Ankit Goyal
17103011

OOAD Assignment-2

① Types of Tracebility tables :-

(a) Feature tracebility table :

It shows how the requirements relate to important usefulness observable system / product features. It will have requirements from customers point of view.

(b) Source tracebility table :-

It identifies source of each requirement.
like ① what is basic need of requirement ?
② what pulls customer to state this requirements?

(c) Dependency tracebility table :-

It indicates requirements are related to one another.
If there are any dependency b/w requirement (and 2)
→ if there exists, how is it going to impact the final product of makes the customer to easily relate the requirements ~~with~~ with each other.

(d) Subsystem tracebility table :-

It categorises the requirements of the subsystem they govern. We can relate each & every requirement with the help of dependency tracebility table & using its result, we can categorise it into subsystems.

② Crossword :-

Down ① → Concerns

- Seperation of concerns (SoC) is a design for doperating computer programs into distinct section such that each section addresses a seperate concern
eg. HTML, CSS & JS
the above three are complementary. HTML for structure, CSS for styling & JS for adding the interactions.

Down ② → Flexible :-

Polymorphism ~~steps~~ or branches from inheritance. The whole idea is that ~~code~~ we have a general base class & more specific derived classes. One can write code for base class & polymorphism makes code working for all derived classes as well.

Across ③ Polymorphism :-

→ Polymorphism allows us to use subclass where superclass has been represented. This makes code writing flexible.

Across ④ Constructor :-

- it is helpful in initiating attributes associated with are object. So when each attribute was need a seperate setter method, a constructor is used instead.

Across ⑤ → Subclass

- in inheritance, the class which is inheriting from another class is known as a subclass.

Across-6 Hide =)

→ encapsolation helps is combining the related code in a class. This can be helpful to hide the data also called abstraction.

Across-① Superclass:

The class which is being inherited is called superclass.

~~② Organizational changes the exec~~

③ change management attributes to a proper cycle for making changes to IT frameworks. The objectives of progress the executive is to build meaningfulness & comprehension of perceived changes over an association & generates. that progressions are made is a smart manner that ceased negative effect on administration and clients.

- organizational change the execution is a cycle that ~~o~~ itemized arranging, clear objectives, open correspondent and consultant consideration laid to critical ~~for~~ labourers. Moreover, it change can be ~~criti~~ hosted chiefs may discover labourers. all made ~~are~~ ready to adjust their current schedules. Correspondence may work by & large, however in others it might may be profit direction ~~of~~ to form an arrangement for remonstrations for labourers who go through the change. on the ~~the~~ off change, that the change can be carried as a ideal cycle an opposed to an interruption to daily schedule.

change the execution for most part incorporate accompanying advances

Submission:- During this stage, a change is recognized and a change demand is submitted. The change is accessed, including deciding, the need level of administration and danger of proposed change, decide change type & change cycle to utile.

planning:- Plan the change, including the execution plan, booking, correspondence, plan, test plan, & move back arrangement.

Approval:- Acquiring endorsements for change plan from the execution are required.

Implementation:- Implement the change

Review:- Communicate and audit change, plan from execution as required.

Close:- The stage when the convey is effective & change is shut.

(4) **Crossword:-**

~~Down~~

Down 1 ⇒ functionality
Down 2 ⇒ Meaning
Down 3 ⇒ Essence
Down 4 ⇒ Risk
Down 5 ⇒ Scenarios
Across 6 ⇒ Significant
Down 7 ⇒ Great software
Down 8 ⇒ Simplicity
Across 9 ⇒ key
Down 10 ⇒ well ordered
Across 11 ⇒ wrong

Across (12) ⇒ Basic
Across (13) ⇒ increases
Down (14) ⇒ Great
Across (15) ⇒ Commonality
Across (16) ⇒ Most important
Across (17) ⇒ flexible

⑤ Availability :-

It refers to percentage of time that infrastructure, system or a solution remains operational under normal circumstances in order to deliver its intended purpose.

- The mathematical formula for this :-

$$\text{Percentage of availability} = \frac{(\text{total elapsed time} - \text{sum of down time})}{\text{total elapsed time}}$$

- This number percentage portrays a precise image of system availability, allowing organizations to understand exactly how much surplus they should expect from IT service providers
- merely having a service is not sufficient. when an IT service is available, it should actually serve the intended purpose under varying and unexpected conditions.

Reliability :-

Reliability refers to probability that system that systems will meet certain performance standards, in yealding correct output for a defined time duration. Reliability can be used to understand how well the service will be available in context of different real-world applications.

- Realiability is challanging to measure
- A common metric is to used a mean time b/w failure (MTBF)

$$MTBF = \left(\frac{\text{Total elapsed time} - \text{sum of down time}}{\text{no. of failures}}\right)$$

why MTBF is better?

$$MTBF = MTTF + MTTR$$

MTBF = mean time b/w failures

MTTF: Mean time to failure
MTTR: Mean time to repair

- Many researchers argue that MTBF is a far more useful measure than defects / KLoC or defects / FP. An end user is considered with failures, not with the total error count. Because, each error contained within a program does not have a system failure rate, total error count provides little indication of reliability of system.

For example:-

Consider a program that has been in operation for 14 months. Many errors in this program may remain undetected for decades before they are disconnected. MTBF of such obscure errors might be even 50 on even 100 years. other Errors yet ~~de-discovered~~ undiscovered, might have a failure rate of 18 or 24 months.