**Q1. Draw a Use Case Diagram**

**Answer-**



**Q2. Derive Boundary Classes, Controller classes, Entity Classes**.

Answer-

1. **Boundary Classes:**

 • Definition: Represent the interaction between the system and external actors.

 • Purpose: To define the boundaries of the system's interface with users or external systems.

• Features:

1. Often associated with UI elements or system interfaces.
2. Help in defining how external entities interact with the system.
3. **Controller Classes-**

• Definition: Represent the logic that manages use cases in the system.

• Purpose: To handle the flow of control and interaction between boundary and entity classes.

• Features:

1. Typically have few attributes.
2. Focus on behaviour and interaction logic.
3. Represented by a specific icon in UML diagrams.
4. Achieve the logic of use cases as defined in the Use Case Diagram.
5. **Entity Classes** :

 • Definition: Represent the data or persistent objects in the system.

• Purpose: To encapsulate data and its associated behaviour, often corresponding to database tables or records.

 • Features:

* 1. Store and manage system data.
	2. Frequently mapped to database entities (e.g., tables).
	3. Extracted similarly to entities in ER (Entity-Relationship) diagrams.
	4. Related to the DOA (Data-Oriented Approach).
	5. High module cohesion ensures stability and minimizes changes.
	6. Represented by a specific icon in UML diagrams.

**Q3. Place these classes on a three tier Architecture.**

**Answer-**

|  |
| --- |
|  User Layer |
| Payment Method Selection Boundary |
| Card Payment Boundary |
| Wallet Payment Boundary |
| Net Banking Boundary |
| Business Logic |
| Payment Controller |
| Card Payment Controller |
| Wallet Payment Controller |
| Net Banking Controller |
| Data Tier |
| Customer (Entity) |
| Card (Entity) |
| Wallet (Entity) |
| Net Banking (Entity) |

**Q4. Explain Domain Model for Customer making payment through Net Banking**

**Answer-**

A domain model is a system of abstractions that describes selected aspects of a

Sphere of knowledge, influence or activity. The model can then be used to solve problems related to that domain.

Customer making payment through net banking

1. First of all, he needs to login to the bank website
2. Enter the login credential
3. After successfully login happens, payment will be processed
4. Basically it contains Entity & attributes



**Q5. Draw a sequence diagram for payment done by Customer Net Banking**

**Answer-**

****

**Q6. Explain Conceptual Model for this Case**

**Answer-**

A conceptual model is a high level representation of a system that helps in understanding, visualizing and communicating the essential aspects of a domain.

It provides a clear and simplified view of domain, making it easier to understand.

Key elements of conceptual model are-

1. Entities- Customer, Product, Payment and Order
2. Attributes- Customer ID, Name, Email, Phone number
3. Relationship- For eg. A customer places an order

**Q7. What is MVC architecture? Explain MVC rules to derive classes from use case diagram and guidelines to place classes in 3-tier architecture.**

**Answer-**

MVC is an architectural design pattern that organises an application’s logic into distinct layers.

It is a software architecture pattern that separates an application into three main components i.e Model, View, and Controller.

* View- Represents the presentation layer of the application.
* Model- Represents the data and the business logic of the application.
* Controller- Acts as an intermediary between model and View

**MVC rules to derive classes from use case diagram:**

1. Combination of One Actor and a use case results in one Boundary class

2. Combination of Two Actors and a use case results in two Boundary classes

3. Combination of Three Actors and a use case results in Three Boundary classes and so

 on.... Note: only one primary actor is to be considered with a use case.

4. Use case will result in a controller class

5. Each Actor will result in one entity class

Once command is entered by user it goes to model through controller where processing is done and user can view the output of command on GUI.

Let us take an example where a Use Case: Employee is logging into his system

**Q8. Explain BA contributions in project (Waterfall Model – all Stages)**

Answer-

A BA is a facilitator for a project. The roles and responsibilities of business analyst are fundamental in satisfying stakeholder expectations and delivering a viable solution.

Contributions of a BA in different stages of waterfall model is as below:-

**Pre Project**- This involves performing Enterprise Analysis using various methods like SWOT Analysis, GAP Analysis, Market Research, Feasibility Study, and Root Cause Analysis. These methods help in preparing a Business Case, Project Scope and understanding the risks involved.

**Planning, Estimations & Assessments-** Planning an approach strategy for the project plan is extremely important for the success of the project. A BA conducts stakeholder analysis and understands the assumptions and constraints along with business rules and business goals.

**Requirements Gathering**- Requirements are an essential ingredient in any project because they form a foundation upon which projects are built. The requirements gathering process is pretty much a partnership between the business analyst, stakeholders, and the development team. This includes identifying stakeholders and using elicitation techniques to gather requirements. And prepare BRD and make prototype for client understanding, sort the requirements and prioritize and validate them.

**Requirement Analysis**- Preparing UML diagrams and functional requirements from business requirements for better understanding of the requirements. Take signoff in SRS document which is legal binding document between technical team and business, prepare RTM. Track the progress during SDLC till testing phase.

**Design**- A BA communicates with clients on the design and solution documents to make them understand how the solution would look. A BA prepares prototypes and mock-ups along with the design teams for helping the client understand about the final product and also take feedback on the same for any changes and improvements.

**Coding**- BA organizes JAD sessions, clarifies queries of technical team during coding. A developer refers to the diagrams and documents prepared by a BA to code their unit. Regular status meetings are conducted to update the client on the progress.

**Testing**- A BA prepares test cases from the use cases or assist testing team to do so. A BA performs high level testing and prepares client for UAT, updates end user manuals, updates RTM and takes sign off on UAT.

**Deployment and Implementation-** BA forwards the RTM to the client or PM to be attached to the project closure document, coordinates to complete and share end user manuals, plans and organizes training sessions for end users. BA prepares lessons learned from this project to take precautions for upcoming project.

**Q9. What is conflict management? Explain using Thomas – Kilmann technique**

**Answer-**

Conflict management is the process of identifying, addressing, and resolving disagreements or disputes in a constructive manner. It aims to minimize the negative impacts of conflicts and enhance collaboration, ensuring productive outcomes while maintaining healthy relationships among individuals or teams.

The Thomas-Kilmann Conflict Management Model is a widely used framework that identifies five strategies for managing conflict, based on two dimensions:

1. Assertiveness (the extent to which one tries to satisfy their own needs)

2. Cooperativeness (the extent to which one tries to satisfy others' needs)

The Five Strategies in Thomas-Kilmann Technique are as-

1. **Competing (High Assertiveness, Low Cooperativeness)**

Description: This approach is power-oriented and involves pursuing one's goals at the expense of others.

When to Use:

* Quick decisions are needed.
* Issues are critical and non-negotiable.
* Example: Enforcing a policy in a time-sensitive crisis despite opposition
1. **Collaborating (High Assertiveness, High Cooperativeness)**

Description: Focuses on finding a win-win solution where both parties' concerns are fully addressed.

When to Use:

* When the objective is to achieve the best outcome for all.
* To resolve complex conflicts requiring mutual input
* Example: Two departments working together to integrate their processes for mutual benefit
1. **Compromising (Moderate Assertiveness, Moderate Cooperativeness)**

 Description: Involves finding a middle ground where both parties give up something to reach a mutually acceptable solution.

 When to Use:

* + Time constraints exist.
	+ The solution is temporary or of moderate importance.
	+ Example: Splitting a budget equally between two competing projects.
1. **Avoiding (Low Assertiveness, Low Cooperativeness)**

 Description: Involves sidestepping the conflict without resolving it.

 When to Use:

* The issue is trivial or not worth the time.
* To allow emotions to cool down before addressing the conflict.
* Example: Delaying a heated discussion until both parties calm down.
1. **Accommodating (Low Assertiveness, High Cooperativeness)**

Description: Involves prioritizing the other party's needs over one's own.

When to Use:

* Maintaining harmony is more important than the outcome.
* The issue is more critical to the other party.
* Example: Agreeing to a colleague's proposal to maintain a good working relationship.

**Q10. List down the reasons for project failure**

Answer-

Project failure can occur due to various factors, often stemming from poor planning, communication, or execution. Below are some common reasons for project failure-

1. Improper planning

2. Continuous Change in Requirement

3. Lack of User Involvement

4. Improper Requirement Gathering

5. Inconsistently defined resources

6. Unclear objectives

7. Lack of detail control

8. Lack of transparency

9. Lack of communication

10. Change of direction

11. Unrealistic expectations

12. Unrealistic due dates

13. Poorly assigned roles

**Q11. List the Challenges faced in projects for BA**

**1. Lack of training:**

When it comes to a BA, it is a myth that they do not require technical skills. BA’s are involved in every step of the product development cycle; hence, they must understand the technical and functional side of the business as well. For this, they should be given enough training to understand the processes better.

**2. Change Management –**

With respect to cost and timelines there may be instances where the management has been changed and this may result in re doing of all the tasks that were discussed with the earlier management again.

**3. Coordination between developers and testers**

Keeping both the developers and testers in sync is a biggest challenge. The BA has to ensure and have frequent discussion with the developers and testers. Finding common time for both is a challenge.

**4. Conducting Meetings**

Several meetings need to be conducted between different stakeholders for discussions and requirement gathering. In order to organize the meeting, the BA needs to confirm their availability.

**5. People Management**

Coordinating with different people and different teams is a challenging task because they maybe involve in their day-to-day tasks and finding their time slot would be difficult.

**6. Unrealistic Timelines**

As a business analyst, you may find yourself in a problematic situation where timelines might be the concern. In that case, pressure is created, which might hamper your work. In that case, understand how to tackle such a situation while maintaining the quality of the work.

**7. Lack of Domain knowledge**

A business analyst needs to collaborate with the business users to understand the requirements. Domain knowledge plays a vital role in having a clear and complete understanding of the requirements. It is challenging for business analysts to be assigned to a wide variety of projects as learning new domains needs time and energy.

**8. Lack of up-to-date process**

The success of a project does not happen overnight. First, much effort and mental exhaustion are poured in to bring results. Following this, the lion’s share is the up-to date process of maintaining and evolving the project. The biggest challenge is the lack of up-to-date techniques and documentation. In most cases, the project documentation is incomplete, which hampers productivity**.**

**Q12. Write about Document Naming Standards**

Answer-

Document Naming Standards-

Document naming standards are predefined conventions used to create consistent, clear, and structured names for documents. These standards help in organizing, identifying, and retrieving documents efficiently, reducing confusion and errors. They are essential for maintaining uniformity across teams and projects.

Benefits of Naming Standards-

* Simplifies document search and retrieval.
* Enhances collaboration by reducing ambiguity.
* Supports version control and traceability.

• Project lead's last name or initials

• File creator's last name or initials

• Project name/acronym

• Date file created/generated (in YYYY-MM-DD format)

• Version number (with leading zeroes)

**Q13. What are the Do’s and Don’ts of a Business Analyst?**

* Never say NO to client.
* There is no word as “ by default”.
* Never imagine anything in terms of GUI.
* Never question the existence of the existence.
* Consult an SME for clarifications in requirements.
* Every problem of client is unique. No two problems are the same. The approach or technology may be different .
* Go to client with a plain mind with no assumptions. Listen carefully and completely until client is done and then you can ask queries.
* Do not interrupt the client when he is giving you the problem.
* Try to extract maximum leads to solution from the client himself.
* Never try to give solutions to client straight away with your previous experience and assumptions.
* Concentrate on the importantly required requirements.
* Don’t be washed away or don’t imagine solutions on screen basis.

**Q14. Write the difference between packages and sub-systems**

|  |  |
| --- | --- |
| **Package** | **Subsystem** |
| 1) Collection of Components which are not reusable in nature | 1) Collection of Components which are reusable in nature |
| 2) a package is a collection of elements with a namespace | 2)subsystem is a smaller system within your system |
| 3) In the UML, packages are used in a manner similar to the way directories and folders in an operating system group and organize files | 3)Subsystems are used for system decomposition |
| 4) A public element is visible to elements outside the package, denoted by ‘+’ | 4) One of the aspect is external view, showing the services provided by the subsystem |
| 5) A protected element is visible only to elements within inheriting packages, denoted by ‘#’ | 5) Another aspect is internal view, showing the realization of the subsystem |
| 6) A private element is not visible at all to elements outside the package, denoted by ‘-‘ | 6) A subsystem has a specification and a realization to represent the two views |
| 7)example: Application Development Companies work on Packages | 7) example: Product Development Companies work on Subsystems |

**Q15. What is camel-casing and explain where it will be used.**

**Answer-**

CamelCase is a way to separate the words in a phrase by making the first letter of each word capitalized and not using spaces. It is commonly used in web URLs, programming and computer naming conventions. It is named after camels because the capital letters resemble the humps on a camel's back.

Example: goThere(), completeTheProject(), getEmpId(), turnLeftAndSlowDown() Entire first word will be in lowercase and subsequent words first letter should be in Upper Case. There will be no gap in between words

Benefits of Camel-Casing

1. Readability: Improves the clarity of code by making compound words more distinguishable.

2. Consistency: Promotes uniformity in naming conventions, making code easier to understand and maintain.

3. Standardization: Aligns with best practices and conventions in many programming languages.

**Q16. Illustrate Development server and what are the accesses does business analyst has?**

**Answer-**

A development server is a type of server that is designed to facilitate the development and testing of programs, websites, software or applications for software programmers. It provides a run-time environment, as well as all hardware/software utilities that are essential to program debugging and development.

BA is involved in driving business requirement and converting them to solution requirements. He is involved in translating the solution features into software requirements. Then leads in analysis and designing phase, dictates in code development, then follows the testing phase during bug fixing as a change agent in the project team and ultimately fulfills the customer requirements.



Business Analysis overlaps heavily with analysis of requirements of the business to function as usual and to optimize how they function. Some examples of Business Analysis are –

• Creating Business Architecture

• Preparing a Business Case

• Conducting Risk assessment

• Requirements Elicitation

• Business Process Analysis

• Documentation of Requirements

The responsibility set of a business analyst would require him to fulfil different duties in different phases of a project and they are elucidated below –

Initiation Phase: This phase will mark the beginning of a new project and a business analyst will vary out the following responsibilities −

• Assist in carrying out the cost-benefit analysis of the project.

• Understand the business case.

• Ascertain the feasibility of the solution/project/product.

• Help in creating the project charter.

• Identify the stakeholders in the project.

**Q17. What is Data Mapping:**

Answer-

Data mapping is the process of matching fields from one database to another. It's the first step to facilitate data migration, data integration, and other data management tasks. Steps of Data Mapping are as mentioned below:

* Step 1: **Define** — define the data to be moved, including the tables, the fields within each table, and the format of the field after it's moved. For data integrations, the frequency of data transfer is also defined.
* Step 2: **Map the Data** — Match source fields to destination fields.
* Step 3: **Transformation** — if a field requires transformation, the transformation formula or rule is coded.
* Step 4: **Test** — using a test system and sample data from the source, run the transfer to see how it works and make adjustments as necessary.
* Step 5: **Deploy** — once it's determined that the data transformation is working as planned, schedule a migration or integration go-live event.
* Step 6: **Maintain and Update** — for ongoing data integration, the data map is an living entity that will require updates and changes as new data sources are added, as data sources change, or as requirements at the destination change.

**Q18. What is API. Explain how you would use API integration in the case of your application Date format is dd-mm-yyyy and it is accepting some data from Other Application from US whose Date Format is mm-dd-yyyy**

**Answer-**

An application programming interface (API) is a messenger that processes requests and ensures seamless functioning of enterprise systems. API enables interaction between data, applications, and devices. It delivers data and facilitates connectivity between devices and programs.

If an application needs the date format to be DD/MM/YYYY, then we can apply this validation on the form collecting information to ensure that the application is receiving and processing a valid date format. This, however, is not the same for API applications. We need to ensure that the API is well written and is able to enforce all these validations, distinguish between valid and invalid data and return the status code and validation error message to the end-user through a response.

The shipping date is considered valid if it is Not in the past Greater or equal to today’s date Is inacceptable format: DD/MM/YYYY

<For all valid shipping dates >

Response Status code = 200

Message: OK

The shipping date that does not meet the above criteria should be considered invalid. If a customer sends an invalid shipping date, then it must respond with the following error message(s):

<If date format is incorrect>

Response Status code NOT 200

Error: The shipping date provided is invalid; please ensure that the date is in DD/MM/YYYY

format

<if date is in past>

Response Status code NOT 200

Error: Provided shipping date is in the past