**Question 1 – Draw a Use Case Diagram for Customer Payment:**

****

**Question 2 - Derive Boundary Classes, Controller classes, Entity Classes:**

**Answer-**

Boundary Class- used to handle the interactions between the system and external actors.

Actor speaks to system and viceversa through boundary

e.g.- Payment option boundary

Card Payment Boundary

( User registration page, login page, Bank Server login page)

Controller Class- acts as intermediaries between boundary and entity class

Whenever user sends a request for something then it always go through controller. Controller is responsible for intercepting the request from boundary(view) and passess it to the model for appropriate action

e.g.- Payment initiated



Card payment

Entity class- represents the core data and business logic of application

It represents enterprise data and business rules. All model classes represented as an entity class

e.g.- Customer

Payment

**Question 3- Place these classes on a three-tier Architecture.**

**Answer-**

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

**Question 4- Explain Domain Model for Customer making payment through Net Banking**

**Answer-**

A diagram of a payment process

Description automatically generated

**Question 5- Draw a sequence diagram for payment done by Customer Net Banking**

**Answer-**

A diagram of a diagram

Description automatically generated

**Question 6 –Explain Conceptual Model for this Case :**

Conceptual modeling is a representation of the business model we have. There are entities, and their relationships among them based on which this is created. This is a Representation of a system that will be created, and this is not a language used to communicate with the technical team.

Entities

In our case study, since the customer is initiating online payment through net banking, the following entities will be there. Bank, Customer, Net banking system of a Bank, Final beneficiary (Customer’s customer).

Relationships

The customer will be initiating payment with multiple beneficiaries, so it will be One to Many

Relationship. The Bank System will interact with the Customer, so it will be a one-to-one relationship.

The bank will be facilitating payment for multiple customers; hence it will be One to May Relationship.

Based on the above Entities and Relationships we will be creating an Entity Relationship diagram, which will be used as a representation to take sign-off from the client.

**Question 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 class is known about all the data that is needed to be displayed.

It only represents the data of an application.

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 give the requested data.

All the model classes are nothing but the entities.

View-

View is nothing but what client sees on the screen.

Generates UI for the user.

It represents the presentation of the application.

View refers to the model.

Controller-

Takes input from the user/ client.

It interacts with the model and view.

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.

Combination of two actor and one use case results in two boundary class.

Combination of three actor and one use case results in three boundary class.

2. Use case will result in controller class.

3. Each actor will result in one entity class.

Consider the example of Online shopping application with the following usecase:

Model Classes- represents the data objects in the system

Customer

Payment

Net Banking

Card

Cash

View Classes- represents the user interface

Login View

Payment Option View

Net Banking View

Bank Selection View

Credentials View

Payment Amount View

Payment Confirmation View

Logout View

Controller Classes- involves the action handling the user interactions or system operations.

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.

These classes interact directly with the user.

Presentation layer is responsible for displaying information and also receiving the input from the user.

Application Layer-

This layer is nothing but business logic.

Model and controller are inside this layer.

Controller handles the user input, process the request and co-ordinates interaction between the model and view.

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.

**Question 8 – Explain BA contributions in project (Waterfall Model–all Stages) :**

**Answer-**

A waterfall model is very old and traditional model in IT industries. It is a progressive implementation of the projects which is divided into different phrases of SDLC.

Business Analyst will verify if the product is delivered as per the requirements, and it is meeting the business need. Maintenance: Once the implementation is done the team has to give support by installing patches, handling change requests, etc.

Stages in Waterfall Model are as follows:

Requirements Gathering-

In this phase, all the requirements are gathered from the stakeholder.

BA and Project Manager participates in this phase.

In this phase, BRD will be generated.

Requirements Analysis-

The requirements are analysed to understand the scope of the project.

BA will prepare functional requirement (FRS).

Technical team will prepare non-functional requirement (SSD).

BA will combine FRS and SSD to form SRS.

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.

HDD, ADD and solution document will be generated here.

Development-

The Development phase include implementation.

It involves coding the software based on the design specifications.

Programmers or developer are involved in this phase.

Testing-

In the testing phase, the software is tested as a whole to ensure that it meets the requirements and is free from defects.

Testers are involved in this phase.

Test documents are generated here.

Deployment-

Once the software has been tested and approved, it is deployed to the production environment.

Implementation-

This is the final stage of waterfall model.

It involves running the code for the very first time in production phase.

Release manager handles this phase.

Maintenance-

Running the code for second time in the production phase is called maintenance.

This is done by support team.

**Question 9-** What is conflict management? Explain using Thomas – Kilmann technique

**Answer-**

Conflict is an inevitable part of any workplace.

So it is important to resolve it to promote learning and growth.

Conflict management is a practise of identifying the conflicts efficiently which 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 the disagreement.

HIGH

Collaboration

Competition

Compromise

Assertiveness

Accommodation

Avoidance

LOW

HIGH

cooperativeness

High Assertiveness and High Cooperativeness - Collaboration

High Assertiveness and Low Cooperativeness - Competition

Low Assertiveness and High Cooperativeness - Accommodation

Low Assertiveness and Low Cooperativeness - Avoidance

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 persons concerns.

Competing- means defensive, that is standing for your individual beliefs and trying to win.

**Question 10 – List down the reasons for project failure:**

**Answer-**

* Improper Requirement Gathering
* Continuous change in requirements
* Lack of user involvement
* Lack of executive support
* Unrealistic expectations
* Improper/Poor Planning

**Question 11 – List the Challenges faced in projects for BA :**

**Answer**

A BA is responsible for multiple tasks at the same time. From managing the projects, maintaining client relationships, interacting with stakeholders, and managing project deadlines, Business Analysts got a lot on their plate. Below are the challenges faced by business analyst.

* Lack of training
* Obtaining Sign-off on requirement
* Coordination between developers and testers
* Conducting meetings
* Preparing effective status reports which satisfies all the project stakeholders.
* Driving client for UAT completion
* People Management (Coordination between different people and different teams)

**Question 12 –Write about Document Naming Standards :**

**Answer-**

File Naming Standards are used to save the file with particular name or format. This is important in Sharing and keeping track of data files. Following are the best standards in Naming Convention.

1. It should be Named Consistently.

2. File names should be short (<25 characters)

3. Avoid special characters or spaces in a file name.

4. Use Capital and Underscores instead of spaces or slashes.

5. Use date format as per ISO 8601: YYMMDD

6. Include a version number.

7. Write down naming convention.

We must consider following naming conventions:

Date of Creation

Short Description

Work

Location

Project name or number

Sample

Analysis

Version Number

**Question 13 – What are the Do’s and Don’ts of a Business analyst :**

**Answer-**

As a Business Analyst we have to follow certain Rules and Regulations, which help us in improving our productivity as well as quality of work. Following are the Do’s and Don’ts of Business Analyst.

Never Say No to client: While client is saying something, we must never interrupt and say NO to client. As client is expecting from us to listen to him and provide the solution to his problems.

Never Imagine anything in terms of GUI: We must not imagine the requirements by just seeing the graphical representation ask right question to client and get clarity on the requirements. I.E. As login page may appear same for all the websites but Functionality is different.

Question Everything: We must not feel bad to ask questions and should get clarity from client. We can ask the questions till we get clarity from client. Sometimes client may not tell us the entire requirement until we probe them.

Consult an SME for clarifications in requirements: If requirement is not clear and you need more clarity on the requirement, then we can discuss with SME. And ensure to document the requirements what you discussed with SME and get approval from solution owner.

There is no Word called as “By Default”: In every requirement we have some Unique specifications, hence we cannot consider anything as By Default. Also, we should never Imagine something as by default by listening to client’s requirements, we must either acknowledge it or probe it further.

**Question 14 - Write the difference between packages and sub- systems :**

**Answer-**

Below are few differences between package and sub systems:

Packages- it is a group of classes or use cases that are used to organize model elements.

Packages can be nested within other packages.

These are used as containers to organize elements.

It is very useful to represent system architecture.

package

Subsystems- it is logical grouping of related components.

It is collection of classes, packages, libraries and other sub systems that work together to deliver a specific set of functionalities.

**Question 15 - What is camel-casing and explain where it will be used:**

**Answer**-

The “camel case” is the practice of writing sentences so that each word or abbreviation in the middle of the sentence begins with a capital letter, without spaces or punctuation. For example: “iPad” and “eBay”

The camel case is normally used for variable names in computer programming. Some programming styles prefer uppercase and lowercase letters with the first letter capitalized, others do not. The Camel case is distinct from the Title Case (or Capitalized Case), which capitalizes all words, but maintains the spaces between them.

**Question 16 -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.

**Question 17 – 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.

**Question 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.

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- set up API communication between your application and other application to exchange data.

Do Data formatting- while sending the data from one application to other, 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.