POLITECNICO
MILANO 1863

Elk River project

Digital Innovation Lab

LUCA COSTA

RICCARDO FACCHINI

NICCOLÒ POZZOLINI

NIKOLA VUKOVIC

June 10, 2019

Deliverable specific information

Deliverable:	Project report
Title:	Requirement Analysis and Verification Document
Authors:	Luca Costa - Riccardo Facchini - Niccolò Pozzolini - Nikola Vukovic
Version:	1.0
Date:	June 10, 2019
Copyright:	Copyright © 2019, Costa - Facchini - Pozzolini - Vukovic – All rights reserved

Contents

List of Figures

List of Tables

1 Introduction

1.1 The Elk River problem

The Elk River is a situated in central West Virginia, USA. Elk River watershed is a relatively big but low populated area with Charleston being the only big city. It is characterized by vast natural resources and less developed industry and infrastructure. In 2014, one of the biggest employers in the area was involved in a major ecological incident, namely a chemical spill, which led to closure of the plant, leaving many people jobless and making impact on the local flora and fauna. Since then, the area has not been able to resume economic development, but still it attracted a lot of tourists, especially fishers who like to enjoy many fishing spots that the river offers. After analyzing the current situation in the area and all groups of people living and visiting the area, it becomes evident that one of the things that is most probable to succeed in improving the well-being of all groups could be improving the tourist offering. The goal of this document is to further explain the platform using user-centered approach.

1.2 Stakeholders

Stakeholders are the locals, whose economy is struggling since the 2014 spills. Many of them will be able to establish a job position with a stable income thanks to our service, with the possibility of working part time as a second job. Our platform will also boost the economy of the area by attracting tourists, so many activities already in place will indirectly benefit from the service, and others will born.

1.3 Definitions and acronyms

What follows is the list of all the main definitions and acronyms used in the document.

1.3.1 Definitions

- **Reservation**: data referring to the wish of the user to have the specified service at that time for himself.
- **System**: All the software needed to deliver every functionality needed.

1.3.2 Acronyms

- **BPMN**: Business Process Model and Notation
- **SDK**: Software Development Kit
- **API**: Application Programming Interface
- **DB**: Database
- **DBMS**: Database Management System
- **UID**: Unique Identifier
- **URL**: Uniform Resource Locator
- **UI**: User Interface

2 Overall Description

2.1 Elk River area resources

The Elk River area doesn't have particular infrastructures outside of asphalted roads connecting one small countryside village to the others, more or less coasting the entire length of the river. Therefore the main and basically only resources are the ones given by the environment itself, meaning that the rural composition of the area allows for sports and activities that aren't possible in a city center, such as fishing, rafting, camping, hunting etc...

Some structures like restaurants and also hotels/farmhouses where one could spend a weekend are already present in the area, meaning that they would work as an incentive for people living further away to visit the area anyway without the need of doing two long drives in one single day given the lack of public transport outside of the main cities.

2.2 Proposed project solution

As we will discuss more in detail during the next sections, we decided to opt for a solution involving fishing activities, given that some of the few records of what people do while visiting the Elk River area is indeed fishing related.
It should be noted that the system developed could be easily scaled up to accommodate other activities but we chose to keep it simpler given that it's meant to be a first prototype and it seemed more logical to focus on what looked like the main venture.

2.3 Project structure

2.3.1 Data Analysis

We conducted some data analysis in section 6.3 to find what the potential customers may need and/or want, the main techniques used were:

1. Clustering
2. Classification

2.3.2 Prototype

The final system is going to be divided in two main components:

1. Mobile application for phones and tablets.
2. Backend structure to support the functioning of the service.

While the backend structure is needed for the functioning of the service provided, the user will never interact with it but will ever only see and use the mobile application. A more detailed view will be explained in 8.1.

2.4 Product Functions

1. Register to the system with email and password or other services (i.e. Gmail).
2. Logging into the service.
3. Manage the information of an account.
4. Create and delete a reservation.
5. Provide a communication system between customers and employees.

2.5 User Characteristics

The users interested in using the system should be at least familiar with the concept of using a smartphone in the day to day routine without needing any technical competence regarding the topic, they must be aware of the laws regarding fishing, know how to traverse the local environment and have basic first aid knowledge if they wish to work as fishing experts.

2.6 Goals

The project is designed to satisfy the user needs, or (in other words) to achieve certain specific *Goals* stated in the following list.

- Allow anyone that owns a smartphone to become a registered user of the service.
- Allow locals to find a new source of income.
- Allow tourists to have direct access to experts of the area.
- Allow tourists to find fishing spots suggested by experts.
- Bring outside people to the Elk River area with the intent of boosting the economy through fishing and tourism.

3 Target Group and Personas

3.1 Target Group - Fishing Enthusiasts

The main purpose of the application is to bring closer people living in the area, mainly those that are fishing pros and people who come to the area because of fishing and other similar activities in the nature, everything to the economic benefit of locals without damaging the already fragile environment. It is expected that it could also attract new people unaware of the region to visit the area. Since we decided to focus on fishing aspect, our target group are **fishing enthusiasts**, which can actually be divided into two different groups:

Figure 1: Fishing Enthusiast.

- Locals.
- Tourists.

3.1.1 Their needs

Locals need a new income system, something new that could bring jobs and opportunities in the area using the natural environment of the zone such as the river and the woods that follow the Elk River.

Tourists coming from other counties want instead a different experience, somewhere new where they can fish different species and enjoy a new area different from the ones they usually go to.

3.1.2 What are we offering them?

A service that will facilitate the organization and the building of an infrastructure for fishing enthusiasts using modern technologies, this would create new job opportunities (such as local guides, fishing instructors, boat rentals, ecc...) for the locals and allow tourists to easily come to the Elk River and enjoy a weekend without the stress of organizing everything by themselves.

3.1.3 Other information

How many are there? About 1.5Million.

How many will we reach? Around 50.000 people.

How frequently will we interact with them? Weekly.

What do we get in return? Economic growth.

How can our relationship grow? With people talking about the service.

3.2 Personas

3.2.1 Bob The Fishing Enthusiast

Figure 2: Bob the local.

Who is Bob? Bob (40) is father of 2 young children (10 and 12) and has lived his entire life in the small town (700 people) of Webster Springs West Virginia. He used to work at Freedom Industries before the spill of 2014 when shortly after the company filed for bankruptcy and Bob was left without a job. Given that he never attended college and the little possibilities in his hometown, Bob now works at his wife's bakery while he is still searching for another job to help the family put aside some money for the children tuition and health care plan.

His interests: Interests: Fishing, camping, hiking, restoring classic cars and classical music.

Reasons for him to engage with us: He is looking for a new job, knows the territory and believes it can be used for tourism and fishing enthusiasts like himself, has been fishing all his life and would love to teach people how to fish.

Reasons for him NOT to engage with us: He isn't sure it would provide enough for his family the water is still not completely clean, too many tourists may destroy the natural environment of the area.

His skills: Fishing expert, Great sense of direction, it's almost impossible for him to get lost in the local area he has been roaming since childhood, good survival skills, basic medical training, he can also cook the best trout on the grill of the state.

His typical day: Bob wakes up with his wife every weekday at 4:40 AM and they both head up to the bakery in order to prepare everything the little community needs during the day. At around 7:00 AM he takes the little van they have for the activity and starts the delivery for the other businesses of the area and once he is done he goes back home to work on his latest car project until he has to go and pick up the kids from school. Once home again, they eat all together and then he watches some baseball with the kids before helping then with their homework until it's time for dinner. After the boys are in bed he likes to relax for some time on his chair listening to some classical music before going to bed himself, always waiting for the next fishing trip during the weekend.

His personality: Bob is a kind man that always has a smile on his face, no matter the situation. He is an optimist at heart and always believes that everything will be alright and no problem can't be solved if people work together. He is always the first one to volunteer when work for the community has to be done and takes pride and joy in helping the others when they are in need, often refusing any form of compensation.

His social environment: Active member of the community in the small town of Webster Springs West Virginia he is well known and respected. He's usually more forward thinking than the older people that live in town and since he is more accepting of new ideas for the future he often can convince the others to back up new opportunities.

His dreams: Have an independent job, spend more time with his kids, provide more for his family, buy a 1968 Mustang and then restore it with his children

Jack The Tourist Fisherman

Figure 3: Jack the tourist.

Who is Jack? Jack(36) is a father of a little girl (6 years) and lives with her and his wife in Columbus, Ohio. He is an IT technician in a medium size company where he has been working since 2007 and is a respected employee. He never obtained a degree but studies IT in high school and has always been tech savvy. Jack's father always brought him to do outdoors activities when he was younger, from rafting to camping, he always loved playing football until he got a knee injury, but his true passion has always been fishing, which he does to this day with his father and friend.

His interests: Fishing, camping, driving, football, rafting and bowling.

Reasons for him to engage with us: He wants a new place to spend the weekend fishing, even though he knows how to fish the "usual" way he never tried fly fishing and would like to try, he also loves natural places and off the grid camping spots.

Reasons for him NOT to engage with us: He's never been to West Virginia, he knows of the spill and still doesn't trust the area after it and it's a longer drive than usual.

His skills: Fishing expert, tech enthusiast, problem solver, knows his way in the great outdoors.

His typical day: Jack wakes up every weekday at 7:00 AM with the family, they all have breakfast together and then he brings his daughter to preschool and then goes to work which starts at 8:00 AM. Around 12:30 AM he heads back home for lunch, which he usually prepares when his wife arrives home with the little girl. After lunch he helps his daughter to bed for the afternoon nap and heads back to work for the afternoon shift. Once home again in the evening they have dinner and spend some more time together watching some TV until it's bedtime for the kid, once she is asleep Jack usually caches up with the latest football updates until it's time for him too to go to sleep.

His personality: Jack is a loving father and husband, a great friend and he is know at work for his great work ethic. He rarely gets upset and is always glad to learn something new, from a simple fact, to a new tech that is going to change the marker, up to a better way to do his job. He worked for what he has but does not like to remind people of this and is a great teacher, whether it is on the job, at the bowling alley or during a fishing trip with someone new joining the group.

His social environment: One of the most experienced people in the company he works for and between the ones that have been there since the early years, Jack is well known and respected by his colleagues and loved by his friends. He is always present at every social event, from birthdays to a quite evening in the bar.

His dreams: Become a manager of the company, teach his daughter to fish, buy a house with a pond.

4 Thinking Hats

	Scarcely populated area, zone has been polluted in 2014, avoid eating too many fish, supposedly people lost their jobs after the spill, no economic growth, rich of natural places where it is possible to camp, raft, fish etc. River is not very deep (2m), road are present but not state of the art, no railroad nor public transportation. The only economic/social center is Charleston at one of the ends of the river. The area is already known as a fishing spot.
	I feel sad about the situation in the zone and the people living there, it does not feel like a place where I would go to spend my time as it seems to be a shadow of its former self. I would work on the place using the extensive natural resources to bring value to the zone.
	The lack of jobs and the unemployed could easily start a new work environment based on a new system that we can propose. The area is already known as a fishing spot and we could use that to start a new flow of people as fishing tourists. Given that the river is quite long, the amount of people we can attract is as large as the area can afford to hold and manage. Giving new jobs to the people will surely improve the quality of life in the area.
	The health care institution suggest not eating more than one fish per month because of the pollution, the river be a hazard if there is too much rain during a season, while if there is a drought there will not be enough water for fishing. The area is not easily reachable and there aren't many structures where people could spend time. Since the river is not very deep, it is unsuitable to use it with ferries or other boats. During winter there isn't much to do.
	The natural area is ideal for camping and cabins where people could live during the summer as a different type of vacation. Hunting is an option in the woods. Music festivals in the woods are a small but researched type of entertainment. Cryptocurrencies.
	All ideas given can be put in place, probably the safest bet is correlated to fishing as it is the one that requires the least amount of change to the natural area and is already in place in non competitive and non commercial way. Hunting is already practised but it involves safety hazards, especially if we consider a larger number of people being involved.

Table 1: Thinking hats table

5 Customer Journey

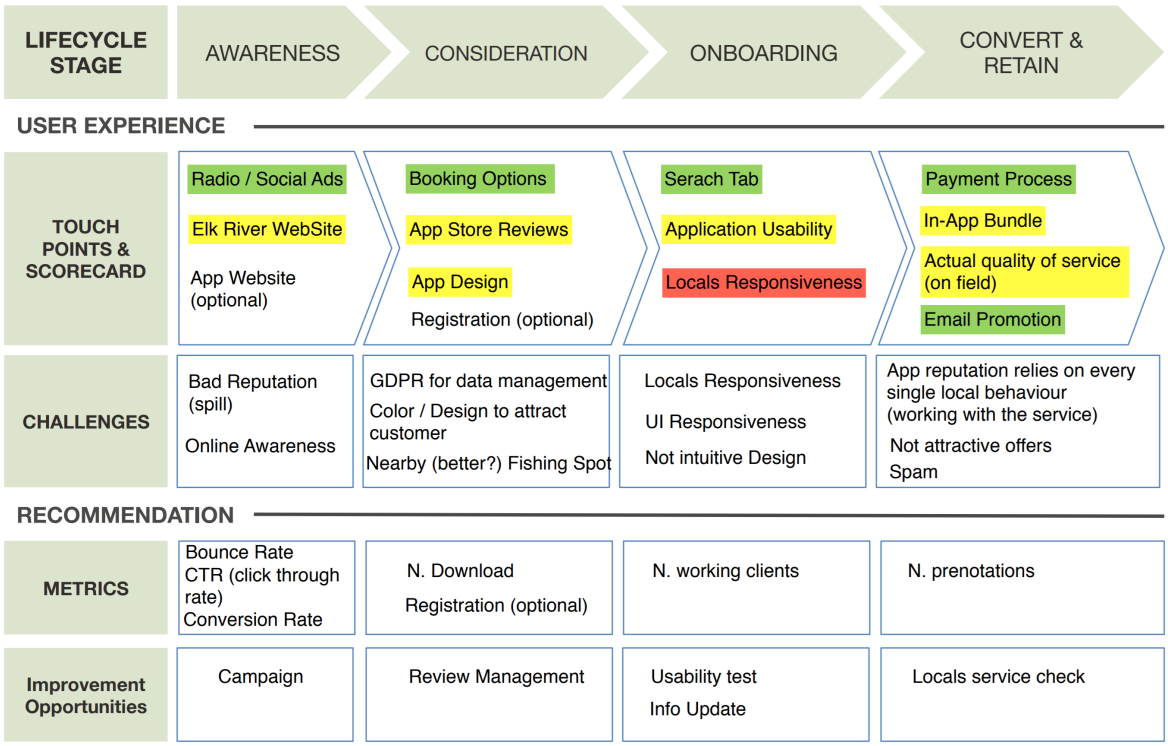


Figure 4: Customer journey map of a person who visits the area and uses services offered by locals

In the **awareness phase**, a potential customer becomes aware of the application existence. It is therefore important to find channels with biggest probability that a customer will notice the application. The easiest way to reach new customers is using social networks. Still, another channel very likely to succeed are radio stations in the area, since the majority of the people traveling to the area will probably be listening to one. The Elk River website is a viable marketing channel as long as it is regularly updated with new information, inviting existing visitors to visit the website, and new visitors to search for information there. The spill of 2014 can be considered as the major challenge when bringing new visitors to the area but it is not something controllable.

Most users with option to download an application first take into **consideration** the reviews of the application. They will read the application description, reviews and look at the application screen-shots because they want to see what the application offers exactly. This application, being an application used for booking activities and places should present them in a clear way what are their options. Special attention must be payed to fulfilling customers' needs from the beginning as this will make them spread the word. Bad reputation will surely make people avoid the application. Following good practices such as private data management is very important in today's ever growing concern for privacy.

After a user downloads the application and after verifying that the application look as expected, they will be able to see who are the locals offering services in the area. Therefore, it is important to bring as many locals as possible to use the application so that the visitors can immediately see the value in using it. On our side, we need to make sure that the platform works as expected and that it is simple to use for everyone.

After a user completes at least one booking using the application will they be able to give final verdict. Unfortunately, the quality of the service depend also on the real offer and so it is in the hands of the locals. Working with locals in the first period by giving them instruction how to improve could bring greater acquisition rate. Then, staying in touch with users by sending them promotional emails will make sure that they keep using the application.

6 Business Process Model and Notation

6.1 Normal Use

We consider normal use the main feature of the system, meaning making a reservation for a service. This is achieved by the user starting a search for the service they wish to use and if at least one result is found the list is displayed to the customer that will then select the one he/she wishes to reserve.

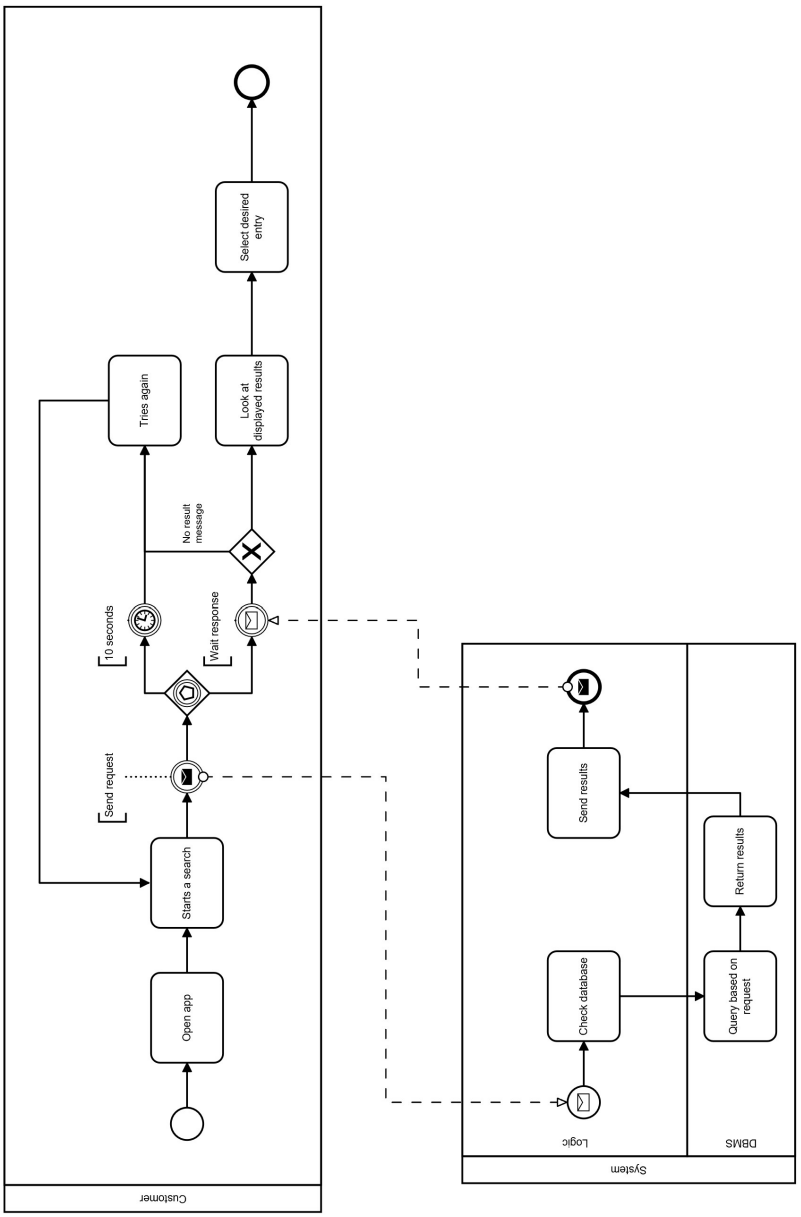


Figure 5: Ordinary application use diagram

6.2 Inserting a Review

When inserting a review a customer needs to open the application and write one, then if the back-end system accepts it, it will be stored in the database for other customers to see (N.B. in the implementation we opted for a simpler stars system, meaning that there is not an actual "writing" and unless data is corrupted or a connection is missing no method of rejecting the review is in place).

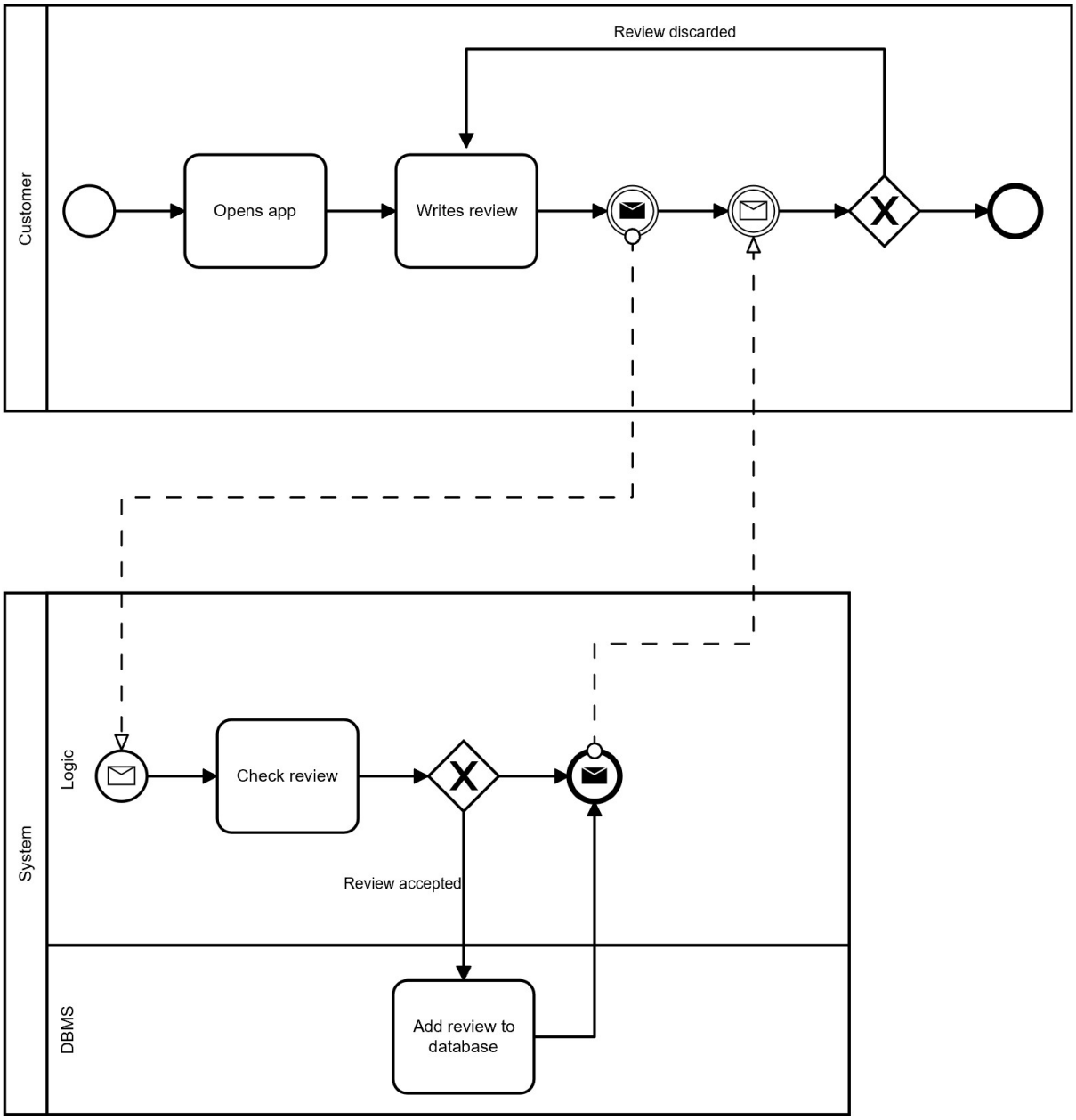


Figure 6: Review making diagram

6.3 Payment

We also considered the possibility of a payment system inside the application during development, but ultimately decided not to include it in the prototype as there would be no need for it for demonstration purposes since it would relay almost entirely on external tools/APIs (i.e. PayPal).
A brief diagram is shown below anyways, highlighting how the user needs only to send a payment request and wait for a response, much like our system itself that will only act as "bridge" between the user and the black box that it is the service by forwarding messages to one another.

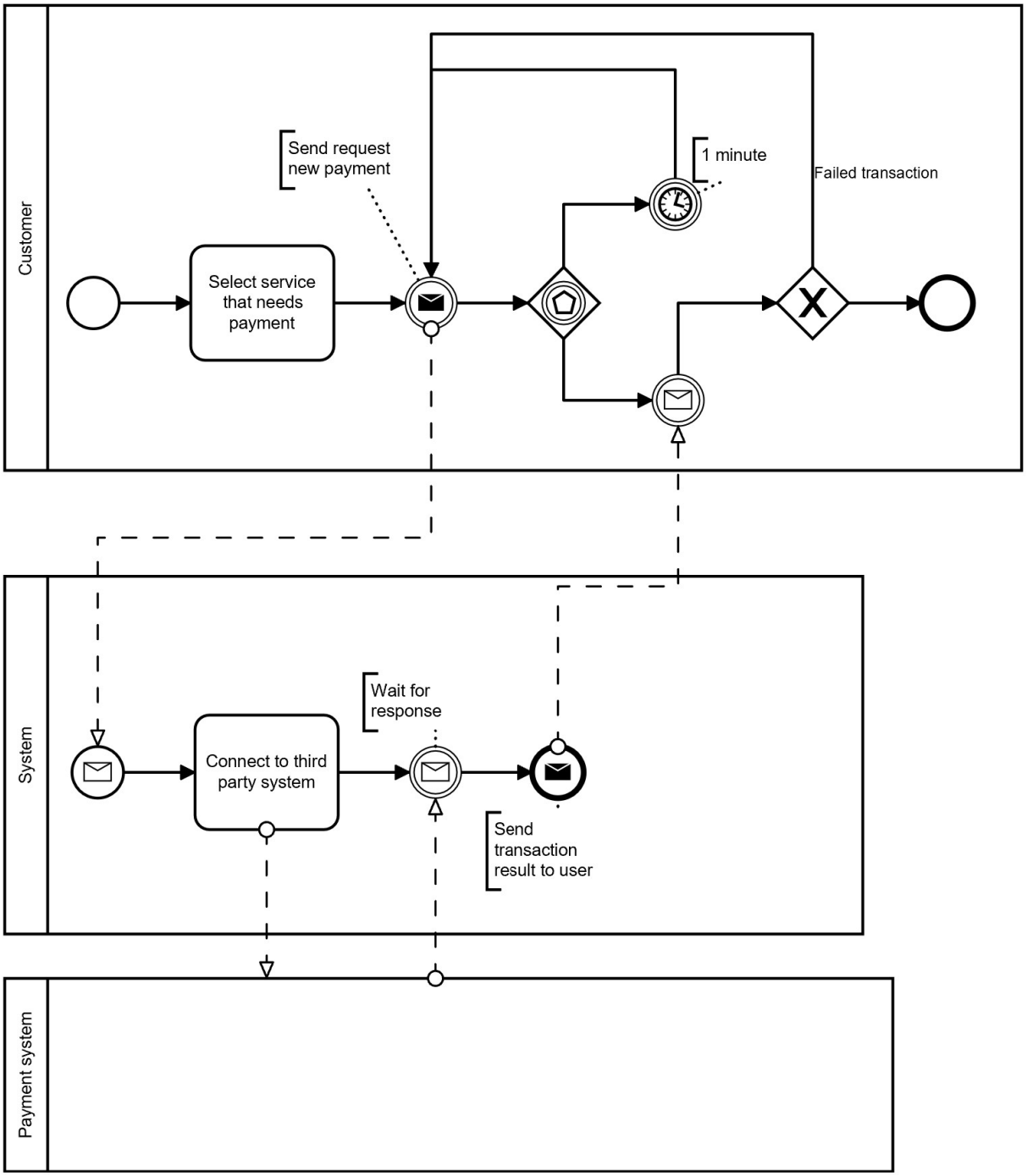


Figure 7: Payment execution diagram

7 Data Analysis

7.1 Introduction

The main goal of the analysis part has been discovering insights about what the customer did, behaved and liked. Those are key aspects to know what to change in our service and how to improve, from the customer journey to the App. To do that we decided to use two different dataset, in order to get different kinds of insights.

7.2 Clustering

The first insight we wanted to address was "What customers did during their stay? What did they liked the most? And what can we propose them the next time they will come?".
In order to do that we used a Clustering analysis and a particular dataset from the UCI repository. The dataset[1] is a (249 x 7) .csv file describing the number of reviews each user did. Every row has a specific user ID (249 users), each column reflects a different field.

- **Sports**: Number of reviews on stadiums, sports complex, etc.
- **Religious**: Number of reviews on religious institutions.
- **Nature**: Number of reviews on beach, lake, river, etc.
- **Theatre**: Number of reviews on theatres, exhibitions, etc.
- **Shopping**: Number of reviews on malls, shopping places, etc.
- **Picnic**: Number of reviews on parks, picnic spots, etc.

By the end of the first two months of opening, we assume to have available a similar review dataset about our activities.

	User Id	Sports	Religious	Nature	Theatre	Shopping	Picnic
0	User 1	2	77	79	69	68	95
1	User 2	2	62	76	76	69	68
2	User 3	2	50	97	87	50	75
3	User 4	2	68	77	95	76	61
4	User 5	2	98	54	59	95	86
5	User 6	3	52	109	93	52	76
6	User 7	3	64	85	82	73	69

Figure 8: Review Dataset

[1] https://archive.ics.uci.edu/ml/datasets/BuddyMove+Data+Set

7.2.1 KMeans

In order to perform the clustering analysis we chose to set the number of clusters before the fit. K-means algorithm allowed us to follow this assumption. We applied the Elbow method to choose k. To get info about what activities customers did the most We followed two different paths:

Nature - Picnic :

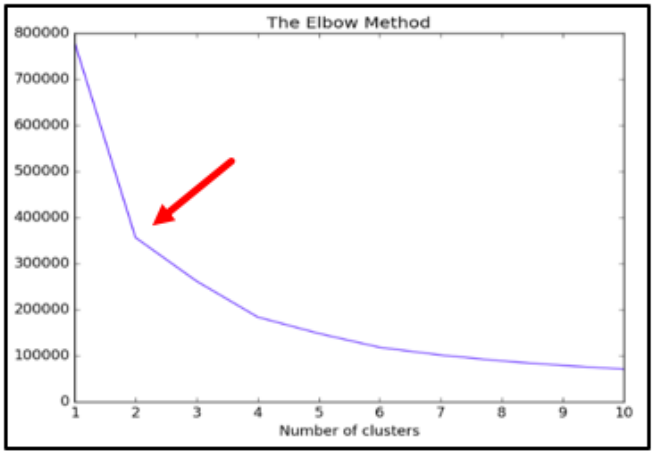


Figure 9: Nature-Picnic Elbow method

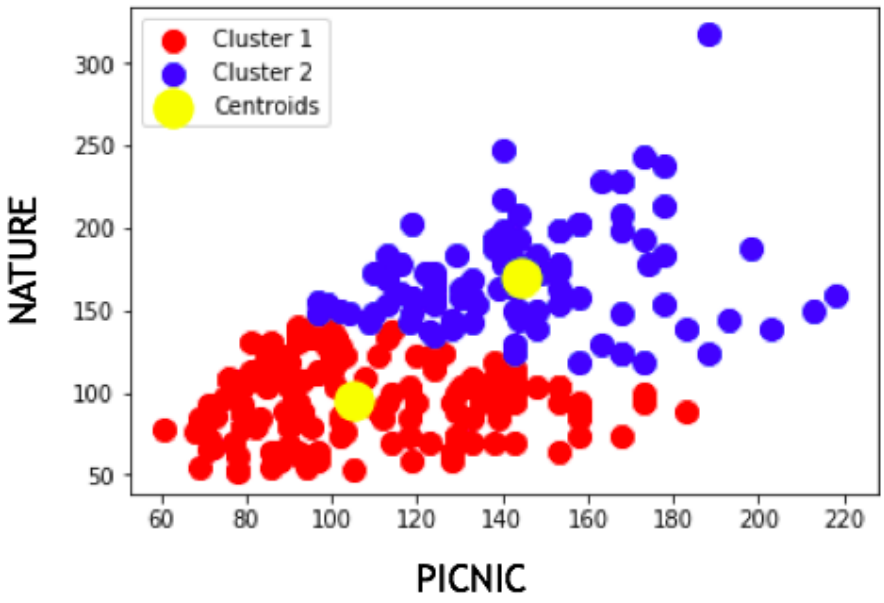


Figure 10: Nature-Picnic Clustering

From the plot is clear that people did more picnics than nature activities, as the points are totally growing to the right. Recommending picnics on the app to natural people maybe would be a waste of time.

Shopping - Nature :

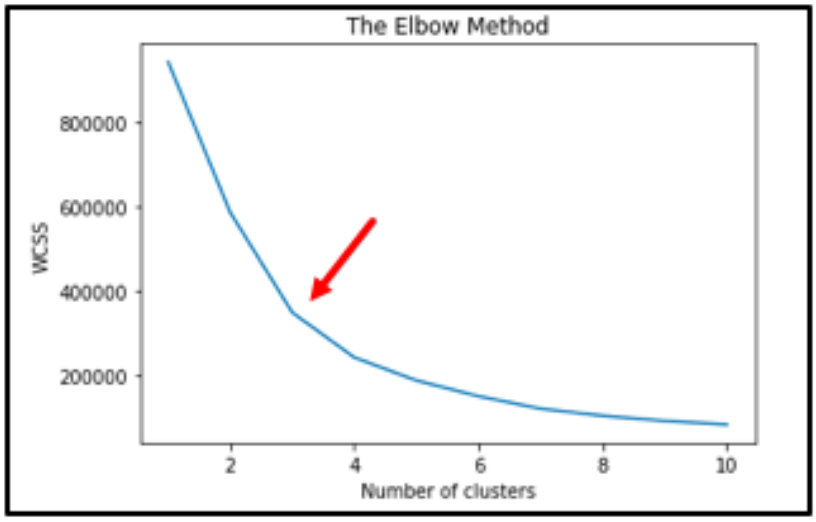


Figure 11: Shopping-Nature Elbow method

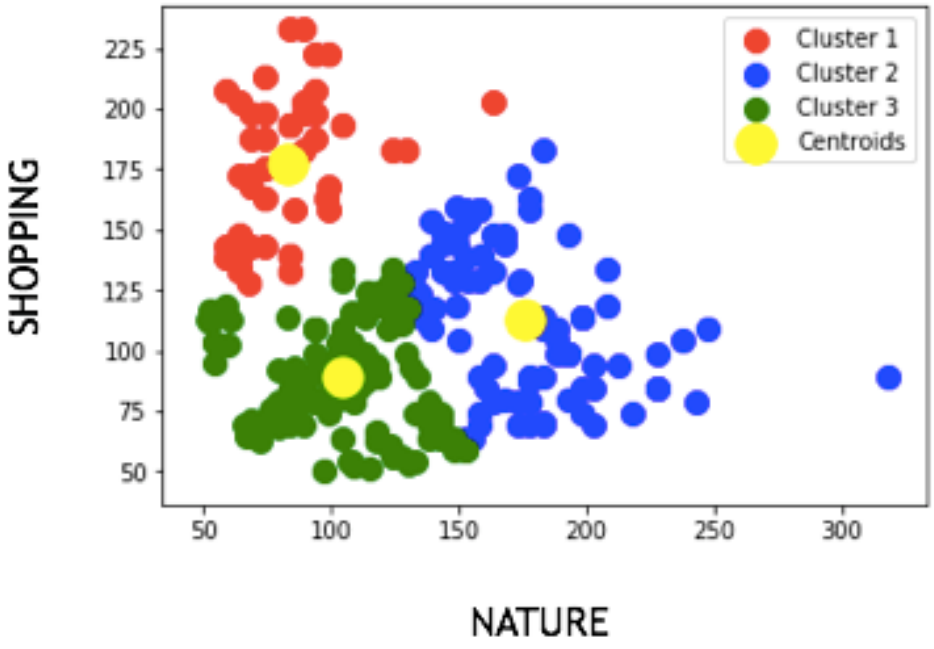


Figure 12: Shopping-Nature clustering

The plot is very different from the previous one: it is reasonable to use three clusters as one of them is clearly apart. It is not easy to understand the correlation between the variables as people seem to behave very differently. Certainly shoppers don't normally go into the wild.

7.2.2 Correlation Matrix

In the end we realized that a clustering analysis was too much to be applied to this kind of dataset and it wasn't even giving us the kind of answers we were looking for. So we opted for a correlation matrix.

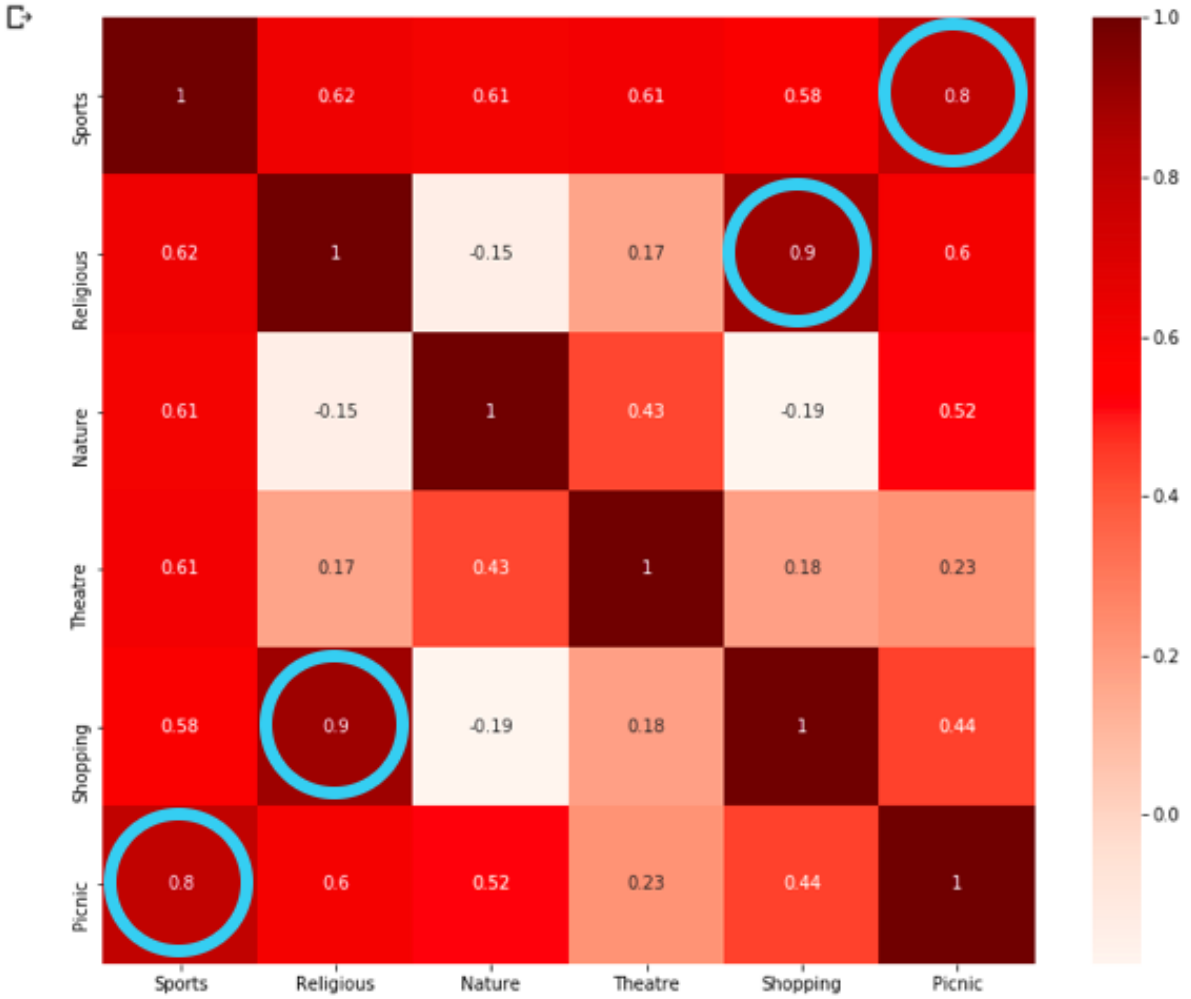


Figure 13: Reviews Correlation Matrix

This type of plot let us understand if features are actually correlated or not. As seen in the clustering analysis, we have a 0.52 between nature and picnic, while shopping and nature are absolutely not linked at all (-0.19).
It's remarkable who Religion and Shopping also seem to be correlated at 0.9 out of 1, which it's quite surprising and unexpected.

7.3 Classification

The second insight we wanted to address was "Are we able to predict if the customer will return to the watershed after his first stay?".
In order to do that we used a Classification analysis and a particular dataset from the UCI repository. The dataset[2] is a (45211 x k) .csv file describing data related to a direct marketing campaigns of a Portuguese banking institution, based on phone calls. The campaign is divided in 3 steps (calls); our goal was to predict the customer behaviour at the end of the first call, without any data from the next calls. The final dataset, indeed, will be a (45211 x 9) .csv file.

- **Age**: age.
- **Job**: type of job.
- **Marital**: marital status.
- **Education**: level of education.
- **Default**: has credit in default?
- **Balance**: balance of the year.
- **Housing**: has housing loan?
- **Loan**: has personal loan?
- **y**: has the client subscribed a term deposit?

After the first stay of a customer, we assume to have available a similar dataset comprehensive of age, job, marital, education, in order to know if that customer will return or not.

	age	job	marital	education	default	balance	housing	loan	y
0	58	management	married	tertiary	no	2143	yes	no	no
1	44	technician	single	secondary	no	29	yes	no	no
2	33	entrepreneur	married	secondary	no	2	yes	yes	no
3	47	blue-collar	married	unknown	no	1506	yes	no	no
4	33	unknown	single	unknown	no	1	no	no	no
5	35	management	married	tertiary	no	231	yes	no	no

Figure 14: Marketing Campaign Dataset

[2] https://archive.ics.uci.edu/ml/datasets/Bank+Marketing

7.3.1 Feature Engineering

Dealing with the second dataset required more data cleaning and engineering. First, we deleted all the next calls features, in order to cut all those information we couldn't use. Second we analyzed each variable one by one and decided how to encode it:

	age	job	marital	education	default	balance	housing	loan	y
0	58	management	married	tertiary	no	2143	yes	no	no
1	44	technician	single	secondary	no	29	yes	no	no
2	33	entrepreneur	married	secondary	no	2	yes	yes	no
3	47	blue-collar	married	unknown	no	1506	yes	no	no
4	33	unknown	single	unknown	no	1	no	no	no
5	35	management	married	tertiary	no	231	yes	no	no

CATEGORICAL BINARY BINARY BINARY BINARY

Figure 15: Feature Engineering Strategy

- **Education**: primary, secondary, tertiary (Label Encoded, since we want to keep track of the order. Tertiary brings higher value and score to the user).
- **Marital**: married, single, divorced (binary modified, there's no need to introduce other categories).
- **Default**: yes, no (binary).
- **Housing**: yes, no (binary).
- **Loan**: yes, no (binary).
- **y**: yes, no (binary).

```
cleanup_nums = {"marital":     {"married": 1, "single": 0, "divorced":-1},
                "education": {"primary": 1, "secondary": 2, "tertiary": 3},
                "default":     {"yes": 1, "no": 0},
                "housing":     {"yes": 1, "no": 0},
                "loan":     {"yes": 1, "no": 0},
                "y":     {"yes": 1, "no": 0}}

dataset.replace(cleanup_nums, inplace=True)
```

Figure 16: Cleanup

- **Job**: management, technician, etc (Categorical variable encoded as Dummy).

	age	job	marital	education	default	balance	housing	loan	y
0	58	management	1	3	0	2143	1	0	0
1	44	technician	0	2	0	29	1	0	0
2	33	entrepreneur	1	2	0	2	1	1	0
3	47	blue-collar	1	unknown	0	1506	1	0	0
4	33	unknown	0	unknown	0	1	0	0	0

DUMMY

default	balance	housing	job_admin.	job_blue-collar	job_entrepreneur	job_housemaid	job_management
0	2143	1	0	0	0	0	1
0	29	1	0	0	0	0	0

Figure 17: Dummy Variables

7.3.2 Correlation Matrix

Before any kind of analysis, we tried to understand feature correlation, in order to understand apriori if there were any dependencies. The matrix showed us there were no correlations at all.

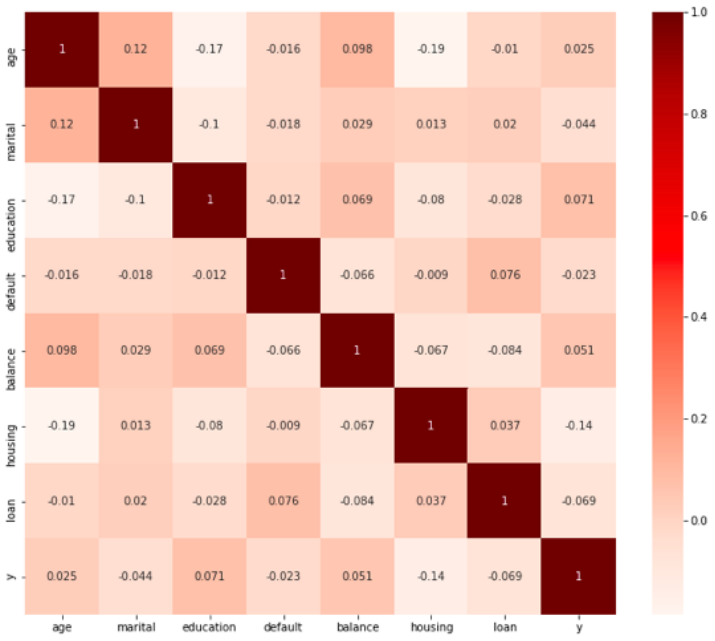


Figure 18: Marketing Campaign Correlation Matrix

7.3.3 Metrics

Since we were dealing with a classification problem, the first common solution we chose for metrics was confusion matrix, so that we could calculate accuracy and precision and use them to rate the models. During the process, however, we realized that we missed an important aspect of our dataset: the target, y, was imbalanced. This event brought us to rethink our metrics and overall every consideration we were going to make. In the end we chose RECALL and F1 as main metrics, with a particular focus on RECALL as we want to identify the completeness of the classifier and lower as much as possible all the false negatives output, in order to focus on the customers who subscribed the service.

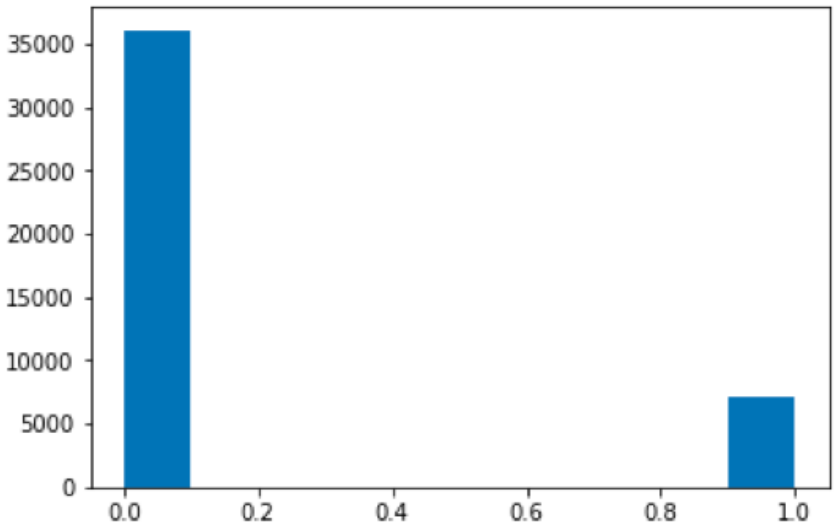


Figure 19: Target Plot

7.3.4 Models

The second best way to solve a machine learning problem in general is to try several models and compare their results, as each model is characterized by unique features that, depending on the problem, are able to make it perform better or worse. This is specifically true for imbalanced dataset. In our analysis we focused on trees as decision trees frequently perform well on imbalanced data.
They work by learning a hierarchy of if/else questions that normally can be forced to address both classes.

- **Decision Tree Classifier**: DT seemed learning quite well from the dataset and, even if the recall was not so high, averall the F1 score was very curious.

	precision	recall	f1-score	support
NOT-sub	0.87	0.86	0.87	9052
Subscribed	0.31	0.31	0.31	1747

Figure 20: Basic DT Score

- **Random Forest Classifier**: as a generalization of decision tree, RF should have been performing better. On the contrary, during the process it had a different behavior. We'll discuss more about that after more analysis.

	precision	recall	f1-score	support
NOT-sub	0.86	0.94	0.89	9052
Subscribed	0.36	0.18	0.24	1747

Figure 21: Basic RF Score

- **K-NN**: Besides others, the third best one was the K-NN. We decided to put it into the documentation to proof the fact we considered also other kind of models, even if the result was worse.

	precision	recall	f1-score	support
NOT-sub	0.85	0.95	0.90	9120
Subscribed	0.29	0.10	0.15	1679

Figure 22: Basic K-NN Score

For any doubt, in the notebook each model has also it's confusion matrix, in order to make the reader understand better the overall trend.

7.3.5 Under-Sampling

Third way to address an imbalanced dataset is to re-sample specific features. In our case we lowered the number of rows of the biggest class. Under-sampling, in fact, can be defined as removing some observations from the majority class (the customer doesn't subscribe to the service or doesn't return to the watershed). The selection was performed completely randomly. The only drawback was that we were removing information that may be valuable. For the sake of simplicity (and not to bore the reader with hundreds on analysis) we will focus only on trees.

- **Decision Tree Classifier**: like in the case before, the highest subscribed recall score has been performed by a decision tree. Remarkable is the fact that the value almost doubled (0.65).

	precision	recall	f1-score	support
NOT-sub	0.93	0.58	0.72	9577
Subscribed	0.17	0.65	0.26	1222

Figure 23: Under-Sampled DT Score

- **Random Forest Classifier**: the greatest improvement, however, was performed by RF as from a very low 0.18 it reached 0.61 (lower than DT, but still very similar).

	precision	recall	f1-score	support
NOT-sub	0.93	0.65	0.76	9577
Subscribed	0.18	0.61	0.28	1222

Figure 24: Under-Sampled RF Score

F1 scores were slightly different from the normal case, but they were still keeping the small difference they had before. Between the two models, RF F1 did benefit the most from under-sampling.

7.3.6 SMOTE

In order to reach a complete overview of our analysis, we decided to apply another re-sampling technique, opposite from the first one, based on the concept of over sampling the minority dataset. Synthetic Minority Over-sampling Technique (SMOTE), in fact, is able to generate synthetic (fake) observations based on the real ones from the minor class, in order to reach a high similar volume of samples in both classes.

- **Decision Tree Classifier**:

	precision	recall	f1-score	support
NOT-sub	0.92	0.88	0.90	9577
Subscribed	0.28	0.37	0.32	1222

Figure 25: SMOTE DT Score

- **Random Forest Classifier**:

	precision	recall	f1-score	support
NOT-sub	0.91	0.92	0.91	9577
Subscribed	0.33	0.32	0.32	1222

Figure 26: SMOTE RF Score

After the algorithm implementation, both models seemed to behave in a similar way. Precision and F1, in fact, were almost the same. RF, however, was the one which did benefit the most.

7.4 Data Analysis Conclusion

From a brute-force clustering attack we understood that a simple correlation matrix would have been more than fine to get the info we needed out of the dataset. It allowed us to understand connections we knew about (picnic - nature) and the ones we did not (shopping - religion).
The most challenging one has been the last analysis as we had to deal with an imbalanced dataset and decided to follow several paths. At first we studied the different metrics we could use to rate our results and recall was preferred. We then chose different models to use (decision tree and random forest were the best ones). At last we tried to re-sample our data and perform again each analysis.

- **Recall**

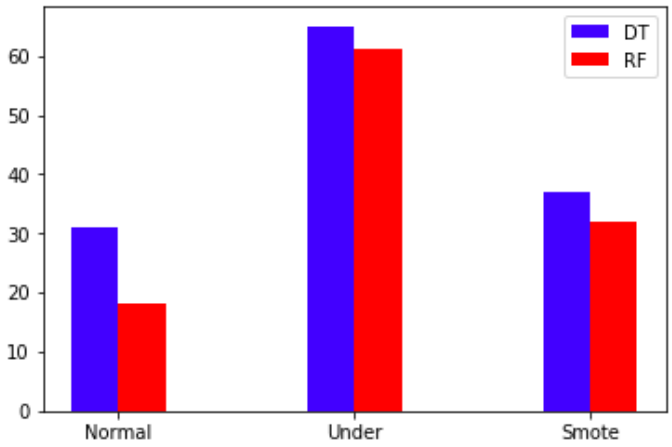


Figure 27: Recall Recap

- **F1 score**:

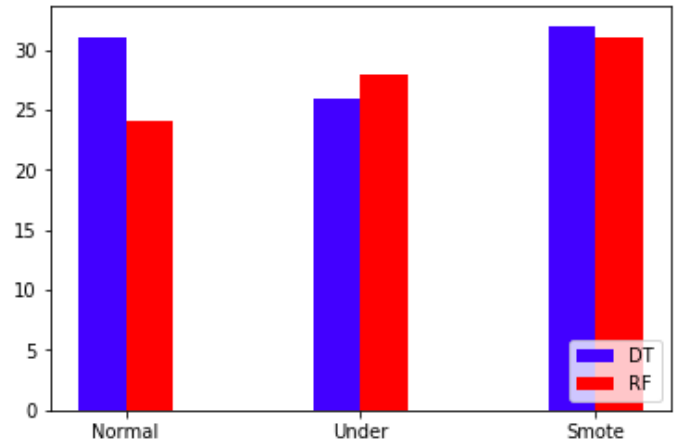


Figure 28: F1-score Recap

From the beginning we decided to use RECALL as main metric. It is clear that Decision Tree is overall the best model to address this dataset, that combined with the Under-sampling technique provided the best results.
However, considering the F1 score, it is interesting to notice how random forest model improved across the analysis and almost reached the same value as the decision tree, which, on the other hand, seems to behave totally fine even from the beginning, without any kind of sampling. This is maybe the proof that DT is effectively our best choice.

8 Prototype - Mobile Application

8.1 Structure

In this section we will describe how the prototype for the project has been realized, starting with and introduction to the technologies and services used, a description of the goals the app aims to achieve and details such as the structure of the database. Technical details of the implementation have been intentionally left out of this report given that it's not meant to be a design document or a how-to guide on how the application was built, since that would need a lengthy and appropriate document by itself. Instead, the aim of the following sections is to explain what the application does and the logic/reasoning behind the main choices that have been made during its development such as which services have been implemented and how data is handled.

8.2 Introduction

8.2.1 Operating System: Android

Given previous experience with it, we decided to build the prototype on Android (with minimum sdk 23 and compiled on skd 28), since the popular open source operative system is free to develop on by anyone without needing any particular machine or license (unlike for example iOS), and is instead sufficient to use Android Studio[3] that comes also with the proper emulator software for testing purposes without the need to connect a physical device (which is still possible to do by simply installing the USB drivers).

8.2.2 Backend: Firebase

Firebase[4] is Google's platform to develop web and mobile applications, which offers a number of services free to use on a small scale (but more than enough for our project purposes). Specifically we took advantage of:

- Authentication: handles in a standard way the registration and login of the users by allowing them to access with different methods by using a normal email address and password or connecting via popular verified systems such as Gmail, Facebook, Twitter and others, assigning them a UID to each identity.
- Database: as we will discuss more in detail in subsection 8.4 we used both database systems:
  - Realtime database: fast and reliable, worse web interface and poor query capability as it doesn't allow chaining commands.
  - Cloud Firestore: just released from beta, it has an easier to read web interface and it allows more powerful queries with chaining and indexing.
- Storage: since neither of the database systems allow to store files but only Objects (strings, numbers, maps, arrays, etc...) a storage system has been used to store the users' profile images.
- Cloud Functions: used to perform server side actions when a given trigger is fired, a more in depth description can be found in subsection 8.5.
- Cloud Messaging: used together the cloud function *SendNotification* (subsubsection 8.5.2) it was used to send users notifications for new reservations and incoming chat messages.

[3] https://developer.android.com/studio/
[4] https://firebase.google.com/

8.3 Goals

We wanted to build a single tool for the entire target group (subsection 3.1) and after some considerations we came up with the following goals for our platform:

- Allow locals to create a profile to offer fishing services with working hours.
- Allow customers to create a profile with basic information to make reservations.
- Allow local fishermen to be found by customers that need fishing lessons.
- Allow locals to be found by customers that need fishing equipment.
- Allow local expert fishermen to add fishing spots to a public list.
- Allow customers to search for fishing lessons, renting equipment and fishing spots in the Elk River area.
- Allow customers to contact the employees using a simple chat system and vice versa.
- Allow customers to leave a review for any service offered.

8.4 Database Structure

As said in subsubsection 8.2.2 Firebase was used among other things to store data.
Contrary to what it's commonly used, Firebase doesn't offer a relational database and it's instead a NoSQL JSON system, where the use of duplicate data is actually encouraged by the creators to avoid a higher number of reads in order to search for the related entry.
What follows is the representation of the online database structure of both the *Cloud Firestore* and the *Realtime* (used only for storing chats) by using some examples to fill the fields and definitions instead of some of those that are not meant to be read by humans, such as UIDs (Unique Identifiers) or URLs.

Chats section 8.4 are stored in a different way, in the Cloud Firestore database the main collection *chats* contains for each user that started at least one chat a document with that user UID, inside of this document a structure Map-like is present where the key is the *other* user with whom the chat is started and the value is an object with the necessary data, meaning that for each chat two different entries are made in the Cloud Firestore database.
In other words if Kevin Smith starts a chat with Bob Simmons there will be two structures as follows:

- chats -> Kevin Smith's UID -> Bob Simmons' UID -> data
- chats -> Bob Simmons' UID -> Kevin Smith's UID -> data

This is done because each user needs to access all their conversations when opening the chat homepage, and retrieving a single document (using their UID) is much faster than doing a query and filtering it.
The single conversations are instead stored in the Realtime database, where each one can be found by their UID created as *smallUID_bigUID* (where the comparison of the two is simply the result of the *Java String.compareTo* method) and each message is stored under that with an auto-generated UID as key while as values only the message text and the sending user name are saved.
Notifications (section 8.4) are stored only temporarily, once the Cloud Function *sendNotification* (subsubsection 8.5.2) is triggered and it sends the notification, the entry is deleted from the database.
Further comments needed are written besides each field in green colour.

Customer :

```
{
  "uid": "Firebase generated ID",
  "name": "Kevin",
  "surname": "Smith",
  "phone": "3343200266",
  "mail": "kevin.smith@gmail.com",
  "profilePicUrl": "URL to Firebase storage" File storage is a different service
}
```

Fishing Spot :

```
{
 "uid": "Firebase generated ID",
  "name": "Spot Fly Fishing",
  "nameLowercase": "spot fly fishing",
  "latitude": 38.53497378591452,
  "longitude": -81.71645309776068,
  "averageReviews": 0,
  "numReviews": 0
}
```

Reservation :

```
{
  "time": "2019-08-10 17:00:00 UTC+2",
  "type": "expert_instructor",
  "employeeUid": "UID of employee reserved", null if spot reservation
  "spotUid": "UID of spot reserved", null if employee reservation
  "customerUid": "UID of customer reserving",
  "customerName": "Kevin Smith",
  "customerPic": "URL profile pic customer",
  "employeePic": "URL profile pic employee"  null if spot reservation
}
```

Chat : From Bob Simmons' database part

```
{
  "thisName": "Bob Simmons",
  "otherName": "Kevin Smith",
  "otherUid": "Kevin Smith's UID",
  "lastText": "See you there!",
  "otherProfilePic": "Kevin Smith's profile pic URL",
  "isRead": true,
  "lastMsgDate": "2019-08-10 13:11:17 UTC+2"
}
```


Employee :

```
{
  "uid": "Firebase generated ID",
  "name": "Bob Simmons",
  "mail": "bob.simmons@gmail.com",
  "address1": "Daniel St",
  "address2": "51",
  "city": "Webster Springs WV",
  "zip": "26288",
  "phone": "332655953",
  "averageReviews": 5,
  "numReviews": 1,
  "profilePicUrl": "URL to Firebase storage", File storage is a different service
  "tags": [ ArrayList<String>
    "expert_instructor"
  ],
  "hours": [ Map<String, List<String»
    {
      "Monday": [
        "Closed",
        "Closed",
        "13:00",
        "20:00"
      ],
      "Tuesday": [
        "Closed",
        "Closed",
        "13:00",
        "20:00"
      ],
      "Wednesday": [
        "Closed",
        "Closed",
        "13:00",
        "20:00"
      ],
      "Thursday": [
        "Closed",
        "Closed",
        "13:00",
        "20:00"
      ]Cut for page length purposes
    }
  ]
}
```


Review :

```
{
  "reviewScore": 4,
  "serviceUid": "UID of employee/spot",
  "customerUid": "UID customer making the review",
  "type": "review type" spot, rental or expert_instructor
}
```

Notification :

Title and Body depend on the type (new reservation or chat message), but there is no need to store it.

```
{
  "recipientUid": "UID receiving user",
  "title": "notification title",
  "body": "notification message"
}
```

8.5 Cloud Functions

Written in JavaScript (NodeJs) When the specified trigger is fired server side actions can be executed in order to relieve the clients from doing numerous actions that would slow down their device.

8.5.1 New Review

Every time a new review is added to the database, it takes the given score and going into the employee entry it computes back the total score by multiplication between the average and the number of reviews, then it adds the new score and increments by one the total and the new average is computed and stored. If the new score is a modification of an older review, then the average is updated instead by subtracting the old score from the sum before adding the new one and the total number of reviews is not incremented.

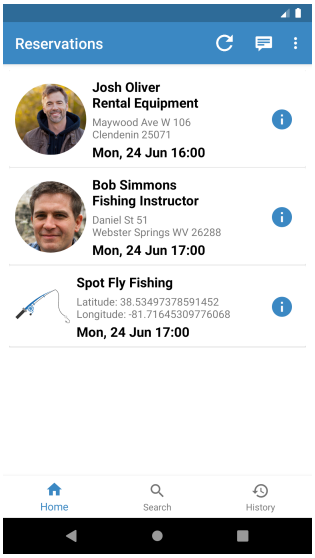
8.5.2 Send Notification

When a new notification is stored in the database it takes sends the content to all the users subscribed to the topic specified, since the topic is the UID of a user and each one is subscribed to its UID topic after the login process, only one user will receive the notification.

8.6 User Interface

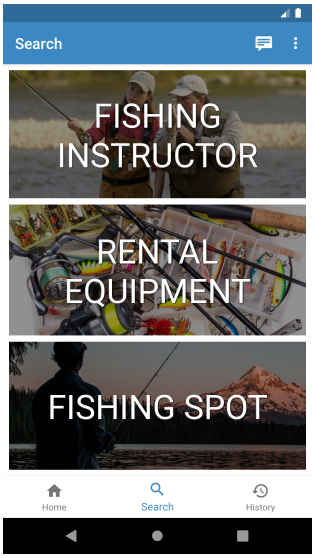
In this section the main UI screens are shown, note that the chat screen has the same interface for both kinds of users.

8.6.1 Customer UI

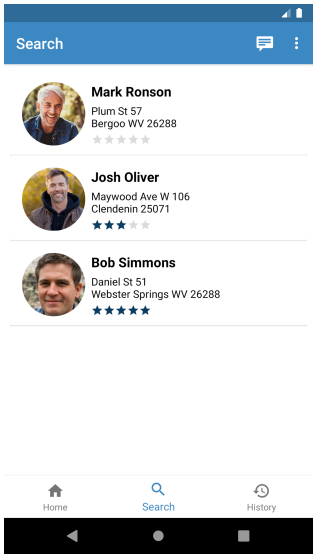


(a) Customer Homepage

(b) Chat main page



(c) Customer main search page

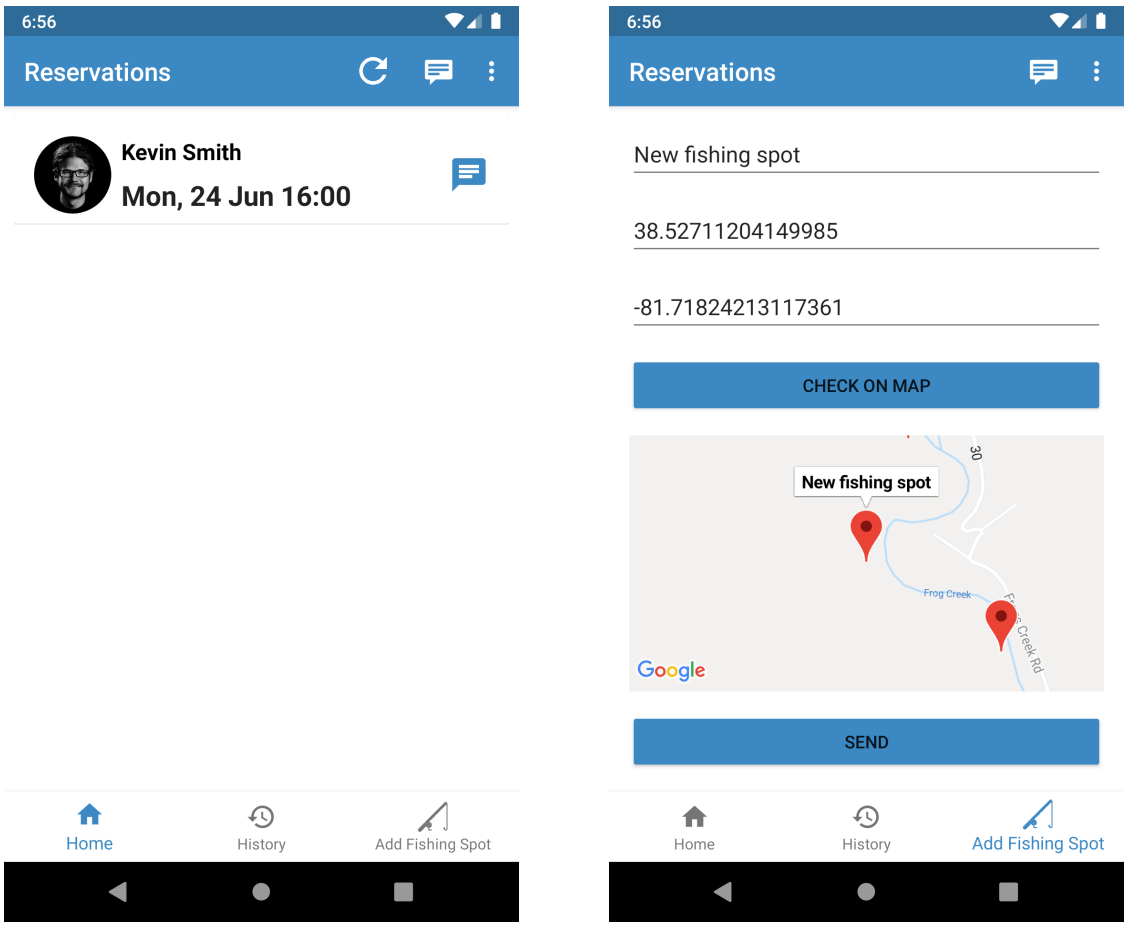


(d) Result of searching an instructor

Figure 29: Customer Homepage (a), search pages (c)(d) and chat mainpage (b)

8.6.2 Employee UI

It should be noted that in Figure 30b the employee will have the option of dragging on the map (which displays all the already registered spots) to move the center cursor which will update the latitude and longitude values in real time, or he/she will be able to manually insert the values in the text-boxes, using then the "check on map" button to put the map in the specified position (only if the values correspond to a valid coordinates, otherwise an error will be shown asking the user to insert correct values). The name of the fishing spot **must** be unique to avoid confusion to the customers and since Firesbase doesn't allow queries where a string parameters ignores upper and lower cases differences as shown in subsection 8.4 fishing spots also store the name in lowercase. If no name is inserted or it's already in use an error is displayed to the user asking for a new one.



(a) Employee Homepage (b) Adding a fishing spot

Figure 30: Employee Homepage (a) and new fishing spot insertion page (b)

9 Appendix

9.1 Software & Services Used

1. Texmaker as an editor for LaTeX.
2. Git & GitKraken
3. Android Studio
4. Firebase

