**Capstone Prep 3- part 1/2**

**Case Study 1 (Q1-Q6  24 Marks)**

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

 **Q1. Draw a Use Case Diagram - 4 Marks**

**Answer :**

**Q2. Derive Boundary Classes, Controller classes, Entity Classes. - 4 Marks**

**Answer :**

**Boundary Class (All use cases)**

Combination of 1 actor and use case in one boundary class

Combination of 2 actor and use case in two boundary class

Combination of 3 actor and use case in three boundary class

Actors should be primary actors

**Controller class (handles user input and process the data)**

Use cases will be considered as Controller

**Entity Class ( All actors)**

Each actor will be considered as one entity

|  |  |
| --- | --- |
| **Boundary Class** | **Customer Registration****Customer login****BankServer Login****Customer logout****BankServer logout**  |
| **Controller class** | **Registration Controller****Login Controller****Payment Controller****Credentials Controller****Netbanking Controller****Email Controller****Logout Controller** |
| **Entity Class** | **Customer****Bank server****Cash****Card****Net banking**  |

**Q3. Place these classes on a three tier Architecture. - 4 Marks**

**Answer :**

Application layer : Customer Registration

 Customer login

 Bank server login

Business Logic Layer : Customer

 Bank server

Data layer : Customer

 Bank server

 Cash

 Card

 Net banking

**Q4. Explain Domain Model for Customer making payment through Net Banking - 4 Marks**

**Answer :** Domain modelling is also known as conceptual modelling. A conceptual model depicts the concepts that are easily identifiable in the problem description

****

 **Q5. Draw a sequence diagram for payment done by Customer Net Banking - 4 Marks**

**Answer :**

 **Q6. Explain Conceptual Model for this Case - 4 Marks**

**Answer :** A conceptual model helps in understanding the key concepts, their relationships and overall structure of the net banking system. It serves foundation for designing the dataschema, 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 of this node is to protect the privacy and confidently of customer data in the context of net banking
4. Technology awareness : the significance of this node is that customers should be aware and comfortable with the underlying technology used in net banking services
5. Bank : this node represents a service provider responsible for offering net banking services
6. Security and privacy : the bank should adapt the security policies which will help the customers to keep their data related to their transaction secure and private.
7. Infrastructure : the component suggests that the underlying technological infrastructure, including hardware and software systems plays a important role in enabling net banking services.



**Q7. What is MVC architecture? Explain MVC rules to derive classes from use case diagram and guidelines to place classes in 3-tier architecture - 8 Marks**

Answer : MVC is a design pa ern 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 is needed 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 give the requested data. All the model classes are nothing but the entityes. Model classes are represented as enti ty 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 query method. View does not depend upon application logic. View class 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 for anything, that request first goes to the controller.

Controller works on the users 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

**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.

**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, processes the request, and coordinates 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 interact with the database, files, and other data sources.

**Q8. Explain BA contributions in project (Waterfall Model – all Stages) – 8 Marks**

**Answer :**
Waterfall model follows sequential approach.

Waterfall model is useful in the situation where the project requirements are well defined and the project goals are clear.
In this model each phase is completed entirely and then only moved to the next phase.
Waterfall model relies on documentation to ensure that the project is well defined and project team is working toward clear goals.
Ones that particular 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 analysed to understand the scope of the project.

Analysing means the BA will check all the requirements, if he finds conflicting requirements then the BA will talk to the concerned stakeholder to clear it, remove the ambiguous requirements.

BA will prepare functional requirement the document is called as FRS

Technical team will prepare non functional requirement the document is called as SSD

BA will combine FRS and SSD to form SRS

BA will prepare RTM by referring SRS

**Design** –

After the requirement is clear 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 [High level Design Doc]

BA contribute with designers, architect and developers to translate requirements into system design

BA ensures that the design aligns with the documented requirements and stakeholder needs

**Development –**

The Development phase include implementation.

It involves coding the software 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.

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 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.

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 also 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 waterfall model.

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

**Q9. What is conflict management? Explain using Thomas – Kilmann technique – 6 Marks**

**Answer :** Conflict 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 the process of identifying and addressing conflicts in a healthy and constructive manner. It consist of stratergies and techniques aimed at resolving disputes, disagreement or differing perspectives amongst individuals or groups.

**Thomas – Kilmann technique is widely used to recognize the approaches foe conflict management**

****

**Y axis assertiveness X axis Co- operativeness**

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 beleifs and trying to win

Low assertiveness and high Cooperativeness – Accommodation - Stakehloder will prioritize their needs over others

Low Assertiveness and low Cooperativeness – Avoidance – means ignoring the conflict

**Q10. List down the reasons for project failure – 6 Marks**

**Answer : Improper Requirement gathering** -- If requirements are not gathered properly it can lead to project failure.

**Lack of stakeholder involvement –** A project can fail if the stakeholders are not participating in the process. The stakeholders' input and feedback play a 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 the 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 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, goals should be discussed.

**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 – 6 Marks**

**Answer** :

* Lack of training
* Obtaining sign-off on requirements
* Change Management – with respect to cost and timelines
* Coordination between developers and testers
* Conducting Meetings
* Making sure status reporting is effective
* Driving clients for UAT completion
* People Management (coordinating with different people and different teams)
* Overall making sure project health is in good shape and delivered as per the time-lines without any issues.
* Making sure that the project is going on right track and delivered as per the timelines without any issues.
* Gathering clear and unambiguous requirements can be challenging.
* Unable to understand what stakeholder is 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 stakeholder and BA can affect the process of gathering the information.
* Technical complexity

**Q12. Write about Document Naming Standards – 4 Marks**

Answer : All documents will be named using some standards like

[projectId][document type]V[x]D[y]. ext

Example : PQ789BRDV1V2.docx

 PQ86BRDD1.2.docx

**Q13. What are the Do’s and Don’ts of a Business analyst – 6 Marks**

**Answer :**

Never say NO to Client

There is NO word called as "BY DEFAULT"

Never imagine anything in terms of GUI

Question the existence of existence / question everything in the world
ex: what client gives is not always correct

 Consult an SME for Clarifications in Requirements

 Every Problem of Client is unique. No two problems of different Client are same. May be the approach, technology, place of use, local laws may be varied to make them (Problems) to 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 your Queries. Please do not interrupt the Client, when he/She is giving you the problem. Maximum Try to extract the leads to Solution from the Client itself. Never try to give Solutions to Client straight away with your previous experience and assumptions. Try to concentrate on the important and truly required Requirements. Don’t be washed away by add on Functionalities or don’t imagine solutions on Screen basis.

**Q14. Write the difference between packages and sub-systems – 4 Marks**

**Answer :**

|  |  |
| --- | --- |
| **Packages** | **Subsystems** |
| Group of classes or usecases that are used to organize model elements | Logical grouping of related components |
| Collection of components which are not reusable in nature | Collection of components which are reusable in nature |
| Application development companies work on packages | Product development companies work on subsystems |

**Q15. What is camel-casing and explain where it will be used- 6 Marks**

**Answer :** Camel Casing is naming convention in computer programming. It is used for naming variables, functions and identifiers. In camel case first words starts with a lower case letter and each subsequent word begins with a uppercase letter Example : validateCustomerDetails

Camel casing is used in requirements documentation. To name entities like use case, features, user stories. The database tables name also uses camel- casing.

Requirement naming – To name functional and non- functional requirements

**Q16. Illustrate Development server and what are the accesses does business analyst has? -6 Marks**

**Answer** : A development server refers to a dedicated enviorment that is used during the software development process. It provides platform for the developers and testers to build test and debug the application.

The accesses a BA has are –

Read only – BA’s may be granted with the readonly access to the development server. This will allow them to view the user interface of the application, navigate through the featues and also they will be able to observe the behaviour 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 – Means BA have the authority to make changes in certain areas of application where they have the access.

**Q17. What is Data Mapping 6 Marks**

**Answer :** Data mapping is the process of connecting data from one source to another. It is like creating a guide or map that shows how data in one place is corresponds to data in another place. This is especially important when moving data between different systems or databases to snsure that the data stays consistent and accurate.

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 convert data into the format that is required by the new system

Data Transformation : Converting the data from one format to another, Data transformation plays important role which ensures that the daya of legacy system (source) is mapped correctly to the data in new system(destination)

**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 10 Marks**

**Answer :** API stands for Application Programming Interface is a set of protocols, tools, and definitions that allow different software applications to communicate and exchange data with each other. An API acts as a bridge between two systems, enabling them to send and receive data in a structured way — without needing to understand the internal logic of each other.

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 data 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 data remains in a valid format.