25/11/2022
Revisor	Laura Llorens Angulo	**Date of approval**	30/11/2022
Source	Initiative Owner	**Version**	v1
		Date of review	30/11/2022
Description	This association represents that each of the SDG has defined a set of targets previously that can contribute to the SDG and that a target is related to exactly one SDG.		
Multiplicity	The SDG has multiplicity 1 and the Target has multiplicity one or many		

Association	**Target is focused on Action**		
Author	*Erik Artigas Reverter*	**Date of creation**	25/11/2022
Revisor	Laura Llorens Angulo	**Date of approval**	30/11/2022
Source	Initiative Owner	**Version**	v1
		Date of review	30/11/2022
Description	This association represents that each target will be achieve in the real world by doing a determined action and that each action can contribute to the achieving of many targets.		
Multiplicity	The Target has multiplicity one or many and the Action has multiplicity one.		

Association	**Initiative corresponds to SDG**		
Author	*Erik Artigas Reverter*	**Date of creation**	25/11/2022
Revisor	Silvia Garcia Gomez	**Date of approval**	30/11/2022
Source	Initiative Owner	**Version**	v1
		Date of review	30/11/2022
Description	This association represents that each initiative must be created in order to achieve only one SDG and that each SDG must be achieved by a single initiative.		
Multiplicity	The Initiative and the SDG has multiplicity one.		

5.3. CD/UCD verification and validation

In this section, we will follow the technique already used in 4.4. This way we will simultaneously verify and validate each UC one by one. That is, before declare a UC as definitive we will verify if it is strictly needed in order to achieve some requisite (already verified and validated). Then, we will only have to check if the addition of this new UC or relation (actor–UC) suggests the change of one of the already validated and verified UCs.

The only difference between this section and 4.4 is that, as in this section we can ensure all our requirements are already verified and validated, we will only have to focus on what that requirements implicitly requires at an analysis level instead of having to interpret the brief information given in the section “Goals and scope of the product”.

6. Design

The software design is the last specification of the software that has been logically documented in the analysis. Based on the design, we will be able to implement the product and, that way, achieve our objectives.

During this design phase, alternatives should be identified and evaluated, and the one that best ensures the implementation of a quality product must be selected, considering all the components that interact in the development of a software product.

In order to develop the design part, we have already previously developed the system requirements and the analysis. So in this section we are going to make a design class diagram first to check that the entire system we have proposed is correct.

Finally, we design the Graphical User Interface (GUI) with a free app designed *ad hoc*. For that part, we analyze each of the actors that will use each of our interfaces in order to bring them interfaces adapted to their specific caracteristics.

6.1. Design class diagram

The class diagram that we have made during the analysis has evolved incorporating specific design details that had not been previously specified.

In this part we have added the data types that each attribute contains, the default values in each class and the specific operations that each use case has with the type of parameters that we are going to use in each case and the type that we are going to return in each operation.

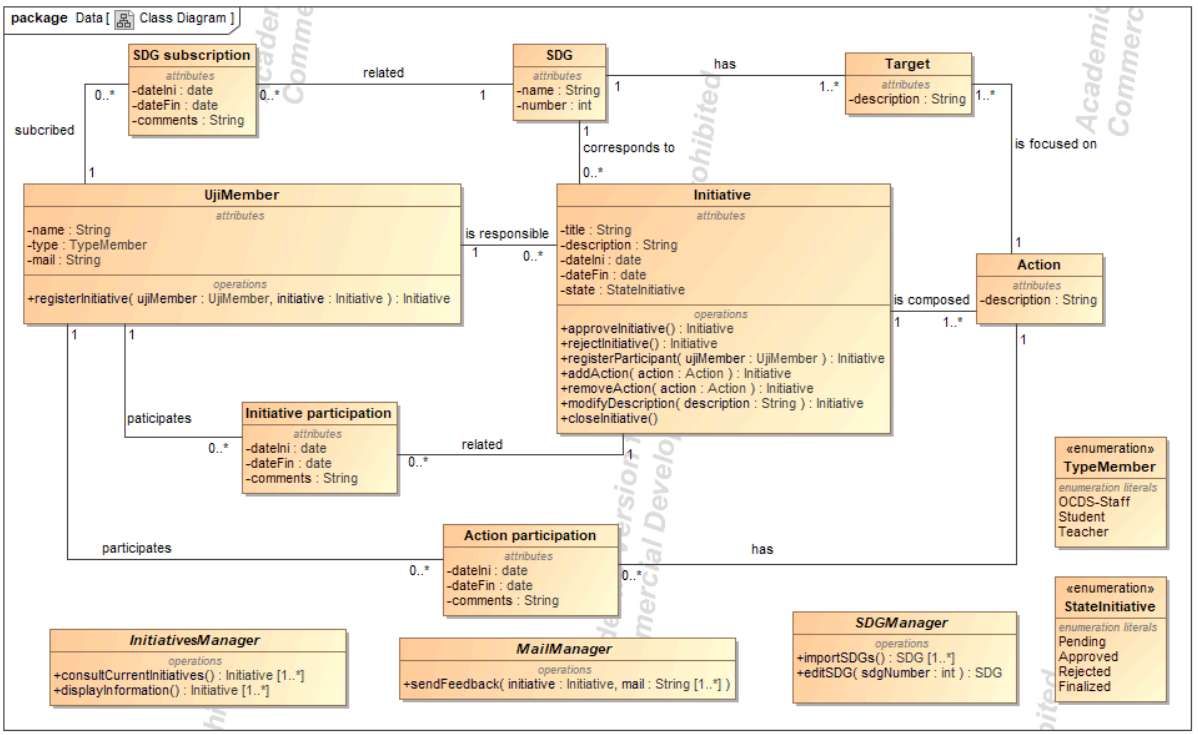


Figure 6: Design class diagram

6.2. Graphical User Interface design (GUI)

The graphical user interface is the mechanism for creating a dialogue between the user and the system. It is important to take into account the human factor in order to establish fluent communication and to obtain an appropriate interface.

To make a good graphical user interface you have to take into account the skill and knowledge of the users. Therefore, before making the graphical user interface, it is necessary to classify the users who are going to use it.

In the section User classification we will look at each type of user and their level of knowledge. In the section List interface we will make a table with all the interfaces, the user that will use each one, and the interface type.

Finally, we will use the list of interfaces as a basis for some examples of graphical screen design. We will take into account the user classification to make an appropriate interface.

6.2.1. User classification

The classification of the users that will interact with the system are public member UJI member, initiative owner and OCDS-Staff.

Public Member and UJI Member: Novice and eventual. Public members will be novel users because everybody is a Public Member so we can not ensure a high acknowledgment about SDGs.

Initiative Owner: Expert and frequent. The initiative owner is an expert because (commonly) being able to register correct initiative requires a medium-high acknowledgment about SDGs. As we can not ensure that he will register more initiative, he will be a frequent user.

OCDS-Staff: OCDS-Staff work in the field of OCDs so they are expert about them and, as they have a regular work journey, they will be frequent users.

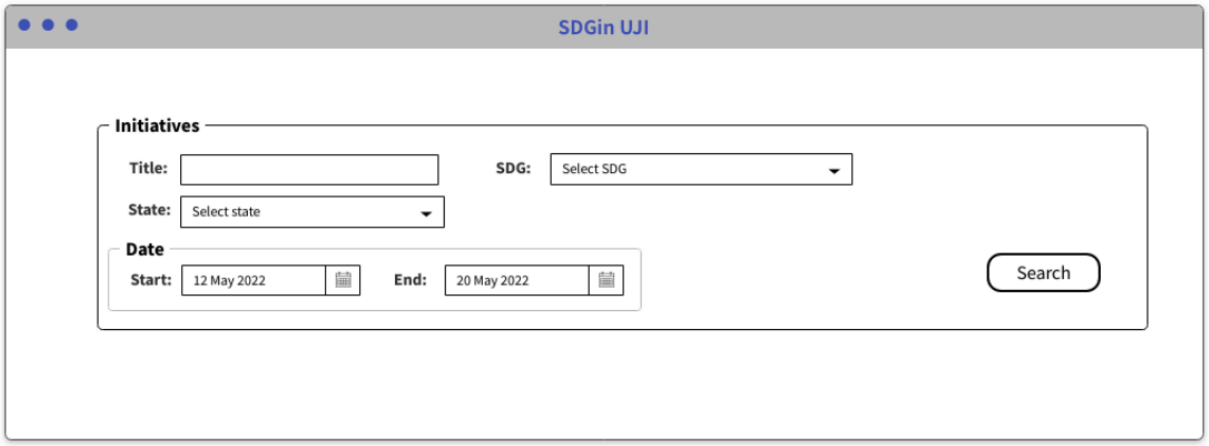
6.2.2. List of interfaces

User	User Profile	GUI Name	Interface Type
Public Member	Novice and eventual	S01 Search initiative	Detail input
Public Member	Novice and eventual	S02 Display a list of initiatives	Detail output
Public Member	Novice and eventual	S03 Asking for confirmation for changes	Dialog
Public Member	Novice and eventual	S04 Search SDG	Detail input
Public Member	Novice and eventual	S05 Display List of SDGs	Detail output
Public Member	Novice and eventual	S06 Display details of SDG	Detail output and input
UJI Member	Novice and eventual	S07 Display details of the initiative	Detail output
UJI Member	Novice and eventual	S08 Register initiative	Detail input
UJI Member	Novice and eventual	S09 Asking for confirmation for initiative creation	Dialog
Initiative Owner	Expert and frequent	S10 Display details and modify initiative	Detail output and input
OCDS-Staff	Expert and frequent	S11 Display details of a pending initiative	Detail output and detail input
OCDS-Staff	Expert and frequent	S12 Display details and modify initiative	Detail output and input
OCDS-Staff	Expert and frequent	S13 Display details and modify SDG	Detail input
OCDS-Staff	Expert and frequent	S14 Search data system	Detail input

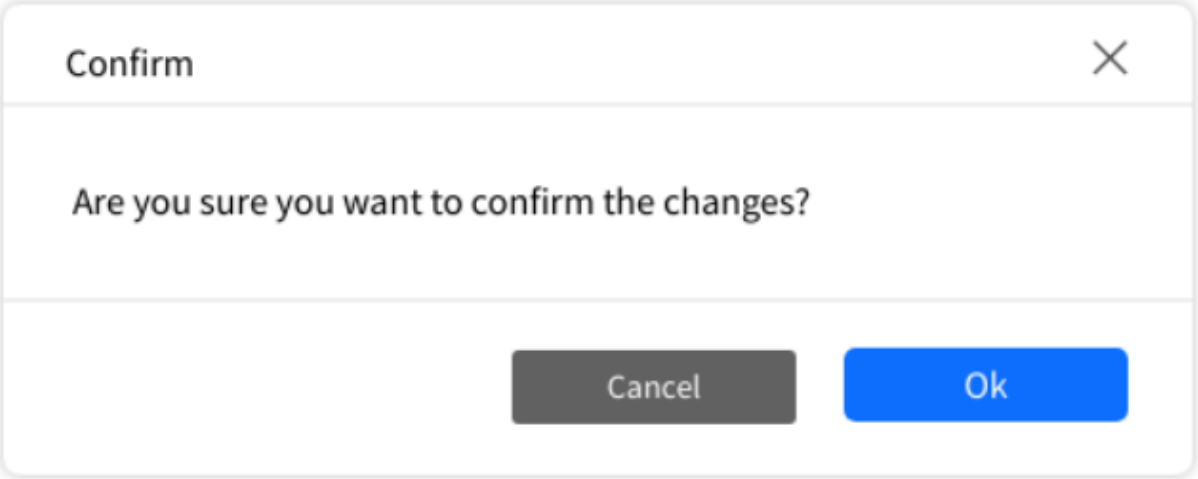
OCDS-Staff	Expert and frequent	S15 Display the inform about the data system	Detail output

Table 4: List of interfaces

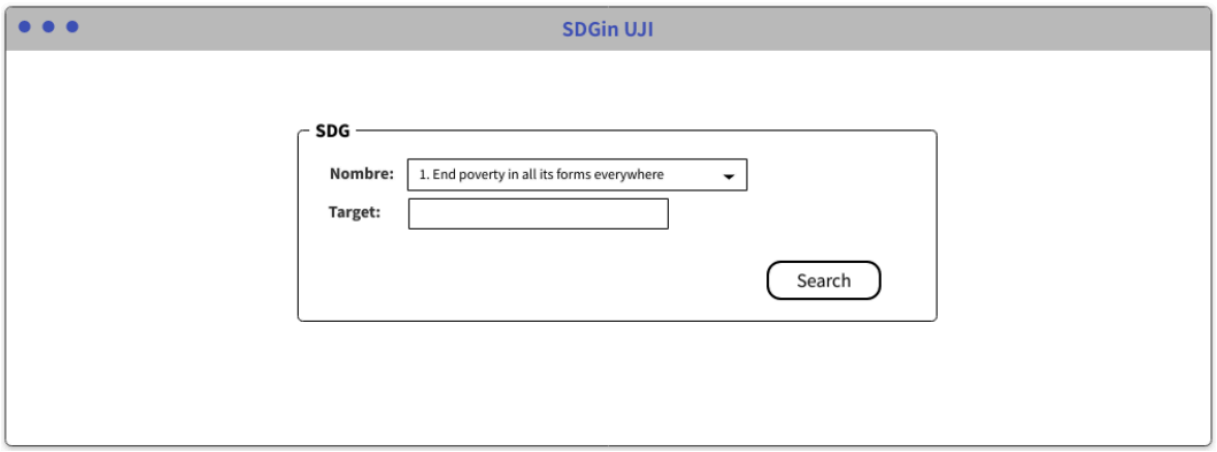
6.2.3. Graphical screen design examples



S01. Search initiative

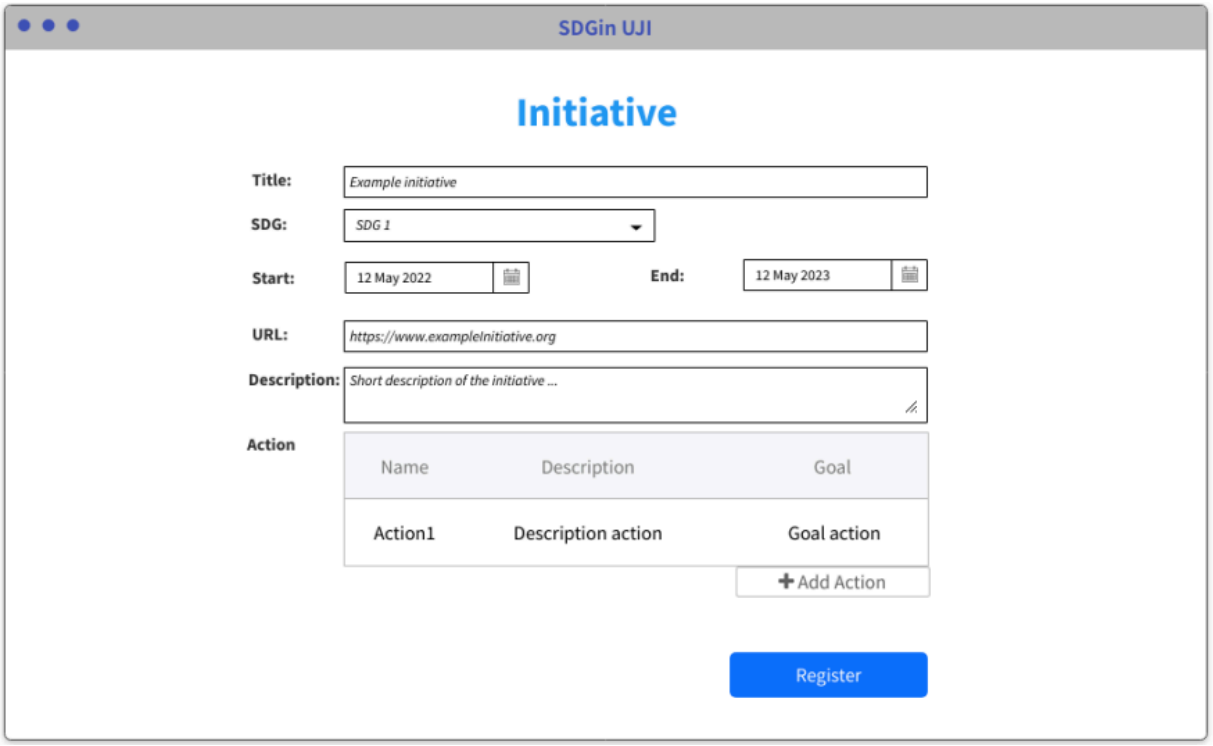


S03. Asking for confirmation for changes

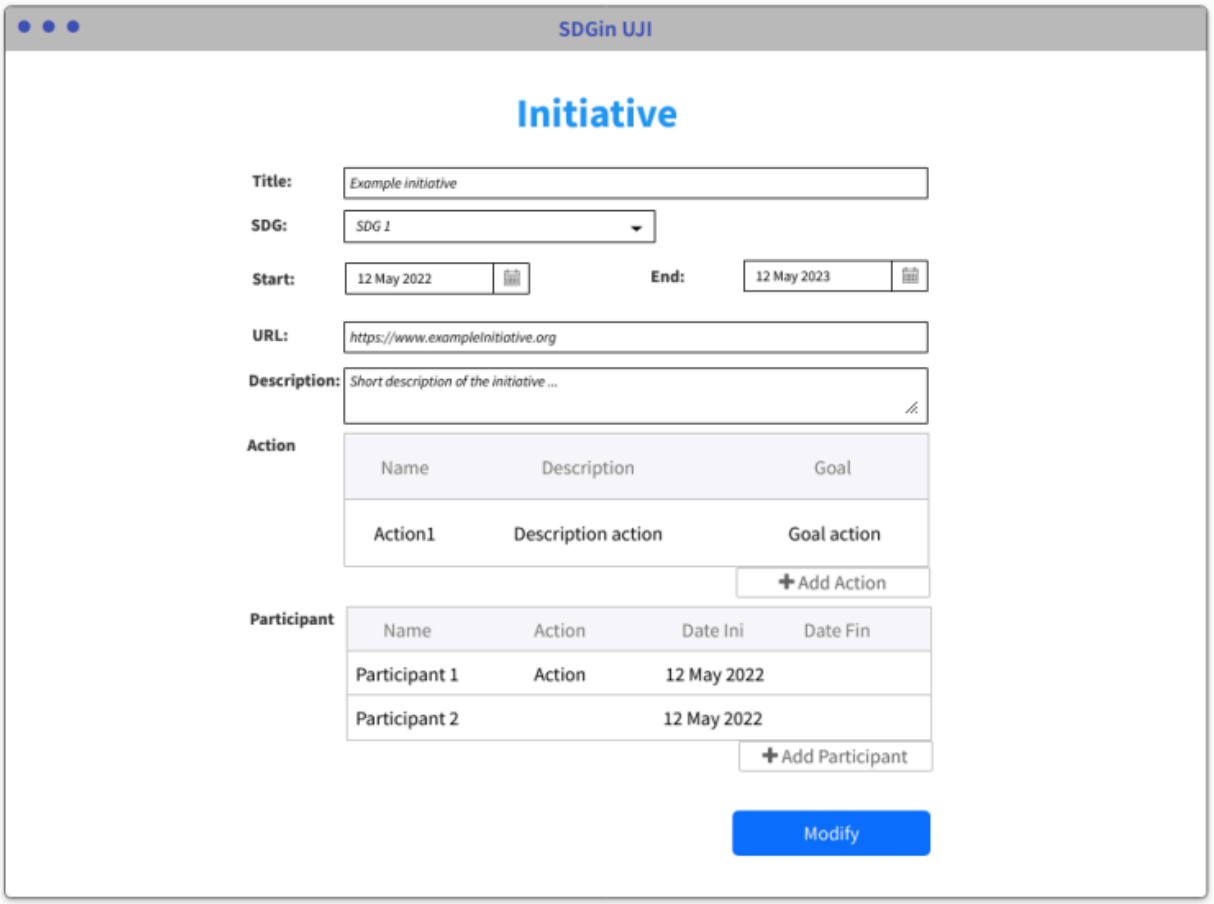


S04. Search SDG

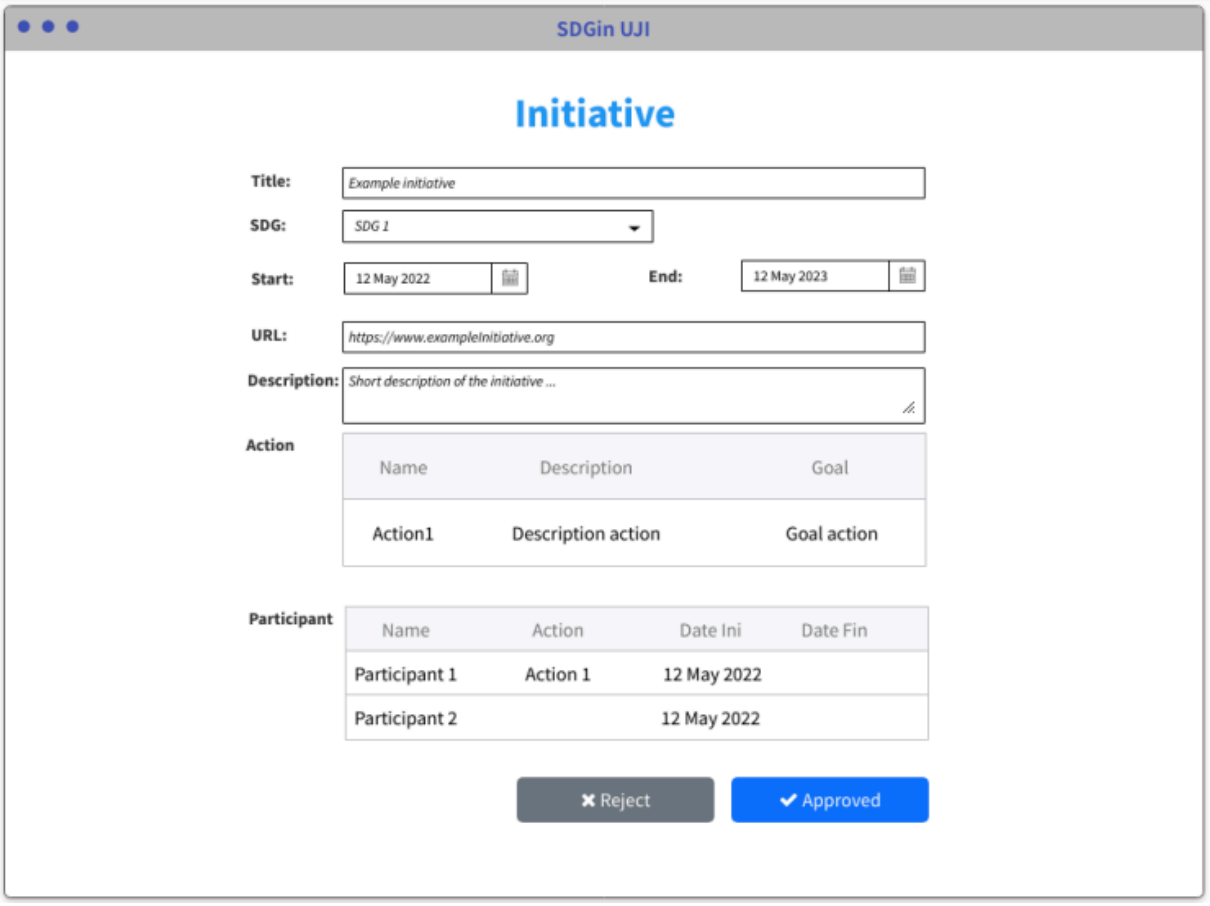
S06. Display details of SDG



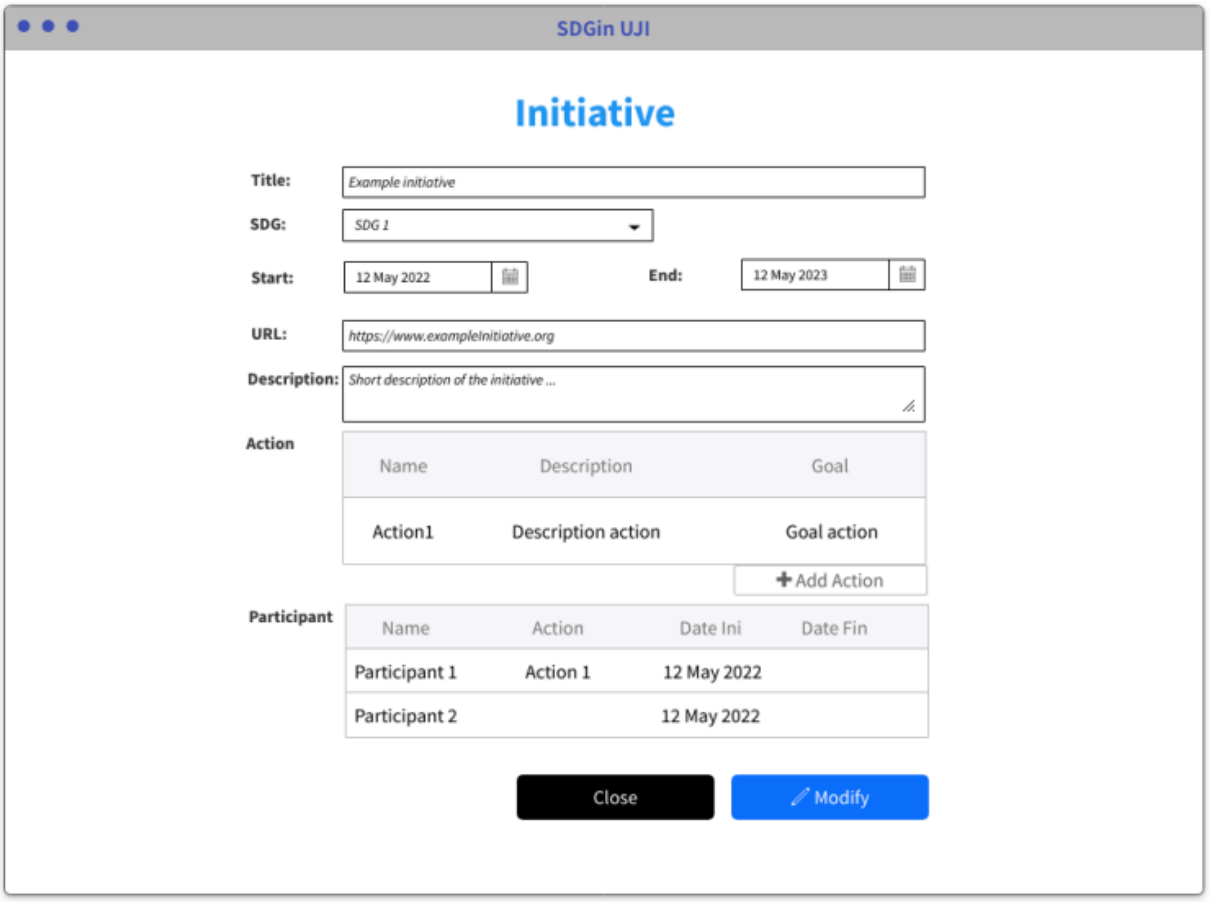
S08. Register initiative



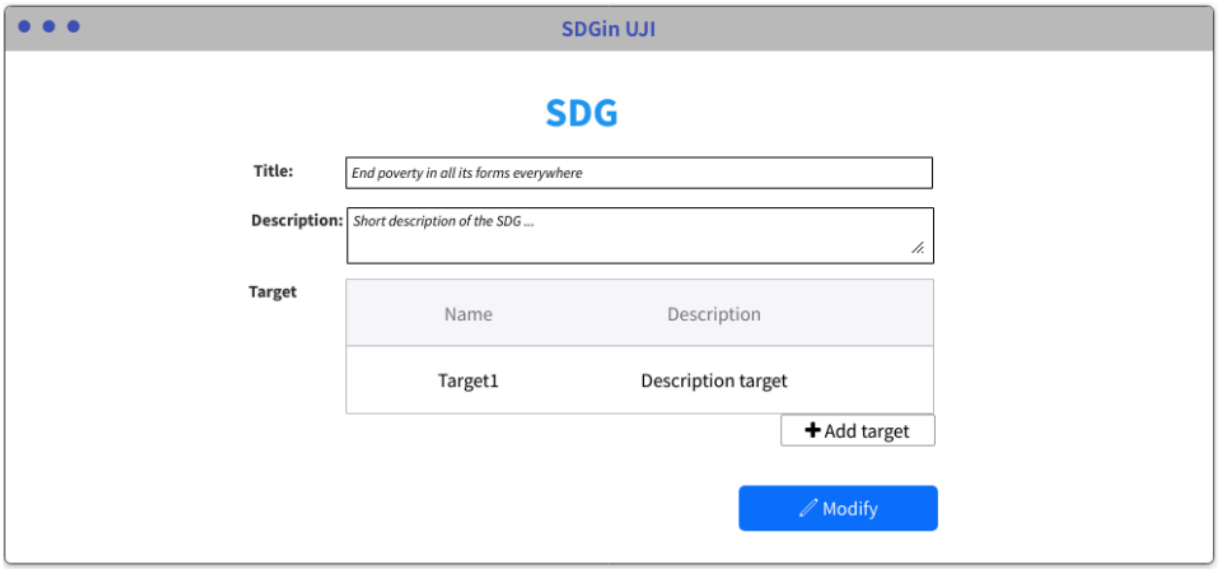
S10. Display details and modify initiative



S11. Display details of a pending initiative



S12. Display details and modify initiative



S13. Display details and modify SDG

7. Conclusions

As a result of what has been explained previously, we have developed a software product part by part and following a specific methodology. The next phase, that will be done in a different subject, will be the implementation of the software product for the clients. And that phase will be impossible to realize successfully without all the work already done because we and only we are the people incharge of converting the initial idea of the product into a potentially useful product. That is, the work done will be crucial for the people in charge of the last phase (implementation).

8. Bibliography/References

1. United Nations. Sustainable Development Goals. Accessed 06/07/2022. https://www.un.org/sustainabledevelopment/
2. Universitat Jaume I. Oficina de Cooperació al Desenvolupament i Solidaritat (OCDS) Accessed 06/07/2022. https://www.uji.es/serveis/ocds/
3. United Nations Statistics Division (UNSD). Global indicator framework for the Sustainable Development Goals and targets of the 2030 Agenda for Sustainable Development. Accessed 06/09/2022. https://unstats.un.org/sdgs/indicators/indicators-list/
4. The Sustainable Development Solutions Network (SDSN) – Australia/Pacific. Sustainability (ACTS) and the global SDSN Secretariat. *Getting Started with The SDGs in Universities*. Accessed 06/09/2022. https://ap-unsdsn.org/wp-content/uploads/University-SDG-Guide_web.pdf
5. Qué es un Project Manager, qué funciones tiene y cómo ser uno. Accessed 07/10/2022 https://www.iebschool.com/blog/project-manager-agile-scrum/
6. Quality Assurance Manager. Accessed 07/10/2022 https://www.shrm.org/resourcesandtools/tools-and-samples/job-descriptions/pages/quality-assurance-manager.aspx
7. Systems analyst. Accessed 07/10/2022 https://en.wikipedia.org/wiki/Systems_analyst
8. United Nations. Sustainable Development Goals. 1 Goal, End poverty in all its forms everywhere. Accessed 30/11/2022. https://sdgs.un.org/goals/goal1
9. Unified Modeling Language https://en.wikipedia.org/wiki/Unified_Modeling_Language. Accessed 20/12/2022