FRANKFURT UNIVERSITY OF APPLIED SCIENCES

Software Engineering Analysis Project:

Online Translation Service Platform

Ahmed Agdmoun, 1347063

Houda ElAbbassi, 1360545

Mohamed Amine Belrhazi, 1343876

Fatima-Zahra ElAzraq, 1344273

Hicham Boulaayad, 1310812

Mohamed Yazid Rebbali, 1312674

Prof. Dr. Peter Thoma

Supervisor: Alexander Mützel

FRANKFURT UNIVERSITY OF APPLIED SCIENCES

Declaration of Authorship

We hereby certify that the project report we are submitting is entirely our own original work except where otherwise indicated. We are aware of the University's regulations concerning plagiarism, including those regulations concerning disciplinary actions that may result from plagiarism. Any use of the works of any other author, in any form, is properly acknowledged at their point of use.

Table of Contents

1. Introduction

Translation *is an* important *part of* our *lives whether we realize it or not. In this era of globalization where mostly everyone wants to explore something new each moment, it becomes impossible to actually avoid how important translation is.* And despite the high number of *free online machine translation services, the human factor still plays a very important role as well. We all do realize that translation is not just transferring words from a language to another but it's a complicated and a very crucial way of explaining things in a way that its actual meaning is delivered without harming its origin.*

Moreover, online translation is also still considered as one of the top domains for start-ups and many native speakers are searching for jobs as freelancers, hence we came up with the idea of creating a new online translation services platform to connect potential customers with freelancers from all over the world.

Especially in these difficult times where everyone should stay most of the times at home and where a lot of people have lost their ordinary jobs in different sectors, our platform will give a lot of these people the opportunity to earn some money from their homes just with a laptop and something they started to learn even before going to school, their native language!

Of course, our platform is not only for native speakers but also for everyone who has a good knowledge in any language, and wants to make use of it.

Our platform is open for professional translators, language teachers, students and any kind of potential freelancers, since the capability of translating depends on the needs of the customers and it can varies from translating a simple sentence to translating a whole scientific work report.

And since we are students and most of us are foreign students who can speak at least four different languages, we are highly confident that our platform will be a very useful place for all students who are looking for part-time jobs, especially foreign and exchange students. This gives us more motivation to make our idea come true, even if we won't use it ourselves but we are sure it will be helpful for a lot of other students around the globe.

2.Requirement Overview

2.1 Backlog Items/ Requirements

ID	Name	Main Actor	Priority	Description	Categorization	Worked by
0	Privacy policy & Terms of use	The company, Customer, Contractor	high	The company must provide a full and clear agreement on how the data will be handeled and used. This agreement can be accepted or refused from the user before registration.	Non functional	Fatima-Zahra (100%)
1	Customer Registration	Customer	high	the customer needs to be able to register when using the system for The first time	Functional	Yazid (100%)
2	Contractor registration	Contractor	high	the contractor needs to be able to register when using the system for the first time. the contractor must additionally answer some questions concerning his knowledge and experiences.	Functional	Amine (100%)
3	User Verification	Contractor & Customer	high	the user must be verified by confirming the phone number.	Non functional	Houda (100%)

4	User login	Contractor & Customer	high	the user must be able to login into the system.	Functional	Ahmed (100%)
5	Account management	Contractor & Customer	medium	the user needs to be able to modify his profile.	Functional	Hicham (100%)
6	contractors search filter	Customer	high	the filter helps the customer to find an appropriate translator based on his needs.	Functional	Amine (100%)
7	New translation request	Customer	high	a registered customer must be able to place a new translation request.	Functional	Hicham (100%)
8	Translation request confirmation	contractor	high	the contractor has the ability to confirm or refuse any open requests	Functional	Yazid (100%)
9	tracking a translation order	Customer	medium	the customer needs to be able to track his translation request and Be notified every time the status of the request has changed.	Functional	Amine (100%)
10	Translation document delivery	Customer & Contractor	high	The customer must receive the translated document in the agreed deadlines.	Functional	Houda (100%)
11	Payment methods	Customer	high	the system must provide different payment methods to the customer.	Functional	Fatima-Zahra (100%)
12	Contractor rating	Customer	medium	It is possible for the customer to rate the contractor after receiving The final result.	functional	Ahmed (100%)

13	Customer service	Support team, customer & contractor	high	the customers and the contractors have the possibility to contact The support team whenever they need help.	Functi onal	Fatima-Zahra (100%)
14	User feedback	Contractor & customer	medium	It is possible to collect feedback from the users to improve the service.	Functi onal	Hicham (100%)
15	Project revenue	The company	high	The revenue can be accessed and checked.	Non functio nal	Houda (100%)
16	Contractors network status	The company	low	The company must be able to track and check the status of the contractors network.	Non functio nal	Ahmed (100%)
17	Company growth and profit per customer	The company	high	Profit per customer can be checked in order to have an overall outlook on the growth of the company.	Non functio nal	Yazid (100%)

Table 1Backlog Items

Sprint 1/2/3

Sprint 1/4/5

Sprint 1/6/7

2.2 Story Mapping

	Registration/Login	Browse for Translator	Document Translation	Document delivery
Version 1	Customer Registration	Contractor search filter	new translation request	Translation document delivery
	Contractor Registration		Translation request confirmation	Customer Service
	User Verification		Payment	
	User Login			
Version 2	Account Management		Tracking the translation process	Contractor Rating
	Privacy policy & Terms of Use			User Feeback

Table 2Story Mapping

3. Definition of Done

The Definition of "Done" is an important term of the Scrum Framework. In a scrum team, every member has to agree on when a product backlog item can be considered "done". If this is not the case, misunderstandings and conflicts can arise between the development team and the product owner, especially during the sprint review meeting. And we don't want that.

In our project we found it a bit difficult to agree on when we can exactly say that an item from the backlog is done, especially since we are doing just the analysis part in this project and no implementation is required.

At the end we agreed to define "Done" for every step in our process working on the project.

For example, a use case is done if it fulfils the structure and the process described in the lecture slides, e.g. if it has name, a brief description, actors, preconditions, a basic flow, alternative flows and postconditions..

4. Requirements Specification

Requirements alone are not enough. Before a development can take place, it is necessary to convert the requirements into a functional specification. The task of this specification is to describe exactly how the requirements are to be covered. It deals with the solution design from the customer's point of view.

Requirements can be modeled by two different approaches, and so we decided to divide our requirements into two categories according to these two approaches:

4.1 Flow or Process Oriented Requirements:

This kind of requirements is usually described through Use Cases. And so in the following, all our flow requirements are specified by their Use Case, a Use Case Diagram, an Activity Diagram, a Sequence Diagram and User Interface prototypes.

For the class diagram, we decided to create only one overall class diagram for all requirements, since there are relations between almost all of the requirements and it makes less sense to have a class diagram for every requirement.

But before, let's take a brief look at some definitions:[1]

- Class Diagram

A Class Diagram shows the classes (objects) of a system in a static way. In addition, it also shows the attributes and the methods of each class and the relationships between different classes

[1] (Standford University IT, 2021)

- Use Case:

A use case is a textual description of the interaction between the user and the software system that helps to discover and engineer the requirements.

- Use Case Diagram

A Use Case Diagram describes how the system behaves and models its functionality while interacting with one or more external actors.

- Sequence Diagram

A sequence diagram is also called an event diagram and describes the flow of a set of events through the system and shows the interaction between the objects and the actors in the system in a dynamic way through sending messages and getting answers.

- Activity Diagram[2]

An Activity Diagram shows the flow from one activity to another. It starts from a pre-condition and goes through different activities in the system and ends with a post-condition.

- UI Prototyping

User Interface Prototyping is a technique to create visual prototype of a later application in which users are actively involved to collect feedback and information. User Interface prototypes usually focus only on certain aspects related to requirements and do not model the whole behavior of the application.

[2] (DS CATIA, 2020)

4.1.1Requirement-„Customer Registration “

4.1.1.1 Use Case Specification

Use Case	Customer Registration
Description	The customer must register if he wants to use the system for the first time. He needs to enter his information (e.g. page email and telephone numbers) in order to have contact with the contractor.
Actors	The customer
Pre-conditions	The customer must have a valid e-mail address and password.
Basic Flow **Description** **Actions** **1** **2** **3** **4**	The customer needs to enter his e-mail address and password to a successfully registration. The customer will register first under website. The customer must input his information and make a profile. The customer will upload his documents under his profile. The customer is already done, and he can anytime contact the contractor.
Alternative Flow **Description** **Actions** **1** **2** **3**	A After the registration the customer will receive a confirmation if whether it worked. The customer can edit anytime his information. The customer can add some information. The customer has the ability to his account.
Post-conditions	The registration has been successfully done.

Table1Customer Registration Use Case Specification

4.1.1.2 Use Case Diagram

This use case diagram describes the phase of creating an account of a customer.

This is the phase where the customer can register themselves. Firstly, if the customer wants to register, he should enter some information like password, email address, surname etc. and he should accept the term and condition in order to get a new account. After that if the information were confirmed, he gets a new profile in order that he stays in contact with the contractor.

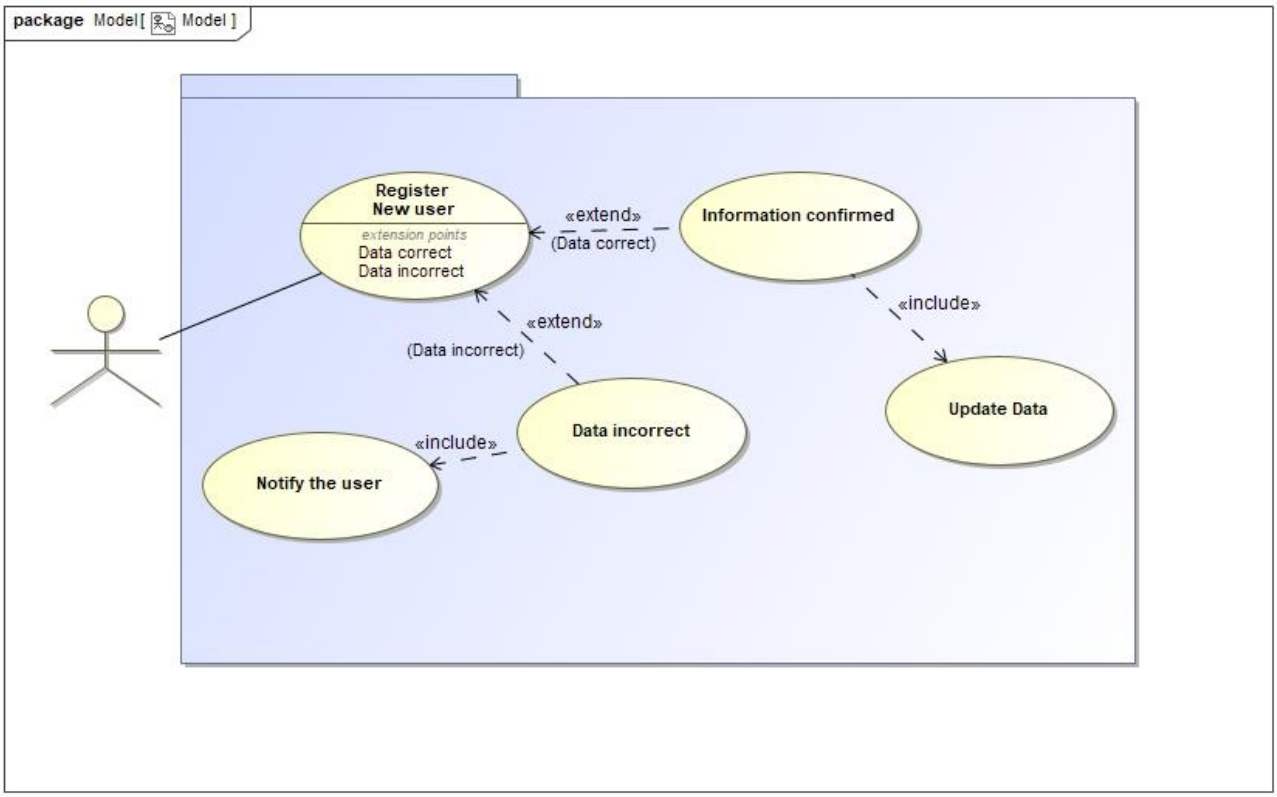


Table 3User Case Diagram "Customer Registration"

4.1.1.3Sequence Diagram

The sequence diagram describes the steps when a customer wants to register.
The customer clicks on the registration button on the website. The website queries the web server for this page and returns it to the website, which is then also displayed to the customer.
The registrant enters the requested data and confirms it. The website forwards the information to the web server. The web server verifies that the data received is correct.
If the data is invalid, the data is returned, and the registrant is shown which data was incorrect. He will be automatically prompted to retype them. This is done until the data is completely correct.
If everything is correct, the web server sends an email to the registrant to confirm and automatically verify the registration. If the email is successfully verified it forwards this to the system and asks to save the account. The system stores all accounts in the database.

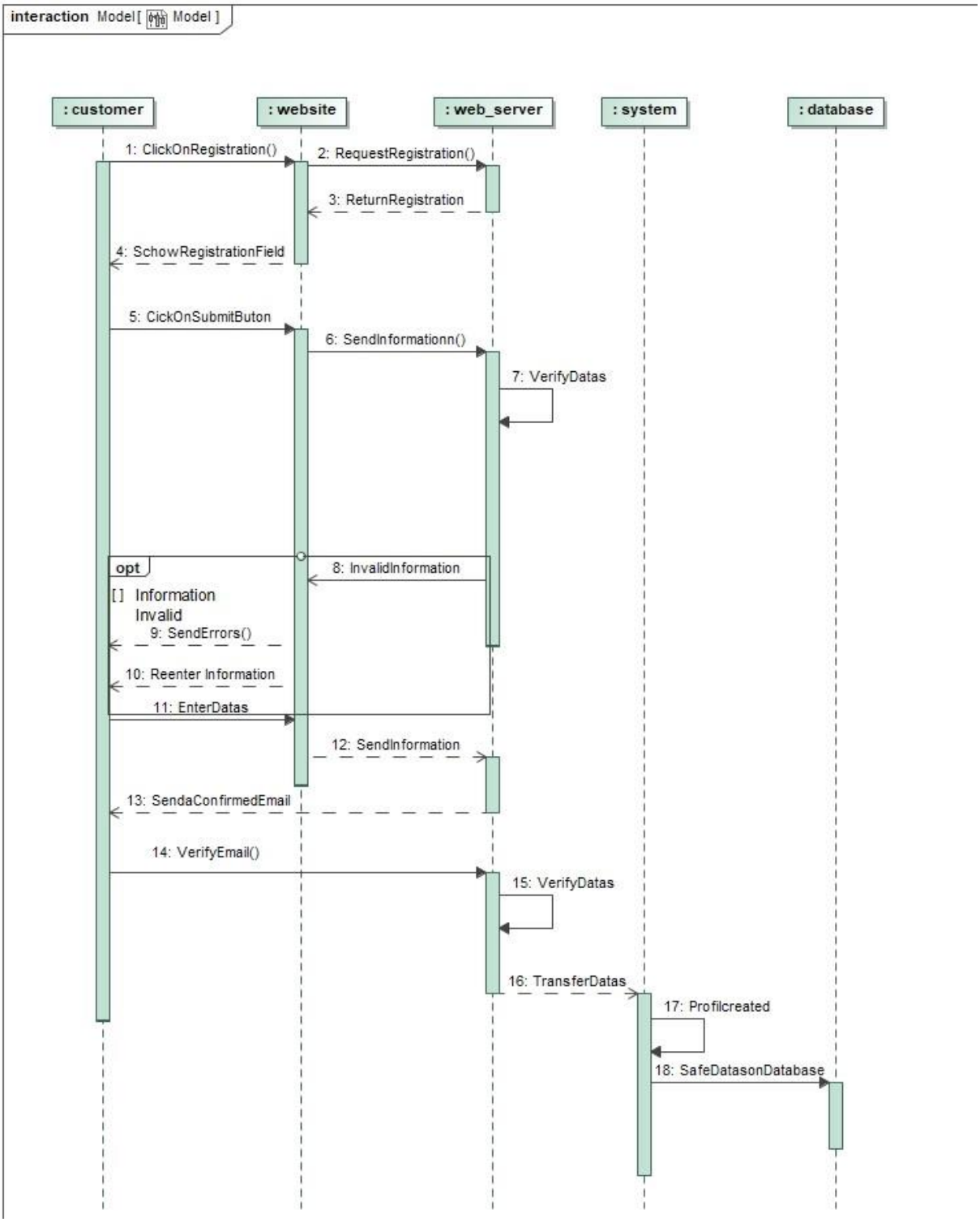


Table 4Sequence Diagram "Customer Registration"

4.1.1.4 Use Case Diagram

This activity diagram shows the process for a customer registration.

The customer needs firstly to register to be able to use our service. Then he must enter his own information like email address, name etc. If the email was not valid, he must enter the correct email again. After that if it is all correct, he can also enter his information to create an account.

At the end the customer can use our service.

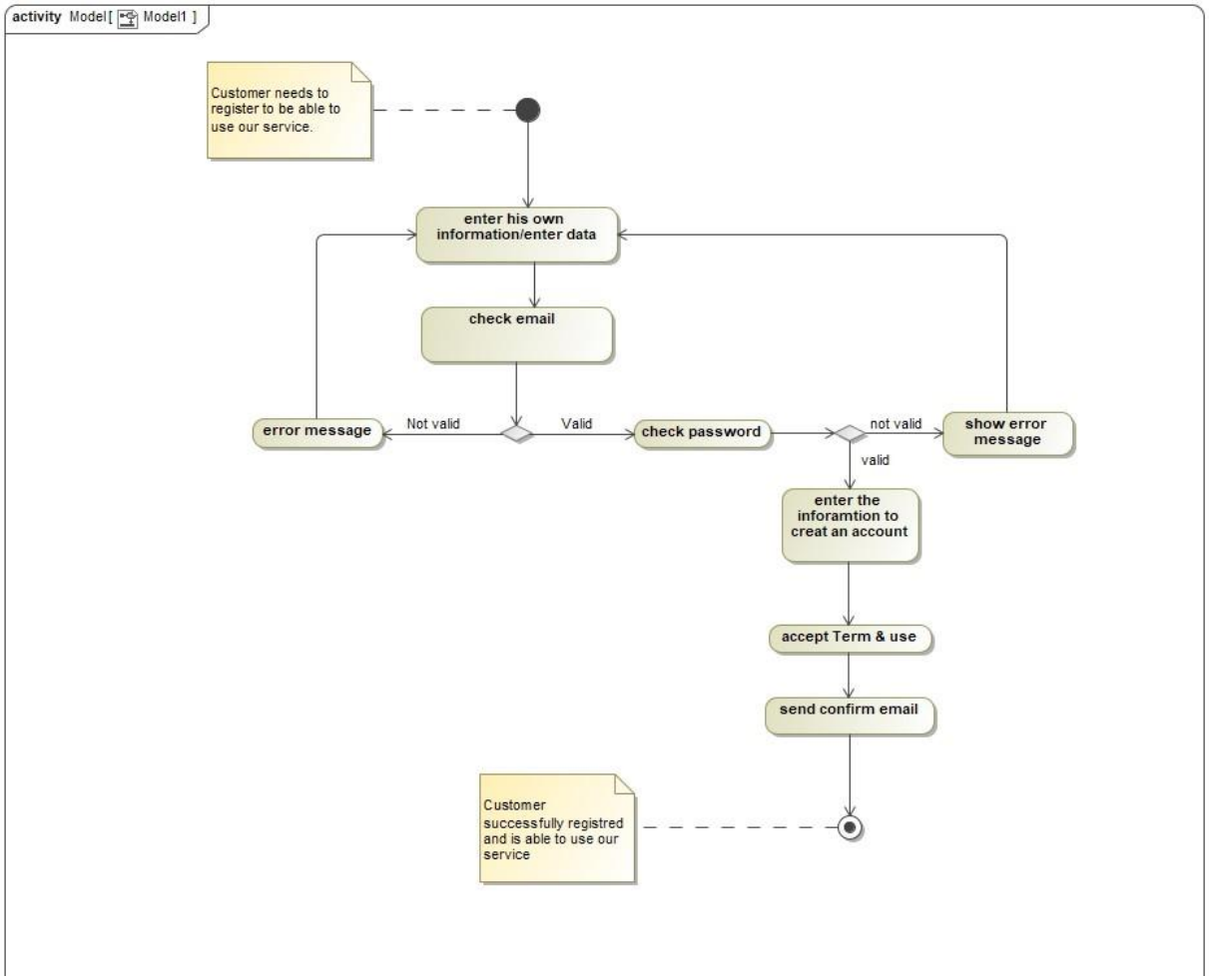
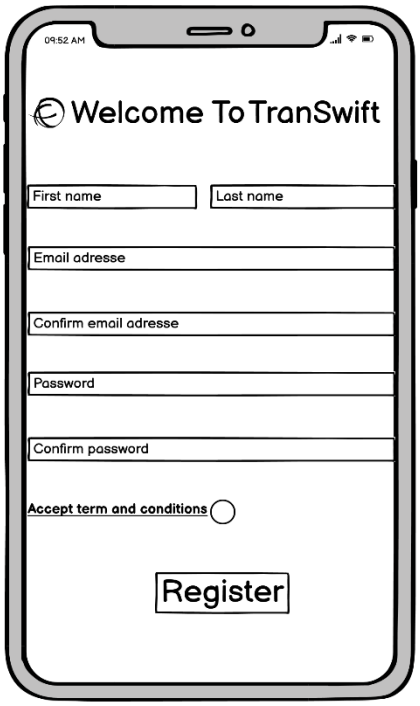
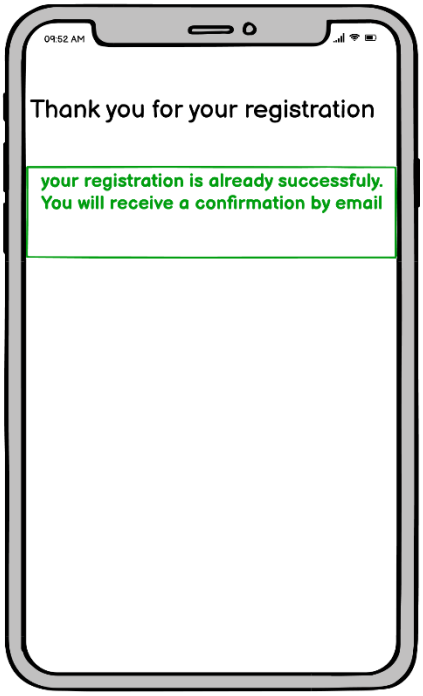


Table 5 User Case Diagram "Customer Registration

4.1.1.5 Graphical User Interfaces

This UI Prototype shows which data is required from the customer to register, and It displays also the entire needed phases to complete the registration phases.





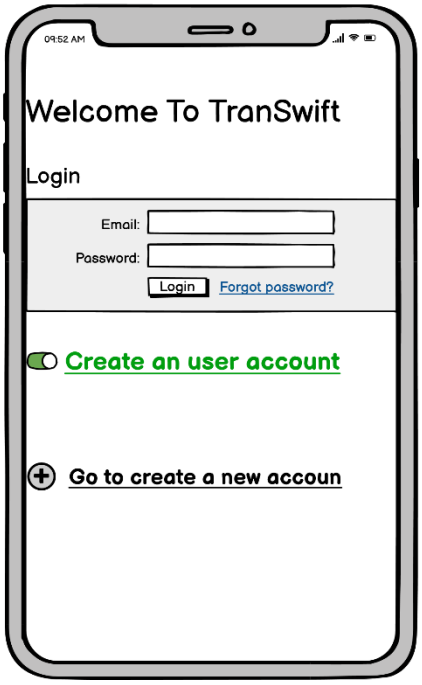


Table 6Graphical User Interface "Customer Registration"

4.1.2 Requirement-„Contractor Registration “

4.1.2.1 Use Case Specification

Use Case		Contractor registration
Description		This use case starts when the contractor wants to create an account and ends once he has successfully entered his real information to the system. The contractor must achieve the conditions requested.
Actors		The contractor
Pre-conditions		The Access to the account creation page.
Basic Flow		
	Description Actions	The contractor needs to create an account based on his data and fulfil it.
	1	Click on Sign Up button.
	2	Open registration form.
	3	Enter the personal info.
	4	Check the given info.
	5	Enter languages skills with the level. Include certificates, if available. Enter the qualifications.
	6	Answer a little quiz and fulfil some tasks.
	7	Account successfully created
	8	Redirect to success page
Alternative Flow		A
	Description	The given information already exists.
	1	The system shows a warning message.
	2	The contractor would be redirected to Login page.
Alternative Flow		B
	Description	The given information doesn’t match the requirements
	1	The system shows a warning message of the specific problem.
	2	The contractor needs to check again his information or answers.
Post-conditions		The registration has been successfully done.

Table 7User Case Specification "Contractor Registration"

4.1.2.2 Use Case Diagram

This Use Case Diagram describe the phase of creating an account of a contractor, to access to the features of the platform and get a membership. In the first steps, it seems like a registration of a customer. But at the end it's characterized by tasks, which the contractor must fulfil it, e.g., "answer some quizzes, upload his certificates and qualifications."

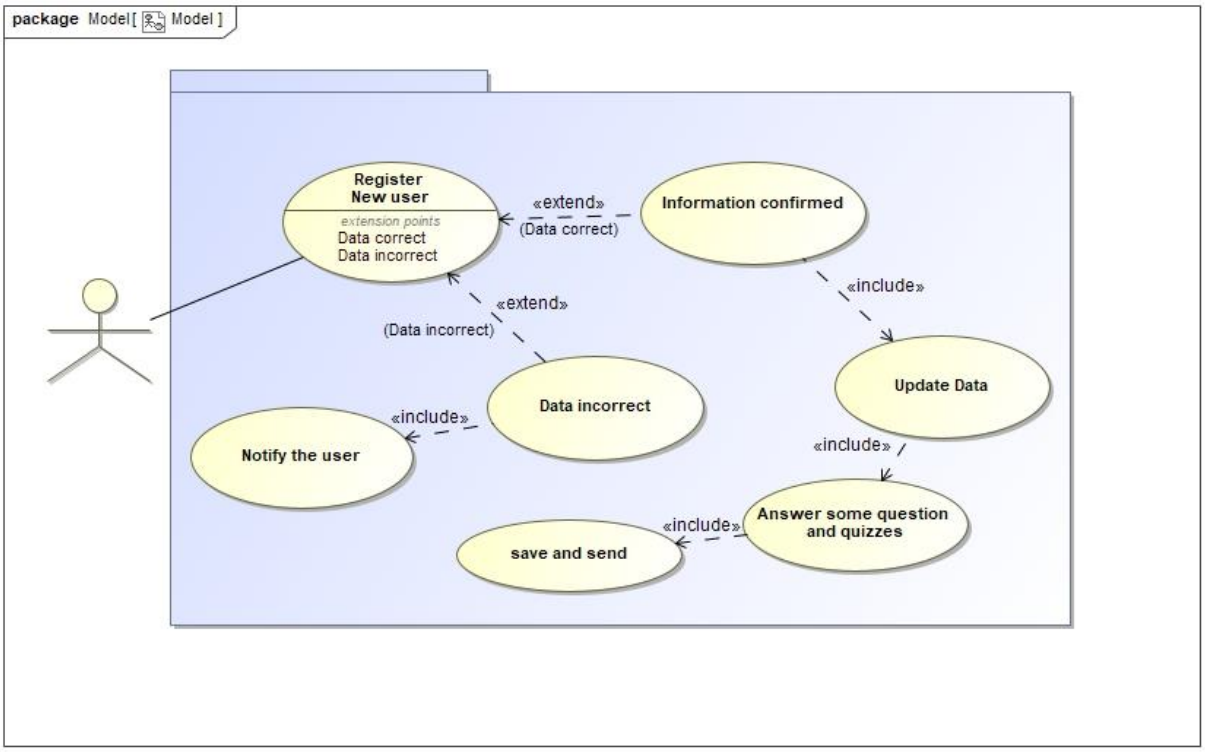


Table 8Use Case Digram "Contractor Registration"

4.1.2.3 Sequence Diagram

The sequence diagram illustrates the different steps when a contractor wants to register.
The contractor clicks on the registration button on the website. The website queries the web server for this page and returns it to the website, which is then also displayed to the contractor.
The registrant enters the requested data and confirms it. The website forwards the information to the web server. The web server verifies that the data received is correct.
If the data is invalid, the data is returned, and the registrant is shown which data was incorrect. He will be automatically prompted to retype them. This is done until the data is completely correct.
After that, the system displays some questions and quizzes to be answered. moreover, the contractor should upload certificates and skills.
Forwards this to the system and asks to save the account. The system stores all accounts in the database.

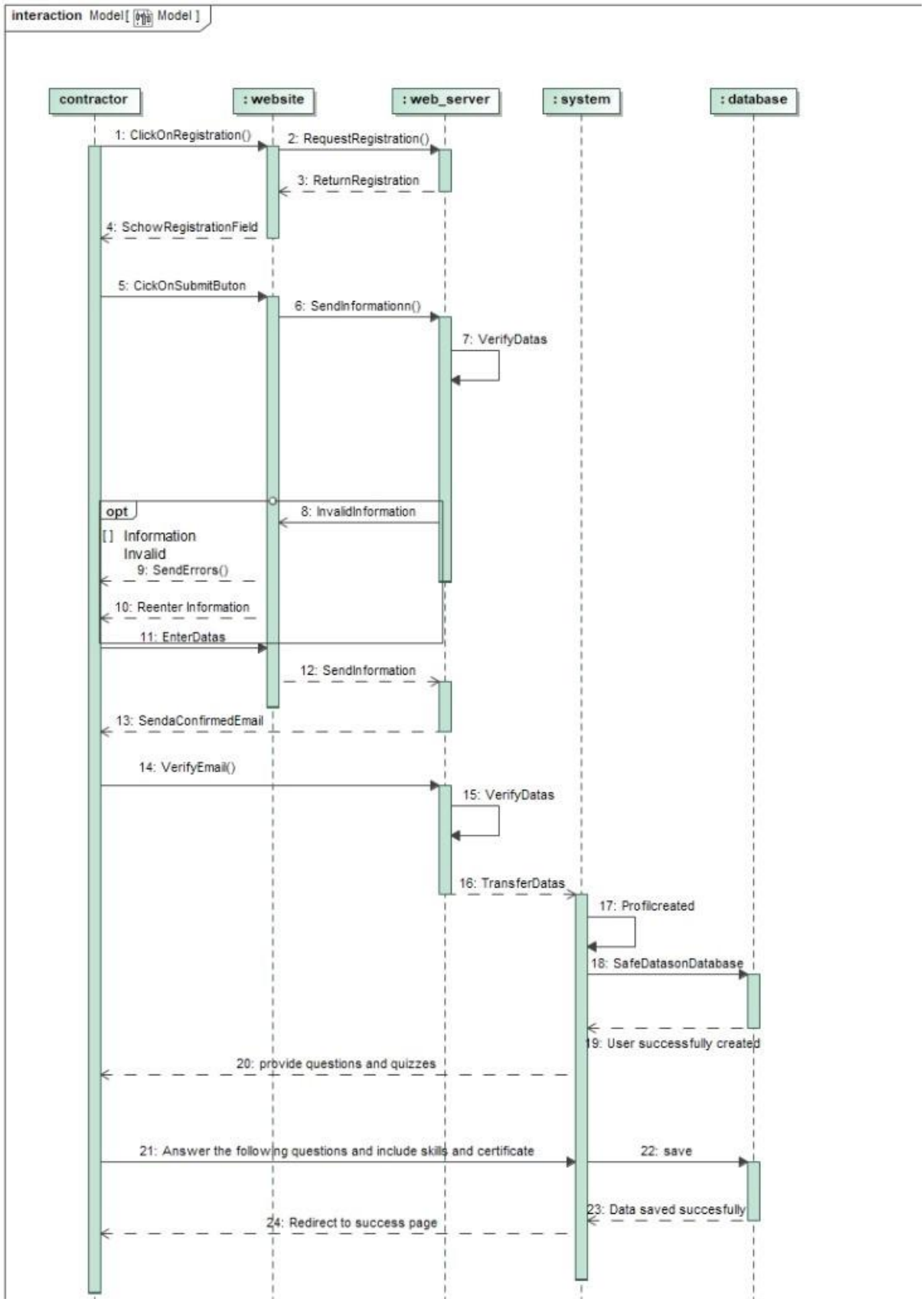


Table 9 Sequence Diagram "Contractor Registration"

4.1.2.4 Activity Diagram

This activity diagram shows the logical flow of the process for a contractor registration. Activity by activity. The contractor arrives at the interface area. From then on, the contractor needs to input his own data including answering some questions related to the assignment. And at the end the account will be successfully created.

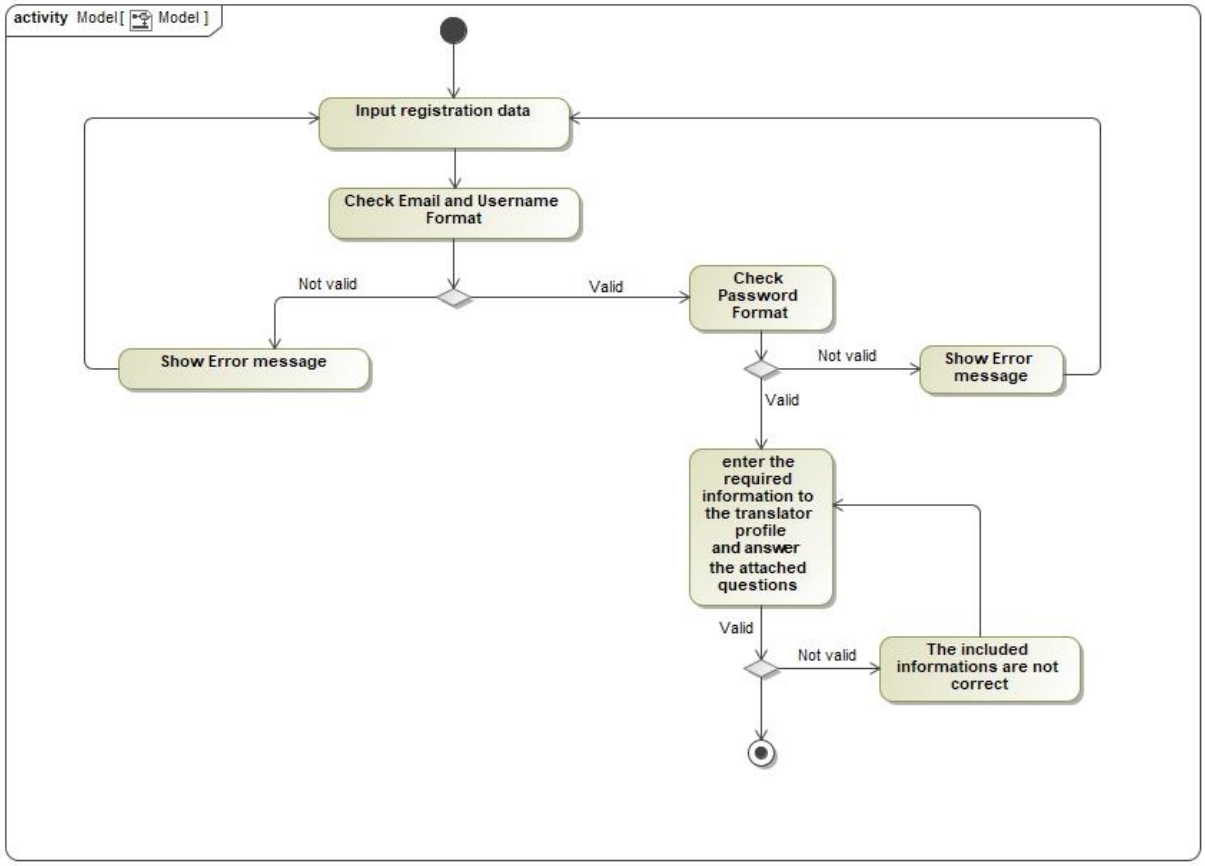


Table 10 Activity Diagram "Contractor Registration"

4.1.2.5 Graphical User Interfaces

This UI Prototype shows which data is required from the contractor to register, and It displays also the entire needed phases to complete the registration phases.

1.

2.

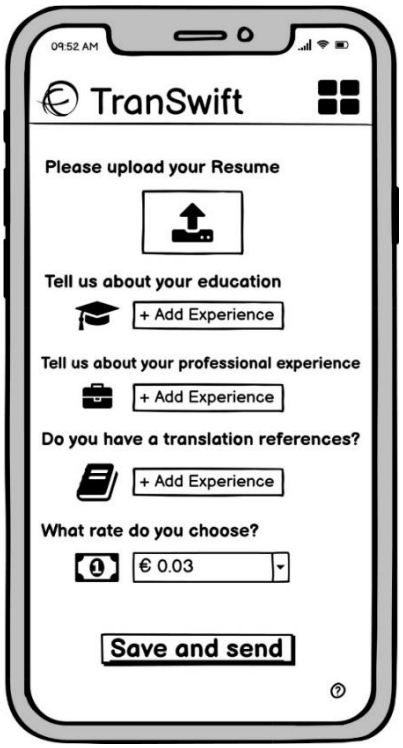


Table 11Graphical User Interface "Contractor Registration"

4.1.3 Requirement-„User Verification “[3]

4.1.3.1 Use Case Specification

Use Case	User Verification
Name	User Verification
Brief description	The Verification process starts when the User wants to register and ends when the User gets verified. The User must be verified by confirming the phone number.
Actors	User
Pre-conditions	The User has entered a correct phone number
Basic Flow **Description Flow**	
1	Server send SMS to the phone number with a PIN number
2	The User receives the PIN on his phone
3	The User enters the PIN in the TranSwift App
Alternate Flow	If the User entered a false number, they get the chance of reentering
ExceptionFlow	The Verification process ends after 5minutes not entering the PIN, the User has the option to get a new PIN resend.
Post Conditions	User completes theprofile if necessary and the Server send the Profile to the Database.

Table 12Use Case Specification "User Verification"

[3] (Xpate, 2019)

4.1.3.2 Sequence Diagram

This sequence diagram for the use case" User Verification" shows the process of Verification of the User. The User Registers and to verify the correctness of the registration information the System sends an SMS to The User. If the User enter the sent PIN right the Registration is complete, and he can login. If the PIN is wrong, he gets the tries of reentering, and der Verification has failed.

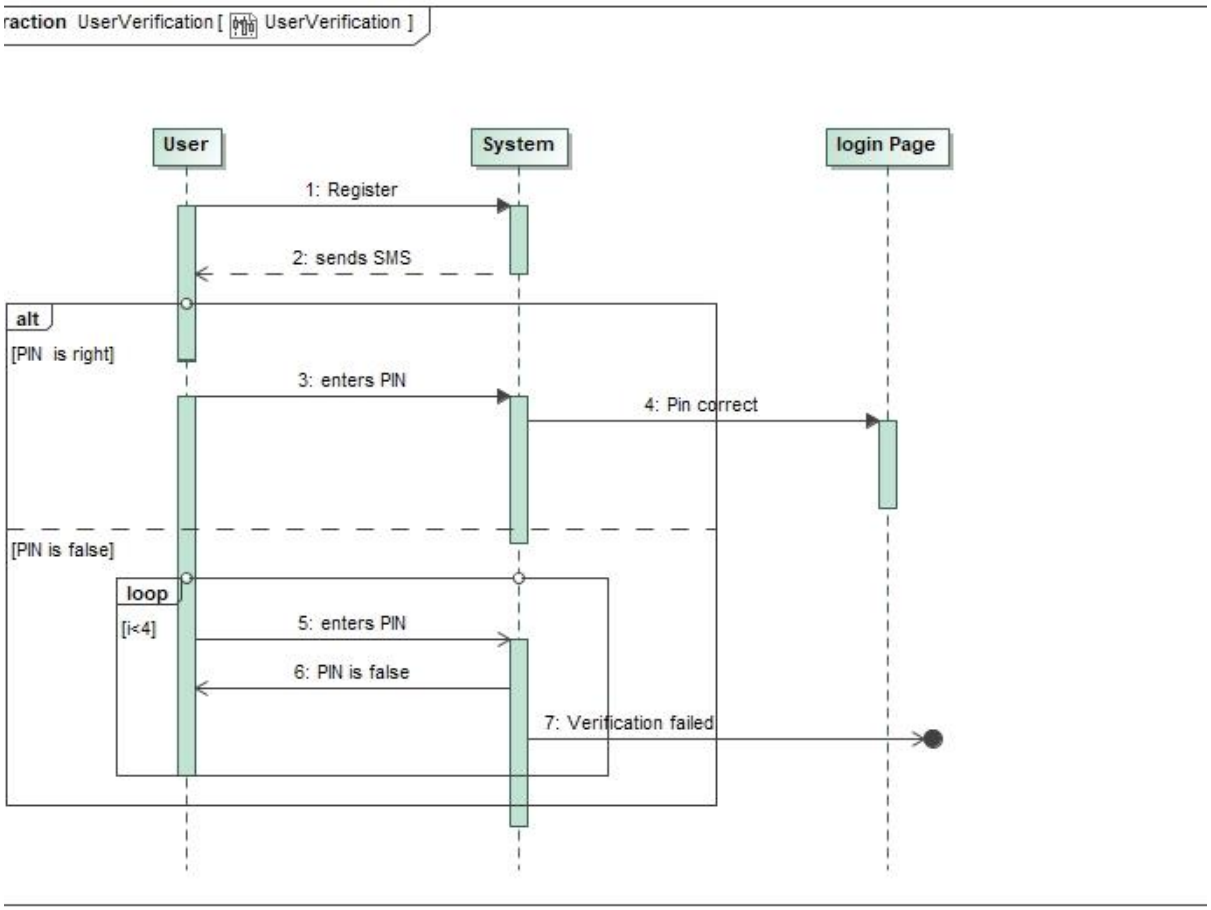


Table 13 Sequence Diagram "User Verification"

4.1.3.3 Activity Diagram

This Activity Diagram describes the use case “User Verification”. It shows the flow of the Verification of the User. The Activity start with the User registration and ends when the entered Pin is correct.

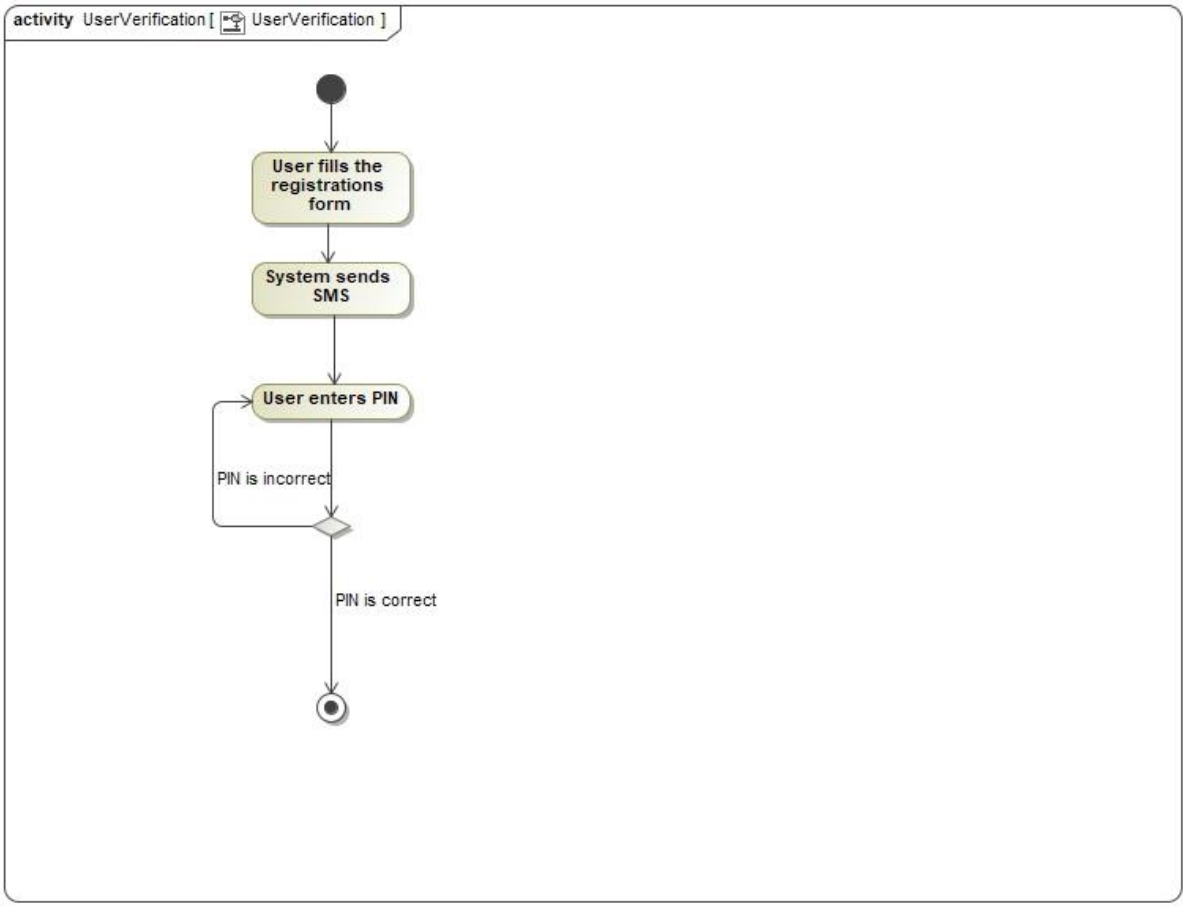


Table 14 Activity Diagram "User Verification"

4.1.3.4 Graphical User Interface

This UI Prototype shows how and where the User can enter the verification Pin. The User can request a new Pin or change the phone number if needed.

Table 15 Graphical User Interface "User Verification"

4.1.4 Requirement-„User Login “[4]

4.1.4.1 Use Case Specification

Use Case	User Login
Description	This use case starts when a user wants to log in and ends once the user has successfully logged in to the system. The user must enter a valid email and password.
Actors	Contractor &Customer
Pre-conditions	App or web browser installed, connected to the internet
Basic Flow **Description Actions** **1** **2** **3**	This is the main scenario where the user already has a valid account and password The user enters his email address The user enters his password The system verifies that the email address and password are valid and compatible. If so, the user is logged in to the system.
Alternative Flow **Description Actions** **1** **2** **3**	**A** The user doesn't have an account yet. The user will be asked to register and clicks on the registration button. The user creates an account. The user will be redirected to the login page and logs in.
Alternative Flow **Description Actions** **1** **2** **3**	**B** The user has already an account but has forgotten the password. The user asks for a new password The user receives an email for resetting the password. The user will be redirected to the login again and logs in successfully.
Post-conditions	The user is successfully logged in to the system

Table 16 Use Case Specification "User Login"

[4] (Tariq, 2020)

4.1.4.2 Use Case Diagram

This diagram shows the use case of log in. The user enters his data and tries to login. In the main scenario the data is correct, and the user will be logged in successfully, otherwise there are two possibilities:

1. The user chooses to reset his password if he has forgotten it, and this case is illustrated by the reset password extension. But before resetting the password, the user must verify his identity which leads to the inclusion of the user verification use case.

2. The user chooses to register if he has no account, and in this case, he will be redirected to the registration page.

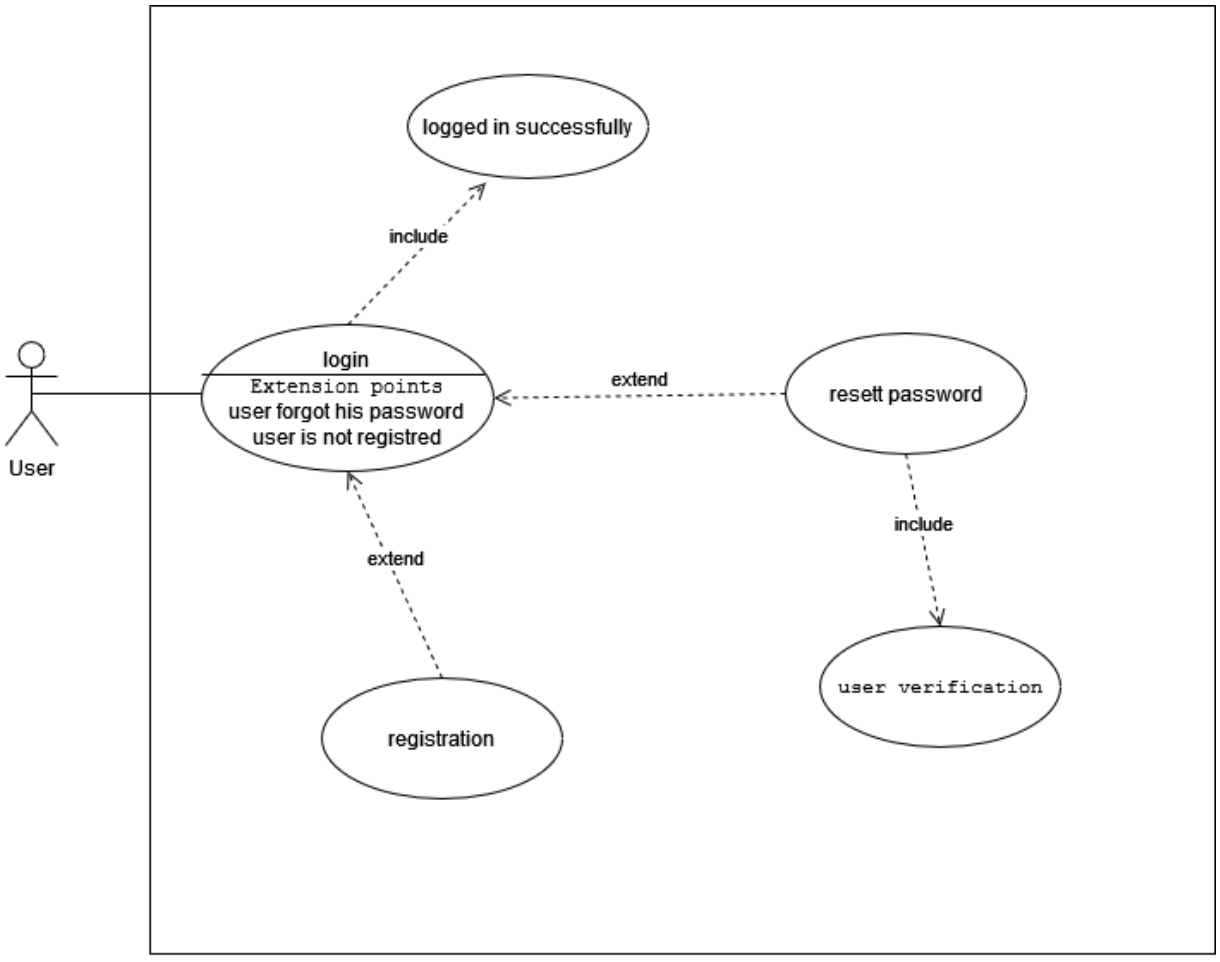


Table 17 Use Case Diagram "User Login"

4.1.4.3 Sequence Diagram

This diagram shows in detail the sequence of actions taking place in the system during the use case of login.

As mentioned before in the use case, the user enters his email and password in the login page. A request will be sent to the server to check the correctness and the compatibility of the data in the database, and this leads to two scenarios:

1. The main scenario is the case where the entered data from the user is correct and in this case the server sends back a positive return message to the login page and the user is successfully logged in.

2. The alternative scenario is the case where the entered data from the user is not correct and in this case the server sends a negative return message, and the user will be told to retry again in the login page. This may happen several times which implies a loop. But to avoid an endless loop, we restrict the number of tries to overall 3 tries.

   After 3 failed tries, the server will automatically redirect the user to the resetting password page where the user has again to enter his email. The email will be checked first if it exists in the database, and again we have here two scenarios:

   - If the return answer is yes, the user will receive automatically a password resetting link in his email, and he will be able to reset the password and try to log in again
   - Otherwise, if the return message is no, which mean he doesn't an account yet, he will be redirected to the registration page to create a new account with his email.

   In this last case, another loop can be implemented to give the user the possibility to try again if he mistyped his email for example, but to keep the diagram simpler and not too long we decided to not include this option, especially knowing that the user can always go back to the resetting password page after being redirected to the registration page.

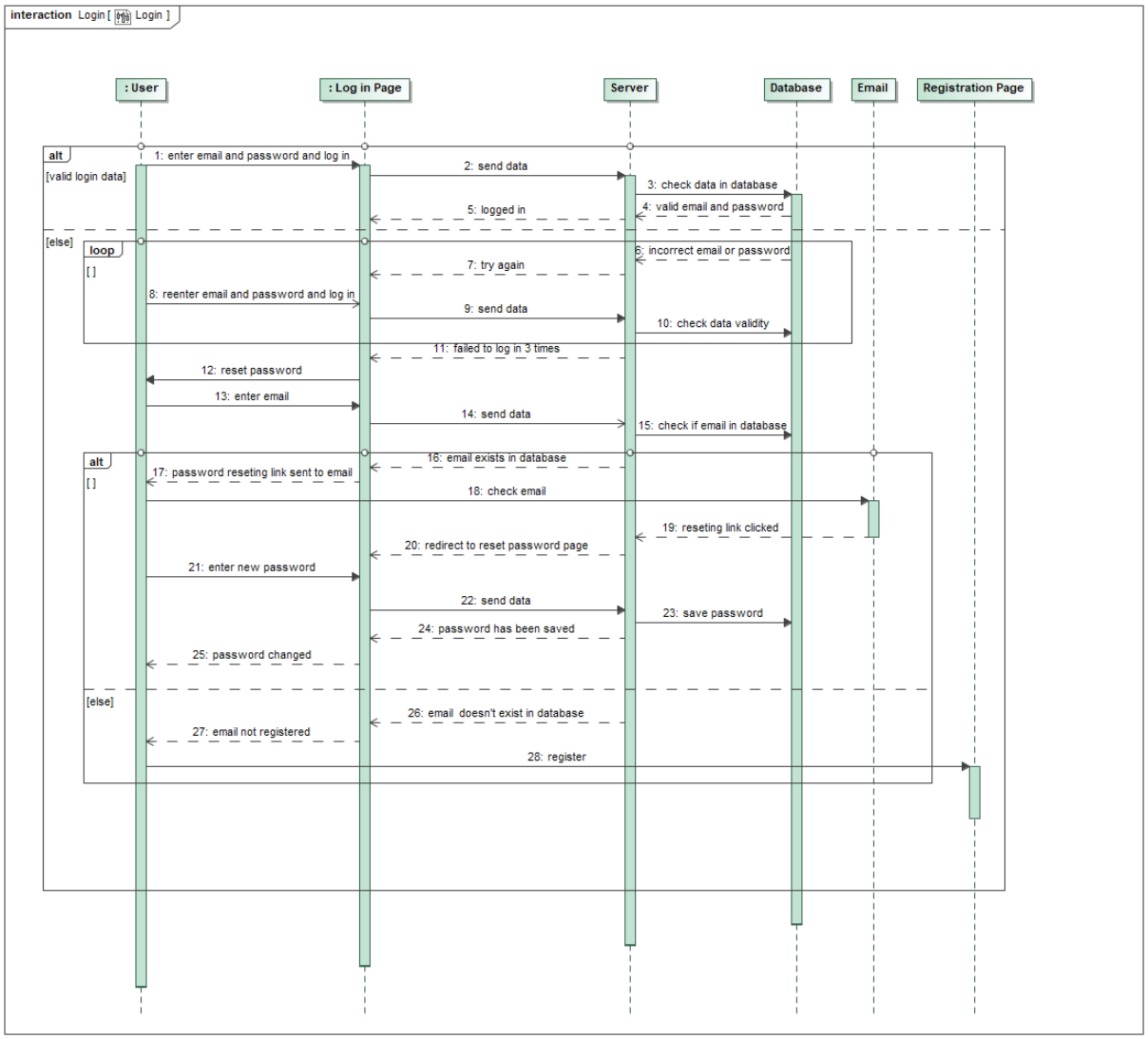


Table 18 Sequence Diagram "User Login"

4.1.4.4 Activity Diagram

This diagram shows the different activities performed by the user and the system during a login session.

The user enters first his data and if it's correct, the session ends immediately with the user being successfully logged in. Otherwise, the system asks the user to retry again until he fails three successive times. Then the system asks the user to enter his email. The user enters his email and if it is already assigned to an account, the System sends a resetting link or code to the user. The user sets a new password and finally he can log in successfully. If the email is not assigned to any account, the system redirects the user to the registration page where he can register and after a successful registration, he can log in successfully.

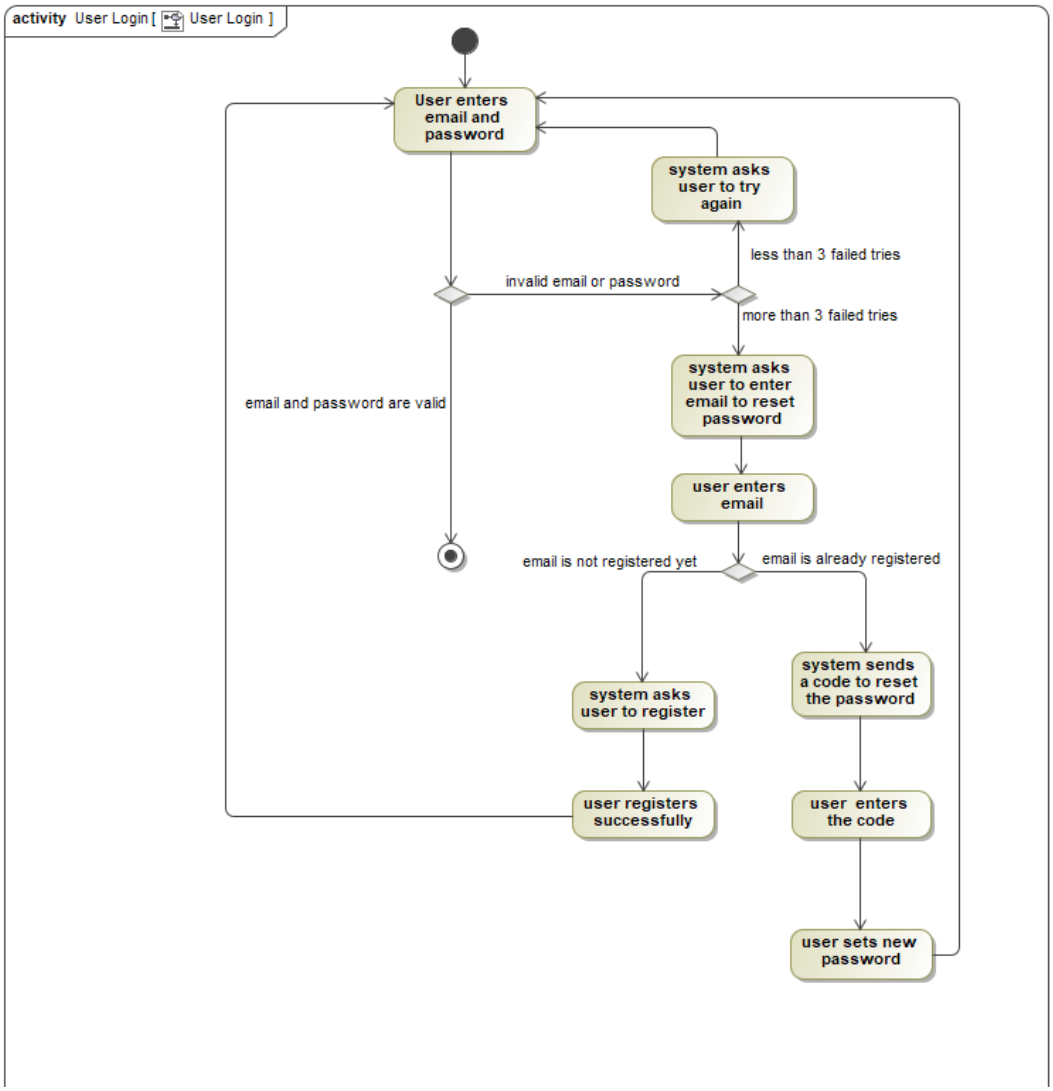


Table 19 Activity Diagram "User Login"

4.1.4.5 Graphical User Interface

This graphical user interface represents the login page where the user has to enter his email and password to login.

The user also has the possibility to click on the “forgot password “link to reset his password, and in this case go to the page bellow. Also, in the case of entering wrong data 3 successive times, he will be redirected automatically to the same page below.

If the button “Create new account” is clicked, the user will be redirected to the registration page that was shown before in the registration requirements.

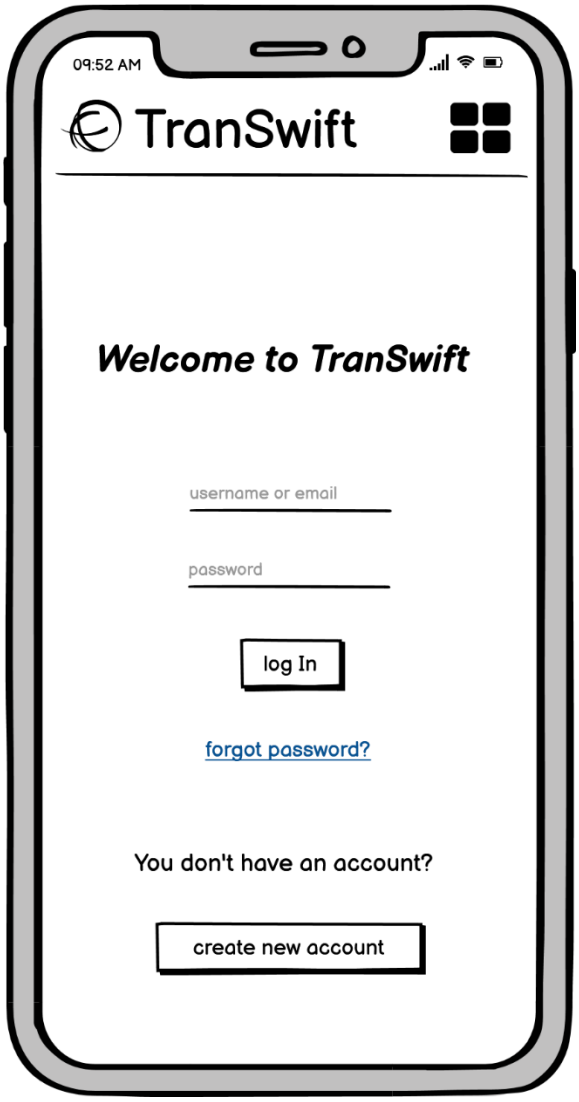


Table 20 Graphical User Interface "User Login"

As mentioned above, this prototype shows what a resetting password process pages would look like.

In the first page, the user must enter first their email and click the “Send Reset Link” button, and after clicking the link sent to them, they will land in the second page where they can enter a new password and confirm it so that they can successfully reset their password by clicking on the “Reset password” button.

Table 21Graphical User Interface "User Login"

4.1.5 Requirement-„Account Management “

4.1.5.1 Use Case Specification

Use Case	**Account management**
Description	This use case starts when the customer/contractor accesses the system that enables him to update the information that is maintained in the account and ends when the updated information is saved.
Actors	Customer/Contractor
Pre-conditions	The customer/Contractor must be registered.
Basic Flow **Description** **Actions** **1** **2** **3** **4**	The customer and the contractor want to modify the profile. The system displays the customer/contractor information stored. The customer/contractor enters the desired information and requests that the system saves the entered information. The system validates the entered customer/contractor information. Information for the customer/contractor are stored in the account. The system notifies the customer/contractor that the account has been updated.
Alternative Flow **Actions** **1** **2** **3** **4**	The customer/contractor can cancel the account update/deactivation at any time. If the customer/contractor enters invalid information: The system describes which of the entered data was invalid and suggest to the customer/contractor to enter a valid data. The system requests the customer/contractor to re-enter the invalid information. The customer/contractor re-enters the information and the system re-validates it. The information is stored if the valid information is entered.
Post conditions	The customer/contractor entered data is saved in the account.

Table 22 Use Case Specification "Account Management"

4.1.5.2 Use Case Diagram

In this use case diagram, we see that the customer and the contractor can change and modify details about their personal data, used payment method and delete or deactivate their account.

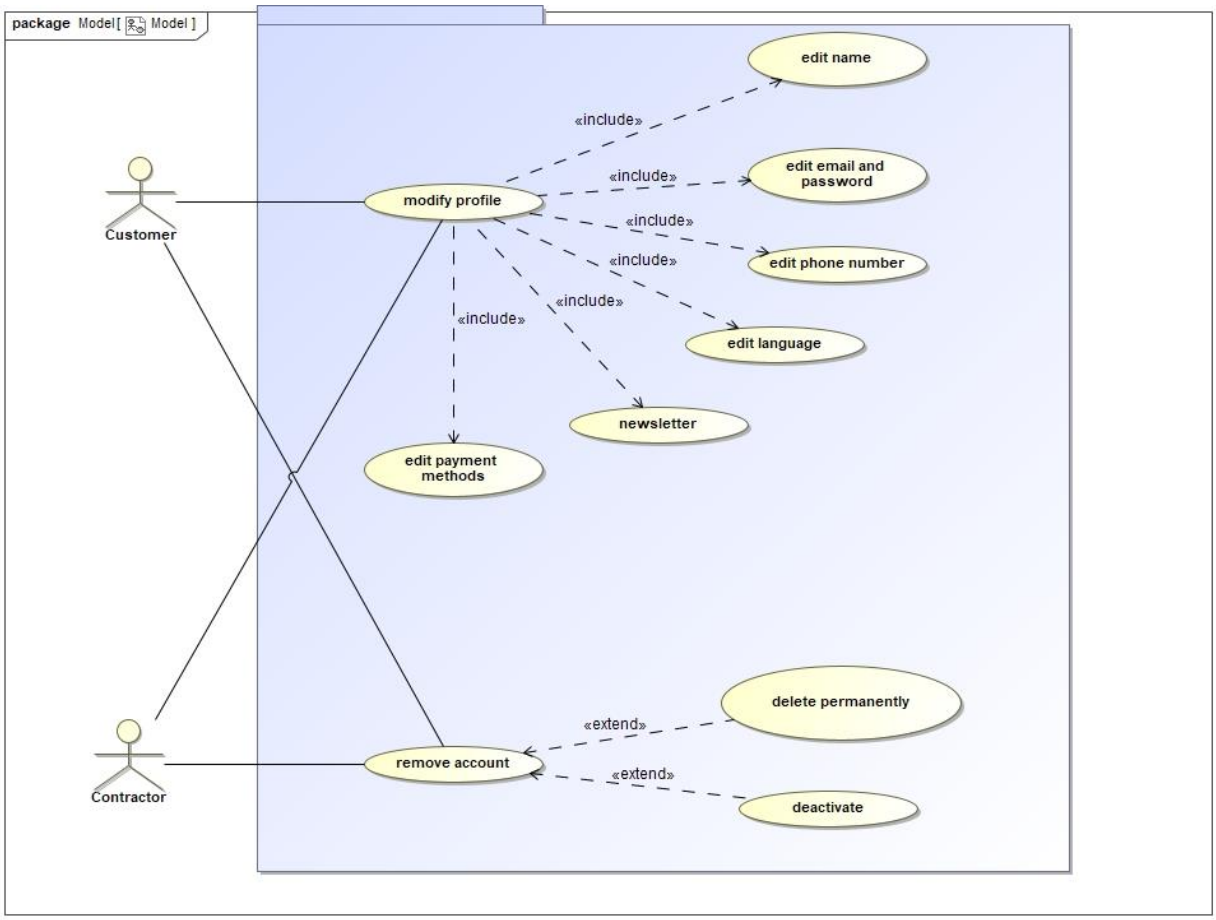


Table 23 Use Case Diagram "Account Management"

4.1.5.3 Sequence Diagram

The sequence diagram describes the steps of “Account Management”. First, the profile page will be displayed and then the customer/contractor can decide what he wants to change or modify. As an example: If the customer/contractor wants to change the email address, he must first enter a code that he receives from the server. If the entered code is incorrect, he must try the procedure again until he enters the correct code. And if the entered code is correct, he can then enter the new email address and will be saved.

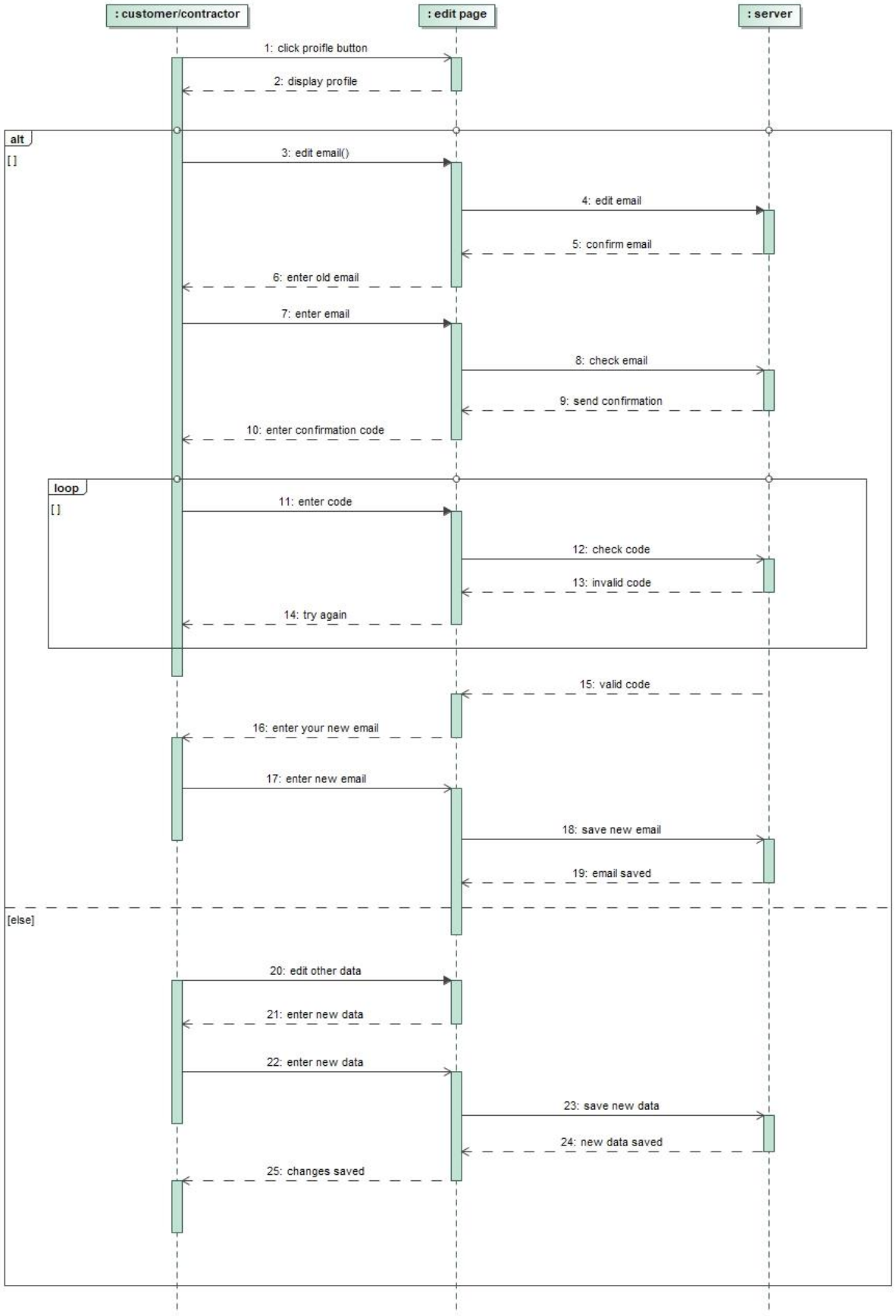


Table 24 Sequence Diagram "Account Management"

4.1.5.4 Activity Diagram

The activity diagram shows the options that the customer/contractor can change. After the changes are made, they will be saved.

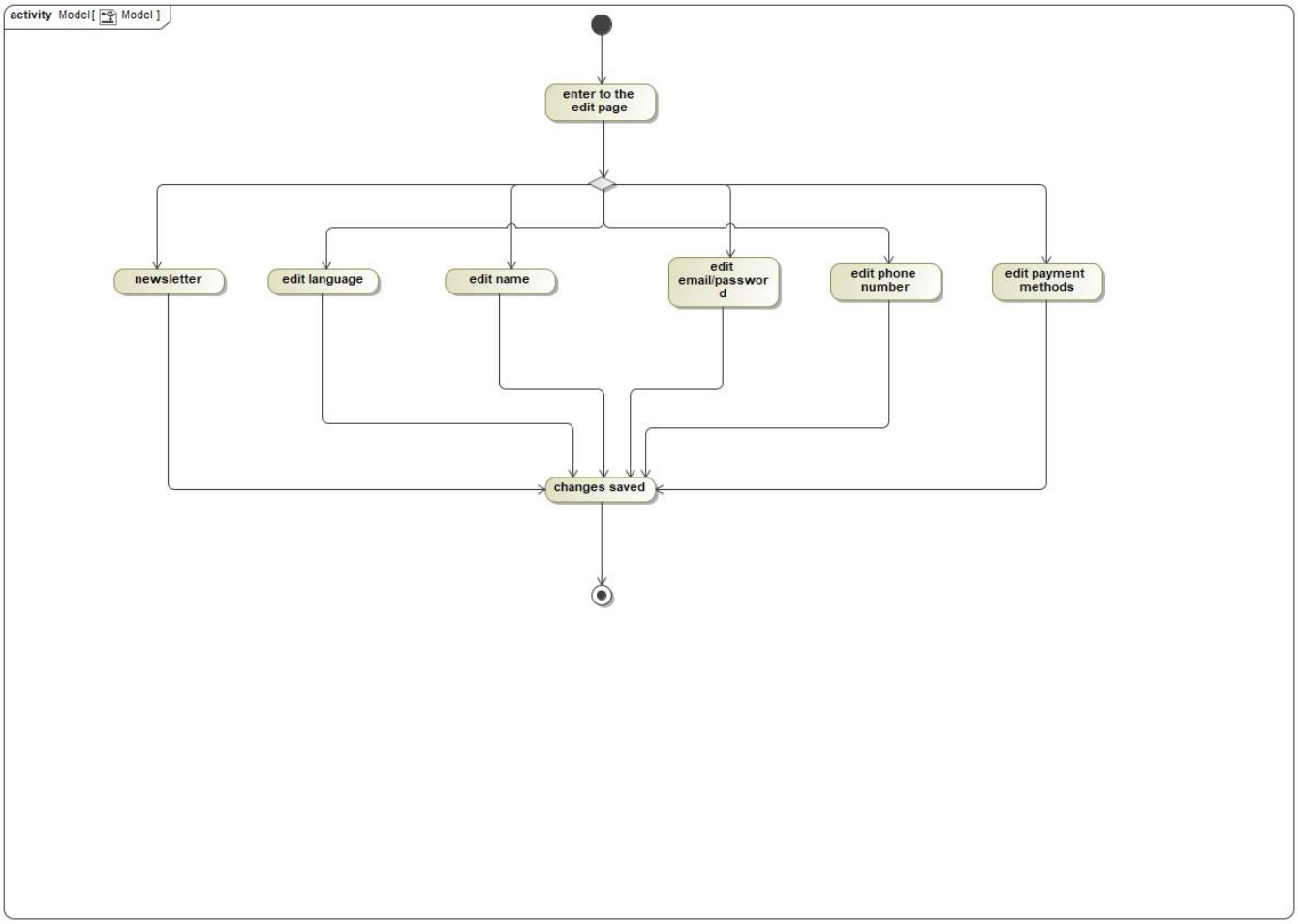


Table 25 Activity Diagram "Account Management"

4.1.5.5 Graphical User Interface

The graphical user interface shows the different steps to modify the account. Each screen shows a different step displayed to the user.

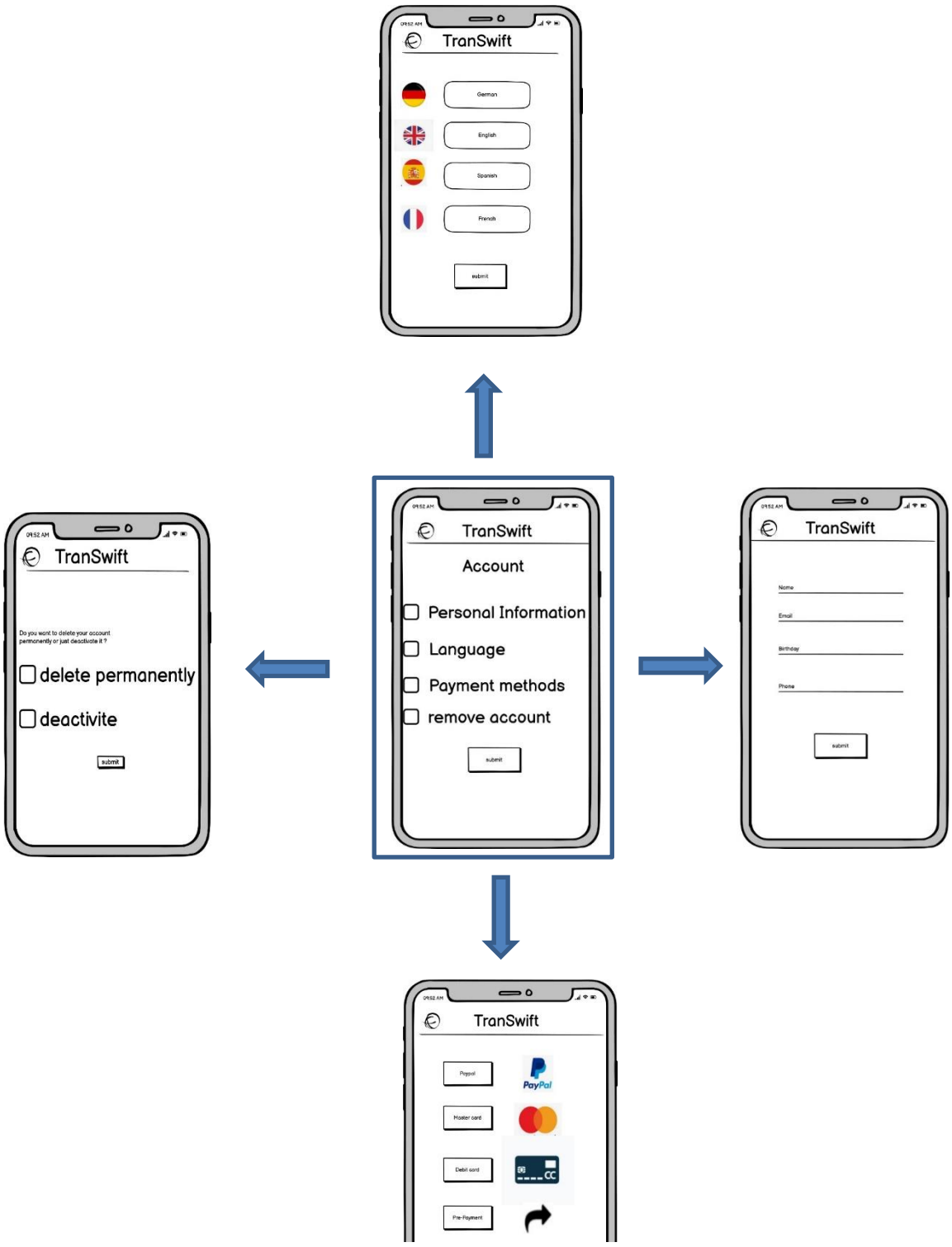


Table 26 Graphical User Diagram "Account Management"

4.1.6 Requirement-„Contractor Search Filter “

4.1.6.1 Use Case Specification

Use Case	contractor search filter
Description	Filters let the customer create a list of records that meet an appropriate translator-profile and ends when the search lets the customer find a single contractor based on his needs.
Actors	The customer
Pre-conditions	The customer account has been logged
Basic Flow **Description Actions** **1** **2** **3**	The system shows results. The customer specifies the required translator features based on his needs using the filter option. The system provides the customer with the possible translator. The customer checks the translator profile.
Alternative Flow **Description** **Actions** **1** **2**	The system shows no results. The filter doesn't provide any translator who can fulfill the task. The customer is redirected automatically to the "User feedback"/"Customer service".
Post-conditions	The customer sends a translation request to the chosen translator.

Table 27 Use Case Specification "Account Management"

4.1.6.2 Use Case Diagram

This Use Case Diagram describe the phase of searching an appropriate contractor based on customer's needs and requirements.

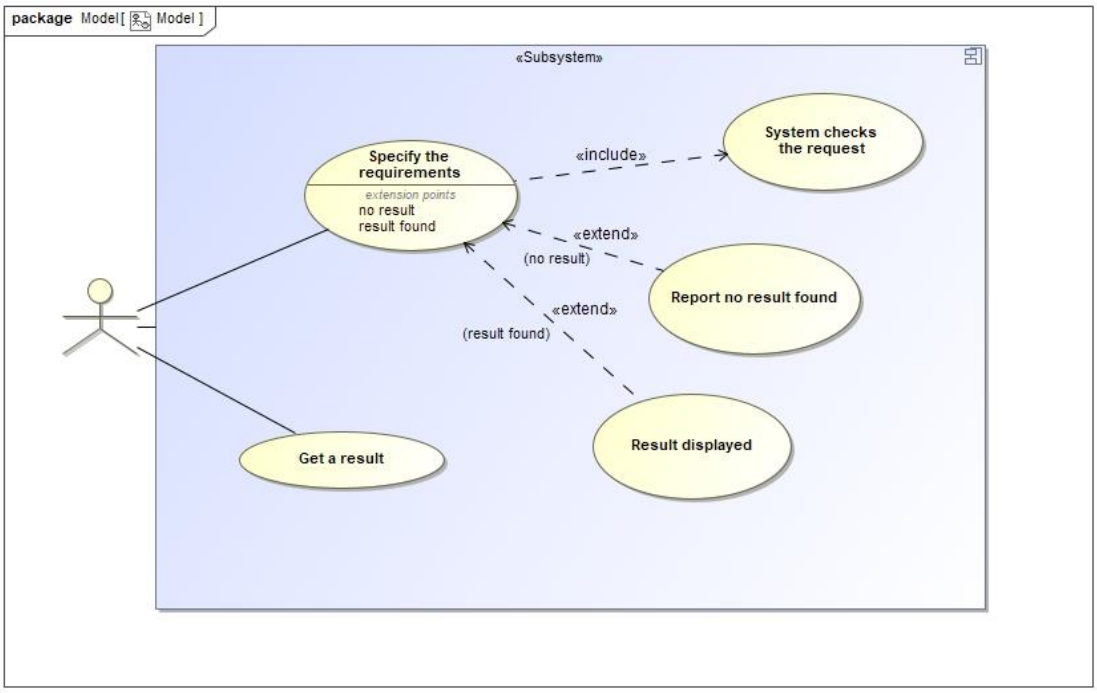


Table 28 Use Case Diagram "Contractor Search Filter"

4.1.6.3 Sequence Diagram

This sequence diagram shows the process of affording the system the specific features in a needed contractor profile by a customer using the filter option. After a check in the Database, the system should either display the customer the search results if it exists, or a Message of "No result".

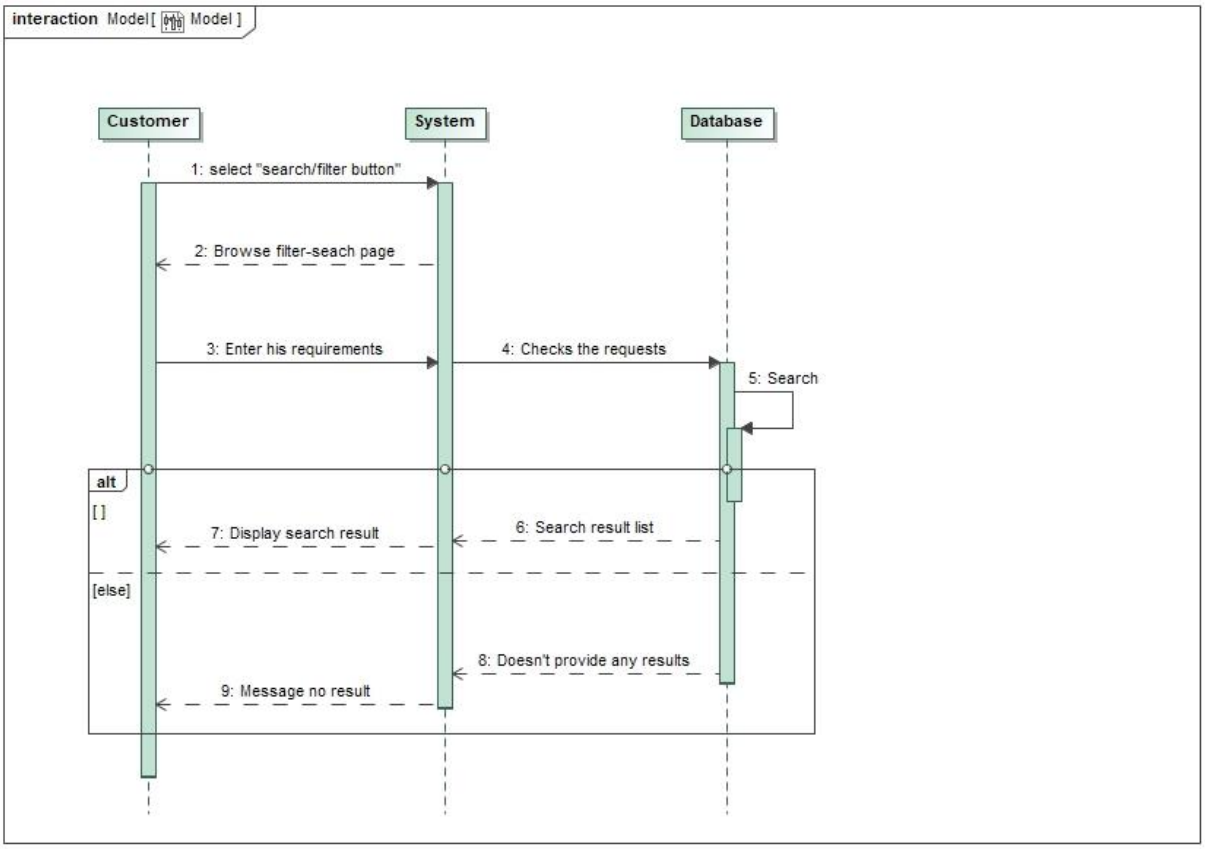


Table 29Sequence Diagram "Contractor Search Filter"

4.1.6.4 Activity Diagram

This activity diagram represents the flow of the process. The customer has the choice to make a simple superficial search, or an advanced search by using the filter option to insert the needed requirements. Thereby the customer gets the best result.

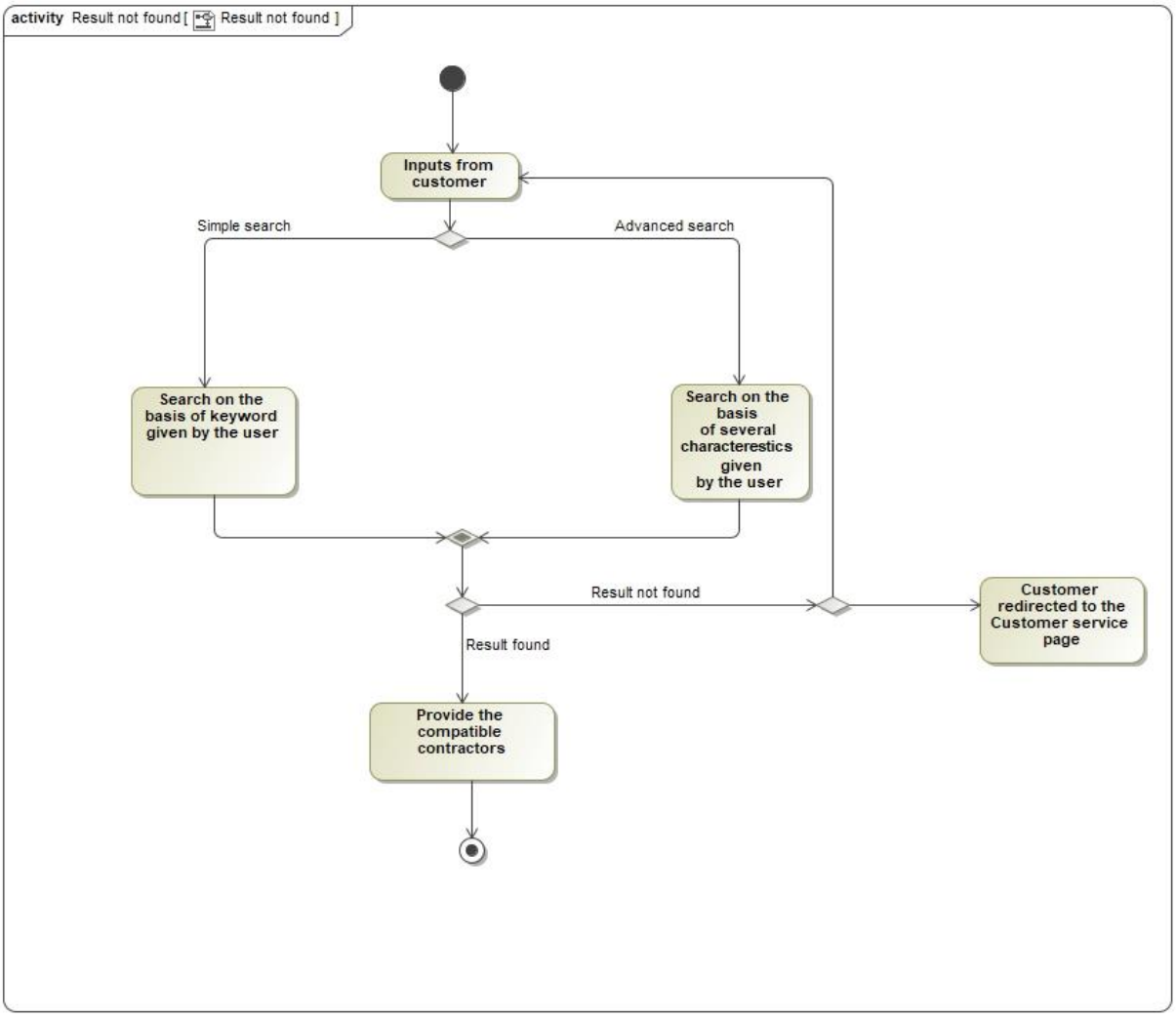
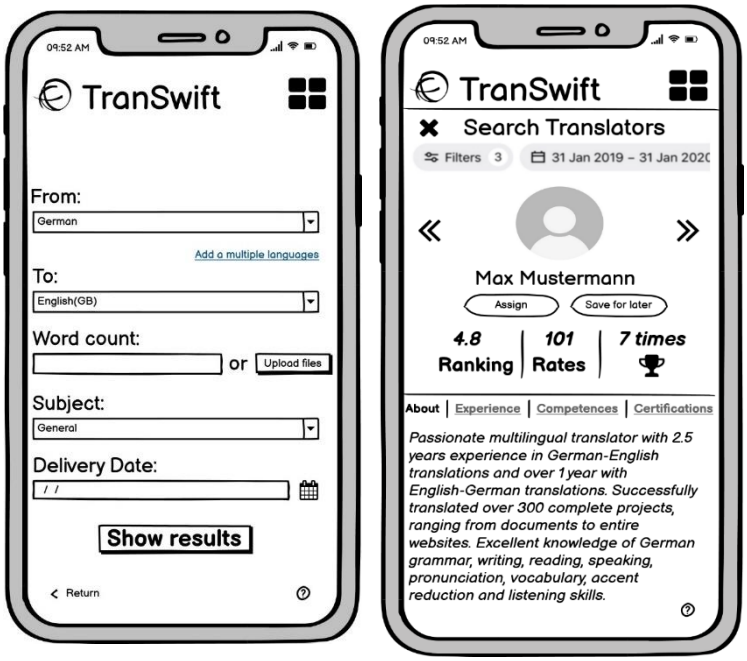


Table 30 Activity Diagram "Contractor Search Filter"

4.1.6.5 Graphical User Interface

This UI Prototype shows the options of filter directed to the customer to get a specific and optimal result.

1.2.Case when a result found.



3. Case when no result found.

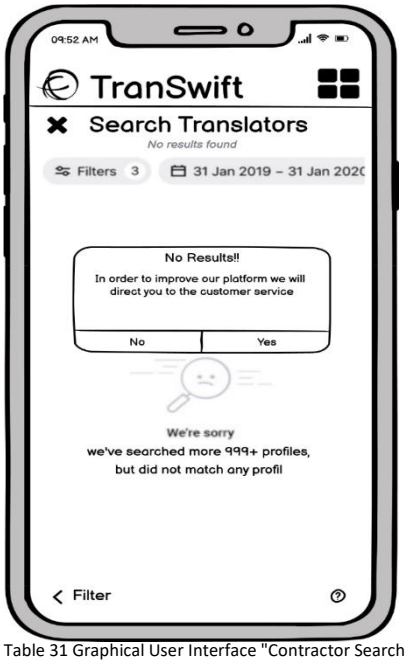


Table 31 Graphical User Interface "Contractor Search Filter"

4.1.7 Requirement-„New Translation Request “

4.1.7.1 Use Case Specification

Use Case	New Translation Request
Description	This use case starts when a customer has chosen a contractor and ends once the customer sends a new translation request. The customer has to make an initial offer including the price and the deadline.
Actors	The customer.
Pre-conditions	the customer must be registered. The customer has successfully found an appropriate translator.
Basic Flow **Description** **Actions** **1** **2** **3** **4**	 The customer sends a new translation request The customer makes an initial offer and chooses a price. The customer enters a deadline. The customer verifies his offer and confirm. The customer sends the request.
Alternative Flow **Description** **Actions** **1** **2** **3**	A The customer decides to change the offer. The customer cancels the request. The customer edits his offer. The customer sends a request
Post-conditions	The customer receives a confirmation of his request per email.

Table 32 Use Case Specification "New Translation Request"

4.1.7.2 Use Case Diagram

The use case diagram shows the steps when the customer wants to make an offer like a new translation.

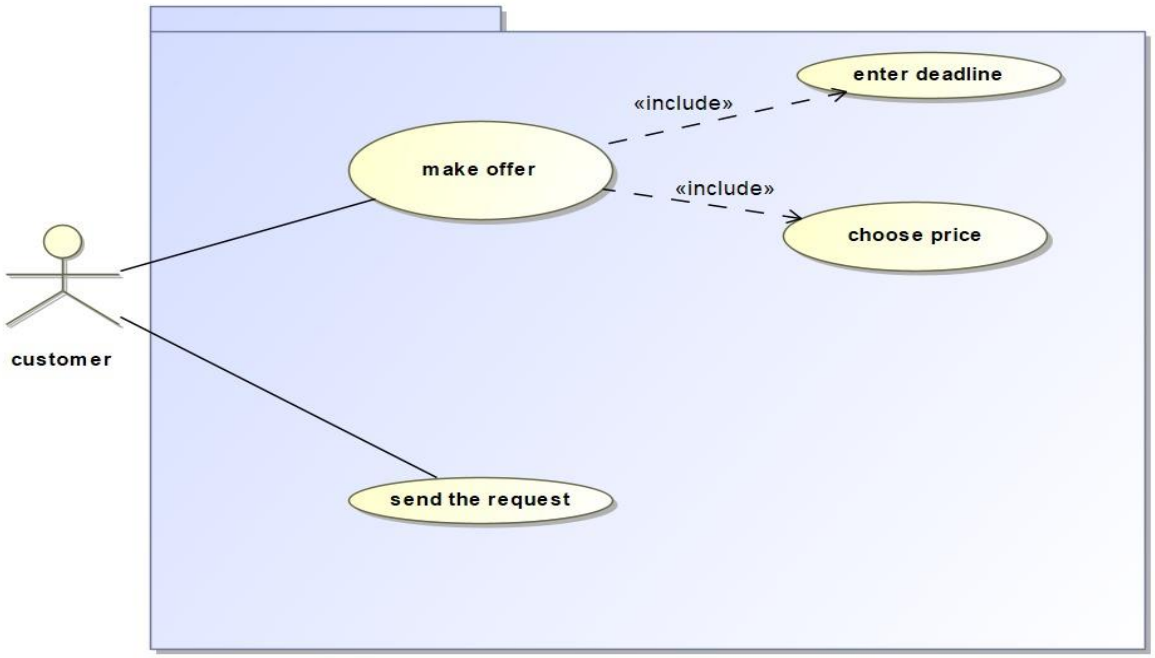


Table 33 Use Case Diagram "New Translation Request"

4.1.7.3 Sequence Diagram

This sequence diagram shows clearly how the customer can send a new translation request.

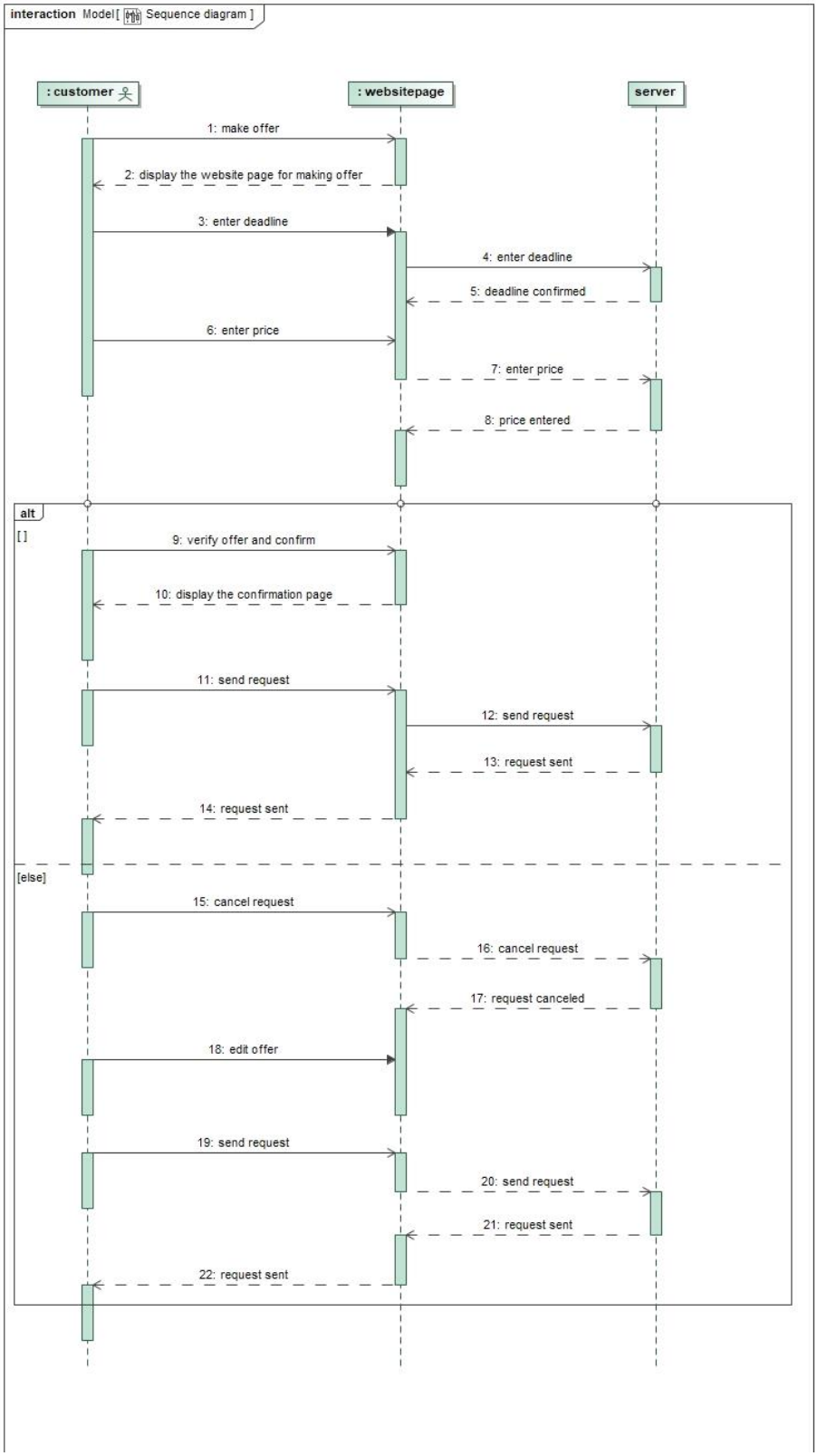


Table 34Sequence Diagram "New Translation Request"

4.1.7.4 Activity Diagram

The activity diagram shows again the process of sending a new translation request after entering the price and the deadline.

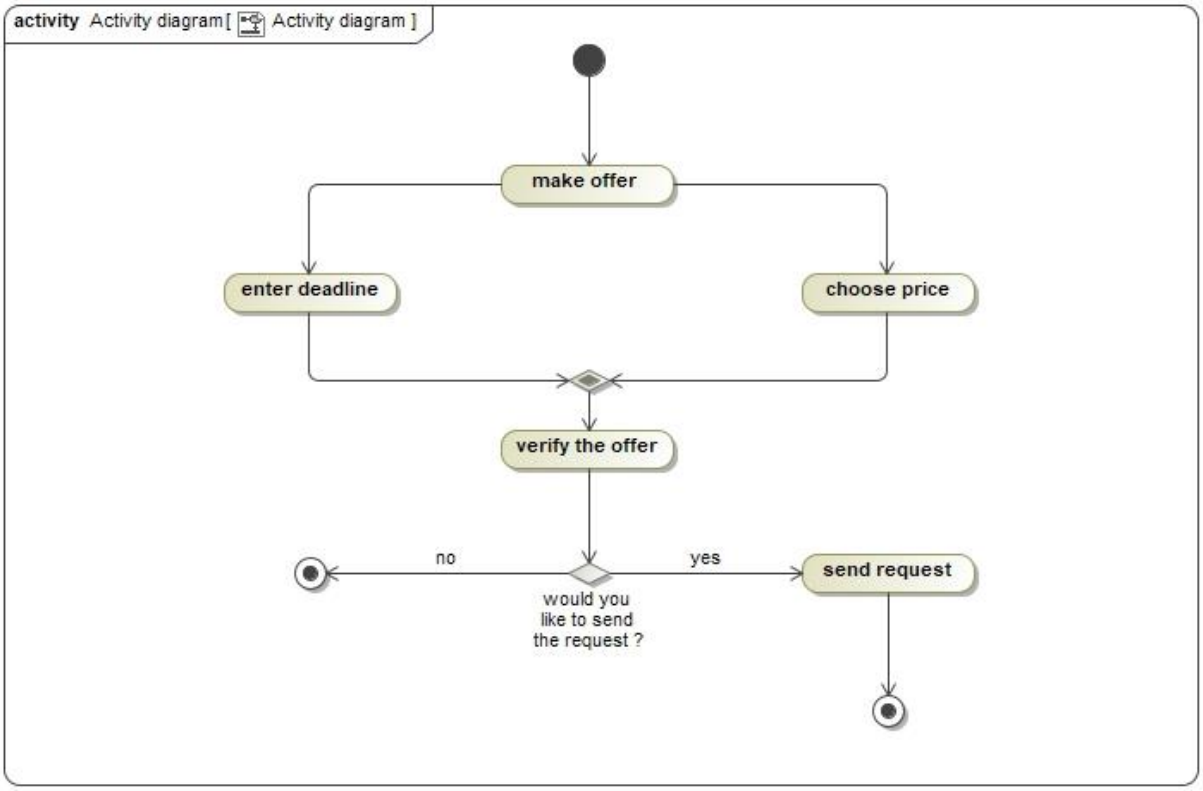


Table 35 Activity Diagram "New Translation Request"

4.1.7.5 Graphical User Interface

The graphical user interface shows how it looks when the customer makes an offer.

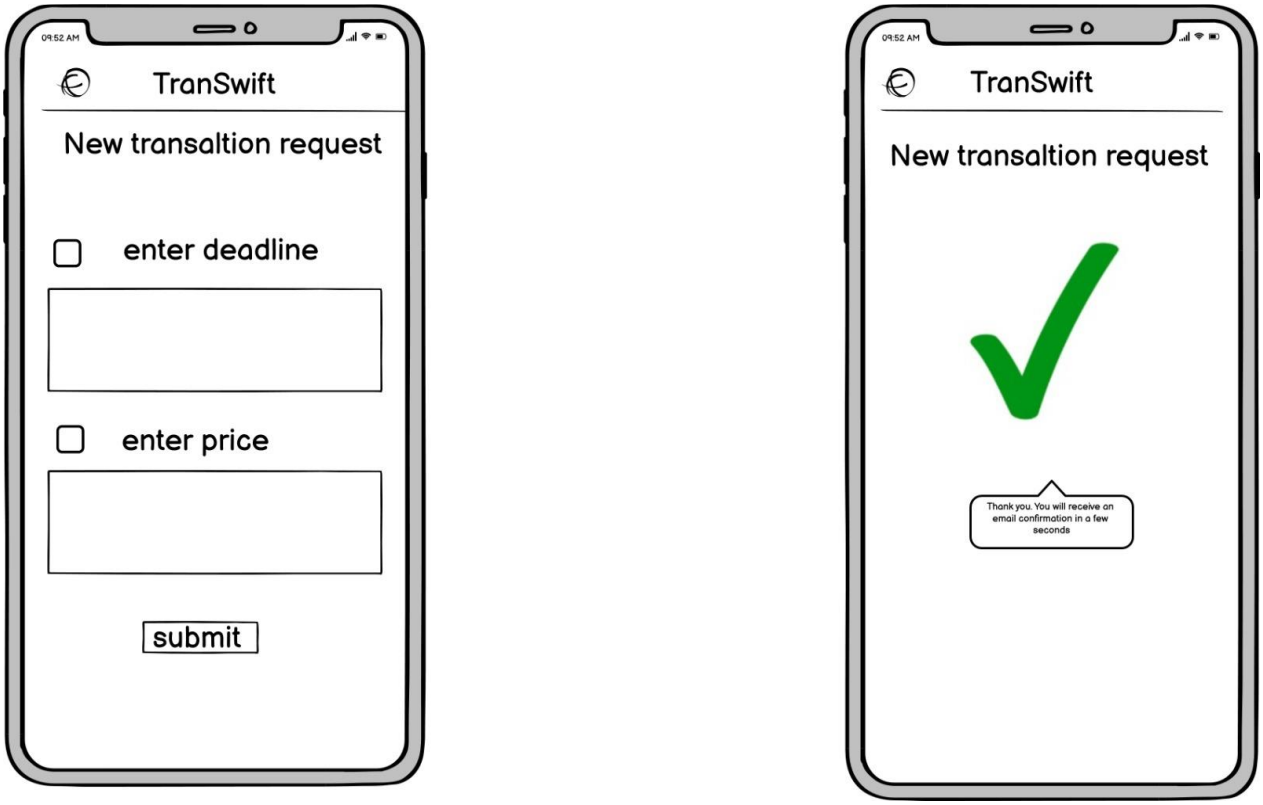


Table 36 Graphical User Interface "New Translation Request"

4.1.8 Requirement-„Translation request confirmation “

4.1.8.1 Use Case Specification

Name	**Translation requestconfirmation**
Description	this use case starts when a contractor gets a new request and ends when they answer the request. The contractor has the option to approve, decline or make a new offer
Actors	The customer, thecontractor
Pre-conditions	The contractor has received a new request.
Basic Flow **Description** **Actions** **1** **2**	 The contractor approves the request. The contractor opens the new request. The contractor approves the request.
Alternative Flow **Description** **Actions** **1** **2** **Alternative Flow** **Description** **Actions** **1** **2** **3**	A The contractor declines the request. The contractor opens the request. The contractor declines the request. B The contractor makes a new offer. The contractor makes new offer. The contractor chooses a new price and/or a new deadline and sends it to the customer The customer must option to accept or decline the new offer.
Post-conditions	the request has been approved or declined. The customer receives a correspondent notification.

Table 37 Use Case Specification "Translation request confirmation"

4.1.8.2 Use Case Diagram

This use case diagram describes the phase when a contractor gets a new request and shows if the contractor can approve the offer or decline it.

This is the phase where the contractor can accept or decline the request. Firstly, if the contractor receives the request, he must filter it. That mean he can accept the price that be from customer recommend and the type of the order. After all that the customer receive a message if his order was declined or accepted.

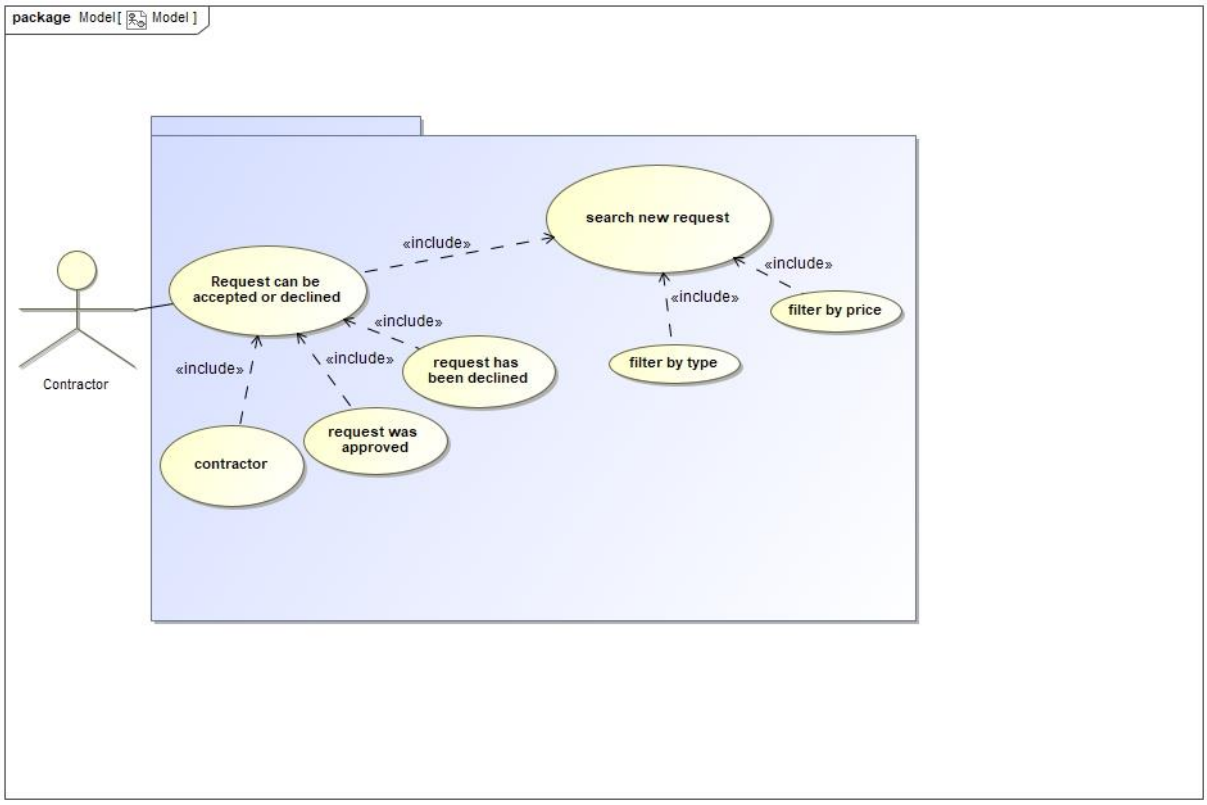
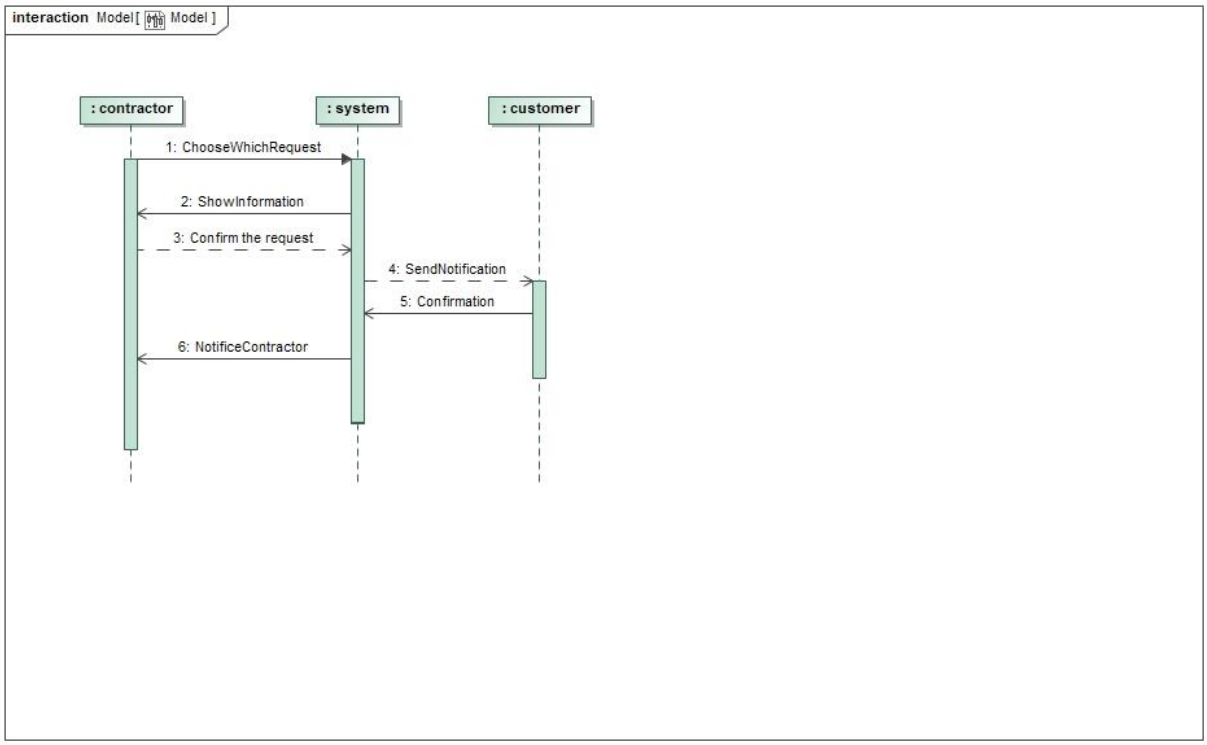


Table 38 Use Case Diagram "Translation request confirmation"

4.1.8.3 Sequence Diagram

The sequence diagram describes the steps when the contractor receives the request.

The contractor receives firstly the order from customer and then he shows information about the Order. Secondly, he declines or accepts the Order and he sends the confirmation to system and then the customer will receive the confirmation and he will notify the contractor.



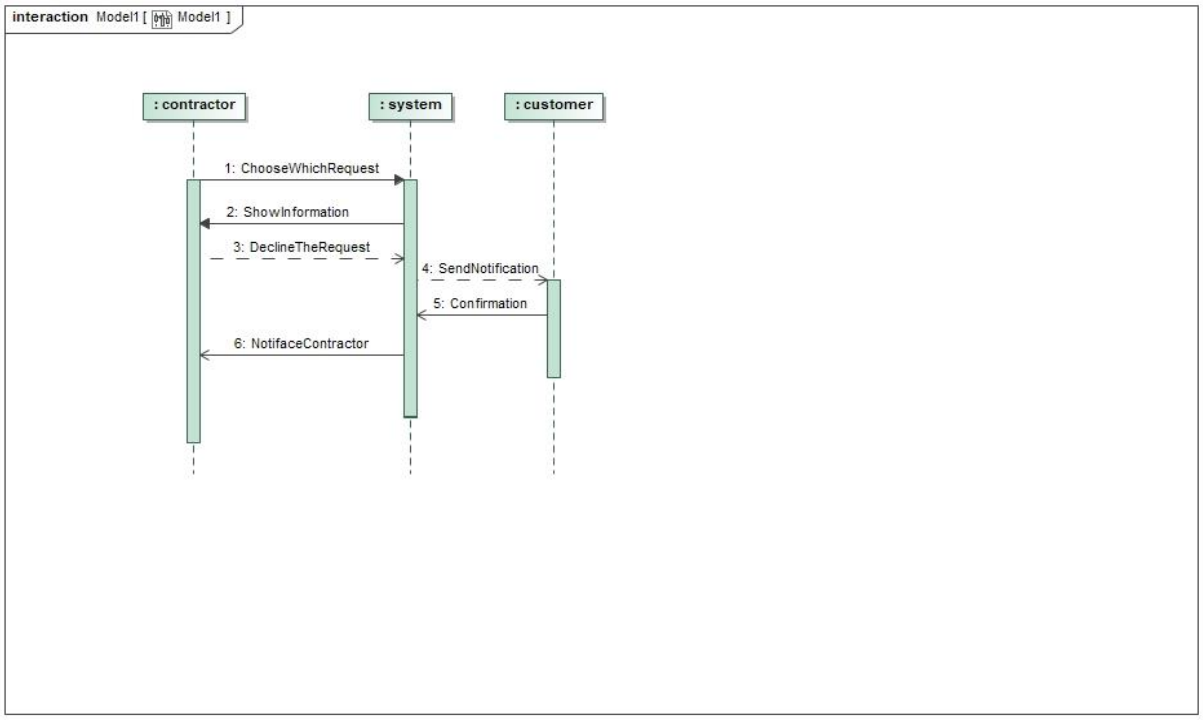


Table 39 Sequence Diagram "Translation request confirmation"

4.1.8.4 Activity Diagram

This activity diagram shows the process for a confirmation the request from the contractor and customer.

The customer needs firstly a new request, and he can also decline or accept the order or make a new offer to the customer. Secondly the contractor receives the order and systemwill stores the request. After that the system will notify the customer and then he gets a notification for the order.at end the customer gets the request from the contractor.

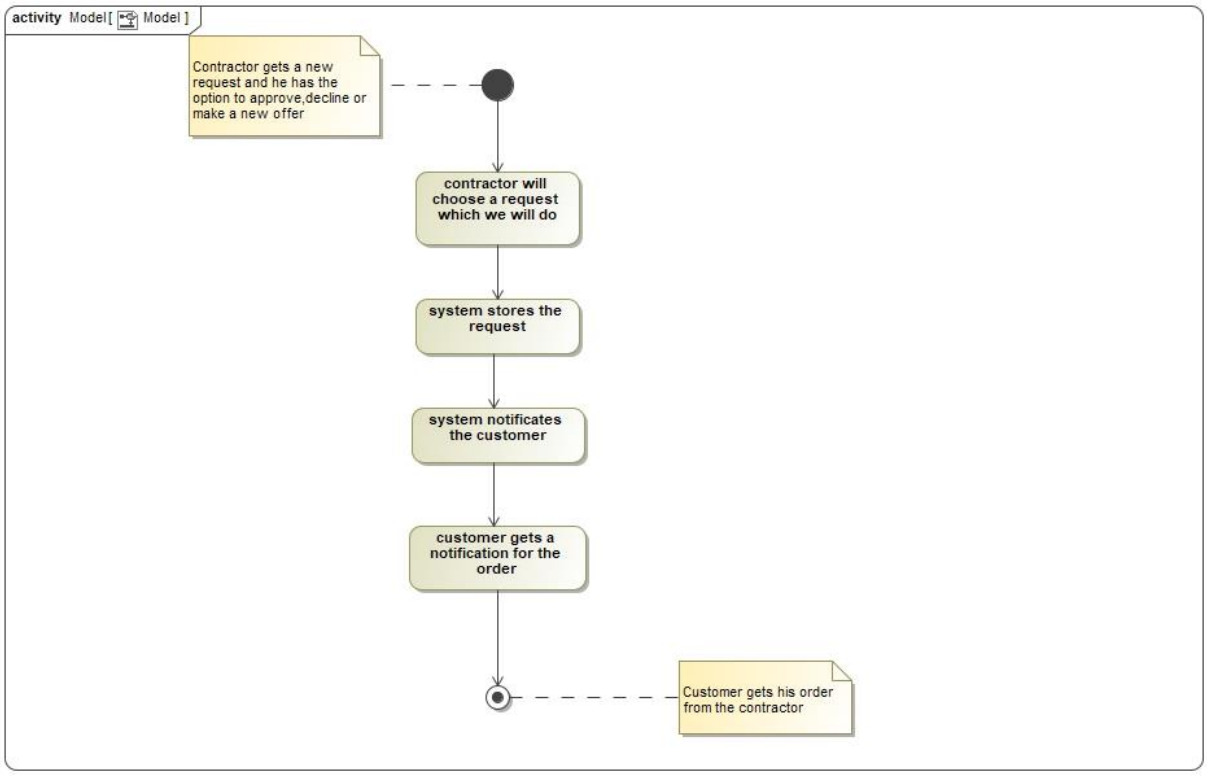


Table 40 Activity Diagram "Translation request confirmation"

4.1.8.5 Graphical User Interface

The graphical user interface shows how it looks when the customer receives his message about his request from the contractor.

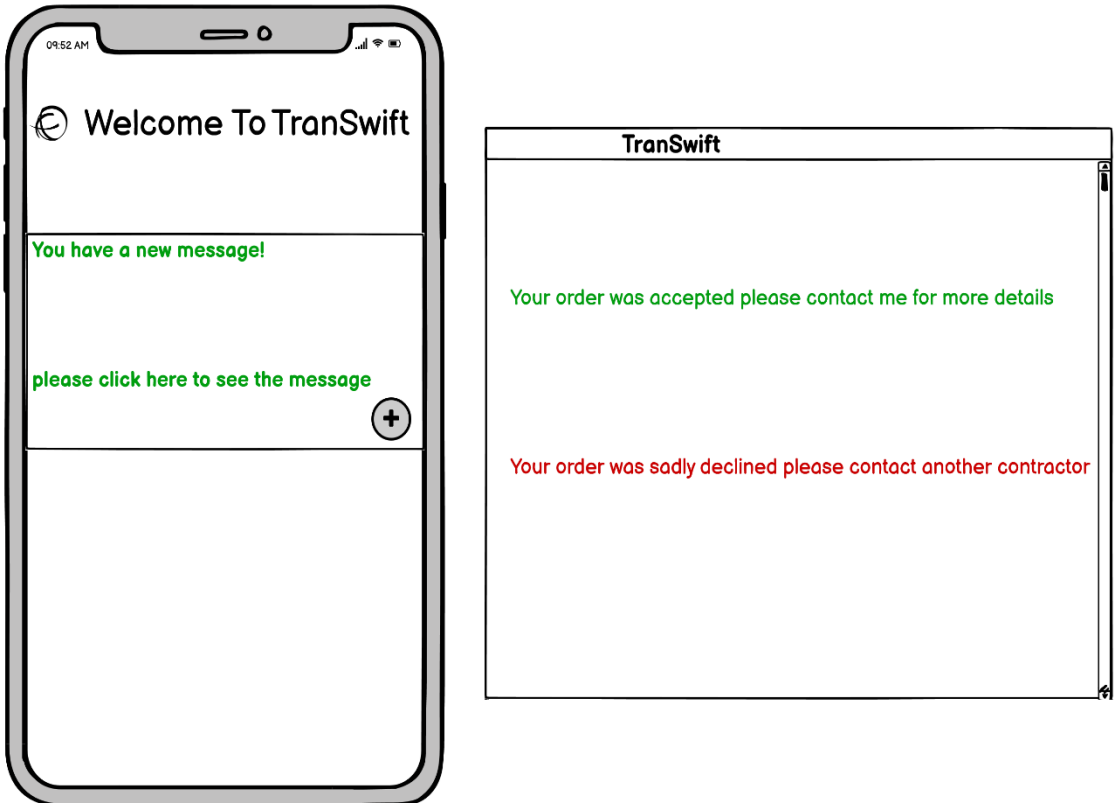


Table 41 Graphical User Interface "Translation request confirmation"

4.1.9 Requirement-„Tracking the translation process “

4.1.9.1 Use Case Specification

Use Case	**Tracking the translation process**
Description	Once the request has been approved, the status of the order changes to “approved” and a tracking process starts. The status is being updated every time a new change occurs. The tracking ends once the final result has been delivered and the status changes to “delivered”.
Actors	The customer, the contractor
Pre-conditions	The payment was successful.
Basic Flow	The customer tracks his order and gets notified every time a new update occurs. The status’s changes meet the agreed deadlines.
Description Actions	
1	The status changes to 20% accomplished once the contractor has finished the terminology research.
2	The status changes to 50% accomplished once the contractor has finished the proofreading.
3	The status changes to 70% accomplished once the contractor has finished the formatting.
4	The status changes to 80% accomplished once the contractor has finished verification.
5	The status changes to 100% accomplished once the contractor has delivered the final document.
Exception Flow	A the status didn’t change in the estimated process time.
Description Actions	
1	The contractor notifies the customer about the delay and offers a new deadline.
2	The customer agrees to the new deadline.
3	The support contacts the contractor.
Alternative Flow of Exception Flow A.2	A.2.a The customer declines the new deadline and wants to cancel the order and get his money back.
Description Actions	
1	The support gets in contact with the customer.
2	The support contacts the contractor.
Post-conditions	The translated document has been delivered in the agreed deadline.

Table 42 Use Case Specification "Tracking the translation process"

4.1.9.2 Use Case Diagram

This Use Case Diagram summarizes the steps of Tracking the translation process.
Using a tracking id, the customer has the ability to see the actual status of the order and get notified after every change in the status.

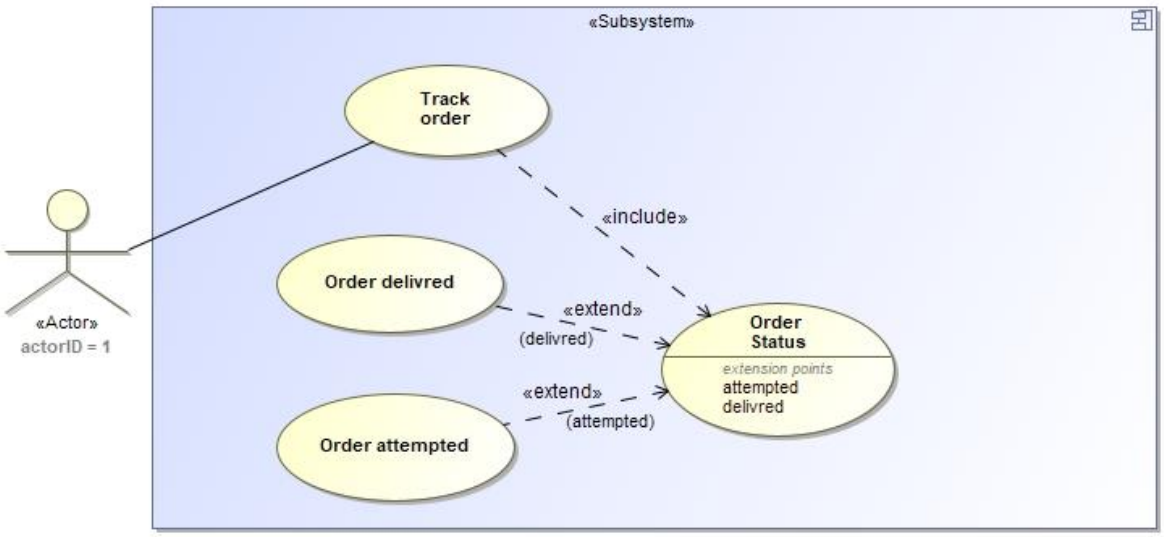


Table 43 Use Case Diagram "Tracking the translation process"

4.1.9.3 Sequence Diagram

This sequence diagram illustrates that the customer enters the tracking ID on the website and requests the status of the order. The website asks our system about the status. The status automatically tracks the contractor again and receives the actual status. This actual status will be refreshed on the website, where the customer can see the news of his order. If the task would been not accomplished in estimated time. The support would been notified. A prolongation of the task time can be if the customer agrees the contractor's offer.

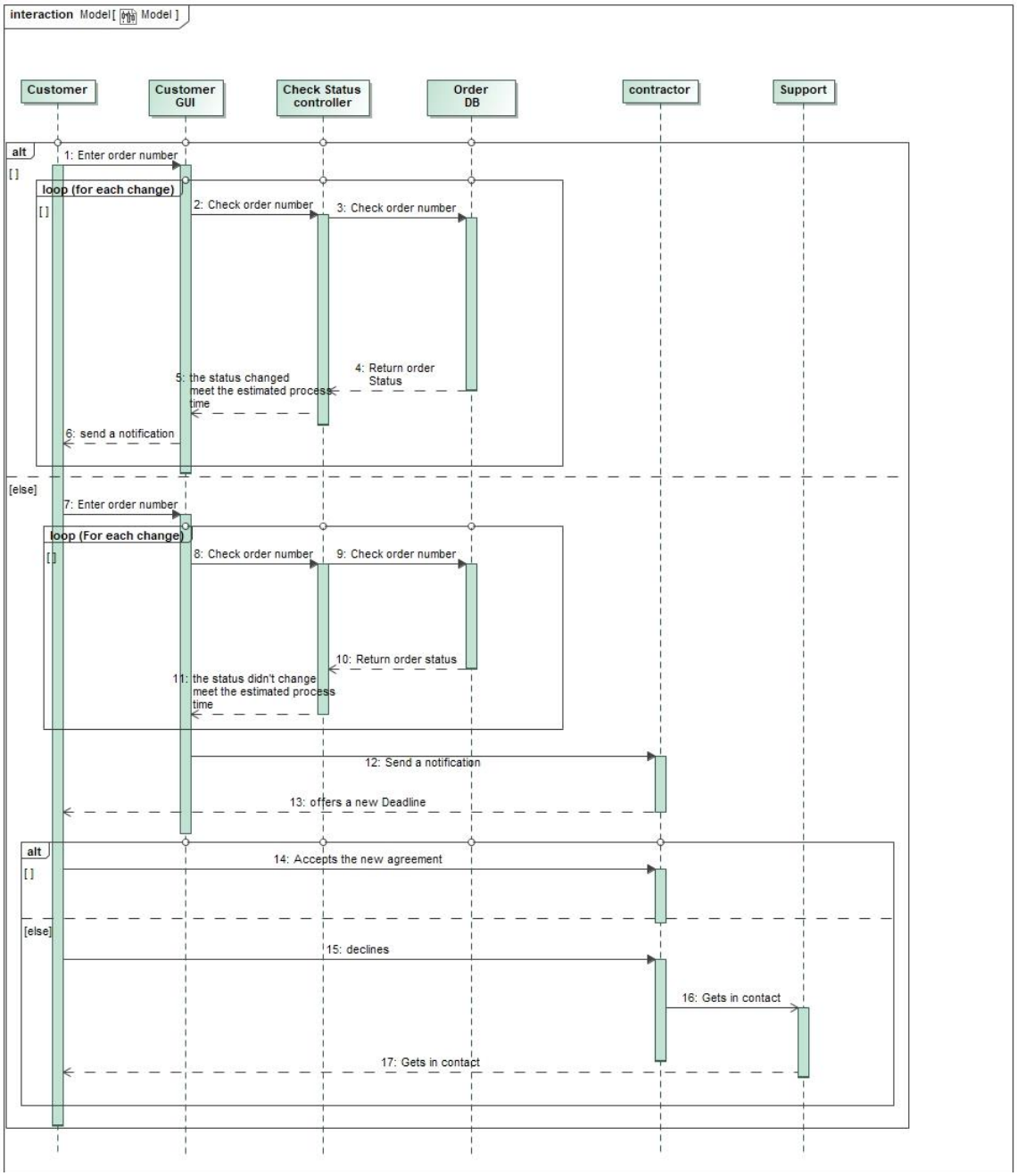


Table 44 Sequence Diagram "Tracking the translation process"

4.1.9.4 Activity Diagram

This activity diagram shows the steps how it works to track an order. First, the customer checks the order by inserting an Id tracking and ends by notifying him that the task has been accomplished successfully.

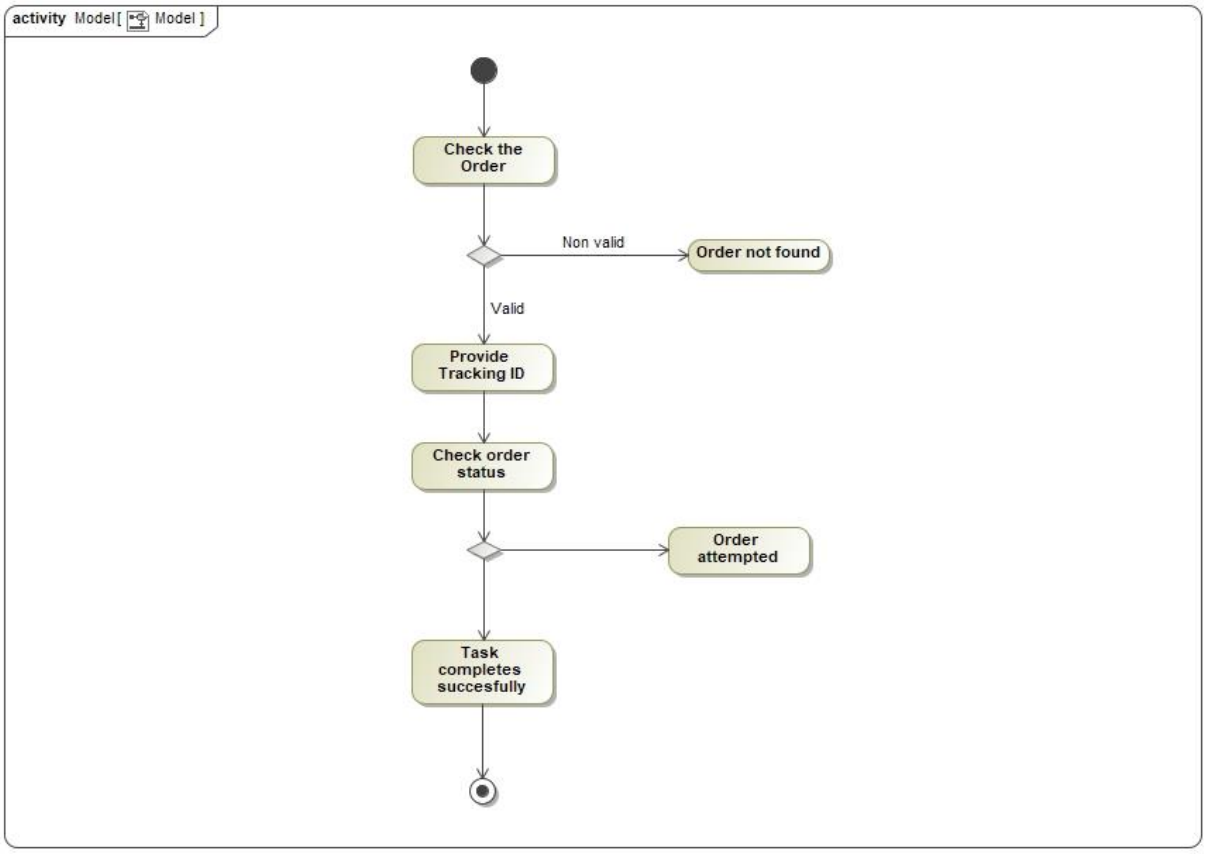


Table 45Activity Diagram "Tracking the translation process"

4.1.9.5 Graphical User Interface

This UI Prototype informs the customer by every change in the process of the translation made by the contractor and estimates the delivery time. furthermore, it offers the customer another options. e.g., "contact support, help, turn on notification".

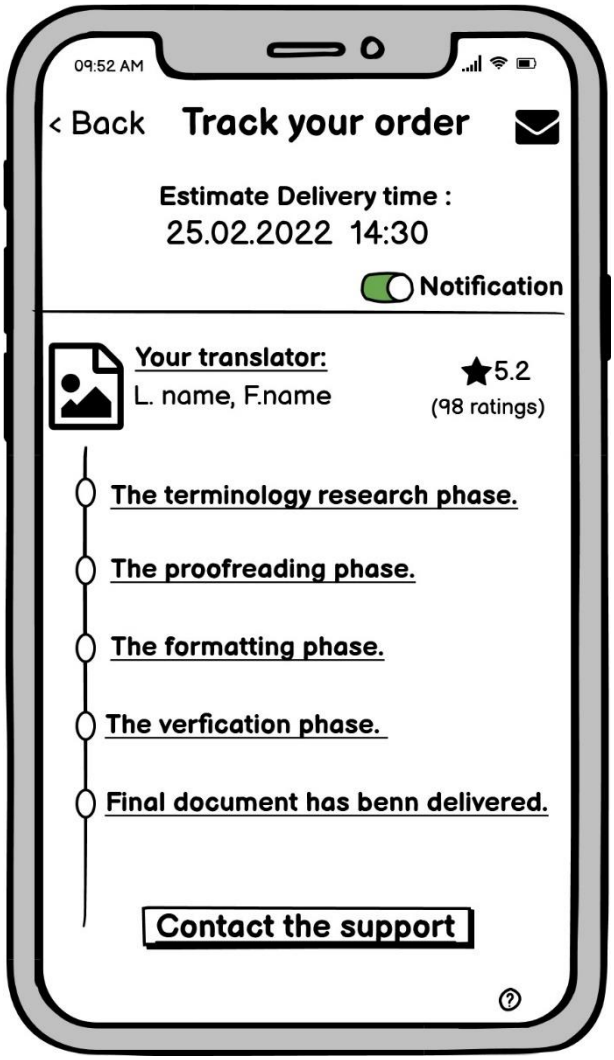


Table 46 Graphical User Interface "Tracking the translation process"

4.1.10 Requirement-„Translation document delivery “

4.1.10.1 Use Case Specification

Use Case	Translation document delivery
Brief description	start: a Translator has agreed on translating a document. End: The Customer receives the translated document. The Contractor delivers/uploads the translated document to the customer
Actors	Contractor & customer
Pre-conditions	
1	Customer request accepted
2	Not over the deadline / agreed to the new deadline (if needed)
Basic Flow	delivery process: (The customer uploads the translated document)
1	opens request/order
2	uploads document
3	customer gets notified
Exception Flows	
	deadline has passed:
1	Contractor gets notification
2	must make a new request of the deadline
2.1	customer has the agrees
2.2	customer doesn't agree
Post Conditions	Contractor gets a rating from the Customer

Table 47 Use Case Specification "Translation document delivery"

4.1.10.2 Use Case Diagram

This Use Case Diagram for the use case “translation document delivery” summarizes the steps of Contractors delivery of the translated document. The Contractor can upload the Document to the server and gets a confirmation. The Customer can then view it.

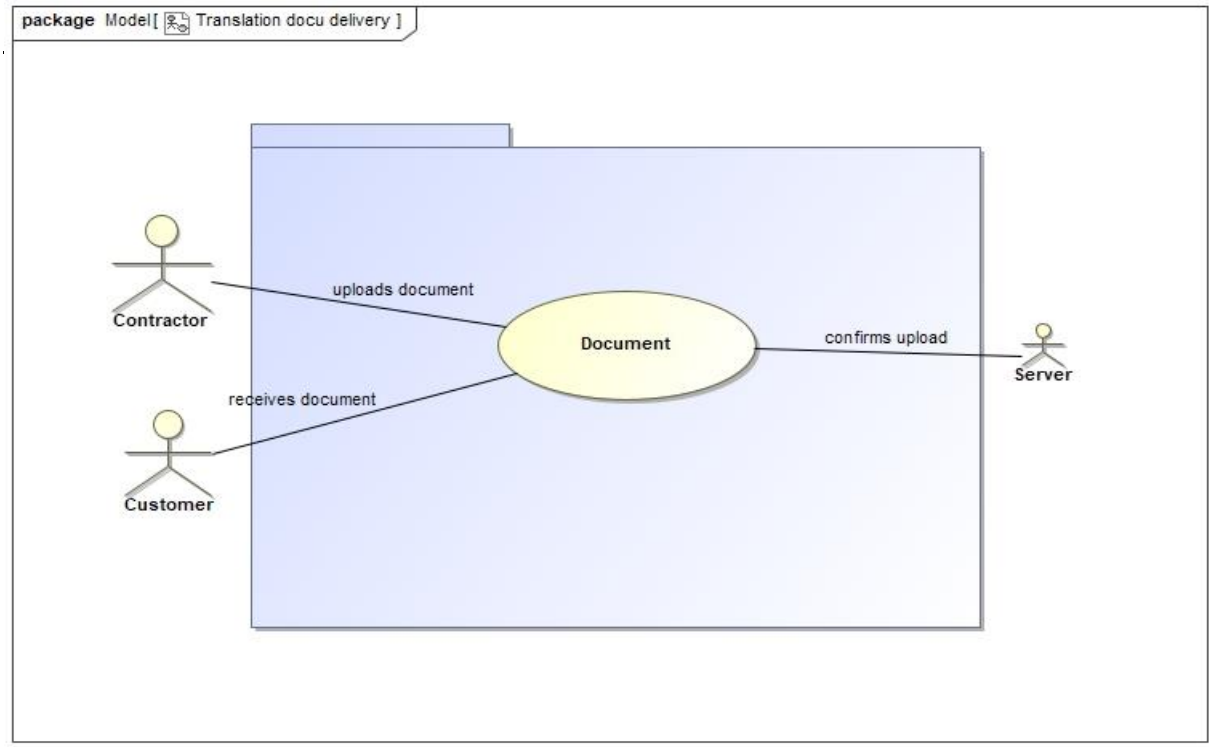


Table 48 Use Case Diagram "Translation document delivery"

4.1.10.3 Sequence Diagram

This Sequence Diagram shows the delivery process in more detail.

It starts when the Contractor clicks on the order. And ends either with an error, if the deadline or the correct format of the document aren't met, or with a successfully uploaded document and a notified customer.

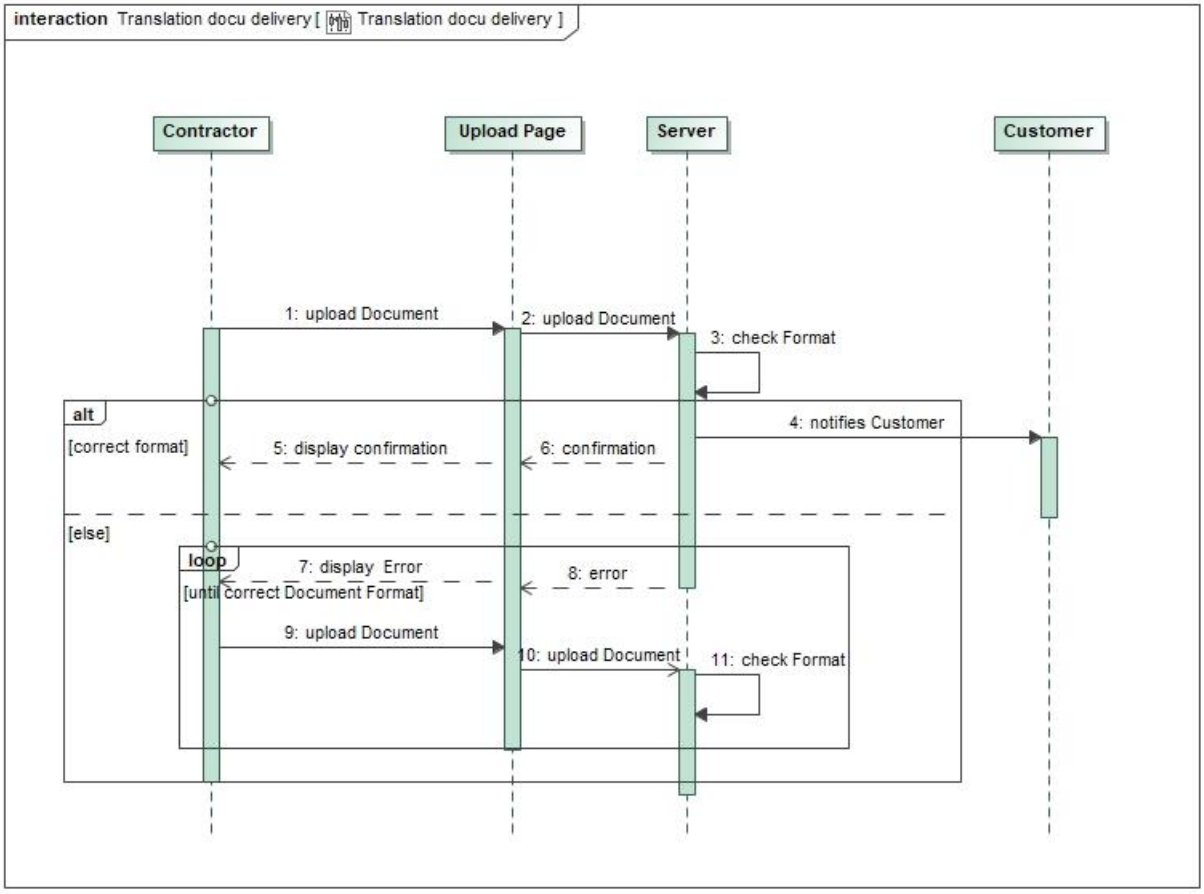


Table 49 Sequence Diagram "Translation document delivery"

4.1.10.4 Activity Diagram

The Contractor has already accepted the order and is now finished with the translation and wants to upload it. This Activity Diagram shows how the Contractor uploads the translated document.

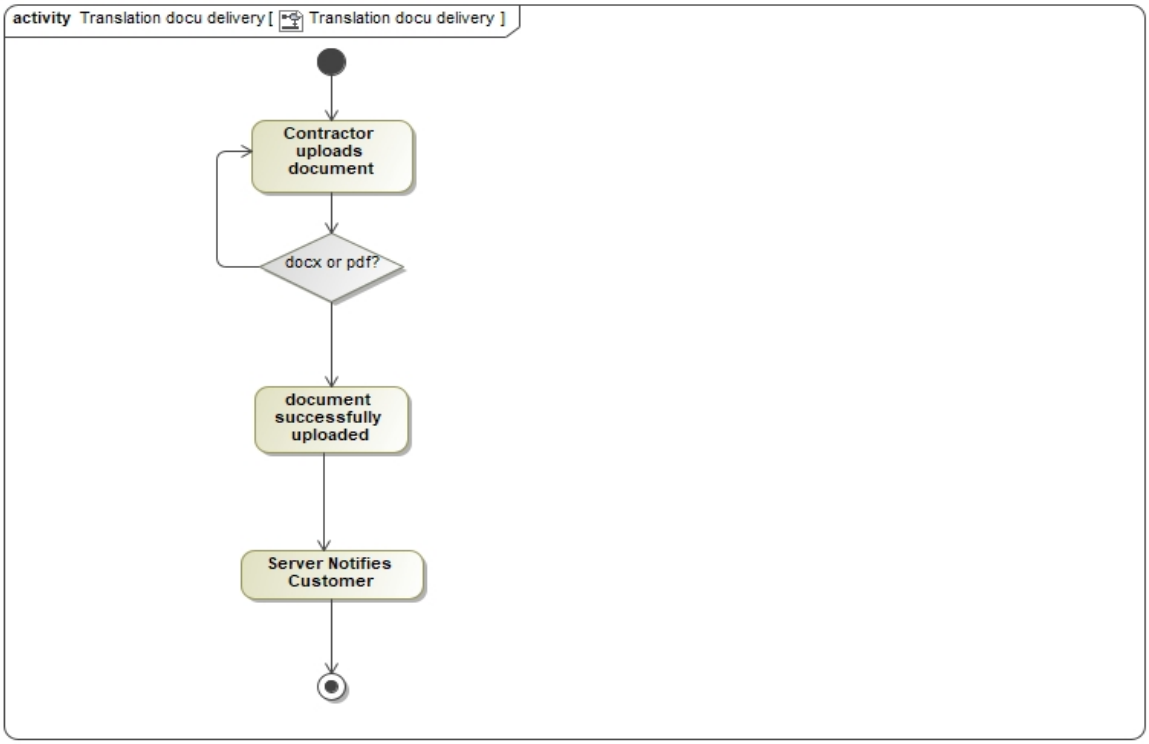


Table 50 Activity Diagram "Translation document delivery"

4.1.10.5 Graphical User Interface

This UI Prototype shows the Application page where the Contractor uploads the Document and can leave a comment for the Customer.

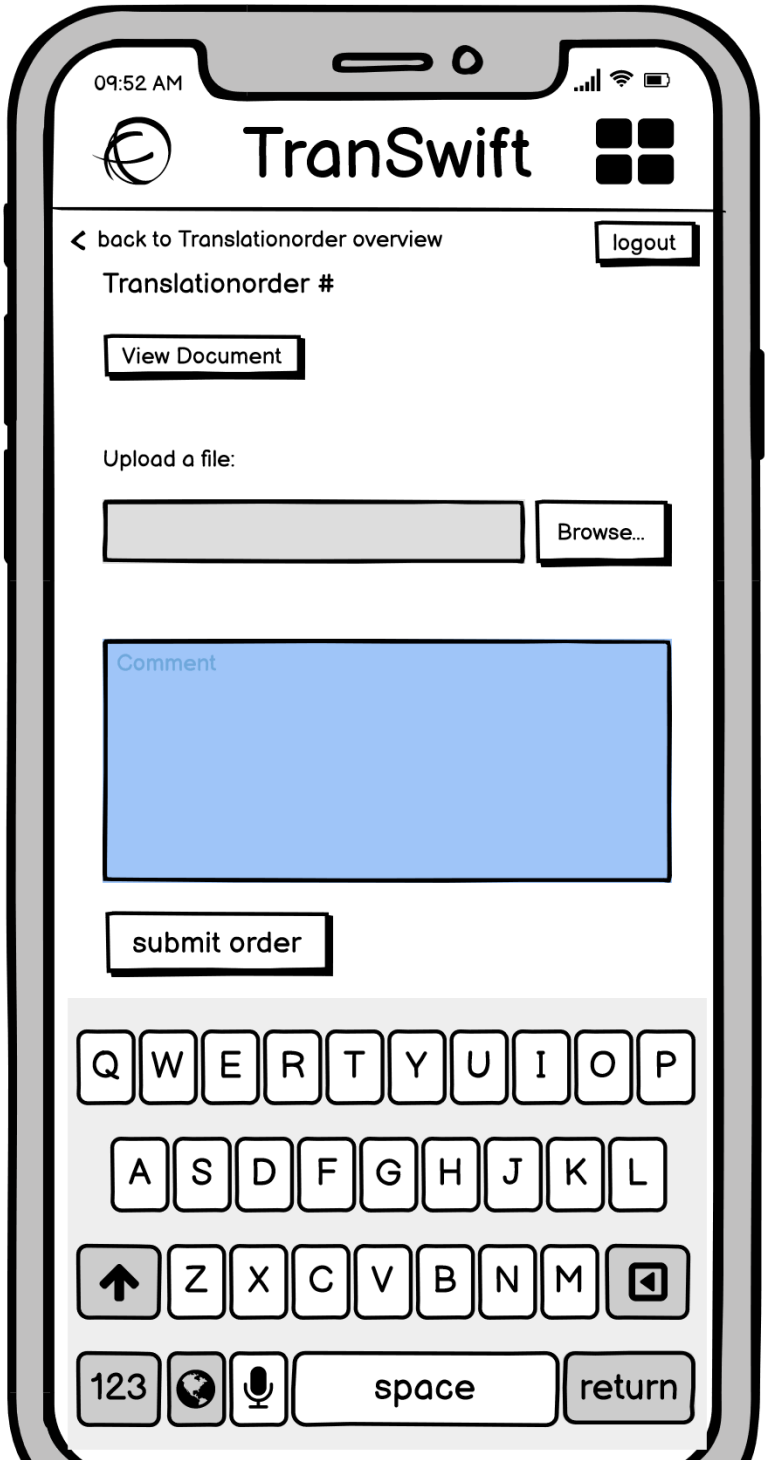


Table 51 Graphical User Interface "Translation document delivery"

4.1.11 Requirement-„Payment methods “

4.1.11.1 Use Case Specification

Name	Payment methods
Description	the case starts when the customer wants to pay and ends once the system gets access to the customer account. The customer chooses their preferred payment method and enters their data.
Actors	The customer
Pre-conditions	the customer must have a card or an application allowing him to pay online.
Basic Flow Description Actions 1 2 3 Exception Flows 1 2	 when the customer wants to pay, several payment options show up: the customer chooses the option that suits him. The customer enters his information. The customer confirms his payment. the Customer cancels the payment. the customer changes the payment method or tries again.
Post-conditions	“Payment was successful” appears on the platform after the customer has paid. the customer also receives a successful payment email.

Table 52 Use Case Specification "Payment methods"

4.1.11.2 Use Case Diagram

This use case diagram describes the use case of the payment methods. The customer can choose their preferred external payment method, such as PayPal, credit card and debit card.

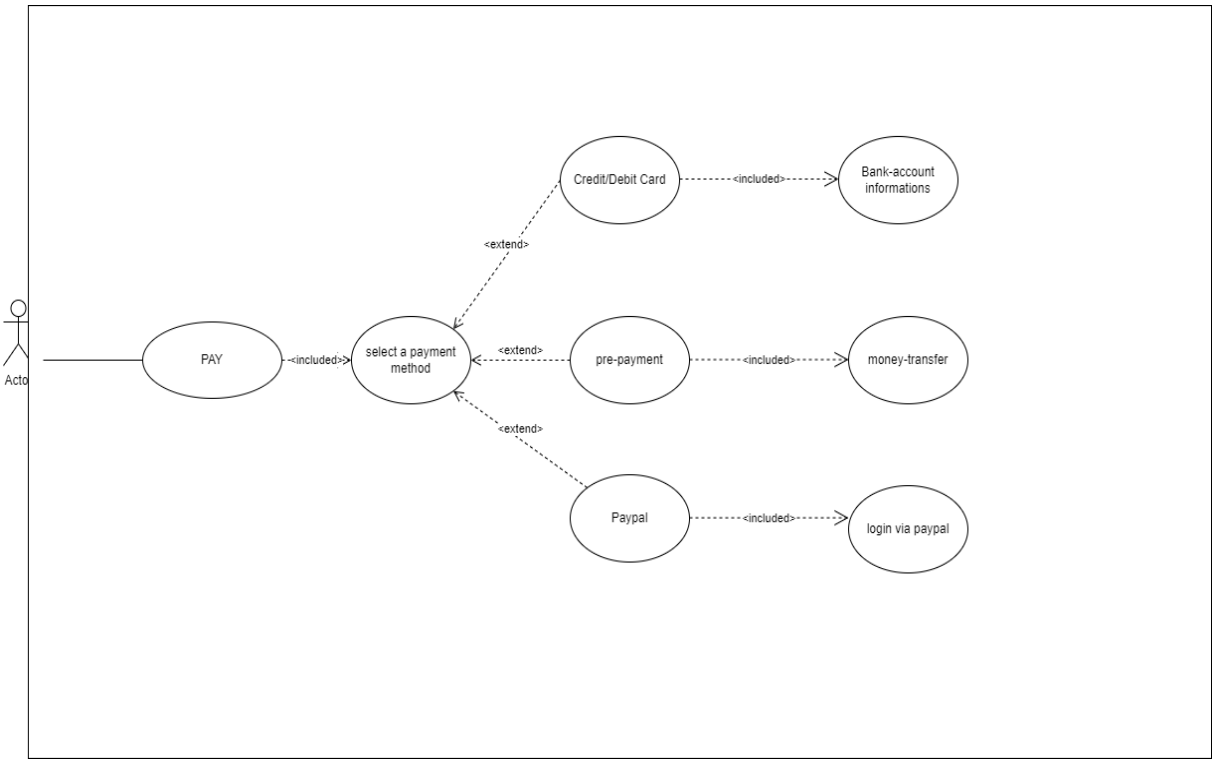


Table 53 Use Case Diagram "Payment methods"

4.1.11.3 Sequence Diagram

This sequence diagram of the payment methods explains the interaction of payment information of the user. The extern service provider will verify this information.

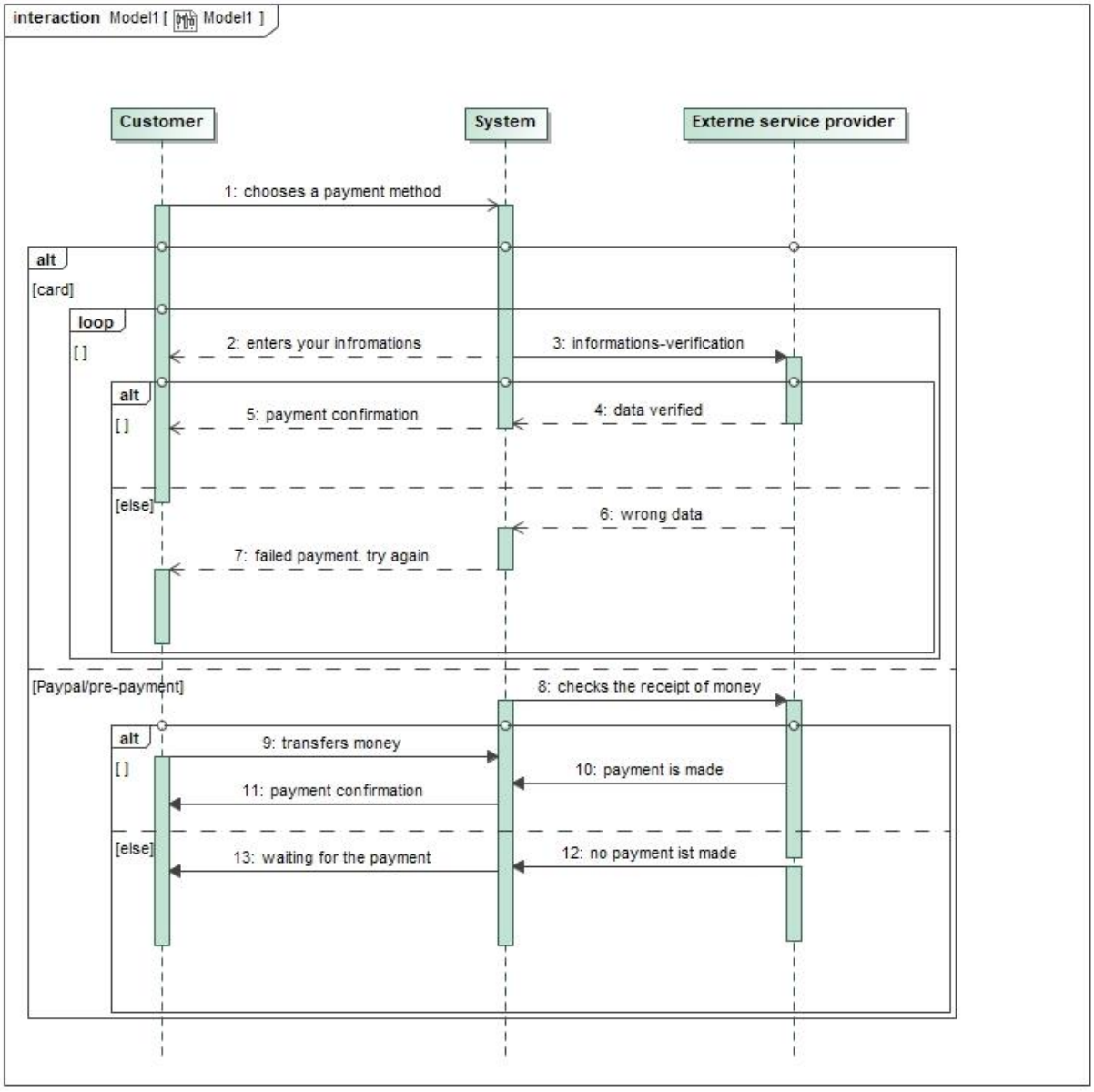


Table 54 Sequence Diagram "Payment methods"

4.1.11.4 Activity Diagram

The customer chooses his preferred method of payment, and enters his information. Once the information is wrong, the customer must give the correct data again to make the payment.

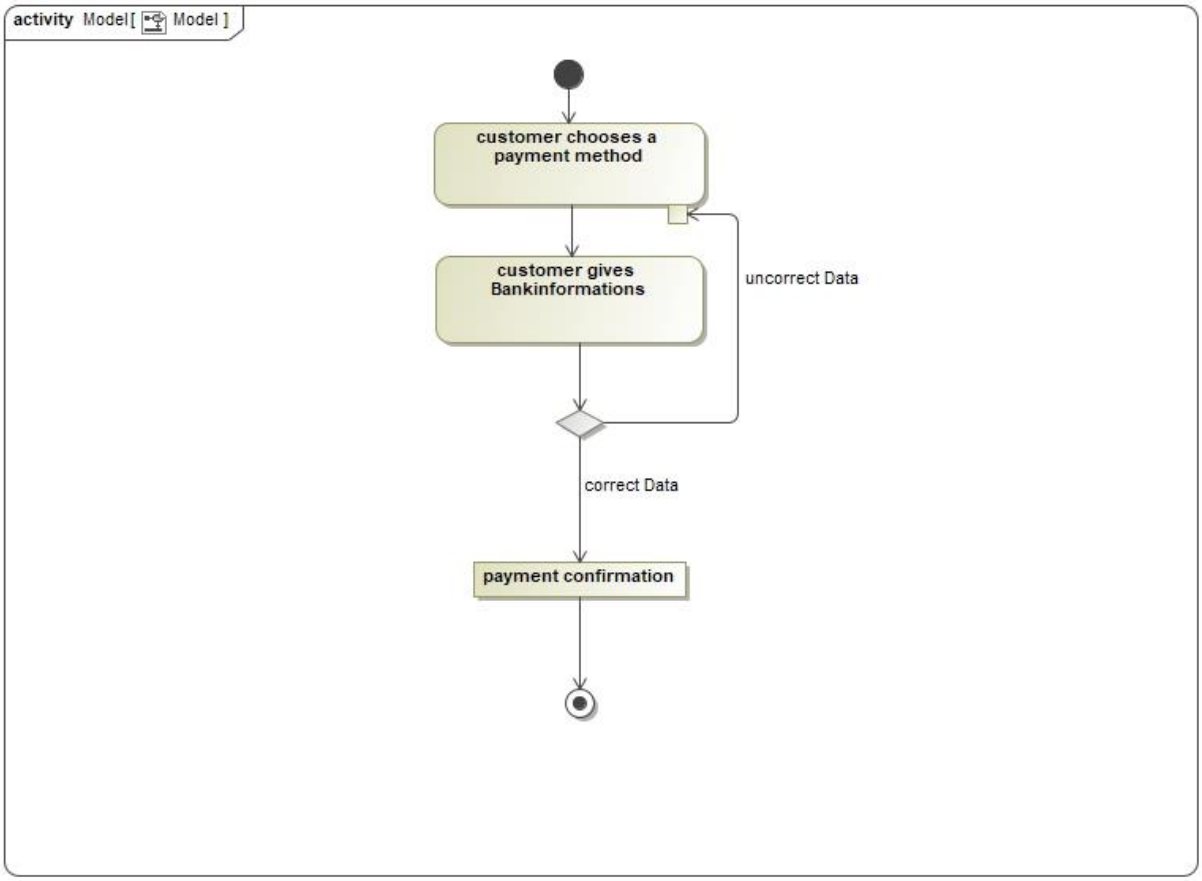


Table 55 Activity Diagram "Payment methods"

4.1.11.5 Graphical User Interface

the User interface shows the steps how a customer can pay on our app.

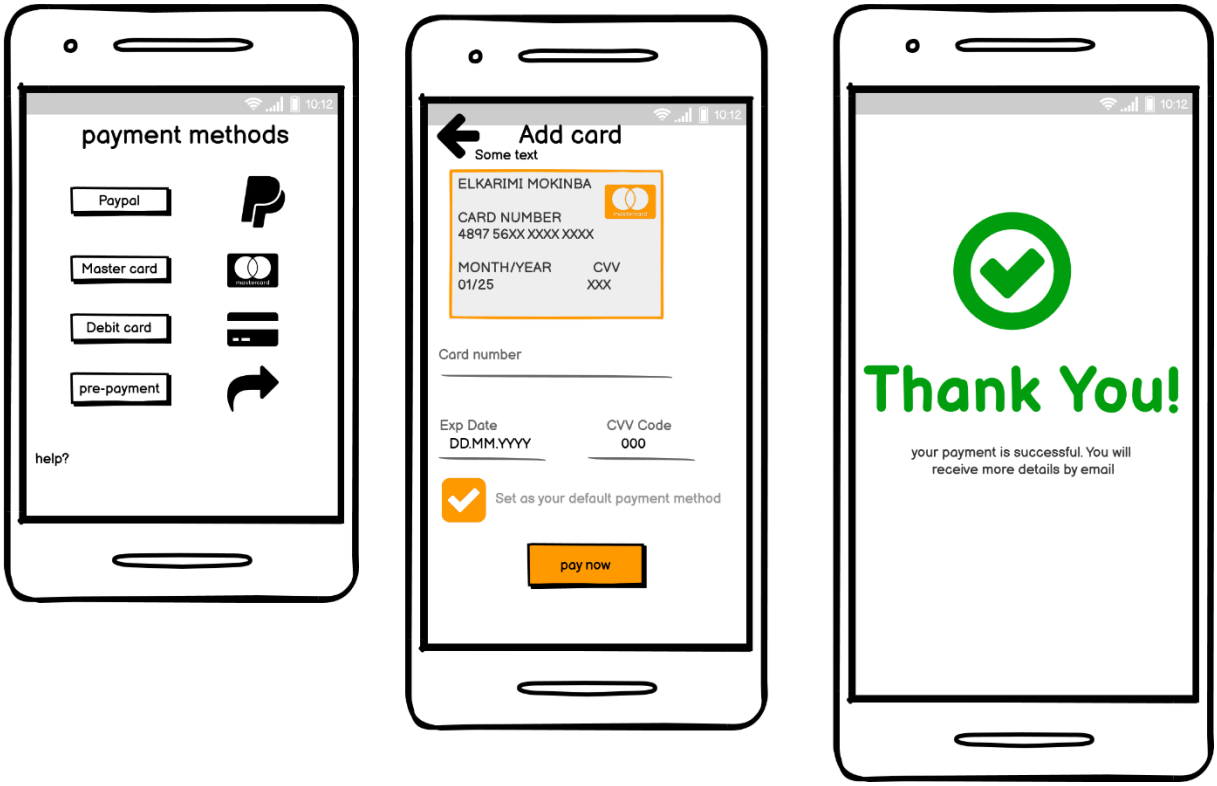


Table 56 Graphical User Interface "Payment methods"

4.1.12 Requirement-„Contractor Rating “

4.1.12.1 Use Case Specification

Table 57 Use Case Specification "Contractor Rating"

Name	**Contractor Rating**
Description	This use case starts once the agreed deadline has expired regardless the completion of the translation from the contractors' side and ends once the customer submits his rating/review. The customer rates the translation and the whole experience with the contractor in a scale of 1-5 based on certain criteria and has also the ability to write a review.
Actors	Customer
Pre-conditions	The deadline of the translation has been expired and the option of rating is enabled.
Basic Flow **Description** **Actions** **1** **2** **3** **4**	 This is the main scenario where the customer rates a contractor and posts his rating and submits it to be posted to the public. The customer confirms reading the terms of use. The customer rates the contractor. The customer writes a review. the customer submits the rating and the review.
Alternative Flow **Description**	A The customer writes a review that violates the Terms of Use.
Alternative Flow **Description** **Actions** **1** **2** **3**	A.1 The review is reported automatically to the support to be reviewed. The customer is notified and asked kindly to be patient till the review is checked. The support deletes the review if it violates the terms of use and notifies the customer. The support posts the review if it doesn't violate the terms of use.
Alternative Flow **Description** **Actions** **1** **2** **3**	A.2 The review has been posted and is reported by the contractor or other users The support checks the review The support deletes the review if it violates the terms of use and notifies the customer. The support keeps the review if it doesn't violate the terms of use and notifies the reporter.
Post-conditions	The rating and the review are successfully posted and can be seen by the public.

4.1.12.2 Use Case Diagram

This diagram shows the use case of contractor rating. The main actor here is the customer. After choosing to rate a translator, the use case includes giving 1 to 5 stars rating which will appear on the translator profile.

Additionally, the customer can write a review which can be seen by the public on the translator profile.

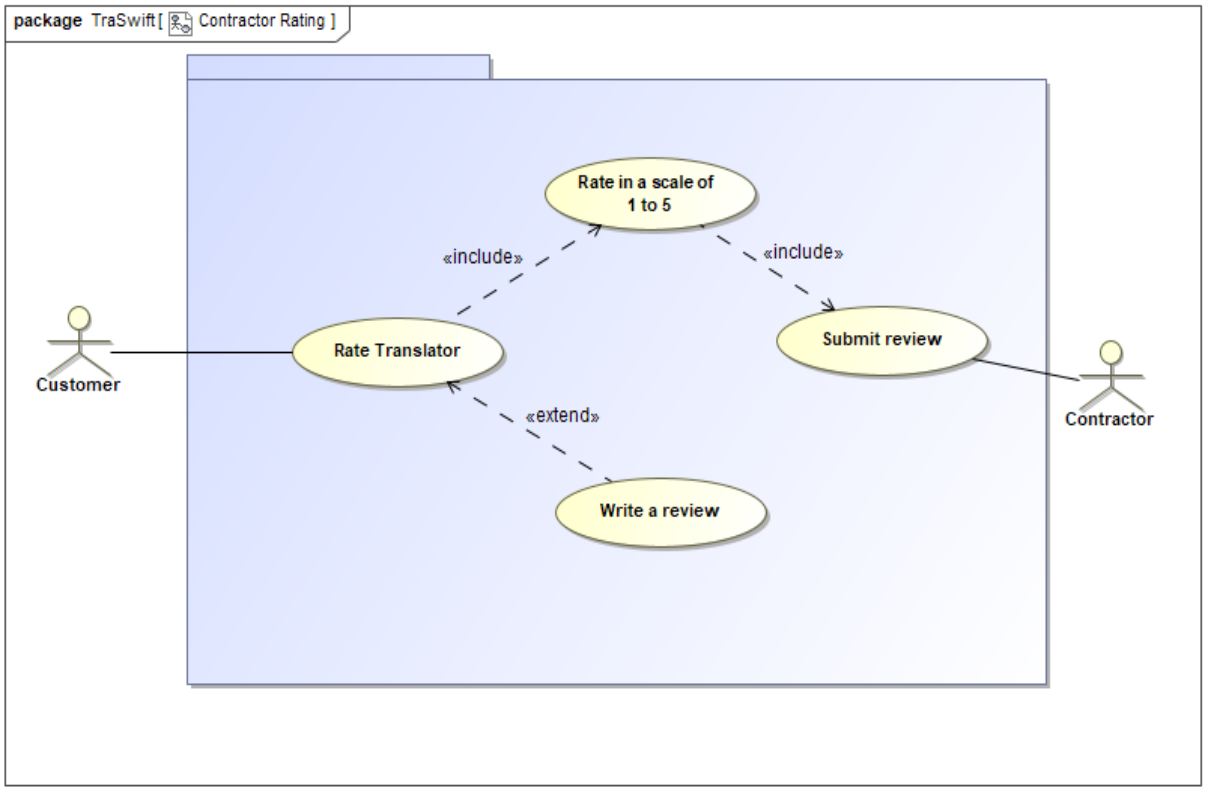


Table 58 Use Case Diagram "Contractor Rating"

4.1.12.3 Sequence Diagram

This diagram shows a sequence of reactions between the customer, the rating page and the server. After choosing to rate a translator, the server checks first if there was a transaction between the two actors, if yes, the customer chooses a number of stars according to his satisfaction with the translator's work, and then has the option to add a review or directly submit the rating. The rating will be then posted to the translator's profile.

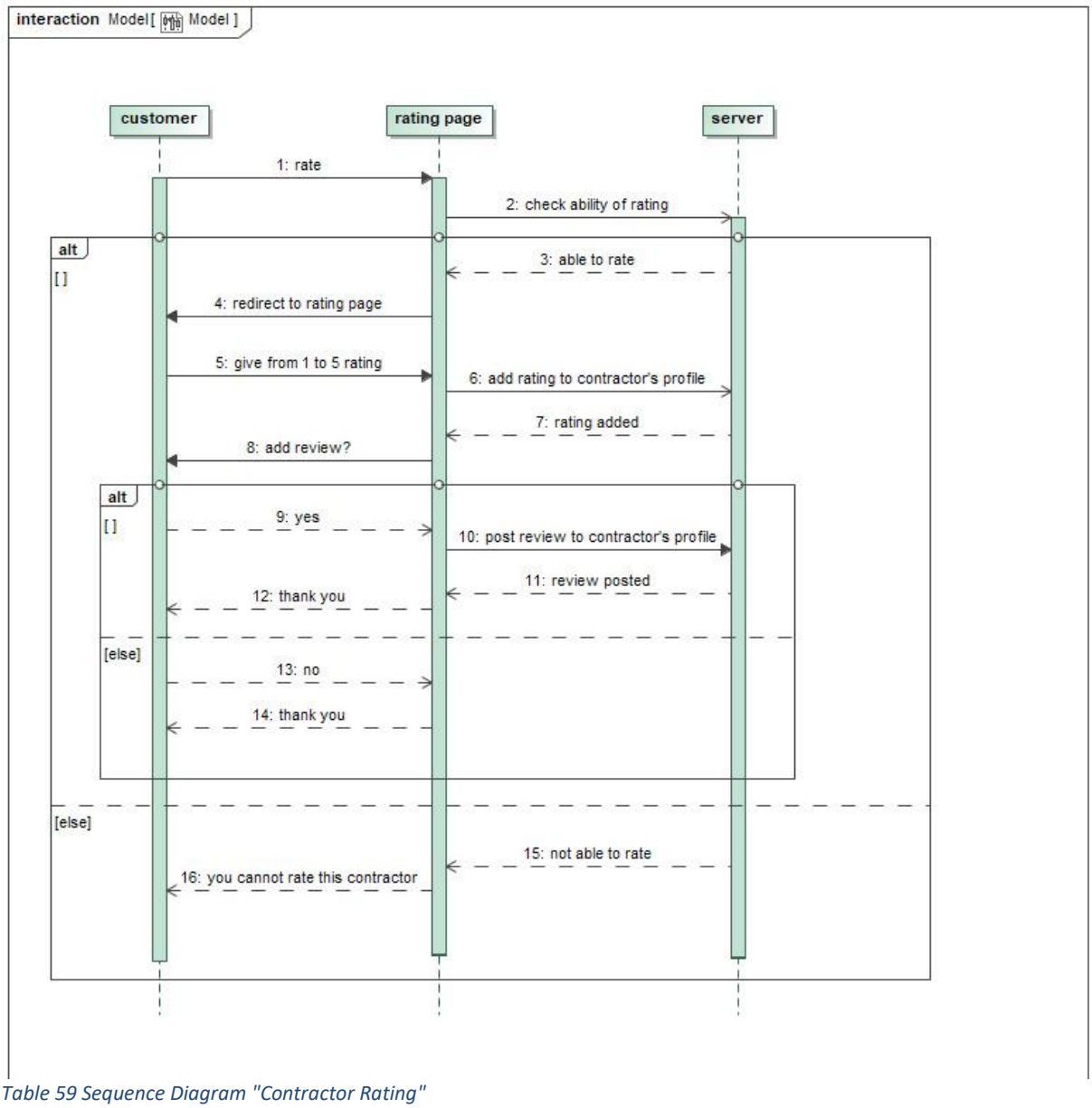


Table 59 Sequence Diagram "Contractor Rating"

4.1.12.4 Activity Diagram

This diagram shows the activities performed by the customer and the system during the use case of translator rating.

After confirming receiving the document, the system asks the translator if he would like to rate the translator, if no, the session ends immediately. If yes, the customer rates the translator in a scale of 1-5 stars and then the system asks again the customer if he would like to add a comment, and then he submits his rating.

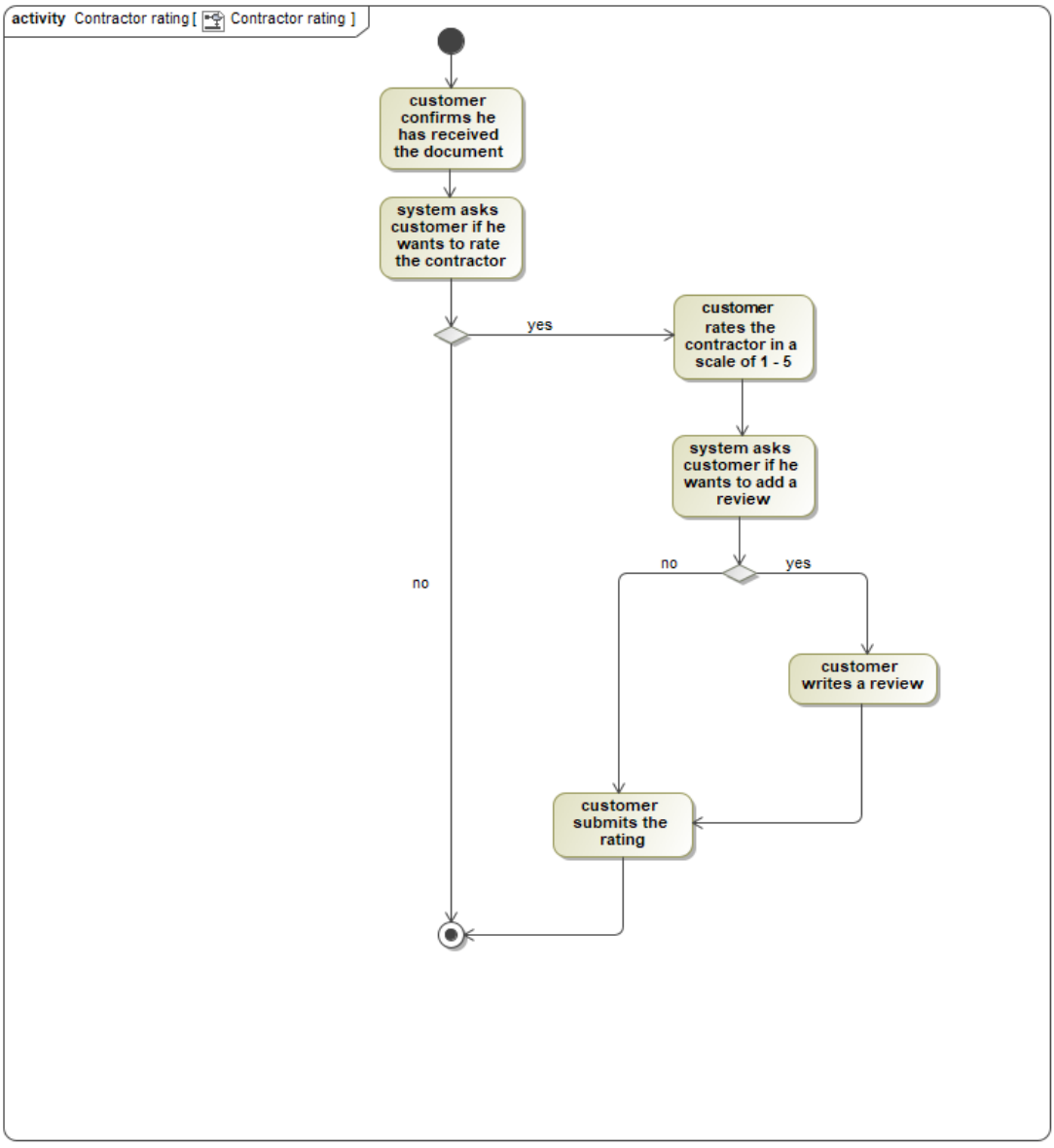


Table 60 Activity Diagram "Contractor Rating"

4.1.12.5 User Interface

This is the graphical user interface of the translator rating page. The customer has to give first an overall rating in scale of 1 to 5 and then an optional section to add a comment will show up the button to submit the feedback as well.

After clicking the button “Submit”, the numerical value of the given number of stars will be added to the average rating score of the translator and if a comment was written it will appear in the translator profile as review.

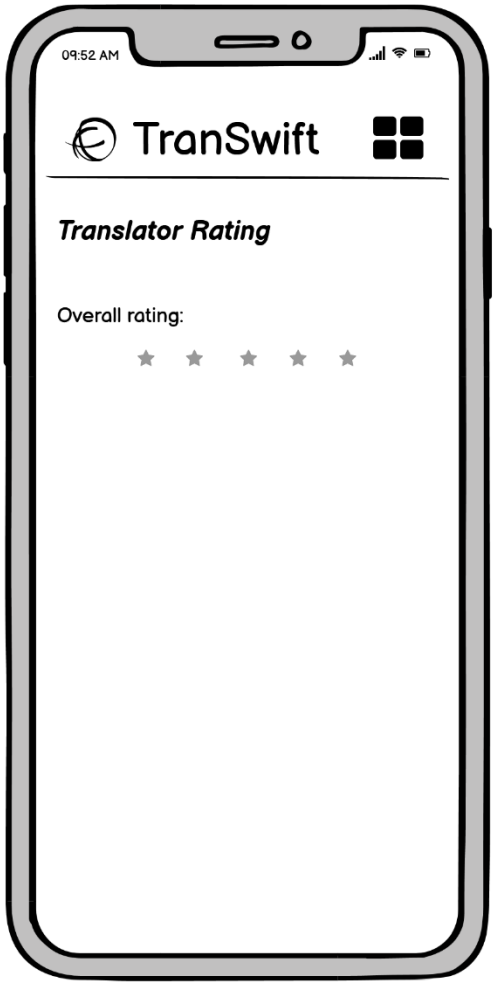


Table 61Graphical User Interface "Contractor Rating"

4.1.13 Requirement-„Customer Service “

4.1.13.1 Use Case Specification

Name	Customer Service
Descriptio n	the case starts when the customer has a question or a problem and ends once the customer gets an answer or solves the problem. The customer Service has a page where the customer can find answers to FAQ. He ‘ll also find answers to many topics, including the payment options available.
Actors	support team, The customer, the contractor
Pre-conditions	a customer or a contractor has a problem.
Basic Flow **Description** **Actions** **1** **2** **3**	The user has a common question that already exists in the FAQ page. the user goes to the FAQ page. the user chooses the correspondent topic to his question. the user gets an answer.
Alternative Flow **1** **2** **3** **4**	The user cannot find an answer to his question or is not satisfied with the answer provided in the FAQ. The user asks for help. The support team gets in contact with user. The support team tries to find a solution. The support team provides the best possible solution.
Post-conditions	the user gets the best possible help.

Table 62 Use Case Specification "Customer Service"

4.1.13.2 Use Case Diagram

This use case diagram “Customer service” shows the interaction between customer and management. Customers can write about a problem & ask for support.

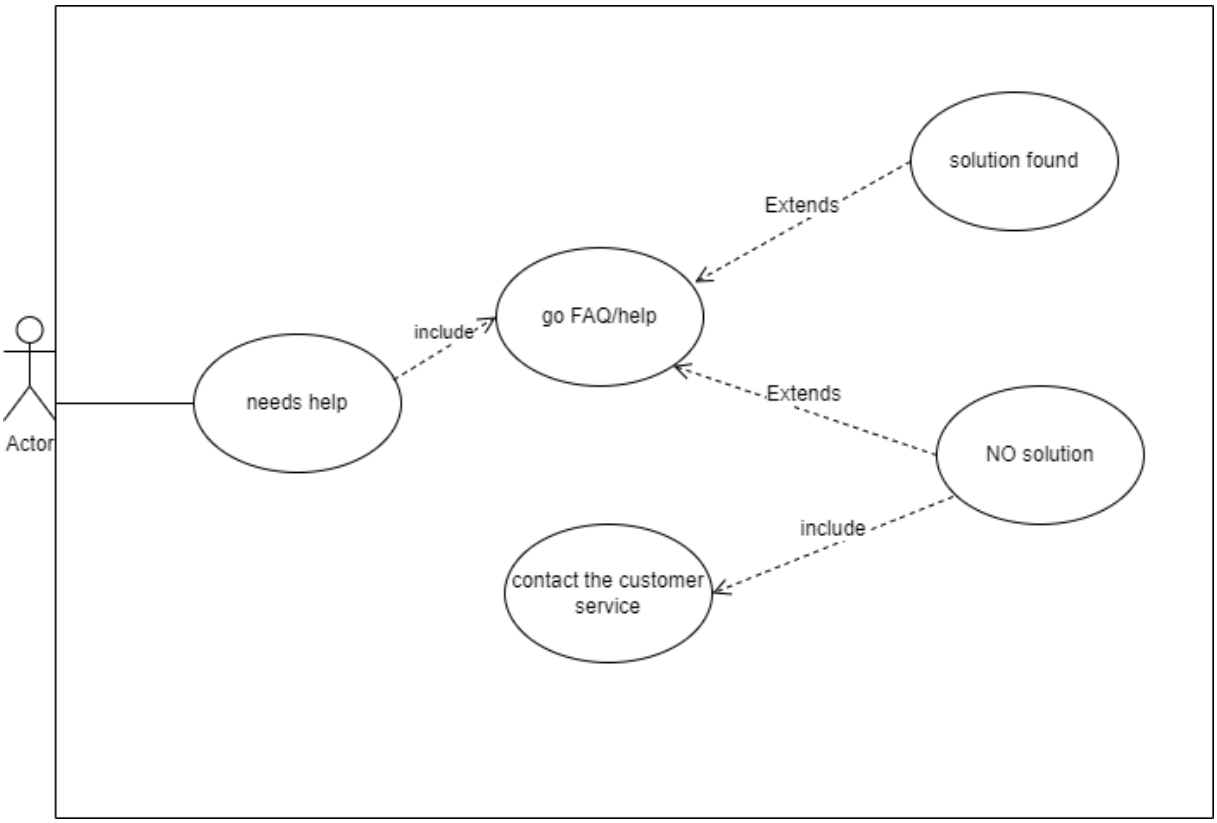


Table 63 Use Case Diagram "Customer Service"

4.1.13.3 Sequence Diagram

This sequence diagram shows the interaction of customer. The customer can get an answer from the suggestion or some help from the support.

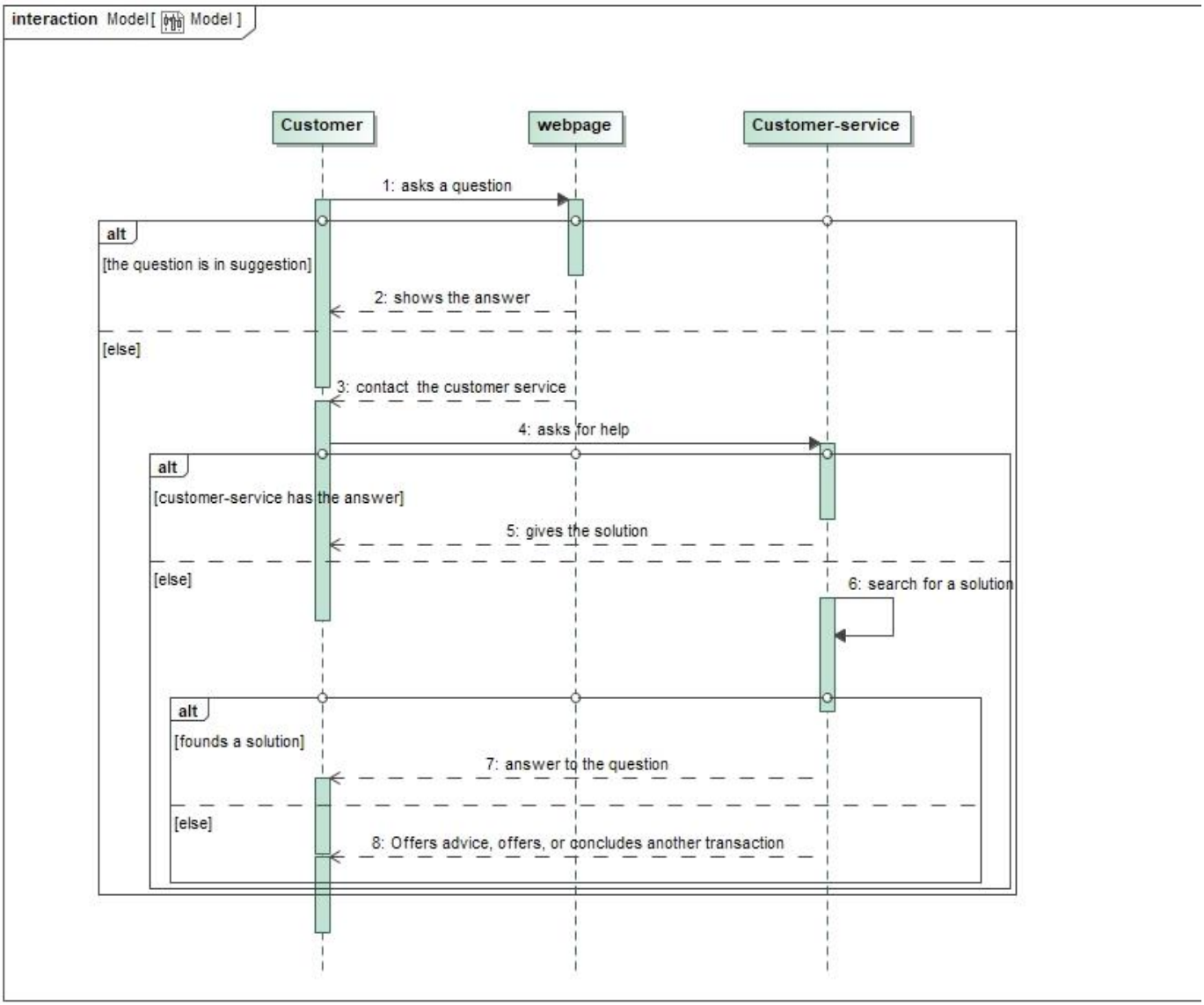


Table 64 Sequence Diagram "Customer Service"

4.1.13.4 Activity Diagram

In this activity diagram shows steps of customers how they get support and help.

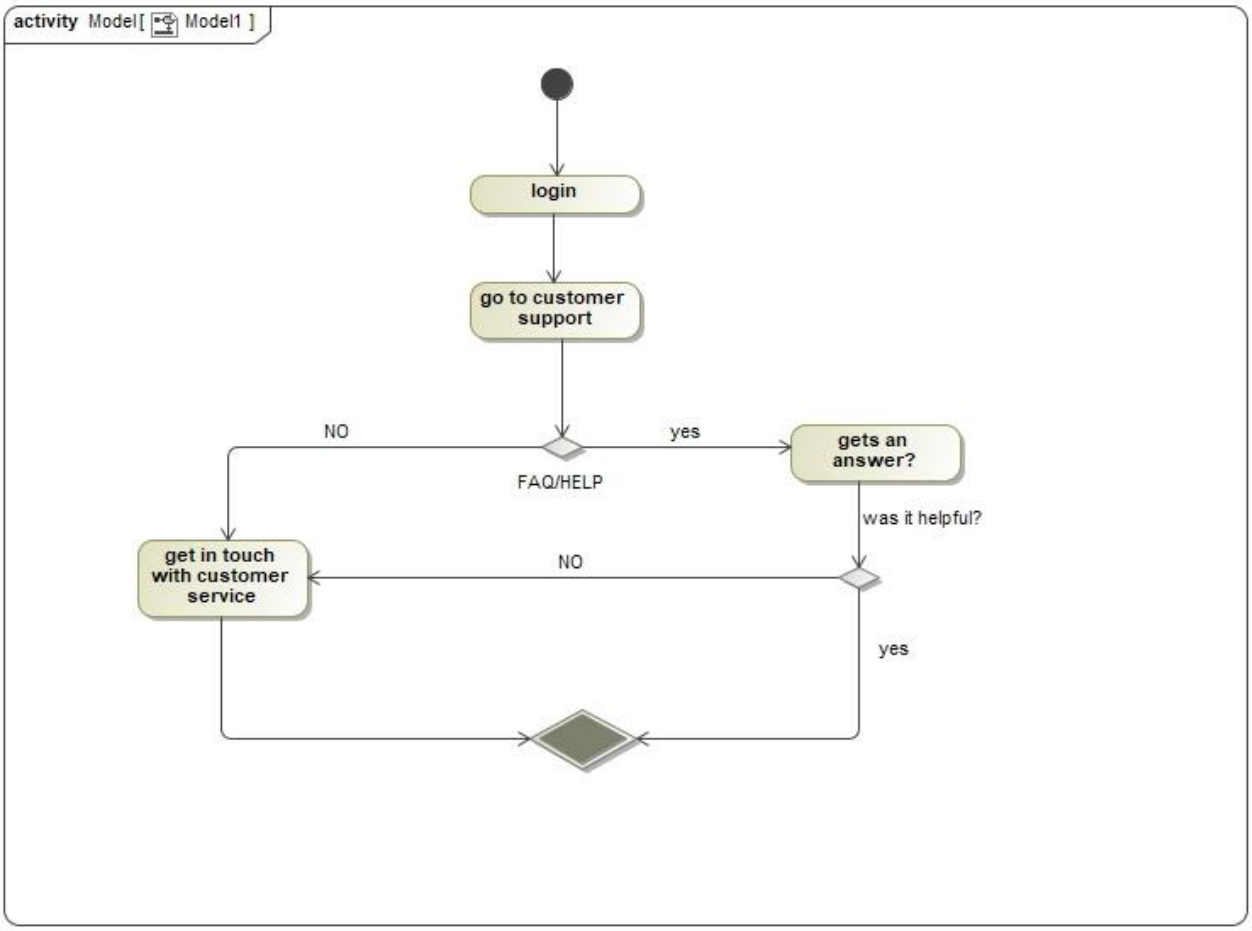


Table 65 Activity Diagram "Customer Service"

4.1.13.5 Graphical User Interface

the Graphical User Interface displays the support option for the customer.

Table 66 Graphical User Interface "Customer Service"

4.1.14 Requirement-„User Feedback “

4.1.14.1 Use Case Specification

Use Case	User Feedback
Description	This use case starts when the customer receives his translation from the translator and ends when he gives feedback.
Actors	Customer
Pre-conditions	The customer has received the translation.
Basic Flow **Description** **Actions** **1** **2** **3** **4** **5**	The customer provides Feedback. The customer receives feedback email. The customer selects “provide Feedback”. System launches the feedback page. The customer enters message. System sends mail.
Alternative Flow **Description** **Actions** **1** **2**	After the customer receives the translation, he doesn’t provide the feedback. The customer receives feedback email. The customer doesn’t provide feedback.
Post conditions	The feedback has been successfully sent.

Table 67 Use Case Specification "User Feedback"

4.1.14.2 Use Case Diagram

This use case diagram clarifies the requirement “User Feedback”. Here the customer has the choice to rate the service after getting the translation. The customer can give stars and comments as feedback.

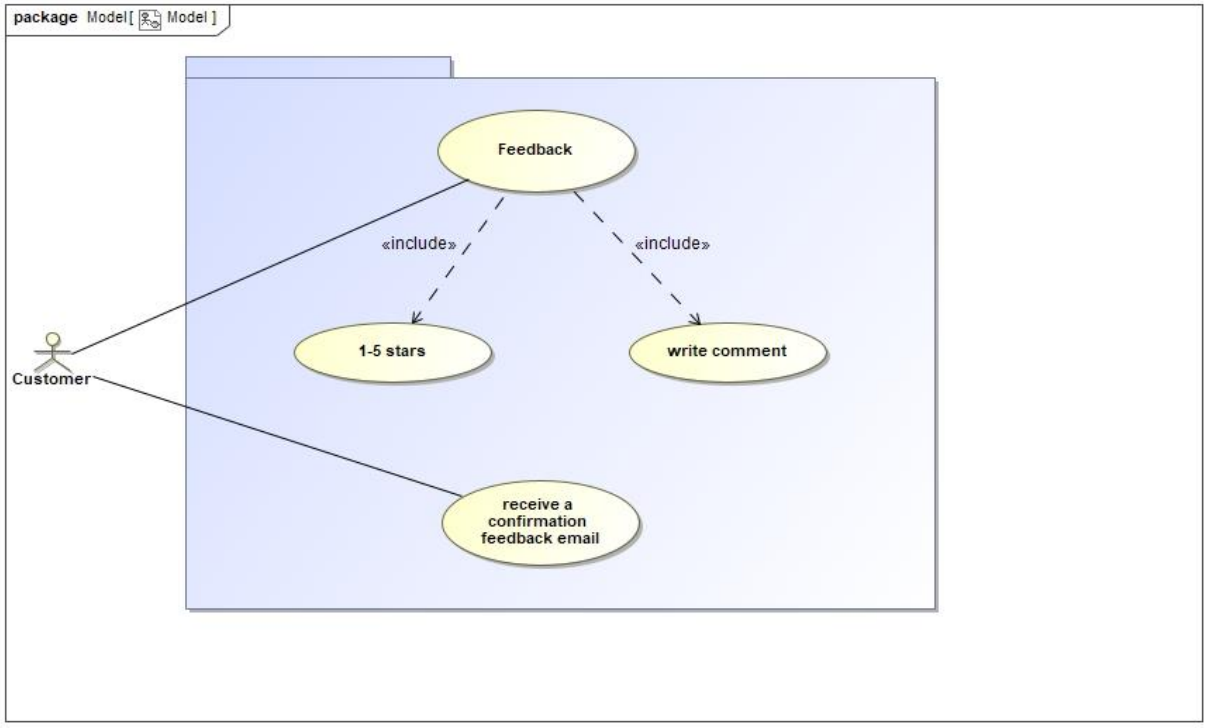


Table 68 Use Case Diagram "User Feedback"

4.1.14.3 Sequence Diagram

In this sequence diagram, the customer starts the program after clicking on the button feedback. The feedback page will be displayed, where he can give the feedback by stars or comments.

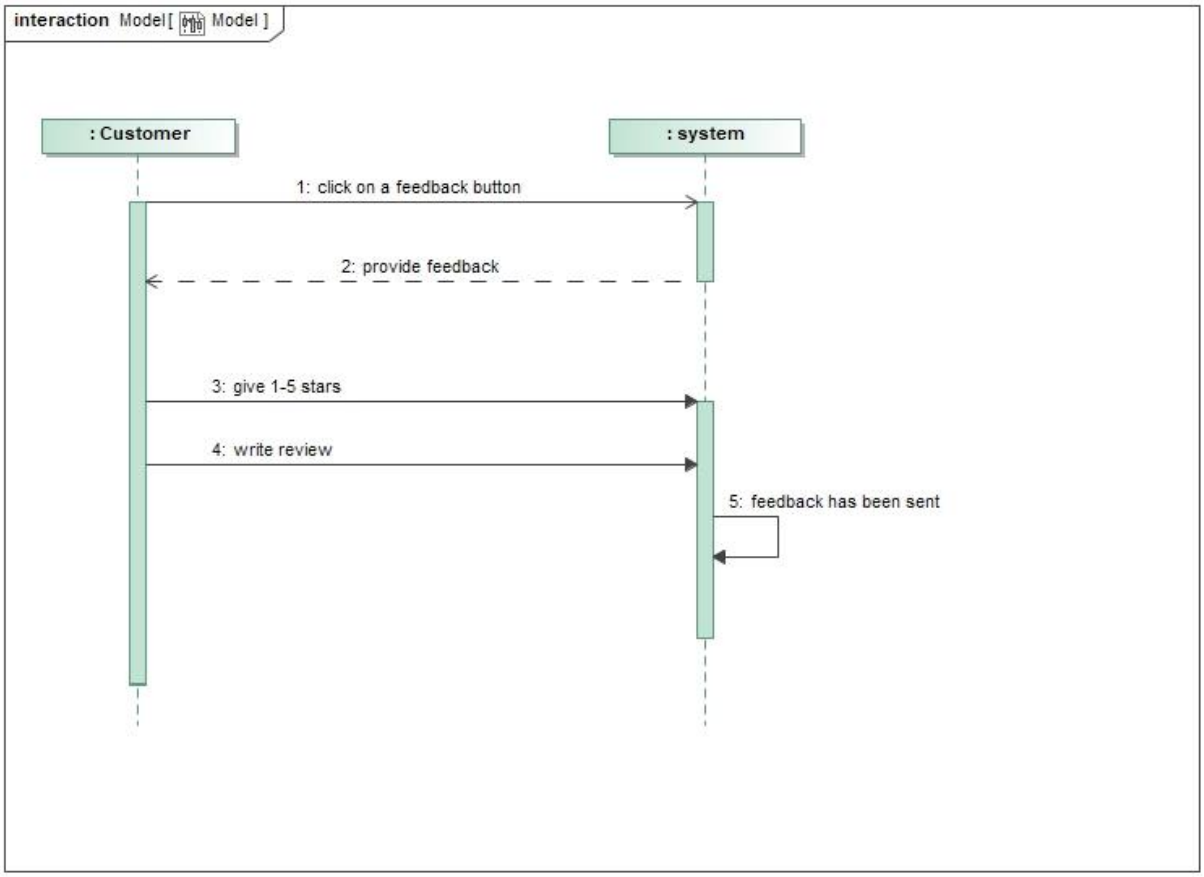


Table 69 Sequence Diagram "User Feedback"

4.1.14.4 Activity Diagram

This activity diagram shows us how the customer can rate the service from the application.

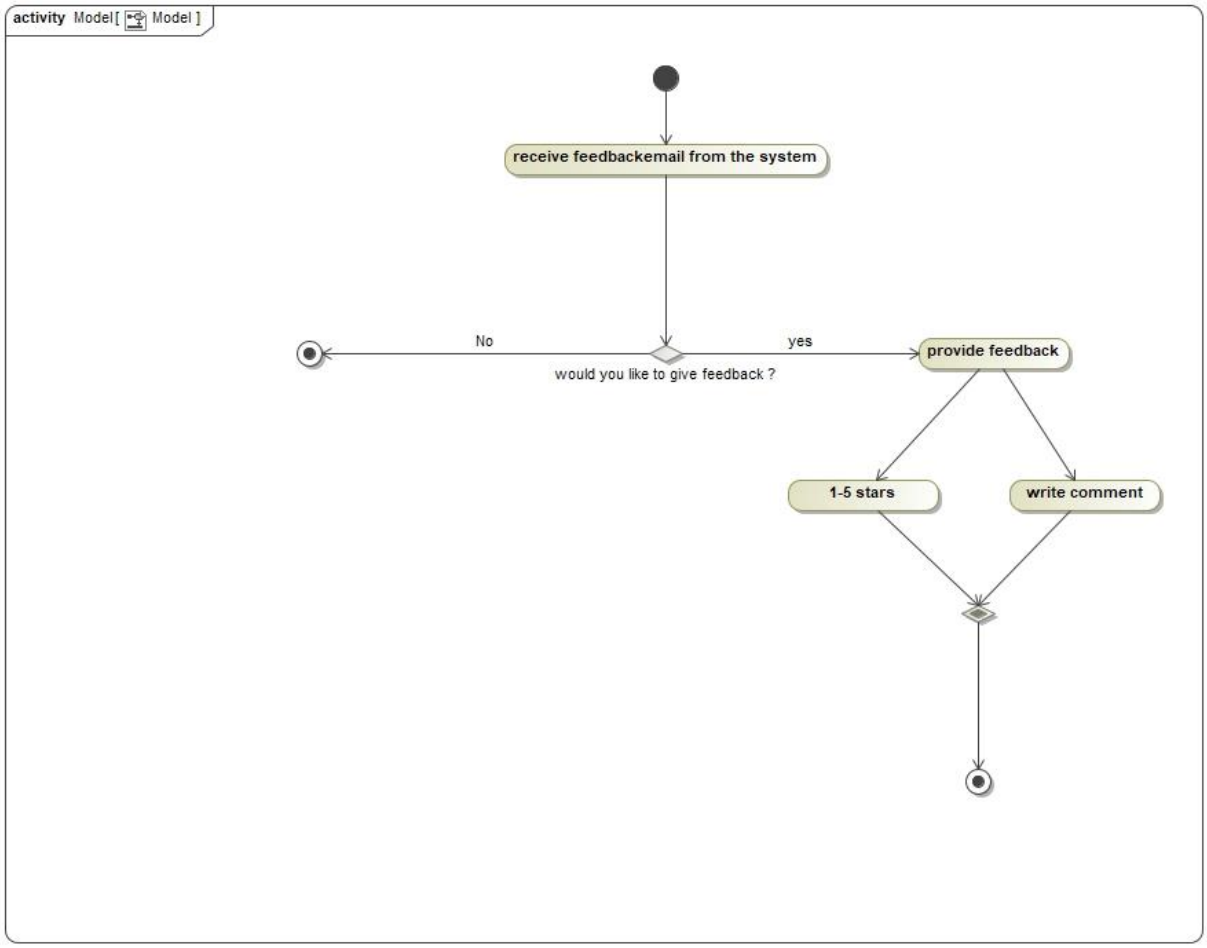


Table 70 Activity Diagram "User Feedback"

4.1.14.5 Graphical User Interface

The graphical user interface shows the 2 steps to give feedback to our service.

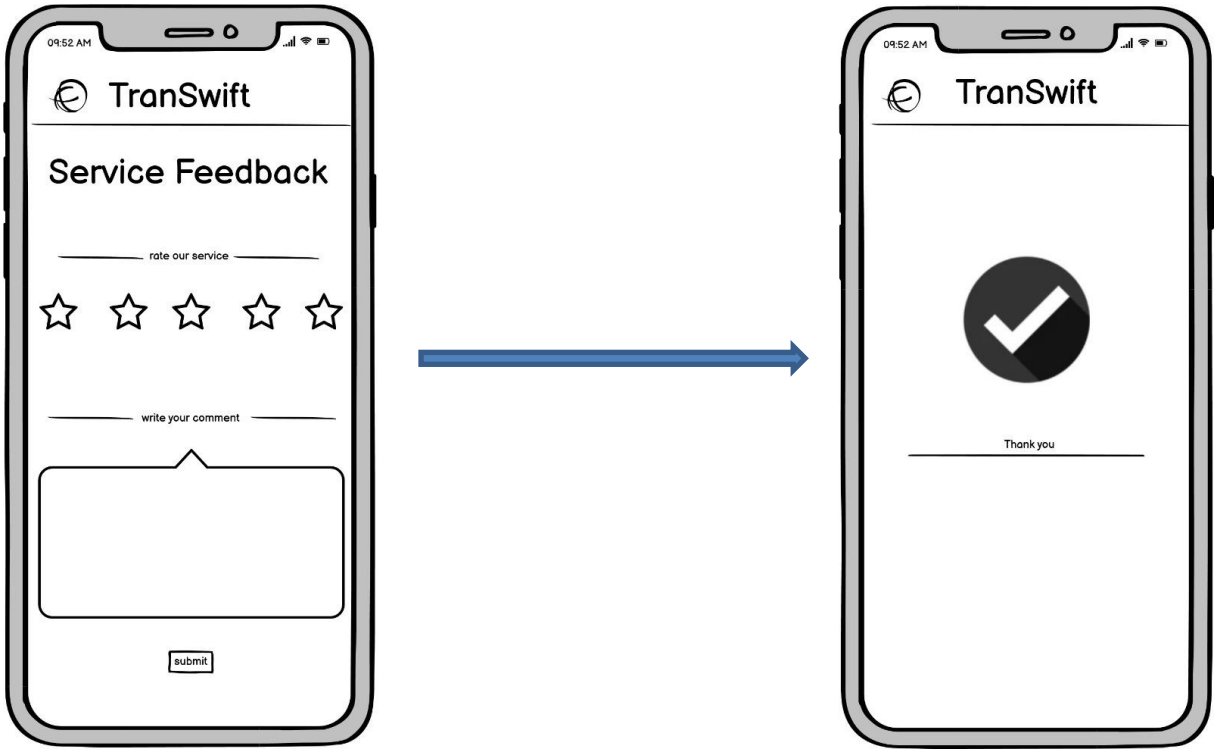


Table 71 Graphical User Interface "User Feedback"

4.2. Atomic Requirements

Atomic Requirements specify a single requirement which is measurable, testable, and traceable.

Unlike the Flow or Process Oriented Requirements, Atomic Requirements describe no flow or a set of actions, thus we decided to describe them with only voleresnowcards.

The UML diagrams used to describe the Flow Requirements are barely applicable for this kind of requirements because of the lack of flow and interactions.

But first, let's define what a voleresnowcard is:

A voleresnowcard is simply a card that records all information about a single requirement.

In our case, our snowcards consist of the following information:

- the requirement Id
- the requirement types
- a brief description of the intention of the requirement
- the rationale: a justification of the requirement
- the user satisfaction if this requirement is implemented
- the user dissatisfaction if this requirement is not implemented
- Dependencies: a list of other requirements that have some dependency on the requirement
- Conflicts: a list of other requirements that cannot be implemented along with the requirement

4.2.1 Requirement-„Privacy policy & Terms of use “

Privacy policy & Terms of use	
Requirement #:	**0**
Requirement type:	functional
Description:	The company must provide a full and clear agreement on how the data will be handled and used. This agreement can be accepted or refused from the user before registration.
Rationale:	A privacy policy explains to users how their personal data will be used, what measures have been taken to keep it safe, and how they can exercise their rights over their personal data. Terms & Conditions set out what’s expected from both the company and the users.
User Satisfaction:	high
User Dissatisfaction:	medium
Dependencies:	None
Conflicts:	None

Table 72 "Privacy Policy & Terms of Use"

4.2.2 Requirement-„Contractor's network status "

Contractors network status	
Requirement #:	16
Requirement type:	Non functional
Description:	The company must be able to check the status of the contractors network and keep track of it.
Rationale:	the status of the contractors network must be checked in order to have an idea on the growth of the network in particular and the project in general. From one side, the larger the network, the customer has a bigger and a better choice. and from the other side the larger the network is, the bigger the profit from both contractors and customers will be.
User Satisfaction:	high
User Dissatisfaction:	medium
Dependencies:	requirements #15 and #17
Conflicts:	None

Table 73 "Contractor's network status"

4.2.3 Requirement-„Company growth and profit per customer“

Company growth and profit per customer	
Requirement #:	17
Requirement type:	Non functional
Description:	Profit per customer can be checked in order to have an overall outlook on the growth of the company.
Rationale:	The company gets bigger in the market and goes really viral.
User Satisfaction:	low
User Dissatisfaction:	low
Dependencies:	requirements #15 and #16 and #11
Conflicts:	None

Table 74"Company growth and profit per customer"

4.2.4 Requirement-„Company Revenue “

Company Revenue

Requirement #:	14
Requirement Type:	Non functional
Description:	The company must be able to check the annual revenue and the revenue per customer, to maintain the cost of the company and to have an overview.
Rationale:	The revenue must be calculated by an accounted of the company. For this start-up companies that have yet to turn a profit, revenue can in some cases serve as a gauge of how much profit they will make in the future.
User Satisfaction:	low
User Dissatisfaction:	low
Dependencies:	none
Conflicts:	none

Software Engineering Analysis

Table 75"Company Revenue“

5. Scrum: Sprints, Meetings, and their Retrospectives[5]

5.1 Product Backlog

ID	Backlog Item	Priority	Estimation
0	Privacy policy & Terms of use	High	1
1	Customer Registration	High	4
2	Contractor registration	High	4
3	User Verification	High	3
4	User login	High	4
5	Account management	Medium	3
6	contractors search filter	High	4
7	New translation request	High	3
8	Translation request confirmation	High	3
9	tracking a translation order	Medium	4
10	Translation document delivery	High	3
11	Payment methods	High	4
12	Contractor rating	Medium	2
13	Customer service	High	4
14	User feedback	Medium	2
15	Project revenue	High	2
16	Contractors network status	Low	1
17	Company growth and profit per customer	High	2

Table 76 Product Backlog

[5] (Ken Schwaber & Jeff Sutherland, 2020)

5.2. Sprints:

5.2.1 Sprint 1:

5.2.1.1 Sprint 1 – Documentation

Contributors:

- **Ahmed Agdmoun, 1347063**
- **Houda El Abbassi, 1360545**
- **Mohamed Amine Belrhazi, 1343876**
- **Fatima-Zahra ElAzraq, 1344273**
- **Hicham Boulaayad, 1310812**
- **Mohamed Yazid Rebbali, 1312674**

Plan of the Sprint 1:

- ✓ **Create a class diagram**
- ✓ **Set a Sprint deadline**
- ✓ **Set an appointment for the Sprint Review**

5.2.1.2 Sprint 1: Project Plan

Project name: Online Translation Services Platform

Overall progress: 10%

Start & End: 17.01.2022-21.01.2022

5.2.1.3 Sprint 1 Backlog

Requirement	Task	Assigned to
All	**Create a class diagram**	**Everyone**

Table 77 Sprint 1 Backlog

5.2.1.4 Sprint 1 Review and Retrospective

This was our first sprint, and hence we thought it would be better if we start with a lightweight task. The task was creating class diagrams, and although it sounded easy at the beginning and we have already discussed class diagrams in our meeting, it turned out most of us still had trouble illustrating the requirements as classes and objects and even deciding in the first place what can be presented as classes and what not. It turned out this problem was especially encountered by those who didn't take the OOP course yet. So, the first challenge was to help each other and get everyone to understand the concept of OOP and what classes, attributes, methods and class diagram components mean. During the Sprint we had several daily scrums to discuss these details and update each other about the progress.

At the end we managed to come out with some good results that were then assembled later into one overall class diagram. Here it was difficult to decide if the class diagrams were "done" or not, but since we all came to the conclusion that an overall class diagram makes more sense because of the strong relationships between different classes, we just decided to define the task as "done" and make an overall class diagram by assembling the single diagrams and connecting the related ones.

5.2.2 Sprint 2:

5.2.2.1 Sprint 2 - Documentation

Contributors:

- **Ahmed Agdmoun, 1347063**
- **Houda ElAbbassi, 1360545**
- **Mohamed Amine Belrhazi, 1343876**
- **Fatima-Zahra ElAzraq, 1344273**
- **Hicham Boulaayad, 1310812**
- **Mohamed Yazid Rebbali, 1312674**

Plan of the Sprint:

- ✓ **Create Snowcards/Use Cases, Use Case diagrams, Sequence Diagrams, Activity Diagrams and UI Prototypes**
- ✓ **Set a Sprint deadline**
- ✓ **Set an appointment for the Sprint Review**

5.2.2.2 Sprint 2: Project Plan

Project name: Online Translation Services Platform

Overall progress: 35%

Start & End: 24.01.2022-28.01.2022

5.2.2.3 Sprint 2 Backlog

ID	Requirement	Task	Assigned to
0	Privacy policy & Terms of use	Create a Snowcard	Fatima-Zahra
1 & 2	Customer Registration & Contractor Registration	Create Use Case and its diagram, sequence and activity diagram + UI	Yazid & Amine
3	User Verification	Create Use Case and its diagram, sequence and activity diagram + UI	Houda
4	User login	Create Use Case and its diagram, sequence and activity diagram + UI	Ahmed
5	Account management	Create Use Case and its diagram, sequence and activity diagram + UI	Hicham

Table 78 Sprint 2 Backlog

5.2.2.4 Sprint 2 Review and Retrospective

In the following Sprint, it was decided to achieve the goal which the team planed for the most important items with the highest priority.

The first challenge was to find a framework which the teams can use to deliver a consistent work in the shortest possible time.
With the help of MagicDraw framework, each member of the team was able to create a Use Case or a Snowcard depending on the given requirements, follows by the sequence and activity diagram. Moreover, the team designed the UI prototypes of each requirement for the actual sprint.
During the sprint, it was a daily review about the planned sprint items and the top requirements related to the sprint goal, which makes the team always informed and updated.
After a status overview and demonstrations of functionalities, there were difficulties to meet the definition of “done”, because of missing some features and details in "Registration requirements", because of the different proceeding to register for the customer and contractor.

That item was added by the team to the Sprint Backlog of the next Sprint to be completed.

5.2.3 Sprint 3

5.2.3.1 Sprint 3 - Documentation

Contributors:

- **Ahmed Agdmoun, 1347063**
- **Houda ElAbbassi, 1360545**
- **Mohamed Amine Belrhazi, 1343876**
- **Fatima-Zahra ElAzraq, 1344273**
- **Hicham Boulaayad, 1310812**
- **Mohamed Yazid Rebbali, 1312674**

Plan of the Sprint:

- ✓ **Create Use Cases, Use Case diagrams, Sequence Diagrams, Activity diagrams and UI Prototypes**
- ✓ **Split the registration requirement into two separate requirements and create diagrams for each one**
- ✓ **Set a Sprint deadline**
- ✓ **Set an appointment for the Sprint Review**

5.2.3.2 Sprint 3: Project Plan

Project name: Online Translation Services Platform

Overall progress: 50%

Start & End: 31.01.2022-04.02.2022

5.2.3.3 Sprint 3 Backlog

ID	Requirement	Task	Assigned to
1	Customer Registration	Create Use Case and its diagram, sequence and activity diagram + UI	Yazid
2	Contractor Registration	Create Use Case and its diagram, sequence and activity diagram + UI	Amine
6	Contractors search filter	Create Use Case and its diagram, sequence and activity diagram + UI	Amine
7	New translation request	Create Use Case and its diagram, sequence and activity diagram + UI	Hicham
8	Translation request confirmation	Create Use Case and its diagram, sequence and activity diagram + UI	Yazid
10	Translation document delivery	Create Use Case and its diagram, sequence and activity diagram + UI	Houda
11	Payment methods	Create Use Case and its diagram, sequence and activity diagram + UI	Fatima-Zahra
12	Contractor rating	Create Use Case and its diagram, sequence and activity diagram + UI	Ahmed

Table 79 Sprint 3 Backlog

5.2.3.4 Sprint 3 Review and Retrospective

In the third Sprint we decided that all the requirements in the sprint backlog item are "Done" included the requirements "Customer Registration" and "Contractor Registration". The requirements "Payment Methods" and "Translation document delivery" needed revision. The Use Case Specification of "Payment Methods" had a lot of details that weren't necessary. At the beginning of the Sprint there was a discussion whether the Use Case Diagram of the "Translation document delivery" was requested. In the end we simplified the Use Case Specification of "Payment Methods" and created the Use Case Diagram of "Translation document delivery". This sprint the division of the requirements went well, and the next division should be similar for the next sprint. We have to improve our communication for the next meeting.

5.2.4 Sprint 4:

5.2.4.1 Sprint 4 - Documentation

Contributors:

- **Ahmed Agdmoun, 1347063**
- **Houda ElAbbassi, 1360545**
- **Mohamed Amine Belrhazi, 1343876**
- **Fatima-Zahra ElAzraq, 1344273**
- **Hicham Boulaayad, 1310812**
- **Mohamed Yazid Rebbali, 1312674**

Plan of the Sprint:

- ✓ **Create Use Cases, Use Case diagrams, Sequence Diagrams, Activity diagrams and UI Prototypes**
- ✓ **Improve some of the results from the sprint 2**
- ✓ **Set a Sprint deadline**
- ✓ **Set an appointment for the Sprint Review**
- ✓

5.2.4.2 Sprint 4 : Project Plan

Project name: Online Translation Services Platform

Overall progress: 80%

Start & End: **07.02.2022-11.02.2022**

5.2.4.3 Sprint 4 Backlog

ID	Requirement	Task	Assigned to
9	Tracking a translation order	Create Use Case and its diagram, sequence and activity diagram + UI	Amine
13	Customer service	Create Use Case and its diagram, sequence and activity diagram + UI	Fatima-zahra
14	User feedback	Create Use Case and its diagram, sequence and activity diagram + UI	Hicham
15	Project revenue	Create a snowcard	Houda
16	Contractors network status	Create a snowcard	Ahmed
17	Company growth and profit per customer	Create a snowcard	Yazid

Table 80 Sprint 4 Backlog

5.2.4.4 Sprint Review and Retrospective

In this sprint, we have taken the last items of our backlog and applied what we have learned so far by creating a use case, sequence, activity diagram and some UI prototypes. However, one problem we encountered and discussed during one of our daily scrums was mixing up one of the items with an item from a previous sprint. We are talking here about “User Feedback” and “Contractor rating”, since these two deal with the same functionality but for two different purposes. In the "User Feedback" the customer should not rate the contractor but the whole user experience with the system. This problem was successfully resolved during the sprint by editing some of the diagrams of the requirement "User Feedback". And since this was our last sprint and all items were defined as “Done”, we didn’t discuss or plan any more sprints.

6.Class diagram

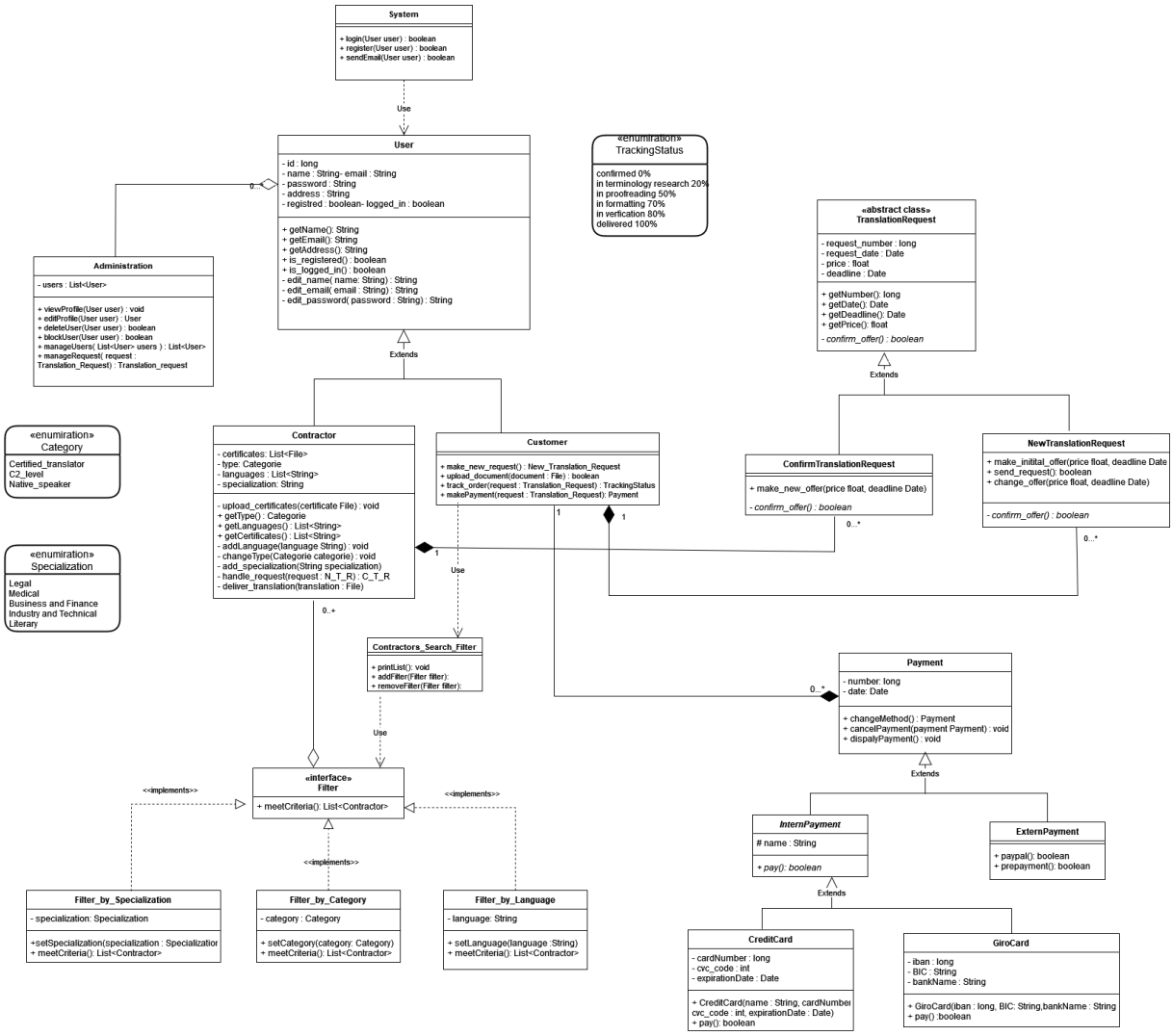


Table 81Class Diagram

Since most of our requirements are related to each other, we decided to design an overall class diagram for all requirements. Some of the requirements are represented as classes, others as enumerations and some just as methods within classes.

At the top we have a class system, which is responsible for the registration and the login of users as well for sending emails to the users. Of course, any other functions of the system can be added here. It uses the class User which can be a contractor or a customer; these two extend User as they inherit from it some attributes and have common methods with it. At the left of the User, we see the Administration class which is responsible for all kinds of management of the users from just viewing users to delete or block a user. It has a list of

users and thus the aggregation relationship. This list has no users at the beginning, and it can be any number then with our system gaining more users.

On the right side, we see the abstract class “TranslationRequest”. We made it abstract to prevent creating objects of the class itself so that only its subclasses will be used as objects. Its subclasses are “NewTranslationRequest” for creating a new one from the customer side, and “ConfirmTranslationRequest” for confirming an existing request from the contractor side. Both classes have to override the method “confirm_offer” in a different way. A customer can have many new translation requests, but a translation request is assigned only to one customer which explains the “one to many” relation. Furthermore, a new translation request cannot exist without a customer which justifies the composition relationship. The same applies for “ConfirmTranslationRequest” and Contractor.

Down at the left, we see a class for the contractor’s search filter, which uses the Interface Filter. This last one has only one abstract method “meetCriteria”, which returns a list of contractors and must be implemented by the different filters that implement the interface Filter according to certain criteria. Of course, other filters can be added easily to the system in the future.

At the bottom on the right, we see the Payment class which has attributes for example a date, a number which must be unique. A Payment can be an “ExternPayment” or an “InternPayment”. On the other hand, an “InternPayment” in return can be a GiroCard payment or a CreditCard payment. These two child classes of “InternPayment” have to override and implement the method pay in different ways, which return true if the payment was successful and false if it was not. Again, payments are dependent on the customer and a customer can make many payments, but every payment belongs always only to one customer, which justifies the composition with one-to-many relationship.

Finally, we have some enumerations like “Specialization” which represents the different specializations of translators and “TrackingStatus” which represents the different status a confirmed request can take during the translation.

We didn’t go through all the attributes and the methods, but we think most of them if not all are comprehensible from the context and their names.

7. Appendix

7.1 Meeting 1

Place and date: Meeting took place in BCN on 09th December 2021

Start/End: 10:00 – 15:00

Participants:

- Ahmed Agdmoun, 1347063
- Houda ElAbbassi, 1360545
- Mohamed Amine Belrhazi, 1343876
- Fatima-Zahra ElAzraq, 1344273
- Hicham Boulaayad, 1310812
- Mohamed Yazid Rebbali, 1312674

Protocol writer: Hicham Boulaayad

Content:

The first thing that we have done in the meeting was to get to know each other a little more and to discuss the core of our project and making sure that everyone understood the tasks. Then, we dealt with the topic and went through the tasks. After that we brainstormed and thought about the organization of the project.

At the end of the meeting, we discussed the plan of the next meeting.

7.2 Meeting 2

Place and Date: Meeting took place in BCN on 16th December 2021

Start/End: 10:00 - 15:00

Participants:

- Ahmed Agdmoun, 1347063
- Houda ElAbbassi, 1360545
- Mohamed Amine Belrhazi, 1343876
- Fatima-Zahra ElAzraq, 1344273
- Hicham Boulaayad, 1310812
- Mohamed Yazid Rebbali, 1312674

Protocol writer: Ahmed Agdmoun

Content:

After the first meeting, our ideas were still too general and not well formulated so we had to keep working on them and tried to optimise them the best. We started to write down our requirements, briefly describe them and categorize them by their priority and also their type if they are functional or non-functional. After that, we decided to start with some practical work and apply what we have learned in the lectures and the exercises, and the first thing was creating some snowcards and use case specifications. In this meeting we learnt more about the difference between atomic and flow or process-oriented requirements and we decided which requirements can be specified by just snow cards and which ones need to be specified by use cases and different diagrams. We also talked more about the team organization and tasks distribution and sat up a new appointment for our next meeting.

7.3 Meeting 3

Place and Date: Meeting took place in the BCN on 06th January 2022

Start/End: 10:00 - 15:00

Participants:

- Ahmed Agdmoun, 1347063
- Houda ElAbbassi, 1360545
- Mohamed Amine Belrhazi, 1343876
- Fatima-Zahra ElAzraq, 1344273
- Hicham Boulaayad, 1310812
- Mohamed Yazid Rebbali, 1312674

Protocol writer: Fatima-Zahra El Azraq

Content:

At first, we had a recap of the last meeting. We didn't have enough requirements so we Brain Stormed and came up with new requirements. We created for these new requirements SnowCards.

After a break we looked into MagicDraw and the different Diagrams. The Course Material was helpful to revision the needed Diagrams for the next weeks.

And we set a plan for the next week.

7.4 Meeting 4

Place and Date: Meeting took place online through Zoom on 13th January 2022

Start/End: 10:00 Till 15:30

Participants:

- Ahmed Agdmoun, 1347063
- Houda ElAbbassi, 1360545
- Mohamed Amine Belrhazi, 1343876
- Fatima-Zahra ElAzraq, 1344273
- Hicham Boulaayad, 1310812
- Mohamed Yazid Rebbali, 1312674

Protocol writer: Mohamed Amine Belrhazi

Content:

This was the first scrum meeting. We began with the creation of a class diagram to get a clear and full vision about the requirements. At end we decided to gather the whole class diagram for a good understanding relationship between them.

Till next meeting each of us will roughly do the other diagrams.

We decided to meet again on 20th of January before the Exercise.

7.5 Meeting 5

Place and Date: Meeting took place online through Zoom on 20th January 2022
Start/End: 11:00 - 15:30
Participants:

- Ahmed Agdmoun, 1347063
- Houda ElAbbassi, 1360545
- Mohamed Amine Belrhazi, 1343876
- Fatima-Zahra ElAzraq, 1344273
- Hicham Boulaayad, 1310812
- Mohamed Yazid Rebbali, 1312674

Protocol writer: Houda ElAbbassi

Content:

We started with presenting our progress to each other and corrected the mistakes. To overcome the difficulties of creating and understanding the diagrams we did some research.

For this week's Sprint everyone has to create a class diagram for his requirements. In this Meeting we worked on that.

7.6 Meeting 6

Place and Date: Meeting took place online through Zoom on 27.01.2022

Start/End : 10:00 till 15:45

Participants:

- Ahmed Agdmoun, 1347063
- Houda ElAbbassi, 1360545
- Mohamed Amine Belrhazi, 1343876
- Fatima-Zahra ElAzraq, 1344273
- Hicham Boulaayad, 1310812
- Mohamed Yazid Rebbali, 1312674

Protocol writer: Mohamed Yazid Rebbali

Content:

In this day we had a meeting, and we discussed a lot about our project. Firstly, as we know in our last meeting we had discussed and created snowcard/use case/use case diagrams/sequence diagrams /activity diagrams and then Ui Prototypes for some requirements. So, in this meeting we have to continue the creation of snowcards and use cases and improve our diagrams. Our overall progress was 50%. During our discussion and the correction of the snowcards, some questions and confusions appeared and we noted all our questions for the next exercise period. Roughly we also gave us a thought about our strategy. We decided to discuss on that in the next meeting. Although it was a productive meeting, and we were satisfied.

For our next meeting we have planned to finish our requirements.

7.7 Meeting 7

Place and date: Meeting took place online through Zoom on 3rd February 2022

Start/End: 10:00 – 15:00

- Participants: Ahmed Agdmoun, 1347063
- Houda ElAbbassi, 1360545
- Mohamed Amine Belrhazi, 1343876
- Fatima-Zahra ElAzraq, 1344273
- Hicham Boulaayad, 1310812
- Mohamed Yazid Rebbali, 1312674

Protocol writer: Hicham Boulaayad

Content:

On this day we continued to add the last touches to our requirements and then we prepared for the presentation.

We held a short presentation in the exercise session where each of us presented one requirement with its diagrams and UI prototyping. The presentation went very well and were pleased with the feedback from our Supervisor Mr. Mützel.

After a short break, we have continued to work on the documentation, and we set the plan for the last meeting.

7.8 Meeting 8

Place and Date: Meeting took place Online per Zoom on 17th February 2022

Start/End: 10:00 till 12:00

Participants:

- Ahmed Agdmoun, 1347063
- Houda ElAbbassi, 1360545
- Mohamed Amine Belrhazi, 1343876
- Fatima-Zahra ElAzraq, 1344273
- Hicham Boulaayad, 1310812
- Mohamed Yazid Rebbali, 1312674

Protocol writer: Ahmed Agdmoun

Content:

This was planned as our last meeting and its main aim was just to take a last look at everything and check the correctness and completeness of our documentation. We also had a discussion about our overall experience with this project during the semester and everyone gave brief feedback on it answering questions like what they have learnt, what they liked most and what they didn't, what could have been done better...etc. We wrote down our conclusion and ended the meeting wishing everyone a good luck in the upcoming exams.

8. Conclusion

To summarize the entire project, we noticed that with each sprint our ideas became clearer, and we understood how to create diagrams and how to categorize the items.

At the beginning we were confused, we found some difficulties and the communication was not so good.

But asking questions in the exercise sessions helped to minimize this problem and the communication with each other improved over time.

The Scrum approach was very useful and helpful to structure the project and its tasks. The longer the approach is used the more experience everyone has gained.

At the end of the project some parts of our working process still have room for improvement, as the fine tuning of our entire project.

In conclusion, during this project all team members learned a lot. It will be easier to deal with such diagrams and Scrum in the future. All of us know now how important they are, and how they can be useful to create a successful project.

9. Bibliography

[1] *DS CATIA*. (kein Datum). Von Translation Activity Diagram: https://docs.nomagic.com/display/MD190/Translation+Activity+Diagram abgerufen

[2] Ken Schwaber & Jeff Sutherland. (2020). *The Definitive Guide to Scrum: The Rules of the Game.* 2020 Ken Schwaber and Jeff Sutherland.

[3] Rolf Hennicker, Nora Koch. (kein Datum). *Systematic Design of Web Applications with UML.* München: Institue of Computer Science and F.A.S.:T.

[4] Standford University IT. (kein Datum). Von Analysis Diagrams: https://uit.stanford.edu/pmo/analysis-diagrams abgerufen

[5] Tariq, O. (1. June 2020). *researchgate*. Von https://www.researchgate.net/figure/USE-CASE-DIAGRAM-USER-LOGIN-MANAGEMENT_fig4_341945176 abgerufen

[6] Xpate. (September 2019). Von Account Verification: https://doc.sg.xpate.com/card_payment_API/account-verification.html abgerufen