Bachelor's thesis

Information and Communications Technology

2019

Tan Nguyen

DATA-DRIVEN SERVICES DELIVERY USING FEEDBACK LOOP

TURKU AMK
TURKU UNIVERSITY OF APPLIED SCIENCES

BACHELOR'S THESIS | ABSTRACT

TURKU UNIVERSITY OF APPLIED SCIENCES

Information and Communications Technology

2019 | 30

Tan Nguyen

DATA-DRIVEN SERVICES DELIVERY USING FEEDBACK LOOP

Data is playing a crucial role in society. More and more businesses move to data-driven delivery of services with customer data as the core. With the data being analyzed, the services can be delivered better, therefore helping businesses increase their revenue and growth. As data is becoming more crucial, privacy has emerged as a serious challenge, which leads to the creation of cyber laws where rights in the cyber environment are protected and businesses need to obey them during their operation.

Following this trend, the objective of this thesis was to design a solution for appropriately collecting user data, which helps a business offer data-driven services. This thesis was commissioned by a mid-size cybersecurity company and consists of researching the solutions to meet the business requirements and the implementation of that solution.

The feature uses Mule software to integrate systems and Surveypal as the feedback service to provide feedback requests to end-users. The result will be stored as ticket fields.

KEYWORDS:

Data-driven, services, feedback loop

CONTENTS

FIGURES

PICTURES

TABLES

LIST OF ABBREVIATIONS

AFI	American Film Institute (American Film Institute, n.d.)
AI	Artificial Intelligence (Data & analytics and Artificial Intelligence, 2019)
AIS	Automated information system (Its.bldrdoc.gov, n.d.)
API	Application Programming Interface (Reddy, 2011, p.1)
ESB	Enterprise Service Bus ("What is Mule ESB?", n.d.)
GDPR	General Data Protection Regulation ("EUGDPR – Information Portal", n.d.)
HTML	Hypertext Markup Language
JDBC	Java Database Connectivity
JMS	Java Message Service
POJO	Plain Old Java Object (Fowler, n.d.)
SaaS	Software as a Service (nibusinessinfo.co.uk, n.d.)
UI	User Interface (Ross, Isgill & Stubbs, 2013, p.4)
UX	User Experience (Hassenzahl, 2006, p.91)
XML	Extensible Markup Language

1 INTRODUCTION

This thesis focuses on implementing data-driven services in a mid-size company in which data is playing a crucial role. Indeed, data is non-replaceable in this 21st century which has another name “Information age” (Nguyen, n.d.). Additionally, the importance of data analysis should be seen as a mindset instead of a tool to help companies to revolutionize sales function management in order to succeed in this new competitive environment (Dearborn, 2015). The relationship between the terms used for information and data is illustrated in Figure 1.

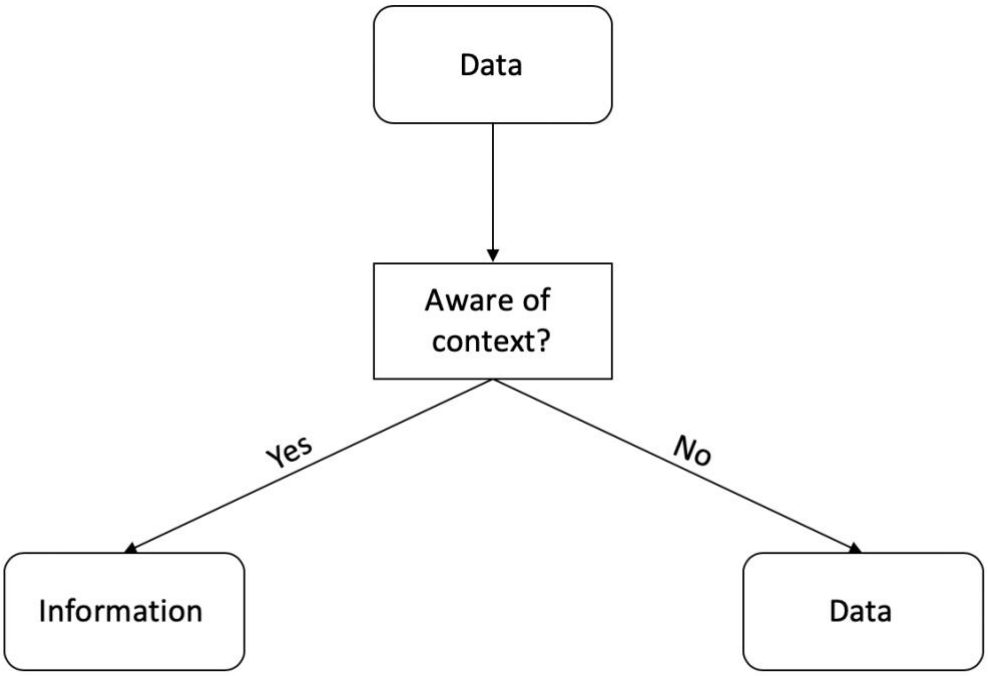


Figure 1. Relationship between data and information (Ahmed et al, 1999)

Mentioning data, according to “The Big Data-Driven Business: How to Use Big Data to Win Customers, Beat Competitors, and Boost Profits”, collecting and analyzing data has even existed since consciousness arrived to help our ancestor to survive like: hunt now, not later; eat this, not that; sleep here, not there (Glass, 2014). Until now, data is still playing a huge role in the way of doing business, which can be seen with Netflix as an example of data-driven services. To be specific, Sebastian Wernicke has pointed out in his Ted Talk about how Netflix has leveraged the data gathered from their viewers to

delivery TV shows such as the ratings that they give to the shows, the viewing histories, etc. From those information, they have analyzed deeper to figure out what type of shows or who are the producers that the viewers like, etc, to create TV series targeting the particular audience, for example, House of Cards (Wernicke, 2015) which was awarded as “TV Program of The Year” by AFI Awards in 2014 along with other 34 wins and 209 nominations ("House of Cards - IMDb", n.d.).

As can be seen above, data is not a new phenomenon/ trend here and it is all around us. It can help businesses to deliver better services and grow with successful data-driven decision-making. As we are in the information age, privacy is strongly important to anyone, which leads to the birth of cyber laws, for example, GDPR in Europe. That is why the data collection method should be also legal and follow “the rules of the game”.

This thesis is organized as follows:

- Chapter 2: the business requirements for the thesis and research of solutions to meet those requirements are studied.
- Chapter 3: the implementation will be described in the.
- Chapter 4: conclusion

2 RESEARCH

2.1 Requirements

The company is aiming to move from professional services or project scale delivery to continuous one with data-driven services on top of the platform as one of its strategies. Hence, data needs to be collected as it is playing a crucial role here, and the company needs to start collecting data now if they would like to reach their target on time.

One of the most crucial data is customer satisfaction. Indeed, as Gerson has stated in his book that all quality achievements and excellent services are meaningless without customer's satisfaction (Gerson, 1993). If the customer is not satisfied, he or she may not trust the service vendor. And especially at continuous services where the relationship between companies is important as there will be unbroken service in a period of time between two companies, this can be unhealthy for both parties as more resources are needed to make it work. As a result, a "feedback machine" will be implemented to serve the purpose.

The new system needs to be ticket-oriented, which means that it will be implemented in a way that after a ticket has been resolved, a simple feedback request will be sent to collect the satisfaction from the customers. Which type of tickets will have this feature will be different between business units.

2.2 Definitions

2.2.1 What is a feedback loop?

Life is always moving forward and getting better. For example, human civilization has been through several revolutions and their sophistication is keeping increasing all the way from agricultural to information civilization which is the current one. Those improvements are implemented based on dissatisfaction or, in other words, negative feedback. Meanwhile, positive feedback helps receivers keep going the right path. They can be from either self-reflection or other people participating in the building process (developers, testers, etc.) or the end-users.

A feedback loop is formed when the processes of receiving and handling feedback are put together repeatedly. It is the way that we reflect on the output, for example:

- What do we want to remove?
- What do we want to keep and improve?

After that, those ideas will contribute to the process of the input. As a result, there will be a better product.

For example, social media listening can be considered as a feedback loop:

1. A complaint from a customer: With a negative experience expressed through social media from a customer, the feedback loop is started
2. Collecting information: Information about the complaint will be collected by listening to the customer, which will be done by a community manager.
3. Routing the issue: Once the situation has been understood by the community manager, the issue is routed to the department where is best equipped to solve it.
4. Solving the issue: Lastly, the solutions will be delivered to the community manager by the department. Then he or she will reconnect to the customer to resolve the issue.

(Newman, 2016)

There is another example of a feedback loop in biology about the change of one variable will affect the other one. To be more specific, an increase in the amount of prey will mean more food and nutrition for predators, which will lead to a rise of predators. Once there are too many predators hunting the prey, the number of prey will decline. Hence, a certain number of predators will disappear due to a lack of food, which will make the amount of prey increase again (Albert Blog, 2016). The relationship on how the amount of prey and the number of predators affect to each other is illustrated in Figure 2.

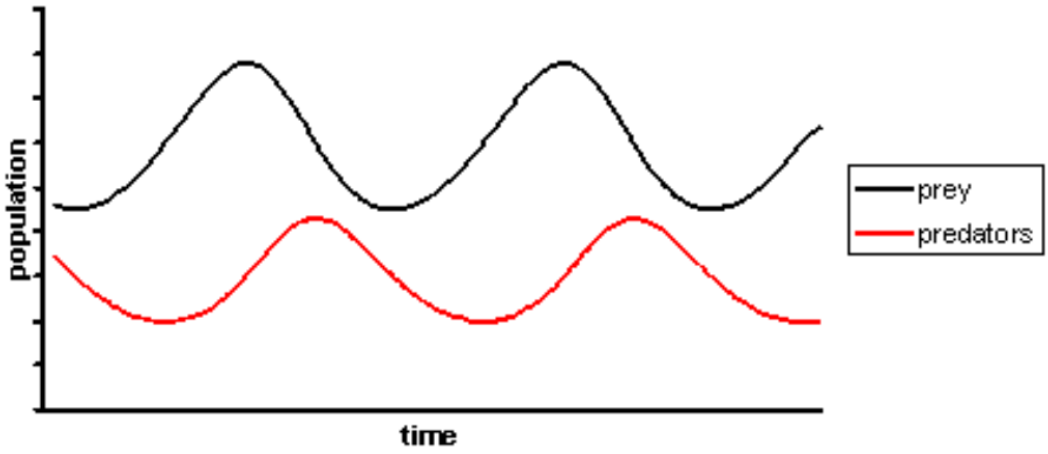


Figure 2. Relationship between the number of prey and predators (Wikimedia Commons contributors, n.d.)

2.2.2 “Feedback machine”

a) What is a “feedback machine”?

Feedback machines will work as a feedback loop but will utilize technologies in order to eliminate people’s work and boosting efficiency.

Currently, there is no official and unified way of sending feedback to customers from the company involved in this thesis. The person who is commercially responsible for the project will ask customers for feedback monthly. Consequently, the information in this thesis is about the general process while the company aims to want to have deliverable-object-oriented feedback which will focus on more features that can be delivered to the customer.

As we are going to digitalization and industrial revolution 4.0, manual repeated work should be eliminated and reduced. That is why a gathering-feedback “machine”, which will gather feedback automatically, is needed to put people’s work at ease, and also it should be easy for customers to give feedback.

b) What will the “feedback machine” do?

After the implementation, once there is a deliverable object, a notification will be sent automatically to the person listed inside that ticket/task. Then, the response feedback

will be gathered and analyzed by a specific team to improve the services to the customers.

Emails will be used mostly to notify the customers. According to the requirement, giving feedback should be easy and convenient for the customer. For example, customers can just give 3/5 stars for the object which will be displayed inside that email and the "feedback machine" will automatically update the response to the ticket/task, and there will also be a link to that ticket/task if the customers would like to give some comments or detail.

2.2.3 Software as a Service (SaaS)

One of the best of the advantages of technology nowadays is heritage. We rarely need to build anything from scratch, and most of the solutions can be found online or at least a part of it. Some of them are free and some of them are not. Especially if there is a need for commercial support, those solutions need to be purchased to be able to use. An example of free services can be Google Drive where clouds are used as online storage. On the other hand, Microsoft Office will act as a helpful tool for office work, which the user will have to purchase.

Most of the services nowadays are cloud-based and software as a service (SaaS) is the highest level in this service model, in which customers will use a complete software/application from services providers as a service. The service consumer has only to configure some application-specific parameters and manage users. The service provider handles all of the infrastructure, all of the application logic, all deployments, and everything pertaining to the delivery of the product or service (Kavis, 2014).

Everything has both its advantages and disadvantages, and SaaS is not an exception. When using SaaS, according to (nibusenessinfo.co.uk, n.d.), the businesses should consider the following advantages and disadvantages:"

Advantages:

- The first easy-to-spot advantage of using SaaS is that it requires not much up-front cost. The businesses using SaaS do not have to pay for license fees and do not have to pay for hardware and maintenance fees for software.

- Most SaaS applications are ready to use with a user-friendly interface, therefore, the company does not have to pay much time to set up and learn how to use the software.
- As the providers of SaaS will take care of the hardware and software maintenance, the upgrading process of the software is easier. By not involving in the updating of software, the company can shift this part of the responsibility to the SaaS providers.
- Most SaaS applications have a user-friendly interface and easy to connect, therefore, the company using them only needs to have a good internet connection and an internet browser to use the SaaS applications, which are accessible for most of the internet users.
- Lastly, there are “packages” that organizations can choose to fit their business. And they can be easily upgraded to a bigger one when scaling the business as more and more users access to the service.

Disadvantages:

- A high degree of control is given to cooperations from services vendor, which produces a lack of control for service users. Specifically, most of the users have to follow the latest version of the application provided by the service provider and hardly able to defer upgrades or changes in the features.
- As data will be stored in third-party storage, this raises a big concern about the privacy and security of confidential and sensitive information.
- Even though more and more SaaS is provided at the market place, not too many of them have a hosted platform option to offer to users.
- Organizations need to ensure that they have a high quality of internet service with extremely low downtime and latency to access applications as all of the services in the SaaS model are cloud-based, in other words, on the Internet.
- Since being cloud-based, this can eliminate the performance of SaaS compared to on-premise client or server applications. The cause for this is mostly about the integration time between application programming interfaces (APIs).

While SaaS will be provided by an external service provider, an automation system is usually an internal product built by a team in which most of the members will be the company employees and some of the roles can be outsourced. In this way, the company

can keep the system under control and not be dependent on externals providers. But come along with those advantages, the companies need to pay more effort and resources (e.g. recruitment, technology research, market research, etc) and take their own risk.

2.2.4 Automated Information System (AIS)

As being defined, a combination of software, firmware, and computer hardware or any combination of these will be an AIS when it is configured to handle specific information-handling operations. These operations can be exchanging information, computation, processing, dissemination or even storage of information. (En.wikipedia.org, n.d.)

In other words, it is a system containing pieces that will integrate with each other to be able to do one or a few specific tasks. And this progress should work fine without human intervention. For example, instead of having separated records in each hospital, they can be put in one place where hospitals can retrieve the information via APIs. Therefore, those records will be synced and help for treatments, for example, when patients travel to somewhere else.

As can be seen from the example above, the strongest advantage of AIS is to eliminate human work and human error. Without that system, if patients would like doctors to keep in track with their records, a print out version is needed and also needs to be carried with them while traveling, which will also increase the risk of being lost. And even after that, those records will need to be updated to the doctor who is in their hometown. Another strength of AIS is scalability. Specifically, if there are more new hospitals and they would like to join the system, they just need to be added to the system's database as long as it has enough space.

Alongside those strengths, AIS also has its weaknesses. One of the big challenges in implementing those systems is a high initial investment. Not only need to buy hardware or software's license, a team, who is for the implementation and even for the maintenance once the system has been put into production, will also need to be recruited. Another problem with AIS could be considered is centralized data. Hackers can retrieve a huge amount of data when the attacks are successful. Hence, keeping them secured is also crucial.

2.3 Ways to collect data

There are several options to deliver the solutions that varied from building from scratch to leveraging services/products that are already in the market.

“Full development” always fits any system and company, because it can be customized to meet the business’s needs. In contrast, it requires resources and time and also needs to be planned carefully.

On the other hand, using those services and products provided by a third-party company (such as Jira plugins) helps us save time and can be inserted quickly into the company’s system. But most of them are built with general-purpose for a specific group of target customers, which makes customization harder compared to the prior option.

The third option is to put those two options together. Nowadays, there are a lot of service companies providing APIs which can be used by developers of using-services companies to customize partly. In this way, the providers can also expand their target market by providing the freedom to change and adapt the products into customers’ systems.

Table 1. Pros and cons between solutions.

Solutions	*Pros*	*Cons*
Jira plugin	▪ Not much effort and resources will be spent (mostly money) ▪ Do not need to build our own system ▪ No need for a maintenance team	▪ Depending on the plugin providers ▪ The services may not be fully customized and hard to further develop
Full development	▪ Independent and we can control the whole thing ▪ Can be further developed and use at other applications ▪ We may reuse a part of the code from the current project	▪ A lot of effort and support. ▪ Need our own people to take care of it during and also after implementation ▪ Maybe more cost
Service with further development	▪ Can be further developed or customized and used at other applications (in the future) ▪ We may reuse a part of the code from the current project ▪ For example: webropol, surveypal, etc.	▪ Still depends on the provider for the service ▪ There might be a need of a team managing after release

Table 1 above highlights the advantages and disadvantages of options, which can provide a better look when making the decision. Each of the options will be analyzed in more detail below.

2.3.1 Jira plugins

Since the company is using Jira, leveraging plugins is considered a good and fast solution for this. The best thing about this is most of the processes can be taken care of by the plugin and the provider.

Meanwhile, we are dependent on them and also to find a meet-requirement-completely plugin could be challenging, especially, when the feature we need is uncommon. For example, there will be a high probability that the company will need to customize and update the feature to fit with the requirements at that time, which is difficult as we are technically dependent on them. Secondly, it is not adaptive enough. We may want to expand the feature into another platform, in which Jira plugins that are designed specifically for Jira might not be able to do the job. Additionally, the fee of using the plugins is also a major element that should be put into consideration.

After researching the Atlassian market place, three plugins that may be able to do the job are found. Table 2 will give a general look at those plugins.

Table 2. Jira plugins comparison.

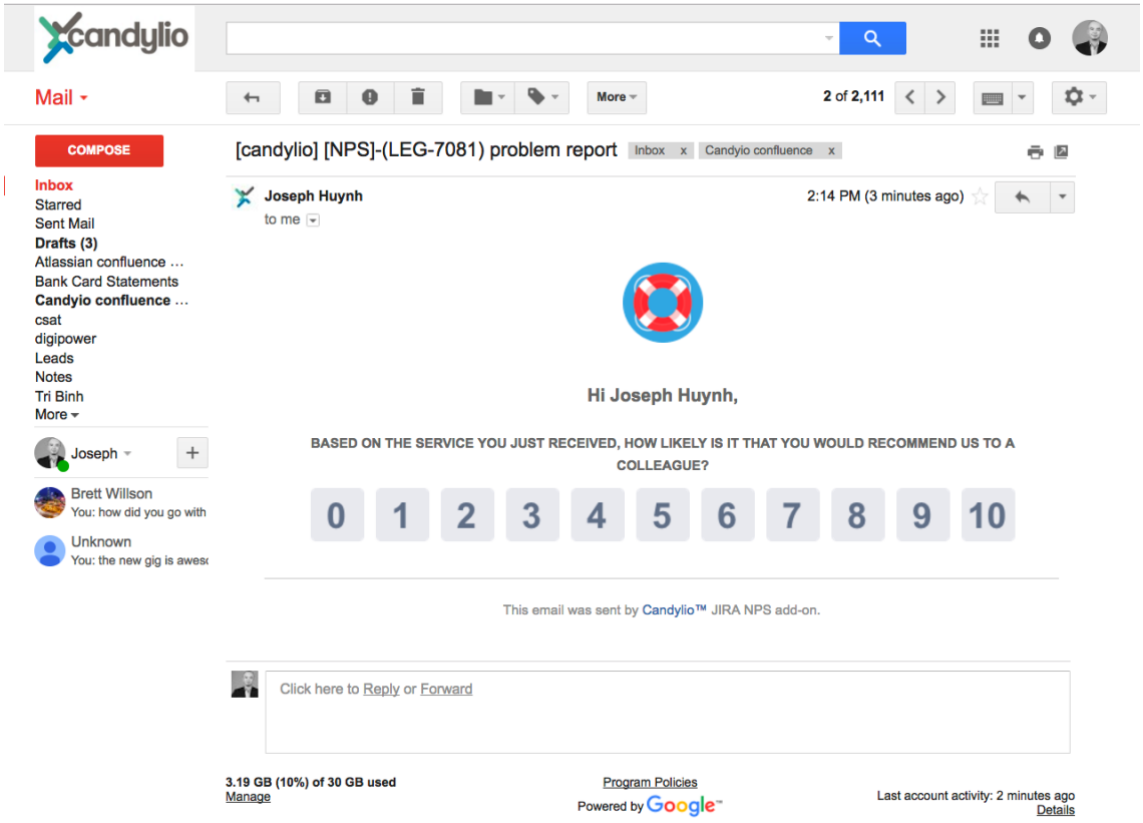
Name	*Price*	*Pros*	*Cons*
Issue feedback	1600$	▪ Low price ▪ Questionnaire is customizable	▪ Only 1 active instant (extracted at 3 pm 6.6.2019) ▪ The answer might not be attached to the ticket
Ovyka Satisfaction	4000$	▪ Questionnaire is customizable ▪ 71 active instances (extracted at 3 pm 6.6.2019)	▪ High price ▪ The answer might not be attached to the ticket
NPS survey for Jira and service desk	4500$	▪ Users may not need to login to give feedback ▪ 54 active instances (extracted at 3 pm 6.6.2019)	▪ High price ▪ Non-editable and only ask NPS questions. ▪ The answer might not be attached to the ticket

While the first two Jira plugins offer normal approaches to get the responses from the customers by sending an email notification to them and ask to click a link to open the

survey on browsers, ‘NPS survey for Jira and service desk’ has another approach and has more potential to increase UX.

To be specific, ‘NPS survey for Jira and service desk’ plugin offers the option that customers can respond directly from the email without being asked to open a link for the survey (picture 1). Those ways may increase the convenience of the end-users, hence, increase the quality of UX.

In contrast, the option cannot be checked thoroughly only from images provided by the plugin provider. Specifically, a browser may be opened when the user clicks to respond to the survey, and the browser may either just say thank you or the user may need to choose the response again on the browser, which may make them feel like they are tricked. That feeling is harmful especially in the cybersecurity field where the customers trust the company for the battle against cybercrime. Additionally, the scale may be fixed into the numbers and cannot be changed, for example, to smiley faces, which may have a better visual effect.



Picture 1. Email notification from ‘NPS Survey for Jira and Service Desk’ (Candylio Software Pte. Ltd., n.d.)

2.3.2 Full development

In contrast to using ready-made Jira plugins, we can fully develop our own system to handle the job. In this way, we are able to freely design the user experience and customize the system that matches the requirement. Also, this system's components can be reused into other projects.

In this case, we can either create our own plugin which serves our needs or an automated information system (AIS) which will automatically integrate with interfaces and handle the job.

- **Jira plugins**

This solution requires knowledge about web development technology (e.g. Javascript, HTML, Node.js, etc). Therefore, extensive time to study how the plugins are developed is necessary.

Because of all the reasons above, this option is not reasonable and will be considered as the last option in case there are no better other options.

- **Automated information system (AIS)**

Instead of building a new plugin for the solution, a whole AIS can also be built to solve the problem. This will include all the way from sending customers notification to integrating interfaces and fetching data.

Developing anything from scratch is not always an easy and wise job, which requires a lot of resources, knowledge, and energy. Instead, "standing on the shoulders of giants" by leveraging things that have been built before may be a wiser choice. That is why we come up with another option.

2.3.3 Service with further development

By leveraging the strengths of both options above, a model where an automated system integrates with a gathering-feedback service may be the best candidate. In this way, if there is a need for this feature in another system in the future, we can just simply adjust

the script to fit with that system. Some parts of the code can be even reused for the new system. For example, setting up and running a daemon or integrating with the service RESTful API.

- **Feedback services**

Several feedback services that have been already used inside the company are those best candidates and will be prioritized first before other options.

Table 3. Gathering-feedback services comparison

Service name	*Pros*	*Cons*
Lyyti form	• Finnish vendor.	• Not too many APIs calls.
Surveypal	• Many APIs calls. • Already in use inside the company.	
Webropol	• Already in use inside the company.	• Documentation for developers cannot be found.
Microsoft forms	• Can be used for free by employees.	• Does not support APIs

Available developers documentation from those services (if any):

- Lyyti form: https://lyyti.readme.io/
- Surveypal: https://developer.surveypal.com/docs/restful-api

As can be seen from Table 3, Surveypal stands out with no observed disadvantages. Additionally, the variety of APIs calls and the availability inside the company make it become the most reasonable option.

In this approach, even though the user interface design is dependent on the feedback services providers, most of them offer various tools for that. For example, Surveypal provides huge numbers of the type of questions that can be from number scale and slider to multiple choices and smiley faces (figure 3). Those tools are enough to help the

company or the designer team to easily customized the user interface design to deliver a better user experience.

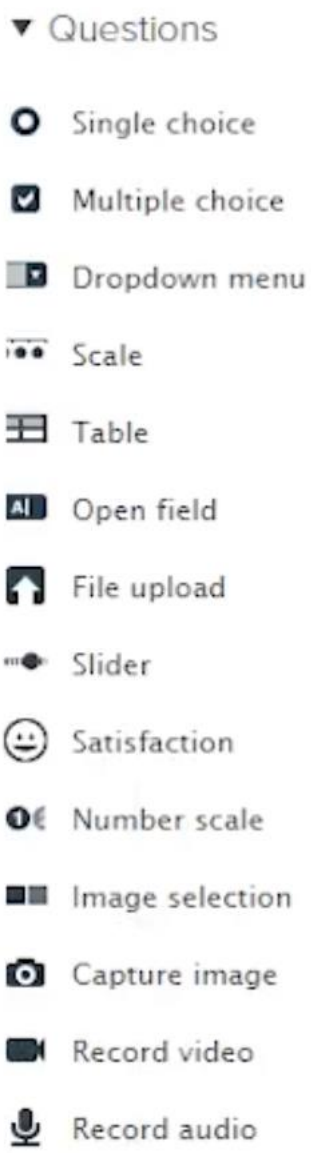


Figure 3. Available questions types from Surveypal

- **Integration**

This part will take the responsibility to interact with different platforms, for example, the ticketing system inside the company and Surveypal APIs. This one should be also able to handle logic such as choices and loop.

There is currently an integration platform inside the company used to integrate with customers' systems, which could be leveraged for this project. The product used here is provided by MuleSoft company.

Mule, the runtime engine of Anypoint Platform, allows applications can be connected easily and quickly to exchange data. It is a combination of a lightweight Java-based enterprise service bus (ESB) and integration platform. The integration between existing systems can be handled easily, regardless of the different technologies that the applications use, including JMS, JDBC, Web Services, HTTP, and more. And events can be integrated and orchestrated in real-time or in batch because of the universal connectivity of the ESB and its availability to deploy anywhere. ("What is Mule ESB?", n.d.)

As we can see above, Mule is a fit option for this role by providing integration between platforms and is also already in use inside the company.

2.4 Conclusion

Because of the flexibility and adaptability, "Service with further development" is chosen into the implementation phase, in which Surveypal will be the service gathering feedback from users and Mule will act as an integration component.

Alongside the technical strengths, this approach provides tools for letting the user interface is able to be customized. Even though this approach cannot offer a totally free environment for the design, the tools are good enough to handle the job. In the future, with the help of the flexibility of this option, another feedback service, which provides better tools for user experience design, can also be used and replace this one if the company sees there is a need for that.

3 IMPLEMENTATION

After choosing reasonable candidates for this feature in section 2, this section will reflect on how the implementation phase takes part in. Firstly, a blueprint will be created picturing how the feature should be implemented. Once it has been done, the feature will start to be implemented. After that, a review of the result of the implementation phase will follow. And lastly, more ideas about how the feature can be further developed and improved will be analyzed.

3.1 Blueprint

This section introduces how components are supposed to be implemented. While the first sector gives ideas on how components are integrated with each other, the other one will focus more on the interaction between the system and end-users. Putting them together provides not only a smooth system but also makes sure end-users will have good experience with the feature.

3.1.1 System architecture

Integration

Mule will be used to integrate between ticketing system inside the company and Surveypal APIs.

The integration part will be built with the direction that putting as much logic as we can into Java POJO classes, which can be covered by unit tests, and reducing logic used in XML. In this way, the system will be more maintainable.

Table 4. Information about components

Component	*Task*
Languages	
Mule	Integration
Java	Scripting
Fields	
Sending feedback to reporter	Verify whether feedback requests will be sent or not.
Feedback status	Status of feedback request
Feedback ID	The ID of the feedback request
Feedback result	Result of the satisfaction answer

Feedback service

The service vendor used here is Surveypal as it has been concluded in section 2. The main two reasons for this choice are that this service has enough APIs and has been already used inside the company which means no budget spent.

Figure 4 illustrates how the components will be connected and integrated. Mule will constantly pull tickets information from the ticketing system to check if there is any ticket that needs to send the feedback request. Once there is one ticket that fits the conditions, with the help of POJO Java classes implemented inside the Mule platform to analyze data and trigger the notification to the ticket reporter aka end-user via Surveypal service. At this time, the ticket will be marked as “Waiting for response”. As soon as, the customer has finished the feedback, via Surveypal APIs, Mule will gather the answer and connect it to a ticket. The ticket will be ignored if no response is received from the customer after 30 days.

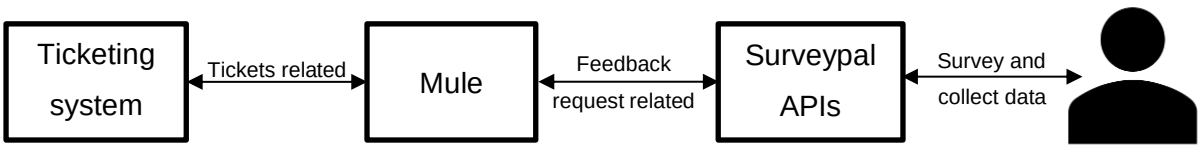


Figure 4. Integration between components

3.1.2 User interface & User experience

Being simple and convenient are the two keys that have been followed to create the user interface, as it should not consume too much time of users. In the article in 2006, Hassenzahl explains further user experience (UX) by dividing it into three different facets to comprehend the interaction between users and technology (figure 5). These three facets are the emotion and affect facet, the experiential facet, and beyond the instrumental. Three facets overlap each other, which means they share some ideas with each other (Hassenzahl, 2006, p.95).

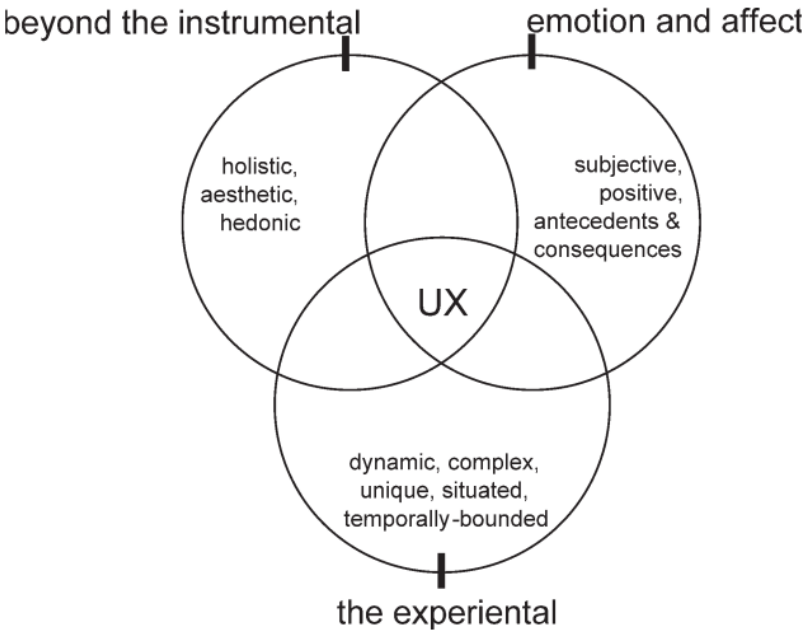


Figure 5. Facets of UX (Hassenzahl, 2006, p.95)

There are two parts interacting with the end-users which are the notification and the survey user interface. They will directly affect the UX about the service, therefore, they will be designed carefully. The designed notification is described below.

- **Notification for customer**

Title:

Hey {user}, how's your experience with {Company_name}?

Body:

Dear {user},

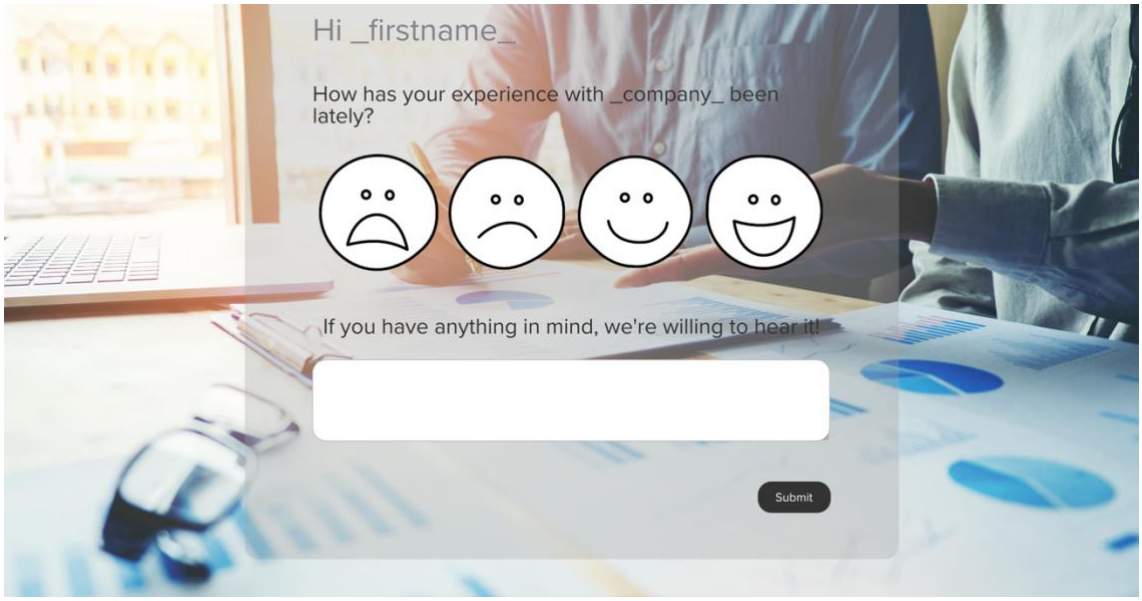
We have resolved your ticket {ticket_key} ({ticket_name}) and would like to hear your feedback on it.

You can give your feedback in here: {feedback_link}

Regards,

{Company_name}

- **Survey**



Picture 2. A part of the user interface in the survey.

Picture 2 shows a part of the user interface in the survey. There are two questions asking about user satisfaction and comment. The number of symbols here is intentionally even as it will not allow the user to pick the middle one, which is mostly pointless in helping the company understand the users. To be specific, the dichotomous option will be suboptimal for people with a neutral attitude as the point, which is most obviously needed to permit accurate mapping, is not available (Krosnick & Presser, 2010).

Furthermore, Jeff Sauro has pointed in his article about the left-side bias meaning the left side of the response scale tends to be more favored from the responders, which may affect the accuracy of the survey (Sauro, 2010). Therefore, customers can be more understood by using the approach with the lowest option on the left side of the survey response because it is riskier to do the other way as some small issues may be unnoticed until they become big enough and more resources will be needed to solve them at this time.

No matter how good of the user experience has been created or how user-friendly of the user interface, there will be always people who are not interested in doing the survey and would like to stop receiving such invites. Besides, according to privacy legislation in the EU and US, all users also have the right to have the possibility to decide if they would like to receive such emails in the future. Therefore, the unsubscribe option should be available to end-users and this system will also give that possibility to the end-users. (surveypal.com, n.d.)

3.2 Implementation

In this section, the timeline of each phase during the implementation will be introduced. Further, there will be a technical information sector containing more detail information about the system.

3.2.1 Timeline

As being new to Mule, the whole of May of 2019 was used to familiarize with Mule. Documentations, code, and logic in previous projects in the company were taken into account and analyzed in order to achieve the goal.

The implementation phase began after the initial study, from where it will be reviewed by the system architect and he/she may suggest some feedbacks. After the feedback was addressed, the testing-and-fixing phase started. The plan is visualized in figure 6 below.

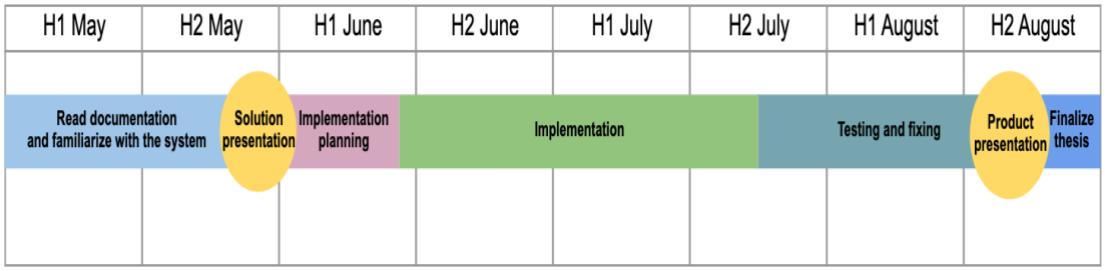


Figure 6. Implementation plan

3.2.2 Technical information

In this section, more detail and technical information about processing tickets will be introduced. The specification of the system where triggering action, input and rules are included. Following the specification, the information about the workflow and how tickets will be polled inside the workflow is described.

Specification

The filter to choose what type of tickets can send feedback request is varied and dependent on the business unit using this feature, and it will be called here as conditions. And after each ticket has been processed, the information of change (if there is any) will be updated into the ticket's fields, which allows retrieval of information easily.

Triggering action / Input

- Once there is a ticket resolved and it meets the conditions to send feedback request, the ticket will be polled from the feedback machine and start to be processed.
- Format: JSON

Rules:

- The field "Sending feedback to customer" should be marked as "Yes".
- All tickets that meet the conditions.

Workflow of processing

- Described in Figures 7 and 8

Mule workflow

Figure 7 shows how the workflow will be in the system. As can be seen, there will have three main processes. Firstly, the system will send a feedback request to those tickets that meet the requirement. Once it has been done, another process will be run to update tickets if its feedback request has been finished. And if no response has been received from the end-user after a period of time, the last one will expire the feedback request, therefore, it will not be found by the system later.

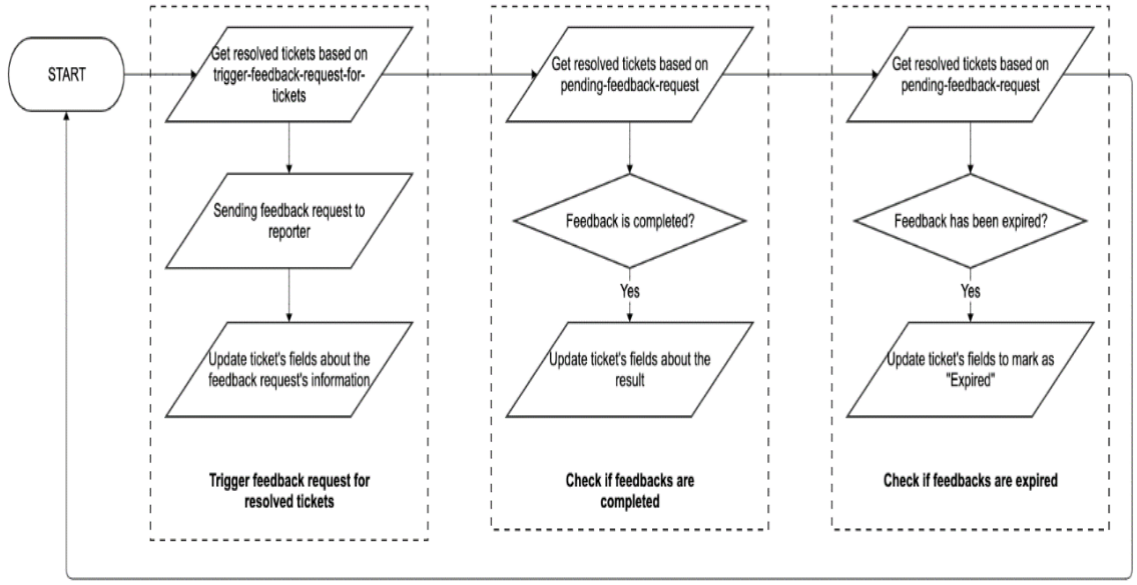


Figure 7. Feedback machine workflow

Polling tickets

The polling will run periodically every 60 seconds for three processes, each process will have its own filter to choose the right tickets. In order to reduce the huge payload between APIs and increase the system's performance, pagination will be used in polling tickets. The API call to poll tickets from the ticketing system, which allows getting at most 1000 tickets per API call, now will get only 50 tickets each API call. After those tickets have been processed, another round of 50 tickets will be polled. Figure 8 clarifies how the pagination process will be.

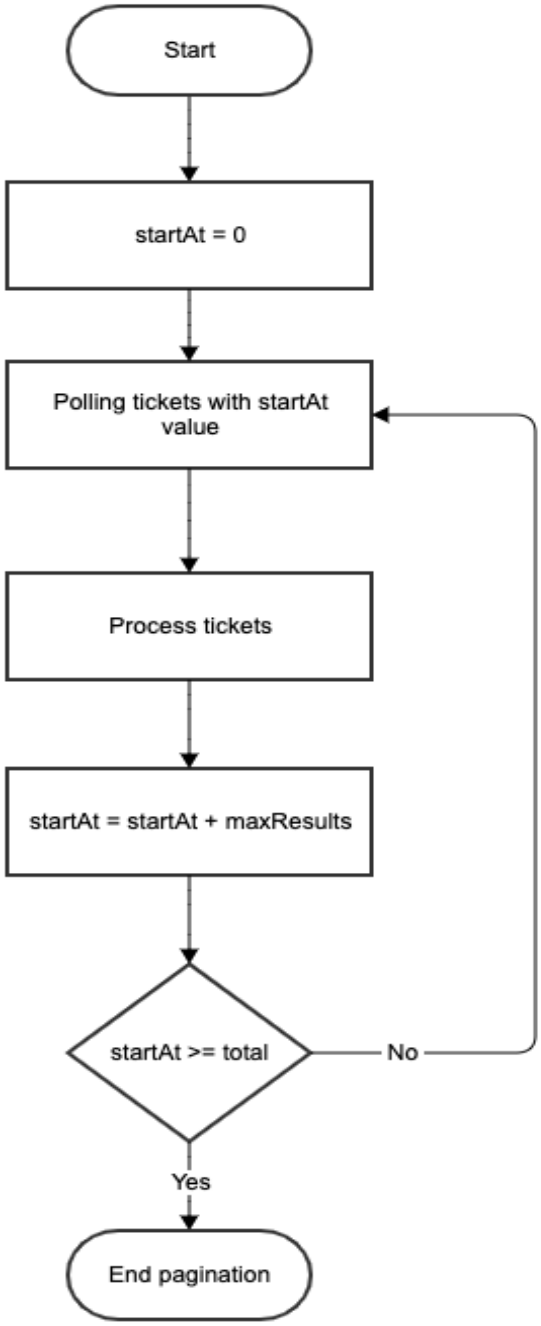


Figure 8. Polling logic

As can be seen from figure 8, startAt value will be set as zero when doing the first poll. After processing those tickets from the poll, a new startAt value will be increased by maxResults – the highest number of tickets per each API call. If the new startAt value is higher than total value – the total number of tickets that need to be processed in a specific

process, that means all tickets have been polled and the pagination will end. Otherwise, another API call with the new startAt value will be executed and the loop continued.

3.3 Result

The system has been piloted internally within the team. The system has successfully triggered the notification to the reporter asking for feedback. And after the feedback has been completed, the answer is updated into the ticket. Because of the limitation of time, the ticket will be ignored if the system has not received the answer for more than one day instead of 30 days but it will be 30 days when it is run in production.

As can be seen from figure 3, feedback service has been implemented separately from the integration platform. This way of implementation will help developers easier to maintain or even change to another feedback service, for example, to improve the user experience. Furthermore, the integration platform can also be integrated into other APIs to build another feature in the future.

Below are the specific figures about the result of the implementation:

- A main Mule file to handle integration.
- Two POJO java classes to handle the logic in parsing.
- 13 unit test cases.

Among POJO java classes, there is one main class to process the logic and the other one is a helper class. All public methods in both classes will be covered in unit tests in order to reduce the probability of breakdown. Those unit test cases are selected with the main aim being to make sure the pattern of data and the data itself are correct during the flow.

Collected data will be stored in the custom fields in the ticketing system, which helps them being organized as the system is meant to be designed in a ticket-oriented way. Besides, the system's capacity will be based on the capacity of the company's ticketing system. As a result, no separate database is needed to be taken care of and if more capacity is required in the future, solely updating the ticketing system is enough.

Regarding user interface and user experience, no customer has seen and assessed the design of the survey except people within the business entity. The design has been

developed solely by the author, which uses resources inside the company and free ones that are available for commercial use. Specifically, the smiley faces in the design are fetched from pixabay.com which has stated that images or videos may be used freely or even commercially if they are released under Pixabay License (Pixabay.com, n.d.).

3.4 Further development

The two important parts of data-driven services are inevitably data itself and data analysis. Specifically, with a lack of data, the services will not be data-driven and even when the company has already had data, it cannot help to improve the business if the data is analyzed poorly.

In order to ensure that we have data, we need to have a high enough rate of feedback response. The expected rate should be 20-30% as stated by a business unit leader in the company. Therefore, the user experience should be a crucial part taken into account here.

There is an option to improve the UX is that to attach the satisfaction question inside the email, therefore, the end-user will not need to open a browser to fill in his or her answer but just simply click on the answer that they would like to respond. Then, a browser will pop up but just only for confirmation that the answer has been sent. This option can be done when the feedback service providers offer more APIs for this feature, otherwise, changing to another service provider can be put into consideration or this upgrade may be developed fully inside the company.

Before going to having enough data, they can also be utilized earlier when the company starts to collect them. To be specific, the collected data will reflect how happy the customer is and can be fetched near real-time in the system without waiting. Therefore, the problems can be identified quickly and prevent them to cause further damage. Also, this data can provide a more general view of how all services are being delivered, from which actions can be decided based on the situation and the company's strategies. For example, more attention can be put to nurture the below-average ones to help them deliver better or the company can know which areas are their strengths to have a more effective commercial campaign focusing on them.

Once enough data has been collected, we need to step up in how to leverage it, in other words, data analysis. As being stated in his book, Cuesta has defined it as the process

in which raw data is ordered and organized, to be used in methods that help to explain the past and predict the future (Cuesta, 2013, p.7). In that way, it can be utilized in making decisions more scientific and helping businesses operate more effectively. Currently, data is being analyzed inside the company by using tools such as Excel, Power BI, etc. These methods can produce high-quality results but require human work. And in the industry 4.0, more and more human interfere should be eliminated and systems tend to be more automated, as human error can be minimized by the help of added benefits from automation (The Impact of Automation on IT Operations, 2017).

A good solution for data analysis can be utilizing artificial intelligence (AI). AI can be seen everywhere at this moment, from a talk inside cooperations to conventions. Even KPMG - one of the Big Four accounting organizations – has put moving to AI under their radar with the reasons:

- The first reason pointed out by KPMG is the fast pace of technology development. In the last decade, the speed of technology development increase significantly. This fast growth of technology has brought in a new type of risk – digital risks and the companies, who cannot adapt fast enough, have to deal with the potential of failure.
- With the help of data and analytics, organizations can identify, predict and mitigate risks from complex and demanding business environments because of the increase of regulatory scrutiny.
- All types of companies have a huge opportunity that market data has created.
- Lastly, efficiency and productivity will be boosted with the help of data and analytics.

(Data & analytics and Artificial Intelligence, 2019)

Hence, it is inevitable that all organizations in the future could encounter some sort of AI during the operation. And companies, which could not catch up with the trend, will be left behind. Currently, the company is also stepping into this field, which can be leveraged into the feedback machine.

4 CONCLUSION

In this thesis, several approaches have been suggested and analyzed to meet the business requirements for a feedback machine. These approaches are varied from building everything which provides total control to leveraging SaaS which offers agility. A hybrid solution by combining both approaches was chosen for implementation, whose features can be utilized to meet the requirements and serve the business. By keeping it hybrid, the integration part will be implemented inside the company to connect the ticketing system and the feedback service. The integration platform can also be used by another feature to integrate with the ticketing system inside the company, which can also offer a faster speed for later features. Additionally, the feedback service can also be changed without causing too much configuration. While the technical approach will help to scale and maintain the system more easily, a user-friendly user interface (UI) will help to keep the users using the feature. That is why the UI has been designed in a user-friendly manner where there are only two questions asking about the satisfaction score and if the user has any comment, and the satisfaction score is also visualized with smiley faces instead of numbers.

The work presented in this thesis will support one of the main strategies of the company which is moving into data-driven services, satisfaction scores will be collected automatically from customers using a ticketing system. Then with the help of analysis, the services provided by the company will be delivered better, and therefore, resulting in more satisfied customers.

In the process of writing this thesis, the writer has learned the hard way that the software version used in the system should have been paid more attention. Specifically, during the implementation phase, an incorrect version of Mule has been used to develop the feature at the beginning. Therefore, a lot of time was spent because of this mistake to make the code work with the correct one as the migrating process is not supported much by Mulesoft. As a result, the whole code needed to be migrated into a new blank project to fit the classpath and make it functional.

REFERENCES

Ahmed S, Blessing L, and Wallace K (1999) The relationships between data, information and knowledge based on a preliminary study of engineering designers. ASME Design Theory and Methodology Conference, Las Vegas, Nevada.

Albert Blog. (2016). Positive and Negative Feedback Loops in Biology. [online] Available at: https://www.albert.io/blog/positive-negative-feedback-loops-biology/ [Accessed 20 Aug. 2019].

American Film Institute. (n.d.). About AFI. [online] Available at: https://www.afi.com/about-afi/ [Accessed 12 Sep. 2019].

Candylio Software Pte. Ltd. (n.d.). Boost Customer Satisfaction. [image] Available at: https://marketplace.atlassian.com/apps/1217893/nps-survey-for-jira-and-service-desk?hosting=server&tab=overview [Accessed 9 Dec. 2019].

Candylio Software Pte. Ltd. (n.d.). Survey email with ticket reference. [image] Available at: https://marketplace.atlassian.com/apps/1217893/nps-survey-for-jira-and-service-desk?hosting=server&tab=overview [Accessed 9 Dec. 2019].

Cuesta, H. (2013). Practical data analysis. Birmingham: Packt Publishing.

Dearborn, J. (2015). Data Driven (p. xi). Hoboken: Wiley.

Drisgill, R., Ross, J. & Stubbs, P. (2013). Sharepoint 2013: Branding and user interface design. Indianapolis, Ind.: John Wiley & Sons, Inc.

EUGDPR – Information Portal. Retrieved 12 September 2019, from https://eugdpr.org/

Fowler, M. (n.d.). bliki: POJO. [online] martinfowler.com. Available at: https://www.martinfowler.com/bliki/POJO.html [Accessed 13 Dec. 2019].

Freeform Dynamics. (2017). *The Impact of Automation on IT Operations* [Ebook] (p. 3). Retrieved from https://www.fujitsu.com/si/Images/user_survey_report_The_Impact_of_Automation_on_IT_Operations_ww_en.pdf

Gerson, R. (1993). *Measuring Customer Satisfaction* (p. 5). Course Technology Crisp.

Glass, R. (2014). *The Big Data-Driven Business: How to Use Big Data to Win Customers, Beat Competitors, and Boost Profits* (p. 3). Wiley 2014.

Hassenzahl, M. (2006). User experience - a research agenda. *Behaviour & Information Technology, 25*(2), pp. 91-97. doi:10.1080/01449290500330331

House of Cards - IMDb. Retrieved 8 September 2019, from https://www.imdb.com/title/tt1856010/awards

Its.bldrdoc.gov. (n.d.). Definition: automated information system. [online] Available at: https://www.its.bldrdoc.gov/fs-1037/dir-003/_0436.htm [Accessed 13 Dec. 2019].

Kavis, M. J. (2014). Architecting the cloud : Design decisions for cloud computing service models (SaaS, PaaS, and IaaS). Somerset: John Wiley & Sons, Incorporated. 17 - 18

KPMG. (2019). *Data & analytics and Artificial Intelligence* [Ebook] (p. 1). Retrieved from https://home.kpmg/content/dam/kpmg/ch/pdf/brq-data-analytics-en.pdf

Krosnick, Jon & Presser, Stanley & Building, Art-Sociology. (2009). Question and Questionnaire Design. Handbook of Survey Research.

MuleSoft. (2019). What is Mule ESB? | MuleSoft. [online] Available at: https://www.mulesoft.com/resources/esb/what-mule-esb [Accessed 13 Aug. 2019].

Newman, D. (2016). How well does your organization use feedback loops? Retrieved from https://www.forbes.com/sites/danielnewman/2016/08/02/how-well-does-your-organization-use-feedback-loops/#8fdba13594b1

Nguyen, Tuan C. (n.d.). Who Invented the Computer?. [online] ThoughtCo. Available at: https://www.thoughtco.com/history-of-computers-4082769 [Accessed 13 Dec. 2019].

nibusinessinfo.co.uk. (n.d.). Advantages and disadvantages of Software as a Service (SaaS). [online] Available at: https://www.nibusinessinfo.co.uk/content/advantages-and-disadvantages-software-service-saas [Accessed 2 Sep. 2019].

Pixabay.com (n.d.). Pixabay License - What is allowed and what is not?. [online] Pixabay.com. Available at: https://pixabay.com/blog/posts/pixabay-license-what-is-allowed-and-what-is-not--4/#:~:targetText=Pixabay%20images%20and%20videos%20may,out%20more%20in%20this%20article.&targetText=Images%20and%20videos%20are%20released,is%20appreciated%2C%20but%20not%20required. [Accessed 9 Dec. 2019].

Reddy, M. (2011). API design for C++. Boston: Elsevier/Morgan Kaufmann.

Sauro, J. (2010). MeasuringU: Survey Respondents Prefer the Left Side of a Rating Scale. [online] Measuringu.com. Available at: https://measuringu.com/left-side-bias/ [Accessed 9 Dec. 2019].

surveypal.com (n.d.). Sender profiles. [online] Surveypal. Available at: https://support.surveypal.com/hc/en-us/articles/360003537494-Sender-profiles [Accessed 9 Dec. 2019].

Wernicke, S. (2015). How to use data to make a hit TV show. Retrieved 8 September 2019, from https://www.ted.com/talks/sebastian_wernicke_how_to_use_data_to_make_a_hit_tv_show?language=en

Wikimedia Commons contributors (n.d.). File:Volterra lotka dynamics.PNG. [image] Available at: https://commons.wikimedia.org/w/index.php?title=File:Volterra_lotka_dynamics.PNG&oldid=234647318 [Accessed 2 Sep. 2019].