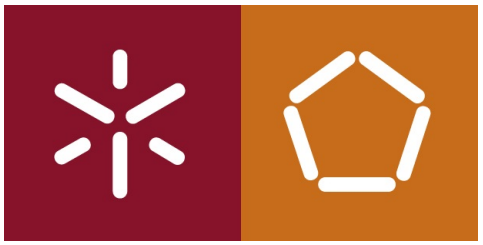

University of Minho

Software Engineering

Requirements Engineering

Sniffler - European Open Banking for Credit Risk Analysis

Authors:	**Student Number:**
Daniel Maia	A77531
Diana Costa	A78985
Marco Silva	A79607
Tiago Alves	A78218
Vitor Peixoto	A79175

7th January 2019

Contents

1 The Purpose of the Product (Volere - 1)

1.1 The user problem/background to the project effort

Imagine how awesome would it be to have someone, or something, managing our bank accounts and expenses. Even more, suppose that all money crisis and bankruptcies could be avoided if we just let Open Banking, machine learning algorithms and companies/individuals work together. The EU and many banks are pushing this development with the new Payments Service Directive 2 (PSD2), which has come into force on January 13th of year 2018. Banks face the need to adapt to these changes which open many technical challenges, but also many strategic opportunities, such as collaborating with fintech providers, for the future.

This new directive will open the opportunity for third-parties to consume data through API of bank accounts, with the proper authorisation of the holders. And here is where the magic happens: by providing access to bank movements, it is possible to "calculate" the credit risk of potential clients and thus create better offers.

That's where the "European Open Banking (PSD2 directive) for credit risk analysis" project comes in, suggested by the Scytale company, and made for the course of Requirements Engineering. Scytale already has a Marketplace project - *Capitalise* -, which consists of a digital commercial credit broker for online platforms. They have made it easier to obtain funders (funding search) and receive funding through matching and profiling. Scytale wants our team to make a system which performs cash flow risk analysis of a client, and suggests the use of Capitalise to obtain funders. In the future, the main goal is also to analyse credit risks at *Capitalise* level. Another view would be to, as suggested previously, gather machine learning data with the team's system to predict monthly spending and better analyse cash flow risk.

1.2 Goals of the project

With this project, the team aims to provide to all of the system clients' a service where they can have an overview of all of their bank accounts in one location. In addition, with the information of each client, the system should throw alerts related to regular expenses and, in general, help clients manage their money. As such, the team intends, with this project, to study the viability of using this information to predict credit risk of potential clients, and thus to create a better offer.

For this, the group will develop an application capable of consuming data through banks' API and extract useful information. Ideally, with the power of machine learning algorithms, possible risk situations can be avoided. Furthermore, all this information can also be used to evaluate credit risk situations for new clients and thus create a better offer.

2 The Client, The Customer and Other Stakeholders (Volere - 2)

2.1 The Client

The client of this project is Scytale. A web development and design company which, among many other projects, developed *Capitalise*, a software product which provides an easier way for individuals and companies to obtain funds. This project is being developed as another piece to integrate the *Capitalise* puzzle.

2.2 The Customer

Given the purpose of this project, we can clearly define the customers for this product. This software's main target is not only the business industry, but also individuals, who need a personal assistant to control and alert to the risk of cash flow in their business.

Making an Importance vs. Influence Matrix helps to map out stakeholders and their relationship to the issue of the project. It helps by generating insights on the importance and influence of each stakeholder.

2.3 Other Stakeholders

In a project of this dimension, the amount of people involved and influenced by this product is relatively sizeable.

After a meticulous analysis of the project context, a list of the primary stakeholders was concluded:

- European Union
- Financial and bank institutions
- Payment institutions (Paypal, Visa, MasterCard, etc)
- Investors
- Governments
- Lawyers and analysts of legislation
- Judicial institutions / Taxing entities (Regulators)
- Accountants / Economists
- Software developer (APIs and apps, mainly)
- Tech "giants" and their respective online payment services (Google, Facebook, Apple, Alibaba, etc).

To better understand all the stakeholders' roles in this project, an importance/influence matrix is crucial to map out the relationships between the stakeholders and the issue of the project. With this information, it becomes possible to develop a specific approach and strategy for the identified stakeholders.

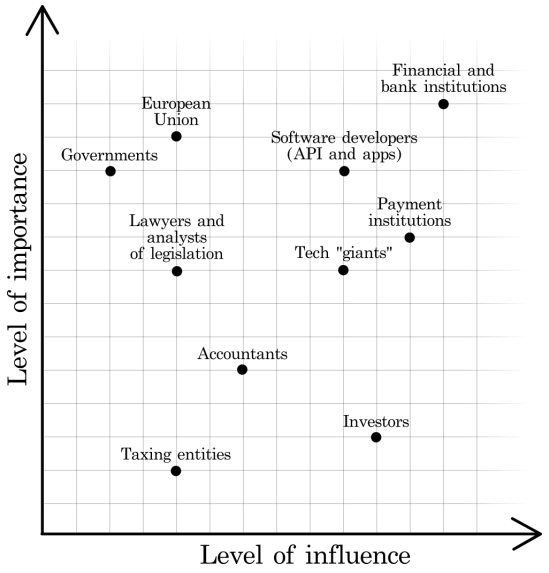

Figure 1: Stakeholders distribution according to a importance/influence matrix.

- **European Union and Governments:** the directives implemented by the European Union and national governments (such as the UK, to cite the leader in Open Banking) take an important role in the development of Open Banking related software, such as this project. Without those directives, the whole system wouldn't even be able to lift off.
- **Financial, bank and other payment institutions:** Open Banking stands on mutual sharing of information on bank accounts. If those institutions don't create standard API's to do so, Open Banking won't work as a global network and our project won't be able to support all bank accounts that the clients may have.
- **Investors:** Investors are responsible for making companies work. So, as this system warns and gives advice about possible cash flow problems, the utilisation of *Capitalise* is very important and the investors make a major role of the investment needed to keep the business running.
- **Lawyers and analysts of legislation:** Open Banking is a very important topic as it deals with large amounts of money and client data. This requires the study of legislation on data protection. The lack of legislation analysis can lead to lawsuits and the fall down of the entire project.
- **Taxing entities (Regulators):** Taxes are an important issue to be analysed as part of the legislation topic since some taxes could be applied to the project. Although, the involvement of taxing entities in the project as a whole is not very significant.
- **Accountants / Economists:** The whole objective of the project is to help companies and individuals manage their bank accounts and warn them about possible cash flow problems. This is part of the job of an accountant, so this software might help a lot of accountants in their client management.
- **Software developer (APIs and apps, mainly)** This is a obvious stakeholder and a very important one. Software developers are necessary to develop the bank APIs that will integrate

the Open Banking network, to develop the software that actually uses those APIs and make them useful to the world, by developing software that helps people manage their daily life. In this case, to develop a system to help companies and individuals manage their bank accounts.

- **Tech 'giants' and their online payment services:** Tech 'giants' are important in any software related problem nowadays. Their influence is so big that it is obligatory to add them as a stakeholder here, specially if they are involved in the economical and financial business.

3 Users of the Product (Volere - 3)

3.1 The Hands-on Users of the Product

- Client

  This project is mostly directed at the client, so this is what makes him the main user of the product. It is a product which requires a very high degree of attention by the client, here he will manage his risks and finances. Because of this, it's assumed that the person who'll use the product has some kind of technological experience. The client must be like a journeyman, at least, when it takes to the knowledge of the business, like credit risks, finances, banks, all about his economical life. He must know what he's doing and what is happening when using the product, otherwise it can bring really bad consequences not only to him, but also to others around him.
  Given the facts above, and as it's a global product, it will be presented in English language, so the client will need to understand this idiom.
  Once again, this product will provide to the client the power to manage better his finances. So, It's a bit logical that this application is for users that are responsibilities and capable of leading their financial assets.

- Administrator

  An administrator is someone who has got the needed rights to manage something. In this case, he or she is the responsible of manage the application. Just like a client, the admin must know the entire business, and be aware of all the happenings. If a certain company wants to enter an be a part of the product, he must know how to get it into the business. If he needs to remove some client from the system, he must be able to do it, understand why and what went wrong. It's an admin, so he'll want to manage the best as he can the product.

3.2 The Priorities Assigned to Users

- **Client** - Key user. The client is the user that can take the product to the success. As has been said above, this product might change the life of a person, in the way that she can manage better her economical life. So, having the usage from the client and we providing the necessary design, tools, and most important, the necessary features for him to manage as he wants his account, makes the possibility of reaching the success even higher. It's our goal to get the satisfaction and the feeling of safe from the client using this product.
- **Administrator** - The manager user. The admin must have the chance of seeing how the product goes. He also finishes as a key user, because with no admin, there's no management of the product, there aren't new entities, new companies, and the product wouldn't give the client the chance of manage his accounts better, which can't happen.

3.3 User Participation

The user participation is a crucial part for the product to be well developed and to have a proper functioning. There are projects that without the required user participation it lacks. So, making this product user friendly and to gain the attention from the user when he wants to manage better his credits, his finances, this is part of the goal. If the client uses this application every time he needs to check his credits, if it's really secure to him, it can be said that the product is reaching the success.

3.4 Maintenance Users and Service Technicians

The maintenance users are special users who have access, rights and requirements for maintaining the system functional. These users will be the people who are developing this product. It cannot happen, but if something in the product goes wrong, these users must fix it.
In general, these users have got access to everything related to the product: source code, database, all of it. The maintaining users are responsible for the system's stability.

4 Mandated Constraints (Volere - 4)

4.1 Solution Constraints

In this section, all the constraints on the eventual design of the product are specified. Therefore, for each constraint, the respective description, rationale and fit criterion are presented.

- Solution Constraint nº1
  - **Description**: In case of forecasting a situation of cash flow failure, the system should suggest the use of the Capitalise platform.
  - **Rationale**: Through this suggestion, the customer will be presented with possible solutions to his problem.
  - **Fit criterion**: If a possible cash flow failure situation is detected, the system should provide a link to the Capitalise platform.
- Solution Constraint nº2
  - **Description**: The cash flow failure prediction service should work only with internet connection.
  - **Rationale**: For security reasons, all data to be analyzed must be obtained from the APIs at the time of analysis and never stored in any device.
  - **Fit criterion**: It should only be possible to initiate any action on the platform if there is an internet connection.
- Solution Constraint nº3
  - **Description**: The product must be available for download on any Android or iOS device.
  - **Rationale**: There should be no technological limitations on the use of the product.
  - **Fit criterion**: The product must be available for download on Google Play (Android) and App Store (iOS).

4.2 Implementation Environment

Containers are the solution to having reliable and easy-to-maintain software. The use of this technology allows the system to be highly portable, since its execution is independent of the operating system used by the host machine. Moreover, the use of containers allow applications to be more rapidly deployed, patched, or scaled.

Finally, in a system implemented on containers, it is enough to apply Agile and DevOps processes, thus accelerating the testing and development processes, as well as facilitates the implementation of production cycles.

4.3 Partner Applications

Since the software product through the banking data of each of your users makes a prediction of possible cash flow failure, it makes sense at this time to offer a solution to his problem. In this way, the integration of this product with a financial credit counseling platform allows the immediate presentation of solutions to client's problem. In addition, the credit products offered should fit the client's needs to the maximum in order to facilitate the customer's decision. In this way, the already existing platform in the market, Capitalise, in addition to contacting a large number of investors from different business areas, also integrates the construction of a customer profile in order to present the proposals that best fit each problem.

4.4 Off-the-shelf Software

Since the application area of the software product is in a fairly permissive and expansion phase, there are still no large-scale solutions applied in this area. Being this a pioneering software, all the components of the product will be developed integrally by the development team. This development option will be taken with the primary goal of eliminating vulnerabilities. Moreover, since a large number of users of the product are anticipated, the lack of applications of these large-scale external components introduces an unwanted uncertainty factor by the development team.

4.5 Schedule Constraints

Again, the fact that the development of this software product is done in parallel with the expansion of this new market area will be decisive for establishing the objectives to be fulfilled with regard to the project completion date. In PSD2, it is expected that by September 2019 all functional and security requirements will be implemented, including the payment service. Consequently, the conclusion and market launch of this product should be made by September 2019, thus taking advantage of this window of opportunity.

4.6 Budget Constraints

Since this is an academic project, there are no monetary limits properly since all the considerations made may not be the most appropriate for a production environment. In a business constellation, there are product development costs. In this case, as the development will be done in the context of a training course, this can not be considered.

Despite this, an estimate of the cost of developing and hosting servers is then presented. Not being a full-time project, since it will be carried out during the training period of the team, it is estimated that 15 hours a week will be carried out. Taking into account this value, it is estimated the time required for development will be approximately one semester. So an estimate of the final value of the product would be around 10,000 euros.

5 Naming Conventions and Definitions (Volere - 5)

5.1 Definitions of Terms and Acronyms

Given the terms used in this document, here are presented a set of definitions for each of them:

- **AISP**: Account Information Service Provider. A service provider which permits a customer to see all account information from various bank accounts in one place, given consent.
- **API**: Application Programming Interface. A set of tools and building blocks which facilitate program development.
- **Bank**: A financial institution which takes deposits and generates credit.
- **Fintech**: Financial technology which delivers services through technological means.
- **Machine Learning**: The field of artificial intelligence in which programs are given the means to progressively improve performance on a given task by being fed data and resorting to a set of statistical techniques.
- **Open Banking**: An initiative whose goal is to instruct banks to share access to account information and payment mechanisms to third-party providers.
- **PISP**: Payment Initiation Service Provider. A service provider which can initiate a transaction on behalf of a customer, given consent.
- **PSD2**: Payment Services Directive 2. A directive put forth by the European Union with the intent of building a more integrated single market of payment services.
- **TPP**: Third-Party Providers. These are Account Information Service Providers and Payment Initiation Service Providers not directly controlled by the banks or the customers.

5.2 Data Dictionary

Given the terms used in the models displayed in this document, here are presented a set of definitions for those which are not specified within their respective sections:

- **Bank**: A financial institution which takes deposits and generates credit.
- **Bank data**: Banking information regarding a client's activity within given bank.
- **Client**: The primary user of the Sniffler system.
- **Credit Risk Analysis Report**: Document which reviews the client's cash flow across its bank accounts through their respective bank data.
- **Open Banking API**: Application Programming Interface through which Sniffler accesses a client's bank data.
- **Payment Data**: Information regarding a client's payment of goods and/or services through a given bank.
- **Scheduled Debit's Data**: Information regarding a client's scheduled payments and outgoing transfers through a given bank.
- **Sniffler**: A web based cash flow tracking service.

- **Transfer Data**: Information regarding a client's outgoing transfers through a given bank.

6 Relevant Facts and Assumptions (Volere - 6)

There are some logical requirements that the clients must have to use the product, and it's assumed that they own them.

- The software's idiom is English.
- It's assumed that all the users of the product must have a certain knowledge of the English language.
- It's assumed that the product will have access to bank accounts and their data.
- The APIs will be available and won't suffer any changes.
- There will be an interface available to the user.
- It's assumed that the user is responsible, when managing his account.
- There are no automatic operations. The client is responsible for all of them.
- It's assumed that the user agreed with the project terms.
- It's assumed that the user has got a smart devices to be able to interact with the application.
- It's assumed that the smart device's internet connection is on.
- It's assumed that the user is signed up in order to use the product.

7 The Scope of the Work (Volere - 7)

7.1 The Current Situation

This project is inserted in a world where companies and industries (more often small companies) still use accountants or the manager himself to coordinate all the cash flow. This can be a stressing task and prone to mistakes. This software eliminates all the stress and mistakes as it manages all the companies cash flow and gives warnings to future problems.

Looking at the big picture of *cash-flow problem detection* software, there are some apps already available that offer the same tools this project is designed to. This is obviously taken as a risk for the success of this project.

However, the context in which this product is being developed gives it leverage to compete with other products. By integrating the problem detection software with a direct link to *Capitalise*, it gives the companies the option to get easy credit immediately, and this is a new concept that isn't available yet in the market.

7.2 The Context of the Work

In order to be able to build this product, the functioning of the economics in the business area needs to be understood. This implies interaction with the business environment (managers and accountants) to support the development of the project.

Open Banking directives, *Capitalise* developers and some stakeholders, such as legislation analysts, also need to be taken in account as a adjacent system of the project. Their interaction is an important part in the development and functioning of the system.

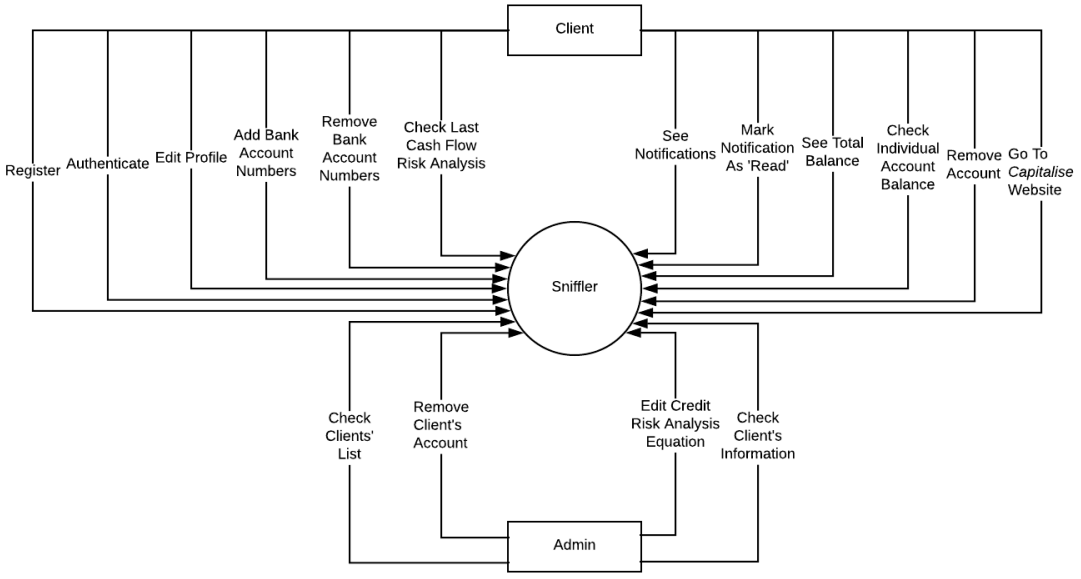

Figure 2: Context Diagram of the system.

7.3 Work Partitioning

The system must be able to respond to a set of multiple basic situations. The ability to respond effectively is crucial to the success of this project, since it is based upon the detection of cash-flow problems.

The response to each possible event must be registered in order to develop a strong Use Case Diagram to contribute to the total functionality of this project.

Event name	**Input**	**Summary**	**Output**
Register	Valid username and password	Generate a new account through which a Client may access the system.	Create a Client User account
Authenticate	Valid username and password	Access the system with a known username and password.	Grant access to the user.
Edit profile	Update profile information.	Verify if new information is valid.	Confirm update.
Add bank account numbers	Insert bank account information.	Check if bank account number is valid and allowed to be analysed by our software.	Add bank account to the client profile.
Remove bank account numbers	Select bank account to remove.	Wipe bank account from the client profile and all of its associated data.	Inform user.

Event name	**Input**	**Summary**	**Output**
Remove account	Select the 'Remove account' option	Remove the user account from the system.	The user's access is revoked and their sensitive information is wiped from the system.
Check last cash-flow analysis	Select analysis to display.	Search for all information and data gathered by that cash-flow analysis.	Display information.
See total balance	Ask for balance during given time period.	Get all bank accounts information.	Display last bank transactions, current situation and balance.
Display notifications		The system generates notifications based on the continuous analysis of the bank transactions of the client.	Display notification.
See notifications	Select Notifications tab	View a chronological set of all system notifications.	Display notifications in detail
Mark notification as 'Read'	Select the 'Mark as Read' option of a notification	Acknowledge to the system that a given notification has been delivered, such that it may cease to draw attention within the system.	Selected notification no longer draws attention to the user.
Go to *Capitalise* website		The system suggests the user to use the *Capitalise* funding provider whenever cash-flow risk is detected.	Display link button to *Capitalise*.
See total balance	Select the 'Total Balance' option	View the sum of all earnings and expenses across the users bank accounts.	Display the combined balance of the user's bank accounts.
Check individual account balance	Select the 'Balance' option of a given bank account	View the sum of all earnings and expenses of a specific bank account linked to the user account.	Display the balance of the user's bank account.
Cash-flow alert		The system detects a problem in the client transactions. If the client maintains the current transaction rhythm, he will not have enough cash-flow to keep his business running at the time set by the client.	Display notification.
Check clients' list	Select the 'Client list' option	View a comprehensive list of all Client User accounts within the system.	Display a list of all client users and their respective basic information.

Event name	**Input**	**Summary**	**Output**
Check client's information	Select a specific client within the Client list	View a Client User's account information in detail.	Display a detailed set of information regarding a specific client user's account.
Remove client's account	Select the 'Remove account' option within the Client list	Remove a client from the system. This option is reserved for use on users who have in some way broken the terms and conditions of use or have remained inactive within the system for long enough that they can be considered to be no longer users of the product.	The user's access is revoked and their sensitive information is wiped from the system.
Edit credit risk analysis equation	Select the 'Edit credit risk analysis equation' option	Make alterations to the system's internal logic through which it analyses a user's credit risk	Confirm update.

8 The Scope of the Product (Volere - 8)

8.1 Product Boundary

Before properly implementing the product in question, it is advisable to formally define the services it provides beforehand, so as to better maintain focus on its required features. To this end, a Use Case Diagram was modelled, based upon some of the user-related events gathered in the previous section. The use cases in which the primary users of the product and the respective actions they can take within it are expressed:

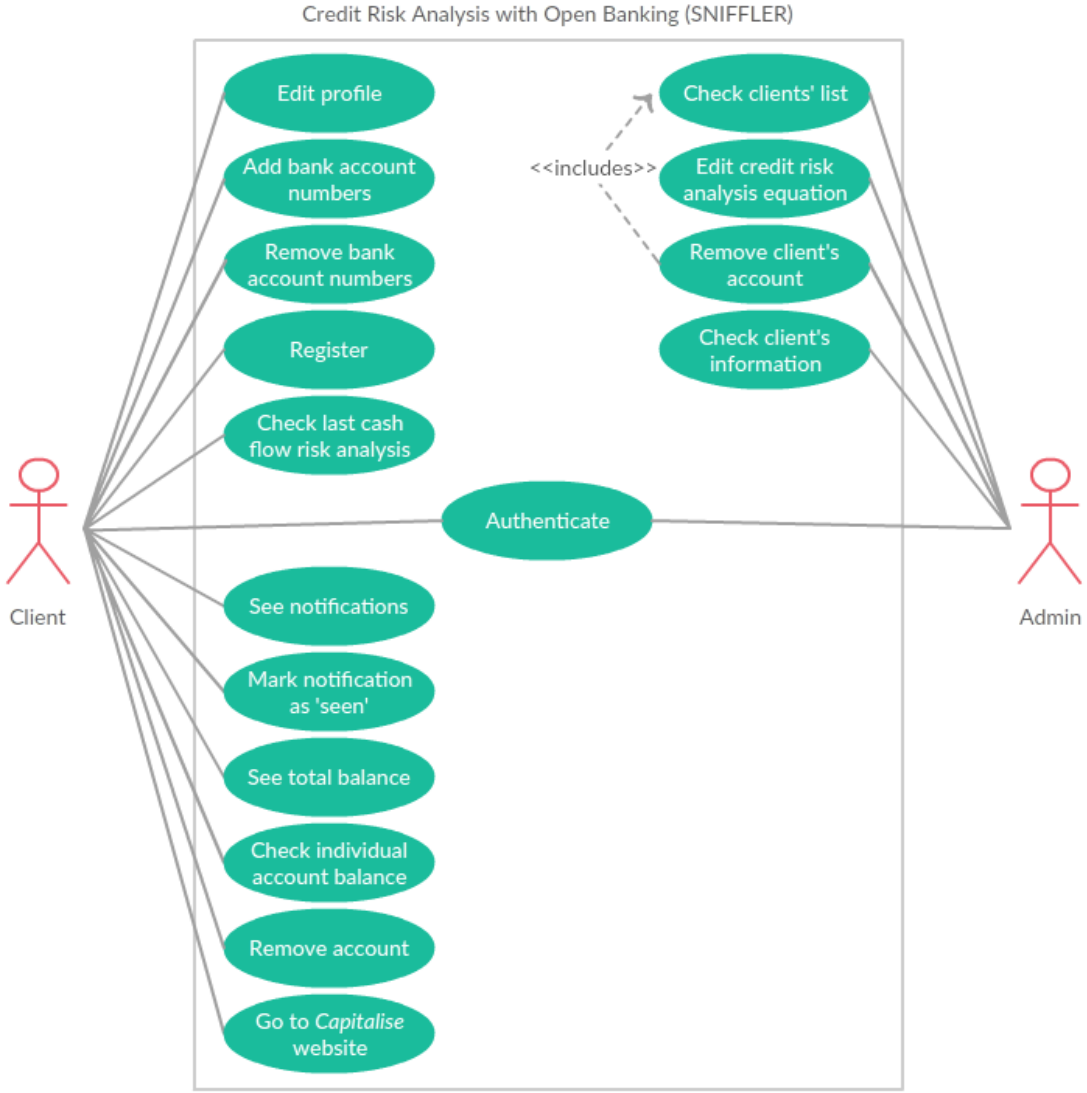

Figure 3: Use Case Diagram of the product.

8.2 Product Use Case List

1. **Authenticate**: The process through which a Client or Admin accesses the system by inputting a user name and password.
2. **Edit profile**: The process through which a Client may alter editable user data (e.g. password).
3. **Add bank account numbers**: The process of linking a bank account to a Client's user account.
4. **Remove bank account numbers**: The process of unlinking a bank account from a Client's user account.
5. **Register**: The process of creating a Client user account within the system.
6. **Check last cash flow risk analysis**: The process of requesting a summary of a Client's most recent cash flow analysis, generated by the product.
7. **See notifications**: The process through which a Client may view all system notifications.
8. **Mark notification as 'read'**: The process through which a Client may express acknowledgement of a system notification, such that it ceases to draw attention in the interface.
9. **See total balance**: The process through which a Client may request to view the sum of the balances of their bank accounts.
10. **Check individual account balance**: The process through which a Client may view the balance of a selected bank account which linked to their user account.
11. **Remove account**: The process of removing a user account from the system. This can be done by a Client at any time.
12. **Go to Capitalise website**: The process through which a Client may be directed to the Capitalise website.
13. **Check clients' list**: The process through which an Admin can request a listing of the current Clients of the system.
14. **Edit credit risk analysis equation**: The process through which an Admin may alter the logic which the system uses to calculate risk analysis.
15. **Remove client's account**: The process through which an Admin, given proper reason to do so, may remove a Client's user account from the system.
16. **Check client's information**: The process through which an Admin may look up a specific client's user information.

8.3 Individual Product Use Cases

Use Case	Authenticate
ID	1
Actors	Client User Admin User
Preconditions	1. The User must be registered in the system
Flow of events	1. The User inputs their user name. 2. The User inputs their password.
Post-conditions	1. The User is granted access to the system.

Use Case	Edit profile
ID	2
Actors	Client User
Preconditions	1. The Client User must be authenticated in the system
Flow of events	1. The Client User opens the Settings menu. 2. The Client User opens the Profile submenu of the Settings menu. 3. The Client alters editable data.
Post-conditions	1. The changes made are saved to the system.

Use Case	Add bank account numbers
ID	3
Actors	Client User
Preconditions	1. The Client User must be authenticated in the system
Flow of events	1. The Client User opens the Settings menu. 2. The Client User opens the Bank Accounts submenu of the Settings menu. 3. The Client User selects the Add bank account option of the Bank Accounts sub-menu. 4. The Client User inputs their bank account information.
Post-conditions	1. The Client User has the new bank account linked to their user account.

Use Case	Remove bank account numbers
ID	4
Actors	Client User
Preconditions	1. The Client User must be authenticated in the system
Flow of events	1. The Client User opens the Settings menu. 2. The Client User opens the Bank Accounts submenu of the Settings menu. 3. The Client User selects the Remove bank account option of the Bank Accounts sub-menu. 4. The Client User selects the bank account they wish to remove.
Post-conditions	1. The Client User no longer has the removed bank account linked to their user account.

Use Case	Register
ID	5
Actors	Client User
Preconditions	1. The user name selected for the user account must not yet be in use within the system. 2. The email account selected for the user account must not yet be in use within the system.
Flow of events	1. The Client User selects the Register option when at the Log in screen. 2. The Client User inputs a user name for the new account. 3. The Client User inputs an email account to link the new account to. 4. The Client User inputs a password for the new account. 5. The Client User confirms their choices.
Post-conditions	1. The Client User is granted the ability to access the system.

Use Case	Check last cash flow risk analysis
ID	6
Actors	Client User
Preconditions	1. The Client User must be authenticated into the system. 2. The system must have executed at least one cash flow analysis prior.
Flow of events	1. The Client User selects the Last cash flow risk analysis option. 2. The Client User selects the Download option to save a PDF file to their device, if they so choose.
Post-conditions	

Use Case	See notifications
ID	7
Actors	Client User
Preconditions	1. The Client User must be authenticated into the system.
Flow of events	1. The Client User selects the Notifcations option of the main menu. 2. The system displays the set of system notifications sent to the user in chronological order.
Preconditions	

Use Case	Mark notification as ‘read’
ID	8
Actors	Client User
Preconditions	1. The Client User must be authenticated into the system. 2. There must be at least one unread system notification sent to the Client User.
Flow of events	1. The Client User selects the Notifcations option of the main menu. 2. The system displays the set of system notifications sent to the user in chronological order. 3. The Client User selects the check mark symbol beside the notification.
Post-conditions	1. The selected notification is marked as ‘read’

Use Case	See total balance
ID	9
Actors	Client User
Preconditions	1. The Client User must be authenticated into the system. 2. The Client User must have at least one bank account linked to their user account.
Flow of events	1. The Client User selects the Bank Accounts menu. 2. The Client User selects the Balances submenu.
Post-conditions	

Use Case	Check individual account balance
ID	10
Actors	Client User
Preconditions	1. The Client User must be authenticated into the system. 2. The Client User must have at least one bank account linked to their user account.
Flow of events	1. The Client User selects the Bank Accounts menu. 2. The Client User selects the Balances submenu. 3. The Client User selects a bank account within the submenu.
Postconditions	

Use Case	Remove account
ID	11
Actors	Client User
Preconditions	1. The Client User must be authenticated into the system.
Flow of events	1. The Client User selects the Settings menu. 2. The Client User selects the Account submenu. 3. The Client User selects the Delete account option of the submenu. 4. The Client User confirms their choice.
Postconditions	1. The Client User's sensitive information is deleted from the system. 2. The Client User's account's access to the system is revoked.

Use Case	Go to Capitalise website
ID	12
Actors	Client User
Preconditions	1. The Client User must be authenticated into the system.
Flow of events	1. The Client User selects Go to Capitalise option in the main menu. 2. The system opens a new browser tab for the Capitalise website.
Postconditions	

Use Case	Check clients' list
ID	13
Actors	Admin User
Preconditions	1. The Admin User must be authenticated into the system.
Flow of events	1. The Admin User selects the 'List clients' option in the main menu. 2. The system lists a record of all Client Users within the system as well as their relevant information.
Postconditions	

Use Case	Edit credit risk analysis equation
ID	14
Actors	Admin User
Preconditions	1. The Admin User must be authenticated into the system.
Flow of events	1. The Admin User selects the 'Credit risk analysis equation' option in the main menu. 2. The system displays the credit risk analysis equation in use. 3. The Admin User selects the 'Edit equation' option. 4. The Admin User edits the equation. 5. The Admin User selects the 'Save changes' option of the 'Equation editing' menu.
Postconditions	1. The changes made to the credit risk analysis equation are saved to the system.

Use Case	Remove client's account
ID	15
Actors	Admin User
Preconditions	1. The Admin User must be authenticated into the system. 2. There must be at least one Client User in the system. 3. The Admin User must have a valid reason to remove a client's account.
Flow of events	1. Check clients' list use case 2. The Admin User selects the Client User account they wish to remove. 3. The Admin User selects the 'Remove account' option in the 'Client list' menu. 4. The Admin User confirms their selection.
Postconditions	1. The Client User's sensitive information is deleted from the system. 2. The Client User's account's access to the system is revoked.

Use Case	Check client's information
ID	16
Actors	Admin User
Preconditions	1. The Admin User must be authenticated into the system. 2. There must be at least one Client User in the system.
Flow of events	1. The Admin User selects the 'List clients' option in the main menu. 2. The Admin User selects a Client User. 3. The Admin User selects the 'Show more' option of the Client User profile.
Postconditions	1. The system displays an in-depth summary of a User Client's information.

9 Requirements

9.1 Functional Requirements (Volere - 9)

In order to better delineate and perceive the requirements, this were divided in "Clients' associated requirements", "Administrator's associated requirements" and "Both actors' associated requirements", in the form of a list. More below, then, the same requirements were formatted accordingly to the Volere's cards.

- Clients' associated requirements
    1. Edit profile
    2. Add bank account numbers
    3. Remove back account numbers
    4. Remove account
    5. Register
    6. Check last cash flow risk analysis
    7. See notifications
    8. Mark notification as seen
    9. See total accounts balance
    10. See individual account balance
    11. Go to *Capitalise* website
- Administrator's associated requirements
    1. Check clients' list
    2. Edit credit risk analysis equation
    3. Remove clients' account
    4. Check client's information
- Both actors' associated requirements
    1. Authenticate

- Clients' associated requirements
  - Edit profile

Requirement Shell		
Requirement #: 1	Requirement type: 9	Event/use case #: 2
Description: The user should be capable of editing his profile.		
Rationale: The user might want to edit his profile in order to, for example, change his phone number.		
Source: Introspection		
Fit Criterion: The system should provide the user a edit option, so he can change his phone number, email or password.		
Costumer Satisfaction: 3		Costumer Dissatisfaction: 4
Priority: Should		Conflicts:
Supporting Materials:		
History: Created 22/11/2018		

  - Add bank account numbers

Requirement Shell		
Requirement #: 2	Requirement type: 9	Event/use case #: 3
Description: The user must be able to add bank account numbers.		
Rationale: The user wants to add more bank account numbers in order to be able to get managment for all his expenses, covering all his accounts.		
Source: Scytale		
Fit Criterion: The system should provide the user the possibility to add a new bank account, to a already existing or empty list, in order to posteriorly access to the respective information of the user expenses.		
Costumer Satisfaction: 4		Costumer Dissatisfaction: 5
Priority: Must		Conflicts:
Supporting Materials: Scytale's "The state of Open Banking" document		
History: 22/11/2018		

  - Remove back account numbers

Requirement Shell		
Requirement #: 3	Requirement type: 9	Event/use case #: 4
Description: The user must be able to remove bank account numbers.		
Rationale: The user might want to remove some bank account numbers, because he closed a bank account.		
Source: Scytale		
Fit Criterion: The system should provide the user the possibility to remove a bank account number from an already existing list of numbers, in order stop the analysis considering that account.		
Costumer Satisfaction: 4		Costumer Dissatisfaction: 5
Priority: Must		Conflicts:
Supporting Materials: Scytale's "The state of Open Banking" document		
History: 22/11/2018		

– Remove account

Requirement Shell		
Requirement #: 4	Requirement type: 9	Event/use case #: 11
Description: The user must be able to remove his account.		
Rationale: The user wants to remove his account because, for example, he no longer wants to have his account analysed, or he is unhappy with the service.		
Source: Introspection		
Fit Criterion: The system must arrange the user the possibility to remove his account, so he can stop having his accounts evaluated for credit risk.		
Costumer Satisfaction: 3	Costumer Dissatisfaction: 5	
Priority: Must	Conflicts:	
Supporting Materials:		
History: 23/11/2018		

– Register

Requirement Shell		
Requirement #: 5	Requirement type: 9	Event/use case #: 5
Description: The user must be able to register to the system.		
Rationale: The user ought to register to the system in order to manage his spendings, use the cash flow risk analysis tool and possibly and posteriorly request money.		
Source: Scytale		
Fit Criterion: The system must present a register option and ask for username, password, email, name and phone number.		
Costumer Satisfaction: 5	Costumer Dissatisfaction: 5	
Priority: Must	Conflicts:	
Supporting Materials:		
History: Created 23/11/2018		

– Check last cash flow risk analysis

Requirement Shell		
Requirement #: 6	Requirement type: 9	Event/use case #: 6
Description: The user must be able to check last cash flow risk analysis.		
Rationale: At any time, the user might want to check the last automatic cash flow risk analysis done by the system, in order to analyze the monthly spending results for his accounts and transactions and keeping informed about his financial situation.		
Source: Scytale		
Fit Criterion: The system should provide the user a way for him to see the report of the last cash flow risk analysis, as well as the history of all the analysis ever done, monthly. The evaluations should appear in a list form, sorted by date.		
Costumer Satisfaction: 5	Costumer Dissatisfaction: 5	
Priority: Must	Conflicts:	
Supporting Materials: Scytale's "The state of Open Banking" document		
History: Created 24/11/2018		

– See notifications

Requirement Shell		
Requirement #: 7	Requirement type: 9	Event/use case #: 7
Description: The user wants to see his notifications about new cash flow risk analysis.		
Rationale: The user should be able to check his notifications, in order to be aware of a new cash flow risk analysis.		
Source: Scytale		
Fit Criterion: The system should present the user all his notifications, whenever he wants to check them. This alerts should be arranged by date.		
Costumer Satisfaction: 3		Costumer Dissatisfaction: 5
Priority: Must		Conflicts:
Supporting Materials:		
History: 24/11/2018		

– Mark notification as seen

Requirement Shell		
Requirement #: 8	Requirement type: 9	Event/use case #: 8
Description: The user should be capable of marking his notifications about a new cash flow risk analysis as seen.		
Rationale: The user wants to mark his notifications as seen, so he can better manage what he has already seen or not.		
Source: Introspection		
Fit Criterion: The system should present the user all his notifications, whenever he wants to check them, and have a marker option so he can check the past and seen notifications.		
Costumer Satisfaction: 3		Costumer Dissatisfaction: 4
Priority: Should		Conflicts:
Supporting Materials:		
History: 24/11/2018		

– See total accounts balance

Requirement Shell		
Requirement #: 9	Requirement type: 9	Event/use case #: 9
Description: The user must be able to see the total accounts' balance.		
Rationale: At any time, the user might want to check his total balance, in order to manage his finances and control his expenses..		
Source: Scytale		
Fit Criterion: The system should provide the user a way for him to see his total balance, by summing all the individual accounts' balances.		
Costumer Satisfaction: 5		Costumer Dissatisfaction: 5
Priority: Must		Conflicts:
Supporting Materials: Scytale's "The state of Open Banking" document		
History: 26/11/2018		

– See individual account balance

Requirement Shell		
Requirement #: 10	Requirement type: 9	Event/use case #: 10
Description: The user must be able to see an individual account balance.		
Rationale: At any time, the user might want to check an individual account balance, for the purpose of managing his finances and control his expenses.		
Source: Scytale		
Fit Criterion: The system should furnish the user a way for him to see an individual account balance, displaying all the balances per account. The system should also let the user to choose the account for which he wants the balance to be shown.		
Costumer Satisfaction: 5		Costumer Dissatisfaction: 5
Priority: Must		Conflicts:
Supporting Materials: Scytale's "The state of Open Banking" document		
History: 26/11/2018		

– Go to *Capitalise* website

Requirement Shell		
Requirement #: 11	Requirement type: 9	Event/use case #: 12
Description: The user must be capable of accessing Capitalise website through the application.		
Rationale: At any time, the user might want to go to Capitalise website through the application, in order to ask for credit.		
Source: Scytale		
Fit Criterion: The system should give the the user a way for him to access Capitalise website.		
Costumer Satisfaction: 4		Costumer Dissatisfaction: 4
Priority: Must		Conflicts:
Supporting Materials:		
History: 26/11/2018		

- Administrator's associated requirements
  - Check clients' list

Requirement Shell		
Requirement #: 12	Requirement type: 9	Event/use case #: 13
Description: The user must be capable of checking the list of all the clients.		
Rationale: The user wants to check a list of all the clients of the system, in order to compile statistics and analyse the data.		
Source: Scytale		
Fit Criterion: The system must provide the user a list with all the clients registered in the application. This list should come with minimal information about the client, such as only the username, name and email.		
Costumer Satisfaction: 4		Costumer Dissatisfaction: 4
Priority: Must		Conflicts:
Supporting Materials:		
History: 27/11/2018		

  - Edit credit risk analysis equation

Requirement Shell		
Requirement #: 13	Requirement type: 9	Event/use case #: 14
Description: The user must be capable of editing the credit risk analysis equation.		
Rationale: The user wants to edit the cash flow risk analysis, in order to change certain mathematical parameters that improve the evaluation done to customers' accounts.		
Source: Scytale		
Fit Criterion: The system must provide the user a way for him to access and change the equation for the credit risk analysis.		
Costumer Satisfaction: 4		Costumer Dissatisfaction: 5
Priority: Must		Conflicts:
Supporting Materials:		
History: 28/11/2018		

  - Remove clients' account

Requirement Shell		
Requirement #: 14	Requirement type: 9	Event/use case #: 15
Description: The user must be able of removing a clients' account.		
Rationale: The user might want to remove a specific clients' account because, for example, of his bad behaviour at the bank level.		
Source: Introspection		
Fit Criterion: The system must provide the user a way for him to select and remove, from the list of clients, a specific clients' account. When this happens, the system must provide a message with the reason for the blocking, when the user accesses his account.		
Costumer Satisfaction: 4		Costumer Dissatisfaction: 4
Priority: Must		Conflicts:
Supporting Materials:		
History: 28/11/2018		

- Check client's information

Requirement Shell		
Requirement #: 15	Requirement type: 9	Event/use case #: 16
Description: The user must be capable of checking clients' information.		
Rationale: The user might want to check some information about a specific client.		
Source: Introspection		
Fit Criterion: The system must provide the user a way for him to check a client's personal information. However, the provided information must be limited, including only the clients's username and name.		
Costumer Satisfaction: 4		Costumer Dissatisfaction: 4
Priority: Must		Conflicts:
Supporting Materials:		
History: 02/12/2018		

- Both actors' associated requirements
  - Authenticate

Requirement Shell		
Requirement #: 16	Requirement type: 9	Event/use case #: 1
Description: The user must be able to authenticate to the application.		
Rationale: The user wants to authenticate to the application, so he can access his services of credit risk analysis.		
Source: Introspection		
Fit Criterion: The system must provide the user a way for him to authenticate. In order to do that, it must be asked for a password and a username. In alternative to this way of authentication, the system should allow the user to insert an email and password.		
Costumer Satisfaction: 4		Costumer Dissatisfaction: 4
Priority: Must		Conflicts:
Supporting Materials:		
History: 02/12/2018		

9.2 Data Requirements (Volere - 9)

In order to specify the essential context of the *Sniffler* product, its stakeholders and relevant entities, a domain model was developed using UML notation. With this diagram, it is intended to provide a very high-level overview of all the major components of the system, as well as the interaction between them.

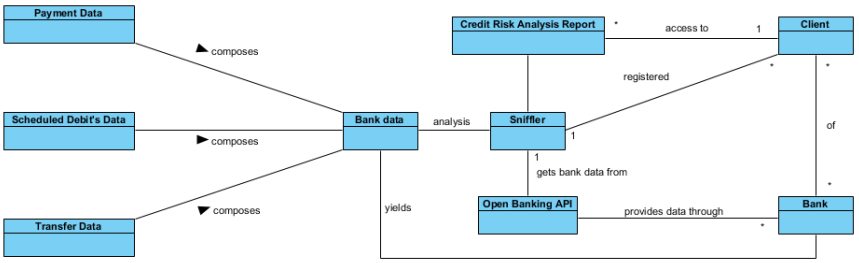

Figure 4: Domain Model representing an overview of the system.

9.3 Non-Functional Requirements

Next are presented each type of non-functional requirements. All of these are, again, formatted accordingly to the Volere's cards. In each section, a small list with a small description of the requirements is shown, in order to better understand and read the cards.

9.3.1 Look and Feel (Volere - 10)

Requirements' list in this section:

- The product should be appealing to a young / young-adult audience (+18y);
- The product must conform to Scytale's trademark graphic appearance standards;
- The product must look reliable.

Requirement Shell		
Requirement #: 17	Requirement type: 10	Event/use case #:
Description: The product should be appealing to a young / young-adult audience (+18years).		
Rationale:		
Source: Introspection		
Fit Criterion: A sample of adults / young adults should be able, without any purposeful stimulus, to initialize the use of the product within 5 minutes of their first contact with it.		
Costumer Satisfaction: 4		Costumer Dissatisfaction: 4
Priority: Should		Conflicts:
Supporting Materials:		
History: 03/01/2019		

Requirement Shell		
Requirement #: 18	Requirement type: 10	Event/use case #:
Description: The product must conform to Scytale's trademark graphic appearance standards.		
Rationale:		
Source: Scytale		
Fit Criterion: The trademark department must certify that the product conforms to current standards.		
Costumer Satisfaction: 5		Costumer Dissatisfaction: 4
Priority: Must		Conflicts:
Supporting Materials:		
History: 03/01/2019		

Requirement Shell		
Requirement #: 19	Requirement type: 10	Event/use case #:
Description: The product must look reliable.		
Rationale:		
Source: Scytale		
Fit Criterion: After the first meeting with the platform, 70 % of a representative population should agree that they feel they can rely on the product.		
Costumer Satisfaction: 5		Costumer Dissatisfaction: 5
Priority: Must		Conflicts:
Supporting Materials:		
History: 03/01/2019		

9.3.2 Usability and Humanity (Volere - 11)

Requirements' list in this section:

- The product must help the user avoid making mistakes;
- The product must make the user want to use it;
- The product should require only a few steps to the user, in order to access to the majority of the functionalities;
- The product must allow the user to choose a language;
- The product must conform to the currency (coin), including symbols and conventions, of the user;
- The product must be easy for an adult / young-adult to learn;
- The product must use symbols and words that are naturally understood by the user community.

Requirement Shell		
Requirement #: 20	Requirement type: 11	Event/use case #:
Description: The product must help the user avoid making mistakes.		
Rationale:		
Source: Introspection		
Fit Criterion: One month of use of the product should result in a total error rate of less than 1%.		
Costumer Satisfaction: 4		Costumer Dissatisfaction: 5
Priority: Must		Conflicts:
Supporting Materials:		
History: 03/01/2019		

Requirement Shell		
Requirement #: 21	Requirement type: 11	Event/use case #:
Description: The product must make the user want to use it.		
Rationale:		
Source: Scytale		
Fit Criterion: An anonymous survey should demonstrate that target users are regularly using the product after a period of 2 months of familiarization.		
Costumer Satisfaction: 4		Costumer Dissatisfaction: 5
Priority: Must		Conflicts:
Supporting Materials:		
History: 03/01/2019		

Requirement Shell		
Requirement #: 22	Requirement type: 11	Event/use case #:
Description: The product should require only a few steps to the user, in order to access to the majority of the functionalities.		
Rationale:		
Source: Introspection		
Fit Criterion: One month of product usage should result in a time to accomplish any task in less than 3 minutes.		
Costumer Satisfaction: 3		Costumer Dissatisfaction: 4
Priority: Should		Conflicts:
Supporting Materials:		
History: 03/01/2019		

Requirement Shell		
Requirement #: 23	Requirement type: 11	Event/use case #:
Description: The product must allow the user to choose a language.		
Rationale: The system should have the following languages available: portuguese (pt), english, french, chinese and german.		
Source: Introspection		
Fit Criterion:		
Costumer Satisfaction: 5		Costumer Dissatisfaction: 4
Priority: Must		Conflicts:
Supporting Materials:		
History: 04/01/2019		

Requirement Shell		
Requirement #: 24	Requirement type: 11	Event/use case #:
Description: The product must conform to the currency(coin), including symbols and conventions, of the user.		
Rationale: The system must have all currencies registered worldwide, achieved through open banking API (according to the participating banks).		
Source: Scytale		
Fit Criterion:		
Costumer Satisfaction: 5		Costumer Dissatisfaction: 5
Priority: Must		Conflicts:
Supporting Materials:		
History: 04/01/2019		

Requirement Shell		
Requirement #: 25	Requirement type: 11	Event/use case #:
Description: The product must be easy for an adult / young-adult to learn.		
Rationale:		
Source: Scytale		
Fit Criterion: An adult / young adult should be able to use 95% of the platform within 10 minutes of the start of use of the product without the need to consult the 'help'.		
Costumer Satisfaction: 5		Costumer Dissatisfaction: 4
Priority: Must		Conflicts:
Supporting Materials:		
History: 04/01/2019		

Requirement Shell		
Requirement #: 26	Requirement type: 11	Event/use case #:
Description: The product must use symbols and words that are naturally understood by the user community.		
Rationale:		
Source: Introspection		
Fit Criterion: An untrained user in the area of economics / management should be able to understand 99% of any symbols and words on the platform.		
Costumer Satisfaction: 4		Costumer Dissatisfaction: 4
Priority: Must		Conflicts:
Supporting Materials:		
History: 04/01/2019		

9.3.3 Performance (Volere - 12)

Requirements' list in this section:

- The system's response must be fast enough to avoid interruption in the user's thought flow;
- Any monetary amount in the system that has decimal point must be exact in 2 decimal places;
- The product must be available for most of the time in a year;
- The system must support the registration of at least 1000 users within 1 year.

Requirement Shell		
Requirement #: 27	Requirement type: 12	Event/use case #:
Description: The system's response must be fast enough to avoid interruption in the user's thought flow.		
Rationale:		
Source: Introspection		
Fit Criterion: The product must respond in less than 2 seconds to 90% of requests. No answer should take more than 3s.		
Costumer Satisfaction: 4		Costumer Dissatisfaction: 4
Priority: Must		Conflicts:
Supporting Materials:		
History: 04/01/2019		

Requirement Shell		
Requirement #: 28	Requirement type: 12	Event/use case #:
Description: Any monetary amount in the system that has decimal point must be exact in 2 decimal places.		
Rationale: Dependind on the coin in use, the system must always be exact in 2 decimal places. This does not happens in some currencies have no decimal places, such as the Japanese yen.		
Source: Scytale		
Fit Criterion:		
Costumer Satisfaction: 4		Costumer Dissatisfaction: 5
Priority: Must		Conflicts:
Supporting Materials:		
History: 04/01/2019		

Requirement Shell		
Requirement #: 29	Requirement type: 12	Event/use case #:
Description: The product must be available for most of the time in a year.		
Rationale:		
Source: Scytale		
Fit Criterion: The product must be available, in the worst case, 99% of the time in a year (361 in 365 days).		
Costumer Satisfaction: 4		Costumer Dissatisfaction: 4
Priority: Must		Conflicts:
Supporting Materials:		
History: 04/01/2019		

Requirement Shell		
Requirement #: 30	Requirement type: 12	Event/use case #:
Description: The system must support the registration of at least 1000 users within 1 year.		
Rationale:		
Source: Scytale		
Fit Criterion:		
Costumer Satisfaction: 4		Costumer Dissatisfaction: 5
Priority: Must		Conflicts:
Supporting Materials:		
History: 04/01/2019		

9.3.4 Operational (Volere - 13)

Requirements' list in this section:

- The product must work in the last 4 editions of the 5 most popular browsers.

Requirement Shell		
Requirement #: 31	Requirement type: 13	Event/use case #:
Description: The product must work in the last 4 editions of the 5 most popular browsers.		
Rationale:		
Source: Scytale		
Fit Criterion: Between the 5 more popular browsers are Google Chrome, Firefox, Internet Explorer, Opera e Safari.		
Costumer Satisfaction: 4		Costumer Dissatisfaction: 5
Priority: Must		Conflicts:
Supporting Materials:		
History: 04/01/2019		

9.3.5 Maintainability and Support (Volere - 14)

Requirements' list in this section:

- It is expected that the product must work on any operating system.

Requirement Shell		
Requirement #: 32	Requirement type: 14	Event/use case #:
Description: It is expected that the product must work on any operating system.		
Rationale: It is expected that the product works on any operating system, since from a browser and with internet access.		
Source: Scytale		
Fit Criterion:		
Costumer Satisfaction: 4		Costumer Dissatisfaction: 5
Priority: Must		Conflicts:
Supporting Materials:		
History: 04/01/2019		

9.3.6 Security (Volere - 15)

Requirements' list in this section:

- The system must garantee that no one can have access to the banking data of the users, except the user himself;
- The system must garantee that only the administrator can have access to some of the user's personal data, such as username and name;
- The product must prevent incorrect data from being entered, such as verifying login data and verifying entered bank numbers;
- The product must make its user aware of its information practices before collection data from them;
- The product must notify customers of changes to its information policy.

Requirement Shell		
Requirement #: 33	Requirement type: 15	Event/use case #:
Description: The system must garantee that no one can have access to the banking data of the users, except the user himself.		
Rationale:		
Source: Scytale		
Fit Criterion:		
Costumer Satisfaction: 5		Costumer Dissatisfaction: 5
Priority: Must		Conflicts:
Supporting Materials:		
History: 05/01/2019		

Requirement Shell		
Requirement #: 34	Requirement type: 15	Event/use case #:
Description: The system must garantee that only the administrator can have access to some of the user's personal data, such as username and name.		
Rationale:		
Source: Scytale		
Fit Criterion:		
Costumer Satisfaction: 4		Costumer Dissatisfaction: 5
Priority: Must		Conflicts:
Supporting Materials:		
History: 05/01/2019		

Requirement Shell		
Requirement #: 35	Requirement type: 15	Event/use case #:
Description: The product must prevent incorrect data from being entered.		
Rationale: The product must prevent incorrect data from being entered, such as verifying login data and verifying entered bank numbers.		
Source: Introspection		
Fit Criterion:		
Costumer Satisfaction: 4		Costumer Dissatisfaction: 4
Priority: Must		Conflicts:
Supporting Materials:		
History: 05/01/2019		

Requirement Shell		
Requirement #: 36	Requirement type: 15	Event/use case #:
Description: The product must make its user aware of its information practices before collection data from them.		
Rationale:		
Source: Introspection		
Fit Criterion:		
Costumer Satisfaction: 4		Costumer Dissatisfaction: 5
Priority: Must		Conflicts:
Supporting Materials:		
History: 05/01/2019		

Requirement Shell		
Requirement #: 37	Requirement type: 15	Event/use case #:
Description: The product must notify customers of changes to it's information policy.		
Rationale:		
Source: Introspection		
Fit Criterion:		
Costumer Satisfaction: 4		Costumer Dissatisfaction: 5
Priority: Must		Conflicts:
Supporting Materials:		
History: 05/01/2019		

9.3.7 Legal (Volere - 17)

Requirements' list in this section:

- The product must comply with all legal standards of Open Banking.

Requirement Shell		
Requirement #: 38	Requirement type: 17	Event/use case #:
Description: The product must comply with all legal standards of Open Banking.		
Rationale:		
Source: Scytale		
Fit Criterion:		
Costumer Satisfaction: 5		Costumer Dissatisfaction: 5
Priority: Must		Conflicts:
Supporting Materials:		
History: 05/01/2019		

10 Risks (Volere - 23)

Like any software product being commercialized, particularly the *Sniffer* product, it presents a number of risks that can potentially lead the project to failure.

So let's take a look at the risks to take into consideration for this product.

Since this is a revolutionary product in the financial arena, there is a possibility that the target audience will not use the product, making it so useless. In addition, the absence of similar implementations already in place, makes the product less interesting. Furthermore, the attitude of users of financial products is particularly conservative as the security of their banking information is compromised.

Taking into account that this product is dependent on obtaining bank information provided by an API (Open Banking) , if the banking entities do not respect 100% of the standard or that it is not available within the time period for the product to be launched on the market , may prevent potential users from enjoying the service.

Finally, given the experience of the development team (still under development), the lack of full documentation and quality of the functioning of the API as well as the lack of knowledge in the financial area can also be decisive for the non-success of this product.

11 Appendix

11.1 Interview

At the beginning of a project, the most important topic is to understand its goals, and what must be done to achieve them. This project was proposed by the company *Scytale*, so the start point was to join all the information they had. As such, an interview was made, between us and the company, being the meeting point the *Scytale* address, clarifying all the points about the project.

What does the project actually consist of?

So, here at Scytale we build the software the client asks for. Our main product is *Capitalise*. It is a marketplace project, that allows the gain of capital without resorts to banks.
This Credit Risk Analysis project is a complement of *Capitalise*. The potential clients will be able to get some safety on their finances, because it will be made a calculation of credit risk. In case of any risk is detected, it means the client is in a *cash flow* situation, then *Capitalise* is proposed.
It involves the concepts of *Open Banking*.

What is the Open Banking?

Banks are suffering pressure to get modernised. There are MBWays, Revolute, N26, etc., all of them services that have many advantages like no need of exchange rates for those people who travels. With Open Banking, there is less interaction between the bank and the intermediate.
An application that can manage bank balance must have access to different bank accounts information.

So, this project only involves clients? Or companies too?

No, just clients. They will receive an "alert" when in a situation of cash flow failure.

Besides the clients, there will be no companies, but an administrator. He can check all the clients registered in the application, change some criteria in the calculation to the risk analysis, remove client account and check information about some client.

What must the client do to enter in the application? And how will the system know if he's in a *cash flow* failure situation?

He must be registered in the system. Then, sign in with his email and password. Once signed in, he will be able to add as many bank accounts as he wants to his profile. And of course, if the client wants to remove some account, he must be able to do that too. The system will execute the cash flow risk analysis monthly to all of his accounts.

If the risk analysis is calculated and the client isn't "online", he will receive notifications?

Yes, the client must receive notifications about the cash flow risk analysis made to his bank accounts. That's the way he has to know his situation. After he sees the notification, he must mark it as "seen", otherwise he will always receive the same notification.

What are the technologies you usually work with?

The technologies we usually work, but it's not a constraint, are:

- Database: MongoDB, MySQL
- API Developping: NodeJS

- Data processing: Scala, Lang
- Front-End Web: React, HTML, CSS
- Front-End Mobile: React Native, Ionic

11.2 *Personas*

Making decisions without knowing the target audience and collect the most possible information is a very big step towards failure. Making decisions based on what the developer thinks the client want is not sustainable is a big risk to be taken.

That's why connecting with the people that it is going to use the product is crucial to its success. To organise all the information obtained, *personas* are the best model to do so.

Below are some *personas* the projects' analysts team collected:

Name	José António Ribeiro Meira	
Area	Textile Industry	
Position	Business owner	
About	Tech skills	●●●○○
	Business dimension	●●○○○
	Cash-flow problems frequency	●●○○○
	Trust in Open Banking	●○○○○
	Need for easy and fast credit	●●●●○
	Fifty-six year-old José Meira is a owner of a small textile industry business in Portugal. After passing a down period during the economical crisis of the decade, José's business is based now in exportations but due to the large number of clients, sometimes it's hard for José to keep track on the business' finances.	
Needs	Financial management tool	
	Easy client ordering system	
	Technologisation of the business	

Name	Maria Helena Teixeira Carvalho	
Area	Seamstress business	
Position	Business manager	
About	Tech skills	●●●●○
	Business dimension	●○○○○
	Cash-flow problems frequency	●●○○○
	Trust in Open Banking	●●●○○
	Need for easy and fast credit	●●●○○
	Thirty-two year-old Maria Carvalho is the manager of her mother's business. The company is specialised in small sewing jobs for the local population. Actually going through a new business idea, the plan is to open to other type of clients. So, a new approach to the financial management is needed.	
Needs	Financial management tool	
	More business growing ideas	
	Technologisation of the business	

Name	Ana Cristina Mendes Pessoa	
Area	Transportation business	
Position	Accounting manager	
About	Tech skills	●●●●●
	Business dimension	●●●●○
	Cash-flow problems frequency	●●●○○
	Trust in Open Banking	●●●●○
	Need for easy and fast credit	●●●●●
	Cristina Pessoa is the manager of the accounting team at a big dimension transportation company based in Portugal. The company is going through a downsizing so, Cristina needs to invest in technology to replace the co-workers who worked at her department.	
Needs	Financial management tool	
	Financial controlling system	
	Technologisation of the business	

Based upon these collected *personas*, we can concentrate on the real opinions given by the actual future users of the system to be developed.

References

[1] James Robertson and Suzanne Robertson. *Volere Requirements Specification Template*. 11th edition, The Atlantic Systems Guild Inc., 2006.

[2] *The State of Open Banking* [Whitepaper]. LendIt Fintech, 2018.