BHARATI VIDYAPEETH'S
INSTITUTE OF COMPUTER APPLICATIONS & MANAGEMENT (BVICAM)
(Affiliated to Guru Gobind Singh Indraprastha University, Approved by AICTE, New Delhi)
A-4, PaschimVihar, Rohtak Road, New Delhi-110063, Visit us at: http://www.bvicam.in/

Course Code: MCA-164   Course Name: Object Oriented Software Engineering Lab

Practice Questions (Practical)

Q1.	An automated teller machine (ATM) or the automatic banking machine (ABM) is a banking subsystem (subject) that provides bank customers with access to financial transactions in a public space without the need for a cashier, clerk, or bank teller. Customer (actor) uses bank ATM to Check Balances of his/her bank accounts, Deposit Funds, Withdraw Cash and/or Transfer Funds (use cases). ATM Technician provides Maintenance and Repairs. All these use cases also involve Bank actor whether it is related to customer transactions or to the ATM servicing. On most bank ATMs, the customer is authenticated by inserting a plastic ATM card and entering a personal identification number (PIN). *Customer Authentication* use case is required for every ATM. *ATM Technician* maintains or repairs Bank ATM. *Maintenance* use case includes *Replenishing* ATM with cash, ink or printer paper, *Upgrades* of hardware, firmware or software, and remote or on-site *Diagnostics*. Model the above scenarios through the following diagrams: i.　　Use case Diagrams ii.　　Class Diagrams iii.　　Object Diagram iv.　　Activity Diagram v.　　Statechart Diagram
Q2.	**Point of Sale (POS) Terminal** or Checkout. A retail POS system typically includes a computer, monitor, keyboard, barcode scanners, weight scale, receipt printer, credit card processing system, etc. and POS terminal software. Checkout **use case** involves Customer, Clerk and Credit Payment Service **actors** and **includes** scanning items, calculating total and taxes, payment use cases. Checkout use case requires Customer actor, hence the 1 multiplicity of Customer. Clerk can only participate in a single Checkout use case. Credit Payment Service

can participate with many Checkout use cases at the same time. Checkout use case may not need Credit Payment Service (for example, if payment is in cash), thus the 0..1 multiplicity.

Checkout use case is an example of a large and complex use case split into several use cases each describing some logical unit of behavior. Note, that including use case becomes incomplete by itself and requires the included use cases to be complete.

Payment use case is represented using **generalization** relationship. It means that only one specific type of payment is accepted - either by cash, or by credit, debit, or with check. An alternative to such representation could be to use **include** relationship so that not just single but several forms of payment could be accepted from the same client during checkout.

Model the above scenarios through the following diagrams:

i.   Use case Diagrams

ii.   Class Diagrams

| Q3. | An **Online Public Access Catalog** (**OPAC**) is an e-Library website which is part of **Integrated Library System** (ILS), also known as a **Library Management System** (LMS), and managed by a library or group of libraries.

Patrons of the library can search library catalog online to locate various resources - books, periodicals, audio and visual materials, or other items under control of the library. Patrons may reserve or renew item, provide feedback, and manage their account. Model the above scenarios through a Use case Diagram. |

| Q4. | Credit Card Processing System (aka Credit Card Payment Gateway) is a subject, i.e. system under design or consideration. Primary actor for the system is a Merchant's Credit Card Processing System. The merchant submits some credit card transaction request to the credit card payment gateway on behalf of a customer. Bank which issued customer's credit card is actor which could approve or reject the transaction. If transaction is approved, funds will be transferred to merchant's bank account.

Authorize and Capture use case is the most common type of credit card transaction. The requested amount of money should be first authorized by Customer's Credit Card Bank, and if approved, is further submitted for settlement. During the settlement funds approved for the credit card transaction are deposited into the Merchant's Bank account.

In some cases, only authorization is requested and the transaction will not be sent for settlement. In this case, usually if no further action is taken within some number of days, the authorization expires. Merchants can submit this request if |

| | | they want to verify the availability of funds on the customer's credit card, if item is not currently in stock, or if merchant wants to review orders before shipping.

Capture (request to capture funds that were previously authorized) use case describes several scenarios when merchant needs to complete some previously authorized transaction - either submitted through the payment gateway or requested without using the system, e.g. using voice authorization.
Credit use case describes situations when customer should receive a refund for a transaction that was either successfully processed and settled through the system or for some transaction that was not originally submitted through the payment gateway.

Void use case describes cases when it is needed to cancel one or several related transactions that were not yet settled. If possible, the transactions will not be sent for settlement. If the Void transaction fails, the original transaction is likely already settled.

Verify use case describes zero or small amount verification transactions which could also include verification of some client's data such as address. Model the above scenarios through a Use case Diagram. |
|---|---|---|
| Q5. | | Website security requirements mandate separation of administrative interfaces from common functions provided to users. This segregation, for example, is required by the Sarbanes Oxley (SOX) in US and is strongly recommended by ISO 17799.
System should have separate applications for administrators and for common users. It is recommended by OWASP Guide 2.0 that website administration applications should not be accessible from the internet without going through some management networks, e.g., via a strongly authenticated VPN or from a trusted network operations center.
Except for administrators, some part of the administrative interfaces should be also available to the Help desk staff, as they need to be able to assist customers having issues while using the customer-oriented website.
Top level use case diagram below shows some administrative functions that administration website could provide.
Two actors using administrative interfaces are Website Administrator and Help Desk. Help Desk uses a subset of functions available to the Website Administrator. All top-level use cases shown are abstract as each represents some group or "package" of administrative functionality. **Manage User Groups** abstract use case is specialized by **Create Group**, **Update Group**, and **Delete Group** use cases. The idea is that website administrator could create different user groups, for example, having different privileges or options, and later some user groups could be modified or even deleted. User management use cases are available both to the Website Administrator and to the Help Desk. There is standard user CRUD (Create, Retrieve/Find, Update, Delete) functionality set. |

	Two other use cases, Lock User and Unlock User are specific to website security. For example, if during some predefined period of time there were several unsuccessful login attempts using wrong user password, user account should be locked for some predefined time to prevent possible brute force password guessing attack. This locking and unlocking is usually done automatically by intrusion detection or website authentication subsystem, but this functionality needs to be available in the manual mode too, just in case. For example, some user might call and ask to lock his or her account. Model the above scenarios through a Use case Diagram.
Q6.	Case study: Online Shopping Paragraphs Corporation sells books and CDs using through online shopping. The customer adds items to her shopping cart. She may remove items or go to the check-out to make her purchases at any time. The customer reviews her purchases, chooses a payment method and pays. A sales employee at Paragraphs Corporation gets the order and purchase confirmation from the system, and sends the electronic order to the warehouse. The warehouse employee updates the order status. The customer may check the order status. Model the above scenarios through a Use case Diagram and Sequence Diagram.
Q7.	Develop an activity and the corresponding state diagram for booking a room in a hotel scenario. Include activities like checking room availability, selecting the room type, entering guest details, and confirming the reservation.
Q8.	A company wants to develop a new software system to manage its inventory. They need a system that can track items, suppliers, purchase orders, and sales orders. Create a UML class diagram that represents the main entities and their relationships in this system. Include attributes and relationships between the classes.
Q9.	A university wants to create a system to manage student enrollment. The system should keep track of courses, students, and instructors. Each course has a title, a course code, and is taught by one instructor. Each student can enroll in multiple courses. Create a UML class diagram and use case diagram for this system. Include classes for courses, students, and instructors. Also include attributes and relationships between the classes. Imagine all the f
Q10.	A car rental company wants to build a software system to manage its rental fleet. The system needs to keep track of available cars, rental reservations, and customer information. Each rental reservation is associated with a customer and a specific car. Create a UML class diagram for this system.
Q11.	A car rental company wants to build a software system to manage its rental fleet. The system needs to keep track of available cars, rental reservations, and customer information. Each rental reservation is associated with a customer and a specific car. Create a UML sequence diagram to elaborate a rental reservation scenario.
Q12.	A car rental company wants to build a software system to manage its rental fleet. The system needs to keep track of available cars, rental reservations, and customer information. Each rental reservation is associated with a customer and a specific car. Create a UML use case diagram to elaborate the functionality of

	this system.
Q13.	A hospital wants to develop a system to manage patient records. The system should keep track of patient information, medical history, appointments, and doctors. Each patient has a unique ID and can have multiple appointments. Create a UML class diagram for this system.
Q14.	A software development company wants to create a system to manage its projects. The system should keep track of project details, team members, tasks, and deadlines. Each project has a name, a start date, and a deadline. Create a UML class diagram for this system.
Q15.	Describe an activity diagram and the corresponding state diagram for the placing an order case in an online shopping system. Include activities such as browsing products, adding items to the cart, entering shipping information, and making payment.
Q16.	Create an activity diagram and the corresponding sequence diagram for the use case: borrowing a book from a library. Include activities like searching for a book, checking book availability, issuing the book, and updating the library records.
Q17.	Design an activity diagram and the corresponding state diagram for performing an ATM transaction. Include activities such as inserting the ATM card, entering the PIN, selecting the transaction type (e.g., withdrawal, balance inquiry), and completing the transaction.
Q18.	Illustrate an activity and the corresponding state diagram for student registration in a university. Include activities such as course selection, adding/dropping courses, fee payment, and generating the registration confirmation.
Q19.	Design a component and the corresponding deployment diagram for an e-commerce system. You may include components such as customer interface, product catalog, shopping cart, payment gateway, and order processing.
Q20.	Develop a component and the corresponding deployment diagram for a social media platform. You may include components for user profile, friend network, news feed, messaging system, and notification service.
Q21.	Design a deployment and the corresponding component diagram for a travel booking system. You may include nodes for web server, application server, database server, and client devices (e.g., desktop, mobile).
Q22.	Develop a use case diagram for a hotel reservation system. Actors may include guest, receptionist, and manager. Use cases may include making a reservation, checking in/out, managing bookings, and handling payments.
Q23.	**Develop a use case diagram for a Chatbot System detailed below:** **Actors:** User, Chatbot **Use Cases:** Interact with Chatbot: The user can initiate a conversation with the chatbot. Ask Question: The user can ask questions to the chatbot. Provide Response: The chatbot provides responses to user questions.

	Provide Help: The chatbot can offer help or suggestions based on user queries. End Conversation: The user can end the conversation with the chatbot. **Extensions:** View Chat History: Both the user and the chatbot can view the history of the conversation. Manage User Profile: The chatbot can assist in managing user profiles, if applicable. Update Knowledge Base: The chatbot administrator can update the knowledge base used by the chatbot. **Include Relationship:** External Knowledge Base: The chatbot uses an external knowledge base to provide accurate responses. User Profile: The chatbot may utilize user profiles to offer personalized responses. **Extend Relationship:** Search Feature: The chatbot can utilize a search feature to find answers to user questions not covered in its knowledge base. Feedback Mechanism: The chatbot can have a feedback mechanism to improve its responses over time based on user inputs.
Q24.	**Develop a use case diagram for a smartwatch as detailed below:** **Actors:** User, Smartwatch **Use Cases:** Display Time: The smartwatch displays the current time. Set Alarm: The user can set an alarm on the smartwatch. Track Fitness Activity: The smartwatch can track the user's fitness activity, such as steps taken, distance walked, and calories burned. Receive Notifications: The smartwatch can receive notifications from a connected smartphone, such as calls, messages, and app alerts. Control Music: The user can control music playback on a connected device using the smartwatch. Check Weather: The smartwatch can display the current weather forecast. Use Timer/Stopwatch: The user can use the smartwatch as a timer or stopwatch. Connect to Smartphone: The smartwatch can connect to a smartphone via Bluetooth to access additional features. **Extend Relationship:** GPS Navigation: The smartwatch can provide GPS navigation assistance to the user. Health Monitoring: The smartwatch can monitor the user's health metrics, such as heart rate and sleep patterns. Voice Commands: The smartwatch can accept voice commands from the user for hands-free operation. **Include Relationship:** Fitness Goals: The smartwatch can include fitness goals set by the user to track

	progress. App Integration: The smartwatch can include integration with third-party apps for additional functionality.
Q25.	**Develop a suitable use case diagram for a smart car operation:** **Actors:** Driver, Smart Car System **Use Cases:** Start Engine: The driver can start the smart car's engine. Stop Engine: The driver can stop the smart car's engine. Lock Doors: The driver can lock the smart car's doors. Unlock Doors: The driver can unlock the smart car's doors. Drive: The driver can drive the smart car. Park: The driver can park the smart car. Navigate: The smart car can provide navigation assistance to the driver. Monitor Environment: The smart car can monitor its environment using sensors (e.g., cameras, radar) to detect obstacles and other vehicles. Communicate with Driver: The smart car can communicate important information to the driver (e.g., traffic alerts, maintenance notifications). **Extend Relationship:** Autonomous Driving: The smart car can operate autonomously in certain situations, such as highway driving. Remote Control: The driver can control the smart car remotely, such as parking it using a smartphone app. **Include Relationship:** Emergency Services: The smart car can include the ability to contact emergency services in case of an accident or breakdown. Maintenance Alerts: The smart car can include maintenance alerts for servicing.
Q26.	Illustrate through a suitable sequence and collaboration diagram showing how the customer interacts with the online shopping system components to complete a purchase.
Q27.	Illustrate through a suitable sequence and collaboration diagram to demonstrate how the user's message is processed and distributed to the social media platform components.
Q28.	A patient record and scheduling system in a doctor's office is used by the receptionists, nurses, and doctors. The receptionists use the system to enter new patient information when first-time patients visit the doctor. They also schedule all appointments. The nurses use the system to keep track of the results of each visit including diagnosis and medications. For each visit, free form text fields are used captures information on diagnosis and treatment. Multiple medications may be prescribed during each visit. The nurses can also access the information to print out a history of patient visits. The doctors primarily use the system to view patient history. The doctors may enter some patient treatment information and prescriptions occasionally, but most frequently they let the nurses enter this information. -- Each patient is assigned to a family. The head of family is responsible for the person with the primary medical coverage. Information about

	doctors is maintained since a family has a primary care physician, but different doctors may be the ones seeing the patient during the visit. Identify the top-level functional requirement for the clock, and model it with a use case diagram.
Q29.	Suppose we want to develop software for an alarm clock. The clock shows the time of day. Using buttons, the user can set the hours and minutes fields individually, and choose between 12 and 24-hour display. It is possible to set one or two alarms. When an alarm fires, it will sound some noise. The user can turn it off, or choose to 'snooze'. If the user does not respond at all, the alarm will turn off itself after 2 minutes. 'Snoozing' means to turn off the sound, but the alarm will fire again after some minutes of delay. This 'snoozing time' is pre-adjustable. Identify the top-level functional requirement for the clock, and model it with a use case diagram.