

# Lab 6

## Group 21

1.

**a. Users:**

Insurance buyer, 3rd party sellers, Database, Insurance policy makers, Insurance Agents

**b. Stakeholders:**

Insurance buyer, 3rd party sellers, Database, Insurance policy makers, Developers, Testers, owners, managers, Insurance Agents, outsearching companies, Banks

**c. User stories:**

1. As an insurance buyer I want to be able to view the policy details so that I can decide if I want to buy it or not.
2. As an insurance buyer I want to be able to download the policy offline so that I can read it conveniently.
3. As an insurance buyer I want to be able to download/view multiple policies so that I can compare the pros and cons of different policies.
4. As an insurance buyer I want to be able to design a custom policy so that I can request for one.
5. As an insurance buyer, I want to be able to feed my customised policy into the software so that I can get an approval on it.
6. As an insurance buyer, I want to be able to view my previously bought policies all together so that I can refer to it.
7. As an insurance buyer I want to be able to view the premium amount pending each month so that I do not miss out on it.
8. As an insurance buyer, I want to be able to generate an automated statement so that it is easy to keep a record for me.
9. As an insurance agent deployed by LIC, I want to be able to view the policies of my customers so that I can view them all at once.

10. As a policy maker, I want to be able to put the policies systematically live so that everyone can view them.
11. As a manager I want my system to analyze policies of other companies so that the system can provide the most competitive policy.
12. As a manager I want that other third party sellers could view and buy insurance from their own sites so that my sales increase.

d.

We would use MoSCoW technique to prioritize the requirements.

- MUST (Mandatory) - 1,2,3,4,5,9,11
- SHOULD (Of high priority) - 6,7,8,10,
- COULD (Preferred but not necessary) - 12
- WOULD (Can be postponed and suggested for future execution)

2.

- a. TV advertising
- b. Referral Program
- c. Easy-to-use Website and native mobile application
- d. Quality social media marketing
- e. Email newsletters
- f. Press releases
- g. Customer video testimonials
- h. Higher reviews on services like Google Maps
- i. Polite and satisfactory customer
- j. Take customer feedback/surveys
- k. Discount on special Occasions
- l. Annual Policy reviews
- m. TV interviews / Podcasts

3. An effective requirement engineering framework should contain all the steps from gathering requirements to documentation.

Major requirement elicitation techniques used in market facing projects are Interviews of the users, Survey and Questionnaires,

Brainstorming, JAD. After performing the elicitation techniques, our framework should have an analysing component with keeping a large audience in mind. So, requirements elicitation technique should be performed on multiple types of clients & identify the conflict. Having collected all data we will try to address the conflict in a prioritised manner. And after this we will document the requirements. In case of multiple releases, traceability & validation of requirements is also very important. The requirement engineering framework containing all these qualities will be an ideal and effective one.

4. A. Online Verification: It is not possible for the users to verify their details online and they will have to rely on the KYC.

There are certain things that cannot be accommodated digitally for instance, the verification of health related issues when one wants to buy policies like those, reimbursement of car accidents, damages that the policy covers would also require manual intervention.

5. The possible reason behind this defect could be that the algorithm faces a glitch: Instead of comparing the customised policy with the predefined ones and seeing if it is in-turn similar in metrics to what is already pre-defined, it sent it to the algorithm for validation if the policy can be feasible to the customer to be sold. This can be a result of poor requirement analysis which in turn would have resulted in poor and unclear documentation which leads to ambiguous development stages. It could've been handled if the requirement analysis after gathering them were detained and fine grained after which a clear aim would've been given to the developers.

6. The conflicts can be as follows:

- When competing for a common scenario policy with a lot of other insurance companies in the market, the user will want a competitive price from the system for the policy while the manager would want to maximise the firm's profits. Hence, all sorts of custom packages would not be entertained by the system, A certain bound on the lowest profit will have to be kept for reference.
- The agents would want the commission from LIC for selling their policies but in the automated scenario, users wouldn't want to go via an agent rather buy policies directly.

## 7.

- Scalability - The system should be scalable because the number of customers might change from time to time.
- Responsive- The system should have a least response time of around 5ms to 10ms.
- Privacy - No user shall be allowed to access any other user's data. It should be secure.
- Availability - The system should be available 24x7
- Reliability- The system should not fail frequently otherwise it will affect the experience of the users.

## 8. Open Issues:

- a. There should be a mechanism whereby a package designed by one user can be added to the list of packages offered by the company, subject to verification by the managers and policy makers.
- b. The role of agents is not very clear as the role of suggesting policies was previously the job of agents but now the system automatically does this.
- c. Live chat feature between company and buyer which requires a lot of people working as a representative so it has been kept open for now.
- d. We have not yet decided that if the user wants to buy a policy should he pay the premium directly on the website or should he contact the agent to buy the insurance.