Capstone Project 5 Answer Sheet

**Question 1- Draw a Use Case Diagram.**

**Ans :**



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

**Ans :**

|  |  |
| --- | --- |
| **Boundary class** | Customermakepayment boundary class |
| Customercashpayment boundary class |
| Customerwalletpayment boundary class |
| Customernetbankingpayment boundary class |
| Customercardpayment boundary class |
| MakepaymentapplicationDBserver boundary class |
| CashapplicationDBserver boundary class |
| WalletapplicationDBserver boundary class |
| NetbankingapplicationDBserver boundary class |
| CardapplicationDBserver boundary class |
| **Controller class** | Makepayment controller class |
| Cash controller class |
| Wallet controller class |
| Netbanking controller class |
| Card controller class |
| **Entity class** | Customer entity class |
| ApplicationDBserver entity class |

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

**Ans:**

Application Layer: Customermakepayment boundary class

Customercashpayment boundary class

Customerwalletpayment boundary class Customernetbankingpayment boundary class Customercardpayment boundary class

Makepayment controller class Cash controller class

Wallet controller class Netbanking controller class Card controller class

CashapplicationDBserver boundary class

|  |
| --- |
| Business Layer: MakepaymentapplicationDBserver boundary class WalletapplicationDBserver boundary class NetbankingapplicationDBserver boundary class CardapplicationDBserver boundary class |
| Database Layer:Customer entity class ApplicationDBserver entity class |

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

**Ans:** Domain model is explaining a part of the process.



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

**Ans:**



**Question 6- Explain Conceptual Model for this Case**

**Ans :**

Conceptual Model means explaining the entire process. Conceptual model means representation of how payment is handled within the organization or system.

Actors: Customer: The end users will interact with the system to make payment.

Payment flow:

1. Initiation- customer will initiate the payment process [ex: cash, card, wallet and net banking]
2. Authorization- payment gateway verify the transaction validity and sufficient funds.
3. Processing- The request will be processed.
4. Error or exception- Insufficient funds or incorrect details entered.
5. Confirmation- Customer will get a confirmation notification

**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**

**Ans :**

MVC architecture has 3 classes, such as boundary class, controller class and entity class.

* **Boundary class:** How the system will interact with the end user. Combination of actor and use case(actions) will result in boundary class.
* **Controller class:** It controls the flow, which way data should flow based on inputs whether to store data or the display. Each use case will be controller class.
* **Entity class:** These are the headings of tables; this class connects to database. Each actor will be entity class.

Guidelines to place classes in 3-tier architecture:

* 1. Place all entity class in DB layer
	2. Place primary actor associated boundary class in application layer
	3. Place controller class in application layer
	4. If governing body influence or reusability is there with any of remaining boundary class, place them in business logic layer else place them in application layer.

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

**Ans:** BA plays an important role in all stages of waterfall.

**Stage 1 Requirement gathering and analysis:** BA are involved from initiate phase. BA are responsible for gathering the preparing the Business Requirement Document (BRD).

**Stage 2 Design:** During this phase the system architect will design the system based on the requirement gathered and documented by the BA. BA can assist the system architect by clarifying doubts related to the requirements.

**Stage 3 Development:** Developers create the product based on the requirement document prepared by BAs. Developers can clarify any doubts regarding the requirement with BA

**Stage 4 Testing:** BA helps the testing team to understand the requirements, ensure proper functional test cases are built. Bas review whether the test cases cover the entire functionality

**Stage 5 Deployment:** Once the code is developed and tested, it is ready for deployment in the production environment. Bas verify that the product meets the specified requirements and fulfills business needs

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

**Ans:** Conflict management means resolving disagreements in a constructive way. It involves identifying and handling conflicts fairly and efficiently to minimize negative impacts and increase the likelihood of a positive outcome. Thomas – Kilmann technique is a graphical representation.



* Low co-operation and low knowledge will always result in avoidance.
* Both co-operation and knowledge will collaborate.
* High knowledge with low co-operation will result in competition.
* Both co-operation and knowledge high will collaborate, however if there are any other 3 cases always bring them to compromise to resolve the conflict.

**Question 10- List down the reasons for project failure**.

**Ans :** Reasons for project failure are:

1. Improper requirement gathering
2. Continuous change in requirements- Client not giving correct requirement or BA not able to understand.
3. Lack of user involvement-User should be available to give requirements, should be able to make time.
4. Unrealistic expectations- To get the project the presales team set unrealistic expectations. Client may lose interest if those are not fulfilled.
5. Lack of executive support- Ex: HR not able to map the developers to project

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

* Lack of domain knowledge
* Changing Business Needs or Requirements
* Lack of active participation from stakeholders
* Pressure to meet tight deadlines
* Effective communication

**Question 12- Write about Document Naming Standards**

**Ans:** Project should have

ProjID- DocName-Version(v)[]-Document.ext Examples: PQ786-BRD-V1-D1.docx 1.1

PQ786-BRD-V1-D2.docx 1.2

PQ786-BRD-V1-D3.docx 1.3

.

.

PQ786-BRD-V2-D1.docx 2.1

Always check V for latest version and document for latest doc.

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

**Ans:**

* Never say NO to client: Even if the requirement is out of scope, never say no to client
* There is no word called “by default”: Never 2 projects are same, there will be some
* difference. Treat every project as a new project, gather requirements from scratch.
* Never think in terms of GUI: Never think graphically about what the application or the project will look like. That is the job of Ui/Ux.
* Question the existence to the existence: If client gives us requirements, check if it can be fulfilled through 5W 1H concept.

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

**Ans :**

**Package:** Collection of components which are not reusable in nature are termed as packages. Components are a collection of class and class are a collection of similar objects.

**Subsystem:** Collection of components which are reusable in nature are subsystem.

Product development companies work on subsystem and application development companies work on packages.

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

**Ans :**

Camel casing is the practice of writing phrases without spaces or punctuation and with capitalized words. The format indicates the first word starting with either case, then the following words having an initial uppercase letter.

Examples: turnLeftAndThenRight

It can be used for variable and method name, language such as python, programming, etc

**Question 16- Illustrate Development server and what are the accesses does business analyst has?**

**Ans:**

Development server belongs to IT companies. A development server is a server environment used by developers to build, test, and improve their software before it is pushed to production. It's essentially a safe space where code can be modified, tested, and debugged without affecting the live application or end-users.

As a business analyst, he will have read only access, collaborative access and limited configuration access.

**Question 17- What is Data Mapping**

**Ans:** Data mapping is the process of matching fields from one database to another. Data mapping involves linking two different sets of data together. It’s like creating a bridge between two databases. It's the first step to facilitate data migration, data integration, and other data management tasks. This reduces the potential for errors, helps standardize your data and makes it easier to understand your data. In each of these cases, as the business analyst, you want to use your knowledge of the business process to identify how to handle that mapping issue or collaborate with the business stakeholders to really discover their needs and their desires and what they're willing to invest in potential data cleanup.

**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**

**Ans:**

An application programming interface is a way for two or more computer programs or components to communicate with each other. It defines a set of rules, protocols, and functions that allow these programs to interact with each other. The purpose of APIs is to allow programs to exchange data, features, and functionality.

Your application (1) needs to communicate with another application (2) that uses a different date format.

App 1 sends request to app 2 or receives data from it via APIs.

App 1 accepts data in the dd-mm-yyyy format, while app 2 provides data in the mm-dd- yyyy format

To ensure seamless integration, you’ll need to converst the dates between these

formats.

Parse the incoming data (in mm-dd-yyyy format) into a standard internal representation

Convert this internal representation to the desired dd-mm-yyyy format for further processing within App 1

When App 1 sends data to App 2, it should:

Convert the date from the dd-mm-yyyy format to the mm-dd-yyyy format before including it in the API request.

Ensure that the API request body or query parameter contains the correctly formatted date.

The system connected through this relation will convert the shared data in the format of the target data. While sharing the data it is validated.