

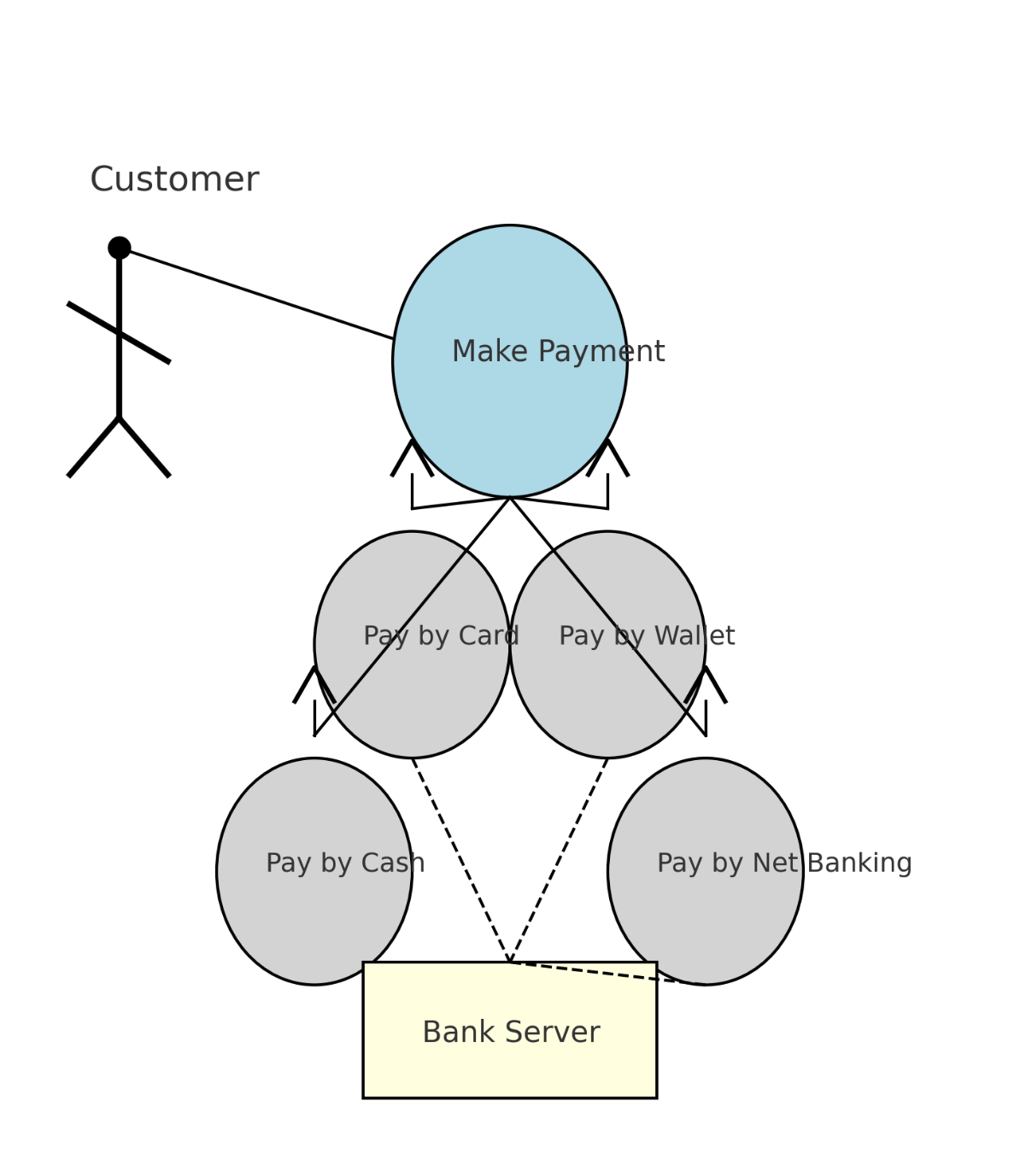
|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | | | | | | | |
|  | | CAPSTONE PROJECT-5 | | | | |  | |
|  |  | | | | | | |  |
|  | | | |  |  | | | |
|  | | | | Tanisha Mohane |  | | | |
|  | | | | February 2024—COEPD -prep exam 3— |  | | | |
|  | | |  | | |  | | |

A customer can make a payment either by Card or by Wallet or by Cash or by Net banking.

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |

Question No 1)- Draw a Use Case Diagram

Answer –



Question No 2)- Derive Boundary Classes, Controller classes, Entity Classes

Answer –

|  |  |
| --- | --- |
| Boundary class – (Use Cases) Actors speak to the system (Authentication information)   * Combination of 1 actor and a use case 1 boundary class * Combination of 2 actors and a use case 2 boundary class * Combination of 3 actors and a use case 3 boundary class and so on   All the actors should be – (Primary Actors)  Primary actors- Who initiate the use cases and interact with the system | Customer registration   * Customer login * Bank server logs in * Customers log out * Bank Server Logs out |

|  |  |
| --- | --- |
| Controller class – (Handles Users (Primary actors) input and processes the data   * Use case will consider as Controller class system | Registration Controller   * Login Controller * Payment Controller * Credential controller * Net Banking controller * Logout controller |

|  |  |
| --- | --- |
| Entity Class – All Actors   * Each actor will be considered as an entity | * Customer * Bank server * Cash * Card * Net Banking |

Question No 3)- Place these classes on a three tier Architecture.

Answer – In this three-tier architecture, the application tier handles the user interface, the Business Logic Layer manages the Business logic and coordinates between the other tiers, the Data layer tier handles data storage and retrieval.

|  |  |
| --- | --- |
| Application Layer | Customer Registration  Customer Login  Bank server login |
| Business Logic layer  (Primary actors associated with Boundary class) | Customer  Bank Server |
| Data base layer  (All the entity classes)- All actors | Customer  Bank server  Cash  Card  Net Banking |

Question No 4)- Explain Domain Model for Customer making payment through Net Banking

Answer –

Domain model is like the entity relationship model. The tables are connected to each other. In the diagram below, the customer table is connected to the bank table, which is why the customer is able to make payment.

Customer table is also connected to payment table, because he should make the payment. Now the payment is done by net banking, so payment table is connected to net banking table.

The account is in the bank, so the account table is connected to the bank table. The authentication table is connected to both net banking table and bank table, because authentication is to be performed there.

Also, the authentication table is connected to the transaction table, because authentication will be done during transaction.

Difference between ER diagram and domain model-

ER Model – do not have attributes inside the box

Domain Model- do have attributes mentioned inside the box

ER Model – it is a data modelling technique used in database design to represent table

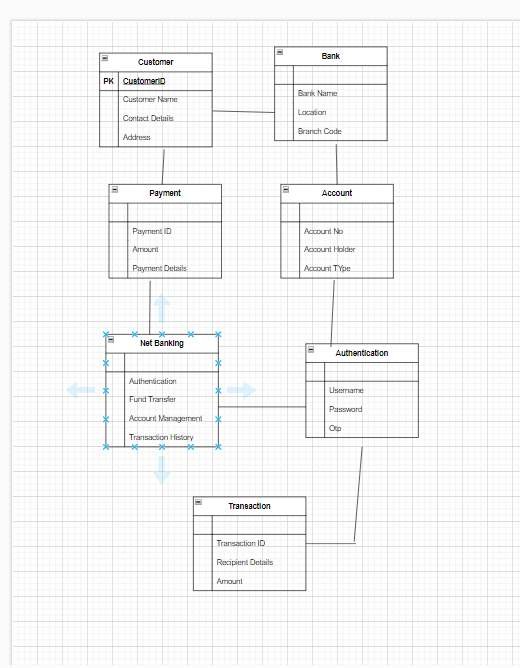
Domain Model- it is a conceptual model that represents real world entities.

ER Model – focuses on relationships required for storing and retrieving the data

Domain Model- It focuses on capturing the behaviors of application

ER Model –primarily used in database design

Domain Model-used throughout the software development lifecycle



Question No 5)- Draw a sequence diagram for payment done by Customer Net Banking

Answer –

This diagram shows how the objects in the system interact and communicate with each other with time to achieve specific tasks. Developer will draw this.

It is used to show the ow of messages, events or actions between the objects of the system.

This diagram helps to visualize the behavior of the system by showing the process in detail

A diagram of a customer

AI-generated content may be incorrect.

Question No 6)- Explain Conceptual Model for this Case Answer

Answer –

The conceptual model helps in understanding the key concepts, their relationships, and the overall structure of the net banking payment system.

It serves as a foundation for designing the database schema, defining the application architecture, and implementing the necessary functionalities within the system

The relationships between these entities can be described as follows:

1. Customer: This node represents the customers or users of net banking services.
2. Service awareness: Customers should be aware of the available net banking services and their features.
3. Privacy of data: The importance/significance of this node is to protect the privacy and confidentiality of customer data in the context of net banking.
4. Technology awareness: The significance of this node is that customers should be aware of and comfortable with the underlying technology used in net banking services.
5. Trust & Support: This node indicates that the bank provides such good services that it will help to enhance the customer’s trust.
6. Bank: This node represents a service provider responsible for offering net banking services.
7. Online information: This aspect highlights the importance of providing accurate and up-to-date online information about net banking services to customers.
8. Security & Privacy: The bank should adopt security policies which will help the customers to keep their data related to their transaction secure and private.
9. Infrastructure: This component suggests that the underlying technological infrastructure, including hardware and software systems, plays an important role in enabling net banking services.
10. Policies: This node represents the various policies and regulations that govern the implementation and operation of net banking services, ensuring compliance and customer protection

Question No 7)- 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 a design pattern where the application is divided into 3 logical parts-

Model, View and Controller.

Each of these parts will have specific responsibility.

Model-

The Model represents the data and the business logic of the application. Model is responsible for multiple tasks like managing the application's data, performing data validation, implementing business rules, and handling data access operations.

Model does not depend on how the data is presented or how the user interacts with the application

The model class is known about all the data that needs to be displayed. This layer corresponds to the data-related logic that the user works with. It represents the data that is being transferred between View and Controller. It can add or retrieve the data from the database

It responds to the controller’s request because the controller cannot interact with the database by itself.

The model interacts with the database and gives the requested data. All the model classes are nothing but the entities. Model classes are represented as entity class.

View-

The View is responsible -for presenting the data to the user for handling the user interface. The View can be a web page, a desktop application window, or any other form of user interface. It receives input from the user and passes it to the Controller for processing.

It represents the presentation of the application. View refers to the model.

It takes the data from the Model and renders it in a way that is suitable for the user's display or interaction.

For rendering the data, it uses a query method. View does not depend upon application logic. View classes are represented as boundary class

Controller-

The Controller acts as an intermediary between the Model and the View. It receives input from the user (via the View), processes the input by invoking the appropriate methods in the Model, and then updates the View with the new data or state.

The Controller handles user interactions, interprets user input, and translates it into instructions for the Model or the View. It coordinates the flow of data between the Model and the View, ensuring that they remain separated and independent of each other.

Whenever the user requests anything, that request first goes to the controller. Controller works on the user’s request.

Takes input from the user/ client. It interacts with the model and view. Controller class represents as use case. Controller acts as a mediator between model and database

Controller cannot directly get the data from the database. So, controller interacts with the model

Advantages of MVC-

MVC has the feature of scalability, which in turn helps the growth of application. The components are easy to maintain. A model can be used by multiple views that provide reusability of code. By using MVC, the application becomes more manageable. As all the three layers are different and independent, they are maintained separately

Rules to derive the classes from use case diagram-

* 1. Combination of one actor and one use case results in one boundary class.
  2. Combination of two actor and one use case results in two boundary class.
  3. Combination of three actor and one use case results in three boundary class.

1. Use case will result in controller class.
2. Each actor will result in one entity class.

Consider the example of Online shopping application with the following use case

Model Classes-Customer, Payment, Net Banking, Card, Cash

View Classes-Login View, Payment Option View, Net Banking View, Bank Selecton View, Credentials 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 classes in 3-tier architecture-

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

Tier architecture: - has 3 layers

* Application layer
* Business logic layer
* Data base layer

Data flowing from the 3 layers explains the 3-tier architecture where the information is passed. Here in elicitation techniques 3 tier architecture works as a medium where as a BA we get to know the information shared by the requester will fit in application layer, business logic layer or data base layer

Question No 8)- Explain BA contributions in project (Waterfall Model – all Stages)

Answer –

The waterfall model is useful in the situation where the project requirements are well defined, and the project goals are clear.

* Waterfall model follows sequential approach.
* In this model each phase is completed entirely and then only moved to the next phase.
* The waterfall model relies on documentation to ensure that the project is well defined, and the project team is working toward clear goals.
* Ones that phase has been completed and ones we move to the next phase, we cannot go back to the previous phase to make changes.
* This model is stable for the projects when the requirements are clear.

Requirements Gathering-

First, the stakeholders are identified. In this phase, all the requirements are gathered from the stakeholder.BA and Project Manager participates in this phase. After completing this phase, BRD will be generated.

Requirements Analysis-The requirements are analyzed to understand the scope of the project. Analyzing means the BA will check all the requirements, if he finds convincing requirements then the BA will talk to the concerned stakeholder to clear it, remove the ambiguous requirements

BA will prepare functional requirements. The document which contains the functional requirements is called (FRS). [Functional Requirement Specifications] Technical team will prepare non-functional requirements. The document which contains the non-functional requirements is called (SSD). [Supplementary Support Document] BA will combine FRS and SSD to form SRS. [Software Requirement Specifications]BA will prepare RTM by referring SRS

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’, ADD and solution document will be generated here. [High-level Design Doc.]BA Collaborate with designers, architects, and developers to translate requirements into system design.BA Ensure that the design aligns with the documented requirements and addresses stakeholder needs

Development-

The Development phase include implementation. It involves coding the software based on the design specifications. Programmers or developers are involved in this phase. Here BA acts as a mediator between the development team and the stakeholders.BA clarifies the requirements, check if the development is going on right track or not.BA also participates in scrum meetings.

Testing-

In the testing phase, the software is tested to ensure that it meets the requirements and is free from defects. Testers are involved in this phase Test documents are generated here.BA works with the testing team to ensure that the solution meets the requirements.BA facilitate UAT.BA helps the users to know the functionality of the system and helps them to use the system

Deployment-

Once the software has been tested and approved, it is deployed to the production environment.BA ensures that there is smooth transition from development phase to the production phase

Implementation-

This is the final stage of the waterfall model. It involves running the code for the very first-time introduction phase. Release manager handles this phase.BA will Update documentation and requirements specifications to reflect changes in the system over time Maintenance-Running the code for second time in the production phase is called maintenance. This is done by the support team

Question No 9)- What is conflict management? Explain using Thomas – Kilmann technique

Answer – Conflicts can occur due to various reasons, such as differences in goals, values, personalities, resources, or communication breakdowns.

Conflict is an inevitable part of any workplace. So, it is important to resolve it to promote learning and growth.

Conflict management is nothing but the process of identifying and addressing conflicts in a healthy and constructive manner. It consists of strategies and techniques aimed at resolving disputes, disagreements, or differing perspectives among individuals or groups

By identifying the conflicts efficiently, it will in turn be helpful to reduce negative impact and increase positive impact.

It is a process or skill to find creative ways to handle disagreement. Thomas – Kilmann approach is widely used to recognize the approaches for conflict management

Y axis- assertiveness

X axis- co-cooperativeness

High Assertiveness and High Cooperativeness – Collaboration- means working together to find solution

High Assertiveness and Low Cooperativeness – Competition- means defensive, that is standing for your individual beliefs and trying to win

Low Assertiveness and High Cooperativeness – Accommodation- stakeholder will prioritize their needs over others.

Low Assertiveness and Low Cooperativeness – Avoidance- means ignoring the conflict Assertiveness- the extent to which the person attempts to satisfy his own concerns. Cooperativeness- the extent to which the person attempts to satisfy the other person’s concerns.

Question No 10)- List down the reasons for project failure

Answer – Reasons for project failure are:

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 a very important role in meeting 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.

Continuous change in the requirement-

If the requirements keep on changing frequently, this can also lead to project failure. Because the scope of the project will also keep on changing which will 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 the project. Lack of user involvement. Lack of executive support.

Unrealistic expectations-

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 and goals should be discussed. If there is no proper planning, then the team may face difficulties in addressing the issues or tracking 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 finance.

Question No 11)- List the Challenges faced in projects for BA

Answer –

* Lack of training
* Obtaining sign-off on the requirement
* Change management
* Co-ordination between developers and testers
* Conduct meetings.
* Making sure status report is effective
* Driving clients for UAT completion
* Making sure that the project is going on the right track and delivered as per the timelines without any issues
* Gathering clear and unambiguous requirements can be challenging
* Unable to understand what stakeholders are trying to convey
* Scope creep- change in requirement or scope of the project during the project lifecycle can lead to scope creep
* Managing the stakeholder with conflicting interest can be a difficult task for BA
* BA may face difficulties in understanding the requirements if the domain is not familiar to him
* Poor communication between stakeholders and BA can affect the process of gathering information
* Technical complexity
* Changing business needs or requirements.
* Mindset- Business analysts must be prepared to deal with various difficulties throughout their work, from limitations of the technologies they employ to push back from stakeholders and other team members.

Question No 12)- Write about Document Naming Standard

Answer –

A document numbering standard is a systematic approach to assigning unique identifiers to various documents created and used throughout the development process.

EX-Suppose we have a project with ID “PROJ123”, we are working with a requirements specification document.

Project ID- Proj123 Document type-REQ Version-1.0

Date-2024-10-23

The document identifier could be Proj123-REQ-1.0-2024-10-23.

* Keep file name short and meaningful

ex-/.../orientation/20241023capstone1.pdf

* Avoid unnecessary repetition and redundancy in file names and folder names. Ex-/.../Project/20241023capstone1.pdf
* Simple title case- no separate file name, underscore dashes, or spaces.
* When including a number, use leading zeros to ensure less sort properly, i.e., “001, 002…101” instead of “1, 2…101”.
* Date format should be YYYYMMDD (or YYMMDD) so years of files are sorted in chronological order.
* When including the personal name add surnames instead of first name. Ex-Mishra2024.jpg.
* Order the elements in a file name in the most appropriate way to retrieve the record.
* Avoid using special characters.

Question No 13)- What are the Do’s and Don’ts of a Business analyst

Answer –

|  |  |  |
| --- | --- | --- |
| S No. | Do’s | Don’t |
| 1 | Consult an SME for clarifications on requirements | Never say NO to the client. |
| 2 | Go to the client with a plan mind with no assumptions. Listen carefully and completely until the client is  done, and then you can ask queries. | There is no word as “By default”. |
| 3 | Try to extract maximum leads to the solution from the client himself. | Never imagine anything in terms of GUI. |
| 4 | Concentrate on the important requirements | Don’t interrupt the client when he is giving you the problem. |
| 5 | Question the existence of existence. Question everything | Never try to give solutions to the client straight away with your previous experience and assumptions |
| 6 | Be like a lotus in mud- if a client comes with a fancy requirement, then talk to the PM first. | Never criticize the stakeholder. |

Question No 14)- Write the difference between packages and sub-systems

Answer –

Package-

Collection of components that are not reusable in nature. A package is a grouping and organizing element in which other elements reside, which must be uniquely named. In UML, packages are used in a manner similar to the way directories and folders in an operating system group and organize files.

For example, the project management system may be decomposed into a collection of classes organized into packages as follows.

Ex- Application development companies work on packages.

Subsystem-

Collection of components that are reusable in nature. Recall that a system is an organized collection of elements that may be recursively decomposed into smaller subsystems and eventually into non-decomposable primitive elements.

For example, the project management system may be decomposed into the following: A user interface subsystem responsible for providing a user interface through which users may interact with the system business processing subsystem responsible for implementing business functionality. A data subsystem is responsible for implementing data storage functionality.

Ex- Product development companies work on sub-systems.

While a package simply groups elements, a subsystem groups elements that together provide services such that other elements may access only those services and none of the elements themselves. A subsystem is shown as a package marked with the Subsystem keyword.

Question No 15)- What is camel-casing and explain where it will be used

Answer –

Camel-casing refers to the naming convention of variables, parameters,

or properties. Here, multiple words are combined. In camel-casing, the starting letter of the first word starts with a small letter, and in other words first letter starts with capital letters.

camelCase is used in a programming language to name different files and functions without violating the naming laws of the underlying language. camelCase is also known as medial capitals and Pascal case. The term camelCase is derived from its appearance, which can resemble a camel's back. It is used in many programming languages that don't allow spaces in file names. camelCase enables the creation of names that are more unique and have more meaning for the developer

Ex- first Name, last Name

In BA, camel-casing is used in requirements documentation. In requirement documentation, BA often uses camel-casing to name the entities like use cases, features, user stories like validating Customer Details, calculating Interest Rate, and business rules, which should be satisfied by the system use of camel-casing. While documenting business processes or workflows, camel-casing can be used to individuals in steps. This will help maintain consistency in the document. The database table name also uses camel-casing. Requirement naming- camel casing is used in the requirement document also, to name the functional and non-

functional requirements. Using camel casing in documents helps to maintain consistency in the entire document and increases readability.

CamelCase is a way to separate the words into 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.

Question No 16)- Illustrate Development server and what are the accesses do business analyst has

Answer –

A development server refers to a dedicated environment or server that is used during the software development process. It provides a platform for developers and testers to build, test, and debug applications before they are deployed to a production environment.

A development server is the core tier in a software development environment, where software developers test code directly. It is comprised of the essential hardware, software, and other components used to deploy and test the software underdevelopment, including bulk storage, development platform tools and utilities, network access, and a high-end processor. Upon testing completion, the application is moved either to a staging server or a production/live server

The accesses a BA has are-

Read-only- BAs may be granted read-only access to the development server. This will allow them to view the user interface of the application, navigate through the features, and also, they will be able to observe the behavior of the application.

Limited Access- Depending upon the project needs, the BAs will be granted limited access to the specific modules in the application. Limited Configuration Access-

Means BA has the authority to make changes in certain areas of applications where they have access

Question No 17)- What is Data Mapping

Answer-

The database contains multiple tables in it.

There may come a scenario, where we need to map the data from one table to another.

Data mapping is necessary in cases where we want quick manner.

Data mapping is nothing but a process to establish connection between multiple data sources.

The purpose of data mapping is to ensure that the data is accurately transferred or converted into different format.

The main purpose of data mapping is-

Data integration-While combining the data from different sources, it ensures that the data is properly matched.

Data Migration- While migrating the data from legacy system(source) to the new system(destination), the data elements are mapped accurately into the new system. Required

techniques are applied to covert the data into the format that is required by the new system.

Data Transformation-Data transformation means converting the data from one format to other. In data mapping, data transformation plays very important role which ensures that the data of legacy system(source) is mapped correctly to the data in new system (destination

Question No 18)- 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

An API or application programming interface is a set of rules and tools that allows different software applications to communicate with each other. It defines the method and data formats that applications can use to request and exchange information. APIs provide a secure and standardized way for applications to work with each other and deliver the information or functionality requested without

user intervention.

An API, or application programming interface, is a set of defined rules that enable different applications to communicate with each other. It acts as an intermediary layer that processes data transfers between systems, letting companies open their application data and functionality to external third-party developers, business partners, and internal departments within their companies.

For the above scenario, Establish API communication- set up API communication between your application and other applications to exchange data. Do Data formatting- while sending the data from one application to another, convert the date format from dd-mm-yyyy to mm-dd-yyyy. While receiving the data from other applications, 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.