Software engineering ambiguities and omission computer science

Ambiguities and omission are statement that can be explained in number of ways. For example, the following statement is ambiguous. The operator identity consists of the operator name and password; the password consists of six digits. It should be displayed on the security vdu and deposited in the login file when an operator logs into the system. According to the question there are so many ambiguities and omission can be found in the given scenario. The main ambiguities and omission includes in given scenario as follows:

The structured approach described in study text can be dividing as follows: Preface, introduction, glossary, user requirement definition, system architecture, system requirements specification, system models, system evolution, appendices, and index. The first category of structured approach is preface. In this stage should define the expected readership of the document and describe its version history, including a rationale for the creation of a new version and a summery of the changes made in each version. A []A In the second stage of structured approach is the introduction. In this stage this should describe the need for the system. There should be a brief explanation about its functions and will it works with other systems. Acooding to the given scenario the ticket machine is made for perches rail ticket quickly rather than waiting in the que to perches rail ticket. When the user enters the start button he can chooses the destination. After selecting the destination user can see the availability of trains , train time, what kind of trains available(slow or fast, overground or underground train). When the customer chooses the destination, train, and the time user can purchases the rail ticket by paying card or cash. In the next stage of the structured approach which is user requirement definition, needs to define the services provide for the user. User requirements can be explained as follows: When user enters the start button he should abele to see the destinations. When the destination chooses he should be abele to see the train time and the ticket price. If the customer confirms the selected destination he should be abele to pay by card or cash. If the customer wants purchases more than one ticket their should be a option to select the numbers of ticket . After that customer should abele to chooses the payment method (cash or card). If the customer user wants to pay by card he should abele to input the card. After input the card if the user change his mind and wants to pay by cash there must be option to cancel the payment method as a card and choose the payment method as cash. If the customer paid by cash change and receipt must be given. The next stage of structured approach involves to given scenario is system requirement specification. This should explain about the functional and non functional requirement s in detail. According to the system requirement the system should be able to display the destination when the user selects the start button. When the customer chooses the destination system should be abele to display the availability of trains, time. and price. If there are no trains for chosen time system should be abele to display the alternatives (eg; replacement bus services). When the customer selects the train the system should be abele display the payment method (cash or card). According to the scenario user can only pay by credit card or cash, but the system should be able to take debit cards as well. Because most people user debit cards more than credit cards). If the user input a card before choosing the payment method or input a invalid card system should abele to displays the error massage. If user has been paid by cash system should able to gives the change back. After purchasing rail ticket by card or cash, the system should able to print the valid ticket to chosen destination and abele to provide the receipt for the payment which has been done? The next stage of the structured approach is ‘ System models’, which has been don in question (e.) . The last stage which involves to given scenario is ‘ system evolution’. This refers to the fundamental assumptions on which the system is based and anticipated changes due to hardware evolution, changing user needs. etc..(Eg: if the user wants to purchases the ticket online at home he should be able to log in to the system and purchases the train ticket.

(c.)Write the user requirements definitions.

The user requirements for a system could be divided to functional and non functional requirements, because it helps to the user to understand the system without technical knowledge. User requirements are defined using natural language, tables and diagrams as these can be understood by all users. There are so many problems can be generated when requirements are written in natural language.

Lack of clarity

It is something difficult to use language in a precise and unambiguous way without making the document wordy and difficult to read.

Requirements confusion-

Functional requirements, non-functional requirements, system goals and design information may be clearly distinguished.

Requirements amalgamation-

Several different requirements may be expressed together as a single requirement

User requirements can be defined as: the software must provide a means of representing and accessing external files created by other tools. A []A According to the above scenario user requirements can be explained as follows.

When the user presses the start button he should be able to choose the options (For example customer selecting a specific destination, the destination is in which zone, etc…)

When the customer chooses the destination user should be able to find out the train times and what kinds of trains (fast train or slow train, underground train or over ground train) are available for that time.

If trains are not available or delays at that time user should be able to find out the alternatives such as when is the next train available?, Is there

any replacement bus service available?, etc..

After choosing the destination, train and the train time, user should able to see the ticket price.

User should able to choose the payment method (card payment or cash payment) to purchases the rail-ticket.

If the user wants to pay by cash he should able to enter the cash and confirms the cash payment.

After confirming the cash payment, rail ticket should be printed and receipt for the payment and change needs to be given.

If the user wants to pay by card he should able to input the credit card or debit card and enter the validation pin.

User should be able to get the rail ticket and the receipt after payment has been made.

(d.)Write the system requirements specifications.

System requirements are expanded versions of the user requirements that used by software engineers as the starting point of the system design. A []A Normally they add details and explain how the user requirements should be provided by the system. According to the given scenario software requirements can be highlight as follows:

When the user enters the start button the system should be able display the destinations.

When customer chooses the destination the system should be displayed the train availability, what kind of trains available (fast, slow train or over ground, underground) of chosen destination and the departures time.

If there are no trains available at that time the system should able display saying that ‘ there are no trains available at chosen time enter the more option button to check the alternatives. When the alternatives selects system should be abele to display the alternatives(eg. take the replacement bus 472 towards London bride and take the northern line towards Morden - estimated time 1 hour and 32 minutes)

If the trains available, after the choosing the destination and the departure time, the system should be able to display the ticket price for the all kinds of trains. For example if the user wants to take underground train within zone 1-6 the travel card will be A? 6. 30.

When the customer selects the ticket type for the chosen destination the system should be able to display the payment method (pay by card or cash).

If the customer chooses the payment method as cash system should display how much user needs to pay totally and also should display a massage saying ' input the cash for perches the ticket)

When the customer input the cash the system should be able to charged exactly for the ticket price and change need to be given. Because most of the time users do not keep exact amount for the ticket. mostely they keep A? 10 or A? 20 notes.

If the customer chooses the payment method as card he should confirmed the payment method as card and needs to input the card. When the customer enters the pin the system should abele to verify the card and take the money from users account, but the card is invalid there should be a error massage should be displayed saying ' you have entered a invalid card please enter the valid card’. I f the validation is successful system should de abele to charge from uses account and provide the receipt.

(e.)Draw a sequence diagram showing the actions performed in the ticket-issuing system. You may make any reasonable assumptions about the system. Pay particular attention to specifying user errors.

Sequance diagram

(f.)Write a set of non-functional requirements setting out its expected reliability and its response time.

Requirements that are not directly concerned with the specific functions delivered by the system known as none functional system requirements. None functional requirements are not only concern with the software system to be developed, some may concern with the process that should be used to develop the system.

There are three non-functional requirements. They are

Product Requirements: Which specify the behaviour of the product? Ex: how fast are the system executed and how much memory dose it requires? Speed can be measured by processed transaction, event response time and screen refresh time.

Organisational Requirements: requirements driven by polices and procedures in the customers and developers organisation. Ex implementation requirements such as the programming language or design method used.

External requirements: Requirements that are driven from factors external to the system and its development process.

Also the time that the user take to get familiar with the system and number of help forums that are available, robustness of the system , how much time it take to restart the system in case of a failure occurred. Reliability that measures mean time to failure. Rate of failure accuracy availability and portability of unavailability. Portability of percentage is non-functional requirements that are important when designing a ticket issuing system.

(g.)Develop a set of use-cases that could serve as a basis for understanding the requirements for ticket-issuing system.

Use Case

(h.)Briefly describe the requirements validation process. Discuss all the checks that you have to perform to validate the above requirements in ticket-issue system.

Requirement validation concern with the specification of the system that customer wants is functioning according to the requirements. Requirement validation also examines the specification to ensure that all software requirements have been stated unambiguously; that inconsistence, omission and errors have been identified and correct them.

Following checks have been carried out on requirements

Availability checks

Since this is a ticketing system that is used by public. There are multiple users with multiple requirements. Therefore the requirement validation should be favourable for all users. However some users may find there requirements are fulfilled and some may not.

Consistency Checks

There should be no contradictory constrains or descriptions of the system function.

Completeness Checks

To check all the requirements have been achieved

Realism Checks

Once the requirements being gathered it is important to check that the system can be implemented with the current technologies and also it is possible to finish the project with the given time period with the allocated budget.

Verifiability

To reduce the potential of dispute between customer and contractor, system requirements should always be written so that they are referable

(i.)Create a semantic data model for the above scenario.

Data model

(j.)What is the impact if when the customer pays cash, he is allowed not to have the exact amount?

According to the given scenario if the user pays by cash he needs to pay the exact amount. For example if the rail ticket is A? 6. 30 user must pay exactly A? 6. 30. Specialy the cities like London most people don't carry change with them they keep A? 5, A? 10 or A? 20 notes. It is a user requirement to get the change back if the user inserts cash more than exact amount. System should abele to give the change back. However in the real life most of the ticket machines, if you put cash you get the change back.