**LAB 6**

LIC Market-Driven System LIC, an insurance company, wants to digitize a range of business processes and provide a complete solution that addresses all aspects of the agent-insurer relationship. Consider yourself as a part of the Requirement Analyst team at Retinodes Software Company, and your job is to gather and prioritize the set of requirements. In this new requirement of the project, there are no existing systems that can be analyzed for the development. Requirements have to be gathered, negotiated, validated and prioritized through multiple stakeholders which is a complex process because all stakeholders have different perspectives, requirements and priorities. Therefore, Retinodes want to have a requirements engineering framework that can be used in market-facing projects. To start with, you need to identify the set of stakeholders associated with the system, the domain information about the insurance market, and possible features. The first product LIC wanted you to develop consolidated insurance packages which can compete with the packages provided by other insurance companies. Another product is based on the customer priority, based on the insurance policies available the customer can create his/her own package and send a request for the review. The system has to automatically analyze the package, provide suggestions (if any), and at last give a competing price for the package. To understand the problem domain, existing packages have to be analyzed and the demands and restrictions from the insurance policy and agents have to be understood completely. The requirements and feasibility report generated by you, will be further used by the development team for implementation.

**Q1. Identify all the stakeholders and users of the systems. Enlist all features of the LIC Market-Driven system by each user of the system, in the form of user stories. Can you prioritize them using the requirement prioritization techniques? (e.g., AHP, Numerical Assessment, MoSCoW method, etc.) How? Provide details.**

Stakeholders: -

1) LIC

2) Customer

3) Agent

Users: -

1) Agents

2) Customers

3) Admin/Maintenance

4) Employee

User Stories: -

1. Customer :
   1. As a Customer, I can create my own package so that I can include all requirements that I need.
   2. As a customer I want to be able to see all kinds of packages , so that I will be able to select appropriate packages.
   3. As a customer I would be able to make an account through that I can access information related to policies and connect with the agent
   4. As a customer I would like to pay my premium online
   5. As a customer I want the system to inform me about due premiums
2. Agent :
   1. As an agent I would be able to see the names of all the customers.
   2. As an agent I would be able to see if new customer has made an account and finding for the guidance
   3. As an agent I would be able to make an account through which I can access information related to policies and connect to customers.
3. Employee :
   1. As an employee, I would need to review the customized packages made by the customers
   2. As an employee, I would need to new packages and update the older packages
4. Admin/Maintenance :
   1. Maintain the system and ensure that all the features are working properly.

The four capitalized letters in the MoSCoW prioritization scheme stand for four possible priority classifications:

* Must: The requirement must be satisfied for the solution to be considered a success.

Account creation , policy selection , bill payment

* Should: The requirement is important and should be included in the solution if possible, but it’s not mandatory to success.

Show the best suitable policy given requirements.

* Could: It’s a desirable capability, but one that could be deferred or eliminated. Implement it only if time and resources permit.
* Would: This indicates a requirement that will not be implemented at this time but could be included in a future release.

**Q2. Prepare a list of market-facing technologies helpful for this project. According to you, would marketfacing technologies be helpful in the proper deployment of the product? Why?**

Ans: Market-facing technologies that are helpful for the development of this project are:

1. <https://www.insurancedekho.com/>
2. <https://www.policybazaar.com/>
3. <https://www.policybazaar.com/>
4. <https://www.policyx.com>
5. <https://www.easypolicy.com/>
6. Content Management System
7. Email Marketing, Social-media Marketing
8. Customer Experience Software
9. Marketing Attribution Software

These are a few market-facing technologies that would be helpful for the development of this project as they belong to the same domain as the LIC Market-Driven system. All of them are very user friendly and have similar features to that proposed by the LIC Market-Driven system. This will also help us understand the competition out there and make a system advanced enough to outshine these other market-facing technologies.

Market-facing technologies are the technologies which directly interact with the customers of any organisation. These technologies should be user-friendly and easy to navigate. They are also responsible for casting the first impression of an organisation in front of their users. These technologies are really helpful in deployment of the product as they are very easy to test from the perspective of the users. They are also very helpful in understanding the view of the customer and also ensuring that the website remains easy to navigate and highly user-friendly.

**Q3. Suggest an effective requirement engineering framework that can be used in market-facing projects because there are no existing systems that can be analyzed for the development so we need to consider all requirements from the core.**

Ans :

Requirement engineering framework that would be used is - Agile: Scrum because it’s requirement analysis phase is completely driven by user story so that we can clearly understand what the user wants and we can also verify our understanding of user requirements because customer involvement is very high in agile methodology. The multiple sprints would be useful in making the requirement gathering phase much more easier.

**Q4. List out the possible features those are not feasible to consider. Can you provide justification for each of them in detail?**

Ans :

* One of the features could be that the person can pay the premium of the insurance policy at any time but it is not feasible as by this some people may not pay premium at all. There must be a timeline for paying the premium.
* The insurance policy would remain the same for all the age groups and the premiums they would have to pay would also be the same. This feature is not feasible as the premiums would be high for the people of higher age group as in that age group there is more risk involved.
* There can also be a feature of group policy where 3 to 4 people are covered under the same policy. This policy looks attractive and many people would be attracted to it but it is not feasible as by this the risk to the insurance company will be increased.

**Q5. Let us assume that the customized package developed by the customer (using your second product) is similar to the package available in your pre-defined package. What is the possible reason behind this defect? How can it be ensured that this would not happen? In which requirements engineering activity, this defect can be handled? Please provide a scenario to justify.**

Ans:

Possible reason behind this defect: This defect could exist when the system is analyzing the package that the customer has created. After this, the system is supposed to provide suggestions based on the choices that the user has made. The defect would occur when the system is not able to suggest an already existing, similar package to the user because it is not able to identify that the customized package is very similar to a pre-existing package.

Requirement engineering activity to handle the defect: This defect can be handled during the testing phase. Intensive testing would uncover this defect and allow the team to make the necessary changes in order to ensure that this would not happen again.

**Q6. Identify three different use cases where the conflicts between the requirements occur? Do you think that the conflicts can be resolved? How?**

Ans:

* While analysing the custom packages, will the system be programmed take into account the necessities of the customer or would it focus on maximizing the profit for the company.
  + Resolution: Maximizing and taking into account the necessities of both the customers as well as the company.
* The user and the agent would both be looking for profits in their own aspects; agents would want higher prices while the user would want lower prices.
  + Resolution: Suggest reasonable prices that would benefit both parties.
* The customer might take advantage of the creating their own package feature and create the package in a way that would reduce the suggested price, even though the same package might already exist.
  + Resolution: Program the system in such a way that custom packages that are 85% - 90% similar to an already existing package should not be approved.

**Q7. Considering the set of features you have identified, what are the non-functional aspects associated with this system? Explain rationale behind the selection of each of them.**

Ans:

Non-functional aspects of the system are: -

1. Availability: Our system should be available during the working hours, so that customers could always buy a package of their choice.
2. Our database should be fully secure and protected, as that would contain the customers’ information and should not be made public maintaining the privacy concerns. Also, statistics related to each policy is of vital importance for the company to be competitive. This too should not be leaked along with other information.
3. Usability: Our system should be user friendly so that users can easily use the system.
4. Low latency: The time taken for each query result to be rendered should be as small as possible.

**Q8. Can there be ‘Open Issues’- issues those are identified but not taken care of? If yes, what are they? Are there some alternative ways for their resolution, such that no requirements conflict will happen?**

Ans:

The open issues in the system that have not been taken care of are :

1. The role of the agent of LIC is not quite clear since earlier the work of the agent was to suggest or make customised policies for the customers and act as middle person between the insurance company and customer.
2. People these days certainly are more skeptical than earlier to share their personal information and their income or property related details. So to imbibe trust in the customers for large financial transactions could be very difficult.