Tunisian Republic
Ministry of Higher Education
and Scientific Research

University of Sfax

National School of Electronics and
Telecommunications of Sfax

Telecommunications Engineering

End of year project

END OF YEAR PROJECT

presented at

The National School of Electronics and Telecommunications of Sfax

Telecommunications Engineering

by

Syrine Abouda

Complaint Web Application for DraexImaier

supported on May 30 2022 , in front of the commission

M.	Houssem Lahiani	*Examinator*
M.	Achraf Mtibaa	*Supervisor*

Année Universitaire : 2021-2022

Declaration

I hereby declare that this project work entitled “Complaint Web Application for Draexlmaier” has been prepared by me during the year 2021 – 2022 under the guidance of Mr.Achraf Mtibaa, Department of Telecommunications at National School of Electronics and Telecommunications of Sfax.

I also declare that this project is the result of my own work and my personal effort , except for extracts and summaries for which the original references are stated herein.

Acknowledgement

I would like to express my deepest and sincerest appreciation to all those who provided me with the possibility to complete this report.

I give special gratitude to my final year project supervisor, Mr. Achraf Mtiba, whose contributions in stimulating suggestions and encouragement, helped me to coordinate my project, especially in writing this report.

I also appreciate the effort that he has invested in guiding me to achieve the goal.

Last but not least, many thanks to Mr. Houssem Lahiani, the examiner, for agreeing to evaluate my work.

Summary

General introduction ……..1

I. chapter 1: Preliminary study...……...2

I.1 Introduction…….2

I.2 Context of the project……2

I.3 Study of the existing……2

I.4 Problem Statement…….…2

I.5 Proposed solution ……..…2

I.6 Conclusion……..3

II. chapter 2:Requirements specification and design……..……4

II.1 Introduction…….4

II.2 Requirements Specifications …….4

II.2.1 Functional Requirements ……...4

II.2.2 Non Functional Requirements ……...4

II.2.3 The actors ……5

II.3 Used Language …….……..5

II.4 conceptual modeling ……..6

II.4.1 Use case diagram ……6

II.4.2 Class diagram ……..…8

II.4.3 Sequence diagram ……..11

II.4.4 conclusion ……...……12

III. chapter 3:Realization ……13

III.1 Introduction……..…….13

III.2 Software environment ……13

III.2.1 XAMPP ……13

III.2.2 MySQL ……..…13

III.2.3 Visual Studio Code……...…14

III.2.4 HTML ……...……15

III.2.5 CSS ……15

III.2.6 TypeScript ……16

III.2.7 PHP ……...……16

III.2.8 IBM WATSON ……...17

III.3 Chatbot Implementation ……..……17

III.4 Code ……...……21

III.5 Presentation of the application ……...……23

III.6 Conclusion……..……..28

General Conclusion ……..29

List of Figures

Figure II.3.1 - UML logo ……...6
Figure II.4.1 - Global use case diagram ……7
Figure II.4.2 -Class diagram ……9
Figure II.4.3 - Submit Complaint Sequence diagram ……11
Figure III.2.1 - XAMPP logo ……...13
Figure III.2.2 - MySQL logo ……14
Figure III.2.3 - Visual Studio Code code logo ……14
Figure III.2.4 - HTML logo ……15
Figure III.2.5 - CSS logo ……..15
Figure III.2.6 - TypeScript logo ……...16
Figure III.2.7 - PHP logo ……16
Figure III.2.8 - IBM Watson logo ……17
Figure III.3.1 - Intent ……...18
Figure III.3.2 - Entity ……..19
Figure III.3.3 - system Entity ……..…19
Figure III.3.4 - Dialog ……20
Figure III.3.5 - Chatbot ……...21
Figure III.4.1 - HTML code ……21
Figure III.4.2 - CSS code ……22
Figure III.4.3 - PHP code ……22
Figure III.5.1 - Home page ……..23
Figure III.5.2 - About page ……24
Figure III.5.3 - Complaint Form ……24
Figure III.5.4 - Product System ……...25
Figure III.5.5 - Product ……25
Figure III.5.6 - Subject of Complaint……26
Figure III.5.7 - Reviews ……26
Figure III.5.8 - Numbers ……27
Figure III.5.9 - Contact ……...27

List of Tables

Table II.2.3 -Table identifying the roles of the actors ……5

Table II.4.1-Table identifying Use case ……8

Table II.4.2.1-Table of Multiplicity ……10

Table II.4.2.2-Table of Data Dictionary ……...10

General Introduction

Over the last few years, Customer service optimization has become a major concern for companies, in particular Tech services companies.

Those companies realized the competitive advantage of offering a great customer experience and the value that resides within.

This is not only in terms of the quality of the product or the service they offer but also in the way they interact with their customers and prospects throughout their life cycle.

In light of this, I have chosen to develop a Web application that

enables customers to submit their detailed complaints concerning a specific product or a service.

This report is composed of three chapters structured as follows:

In the first chapter, we present the context of the project, study the existing system, problem statement, and the proposed solution.

Third chapter, is an implementation of the application, besides the results and the evaluation of the interfaces, and ultimately the future work.

I. chapter 1: Preliminary study

I.1 Introduction

In this chapter we describe the context of the project, then we specify the problem statement and we finally propose the ultimate solution .

I.2 Context of the project

This project is devoted for **Dräxlmaier Group which a** is a globally operating supplier Founded in 1958 in Germany.

It supplies world-class, premium automobile manufacturers with complex wiring harness systems, central electrical and electronic components, exclusive interiors, as well as battery systems for electromobility.

Dräxlmaier Group has more than 75000 COLLABORATORS and around 1258 PRODUCTS .

I.3 Study of the existing

If a customer in Dräxlmaier obtains a damaged product, usually he just contacts customer services by phone or Email, or even goes personally to the company office to inform them of the situation.

I.4 Problem Statement

The old way that Dräxlmaier's customers used to submit a complaint about a service or a product is a waste of time and money for both the company and the customer.

It takes a lot of time to specify the damaged part in a received product and as a result , even more time to specify the operating line responsible for the damage.

I.5 Proposed solution

The best-optimized solution to reduce and minimize the waste of time and money is to create an online interface that enables the customer to fill out a form of complaint to specify the problem.

After he submits his complaint and due to the details mentioned in the complaint form, workers can specify the faulty line,

then employees responsible for that line, on the shop floor, will solve the problem and stop producing nonconforming products.

The web application also provides a chatbot that enables customers to obtain more detail and to understand the cause of the problem in the bought product.

I.6 Conclusion

In this first chapter, we defined our objectives and the purpose of the web application.

The study of the existence enabled us to prepare an accurate Design

(conception) for our proposed solution.

In the next chapter, we will present the development and conception approach for our solution.

II. Chapter 2: Requirements specification and design

II.1 Introduction

In this chapter, we start with the analysis of the functional and non-functional needs of our project, then we detail the conceptual model of the web application.

Finally, we close this chapter with a conclusion.

II.2 Requirements Specifications

This phase enables us to understand the context of the application to identify the functional and non-functional Requirements .

II.2.1 Functional Requirements

These functional requirements are directly related to the tasks to be performed and must be as transparent as possible to the users.

- Having a database for information storage.
- Admin can manipulate and update the database.
- Clients can access the web application and fill the complaint form.
- Each client can use the chatbot.

II.2.2 Non Functional Requirements

Non-functional requirements are those that improve the quality of the application's services:

- **User-friendliness:** The system must be easy to use, and the user interfaces must be user-friendly

➔ simple and adapted to the user.

- **Performance:** The system must be efficient.

➔ it must provide the basic functionality in an optimal way and in a minimal time.

- **Portability:** Usable with several operating systems.
- **Ergonomics:** The interfaces must be simple, clear, and professional.

II.2.3 The actors

An actor represents a coherent set of roles played by a user or an external entity that interacts directly with the system.

Actor's Name	Role
Administrator	in charge of: managing the forms
Customer	can fill a form , use the chatbot

Table II.2.3 -Table identifying the roles of the actors

II.3 Used Language

UML language,(short for Unified Modeling Language), is a standardized modeling language that consists of an integrated set of diagrams, developed to help systems, and software developers specify, visualize, construct, and document the artifacts of software systems.

The UML uses mostly graphical notations to express the design of software projects correspondingly it helps project teams communicate, explore potential designs, and validate the architectural design of the software.

Why UML?

- ➔ Provide users with a ready-to-use, expressive visual modeling language as a result they can develop and exchange meaningful models.
- ➔ Provide extensibility and specialization mechanisms in order to extend the core concepts.
- ➔ It's independent of particular programming languages and development processes.
- ➔ Provide a formal basis that enables a better understanding for the modeling language.
- ➔ Integrate best practices.

➔ Maintain higher-level development concepts such as collaborations, frameworks, patterns, and components.

Figure II.3.1 - UML logo

II.4 conceptual modeling

In this chapter we will start the conceptual proposal that responds to the needs analyzed in the previous chapter.

The conceptual analysis is the most important phase that precedes the development of the application.

In this chapter we present a detailed design of our application.

II.4.1 Use case diagram

In UML, Use case diagrams summarize the details of the system's actors and their interactions with the system.

It describes a system's functional requirements in terms of use cases as a result it enables us to relate what we need from a system and how the system delivers on those needs.

The Use case model is generally used in all phases of the development cycle.

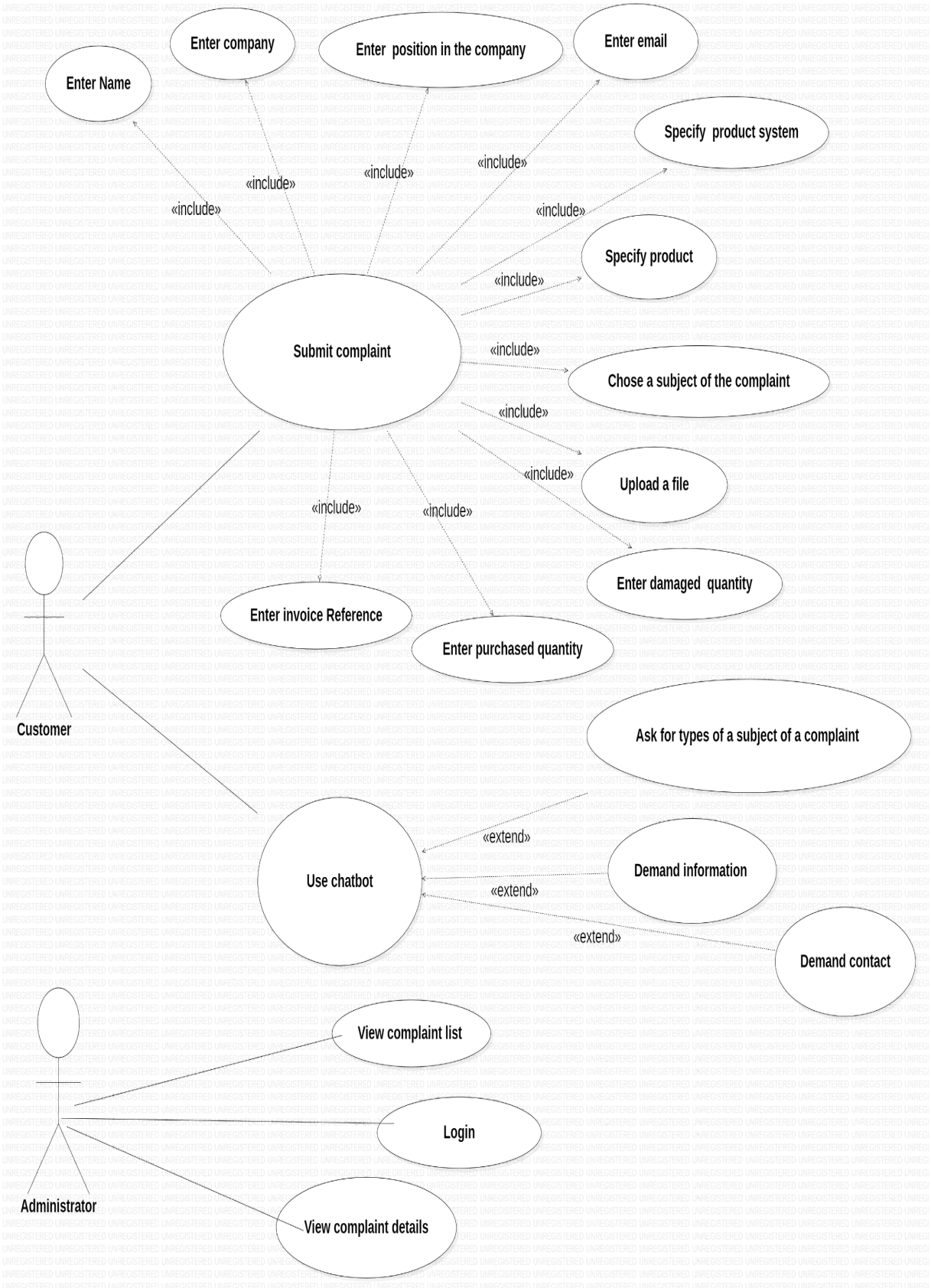

Figure II.4.1 - Global use case diagram

The following table illustrates the different use cases presented in the global use case diagram:

Actors	Use Case
Customer	● submit complaint ● enter Name ● enter company ● enter position in the company ● enter email ● specify product system ● specify product ● chose a subject of the complaint ● upload a file ● Enter purchased quantity ● Enter damaged quantity ● Enter invoice Reference ● Use chatbot ● Demand contact ● Demand information ● Ask for types of a subject of a complaint.
Administrator	● View complaint list ● View complaint details ● login

Table II.4.1-Table identifying Use case

II.4.2 Class diagram

A class diagram in the Unified Modeling Language (UML) is a type of static structure diagram and the main building block of object-oriented modeling

This diagram describes the structure of a system by showing the system's classes, their attributes, methods(or operations), and the relationships among objects.

The classes in a class diagram represent both the Principal elements, interactions in the application, and the classes to be programmed.

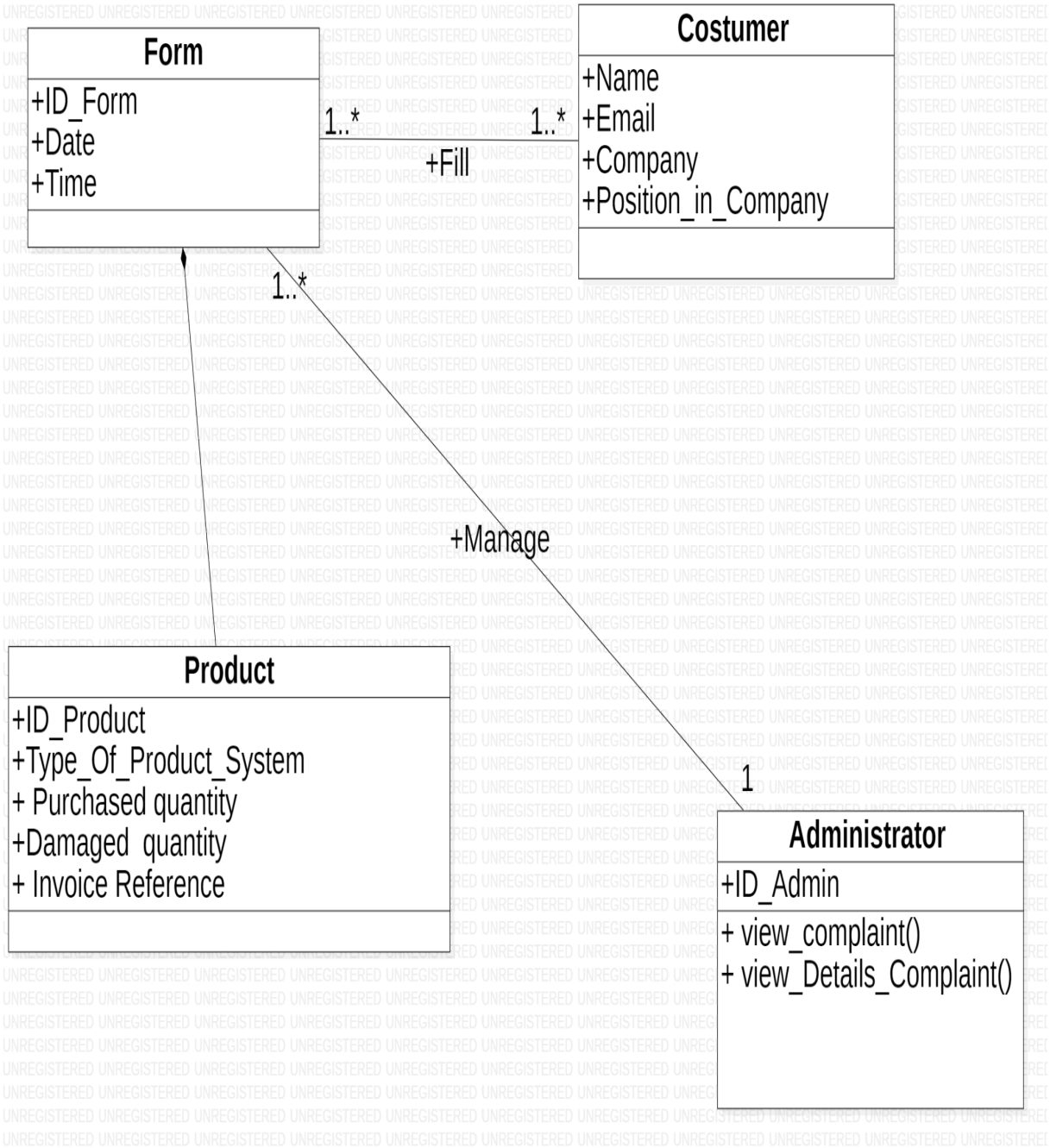

Figure II.4.2 -Class diagram

In UML, an association represents the link between two classes who need to communicate with each other.

We indicate the multiplicity of an association by adding multiplicity adornments to the line denoting the association.

The following table illustrates a representation of the multiplicities :

Multiplicity	**Meaning**
1	one and only one
0..1	zero or one
N	integer
m..n	from m to n (integer)
0..*	from 0 to many
1..*	from 1 to many

Table II.4.2.1-Table of Multiplicity

Field	**Type**	**Description**
ID-Form	Double	Identifier of the form
Date	Date	Date of submitting the Complaint
Time	Date	Time of Submitting the complaint
Name	String	customer’s Name
Email	String	customer’s Email
Company	String	customer’s Company
position_in_Company	String	customer's Position in a Company
ID _Admin	Double	Identifier of the admin
ID-Product	Double	Identifier of the Product

Type_Of_Product_System	String	The type of the System that the damaged Product belongs to
Purchased_quantity	Integer	Total Quantity of the bought Product
Damaged_quantity	Integer	the Damaged quantity of the Product
Invoice_Reference	Double	The reference of the invoice

Table II.4.2.2-Table of Data Dictionary

II.4.3 Sequence diagram

Sequence diagrams specifically focus on *lifelines* and the messages exchanged between them in order to perform a function before the lifeline ends.

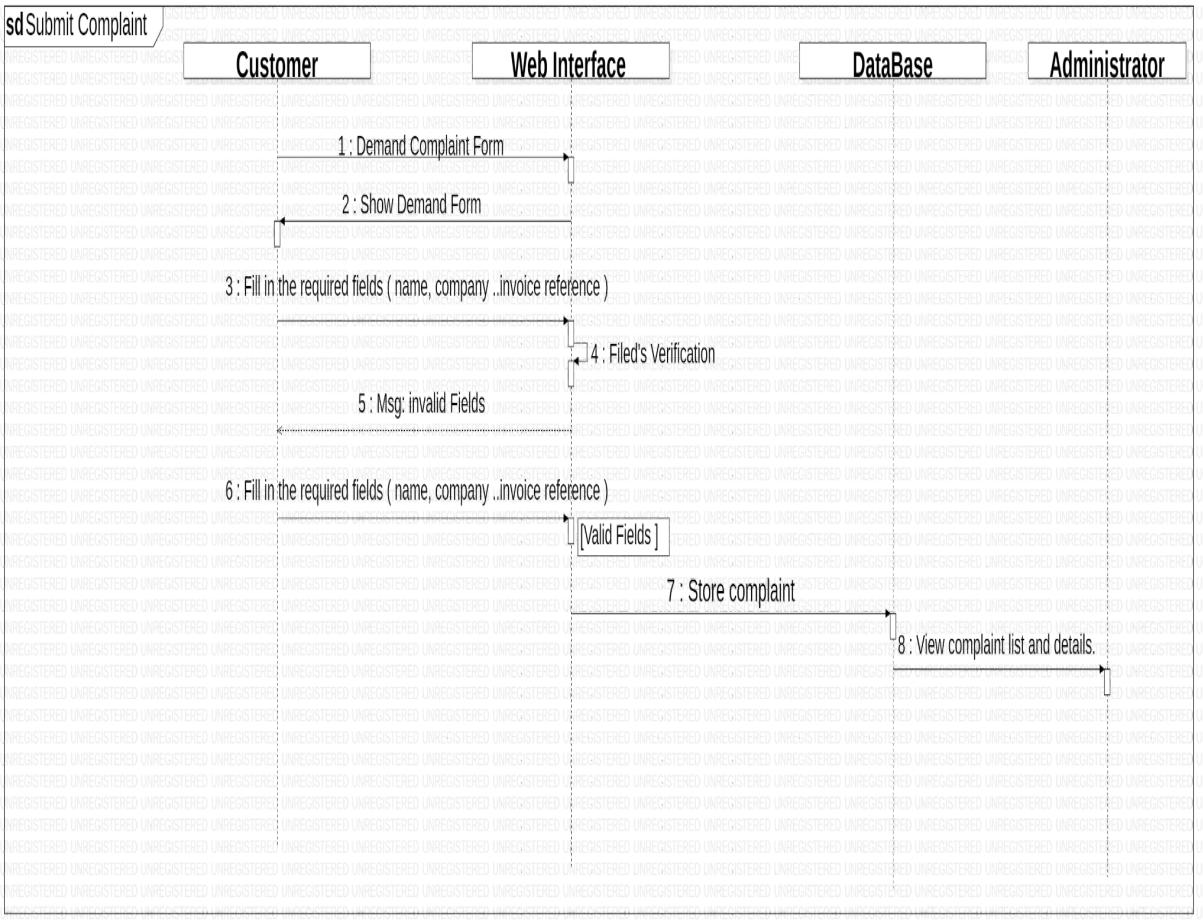

Figure II.4.3 - Submit Complaint Sequence diagram

II.4.4 conclusion:

In this chapter, we have carried out a conceptual study to facilitate the understanding of our work environment.

This chapter is seen as the most important since it brings us closer to the realization.

In the next chapter, we show the Project implementation.

III. Chapter 3:Realization

III.1 Introduction

After having completed the conceptual modeling, we are starting the Realization phase of the website.

During this phase, we first present the Software environment ,then we show Database and Chatbot Implementation and some Code.

We close this phase with some interface captures of the application.

III.2 Software environment

III.2.1 XAMPP

XAMPP is a set of software used to easily set up a Web server, an FTP server and an e-mail server. It is a free software distribution (X Apache MySQL PHP) offering a good flexibility of use, recognized for its simple and fast installation.

It does not require specific knowledge and works, moreover, on the most common operating devices.

Figure III.2.1 - XAMPP logo

III.2.2 MySQL

MySQL is a relational database management system (RDBMS). It

is distributed under a dual GPL and proprietary license. It is one of the most popular and used database management software in the world , both by the general public (mainly web applications) as well as by professionals.

Figure III.2.2 - MySQL logo

III.2.3 Visual Studio Code

Visual studio is a free source code editor for Windows, OSX and Linux that allows you to edit and debug your code without the need for the paid version of Visual Studio.

It includes a package and repository manager and can communicate with Git and other tools to manage your code.

Moreover,it is even possible to develop plugins for VS Code.

Figure III.2.3 - Visual Studio Code code logo

III.2.4 HTML

HTML5 (HyperText Markup Language 5) specifies two syntaxes of an abstract model defined in terms of DOM: HTML5 and XHTML5.

The language contains an application layer with many APIs, adding to an algorithm to handle documents with non-conforming syntax.

Figure III.2.4 - HTML logo

III.2.5 CSS

CSS (Cascading Style Sheets) is a style sheet language used to describe the presentation of HTML.

CSS is a core technology of the World Wide Web, alongside HTML and JavaScript.

It's created to allow the separation of presentation and content, including layout, colors, and fonts.

This separation improves the accessibility of content, allows more flexibility and control in specifying presentation characteristics, and enables multiple web pages to share formatting by specifying the relevant CSS separately.

Figure III.2.5 - CSS logo

III.2.6 TypeScript

TypeScript is a free open source programming language developed by Microsoft, that improves and secures the production of JavaScript code.

TypeScript allows optional static typing of variables and functions, creation of classes and interfaces, and import of modules while retaining the non-binding approach of JavaScript.

Figure III.2.6 - TypeScript logo

III.2.7 PHP

PHP (Hypertext Preprocessor), is a free scripting language used to produce dynamic Web pages via an HTTP server.

It also executes programs on the command line.

PHP is an imperative language with full object model functionality since version 5.

Thanks to the richness of its library, PHP is sometimes referred to as a platform more than just a language.

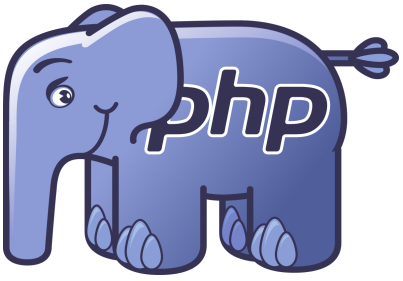

Figure III.2.7 - PHP logo

III.2.8 IBM WATSON

IBM Watson is a data analytics processor that uses natural language processing, it's a technology that analyzes human speech for meaning and syntax.

IBM Watson analyzes repositories of data to answer human-posed questions, often in a fraction of a second.

Figure III.2.8 - IBM Watson logo

III.3 Chatbot Implementation

An intent is a purpose or goal which is expressed in a customer's input, such as answering a question.

based on the intent expressed in a customer's input, the Watson Assistant service can choose the correct dialog flow for responding to it.

Here we have 5 intents :

- agent_contact.
- ending.
- greeting.
- info.
- partners.

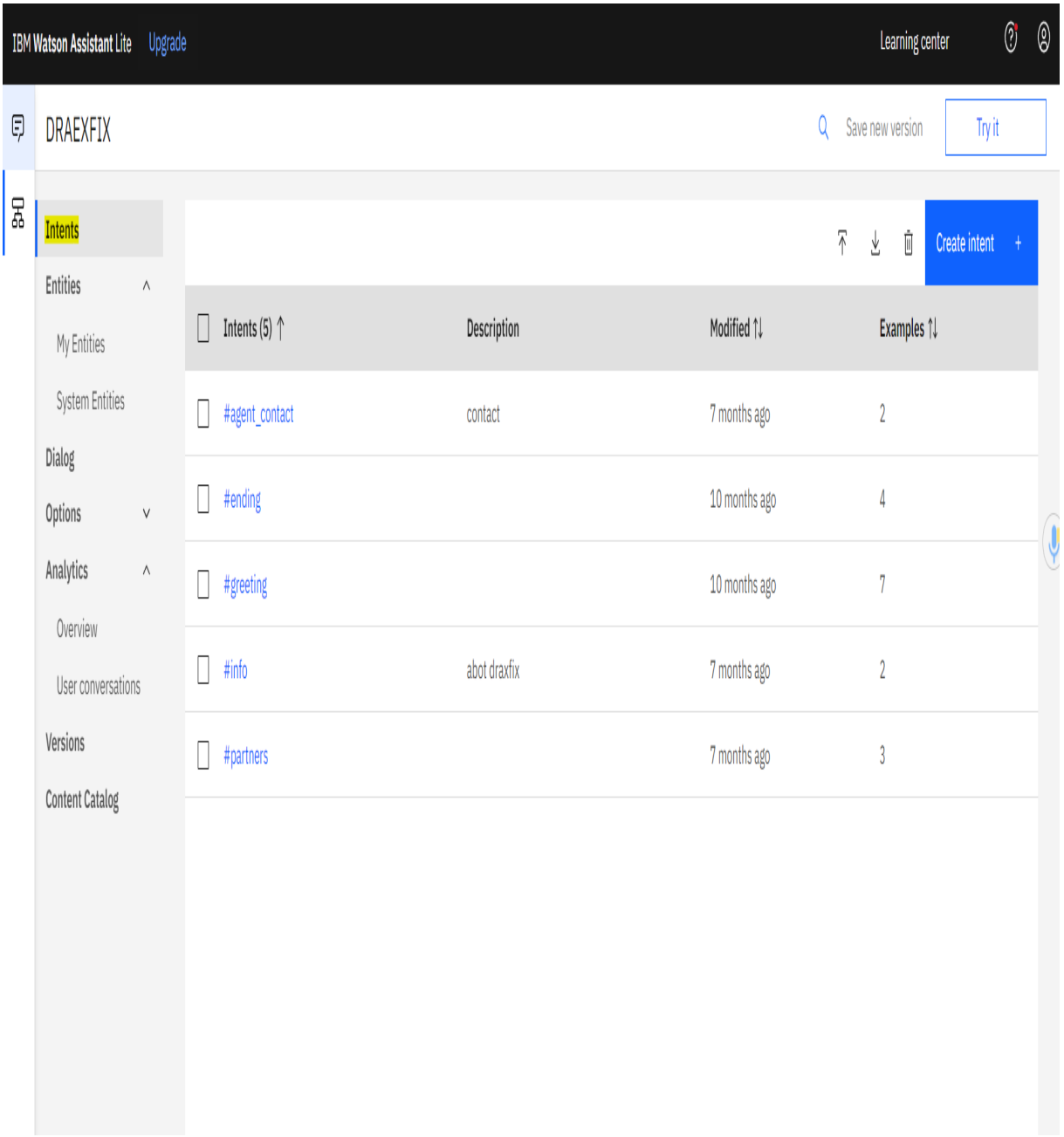

Figure III.3.1 - Intent

Entities identify interesting parts of the user's utterance, such as names and dates.

IBM Watson Assistant already provides system entities (for date, time, names, etc), and helps you define entities with synonyms and fuzzy matching, as well as defining pattern-based entities.

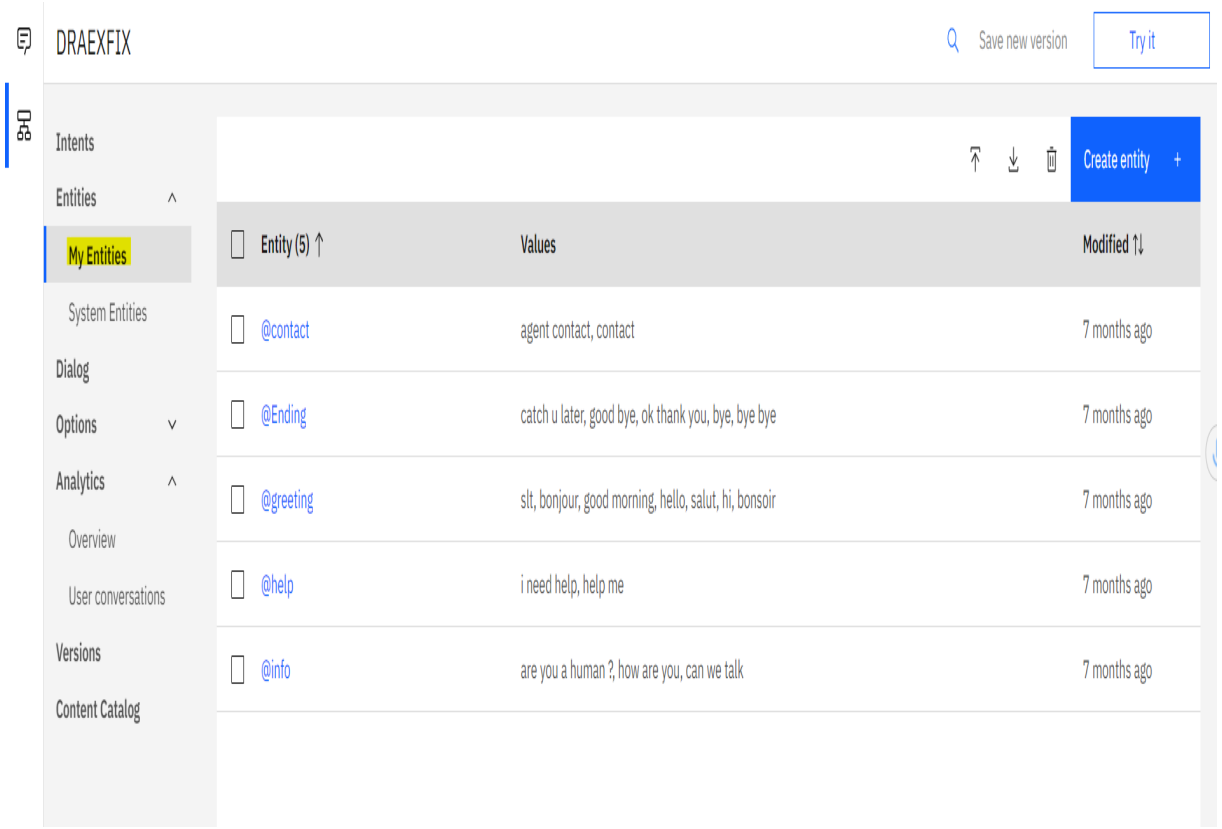

Figure III.3.2 - Entity

The following entities are prebuilt by IBM to recognize references to things like numbers and dates in user input. Turn on a system entity to start using it..

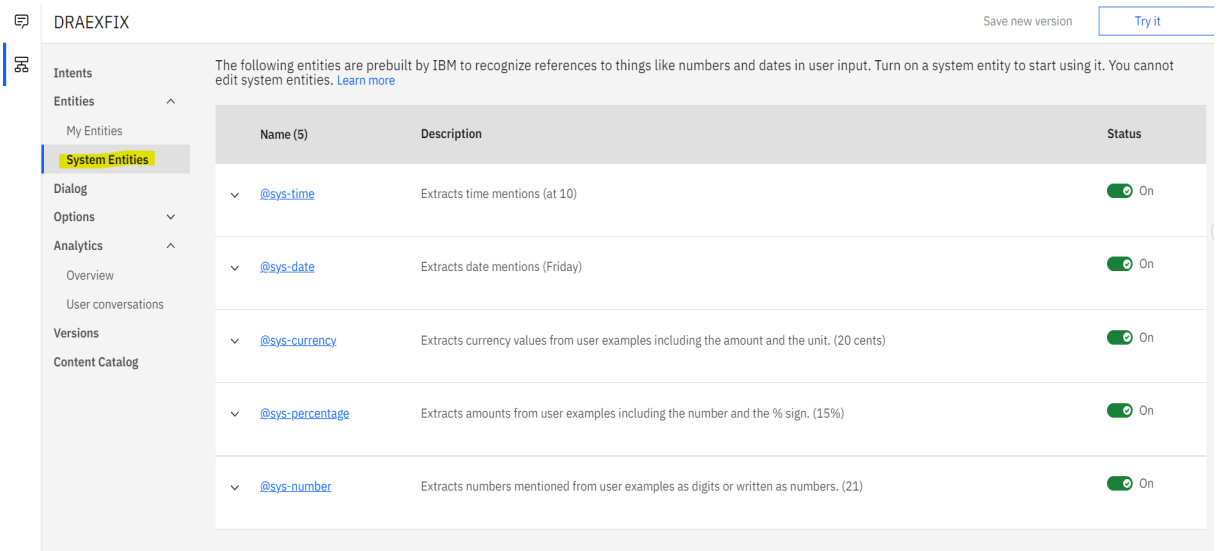

Figure III.3.3 - system Entity

The dialog uses the intents that are identified in the user's input, besides the context from the application, to interact with the user and provide a useful response.

The dialog matches intents (what users say they want to do) to responses (what the bot says back).

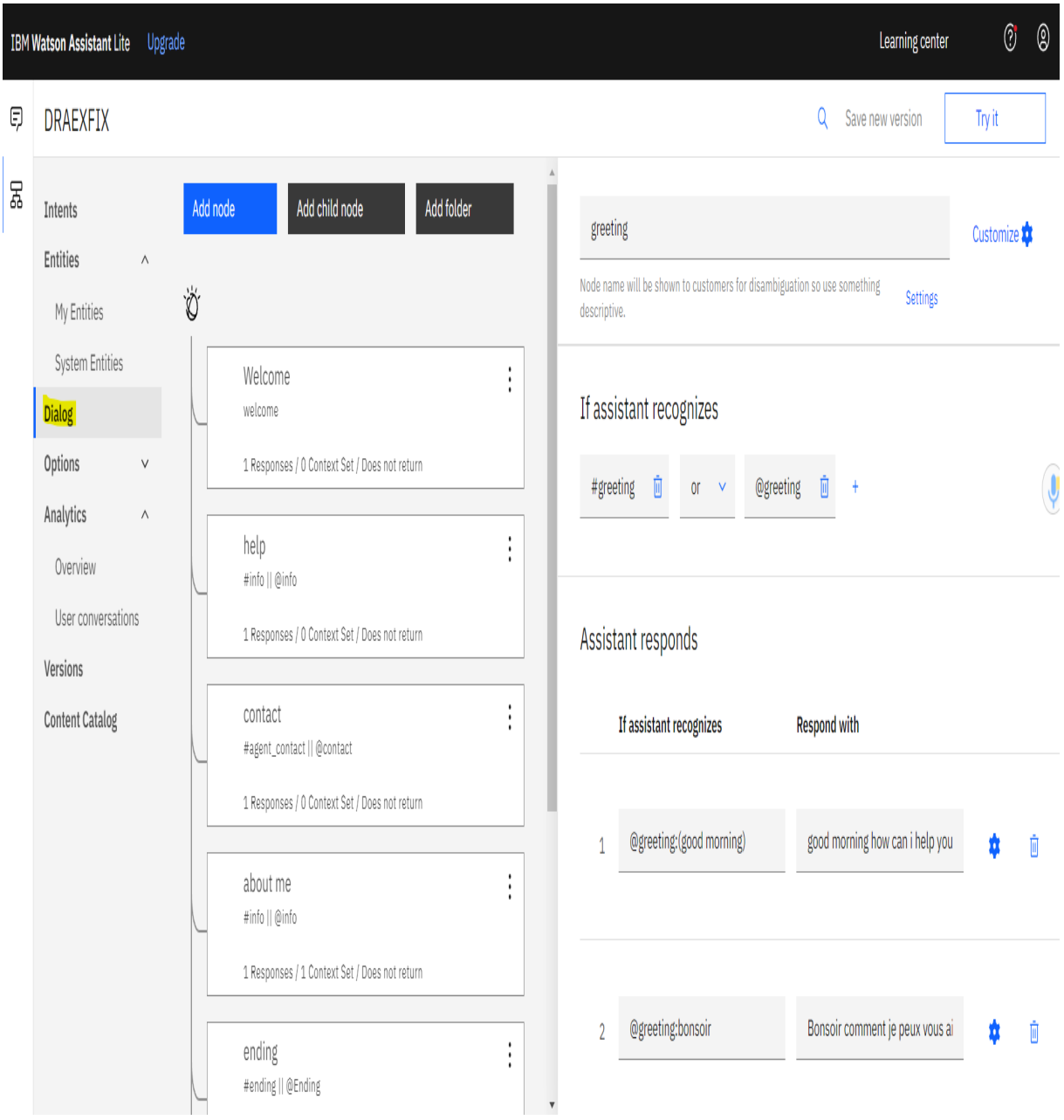

Figure III.3.4 - Dialog

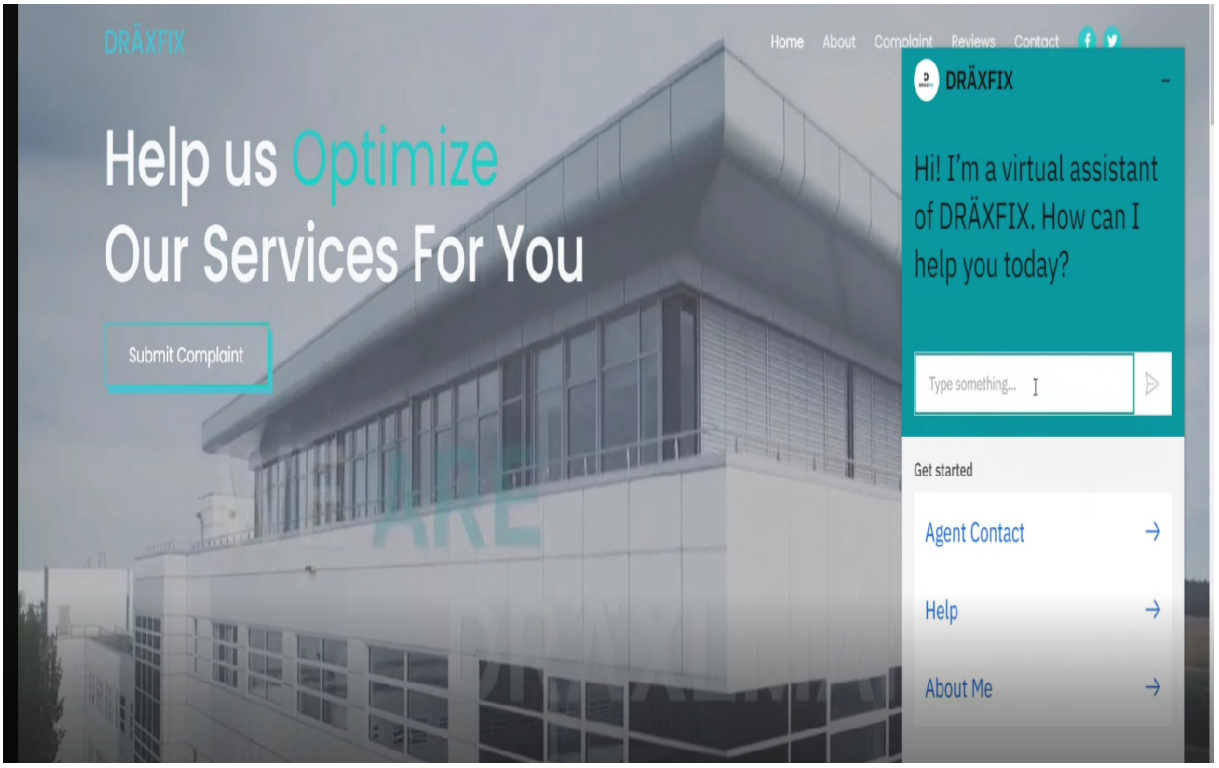

Figure III.3.5 - Chatbot

III.4 Code

I used the visual studio code as a code editor, in those figures, there is the part HTML ,CSS and PHP .

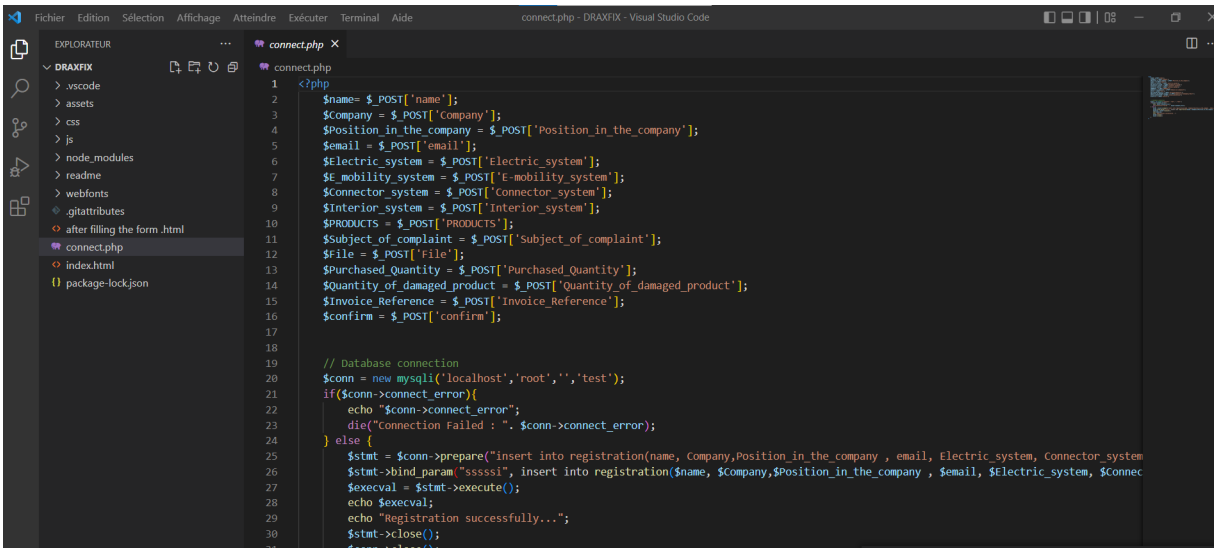

Figure III.4.1 - HTML code

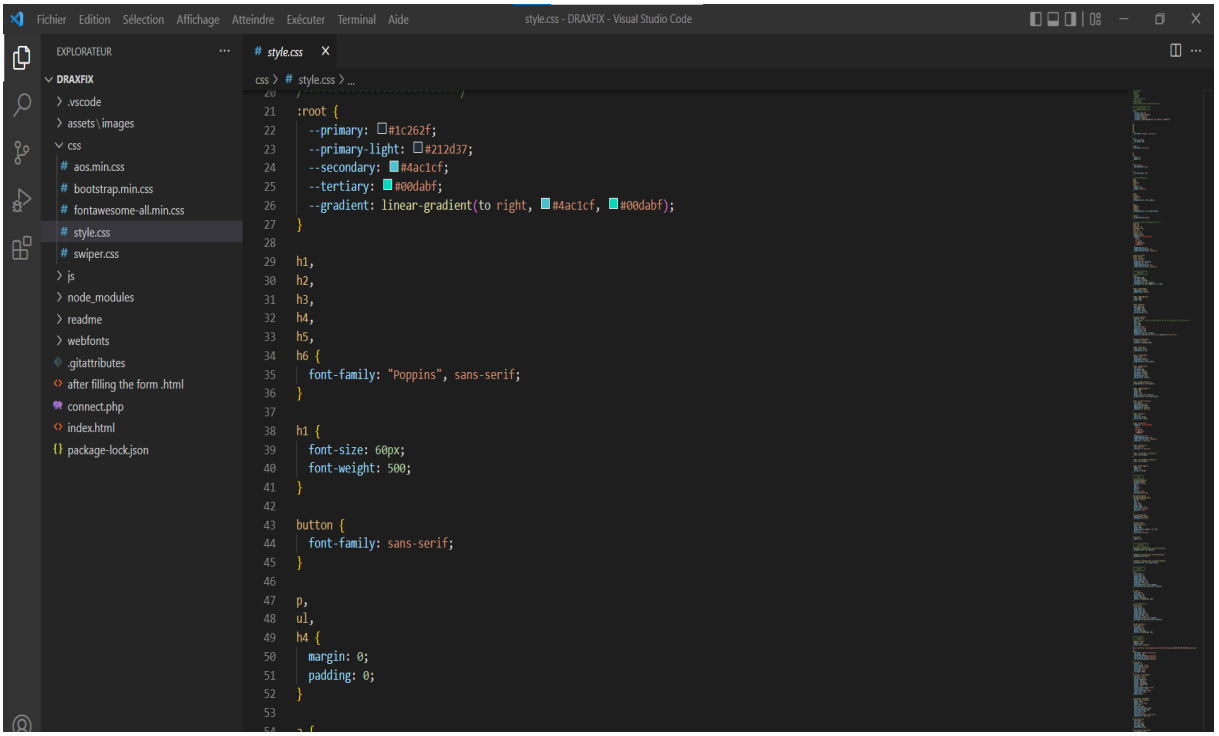

Figure III.4.2 - CSS code

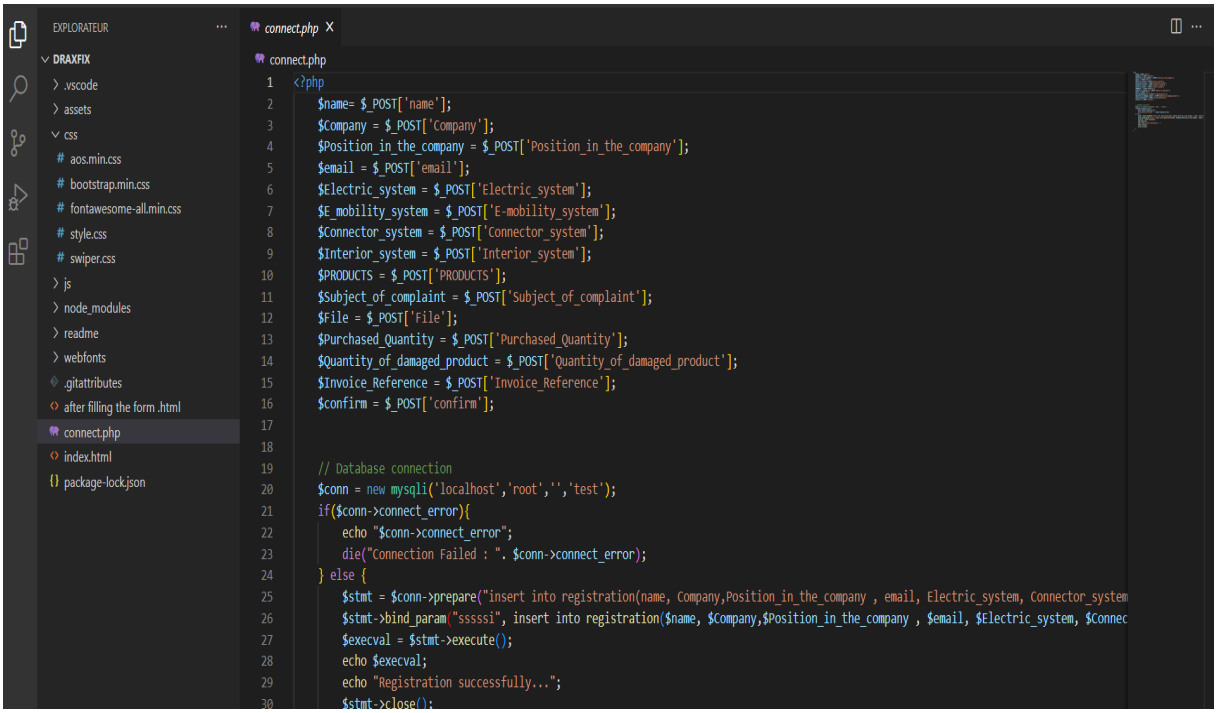

Figure III.4.3 - PHP code

III.5 Presentation of the application

The home page that contains submit button

Figure III.5.1 - Home page

The About page contains general information about Draexlmaier and the complaint web application.

Figure III.5.2 - About page

Before Costumes used to write long paragraphs or go to the company or even complain through the phone but now thanks to the detailed Form helps the administrator define the problem and finds the faulty line easier.

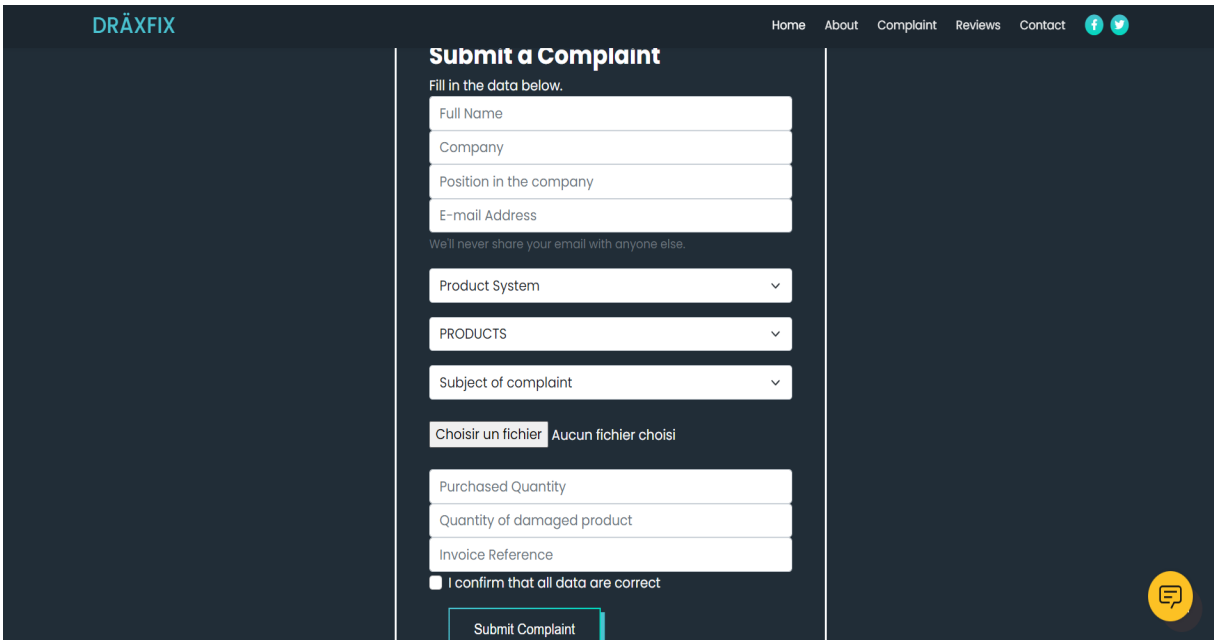

Figure III.5.3 - Complaint Form

Customer can specify product system. .

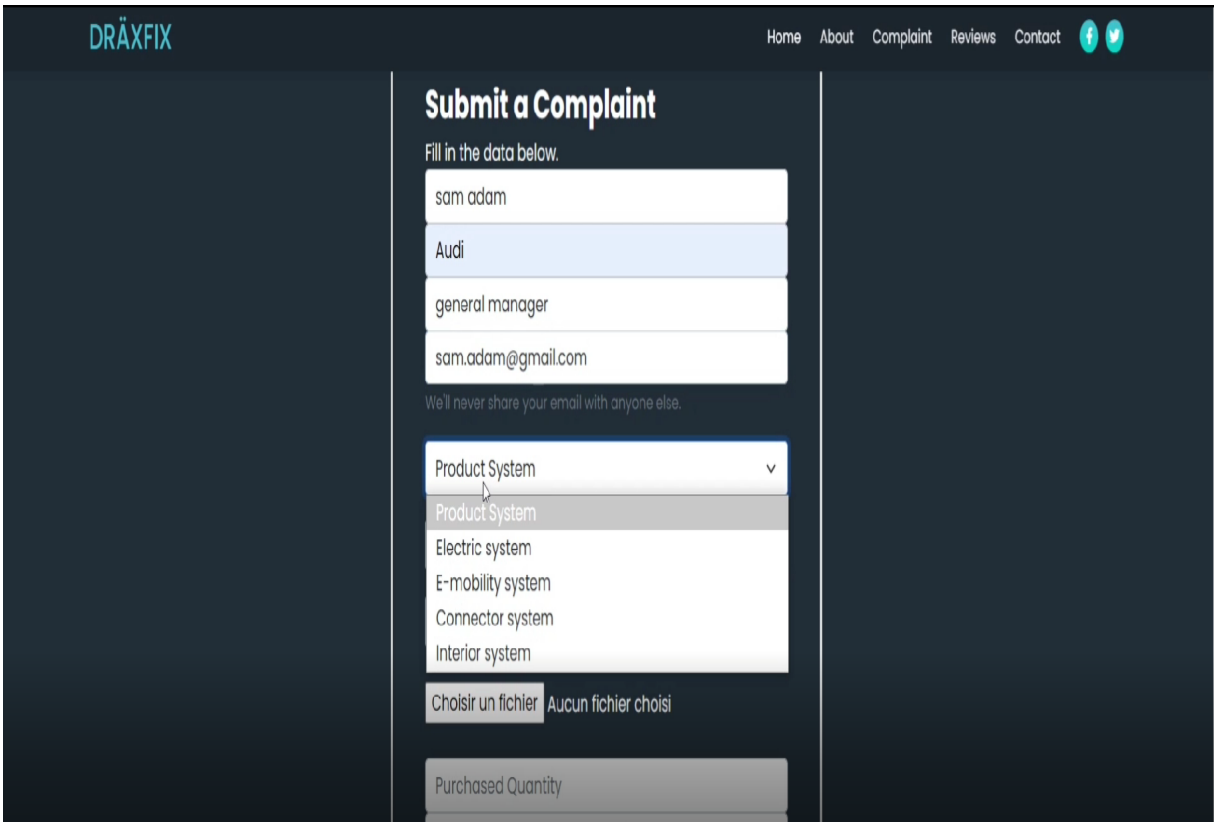

Figure III.5.4 - Product System

After that ,customer chooses the specific product.

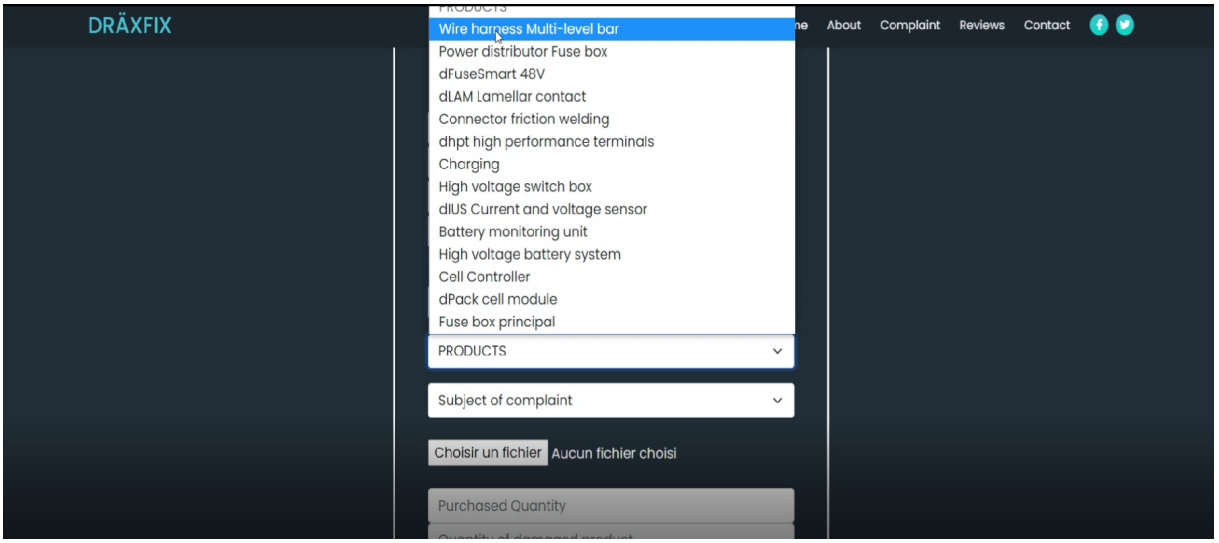

Figure III.5.5 - Product

Also, the customer have to choose subject of Complaint.

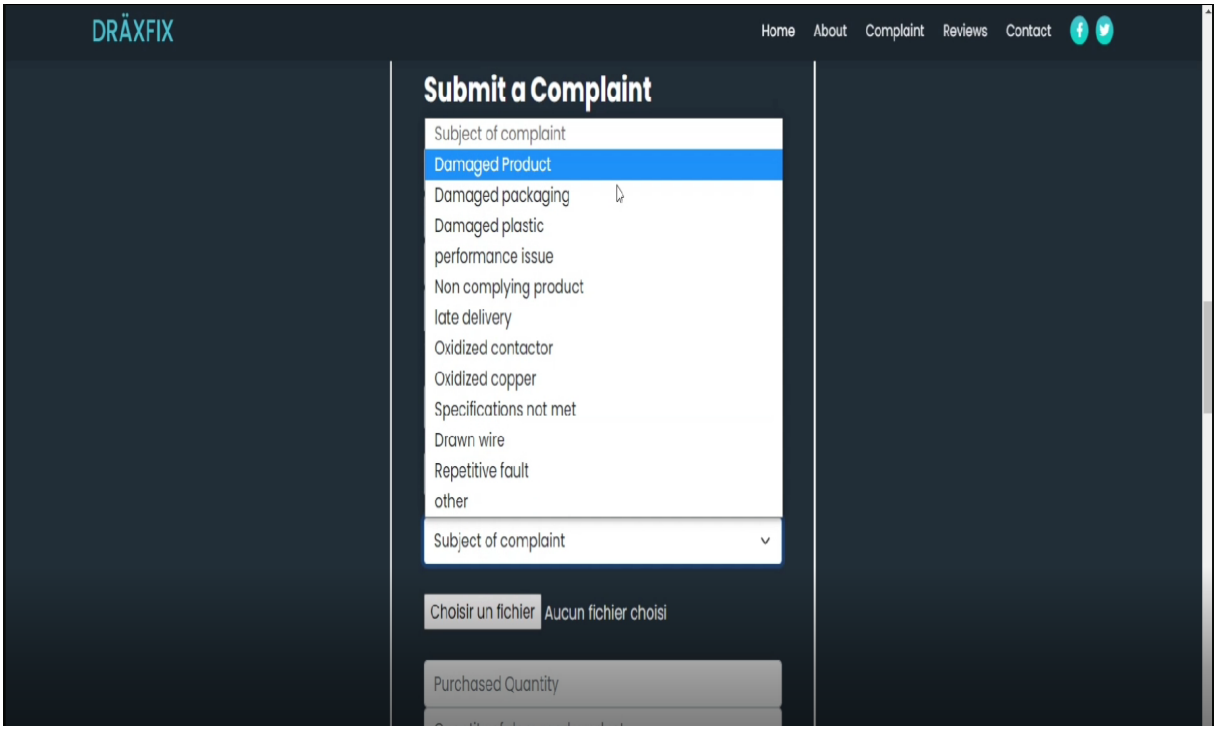

Figure III.5.6 - Subject of Complaint

The web application contains a couple of customers' reviews about the complaint service.

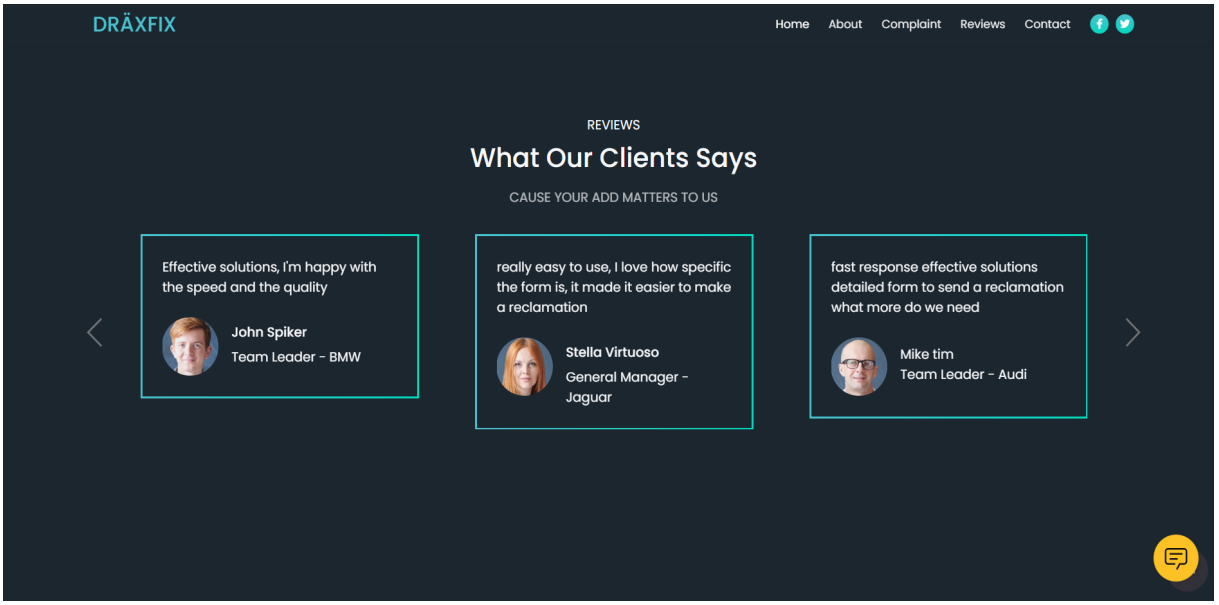

Figure III.5.7 - Reviews

The web application displays certain numbers about Draexlmaier

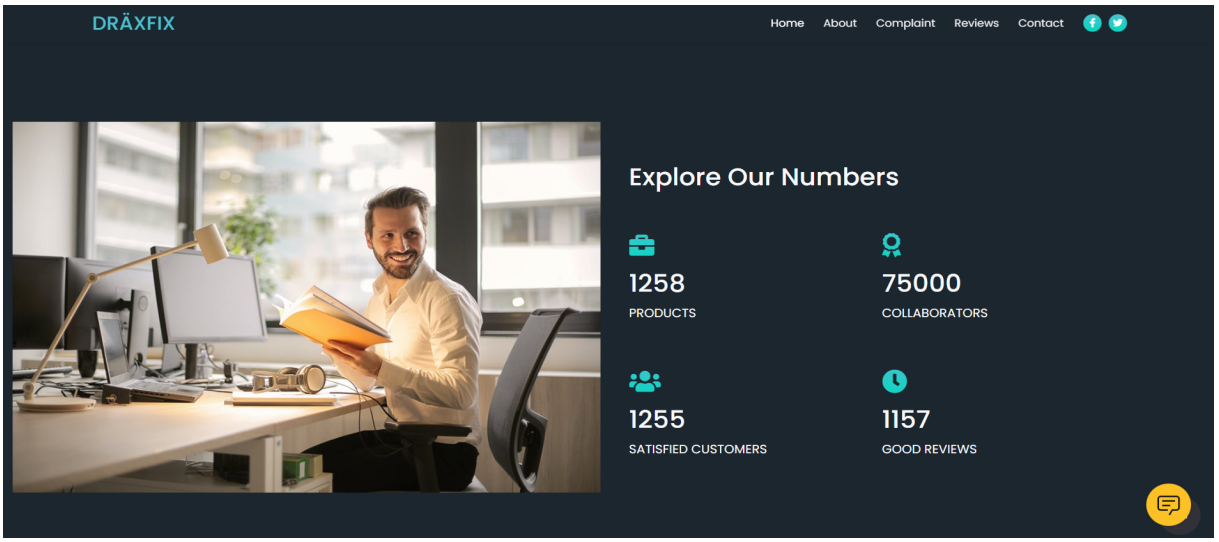

Figure III.5.8 - Numbers

The web application displays certain numbers about Draexlmaier

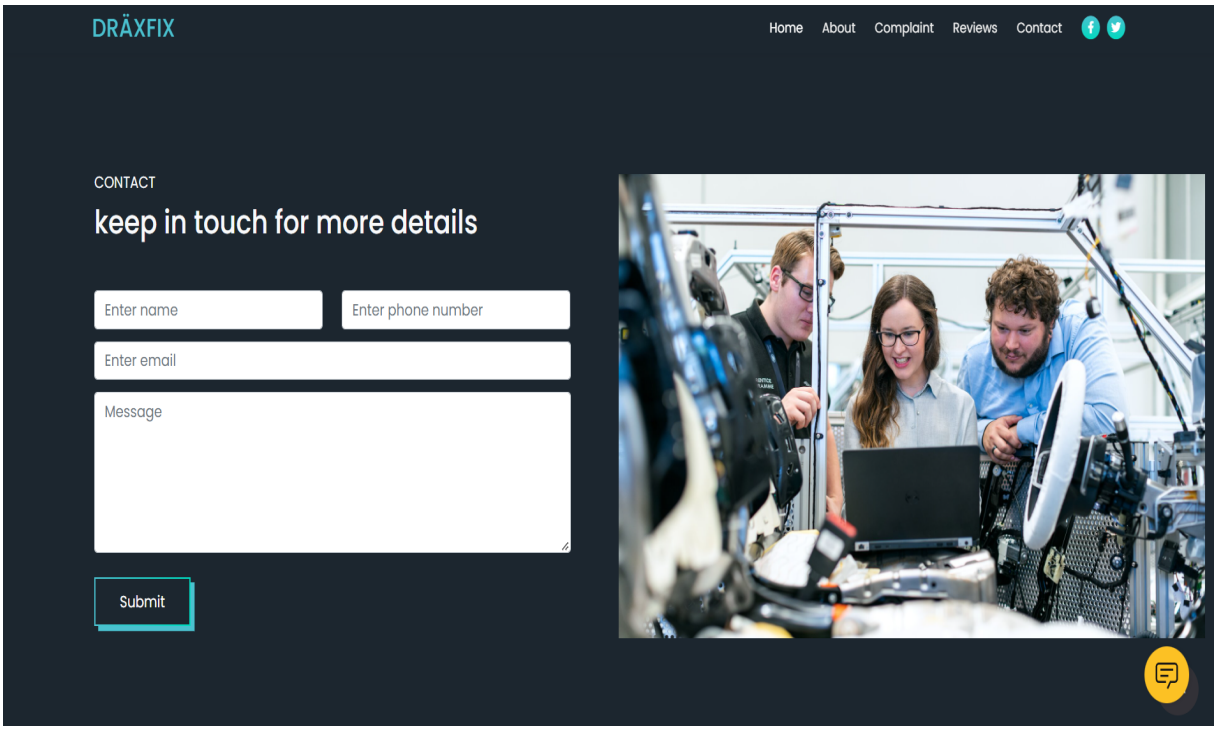

Figure III.5.9 - Contact

III.6 Conclusion

In this chapter, we have first presented our working environment and then exposed some of the interfaces of the site.

General Conclusion

In conclusion, we can mention that this end-of-year project has allowed us to master design and realization tools, namely software that we have not had the chance to know before.

My project consists in analyzing, designing, and developing a platform of complaints to optimize customer service for Draexlmaier's customers.

I have also, throughout this report, exposed the different stages of the

of the development life cycle of our site: I have exposed the study of the existing which led us to the analysis and the specification of the needs, I also presented the design of the project which was crowned by the implementation of the platform.

I hope that my work has accomplished its objectives, but, like any other work, it cannot pretend to be perfect.

Finally, I would like to point out that this application is scalable and extensible and can always be enriched and developed.

In this term, I have as perspective the addition of a mobile application associated with this platform that can boost Draexlmaier's customer service.

Complaint Web Application for Draexlmaier

Résumé

De nos jours, l'optimisation du service client est devenue une préoccupation pour les grandes entreprises telles que Draexlmaier.

Mon projet est donc une application web qui permet aux clients de Draexlmaier de soumettre une réclamation concernant un service ou un produit par un formulaire détaillé et avec l'aide d'un chatbot IBM Watson.

Mots clés :Draexlmaier,Réclamation,formulaire ,IBM Watson.

Abstract

Nowadays Customer service optimization has become a concern for big companies such as Draexlmaier.

Therefore my Project is A web application that allows Draexlmaier Customers to submit a complaint about a service or product through a detailed form and with the help of an IBM Watson chatbot

Keywords :Draexlmaier,complaint,form ,IBM Watson.