

# Software Engineering - IT314

## **Case Based Learning**

GROUP 32

### **Group members**

201801081 KAPADIYA KAUSHAL GAUTAMBHAI  
201801229 MISTRY SANKETKUMAR KIRANKUMAR  
201801009 RAMANI NAYAN GOBARBHAI  
201801054 TEJASWA ALIA  
201801003 POPAT JAYESH CHANDRESHKUMAR  
201801427 RIDDHI ATUL TANNA (Group leader)  
201801074 RITIK MALAVIYA  
201801224 ZANZARUKIYA JIGAR HEMUBHAI  
201801159 JOSHI DIKSHEN RIPAL

1. 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:**

Customer  
Agents  
LIC  
Government  
Investors  
Retinodes Software Company  
Banks

**Users:**

Premium members of LIC  
Customers  
Agents

**Features of the LIC Market-Driven system:**

- As a customer, I should be able to make an account in the given software.
- As a customer, I should be able to login with my credentials.
- As a customer, I should be able to view the insurance package policies available.
- As a customer, I should be able to make my own package.
- As a customer, I should be able to get suggestions about the package that I created.
- As a customer, I should be able to get chat assistance during package making.
- As a customer, I should be able to get a competitive price for my created package.
- As a customer, I should be able to pay for my packages using digital medium.
- As a customer, I should be able to compare the given package with the same package from different companies.
- As a customer, I should be able to receive updates about a better package later if I have used the system even once.
- As an agent, I should be able to view insurance package policies available.
- As an agent, I should be able to view all the details about the customers that I am managing.

- As an agent, I should be able to compare packages provided by LIC to other companies.
- As an agent, I should be able to notify my customers and update their policies to the current best policies and plans offered by LIC.

We can use the MoSCoW method for prioritizing the requirements like this:

S.No.	Requirements	MoSCoW
1.	Make account	Must
2.	Login into account	Must
3.	View already available packages	Must
4.	Provide consolidated insurance packages	Must
5.	Create and purchase customized packages	Must
6.	Analyzing created package and provide suggestions	Must
7.	Provide competitive price for created package	Must
8.	Pay for my customized package through digital medium	Must
9.	Agents should be able to view details about their customers	Should
10.	Comparisons between packages provided by LIC and other companies	Should

11.	Agents should be able to communicate with their customers to keep them updated about their policies	Should
12.	Live chat assistance to the customer during process	Could
13.	Customers should receive updates about better packages in future	Won't
14.	The customer would be able to compare their customized insurance packages with those of other companies	Won't

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

- Predictive analytics
- Machine learning
- Artificial intelligence
- Chatbots
- Content Management System/Database Management System

Yes, these market-facing technologies would be helpful in the proper deployment of the product since these technologies can help fulfil the requirements of the project. For instance, predictive analytics and ML can be used to predict the competitive price for the combination of packages chosen by the customer. Chatbots can be used to provide suggestions. AI can be used to make competitive packages for the product.

3. 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.

- Market and domain research
- Identification of stakeholders
- Interviews, surveys and questionnaires with stakeholders
- Storyboarding and brainstorming
- Use case modelling
- User testing with prototype - eliciting requirements according to their feedback

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

- Verification of the policy holders(new users to the system) cannot be satisfied through AI.
- The claim for the policy cannot be accomplished only through online mode; it requires a physical verification.
- The user won't be able to compare packages provided by LIC with those provided by other companies.

5. 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.

There can be two **reasons** for this defect:

1. The agent/system did not give the customer complete information regarding the pre-defined package.
2. The system does not have a check on whether the customized package is similar to the predefined package. This happens during the analyzing phase of customized packages, when the system is not able to see that the customized package is very similar to the already available package.

We can ensure that this does not happen by adding a check to the system where the system compares the customized package with the predefined packages to make sure that

they are not the same. We can also train the agents to give consistent and complete information to the customers so that they can make an informed decision.

We can identify this defect in the user testing phase.

**Scenario:** When the user is testing with the prototype, and they input custom options such that they make up a predefined package, we can intervene and identify the defect.

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

1. A conflict can occur if the customer is not willing to pay the amount of money that their customized package costs. This conflict may never be resolved since LIC would not negotiate and if the customer does not have enough budget to accommodate it, then we may lose the customer.
2. A conflict can occur if all the data of the user which is available to the agent is exploited. This would cause lack of trust among users in LIC and LIC would also have to deal with legal consequences. For solving this conflict we can limit the amount of data available to the agent and also take user permission before providing data to the agent.
3. If upon receiving information about better packages the user wants to change their package in between, before the expiry of the package, then they will not be able to do so. This conflict can be resolved by making the customer aware of this policy beforehand.

7. 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.

1. Performance and Scalability: System should work well even in the case of heavy load i.e. a lot of customers using it at the same time.
2. Availability: Users should be able to use the system on any device irrespective of OS etc and would be available 24/7 for the user to use.
3. Security: Data entered by the users should be encrypted so that it cannot be misused and it will have two phase authentication during login to prevent fraud.
4. Usability: System will have a simple, user-friendly interface so that anyone can use it easily.
5. Maintainability: System can easily be enhanced and restructured over time to provide better facilities to users.

6. Data Retention: The system should efficiently use all the data of current and previous users and their packages.

8. 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?

1. How will the system be aware of forged documents during verification? - we can link our system to the government
2. Will the system make sure that all recent government guidelines are followed?

**Additional Question:**

How do the requirements of the similar systems (other similar applications) match with the system under study here?

We can see that many of the requirements we have, such as features like secure login, data retention, compatibility on different devices, 24/7 availability, notifying the users and ease of use of the system match those of many other systems that we have come across.