Laboratory 1

Title of the Laboratory Exercise: Requirements Analysis - I

1. Introduction and Purpose of Experiment

   Students get familiar with the documentation and scenario specified for all the lab exercises while analysing the requirements of the scenario

2. Aim and Objectives

   Aim

   - To develop formal software requirements in a standard format for a given engineering problem

   Objectives

   At the end of this lab, the student will be able to

   – Identify software requirements from problem statement
   – Identify type of a software requirement
   – Create an unambiguous list of software requirements based on interaction with a client

3. Experimental Procedure

- Work in teams of 7 students
- Each team should read the problem statement and identify requirements as a group
- Each team will then confirm the requirements and document the requirements in an SRS document
- Each individual will then write their lab manual, documenting their observations

4. Functional Requirements

1. [FR1] - The Software must allow valid users to sign in and sign up and identify the type of user.
2. [FR2] - The Software should display the list of products based on category.
3. [FR3] - The Software must display complete product details of a valid product.
4. [FR4] - The Software allows users to search, sort and filter valid products.
5. [FR5] - The Software must allow valid users to add valid products to cart.
6. [FR6] - The Software must allow to checkout a valid cart of a user with customer details and create an order.
7. [FR7] - The Software must allow the user to choose the mode of payment and complete order transaction process.
8. [FR8] - The System should give an option to provide coupon code during checkout.
9. [FR9] - The System must allow the user to track a valid order.
10. [FR10] - The System must allow the user to cancel or return an existing valid order.
11. [FR11] - The System must allow merchants to add valid products for sale.

5. Non-Functional Requirements

1. [NFR1] - The interface is clean, simple and easy to use.
2. [NFR2] - The search is fast and efficient.
3. [NFR3] - The system should be responsive
4. [NFR4] – The system should have an intuitive UI/UX
5. [NFR5] – The system should be secure.

6. Analysis and Discussions

The customer proposed the problem to us and we identified the requirements of the application that the customer wants. We first clearly understood the scenario of the application and then questioned the customer to gain clarity on all functionalities to be implemented. So as a team, we discussed the scope and operations supported by the application and elicited the requirements. We took note of the requirements and contemplated their nature, reason and feasibility. After analysing them and checking if

they meet the customer's demands, we listed out the requirements, with each requirement dealing with one functionality.

In the real-world, applications aren't that simple to arrive at a solution through a simple algorithm. Problems are multi-faceted and fairly complex. It is essential that the solution is both correct and efficient. The requirements of the application need to be identified and elicited so that the solution covers all aspects. The method and feasibility of implementing the requirements is analysed. The requirements are specified unambiguously, so that there is no confusion and it is clear to anyone in the software development loop presently and in the future. This model of approaching the problem is necessary so that the design is correct, easy to understand and implement and optimal. So, we followed the procedure wherein we first analysed the customer's proposal and then elicited the various requirements clearly, through discussion in the team and confirmation from the customer.

7. Conclusions

We follow the model of requirements analysis and elicitation when we have problems that are complex and the solution can't be determined by a simple algorithm. The problem stated has several requirements which are to be elicited and analysed, so that the design and implementation correctly solves the problem.

8. Comments

1. Limitations of Experiments

The customer's needs can be confusing or changing at times. It is best to clarify carefully what his needs are, and then eliciting the software requirements.

2. Limitations of Results

The requirements stated must be unambiguous, thorough, and each statement dealing with one aspect.

3. Learning happened

I learnt about the necessity and procedure for requirements analysis.

4. Recommendations

Name: SATYAJIT GHANA REG NO: 17ETCS002159

None

Component	Max Marks	Marks Obtained
Viva	6	
Results	7	
Documentation	7	
Total	20	