

## TERMWORK - 1

### Identifying the Requirements for Problem Statement

#### Introduction:

Requirement Identification is the first step of any software development project until the requirements of the client have been clearly identified and verified no other task (design, coding, testing) could begin.

#### Characteristics of Requirements

Requirements gathered for any new system to be developed should exhibit the following 3 properties:

- 1) Unambiguity
- 2) Consistency
- 3) Completeness

#### A Library Information System

Problem Statement : Develop a Library information system (LIS) for SEV Labs Institute which enables the members to borrow a book (or return it). The system also enables a member to extend the date of his borrowing if no other booking for that particular book has been made. The librarian, who has the administrative privileges and control can enter a new record into system when a book has been purchased, or remove a record in case

## Exercise - 1

Consider the problem statement for an "Online Auction System" to be developed:

New Users can register to the system through an Online process. He can access the different features of the system authorized to him/her, after he authenticate himself through the login screen. An authenticated user can put items in the system for auction. Once the auction is over, the item will be sold to the user placing the maximum bid. The user selling the item will be responsible for its shipping.

① following are the ambiguities.

- There's no specification when an auction gets over

② following are the inconsistencies

- An item is said to be sold to the max bidder after auction is over; it can also be sold before the auction is over

③ The problem statement is incomplete because:

- ★ No mention of how a new user registers

- ★ No mention of any dispute regarding the sold product

- ★ No relevant mention of what kind of products could be put on auction

## Exercise - 2

Following functional requirements could be obtained from requirements specifications

- \* Registration : New users have to register themselves online with the site & accept its terms & conditions
- \* User Login : A user has to login into the site using his correct user ID & password
- \* Upload item for Auction : An authenticated user can upload an item into the site, which is to be put on auction subsequently.
- \* Auction Item : User puts an item already uploaded by him into the site on auction
- \* Bid for Item : Any registered & authenticated user of the system could place a bid for an item on auction
- \* Win Auction : After the end auction is over, the maximum bidder for the item owns the item post payment
- \* Ship Item : Seller of the item ships the item to the auction owner after he (seller) receives the payment
- \* Remove Item : Owner removes an item after uploading it, and doesn't put on auction
- \* Remove Auctioned Item : System automatically removes an item from its inventory after it has been successfully auctioned

### Exercise - 3

Following possible non-functional requirements could be identified:

- ★ The system should remain up & running throughout its working hours
- ★ Sessions of different users must not affect each other
- ★ System should maintain privacy of their users and should be able to service 100 users simultaneously
- ★ System could remain unavailable for upto 2 hours for maintenance once in a quarter with 36 hour prior notice

## Conclusion:

In this termwork, we learnt about different aspects that have to be taken care of while writing requirements specification. We also identified different functionalities to be obtained from a system and also identified characteristics that a system should have, but not done by the system itself.

## References:

- ★ <http://vlabs.iitkgp.ernet.in/se/>
- ★ Grady Booch, James Rumbaugh, Ivar Jacobson, "Unified Modeling Language User Guide", Publisher: Addison Wesley

## Termwork - 2

### Problem statement:

Modeling UML use case diagram from problem statement requirements and capturing scenarios of use case diagram

Exercise 1:

Consider a Library, where a member can perform two operations: issue book and return it. A book is issued to a member only after verifying his credentials.

Draw a use case diagram for the problem

### Exercise 2:

Consider your neighboring travel agent from whom you can purchase flight tickets. To book a ticket you must provide details about your journey i-e on which date and at what time you would like to travel. You also need to travel provide your address. The agency has recently been modernized. So you can pay either by cash or by card. You can also cancel a booked ticket later if you decide to change your plan. In that case you need to book a new ticket again. Your agent also allows you to book a hotel along with flight ticket. While cancelling a flight you can also cancel hotel booking. Appropriate refund as per policy is made in case of cancellation.

### Conclusion:

In this framework, we identified the actors & use cases & associated the use cases with actors by drawing a use case diagram

### References:

- \* <http://vlabs.iitkgp.ernet.in/~se/>
- \* Grady Booch, James Rumbaugh, Ivar Jacobson, "Unified modeling Language User Guide", Publisher : Addison Wesley

## Termwork - 3

Problem Statement:

E-R modeling from the problem statements entity relationship model, entity set and entity relationship

### Exercise 1 :

SF VLabs Inc. is a young company with a few departments spread across the country. As of now, the company has a strength of 2004 employees. Each employee works in a department. While joining, a person has to provide a lot of personal and professional details including name, address, phone, mail address, date of birth and so on. Once all these information are furnished, a unique ID is generated for each employee. He is then assigned a department in which he will work. There are around ten departments in the company. Unfortunately, two departments were given same names. However, departments have ID's which are unique.

## Exercise 2 :

The latest cab services agency in the city has approached you to develop a Cab management system for them. They would be using this software to efficiently manage and tracking different cabs that are operated by them.

Cabs are solely owned by the agency. They hire people in contracts to drive the cabs. A cab can be uniquely identified by, like any other vehicle in the country, its license plate. A few different categories of cars are available from different manufacturers & a few of them are AC cars.

Cab drivers are given a identification card while joining the ID card contains his name, permanent address, phone number, date of joining, duration of contract also an unique alphanumeric code is assigned to each number.

The agency provides service from 8AM to 8PM. Whenever any passenger books a cab, an available cab is allocated for him. The booking receipt given to the passenger contains the car #, source and destination places. Once he reaches the destination, he signs on a duplicate copy of the receipt and gives back to the driver. Driver must submit this duplicate copy signed by the passenger at the agency for confirmation.

To evaluate their quality of service, the agency also wants a (optional) customer satisfaction survey, where passengers can provide feedback about journey through the agency website

## Conclusion:

In this framework, we identified the classes & how they are related to each other by drawing E-R diagram. ER modeling helps you to analyze data requirements systematically to produce well designed database.

## References:

- \* <http://vlabs.iitkgp.ernet.in/>
- \* Grady Booch, James Rumbaugh, Ivor Jacobson, "Unified Modeling Language User Guide", Publisher: Addison Wesley

## TERMWORK - 4

### Problem Definition:

Design a relational database for an application involving atleast 5 tables and build GUI using Java Swing / Web / any other ... to perform functional operations of the applications

★ Transactions :

In order for a database management system to operate efficiently & accurately , it must use ACID transactions

Examples :

★ Student table :

SSN	Name	Major	GPA
1234	Jeff	CS	3.2
2345	Mary	math	3.0
3456	Bob	EC	2.7
4567	Wang	EE	2.9

★ Department table

Name	Location	Chairperson
CS	N18 EB	Agarwal
EE	Q11 EB	Jackson
math	CN 2200	Harrison
Biology	210-S3	Smith

★ Courses table

Name	Course	CreditHours	Dept
Database	CS432	4	CS
O.S	CS532	4	CS
Dis math	M314	4	math
Astronomy	AE341	4	Aeronautical

\* Sections table :

S.-id	Course	sem	Year
85	CS432	3	21
9	CS532	4	22
7	M314	3	21
23	AE341	5	22

A Prerequisite table

Course	Prerequisite - No
CS4338	CS433
CS3390	CS441
MAT315	MATH3
AE341	AE243

### Conclusion:

In this termwork, we designed a relational database for an application involving 5 tables & built a GUI using Java-Swing. We also understood working of JDBC connection and operations involving JDBC.

### References:

- \* <http://vlabs.iitkgp.ernet.in/~oe/>
- \* Grady Booch, James Rumbaugh, Ivar Jacobson, "Unified modeling Language user guide", Publisher: Addison Wesley.

## TERMWORK - 5

Problem Statement:

Draw statechart and activity diagram for following Exercises

### Exercise 1

Draw a statechart diagram to graphically represent the following system

Consider a bulb with a push down switch. The bulb initially remains off. When the switch is pushed down, the bulb is on. Again when the switch is pushed up, the bulb turns off. The lifecycle of the bulb continues in this way until it gets damaged.

## Exercise - 2

Draw an activity diagram to graphically represent the following workflow

Let us consider the development activities of SE Virtual labs. The process begins by checking out the code from Subversion repository. Necessary modifications are then made to the checked out code (local copy). Once the developer is done with his changes, the application has to be tested to verify whether the new functionality are working fine. This test has to be performed with two of the more popular web browsers. Firefox & Internet Explorer, to support cross-browser accessibility. If testing fails in at least one of the browser, developer goes back to his code, & fixes it. Only when all the browsers pass the test, a patch is generated from the local copy, & applied to the production code. The local copy is then committed resulting in update of SVN repository. Note that, if the local copy is committed before generating a patch file, then local changes would get registered and one won't be further able to generate the patch file.

### Conclusion:

In this framework, we identified different states of a system & activities performed in each state by drawing a state diagram. We also identified the activities that could be done in parallel & visualised work flow by drawing activity diagram.

### References:

- \* <http://vlabs.iitkgp.ernet.in/se/>
- \* Grady Booch, James Rumbaugh, Ivar Jacobson, "Unified Modeling Language User guide", Publisher : Addison Wesley.

## TERMWORK - 6

### Problem Statement:

Draw Class Diagrams and Sequence Diagrams for the following Exercises

## Exercise 1

A web browser is a software that helps us access a resource (web page) available on the World Wide Web & identified by a URL. A web browser consists of different sub-components, which can be primarily categorized into browser rendering engine, and browser control.

The rendering engine is responsible for displaying a requested page in web browser. Today browsers are not only limited to displaying text and images, but can provide access to audio & video components also.

The web browser control too, consists of several sub-components including navigations, window control, event handlers, page display. The navigation control aids users to request for web pages by specifying a URL & to navigate to other resources. Event handlers are responsible to identify the kind of activity that user is trying to do.

Represent the above problem with a class diagram. In particular,

1. Represent the individual classes
2. Represent how browser rendering engine, & browser are related to web browser class
3. Add methods that let a web browser retrieve a resource from the web server, and convince yourself.

### Exercise - 2

How would you represent the three-way handshaking mechanism of TCP with a sequence diagram?

### Exercise - 3

The web traditionally worked in a client-server model, where a web browser would send a HTTP request to the web server, and the server would send back a HTTP response to the browser. The HTTP request encapsulates contents of the requested resource in some format. If it requires a user authentication, HTTP request encapsulates login credentials & sends to the server. The server then checks if credentials are correct. The status of verification is sent back to the browser.

In the recent years there has been a shift from the traditional way of how HTTP works. A new technique called AJAX, has been proposed that lets asynchronous connection between a browser & web server. In traditional model, the browser sends HTTP request & wait for a HTTP response. The next response was sent after getting response from server.

AJAX, however, lets a web browser to send multiple HTTP requests one after another, without waiting until a response is received. This approach is found to be very helpful in cases when contents of only a portion of web page has to be updated rather than refreshing the entire page. Web 2.0 uses AJAX in many different cases for better user experience.

From the above problem statement :-

1. How would you represent the traditional web with a sequence diagram (in both cases when user verification is required or not)

Solution:

The figure shows how "traditional" Web typically works. "Resource 1" and "Resource 2" represent two resources, which could be accessed only after an user has authenticated himself. As seen in the picture, the "Web Browser" sends a synchronous "HTTP request" to the server for transferring any information from the web browser to web server. The "Web Server" in turns sends back any information by encapsulating it in a "HTTP response" object. The point to note here is that the message sent by web browser is synchronous. The web browser has to wait for a response and update its display when it arrives, and update its display when it arrives. Only after then the web browser could send out a request for any other resource.

2. What changes would appear in your sequence diagram if you are trying model a scenario where AJAX is being used.

Solution:

The corresponding sequence diagram is shown in the figure. Here, the "Web Browser" sends out asynchronous "XmlHttpRequest" to the "Web Server". As such, it doesn't have to wait for a response to arrive from the web server. Instead say, as a result of the user clicking on 2 buttons, web browser can simply send out two "XmlHttpRequest". It updates the display as and when a response arrives from the web server

### Conclusion :

In this framework, we identified classes, behaviour of classes & associated different classes of the system to get meaningful work done. We also identified Objects and arranged events in time sequence of their executions. Finally, we understood the use of asynchronous messages.

### References:

- \* <http://vlabs.iitkgp.ernet.in/se/>
- \* Grady Booch, James Rumbaugh, Ivor Jacobson, "Unified modeling Language user guide", Publisher: Addison Wesley

## TERMWORK - 7

Problem statement:

Draw Data Flow Diagrams for following scenarios

### Exercise - 1

Draw a context-level DFD to depict the typical user authentication process used by any system. An user gives two inputs - user name & password

## Exercise - 2

The Absolute Beginners Inc., is planning to launch a revolutionary social networking site, EyeCopy. You have been entrusted with designing a DFD for the proposed application. In particular, you have been asked to show following scenarios:

- \* User registration
- \* User login
- \* User update

Draw a Level 1 DFD to depict the above data flow & the corresponding processes

## Conclusion:

In this framework, we drew a level 1 Data flow diagram to depict the dataflow & corresponding processes. DFD provide critical insights into the systems and ways the information passes through it.

## References:

- \* <http://vlabs.iitkgp.ernet.in/~ae/>
- \* Grady Booch, James Rumbaugh, Ivor Jacobson, "Unified modeling Language user guide". Publisher: Addison Wesley.

## TERMWORK - 8

### Problem Statement:

Designing Test suites for software Testing for  
following Exercise .

## Exercise - 1

The Absolute Beginners Inc seems to have been fascinated by your work. Recently they have entrusted you with a task of writing a web-based mathematical software (using Javascript). As part of this software, your team mate has written a small module, which computes area of simple geometric shapes. A portion of module is shown below

```
* function square(side) { return side * side; }  
* function rectangle(side1, side2) { return side1 * side2; }  
* function circle(radius) { return Math.PI * radius * radius; }  
* function rightTriangle(base, height) { return 1/2 * base * height; }
```

Prepare a test suite that will verify each of above mentioned individual function is working correctly.

Modify the code to address this defect:

- 1) In each function, return -1 if any given dimension is negative.
- 2) Modify the test suite that it reflects desired performance for both correct & incorrect input(s)
- 3) The code has another bug - how would you identify it from testing results? Fix the bug & test it again

## Conclusion:

In this framework , we got familiarized with unit testing , verified implementation of functional requirements by writing test cases & analysed results of testing to ascertain the current state of a project .

## References:

- ★ <http://vlabs.iitkgp.ernet.in/>
- ★ Grady Booch , James Rumbaugh , Ivar Jacobson , "Unified modeling language user guide" , Publisher : Addison Wesley