**Enterprise Systems Development**

**SWE 6002**

Student: Brinza Augustin Constantin

Professor: Prokopakis Georgios

GitHub: <https://github.com/augustinnn/SWE-6002-Enterprise-Systems-Development>

**Table of content**

1. Introduction

- Assignment requires

- Customer Relationship Management (CRM) definition

- Spring boot and MySQL benefits

**2.** Definition of the case study, analysis, and design

-Case study definition

-Analysis overview

-Design overview

**3.** Implementation

-Framework and Technologies

-Requirements and business goals

-Security implementation

4. Evaluation

-Role of the framework

-Design Patterns

-User experience

5. Conclusion

-Personal opinion

-Key of success

-Challenge mitigation

7.References.

**Introduction**

Assignment requires us to design and implement an enterprise solution for a CRM or customer relationship management which can be defined as a combination between a simple front view, for employee usage where they can perform different tasks such as: add data, perform multiple types of contacting, classification of data, remove data, and a complex data base that back the date used. Is mostly used in sales and other places that perform task on data base such as a medical records or application used for electronical voting. As editors in tech target website say about CRM, the goal is improving customer service relationship and assist with customer retention and drive sales growth.

The biggest benefits of CRM in business come from: -automation, where a good CRM can automate menial, sales pipeline, and customer support tasks, -trend spotting where employee can take action on real time customer insights trough reporting and visualization features. This assignment will be designed using Spring Boot as an open-source tool used to create microservices and web applications, Java programing language and MySQL for create and administration of databases.

Spring Boot, as being said by its creators, VMware Tanzu, make job easier for students or programmers that want to develop stand-alone, production grade Spring based applications. To enhance the full power of Spring, developers have access and is recommended to use multiple futures such as:

* Already embed Tomcat, Jetty or Undertow (no need to deploy WAR files)
* Automatically configure Spring and 3rd party libraries whenever is possible.
* Multiple production-ready features are provided such as metrics, health checks, and externalized configuration.
* Not necessary for code generation.
* No requirements for XML configuration.

For developers at the start for journey, multiple Guides are provided, based on level on knowledge, so even if you are an already Spring user, you can find specific guidelines.

As being defined by its developers, MySQL or My Structured Query Language, is a software that deliver very fast, and with future such as multithread, multiuser and robust, data server. Devs created this application with intend for mission-critical, heavy-loaded production system as well as for emending into mass-deployed software. That’s why huge websites like Facebook from META, X rebranding and reinterpretation of Twitter, Airbnb or GitHub are using their services among a very large number of web developers around the world.

**Definition of the case study, analysis, and design:**

In this case, our customer provides a Software as a Service solution, for a better service – customer relationship they chose to use a Customer Relationship Manager (CRM) with some requirements for Contact Management part of the CRM, such as:

* Data base should keep data about all contacted people, customers or non-customers.
* Data base should have multiple records such as: date of contact, for an already contacted customer, employee who perfumed the contact action, reason of contact, brief description, action to be taken (like call, send email, move record), time schedule, a know your client description that help future automation.
* Several search option such as: finding non-customer contacts that have been reached by our employee or finding customer that no longer use our services.
* Send messages to multiple contact using a criterion (such as customer who birthday is close).
* Provide reports and statistics to the managers and investors (such as new customers in past 30 days or most engaging employee).

Analyzing the problem and requirements, we engage terms like relationship between customer-service-employee, from where we get a multitude of deductive requirements for CRM such as: importance of knowing your client and his needs (this is why the data base should contain multiple type information’s about customer), employee needs of fast and reliable way of tracking his customers and implement actions (In this case the employee have a certain need of performing different searches for customers and should have multiple ways of contacting customer ).

Using a simple but on point design will confer to the customer the easiest way to reach any of our services that we provide. As relational overview data base should contain table as: employee, customers and non-customer, where one employee can have multiple customers and being able to assign any non-customer, inside this table we should see data like id, name and specialization for employee, and data like name, location, telephone, description, calls description etc.

**Implementation:**

In order to achieve this, we will use a framework like spring boot, that will provide us a way to build a simple web application and accelerate the development using some useful services to it, and to build, contain, access and modify data base we will use MySQL database services.

I chose Java and the Spring framework because of the high level of complexity to create, debug and deploy Java applications contained in the developers favorite coding language, as many programmers and tech geeks call it. And MySQL for being a high-speed data processing and data productivity tool, easy to learn without a programing background.

In order to deliver not only all requirements and expectations to our business customer, but to also implement new futures our project need to follow some backlog requirements such as: user requirements, prioritization, dependences, acceptance criteria, estimation, release planning, backlog refinement and traceability.

Our business partner is requiring us to:

* Keep record of all people contacted, either they are customer or not. In this case our data base should contain 2 tables, one for customers and the other for contacted persons that did not want to work with us.
* Record contact details, employee who did the contact, reason of the contact, brief description, action to be taken. For this we will have to implement to our contacted customer table columns were information about topics like employee who did the contact, reason of contact, etc. can be added.
* Several options to search who to contact. In order to accomplish this requirement employee have to be able to search inside the database based on different criteria, and database must have those criteria. The search will be performed by a function written with java.
* Send messages to selected contacts, based on criteria. In this case employee should be able to do database search based on criteria and perform action for the result.
* Provide reports and statistics to the managers and investors. In this case our program should perform inside database some predefined searches and show information inside a managerial area of CRM.

In order for the product to be web based and open for extension, the project need to follow some key building steps such as:

* Using appropriate web technologies such as HTML as main web page language, CSS language used to design web pages and JavaScript frameworks as a collection of pre-written, reusable code.
* The web servers need to handle HTTP request and response.
* Database that we use have to be web compatible. In our case MySQL.
* Web security measures have to be implemented.
* Regularly updates have to be implemented based on user’s feedback.

As security implementation, this can be implemented by developers such as password requirements, disable integrated OS authentication or by a 3rd party software such as Cloudflare, the second one requires money compensations but being a plus to protections from already existing malware.

**Evaluation:**

Framework role of our CRM service is transforming organization effort and investments into positive customer experience. So, in order to do that we use framework ass a strategy to help customer to meet his objectives and also to fulfill to his requirements, and to obtain that we need a fast, efficient service with no or fewer errors. And how can you do that better than using an already existing open-source collection of pre-written code, used by java developers to create java applications or web apps. In our case Spring Framework fulfill our requirement with an all-inclusive programing and configuration model for modern java-based enterprise application.

Maybe the biggest challenge is how to obtain the fastest service, here intervene the design patterns that take care of repetitive task in software development. Design patterns are vital in organizing, reusability and simplifying the code and that confer it numerous benefits for users.

As main objective, a good user experience is defined by consistency and the level of intuition. Also, scalability between different platform and devices confer some past usage experience.

**Conclusion:**

My personal and subjective opinion as a customer type user and ex-employee that worked with a CRM program and without one, I can say that using this feature in business, cut down the time requirement to take action with a customer, keep all data in one place easy to be accessed. And as a customer, I can see some flaws, where some futures should be implemented such as: call appointment, better implementation of pattern recognition or, prediction of future customer behavior.

Going through the business requirements we realize that key of success is good user experience. This can be obtained with different tools like frameworks and design patterns, or with the fastest and reliable connection between customer – employee.

We mitigate challenges such as storing huge amounts of data using MySQL services or efficiencies service using Spring Boot.

In conclusion, the journey that this assessment set us on, from conceptualizing and analyzing the requirements to the implementation of our Customer Relationship Manager (CRM) enlighten us with reward of knowing what should we do next, how a business is build and what is the important keystone that we should accomplish for its success, we also gain a deep understanding of how certain processes are performed such as backlog, analyzing, attribution of roles and usability of cases.

Being used by many company’s, a customer relationship management (CRM) is a critical tool for success is business. Being a major investment and depending on the solution you chose, require significant time and have the necessity to follow a guideline due to its power to manage and nurture relationship with your customer. Having his key futures is a necessity and the rest depending on company unique needs.

We should never forget to enjoy the time that we got and to improve because is in our power to choose what comes next.

**References:**

1. CRM tools and strategy. By Cameron Hashemi-Pour and Wesley Chai. [www.techtarget.com/searchcustomerexperience/definition/CRM-customer-relationship-management](http://www.techtarget.com/searchcustomerexperience/definition/CRM-customer-relationship-management)
2. 6 Main components of customer relationship management (CRM). By Tristan Ovington. <https://www.walkme.com/blog/components-of-crm/>
3. Building and Application with Spring Boot. <https://spring.io/guides/gs/spring-boot/>
4. What is Java Spring Boot?. <https://azure.microsoft.com/en-us/resources/cloud-computing-dictionary/what-is-java-spring-boot>
5. Top 25 Best Database Management Software in 2023. By Dimitry Nasikanov. <https://improvado.io/blog/top-25-best-database-management-software>
6. What is a backlog. <https://www.productplan.com/glossary/backlog/>
7. Top 10 Best JavaScript Frameworks in 2024. By Robert Johns. <https://hackr.io/blog/best-javascript-frameworks>
8. The IT regulatory and Standards Compliance Handbook. By Craig Wright. 2008
9. How the web works. By MDN contributors. <https://developer.mozilla.org/en-US/docs/Learn/Getting_started_with_the_web/How_the_Web_works>
10. How to secure web application. By Ankita Kapoor. <https://medium.com/quick-code/how-to-secure-web-apps-a-web-app-security-checklist-bb27cf049d1d>
11. How to build a framework for CRM success. <https://www.pipedrive.com/en/blog/framework-crm>
12. Spring Framework – Overview. <https://spring.io/projects/spring-framework/>
13. Introduction to MySQL. <https://www.w3schools.com/mysql/mysql_intro.asp>