
Software Requirements Specification

for

DIGITAL PLATFORM FOR INDUSTRY

Version 2.0 approved

Prepared by *Javad malek Shahkoohi and Karim Zaitov*

Industrial-Cloud Co.

September, 2017

Contents

1. Introduction.....	1
1.1 Purpose	1
1.2 Intended Audience and Reading Suggestions	1
1.3 Refactoring strategy.....	1
1.4 Product Scope	2
1.5 Glossary	3
2. Overall Description	5
2.1 Product Perspective	5
2.2 Product use cases	6
2.3 Product Functions.....	7
2.4 User Classes and Characteristics	11
2.5 Process Sequences	13
2.6 Operating Environments and Technologies	14
2.7 Design and Implementation Constraints	15
2.8 User Documentation	15
3. External Interface Requirements	16
3.1 User Interfaces.....	16
3.2 Software Interfaces.....	16
3.3 Communications Interfaces	16
4. Other Nonfunctional Requirements.....	17
4.1 Performance Requirements	17
4.2 Safety Requirements.....	18
4.3 Security Requirements	18
4.4 Software Quality Attributes.....	19
4.5 Business Rules.....	19
5. Other Requirements	Error! Bookmark not defined.

Revision History

Name	Date	Reason For Changes	Version
IC.V1.0	2017	<ul style="list-style-type: none">- Complicated UI/UX- Improving in a set of processes specially in the categorization section- Enhancing with a couple of new practical ideas like Marketplace, Channel, PGO¹ and etc.- Changing the development technology from PHP-MY/SQL to REACT-MONGODB-GRAPHQL- Adding some new functionalities	1.0

¹ PGO means Purchasing Group Organization

1. Introduction

1.1 Purpose

Since the first version of DIGITAL PLATFORM for INDUSTRY (IC.V.1.0) developed many years ago with the primitive ideas and old technologies, in the new version of the platform a bunch of enhancements and new features that will apply and can see them in the following sections. Also, all the platform would rebuild all the current functionalities and some new ones that contain data management subsystem, catalog, and categorization and industrial search engine.

1.2 Intended Audience and Reading Suggestions

The goal of this document is to make a clear path in transforming a monolithic application into microservices, is a form of application modernization, of platform from IC.V.1.0 to IC.V.2.0. Therefore, all the different types of readers should be able to understand the changes and new items of the project.

For readers such as developers and project managers is better to first look at the scope and new architecture of platform then they could follow by use-cases and other software diagrams.

Regarding the marketing staff, they should focus on the major customer acquisition channels of company and collect the end-user's feedbacks and they should make a paradigm for platform improvements.

1.3 Refactoring strategy

*The refactoring strategy composed of two main policies; first, **Stop Digging the current platform** that means stop making the monolith platform bigger. The second one is **Split Frontend and Backend** means, the monolithic application would to split the presentation layer from the business logic and data access layers.*

Based on the refactoring plan, shown in Figure 1, The platform decomposed into four vertical layers such that one in frontend and three layers on backend. Their modules use the REST API for communication between each other. Also respect to development process is stopped now and there would be no new feature. The layers are as follows:

- **Frontend user interface layer:** contains of three main customer communication channels which are web-browsers on desktop machines, mobile and Tablet devices for both Android and IOS platforms.
- **Authentication layer:** contains authentication, mapping and routing of incoming requests from any input UIs to an endpoint at the backend of platform. while some of communications to backend are need to be verified and authorized, the authentication segment is designed to satisfy these requirements.
- **Backend layer:** contains all the endpoints of the platform are in charge of response to incoming requests.

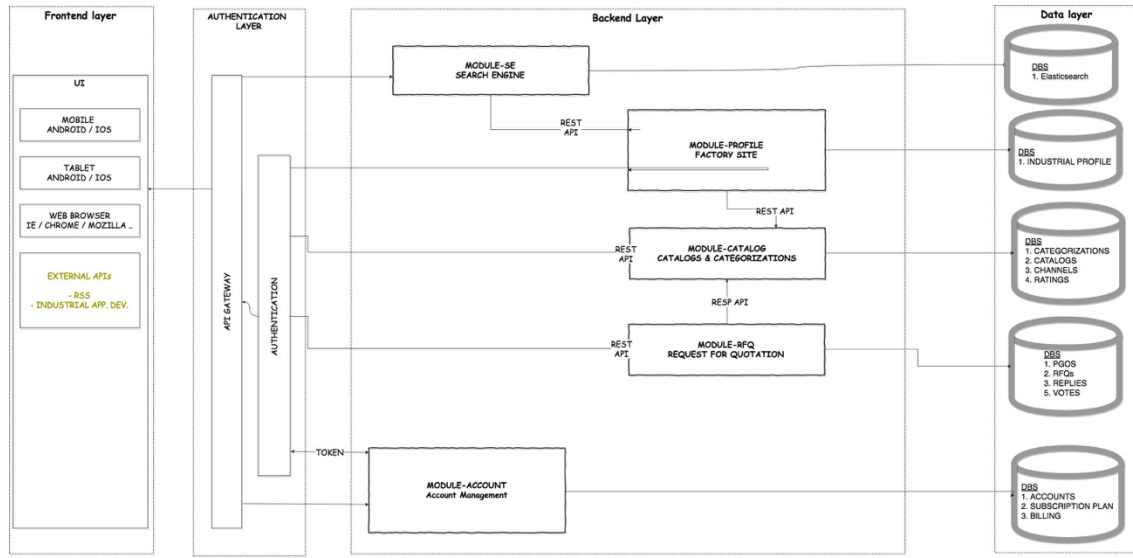


Figure 1: The layer approach of Refactoring Strategy plan

1.4 Product Scope

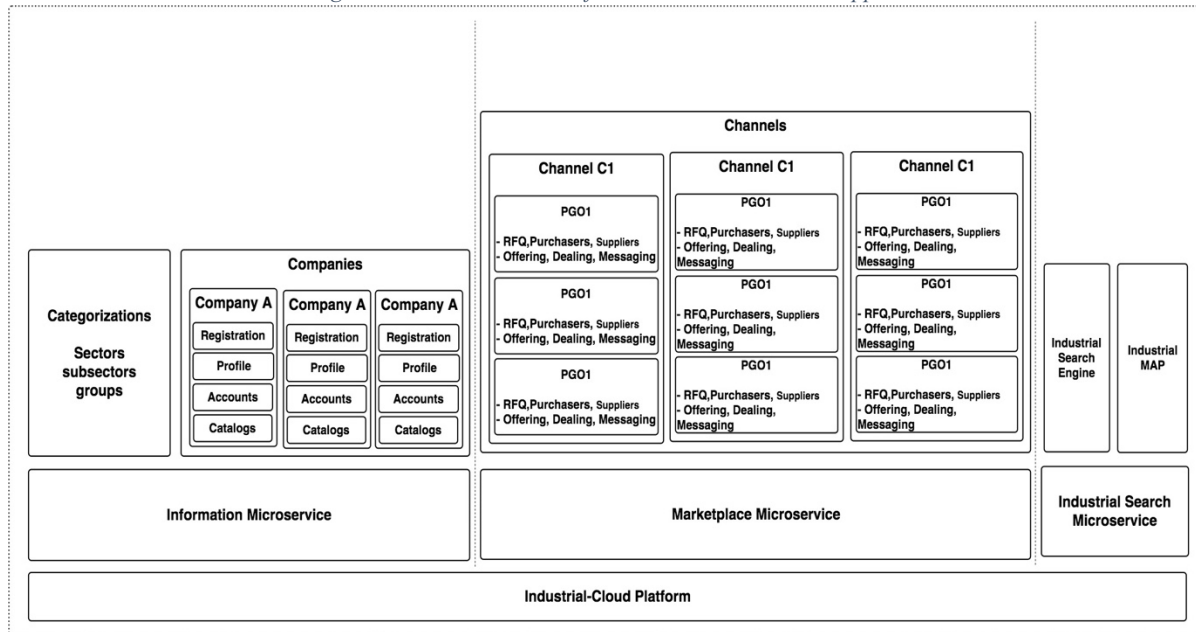
The IC.V.2.0 is a web-application that provides a reliable environment for companies to introduce themselves and try to extend their market. Then in the scope of platform are many microservices like a marketplace environment for companies from different sections of industry to make business easier and faster.

The environment of IC.V.2.0 composed of a couple of microservices, as you can see in the Figure2.

- **Information microservice:** is the data structure of platform and contains all the data interactions, analysis, storage and data management. Also the machine learning layer would be added there.
- **Marketplace microservice:** is the main service of platform where users can negotiate together respect to their requests and offers. This section contains a list of channels and PGOs.
- **Search engine Microservice:** is almost a separated component than two others and formed by two big modules one is industrial search that crawls through the web and another is industrial map of company's plants and their locations.

The potential customers are all the companies are producing any kind of industrial products or providing any sort of service for industry. Then the customer acquisition channels are industrial zones or associations, commercial offices, industrial yellow pages and etc.

Figure 2: The Architecture of IC.V.2.0 in microservice approach



More details will be in next sections.

1.5 Glossary

The list of terminologies would

	Term	Definitions
1	Company	<ul style="list-style-type: none"> - Company is the main player of platform, who uses all the prepared functionalities of the system. - Each company has two different types of user account Purchasing and Sales office account
2	Categorization	<ul style="list-style-type: none"> - A set of fields of industry that companies are interested to introduce themselves and their activities by these fields
3	Catalog	<ul style="list-style-type: none"> - A list of products, services, machines and processes that companies can supply in the platform for other companies.
4	Purchasing office	<ul style="list-style-type: none"> - The purchasing account of a company can register a new request and create a group in the platform. - Talks with suppliers through their offers to make a deal
5	Sales office	<ul style="list-style-type: none"> - Sales account of a company can apply for different requests of purchasing offices and makes an offer for every one.
6	Marketplace	<ul style="list-style-type: none"> - It is a place to publish a request/solution for a group of possible suppliers and customers. - The marketplace has a list of channels.
7	Channel	<ul style="list-style-type: none"> - Every field of industry is a sample of categorization and for each one of them, there is one unique channel. - Each channel carries a list of PGOs are related to the same category of industry.

		<ul style="list-style-type: none">- Only Administrator and Exclusive users can create a new channel in the market and invite other companies to join to their channels.												
8	PGO	<ul style="list-style-type: none">- Purchasing Group Organizations- A group of companies is participating in a group such that a purchasing office proposes a request and other suppliers can inspect it and make an offer for the request. As an output they could have a deal.												
9	RFQ	<ul style="list-style-type: none">- Request for Quotation- RFQ is the basic component in our marketplace and the request owner would decide to publish it on a PGO- Contains a list of items that a company is needed to buy them- RFQ details are flexible.												
10	RFQ-template	<ul style="list-style-type: none">- Since different fields of industry uses different specifications for ordering a request, the platform provides a flexible entity for an RFQ (Customizable request).- This flexible RFQ could be imported directly from RFQ-template collection and modified by user.												
11	Open-offering	<ul style="list-style-type: none">- It is a PGO's state- When a new RFQ registered and other suppliers can still apply for it.												
12	Close-offering	<ul style="list-style-type: none">- It is a PGO's state- When a purchaser confirms an offer and a deal does start. Then it is impossible to register a new offer.												
13	Dealing	<ul style="list-style-type: none">- It is a PGO's state- The PGO members start to make a deal.												
14	Terminated	<ul style="list-style-type: none">- It is a PGO's state- When a deal is finished.												
15	Rating	<ul style="list-style-type: none">- The PGO members can vote each other and inspect their rates before confirming their offers.-<table><tr><td>Votes</td><td>Rates</td></tr><tr><td>Very bad</td><td>-2</td></tr><tr><td>Unsatisfied</td><td>-1</td></tr><tr><td>OK</td><td>0</td></tr><tr><td>Good</td><td>+1</td></tr><tr><td>Excellent</td><td>+2</td></tr></table>	Votes	Rates	Very bad	-2	Unsatisfied	-1	OK	0	Good	+1	Excellent	+2
Votes	Rates													
Very bad	-2													
Unsatisfied	-1													
OK	0													
Good	+1													
Excellent	+2													
16	My PGOs	<ul style="list-style-type: none">- List of PGOs are created by purchaser account of company.												
17	My Offers	<ul style="list-style-type: none">- List of Offers are done by sales account of company.												
18	Subscription plan	<ul style="list-style-type: none">- A company is registered in the platform by a subscription plan that specifies its payment method.- Each company has a list of payments and bills.												
19	Admin User	<ul style="list-style-type: none">- IC-administrator who can do any thing in the platform												

2. Overall Description

2.1 Product Perspective

As a general view of the IC.V.2.0. platform provides a web-based service for companies from various sectors of industry to have an environment to satisfy their demands. The platform architecture is shown in figure 2:

- **Information microsystem:** *in this microservice all the collections or tables are located in databases as a set of raw data. These data would be used by different modules in microservices. As you can see in the figure2, there are two correlated entities; categorizations and companies. The categorization entity includes the various sections of industry that are going to have a channel in the platform. Another one is company's profile information² that keeps all the relevant information of a company. The important point in this segment is providing a clear definition of every entity, entity relationships and its features, because the second layer should be deployed in the future. This layer is a machine learning and data mining module and it does able to extract knowledge from lower data layer and generates an augmented value for our platform.*
- **Marketplace micro-system:** *Since the marketplace is the major service of platform to generate revenue for IC then it should be developed before than all the other services. The structure of the marketplace microservice is based on the active channels. These channels are defined by an administrator or by an exclusive user. As the number of channels growth continuously along a period, it would make a problem in terms of user experiments. Then a solution is to add a couple of features to channel object then filtering base on them in the frontend layer. The most important concept in the channels is PGO because a company has a request and ready to pay for it, also this company expected to receive many offers for its request. These objectives would be achievable in a PGO where a company can announce its request with details then waits to receive an offer from a supplier. After a new arrival offer, the negotiation between the purchaser and different suppliers is ready to begin. As a consequence, the best-fit offer would lead to a deal.*
- **Search engine Microservice:** *this microservice is an augmented valued that is introduced by IC. It is divided into two modules; one is industrial search and another one is industrial map. The industrial search provides a search engine that does search through the industrial websites³. The magazine of search engine would be collected from crawling the websites and storing them in the data layer (Elasticsearch). Another module in this microservice is industrial map that provides a service for user to find a company on the map with all of its plants and offices. The geo-position information could be collected from industrial profile or by web crawler.*

These microservices communicate with each other through the REST-API.

The marketplace is an interactive ecosystem for all members to communicate with each other with the aim of finding the best-fit suppliers for their request. To make the market more user-friendly, it is divided into a set of channels such that each channel is equal to a sector of the industry. Also, these channels are searchable for a user by different filter items.

² The company information includes company profile with details, catalogs of the company and its requests. Also, regarding to search engine microsystem, the web crawling data would be store here inside information microsystem.

³ A set of websites are belonging to the industrial companies from different parts of world. This list already provided for around 20.000 websites.

When a purchaser user of a company does select a channel, there are many options like observing the list of PGOs are published in channel and another option is, purchaser can create a request by publishing a new PGO.

Besides, the sales account of company can inspect a PGO then may try to make an offer for it. By doing this, the dialog between the sales office and purchaser got started. At the end, the purchaser can confirm an offer for dealing after that a deal between them would be established.

The above scenario is a general view of marketplace that platform wants to propose.

2.2 Product use cases

This section referring to the main uses of platform based on its actors. The actors of the platform are categorized into two kinds of account Administrator and Company. The administrator has access to all functionalities of the platforms and is a powerful user. While Company user decomposed into purchasing and sales accounts. Purchasing office of a company uses the platform like a buyer of items and Sales office is like a salesperson.

In the following diagrams will see the usage of the platform by different actors: the first one shows the relation between the company and administrator, their main activities are related to company registration and profile completion. The next two plots show the relationships between a purchasing office a company and a sales office from another company. The focus of these plots is on the activities of a group.

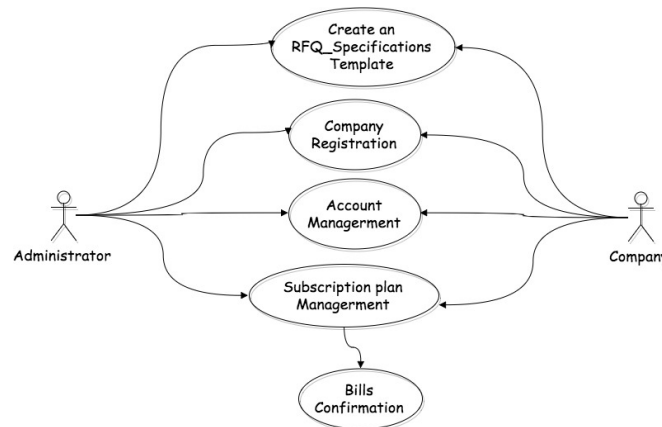


Figure 3: Administrator and Company related activities

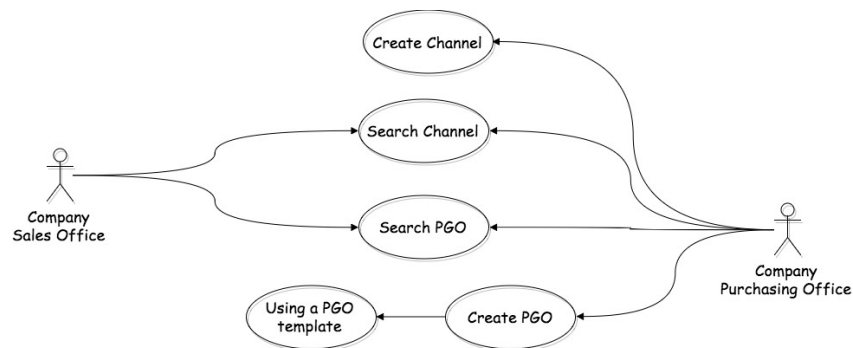


Figure 4: the related activities of Purchaser and supplier from two Companies

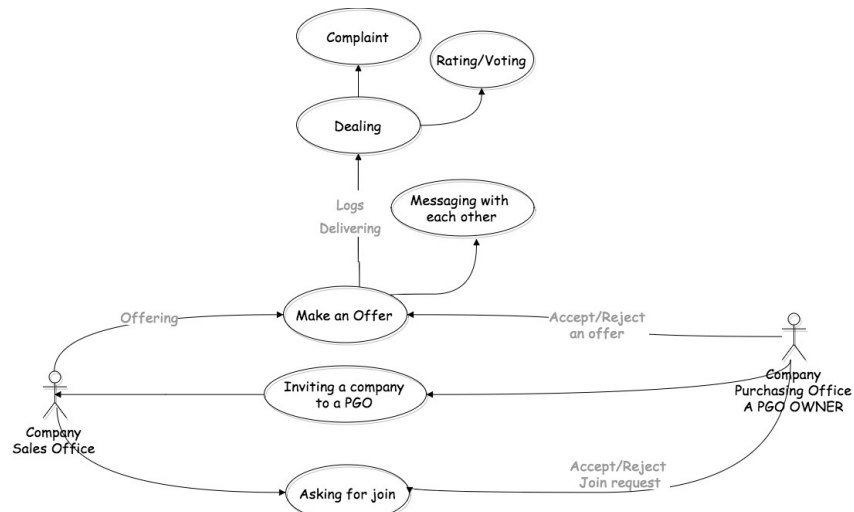


Figure 5: the related activities of Purchaser and supplier from two Companies

The search engine processes are as follows:

Searching through Search Engine

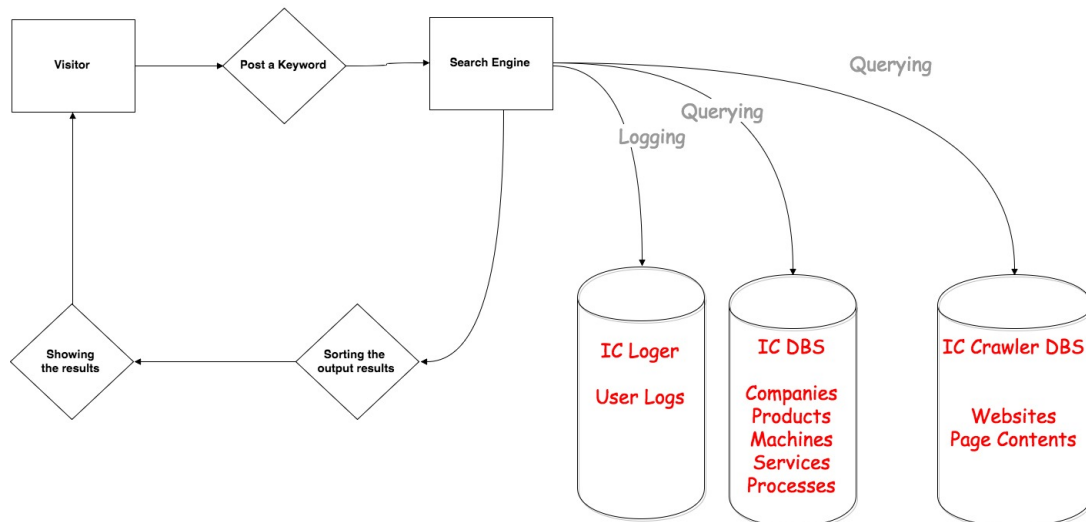


Figure 6: search engine functionalities

As you can see above, there are two main sources of search one is website crawling documents and other one is the company profiles. Adjust to them is a logger that collects the activity log of users, these would be useful in ML layer development and marketing.

2.3 Product Functions

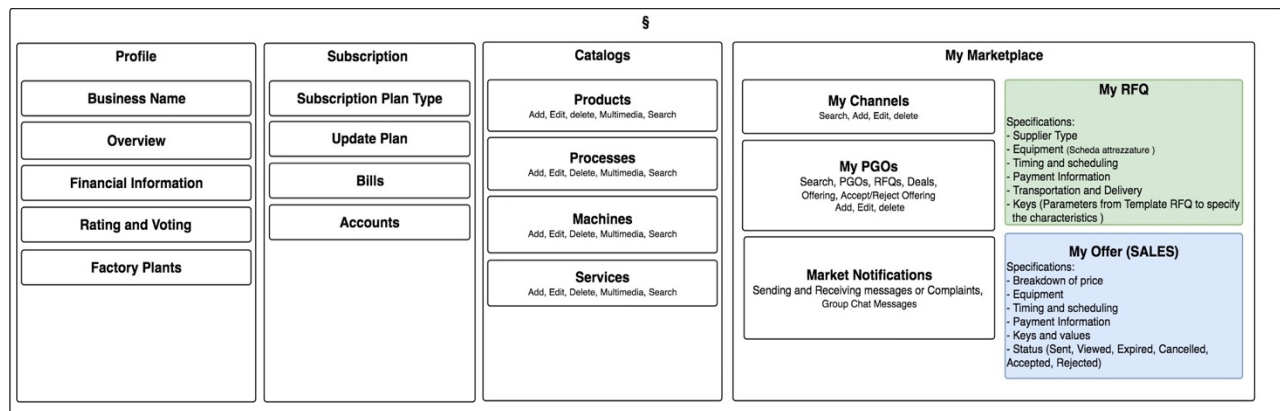
The functionalities of product are scattered among different microservices and they would communicate with API endpoints, at below you can see the most important ones and they should develop first.

- Company registration routine

- Profile completion
- Catalog registration
- Account management
- Subscription plan
- Bills and payment
- Channel creation by administrator or Exclusive users.
- Searching for a channel and filtering by categorizations
- Managing the RFQ-template by admin user.
- PGO creation by purchasing account of a company
 - Define the specifications of RFQ as pairs of (key, value) or can be imported from an RFQ-template collection.
 - Assigning values for RFQ keys by purchaser
 - Inviting suppliers to make an offer
 - Messaging with sales office
 - Accepting/Rejecting an offer
 - Dealing and goods delivery
 - Voting and rating
- Searching for a PGO by Sales office of a company and selecting one of them.
 - Making an offer for PGO
 - Messaging with purchaser office
 - Voting and rating
- Crawling the websites and store them in Elasticsearch
 - Search in the Elasticsearch database
 - Applying the marketing policies for search engine
- Search through the stored geo-location data of companies

The blow figure shows a general view of all functionalities of a company that are going to be developed in its dashboard.

Figure 7: The Functionalities of a Company in its Dashboard



The industrial profile is a profile for a company to introduce itself and its business that what is doing. To achieve this aims there is a set of sections for the company to publish all of its products, services and processes and etc. are called catalog.

At the below figures you will see the functional view of a purchaser user (green color) and a salesperson user (blue color). These plots contain the most of functionalities that a company can do and proposed each as a single box, but the objective is not to put some constraints on the user interface design to be same as the following diagrams.

The Channel environment in **Purchasing office** pint of view



The **PGO** environment in **Purchasing office** pint of view.

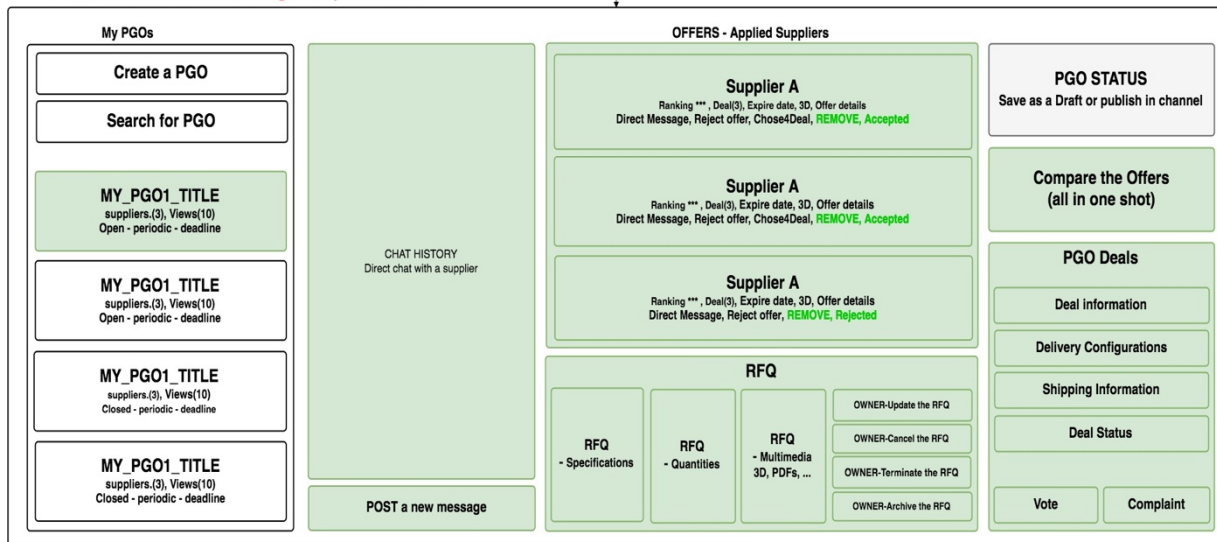


Figure 8: the Purchaser user dashboard in a channel and a PGO

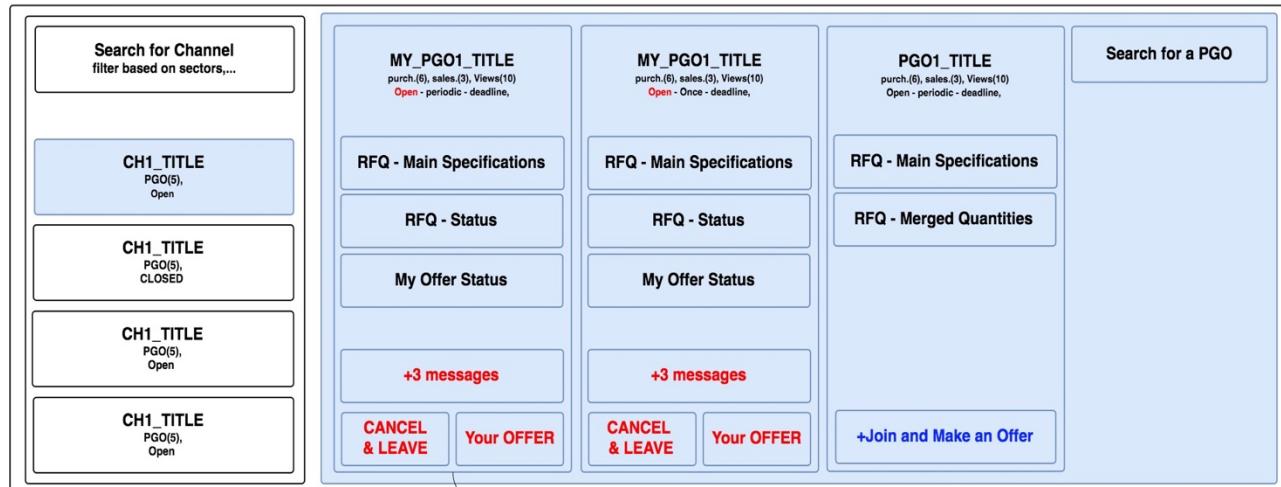
Since the platform is going to meet the **SIGIT**⁴ needs first and the strongest point for them is, they have long-term contracts for each RFQ and doesn't like to share their RFQs with others. This means that they have PGOs with only one purchaser and a group of suppliers. While there is another teamwork model for PGO with a number of purchasers who are going to merge their RFQs with together and make a bigger RFQ to reduce costs and find a better offer.

Regarding to merge RFQs, in the next development, it is necessary to define a concept like public or private PGO, such that if a PGO is public means its owner wants to merge with all the others automatically else if a PGO is private, means the merge request should be confirmed by the RFQ owner.

The next figures show the functionalities in the sales office dashboard.

⁴ The SIGIT is a potential customer for this platform to use it.

The Channel environment in **Sales office** pint of view



The PGO environment in **Sales office** pint of view

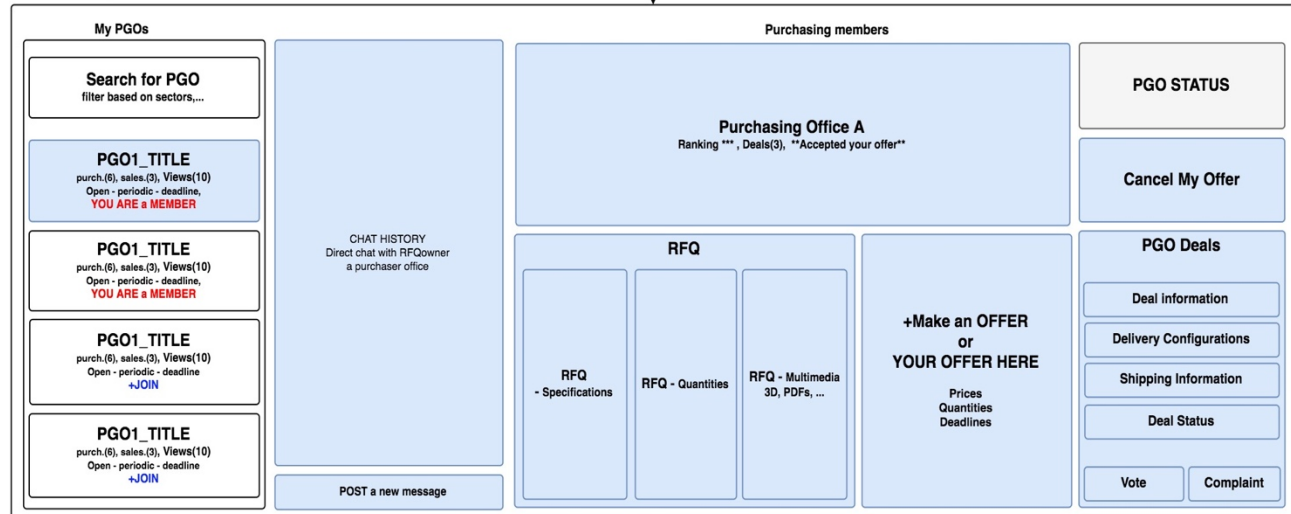


Figure 9: the Salesperson user dashboard in a channel and a PGO

It is possible for a supplier to make an offer for a request or cancel it. Offer canceling means supplier wants to quit PGO and not participating more. Also joining to a PGO and offering would be at the same time in one operation to have a fast and easy process.

The states of a **PGO** are as follow:

- Drafted: the RFQ is defined by purchaser but not published.
- Expired: the deadline passed and not valid anymore.
- Canceled: the request totally canceled by its owner.
- Open Offering: the RFQ is published in a channel and available for suppliers.
- Closed Offering: One supplier is selected to make a deal and no new supplier can apply.
- Terminated: The contract has been finished.

The states of an **offer** are as follow:

- Accepted: beginning to chat and negotiate

- *Rejected: No need to negotiate.*
- *Expired: the deadline passed.*
- *Canceled: the request is totally canceled.*
- *Dealing: Select the supplier and Starting to make a deal.*

2.4 User Classes and Characteristics

List of classes of platform are as follows:

- *Administrator class*
- *Categorization class*
- *Catalog class*
- *Subscription class*
- *Bill class*
- *RFQ-template class*
- *Company Class*
- *Company account class*
- *Channel class*
- *PGO class*
- *PGO_RFQ class*
- *PGO_RFQ Multimedia class*
- *PGO_RFQ Offer class*
- *PGO Message class*
- *PGO Deal class*

The class diagram shown in the following figure. Each class is divided into two sections, one for properties and another for methods of class. There are three same methods in all classes for creation, update and removing an object sample of class. At the same time the class may have more functionalities to implement, this depends on the developing time issues and its developer should detect them. Also, if a method hires an object from other class then there is a link between them and may need a foreign key in data layer.

For developers, remind that all of these classes are our data models then they should be defined more or less the same and may require to shift some of these methods to the resolvers of model, as much as possible.

As you can see, the starting class is the administrator class and other classes branch out from that. These are the root classes and colored with a yellow header. There are other root classes like company class.

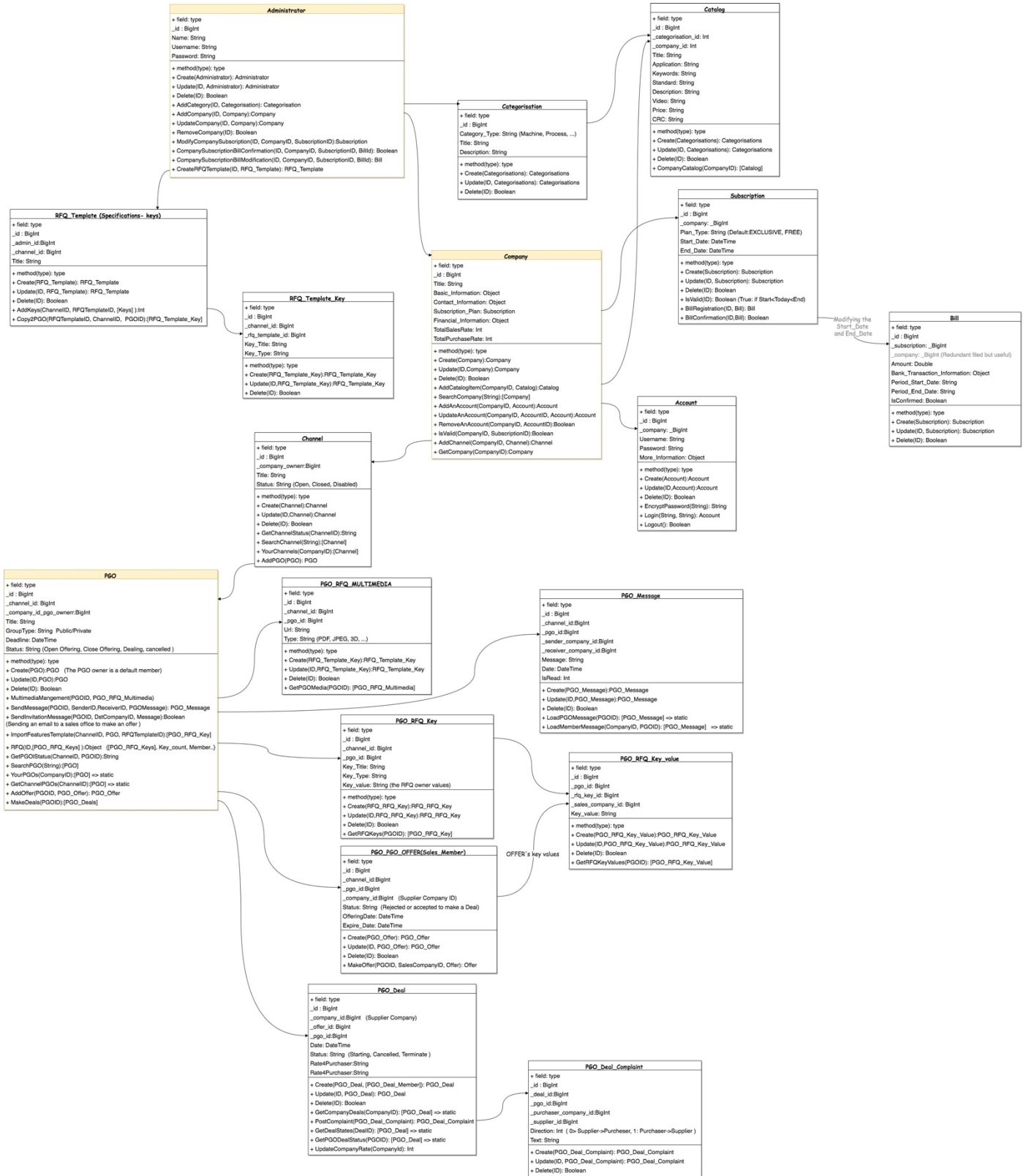


Figure 10: the proposed class diagram

2.5 Process Sequences

In this section a couple of important sequences will be drawn in a set of sequence diagrams. These plots will show that how would be the order of tasks in a process? Who could start a process? And etc. these sequences would apply in final implementation of platform.

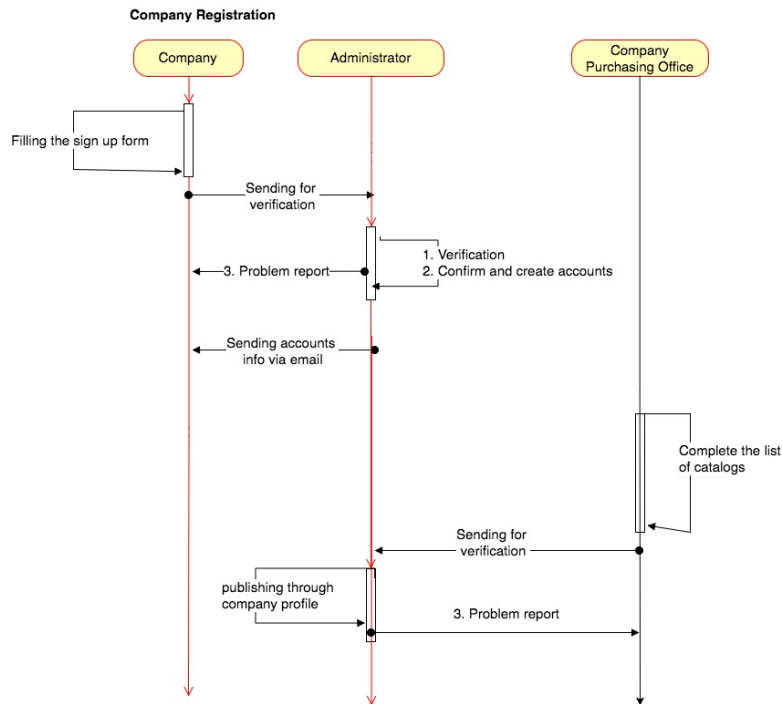


Figure 11: the company registration sequence diagram

During the company registration the IC administrator will check the accuracy of entered information and care about fake accounts. In this section company should complete three segments are profile, subscription and catalog.

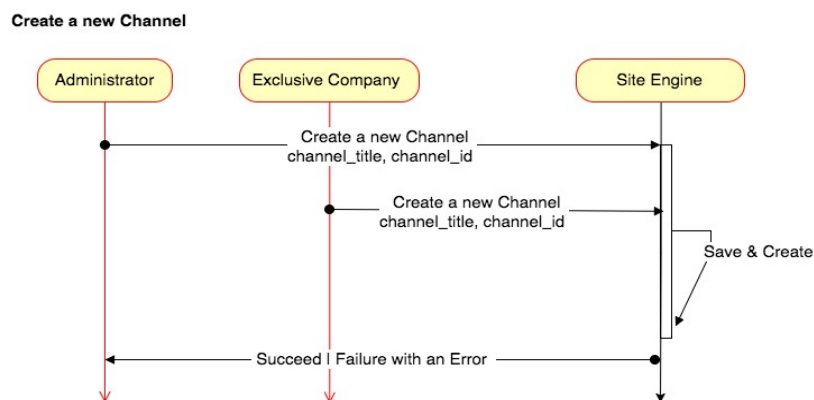


Figure 12: the channel creation sequence diagram

Both of privileged users, Administrator and Exclusive users, can create a channel and share it with other companies to participate there. Each new channel is equivalent to a new category of the industry, so by creating a new channel then a new category should be added to the list of categories. Since the number of

channels will increase continuously then in the close future there would be a large number of channels that seems to make problems in UX, it wrong because each channel has many properties are providing a filtering and easy access to desired channels. Otherwise platform has to use a solution like a hierarchically model for channels that would be based on the sectors and subsectors with higher complexity.

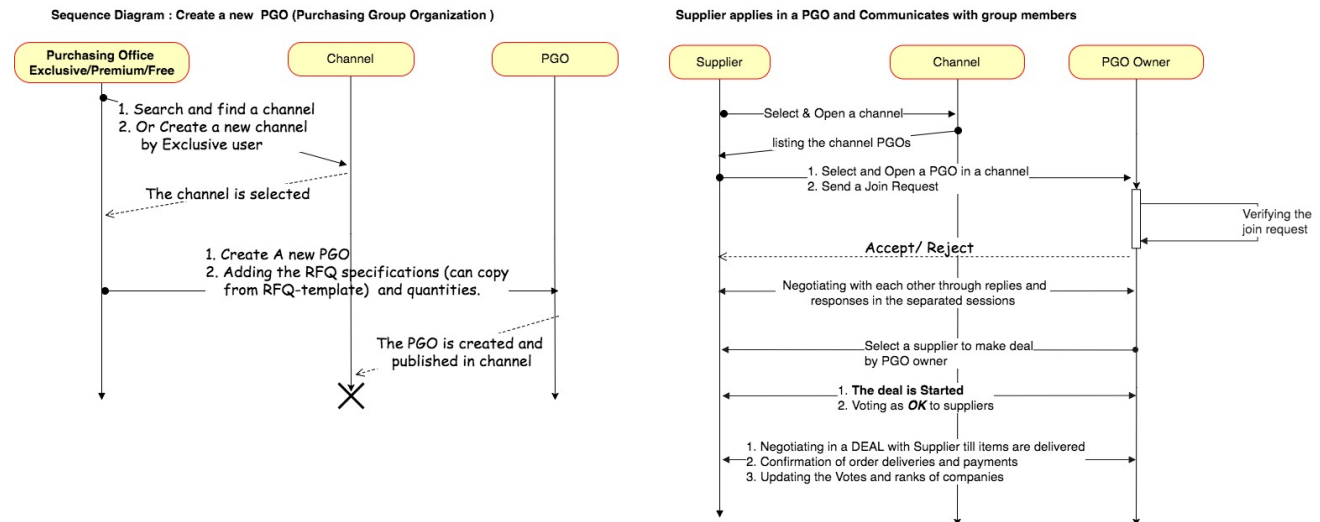


Figure 13: the channel creation sequence diagram

As it shown in the above figure, there are two sequences one for the creation of a new PGO and another for offering. When a supplier applied, the PGO owner would receive a notification then could accept or reject the offer; if the offer is accepted they can start to negotiation till reaching an agreement. at the next step, they could make a deal. The PGO exists in the channel till the deal is valid or terminate then it can be archived.

How to protect the PGOs from fake supplier/buyers?

- Since a company has two different accounts, purchaser and Sales, and inside every PGO a company can just play a role as a Supplier or a Buyer.
- A company might be a big holding with a group of sub-companies then during Registration of it, the Administrator should take into account to count all of them in one account. This policy would help to protect the PGOs from fake supplier/buyers.

2.6 Operating Environments and Technologies

The production environment in which the software will operate is with following specifications:

- Developing and Programming language
 - Backend
 - API, MongoDB, Grapqhl Server, and Express
 - Elasticsearch, Logstash, and Kibana, and Nutch Crawler
 - Frontend
 - Administrator dashboard (React, Redux, Apollo)
 - Company dashboard (React, Redux, Apollo)
 - Search engine (React, Redux, Apollo)
 - Frameworks
 - Mocha and Chai⁵ as a testing tool

⁵ <http://chaijs.com/plugins/chai-http>

- MINIO S3⁶ as Object Storage Server
- Host Server: Linux Server
- Git: GitLab on an internal server

2.7 Design and Implementation Constraints

There are various constraints during development process and should deal with many intuitively ideas. For example:

- The platform should be developed as multi-language platform at least in Italian, English and Russian language.
- regarding to search engine it would be necessary to customize the NUTCH crawler to provide a custom index writer in the elastic search magazine.
- SIGIT is already a potential customer and platform should stratify them by considering all their requirements.
- The number of developers might be limited to only two persons.
- The interface design should be simple and understandable for industry people.

2.8 User Documentation

There are many documentary files and directories at root directory of each microservice to help developers in the future. Which are:

- **“ReadMe.md”**: is a text file that contains some useful information about the maintenance of a microservice such as project structure, files and directories, installation and setup, correlations between services, libraries and frameworks and all the other related information.
- **“ToDo.md”**: shows a list of tasks should be done in the future to cover an issue. In the whole project may somewhere is needed to mention if there is a necessary issue.
- **“ChangeLog.md”**: A change-log is a file which contains a chronologically ordered list of notable changes for each version of a project.
- **“documents”**: Contains list of all manual and document files.

Other relevant documents of project are mainly used by users and marketing department:

- **Administrator Manual**: a manual that should be written as a guideline for an administrator user.
- **Company Manual**: a manual that should be written as a guideline for a company user.
- **Marketing Documents**: a guideline for marketing department how to introduce the platform, communicate with customers.
- **Advertisement Documents**: Brochures, posters, logos and banners
- **Development Control Processes**: A Gantt chart that contains a list of processes of development. It will be used by administrator.

⁶ <https://github.com/minio/minio>

3. External Interface Requirements

3.1 User Interfaces

.....

3.2 Software Interfaces

Since in this project a group of partner frameworks collaborate together, intra-framework communications are big issues in development. The good news is that all of them have a Github repository with a manual and easy to use. Then all the software interfaces will be available.

3.3 Communications Interfaces

*In IC-V.2.0 all interactions between client and servers are **One to One Request/response**⁷ communication through a REST mechanism. There is a level-2 API uses HTTP verbs to perform actions:*

- *a **GET** request returns the representation of a resource, which is in the form of a JSON object.*
- *a **POST** request creates a new resource*
- *a **PUT** request updates a resource.*
- *The request **query parameters** and **body**, if any, specify the action's parameters.*

Inside the REACT project there is a request router that manages all the incoming requests in both backend and frontend parts.

The following table shows the list of URLs⁸ and endpoints for services:

<i>Permission</i>	<i>Operation</i>	<i>URL</i>	<i>Request-Body</i>	<i>Response- JSON</i>
<i>Admin</i>	<i>Creating a Company</i>	<i>/regcompany</i>	<i>Token-header-attributes</i>	<i>{ result: { keys: values }, message: 200 }</i>
			..	
<i>Purchaser</i>	<i>Creating a channel</i>	<i>/regchannel</i>
<i>Salesperson</i>	<i>Making an offer</i>	<i>/makeoffer</i>
		

Whenever a developer extends a new feature should add it here.

⁷ A client makes a request to a specific service and waits for a response. The client expects the response to arrive in a timely fashion

⁸ the list of URLs is more precise during the development process.

4. Other Nonfunctional Requirements

4.1 Performance Requirements

In order to assess the performance of a system the following tables must be clearly specified by QOS team:

- **Platform(Hardware):** this item isn't a big issue for us because there is no hardware. Maybe in the far future when the company is going to allocate a physical place for the server then it could be an issue, but not for now.
- **Response Time:** this item, also, is not a big issue for our platform because there is no penalty in the case of failure request (just for begging!).
- **Workload:** This item could be a big issue for our platform because the loading of pages would increase dramatically in the future by growing the number of companies. Then the following table has to be completed during the development and keep track along the running time.

Permission	Operation	Pages Visiting pages to do the operation	Expected Daily Total
Admin	Portfolio edit	Login, Portal, Profile View, Portal Pref, Catalog View, Add Service, View service, 50% Exit	10
			..
			...
		TOTAL	2000

- **Scalability:** the list of expectations should be provided to satisfy the system growth issues. The following table should be completed by business department based on their predictions and expectations.

Objective Predict a value for critical issues	Expected Value
Total Number of companies	20000
Total Number of Catalogs (Machine, Services, Products, ..)	...
Total amount of required storage
Total Number of channels / categories
Total Number of PGOs
Max number of offer for an RFQ or in general

Total number of documents in Industrial search engine
The number of HTTP request from different interfaces (Mobile, web)

The list of above objectives rooted from the point of view of different users, and should be satisfied in a suitable production environment.

- **Data validation:** since the collected data in platform are provided by companies then there could be a doubt about the accuracy of them. The following solutions are proposed for to do filtration:
 - **Checking the input data validation by engine as much as possible**
 Defining the enumerated data types for a set of variables with a specific domain range for their values.
 Assign a set of limitations by the user interface and database operations to avoid receiving/storing the invalid data.
 - **Checking the input data validation by Administration as a short-term solution**
 This check could be done once for company registration. Honestly, this issue means that our UI/UX design has some problems such that a user can't complete by itself.
 Besides, the platform should provide an analytic dashboard with some meaningful diagrams and statistical reports for administrator user to monitor various sections of the platform.
 - **Inspecting for inconsistencies in data by a bot in the platform as a long-term solution**
 For the next revision of the platform when the machine learning layer is going to be implemented, this bot should be deployed next to other modules to extract the correlations and the accuracy of data automatically.

4.2 Safety Requirements

The safety issues in our platform categorized in the following group:

- **Data corruption:** this isn't a remarkable issue for our platform, unless the company closed or changed its productions and so on. These are the rare cases, therefore, ignore them.
- **Data expiration:** this could be an issue specially in PGOs and channels. Consider the following scenarios; first, when a channel is created but for long time no PGO published in it then this channel is totally useless. Second, a PGO published in a channel but for long time no offer for it. Another examples are possible; all are the expired data then a solution is to use the garbage collection technique.
- **Server crash:** this issue is solvable by providing a reliable server and a regular backup solution.
- **ML feature scaling:** regarding to the development of machine learning layer; a suggestion is to use a separate data structure for ML-Model, adjust to platform's database, because the feature scaling process may change the stored data in DBS then the data safety will be a big issue.
- **Data migration:** the technical administrator should monitor the migration of platform and its data.
- **Code protection:** managing the versioning of software during development to protect it from missing or unwanted merges and so on.

4.3 Security Requirements

The security requirements are provided by:

- **Authentication layer:** a layer via a proxy gateway in front of backend layer that checks all the incoming requests.
- **Data Privacy:** since a partner like SIGIT is going to give us many official documents, it is necessary for development team to be bind.
- **Server virus defender**
- **SSL service for server**

4.4 Software Quality Attributes

The QOS policy based on a cycle;

- developing test-functions by every developer during the implementation.
- functional tests by tester.
- Performance tests by tester
- UI/UX tests by tester.
- Feedback to developers
- Documentary tests
- Maintenance tests

The following table should be completed by tester when performs each kind of test.

Tester	Subject <i>Function, URL,</i>	Date	Test status	Problem	solution
<i>Mr.Javad</i>	<i>Company registration</i>	<i>12.01.2017</i>	<i>OK</i>	-	-

4.5 Business Rules

....

Appendix A: Gantt Chart

All the required processes are drawn in a chart to how are their sequences and timing along the development process of IC-V.2.0.

Appendix B: Analysis Models

All the analysis diagrams are shown in set of RUP diagrams.