AMYSME

A mystified smart experience

Title :

AMYSME case study

A case study by :

Hmaidi Hamza

Jaziri Mayssa

Kammoun Rami

Kanoun Zied

Krichen Marouen

Lassoued Yassine

Latrous Ramzi

Mahjoub Hamza

Mahou Marouen

Mahrouk Wadhah

Marzouk Tahar

Mekni Toujeni Wassime

Host :

M. Lilia Sfaxi

Academic Year : 2022/2023

Contents

Figures . . . iii

Acronyms . . . iv

Introduction . . . 1

1 Enterprise overview **2**

1.1 What is AMYSME ? . . . 2

1.2 The problems it faces . . . 3

2 Architecture description **5**

2.1 Functional cartography . . . 5

2.2 Applicative cartography . . . 6

3 Project phases **7**

3.1 preliminary phase . . . 7

3.1.1 Business Goals Associated with the Driver Profit . . . 7

3.1.2 Application Usage View . . . 8

3.1.3 Application Landscape . . . 9

3.1.4 Data Dissemination Diagram . . . 10

3.2 Phase B: Baseline Business Architecture . . . 11

3.2.1 Organization View . . . 11

3.2.2 Business Function View . . . 12

3.2.3 Application Co-Operation View . . . 13

3.3 Phase C: Baseline Information Systems Architectures (Data) . . . 14

3.3.1 Information Structure View . . . 14

4 Realization **16**

4.1 Project Design and Architecture . . . 16

4.1.1 Project Architecture . . . 16

4.1.2 Class Diagram . . . 17

4.1.3 Rabbit Mq architecture . . . 17

4.2 Development Tools . . . 18

4.2.1 Software environment . . . 18

4.2.1.1 Integrated Development Environment (IDE) . . . 18

4.2.1.2 Database design tool . . . 18

4.2.1.3 Version Control System (VCS) . . . 18

4.2.2 Development Technologies . . . 18

4.2.2.1 Angular . . . 18

4.2.2.2 ReactJs . . . 19

4.2.2.3 NestJs . . . 19

4.2.2.4 Python . . . 19

4.2.2.5 RabbitMq 19

4.3 Implemented Services 20

4.3.1 Web Portal Service 20

4.3.2 File Transformation Service 21

4.3.3 Reports Generation 21

4.3.4 Human Resources Service: CV evaluation and employees management 22

4.3.5 Client Demand Handling Service 23

4.4 WorkFlows 24

4.4.1 workflow 1 24

4.4.2 workflow 2 24

4.5 Further Analysis 25

4.5.1 Department of finance 25

4.5.2 Department of human resources 25

4.5.3 Department of marketing and communication 26

5 Conclusion **27**

List of Figures

1.1 Different departments and interactions 3

2.1 Functional Cartography 5
2.2 Applicative Cartography 6

3.1 Business Goals Associated with the Driver Profit 8
3.2 Application Usage View 9
3.3 Application Landscape 10
3.4 Data Dissemination Diagram 11
3.5 organization view 12
3.6 Business Function View 13
3.7 Application Co-Operation View 14
3.8 Information Structure View 15

4.1 Project Architecture 16
4.2 Class Diagram 17
4.3 Rabbit Mq Architecture 17
4.4 Git logo 18
4.5 Github logo 18
4.6 Angular features 19
4.7 Waiting time 25
4.8 Departments portions 26
4.9 Purchase percentage divided by access portals 26
4.10 The interval of sales prices per social status 26

Acronyms

AI Artificial Intelligence. 23

API Application programming interfaces. 19, 21

CLI Command Line Interface. 19

CV Curriculum Vitae. ii, 22, 23

DMS Data Management Service. 21

DOM Document Object Model. 18

FP Functional Programming. 19

FRP Functional Reactive Programming. 19

GIT Global Information Tracker. 18

HTML Hypertext Markup Language. 18

IDE Integrated Development Environment. 18

OOP Object Oriented Programming. 19

RBAC Role Based Access Control. 20

UI User Interface. 19

VCS Version Control System. 18

Introduction

Nowadays, it is impossible to turn a blind eye on the various challenges a company has to handle in order to ensure its survival through the complex and entangled issues it faces. Challenges are a reality, facts that should be considered as an opportunity to take a step forward for companies.

As part of the urbanization project, this report revolves around an imaginary company we managed to conceive and dive into its different details. The company deals with a set of problems concerning basically the interactions difficulties between its different departments as well as its external communication.

This project aims to find solutions to resolve these problems while following a certain methodology: we first give an overview on the company, its architecture,activities and problems.Then, we try to dive deeper into its different angles for sake of identifying the causes and details of those problems. Finally, we proceed to the execution stage where we implement our solution that is resilient to those kinds of problems we mentioned at first.

Chapter 1

Enterprise overview

1.1 What is AMYSME ?

AMYSME is an acronym that stands for "A MYstified SMart Experience".It is the name of our company that builds an end-to-end smart house solutions.

AMYSME's solutions cover the entire process of smart houses production ranging from the hardware components assembly (sensors/cords...) to softwares development and finally the on-site installation of the whole product and its maintenance.

Regarding the general architecture of the company, it has 7 departments revolving around its key activities and they are :

- Executive Board
- Human Resources
- Engineering : the Engineering Department represents the core of the company and it is composed of three sub-departments which are the Hardware engineers, Software engineers and the engineers taking care of installation and maintenance.
- Customer Service
- Supply
- Finance
- Research & Development
- Marketing & communication

The following figure is a small overview that represents the different departments and summarizes the interactions and data exchanged between them:

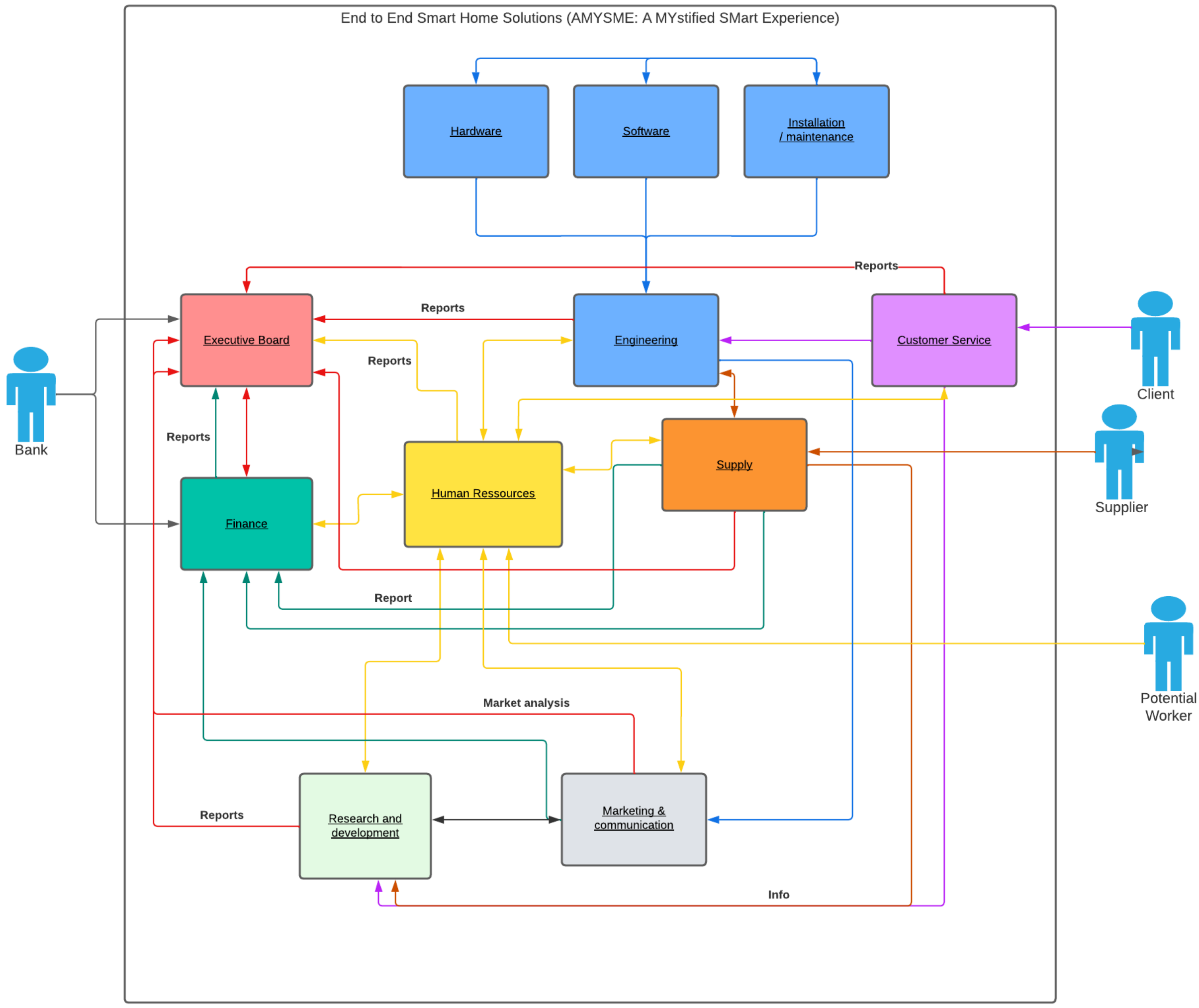

Figure 1.1: Different departments and interactions

1.2 The problems it faces

Based on the previous figure (Figure 1.1), it is easy to notice a set of obvious problems AMYSME has to deal with in order to take a step forward to progress:

- Every department is connected to the other departments directly through interactions and exchange of data which leads to the well-known "Spaghetti architecture".
- Some work is saved on papers like suppliers' bills for example. Therefore, it is necessary to find a way to save this data digitally.
- R&D can take too much time to collect needed information because the data that the department needs is scattered all over the company which leads to latency communication.
- Finance and Production are tightly coupled. Production can't move without finance.
- Material bill format can differ from a provider to another and certain departments can provide their own bills. It is crucial to standardize this data and pass it as an input to Finance Department.
- Data collection for directors should be easier for report generation.

- Data duplication across departments poses another huge problem. Therefore, Data synchronization & aggregation across departments are necessary.

Chapter 2

Architecture description

2.1 Functional cartography

AMYSME's different activities focus basically on 6 functional key parts which are:

- Overseeing: its focal points are Finance operations, risk management, quality control, workflow management as well as dealing with regulatory tasks.
- Resources and Support: contain different tasks such as taking care of the logistic requirements, human resources management, hygiene & establishment maintenance. This key activity covers also conducting Software/hardware training sessions as well as commercial management.
- Core Operations: it is the core of the company and it handles the activities of the company's engineers: Software Management, Hardware Management and Installation Management containing each set of tasks to be performed as shown in the next figure. Core operations cover as well Products audit.
- Access Service: takes care of the different types of communication between the company and its clients: Website, Mobile, on-site communication, and payment (cash/online...).
- Research & Development: is responsible for feedback analysis and generating reports about it as well as product improvement.
- Transverse: is the last functional key part. It basically revolves around dealing with the different documents circulating across the company and the internal and external collaborations.

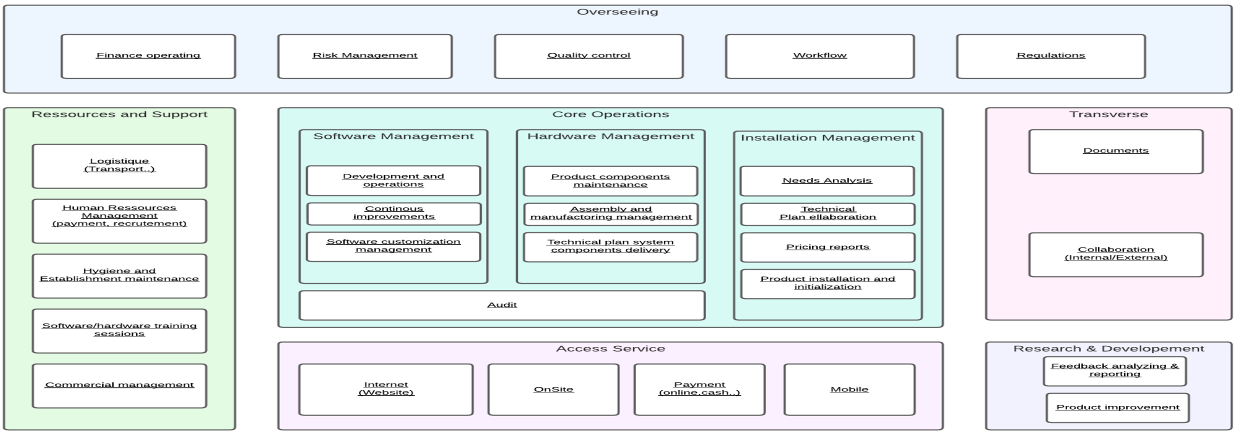
Overseeing
Finance operating
Risk Management
Quality control
Workflow
Regulations
Ressources and Support
Logistique (Transport..)
Human Ressources Management (payment, recrutement)
Hygiene and Establishment maintenance
Software/hardware training sessions
Commercial management
Core Operations
Software Management
Development and operations
Continous improvements
Software customization management
Hardware Management
Product components maintenance
Assembly and manufactoring management
Technical plan system components delivery
Installation Management
Needs Analysis
Technical Plan ellaboration
Pricing reports
Product installation and initialization
Audit
Transverse
Documents
Collaboration (Internal/External)
Access Service
Internet (Website)
OnSite
Payment (online,cash..)
Mobile
Research & Developement
Feedback analyzing & reporting
Product improvement

Figure 2.1: Functional Cartography

2.2 Applicative cartography

A good applicative cartography starts with a good structuring of the application components. in the following figure 2.2 we find a thorough, detailed cartography on the interchangeable systems within our company's information system. As shown in the figure 2.2 down below, our system is composed of 6 components: The data warehouse, the data synchronization, and aggregation component, the middleware, the access portals, the ERP and DMS.

When building and defining our system, we thought of those components to be sufficient and enough into building the data management system that will help refine and extract what we exactly need from the imputed data.

First, DMS or Document Management System is a component (or a service) of our system where we keep, deal, and generate the different documents that we are going to need in our company. This system is cross-department, it helps us generate reports, bills or receive bills. Also, store and send the documentations of our coded software.

Second, the ERP or Enterprise Resource Planning. ERP refers to a type of software that organizations use to manage day-to-day business activities such as accounting, procurement, project management, risk management and compliance, and supply chain operations. We also find a database in which all the outputs of the ERP software are stored.

The third, and most complicated component is the access portals. The latter defines the means of communication either internally or with external partners. It is divided into four branches: Our web/mobile application, supplier portal, engineers portal, and communication.

The web/mobile portal is from where our system can be contacted by our clients or the external world. It is a method where they could leave their feedback or make a purchase of any of our products and services.

The supplier portal helps our suppliers to have a look at our offers or the raw material we need to re-stock our warehouses.

The engineers portal is made for internal communication between the different actors in our manufacturing and engineering process.

And the communication portal is for communication between our departments. Fourth, our middleware which is the mean of communication between the data warehouse and the different entities of our system. In it we find some data cleaning and ordering, in order to prepare it for more aggregation and synchronization.

Fifth, Data synchronization & aggregation. In it we find an internal database where we will store our aggregated data, also, it is the final step where our data will be cleaned and fetched to the data warehouse. Finally, the data warehouse where our cleaned and fetched data will be stored. The next step will be the use of the latter for dashboard visualization and smart decision-making.

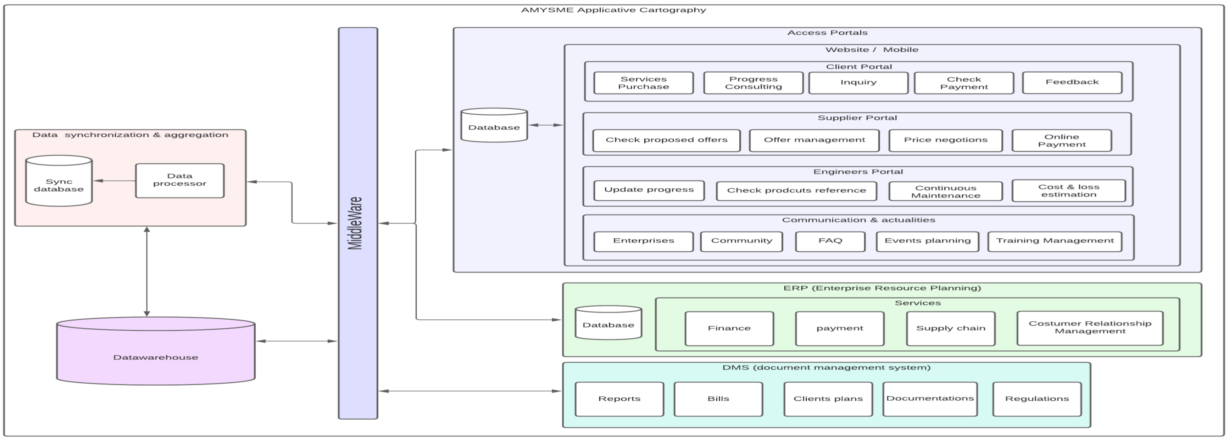

Figure 2.2: Applicative Cartography

Chapter 3

Project phases

Introduction

In this chapter we will introduce the different phases of the project following the principles of Togaf for the new system. They will define the different needed components for the implementation later on.

3.1 preliminary phase

3.1.1 Business Goals Associated with the Driver Profit

The main objective and concern when it comes to ensuring the sustainability of a business in the long term is the answer to the question: How to turn a profit?

Our goals for profit generation consist of three components as listed in figure 3.1: Reduction of costs, Increase of revenues and Explore new markets. Following our vision for cost reduction, we define several cuts and reforms as shown in figure 3.1 The first is trying to reduce 3rd services costs where we set the intention of not involving as many tiers of parties and letting our client or the supplier be our direct contact.
Secondly, moving more to a clean energy source will decrease overtime our energy consumption and make our company one of the green leaders in Tunisia. In addition to that, forming energy-aware personnel is our ultimate goal where we try to use machines and electrics as wisely as possible. With that, a whole strategy of enhancing our material and machinery to drive down maintenance costs is a necessity in the long run.
Third, encouraging competitiveness by vouching for better deals in terms of acquiring new primary materials, our strategy dictates that deals need to be transparent to all the personnel and exterior social organizations for the purpose of reducing fraud and corruption. Lastly, boosting productivity by automating tasks and keeping only motivated workers. Through a 10% cut in the working mass, the latter would help us tighten our budget and apply reforms to include only well-formed and qualified personnel.

The second goal is to increase our revenue through a strategy of sales improvement either by an economy of scale approach or by improvement in our pre-existing marketing strategy which would consequently lead to customer satisfaction derived from targeted needs and a reduction of error in our products and services.

The final business goal is exploring new markets. Through the lens of research and development, we get to boost innovation and target a new segmented customer base. The only solution towards that is first, an increase in the budget of R&D department and second, just patience for innovation is all that’s left.

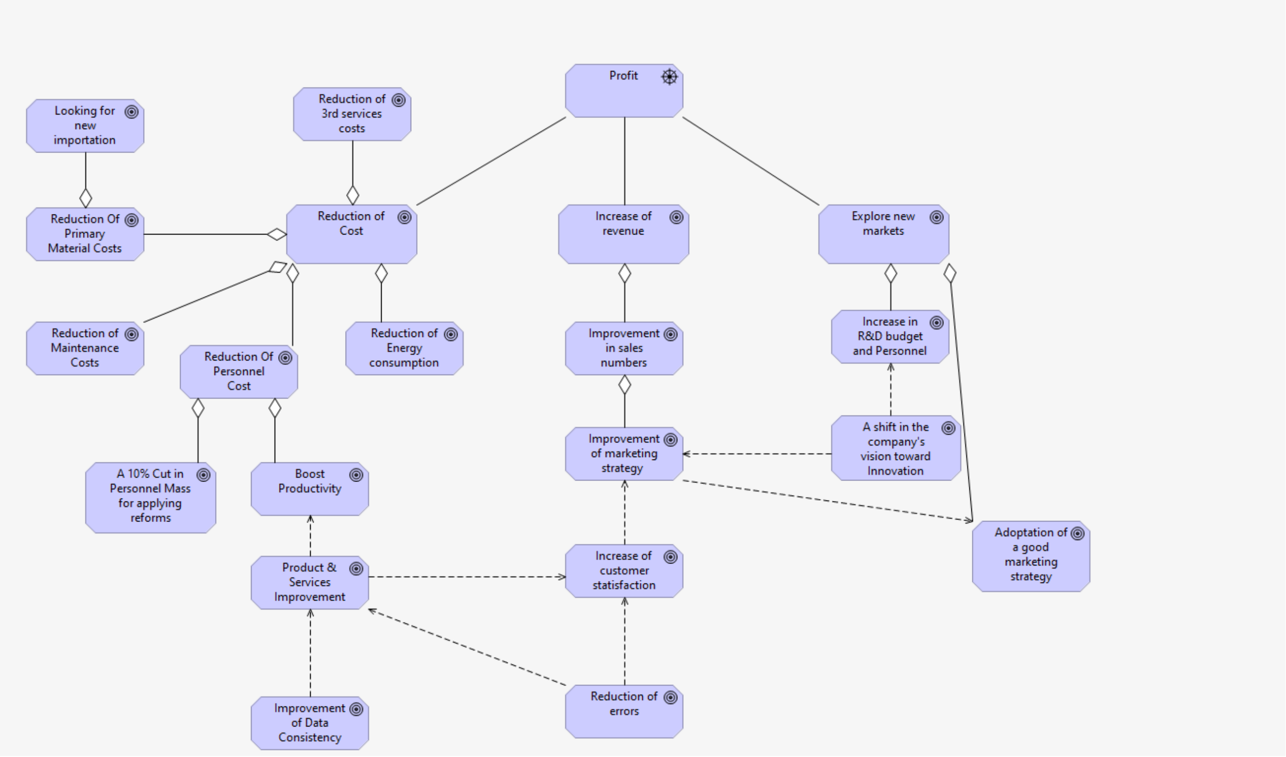
Profit
Looking for new importation
Reduction of 3rd services costs
Reduction Of Primary Material Costs
Reduction of Cost
Increase of revenue
Explore new markets
Reduction of Maintenance Costs
Reduction Of Personnel Cost
Reduction of Energy consumption
Improvement in sales numbers
Increase in R&D budget and Personnel
A 10% Cut in Personnel Mass for applying reforms
Boost Productivity
Improvement of marketing strategy
A shift in the company's vision toward Innovation
Adoptation of a good marketing strategy
Product & Services Improvement
Increase of customer statisfaction
Reduction of errors
Improvement of Data Consistency

Figure 3.1: Business Goals Associated with the Driver Profit

3.1.2 Application Usage View

In figure 3.2 we get to detail our business goals. With the increase in revenue, we get to improve our quality and increase our customers' satisfaction, which will help us reduce our errors and respond to the shift in the taste and demands of our customers. The second goal is the reduction of costs. The latter includes the reduction of the number of personnel, maintenance costs, and other unnecessary costs. In reducing the number of personnel, a 10% cut will take place immediately in a strategy of trying to minimize the cost of production.

In addition to that, the common use of application inter-departments will help us decrease the maintenance cost as the monitoring of machines and applications will be done by everyone. Finally, we can reduce other unnecessary costs when doing a study on the software used in our production chain. Can they be replaced by open-source software? If the answer to that question is positive then the next steps will be getting rid of those extra costs by using free-of-charge and effective software. The third and final goal of our list is exploring new markets. The latter goal can only be achieved through two sub-goals which are: increasing the budget of R&D to boost innovation and targeting new segments to enlarge our customer base. This can happen only if two elements are present in our plan: boosting innovation and focusing on the satisfaction of our customers.

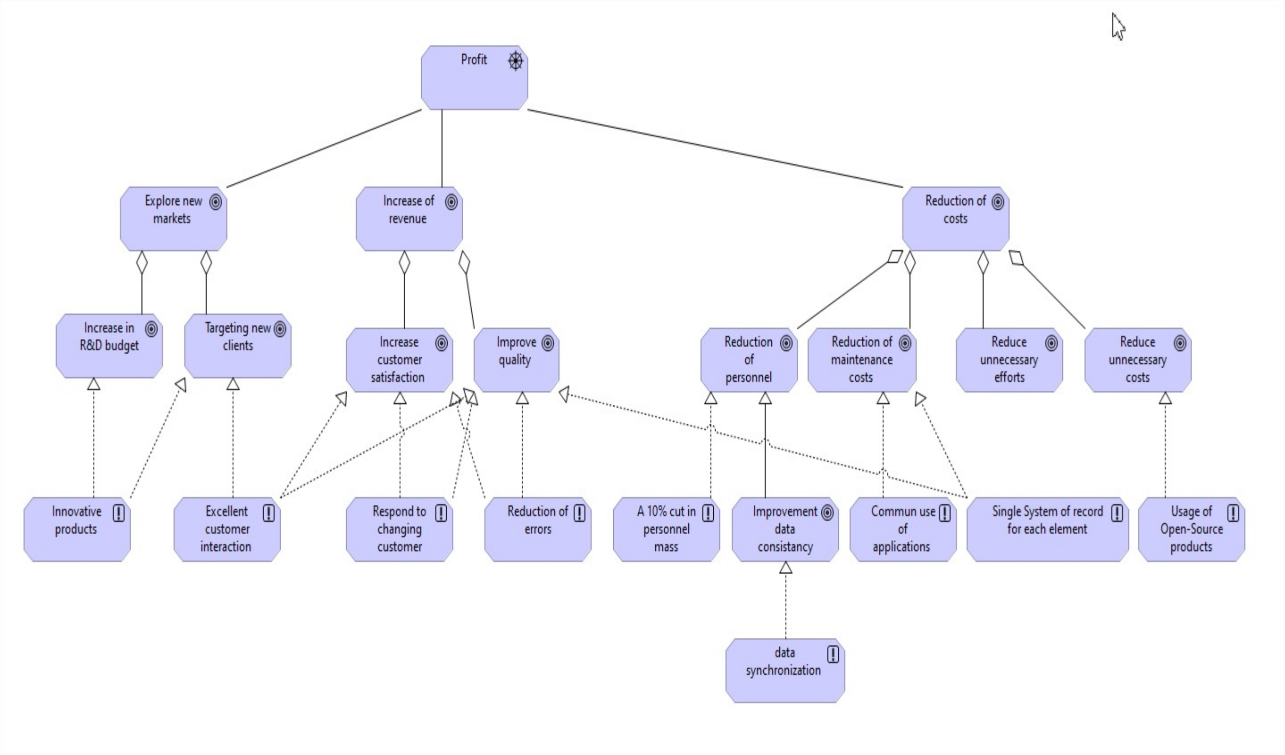
Profit
Explore new markets
Increase of revenue
Reduction of costs
Increase in R&D budget
Targeting new clients
Increase customer satisfaction
Improve quality
Reduction of personnel
Reduction of maintenance costs
Reduce unnecessary efforts
Reduce unnecessary costs
Innovative products
Excellent customer interaction
Respond to changing customer
Reduction of errors
A 10% cut in personnel mass
Improvement data consistancy
Commun use of applications
Single System of record for each element
Usage of Open-Source products
data synchronization

Figure 3.2: Application Usage View

3.1.3 Application Landscape

There is always a dependency between our departments when it comes to the applications or services we're providing. We define our major products as: Smart House application, Maintenance Service, and Customer service. The following services can be accessed directly through our outside portals: Web application, mobile application, and our call center for complaints or assistance.

Our finance team, also, is working on all products in accounting and generating a report at the end of each fiscal year. And finally, our document processing system helps in the management of the document cross-entities.

But not all of our products are managed by all of our business functions. Customer service for example is not included in the contracting function and the maintenance of the products since it is more focused on the satisfaction of our customers and how we can help them in the short term and the making of strategies of how we can make our products more efficient and beneficial to them.

In defining our products and who's responsible for the making or maintaining of each service we're providing, it is important to draw the line where each entity can practice their work and where the work of other entities starts.

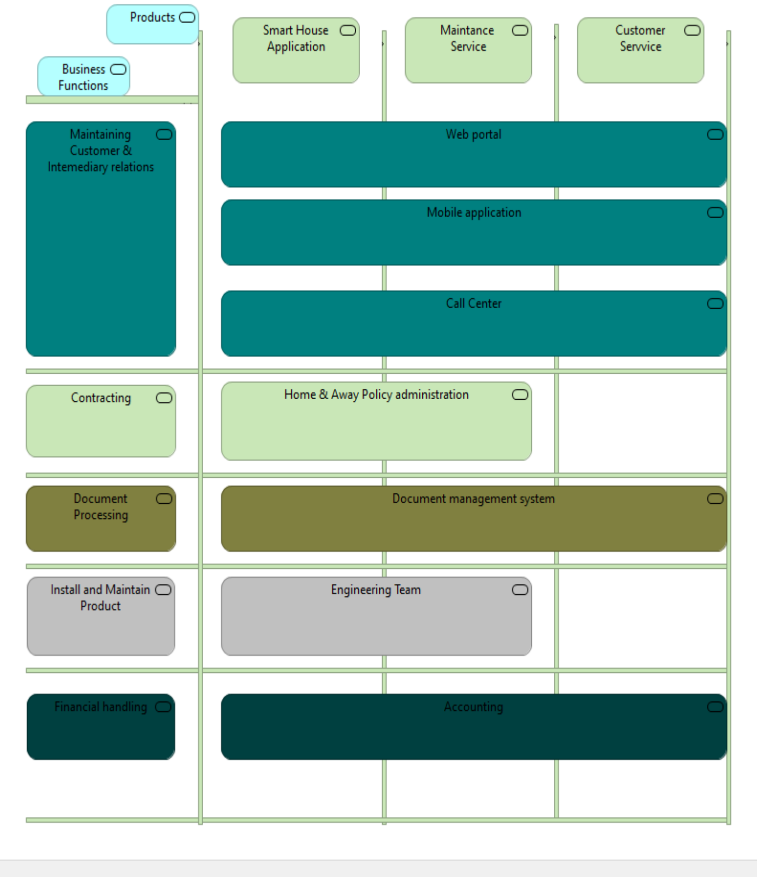

Figure 3.3: Application Landscape

3.1.4 Data Dissemination Diagram

The purpose of the diagram in figure 3.4 is to show the relationship between customer Service, Engineering service, and supply service components.

For the first component, Customer service, it has two main goals to ensure: client access and contract management, every service that our clients need to ensure their satisfaction, and the accessibility of our services. 3 data forms are extracted: client data, report's data and contract data.

For the second component, engineering service, it has one main goal which is to ensure order management. The following goal has two data outputs: product data and order data which are connected to the Product entity and Order entity respectively.

For the third and final component, supply service, is directly linked to inventory data which is linked to the item entity. concerning our entities of data, we have the main entity which is the client. The latter has two subjects dependent on him: the report concerning customer satisfaction and behavior and the contracts of items purchased by clients of our company.

In the contract we find two main objects: the product purchased and the order passed by the client, both objects are considered to be items.

Thanks to this diagram and after assigning business value to data, an indication of the business criticality of application components can be gained.

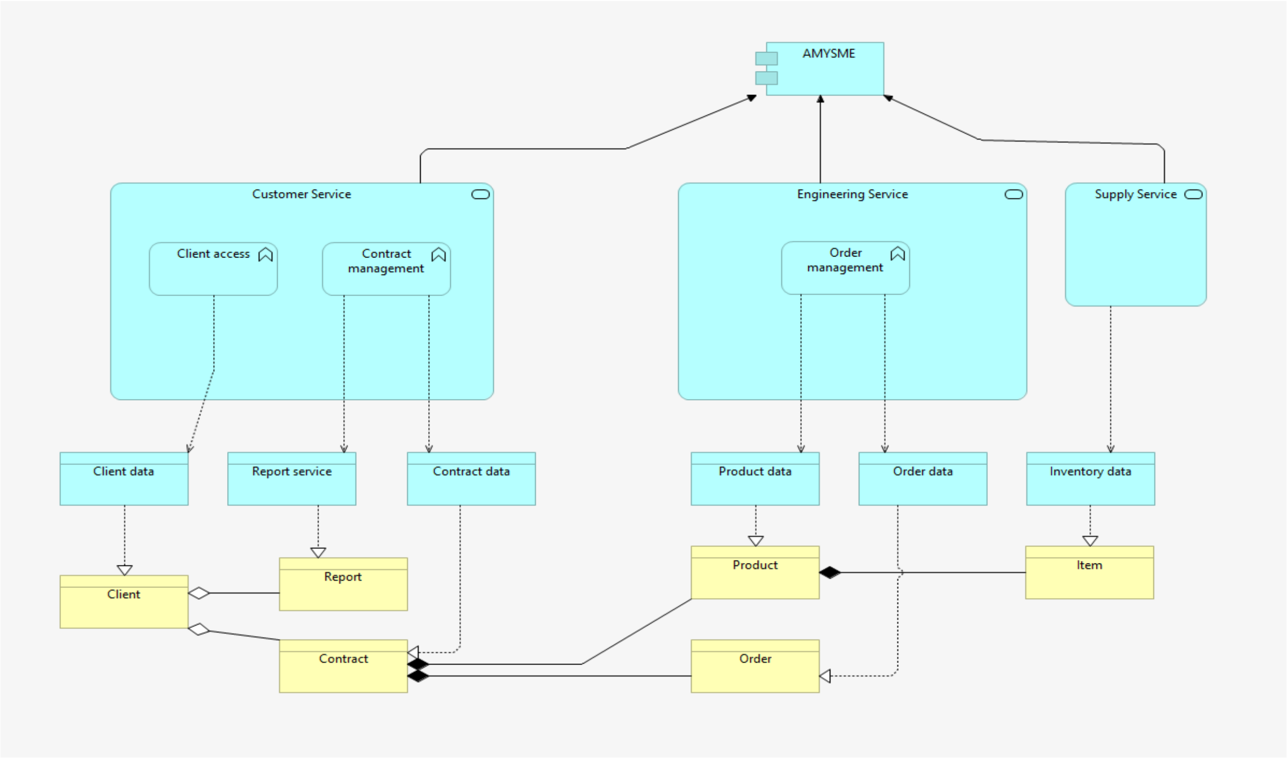

Figure 3.4: Data Dissemination Diagram

3.2 Phase B: Baseline Business Architecture

This section will cover the enterprise organizational structure, products, services, functions, processes and information. This will define the context for the data, application and technology architectures.

3.2.1 Organization View

The organization view diagram (Figure 3.5) details the internal composition of our enterprise (departments). Therefore we denote the existence of 8 departments including the engineering departments which is composed of the software team, hardware team and the installation/maintenance team. All these departments exists within the main and only local of the enterprise.

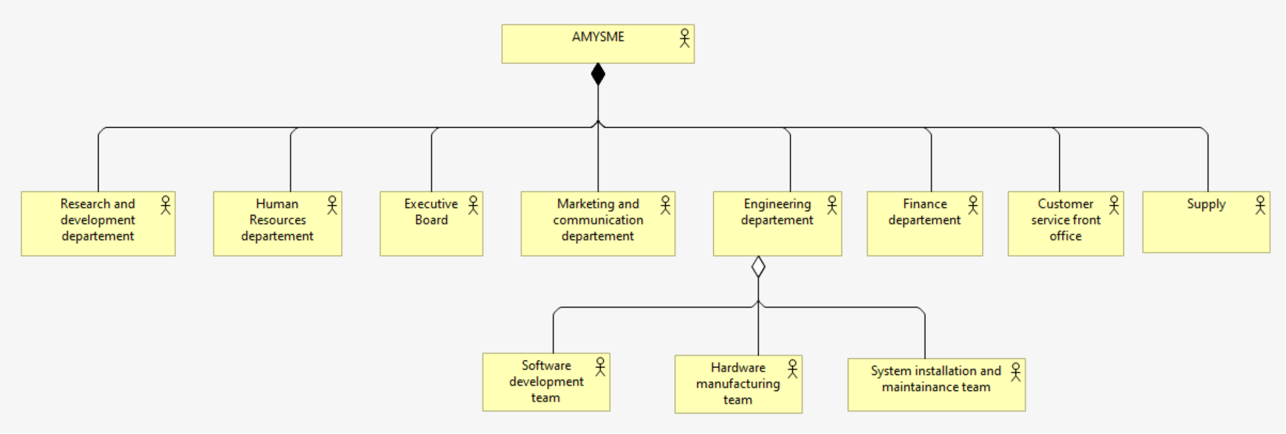
AMYSME
Research and development departement
Human Resources departement
Executive Board
Marketing and communication departement
Engineering departement
Finance departement
Customer service front office
Supply
Software development team
Hardware manufacturing team
System installation and maintainance team

Figure 3.5: organization view

3.2.2 Business Function View

The Business Function view diagram (Figure 3.6) shows the main business functions of AMYSME and the relationships by detailing the flow of information between them.
The main roles that we can distinguish are:

- **Client:** either a person or an enterprise that wants to install and smart house solution.
- **Intermediary:** the potential supplier of the enterprise in terms of raw materials and electronics.
- **Client's bank:** the financial institution nominated by the client in accordance with the debit instructions.

The main business functions that we can distinguish are:

- **Intermediary:** which includes the interactions between AMYSME and its clients; it handles demands, captures incoming claims, and gives customer supports...
- **Finance:** handles bills and controls the flow of value (money) in the enterprise.
- **Supply:** handles the relation with all the suppliers and monitors the enterprise stock.
- **Engineering:** The core function of AMYSME. Conducts needs analysis and contract software/hardware requirements generation. It also takes care of the installation and maintenance of the product.

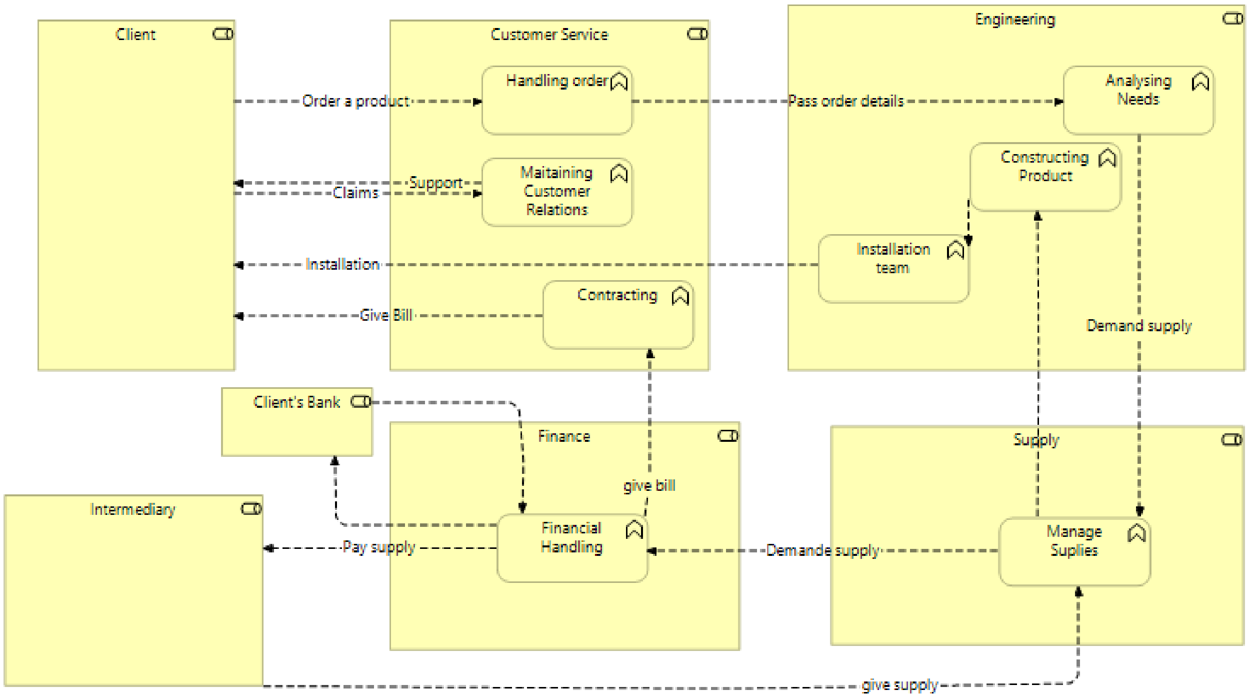

Figure 3.6: Business Function View

3.2.3 Application Co-Operation View

The Application Co-Operation View diagram (Figure 3.7) shows an overview of the application landscape and the dependencies between the applications. It expresses as well the co-operation of services that support the execution of a defined business process.
AMYSME applications are regrouped in 3 categories:

- **Front office:** regroup the set of applications that are created for public use (to assure communication with external roles such as individuals, businesses...).
- **Back office:** regroup the set of internal applications of the enterprise.
- **Shared Service Center:** regroup the set of application that are shared among the backoffice such the document management systemn (holds all the relevant documents for the processes).

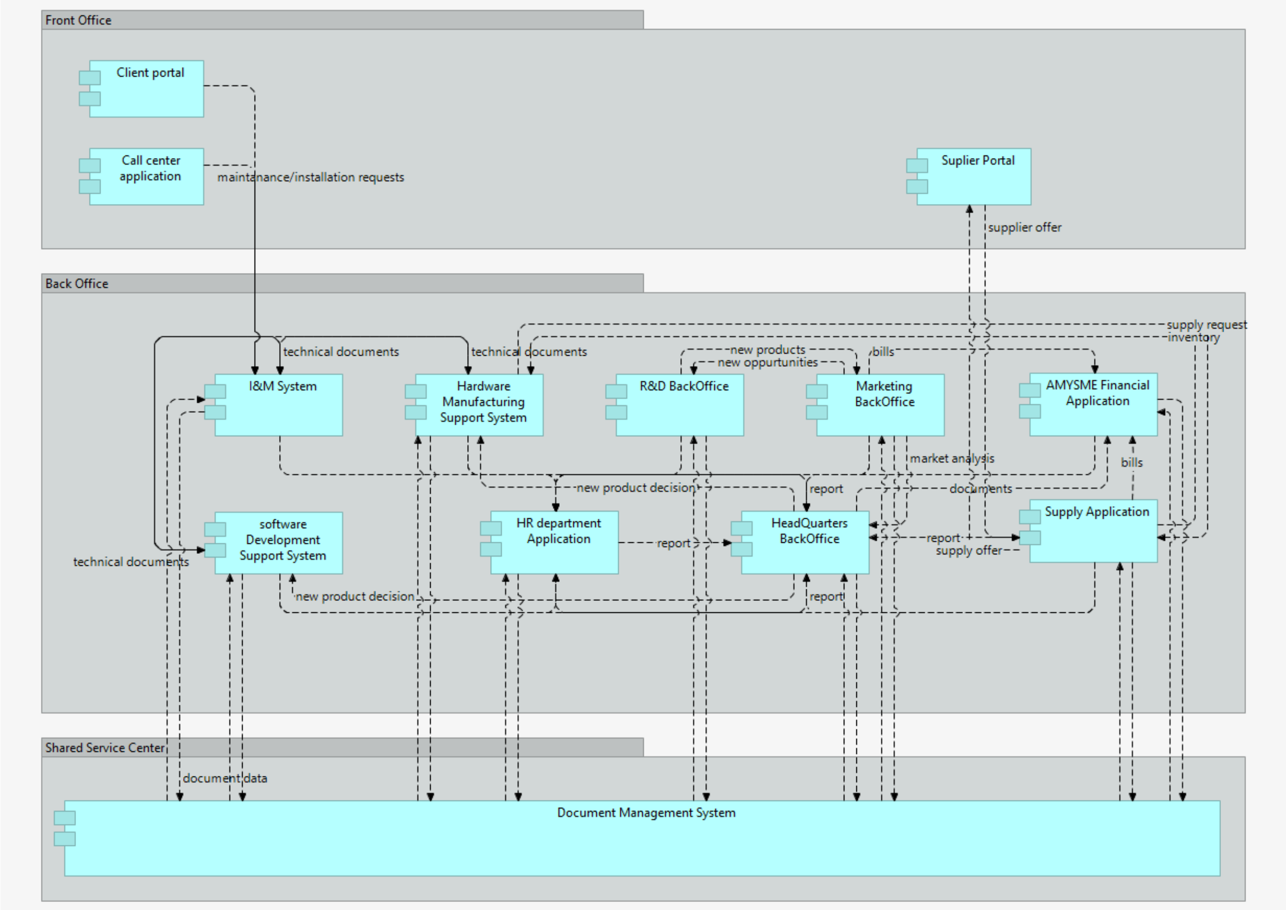

Figure 3.7: Application Co-Operation View

3.3 Phase C: Baseline Information Systems Architectures (Data)

This section will cover The Data Architecture that enables the Business Architecture and the Architecture Vision. The objective is showcase the Data and Application Systems domains in a way that addresses the stakeholder concerns.

3.3.1 Information Structure View

The Information Structure View diagram (Figure 3.8) shows the structure of the information used in AMYSME in terms of data types or class structures.
AMYSME has 10 types of information circulating within the system covering the entirety of the enterprise departments mentionned in the organization view (figure 3.5). There are 2 categories of data flows:

- **Within the same department:** in the engineering department between hardware and software teams as products concept. And the deliverd products data between hardware and installation teams.
- **Between different departments:** Which is the more common dataflow.

We note that the **reasearch** class is a key data element because it concerns a lot of departments such as engineering, executive board, marketing, etc... This is to be expected due to the importance of having detailed reports on the actual state of the enterprise, the desired state and the means to achieve it.

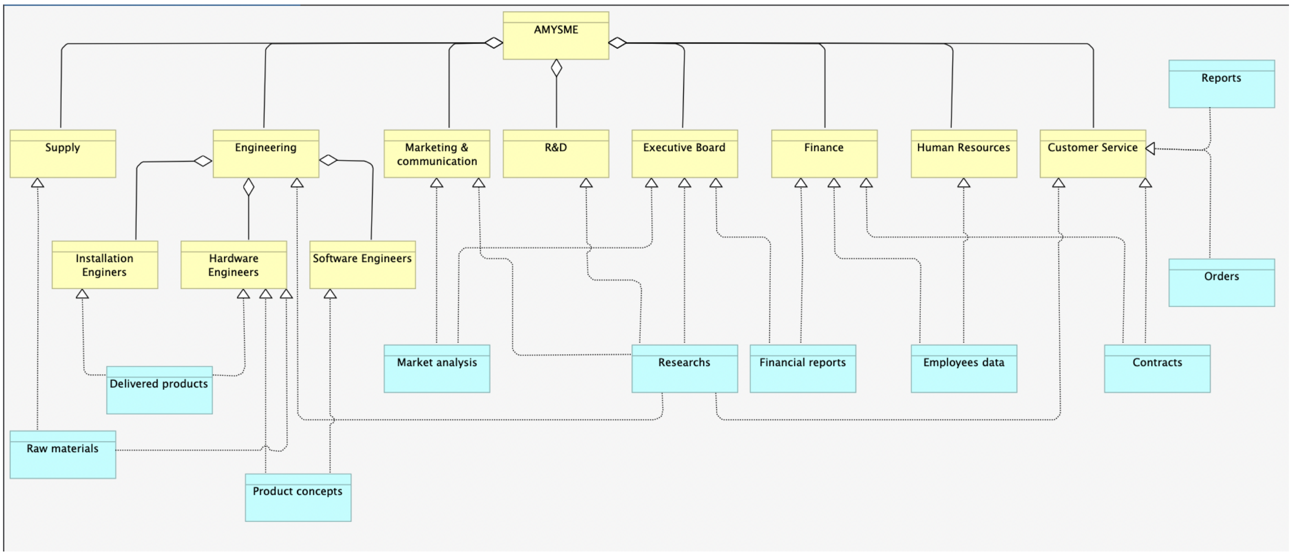

Figure 3.8: Information Structure View

Conclusion

In this chapter we introduced a multitude of Togaf diagrams mapping 3 essential phases which are the preliminary phase, the B phase and finally the C phase. In the next chapter we will introduce the proof of concept.

Chapter 4

Realization

Introduction

This chapter will cover the proof of concept of the system. It describes the implemented architecture as well as explains the different used technologies.

4.1 Project Design and Architecture

In this section, we will introduce the system architecture to be implemented and briefly describe its components as well as presenting the class diagram.

4.1.1 Project Architecture

Below (Figure 4.1) is the developed project architecture. It features the different services and in between relation.

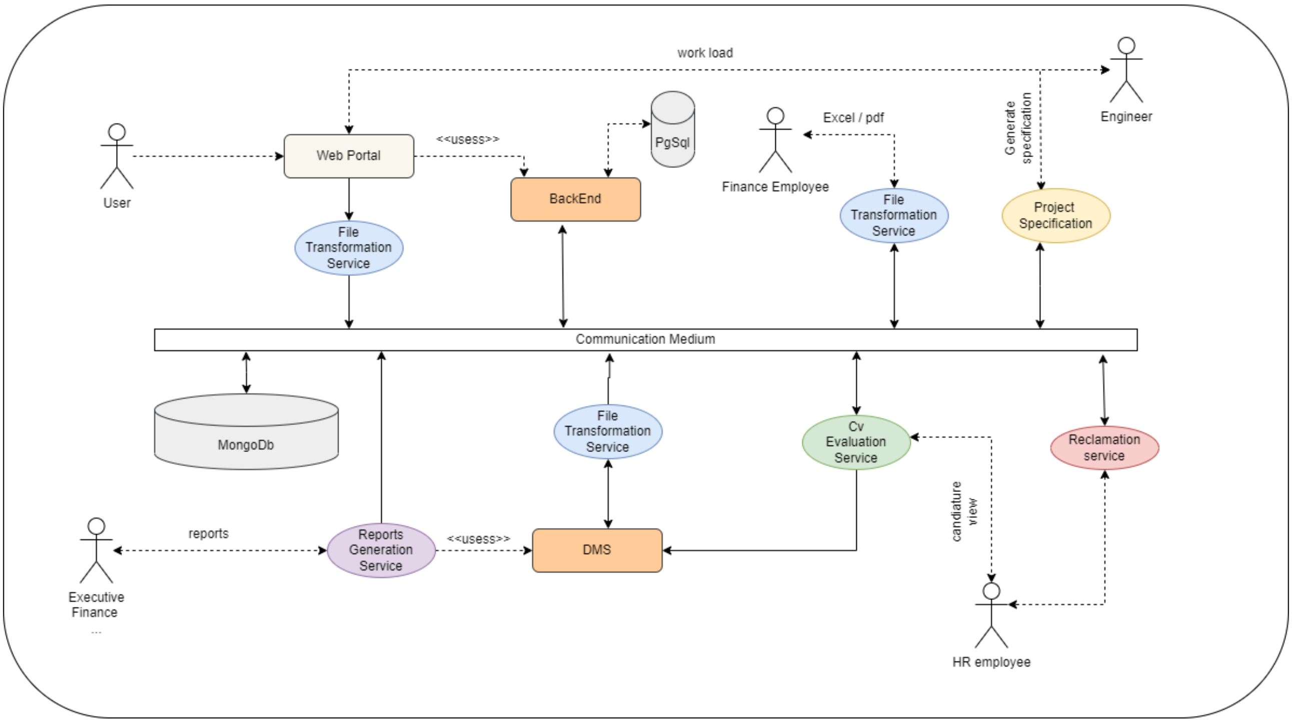

Figure 4.1: Project Architecture

4.1.2 Class Diagram

Below (Figure 4.2) is the class diagram of the project.It defines the different developed entities.

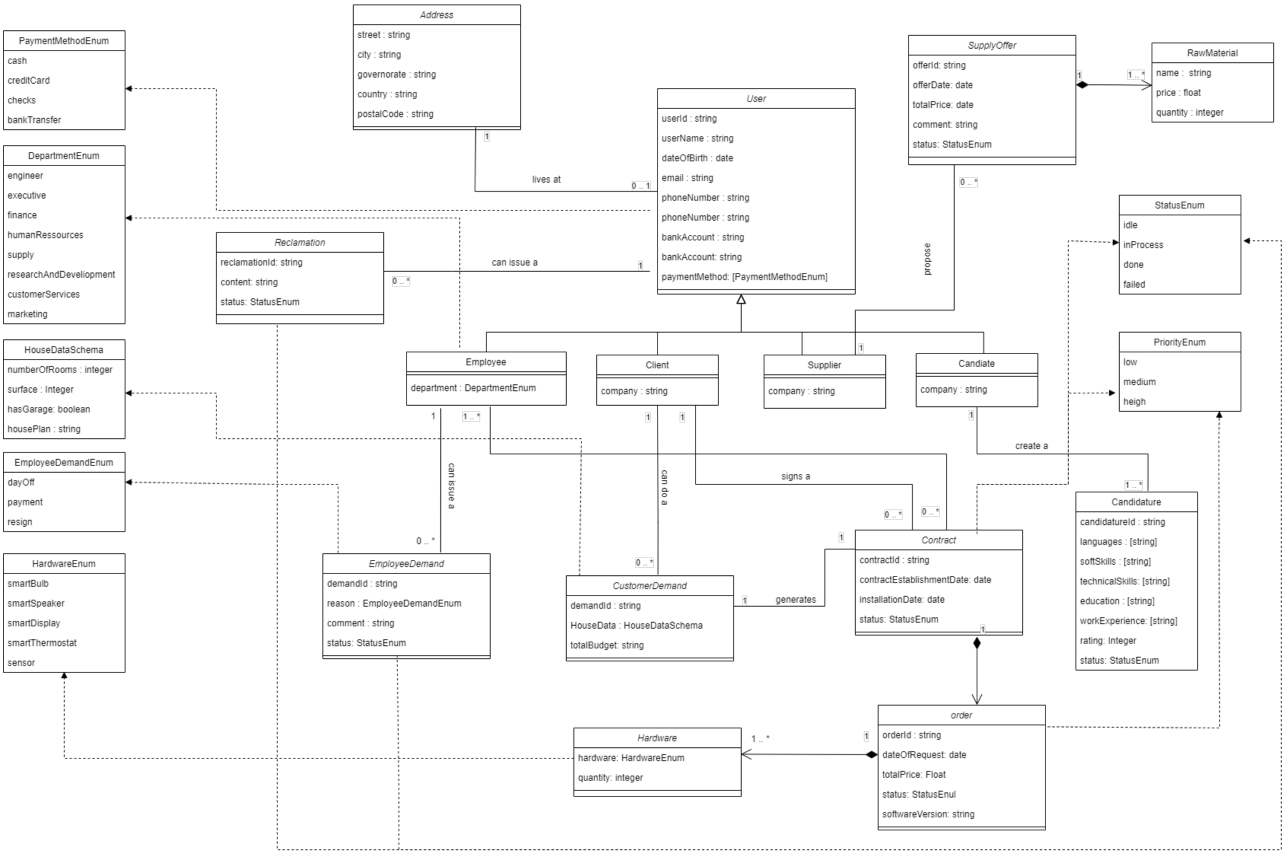

Figure 4.2: Class Diagram

4.1.3 Rabbit Mq architecture

Below (Figure 4.3) is the developed communication network architecture. It features the different queues and exchanges.

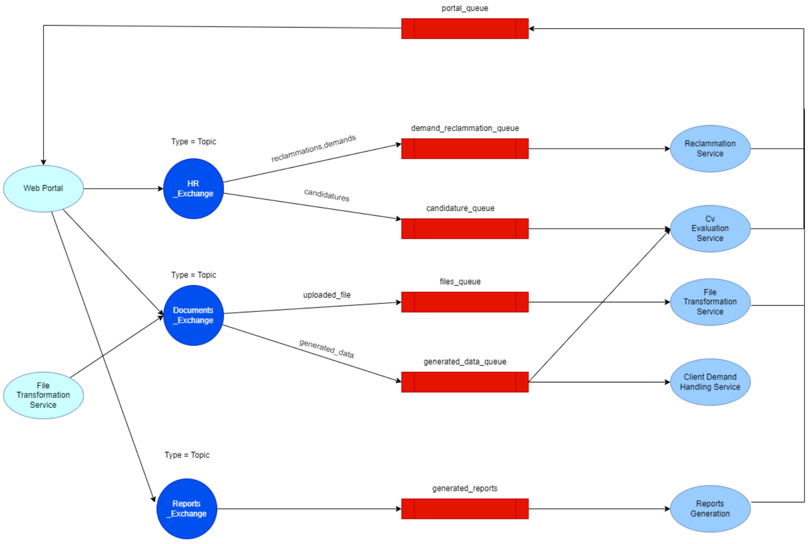

Figure 4.3: Rabbit Mq Architecture

4.2 Development Tools

In this section, we will introduce the software environment and all the technologies that were used in the development of different services. Later on, we will specify where each technology was used.

4.2.1 Software environment

4.2.1.1 Integrated Development Environment (IDE)

As an IDE, we used Microsoft's Visual Studio Code [**vsc**]. It is a lightweight source-code editor for Windows,Linux, and mac-OS. Debugging, syntax highlighting, intelligent code completion,snippets, code refactoring, and embedded GIT are among the features. It is highly customized (theme, keyboard shortcuts and preferences) and allow users to install extensions that offer new features.

4.2.1.2 Database design tool

To be verified

4.2.1.3 Version Control System (VCS)

in VCS we used GIT, Git is a free and open source distributed version control system designed to maintain code history. Git offers many features and it is the most common VCS nowadays. Its main advantage is its branching feature which enables its users to structure the application development into a clear workflow by adapting a branching model.we used Github as a git online platform because it is the first online platform for hosting git repositories, easy to use and learn.

Figure 4.4: Git logo

Figure 4.5: Github logo

4.2.2 Development Technologies

4.2.2.1 Angular

Angular is a structural framework for dynamic web apps. it lets you extend HTML's syntax to express your application's components clearly and succinctly. This extension is established with a new syntax through a construct called directives. **Examples include** : Data binding, DOM,control structures for repeating, managing DOM fragments and attaching behaviour to DOM elements, Support for forms and form validations, Grouping of HTML into reusable components [**angular_intorduction**].

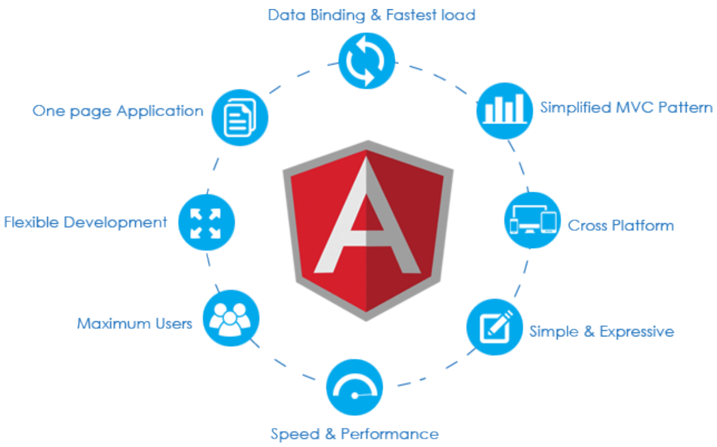

Figure 4.6: Angular features

4.2.2.2 ReactJs

React.js is an open-source JavaScript library developed by Facebook. It's used for building interactive user interfaces and web applications quickly and efficiently with significantly less code than you would with vanilla JavaScript. you develop your applications by creating reusable components that you can think of as independent Lego blocks. These components are individual pieces of a final interface, which, when assembled, form the application's entire user interface.

4.2.2.3 NestJs

Node.js is a cross-platform, open-source back-end JavaScript runtime environment that uses the V8 engine to execute JavaScript code outside of a web browser. its event-loop and non-blocking I/O operations allow code execution at a pace which offers high performance for real time applications. [**nodejs**].

NestJs represents a layer of abstraction of common Node.js micro-frameworks (Express/Fastify) but also expose their API to the developer. It is built with typescript and combines elements of Object Oriented Programming (OOP),Functional Programming (FP) and Functional Reactive Programming (FRP). Another advantage of NestJs is its flexible and well defined structure (controllers, modules, and providers).

4.2.2.4 Python

python is an interpreted, object-oriented, high-level programming language. Its high-level built in data structures, combined with dynamic typing and dynamic binding, make it very adequate for Rapid Application Development. The Python interpreter and the extensive standard library are available in source or binary form without charge with a big supporting community.

4.2.2.5 RabbitMq

RabbitMQ is a messaging broker - an intermediary for messaging. It gives your applications a common platform to send and receive messages, and your messages a safe place to live until received. It supports many programming languages and can run on various Cloud environments and operating systems. It offers browser-based UI for monitoring and management, and also comes with HTTP UI and CLI tools for operations. It is lightweight and easy to deploy on Cloud and premises.

4.3 Implemented Services

In this section, we will introduce detailed explanation of the different services of the system, its context and using what technology.

4.3.1 Web Portal Service

Problem

Keeping things simple is more appealing to clients, that why dividing enterprise services into a lot of component can overwhelm them. That's why the idea was to recentre the entry point of clients in a one package application.

Description

This service represents the core method of communication with the different actors of our enterprise and the entry point of the majority of our data. It is composed of a front end and a back end spanning over a lot of the enterprise departments such as customer services, supply, finance ...

Service Values

- Enhance the user experience and optimize performance.
- Offering all in one access to the majority of the enterprise services.
- Separation of concerns between the different users without presenting different and excessive entry points.

Actors

The actors that can use this service are the public user, client, candidate and employees.

Functionalities

- It implements an Role Based Access Control (RBAC) system with token management.
- Public user can view / consult home page for information.
- Client can fill in a form to demand a service.
- Client can leave a comment for the enterprise.
- Provide a safe payment method.
- Supplier can fill a demand for materials or propose an offer.
- Engineer can check the current progress of the work and their undone tasks.

Technologies

- **Front end**: using Angular.
- **Back end**: using NestJs.

4.3.2 File Transformation Service

Problem

In the big entangled architecture of our enterprise, its hard to keep track of the entirety of documents. There is also a lot of wasted time in storing those data format (Papers for example) and parsing it to present to each department what it really needs.

Description

This service comes to enhance the time and effort of processing different forms of data, analyzing its content and transforming it into a global format that can be used by the rest of the enterprise services. It covers a lot of departments mainly supply, finance, engineering...

Advantages

- Ability to use different data formats.
- Optimize data extraction and transformation time.
- Reduce human error and data loss.

Actors

This works as an autonomous service or one handled by employees (like in the supply department).

Functionalities

- Ability to parse different types of documents based on their structure (Excel, PDF, CSV).
- Transforming data into a global data format recognized by the different services (JSON).
- Save data to the Data Management Service (DMS)

Technologies

- **CLI aspect**: Python script.
- **Integrated Service**: Python API.

4.3.3 Reports Generation

Problem

In order for a business to thrive, the enterprise needs to keep track of what is happening in its environment and between its different components. Tracing, monitoring, analyzing, and taking the right decisions can keep the company afloat, then, facing the inevitable evolution. In AMYSME, one of the biggest problems for our direction is having a consistent date of reports delivery, adequate information presented for decision-making, and the integration of reports across all departments.

Description

This service is to enhance the decision-making process by automating report generation and aggregating all the data coming from the different components of the enterprise.

Advantages

- Aggregate data from different departments.
- Auto generate charts, graphics, tables, and other models of data views.
- Automate reports delivery.
- Reduce human error and delays.

Actors

this service is autonomous/ with minimum human interaction. It requires access to all the available data and it benefits the stakeholders.

Functionalities

- Read different data types and map it to the corresponding view.
- Generate coherent reports across all departments.
- Generate PDF files.

Technologies

- **PDF generation**: python.

4.3.4 Human Resources Service: CV evaluation and employees management

Problem

The bigger the enterprise, the more employees we need to keep everything running smoothly, the more difficult it gets to manage all the possible conflicts and demands. AMYSME suffer, as any other enterprise, from the tension that arises between its personal on a daily basis which reduce performance. We can also notice the candidature system can be improved via integrating bots.

Description

This service comes to enhance feedback collection in the enterprise, manage employees demands in a smart effective way while minimizing human contact. It also provide a bot to easily identify interesting profiles reducing time loss.

Advantages

- Reduce human interaction and conflicts possibilities.
- Reduce cv analyzing effort and time loss.
- Improve future profiles classification.
- Improve reclammation and demands collection/storage.

Actors

This service is specific to the Human resources department from the data collection point of vue and for the employees/candidates to provide the specified inputs.

Functionalities

- Reclamation's and demands management.
- Simplified candidature via forms.
- CV PDF format analyzer.
- A bot to review and classify .

Technologies

- **Cv evaluation**: Angular.
- **Reclamation and demands collection**: Angular.

4.3.5 Client Demand Handling Service

Problem

Treating a client demand to install a smart system in his/her house takes a lot of data in consideration which costs time. Writing the requirements specification plan, estimating budget and duration of the project can present a lot of human error.

Description

This service comes to enhance the time and effort of generating an installation plan for a specific client demand. It introduces a co-pilot based on Artificial Intelligence (AI).

Advantages

- Reduce project plan elaboration and delivery.
- Predict budget, installation duration, necessarily resources based on older projects.
- Reduce human error.

Actors

This works as an autonomous service or on call by employees (In the engineering department).

Functionalities

- Generate auxiliary data more efficiently (description, metrics, metric based characteristics).
- Generate prediction based on the demand type and client profile.
- Give the engineer the option to modify if necessarily.

Technologies

- **CLI aspect**: Python.
- **Web aspect**: Flask.

4.4 WorkFlows

In this section, we will introduce the different implemented workflows the will show the relation between the services mentioned in the precedent section.

4.4.1 workflow 1

This is a first possible workflow named "candidature":

- A user make an authentication.
- He/she fills a form and sends a CV.
- The CV is transferred to the file transformation service.
- The Evaluation of the CV is done.
- At a point in time an executive can generate reports.

4.4.2 workflow 2

this is another workflow named "Client management"

- A user make an authentication.
- He/she fills a demand form.
- The demand is handled by the project specification service.
- Engineer can access the data.
- Finance checks supplier and contract.
- Client complaints.
- Engineer demand against complaint.
- At a point in time an executive can generate reports.

4.5 Further Analysis

The visualization and reporting service is the third service in our system. When it comes to problems concerning the different entities in the company, many don't know exactly how to fix them, while they lack the global vision of each department in the company as a whole, they forget to visualize the functioning of each entity/worker and their role within it. Look at socio-economic background or education when it comes to solving human resources problems for example. In this section, we thought of solving those exact problems, the kind of problems that take some reflection in order to figure out what way to solve them. We're visualizing and working on the statistics of three departments: **Financial**, **human resources**, and **communication** departments.

4.5.1 Department of finance

For the financial department, we believe that finance is at the heart of the company, that's why it is crucial more than anytime before to focus on sales and their effect and maybe spot trends or seasonalities in order to boost or increase production at those times.

The following fig. shows the number of waiting days, we notice a large sum of our customers wait more each day after order of their products. The same amount of customers that wait one day to get their delivery is equal to the same amount of people who wait ten days to get their purchase which deducts us the believe that the delivery service is random and that we need to make it more coherent and work on a system of priority and order.

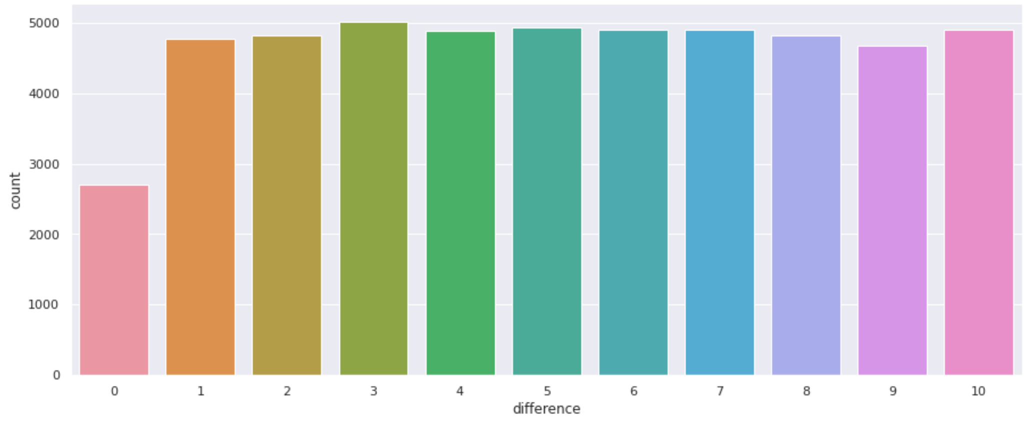

Figure 4.7: Waiting time

4.5.2 Department of human resources

The second part of our visualization is in the human resources department, in order to fix HR problems we need to understand our employees and their background.
The first chart is on the division of gender in our company, we can see her that there is

no equality between our staff since 70% of our workers are women, which implores us to hire more men and equalize our numbers.

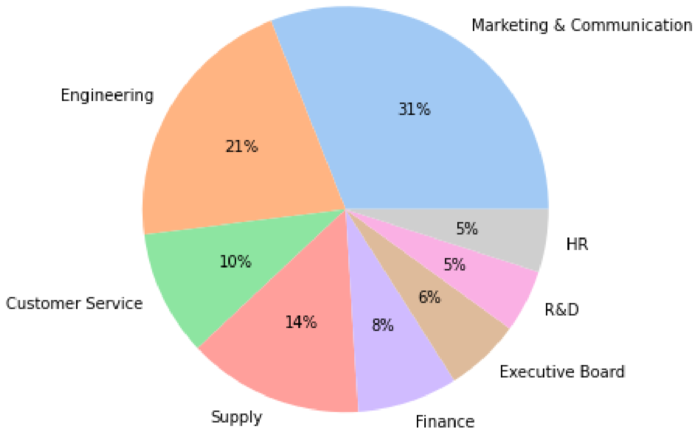

Figure 4.8: Departments portions

In addition to that, 30% of our workers belong to the department of sales and marketing and 21% in the operations department, by looking at our company's needs and customer base, we can confirm that we are satisfied by the number of employees in each department.

4.5.3 Department of marketing and communication

The final visualization for us is in the marketing and communication department where we will be studying our customers which will eventually help us achieve the goals we sat at the beginning of this report.

Half of our customers are married or soon to be, which helps us ensure that they will only demand more of our products. We're targeting married couples with children who would choose eventually to settle down in a smart home and become active clients to us, asking for our services often which will help us retain them. Targeting more of that audience is a must in our marketing strategy.

Mobile users are the future, we should target more mobile users, most of our purchases must be from mobile phones.

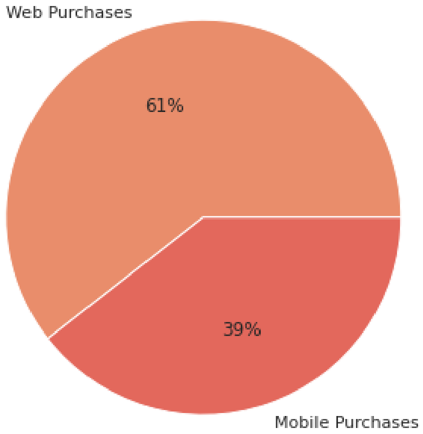

Figure 4.9: Purchase percentage divided by access portals

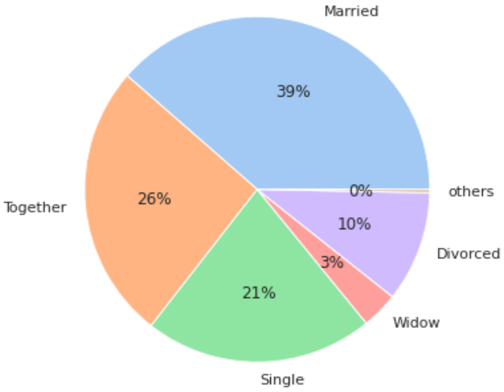

Figure 4.10: The interval of sales prices per social status

Chapter 5

Conclusion

This work was a group project for the urbanization subject we studied at university during the first semester of the final year in Software Engineering field. It is a work that is spread out over the whole semester.

The project represents a line of thought to deal with several tough problems and issues a company we imagined is facing during the execution and management of its different activities and departments.

We first start with a small overview on the company. We define what it is and set the kind of problems it is facing and we are trying to deal with.

We then move on to describing its architecture. We begin to dive more into the company using functional cartography and applicative cartography.

The next step is taking a look at the company from different angles in order to tackle its problems in a clearer way to identify the sources of those difficulties. This step is crucial since it is an opportunity to perform an analysis on the entire company, its components, its activities and interactions .

Finally we proceed to the realization where we deliver not only the project design and architecture but also the implementation of our solution on different services and testing this implementation on two workflows we managed to come up with to evaluate the efficiency of our solution .