**Q1. Draw a Use Case Diagram**

**Answer-**

Use case diagram is a visual representation of how end users interact with a system. It is used by the Business Analysts to model a system’s functionality and capture its requirements.

There are four main components of a use case diagram-

Actor, System, Relationship and Use case.



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

**Answer-**

**Boundary Class** is a type of class used in software design, particularly in the context of object-oriented design and the Model-View-Controller (MVC) architecture. It acts as an interface between the system and external factors, such as users, external systems, or devices. Combination on of 1 actor and a use case is one boundary class, Combination on of 2 actors and a use case is two boundary class.

Example: Customer Registration, Customer Login, Bank Server Login.

**Controller Classes:** A Controller Class is a key component in software design, particularly in the Model-View-Controller (MVC) architectural pattern. Its primary role is to manage the flow of data between the Model (business logic and data) and the View (user interface). It processes incoming requests, handles user input, and updates the model and view accordingly.

Example: Login Controller, Payment Controller, Email Controller.

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

**Answer-**

|  |  |
| --- | --- |
| User Layer | Customer RegistrationCustomer LoginBank Server Login |
| Business Logic layer(Primary actors associated with the Boundary class) | Payment controller |
| Data Layer(All the entity classes) | Customer, payment |

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

**Answer-**

Domain Modelling is the process of creating a conceptual representation on of the key elements, relationships, and rules within a specific domain or problem space in so ware development. It helps define how data and processes are structured in a system by modelling real world entities and their interactions. Key Elements can be mentioned as below-

**Entities**: Represents, the key or the object within the domain. Example: Customer, Payments, Order.

**Attributes**: Properties of a particular entity. Example: Name, Email, Phone, Address, DOB

**Relationships**: Connections or Interaction between the Entities. Example: CUSTOMER places ORDER & ORDER contains multiple PRODUCTS.

**Aggregates**: Entities, which are grouped in a single unit. Example: ORDER contains ORDER ITEM & SHIPPING DETAILS.

**Business Rules:** These can also be called as the Constraints which defines the Domain. Example: Payment is the pre-requisite for the Delivery of the Product.



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

**Answer-** A Sequence Diagram is a type of UML (Unified Modelling Language) diagram that visually represents the interaction between objects or components in a system over time. It shows how objects communicate through method calls or messages in a sequential order.

****

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

**Answer-**

A Conceptual Model is a high-level representation of how a system or process works. It focuses on abstract concepts and relationships between them, providing a simplified overview of the system's structure without diving into technical details. Here are some key components below-

**Entities**: The main objects or concepts in the system, Example: Customer, Product.

**Attributes**: Descriptive properties of the entities, Example: Customer Name, Product Price.

**Relationships**: Connections between entities, Example: A Customer places an Order.

**Constraints:** Rules governing how entities interact, Example: Each order must have at least one product.

**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

**In 3-Tier Architecture:**

|  |  |
| --- | --- |
| Model Classes | Customer, Payment, Net-Banking, Card, Cash. |
| View Classes | Login, View, Payment Option View, Net Banking View, Bank Selection View, Credential View, Payment Amount View, Payment Confirmation View, Logout View  |
| Controller Classes | Login Controller, Payment Option Controller, Net Banking Controller, Bank Selection Controller, Credentials Controller, Payment Amount Controller, Payment Confirmation Controller, Logout Controller. |

**Guidelines to place in 3-Tier Architecture:**

There are 3 Layers stated as below-

**Presentation Layer:** This layer is nothing but a user interface. View is inside this layer.

**Application** **Layer**: This layer is nothing but business logic. Model and controller are inside this layer.

 **Data Layer:** Classes which are responsible for data access and storage are in this layer. It contains the classes which interacts with the database, Files, and other data sources.

**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:-

|  |  |  |
| --- | --- | --- |
| **Sr. No** | **Stage** | **Activities**  |
| **1.** | **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. |
| **2** | **Requirement Gathering** | Stakeholders are identified. All the requirements are gathered from the stakeholder. BA and Project Manager participates in this phase. After completing this phase, BRD will be generated. |
| **3** | **Requirement Analysis** | The requirements are analysed to understand the scope of the project. Analysing means the BA will check all the requirements, if he founds conflicting requirements then the BA will talk to the concerned stakeholder to clear it, remove the ambiguous requirements. |
| **4** | **Design** | After the requirements are cleared, Design phase starts. This has a detailed design document that outlines the software architecture, user interface, and system components. BA will model the requirements through creating prototypes and mock ups along with the design team. And also take feedback for any change or improvement.  |
| **5.** | **Development (Coding)** | The Development phase include implementation. It involves coding the so ware based on the design specifications. Programmers or developer are involved in this phase. Here BA acts as a mediator between the development team and the stakeholders. |
| **6.** | **Testing** | In the testing phase, the so ware is tested to ensure that it meets the requirements and is free from defects. Testers are involved in this phase. |
| **7.** | **Deployment & Implementation** | BA, ensures that there is smooth transition from development phase to the production phase. Implementation is the final stage of waterfall model. It involves running the code for the very first me in production phase. Release manager handles this phase. |

**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 occurs when a project does not meet its defined objectives, deliverables, or expectations. It happens when the project's outcomes deviate significantly from the original goals in terms of scope, time, cost, or quality.

Below mentioned are the Challenges faced by BA in a project-

* **Improper Requirement Gathering**: If the requirements of the project are not gathered correctly, then this can lead to project failure.
* **Lack of stakeholder involvement**: A project can fail if the stakeholders are not participating in the process. The stakeholder’s input and feedback play very important role to meet the goals.
* **Ineffective or less communication**: If there are communication issues between stakeholders, team members then this can lead to misunderstandings or delays in project or even can lead to project failure.
* **Poor Risk Management**: Poor risk management can also lead to project failure. The team fails to identify the risks and do the risk mitigation, which can lead to unexpected challenges or delays in project. Lack of user involvement. Lack of executive support.
* **Unrealistic Expectation**: Means the goals that cannot be achieved or the goals that are out of scope.
* **Improper planning**: The project can fail if the planning is not done properly. The milestones, goals should be discussed. If there is no proper planning, then team may face difficulties in addressing the issues or to track the progress.
* **Insufficient Resources:** Insufficient resources can also lead to project failure. The project may fail due to lack of technology knowledge or lack of finances.

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

**Answer-**

Below mentioned are the Challenges faced by the BA-

 1. Lack of Training.

2. Obtaining signoff on the requirement.

3. Change Management.

4. Co-ordination between developers and testers.

5. Conducting Meetings.

6. Making sure the Status report is effective.

7. Driving clients for UAT completion.

 8. Making sure that the project is going on the right track and deliver as per the timelines without any issues.

9. Gathering clear and Unambiguous requirement.

10. Unable to understand what stakeholder is conveying

11. Scope Creep, change in the requirement or the scope of the project during the project life cycle can lead to scope creep.

12. Managing the stakeholder with conflicting interest, can be a difficult task for BA.

**Q12. Write about Document Naming Standards**

Answer-

Document Naming Standard-

A Document Naming Standard is a set of guidelines that define how documents should be named within an organization or project.

This standard ensures consistency, clarity, and ease of document retrieval and management. 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 dedicated server that is used during the software development process. It provides a platform for developers and testers to build, test, and debug the applications before they are deployed to the production server.

As a BA we have limited access to development server.

Below mentioned are the access for a BA-

**Read Only**: BA’s may be granted with the read only access to the development server. This will allow them to view the user interface of the application, navigate through the features and, they will be able to observe the behavior of the application.

**Limited Access**: Depending upon the project needs, the BA’s will be granted limited access to the specific modules in the application.

**Limited Configuration Access**- Here, BA have the authority to make changes in certain areas of application where they have the access.

**Q17. What is Data Mapping:**

Answer-

Data mapping is the process of connecting the data from one source to another. It's like creating a guide or map that shows how data in one place corresponds to data in another place.

This is essentially important when you are moving data between different system or databases to ensure that data stays consistent and accurate.

 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-**

API stands for Application Programming Interface.

It is the set of rules, protocols and tools that define how different software application should interact with each other.

It is a software intermediary that allows the two applications to communicate with each other.

API allows sharing of only necessary information and keeps the internal system details hidden, which helps the system security.

For the above scenario,

**Establish API communication**-We will set up API communication between our application and other application to exchange data.

**Do Data formatting**- while sending the data from one application to other, we will convert the date format from dd-mm-yyyy to mm-dd-yyyy.

While receiving the data from other application, parse the data and extract the date, month and year and re-arrange them accordingly.

**Perform Data Validation** and ensure that the converted date remains in a valid format.