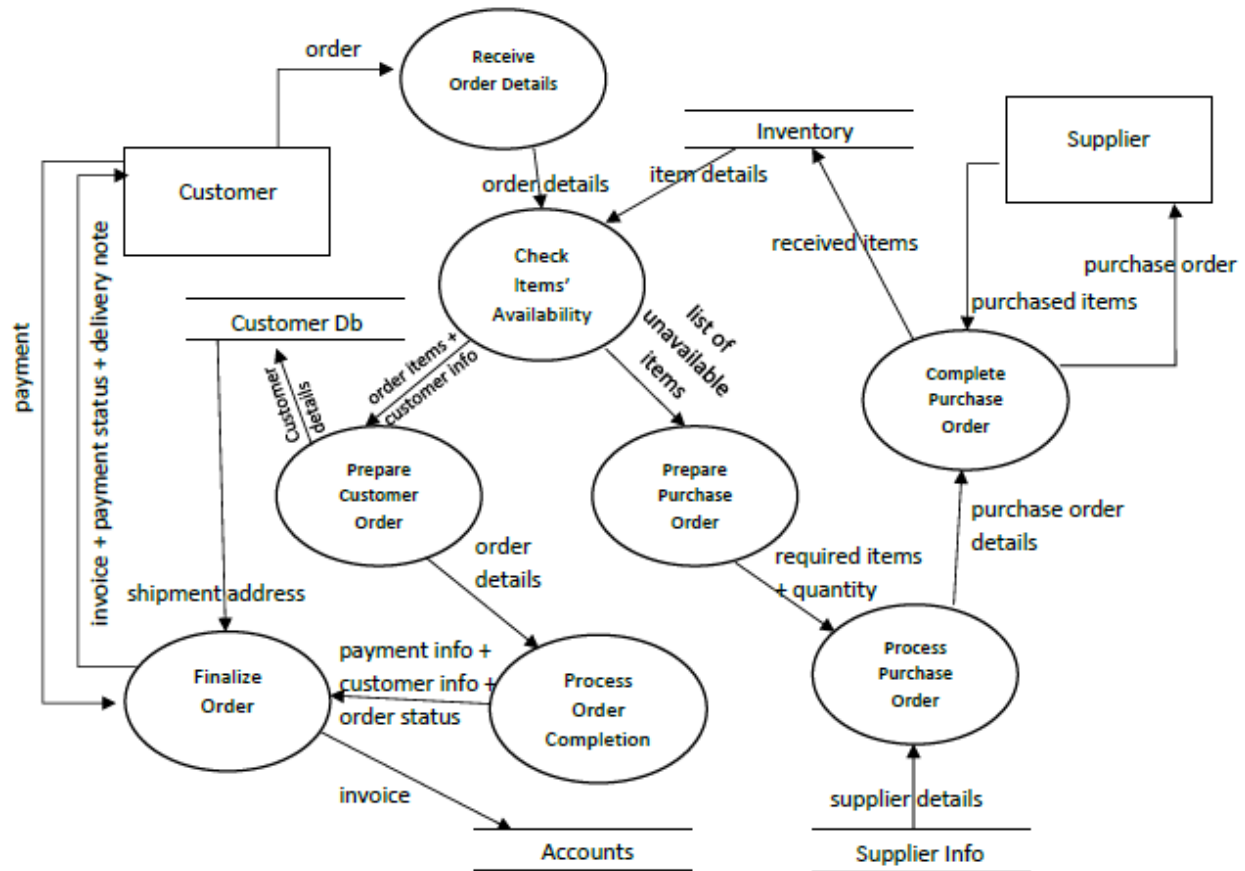# Practice Questions

Q1. Wasim Tiwana and Nasim Tiwana are two brothers who have been running an electronic items shop for quite a few years. Their customers visit them with the orders and the brothers provide the customers with the required items. In case the required items are not available in stock, Tiwanas immediately request their supplier to send the items. Once the items are received and their stock is enough, they prepare the order and move towards completion of the order. During the completion phase they have to generate an invoice, receive payments, and provide a delivery note to the customers.

Develop a use case diagram for their business

Q2. Consider the business mentioned in Q1. Tiwana brothers have now decided to computerize their system to reduce the paper work involved in recording and processing of the orders. They have hired a software engineer Shamim Khatana who has analysed their requirements and has come up with a Data Flow Diagram (DFD). Khatana being a professional, has been following SE principles and has refined the DFD to the last possible level. This DFD can be seen in the figure below. Khatana's next step is to perform structured design and come up with an architecture of this order management system.

order
Receive Order Details

Inventory

Supplier

Customer

order details

item details

received items

purchase order

purchased items

Check Items' Availability

Customer Db

order items + customer info

list of unavailable items

Complete Purchase Order

Customer details

payment

invoice + payment status + delivery note

Prepare Customer Order

Prepare Purchase Order

purchase order details

order details

required items + quantity

shipment address

Finalize Order

payment info + customer info + order status

Process Order Completion

Process Purchase Order

invoice

supplier details

Accounts

Supplier Info

To do:

a) Is this DFD meeting all the notation related requirements? If no, modify the DFD to make it complete in terms of notation and rules

b) Which of the processes are potentially 'Not primitive'? If possible, refine the DFD to the next level. If it is not possible, justify why can't this DFD be further refined.

Q3. In most of the software engineering projects the clients are found to be changing their requirements frequently and the software engineers are found to be dealing with the issue of unstable requirements. The requirements keep on changing even when the major portion of the software has been implemented and tested. The software engineers try to incorporate the required changes even in later phases of software development. Malleability is a property of software that allows the clients to change the requirements that frequently and lets the software engineers attempt the implementation of the changes. What issues can arise if this property is exploited? What can a good requirements engineer do to deal with unstable requirements and yet complete (or continue with) the development of the software?

Q4. **Problem**

Distance learning till today is one-way road as only teachers are delivering content or video lectures. In this way students can't ask questions on parts of lecture, which may leave them with ambiguity on some topic(s).

It is hard to maintain hardcopies of lectures for teachers and rewrite them every time they teach the same class. It is not even a good idea for students to maintain their notes in form of hardcopies and do their

assignments on papers and the much of the research work they have to do now days is through internet and eBooks. So it is a tough ask for students even to maintain hardcopies of lectures in each semester or year of their college and where there are many chances to lose the back track of their previous lectures and notes.

Secondly, for professional students and the students of practical subjects like Mathematics and Arts, it is necessary to look through problems deeply and ask teachers questions on each point of confusion.

Thirdly, students lose some concentration to lecture, as they get more involved in taking down notes instead of listening to the lecture and grasping what the teacher is pinpointing. This is another vital issue that causes a lack of interest in many students and many fail to get most out of lectures due to this problem.

The Proposed Solution

To handle these problems, this project suggests an innovative interface for virtual class environment. The focus of the project is to devise an application that will simulate real-world class environment for distance learning and would provide a platform for teachers and students to perform their tasks with more ease and perfection in a timely manner. The teachers would be able to deliver lectures electronically using tablet (promulgator) that would be automatically copied to the tablets (promulgators) of the students attending the class and they would rather be concentrating on lectures instead of taking notes. Moreover students would be engaging in question and answers in a way to cope up with distance learning interactivity problem. Students would be proposing their answers by writing or drawing the idea at real time on tablet (promulgator) and would be questioning by properly writing and preplanning their query, which would all, be visible at runtime to an online class canvas.

So, the core point is to develop a healthy classroom environment that simulates real-world class environment, which would help students in distance learning and teachers in distance teaching.

To do: Your task is to read the above text and list all the functional and non-functional requirements that are relevant with the problem. All the written requirements should be **Testable** (aka Verifiable). All the requirements should follow the standard format and style for writing the requirements.

Q5. We need to develop an automated ticketing system which allows travelers to get the tickets issued and cancelled. The travelers can pay for the tickets through credit cards only. The ticketing system receives traveler's details (such as name, contact number, credit card number) and travel details (such as destination station, source station, travel date, travel time, number of seats, seat preferences) from a Traveler and issues a ticket to the traveler on the basis of availability of seats. The system checks for availability of seats before processing payments and issuing the ticket. The payments are processed based on already set ticket prices. The system contacts the Bank to process the payments and a ticket is issued only when payment through credit card is successful. The system provides the bank with the credit card (cc) information and the payment amount to receive a confirmation of successful payment. On successful payment the system issues a ticket to the traveler and records the information related to the ticket. The ticket has a ticket number, traveler's name, contact number, seating details, travel time, bus details, payment amount etc.

The system is also capable of handling ticket cancellation requests. A traveler needs to provide ticket number when requesting for cancellation. The system contacts the bank with cc information and the amount to be refunded, before sending the cancellation confirmation to the traveler. If a ticket cannot be cancelled, the system informs the traveler about rejection of the request.

To do: Develop a context level and level 1 DFD for this system. Refine the DFD to the maximum levels possible. Mention the reason to stop the refinement.

Q6. Suppose we are developing a water-monitoring system where data are gathered at many points throughout a river valley. At the collection sites, several calculations are done, and the results are communicated to a central location for comprehensive reporting. Such a system may be implemented with a computer at the central site communicating with several dozen smaller computers at the remote locations. Many system activities must be considered, including the way the water data are gathered, the calculations performed at the remote locations, the communication of information to the central site, the storage of the communicated data in a database or shared data file, and the creation reports from the data.

To do: Will developing an ER diagram for this system help elaborate the requirements? If possible, then take some assumptions regarding the system and draw an ER diagram.

Q7. Suppose we are developing a system to grant authorized access to lab based on RFID cards. The system will have an RFID module and is intended to prevent unauthorized students from entering the university's research lab. This prevention is done by controlling the lock on the doors. When a person tries to enter the lab, he/she touches his/her card with the RFID modules mounted on the door. The system compares the card holder's record with the records of authorized card holders. The door lock is opened only if the person is authorized to pass through the door.

To do: Which of the elaboration tools (diagrams) are applicable for this statement. Draw all the applicable elaboration diagrams.

Q8. My Cute Bank (MCB) has approximately 1500 branches all over Pakistan that use a computerized MCB banking system for daily banking activities. The branches are connected with their Head Office (HO) in Karachi and communicate with the HO at least twice a day. At start of each working day every branch has to mark start of the day through the computerized banking system and update its record from the HO to make the branch's records consistent with the records at the HO. Similarly, at the end of each working day every branch has to mark end of day. At end of day every branch sends complete information regarding the day's transactions to the HO and the HO updates its records related to customer accounts and the branch's status. Normally, the branches and the HO do not necessarily need to communicate with each other through the computerized banking system from start of the day to end of the day.

Approximately 80% branches have ATMs connected with them which accept MCB ATM cards only. During a day, ATM transactions on a machine connected with a branch are managed by the branch. During each ATM transaction, the branch contacts all other branches to determine the total amount of cash withdrawn in this day against a particular ATM card. This information is used to approve or disapprove a particular user request for cash withdrawal when the daily limit is exceeded for a particular card.

The HO also keeps record of branches that have or have not updated themselves at 8:55 AM on each working day. Using the computerized banking system a Support Engineer at the HO sends a message to Branch Manager if the manager's branch has not updated itself till 8:55 AM on a working day.

a. Identify the three subsystems of the MCB computerized banking system in the boxes given below and, in each box, list all modules needed in that subsystem.

Subsystem 1: _____	Subsystem 2: _____	Subsystem 3: _____

b. Assume that MCB banking system is currently operational in the following cities only: Karachi, Lahore, Peshawar, Quetta, Islamabad. There is only one branch of MCB in each of these cities including Karachi. In addition, head office of MCB is located in Karachi. Head Office does not act as a branch. All branches except Quetta and Islamabad branches have ATM machines attached with them.

Draw, using boxes and arrows, the architecture diagram for the given scenario (in part b) using the relevant information in the problem statement as well as in part b. While drawing this diagram, use all suitable architectural styles.

c. Give a comma separated list of all the architectural styles used in diagram of part b. Give a brief justification for using each architectural style.

Architectural   styles   used:
_____

Justification:

Q9. Royal Bank of Kuwait (RBK) needs a banking application that allows its customers to maintain their bank accounts. The system should allow a customer to view account balance and shop online using the bank account number. Customer can also view the status of the account and change currency of his/her bank account. Some other tasks that a customer should be able to perform include Pay Bills, Transfer Funds (to same or other banks), Request Chequebook Generate Mini Statement, Generate Account Statement for particular period (period not exceeding 1 year), Generate Transaction PIN, Add/Remove beneficiary for fund transfers, and Request Bank draft/Pay Order. The account statement of last 3 years can be requested through this application, the customer must visit bank in order to get more than 3 years old bank statement. Funds Transfer, Bill Payment, Online Shopping should be processed instantly. The account balance should reflect each of these activities within 5 seconds of request.

RBK expects that once the application is developed, their Customers can login through their unique account number and a strong password. But a customer cannot create his account, only Admin, residing in Head Office of RBK, will create new account of a customer. The Admin creates an account on recommendation of a Branch Manager who uses the same banking system to send the recommendations for opening an account. The Branch Manager enters customers' details in respective recommendations and sends the recommendations to create new accounts before the end of the day. Admin must ask the customers to verify their emails before end of the next working day. Account creation will be successful only if a customer verifies an email address by using the link sent during account creation process. Once the email is verified the Admin creates the account and provides a customer a unique login id through a verified email. The password is generated once account creation is successful, but the password is not shared through the RBK banking system. The customer must change the password at first login and password (of minimum length 8) must contain at least one capital alphabet of English, one small alphabet of English, one special character, and one digit. Admin will also be able to update/delete account of a customer, if recommended by the Branch Manager.

To do:

a. Draw a use case diagram
b. Draw a sequence diagram for the scenario that ensures the change of password at the first login
c. Draw a class diagram
d. Draw an ER diagram
e. Draw a decision table that specifies the rules to determine if a password is strong or not
f. Draw a swim lane diagram for the workflows mentioned in the statement (for example creation of an account)
g. List all the non-functional and functional requirements in the standards format

Q10. Our job is to develop a system to sell advertising time for Piccadilly Television, the holder of a regional British television franchise. As we shall see, the constraints on the price of television time are many and varied, so the problem is both interesting and difficult.

In Britain, the broadcasting board issues an eight-year franchise to a commercial television company, giving it exclusive rights to broadcast its programs in a carefully defined region of the country. In return, the franchisee must broadcast a prescribed balance of drama, comedy, sports, children's and other programs. Moreover, there are restrictions on which programs can be broadcast at which times, as well as rules about the content of programs and commercial advertising.

Piccadilly attracts most of the Audience in Midlands UK. Thus Piccadilly must set its rate to attract a portion of advertiser's budget. One of the ways to attract an advertiser's attention is with audience ratings that reflect the number and type of viewers at different times of day. The ratings are reported in terms of program type, audience type, time of day, and television company. But the advertising rate depends on more than just the ratings. For example, the rate per hour may be cheaper if the advertiser buys a large number of hours. Moreover, there are restrictions on the type of advertising at certain times and for certain programs. For example;

- Advertisements for smoking may be shown only after 9 PM
- If an actor is in a show, then an ad with that actor may not be broadcast within 30 minutes of the show (15 minutes before and 15 minutes after the show)
- If an ad for a class of products (such as an automobile) is scheduled for a particular commercial break, then no other ad for something in that class may be shown during that break.

Perform the following in the space provided on the next pages:

a. Develop an ER Model
b. Develop a Class Diagram
c. Develop a Sequence Diagram for the scenario of assigning a time slot to a commercial keeping in view the three restrictions mentioned in the description. The time slots are assigned by the class SlotAssignmentManager. Pick classes from your class diagram where required.

Note: Identify things (nouns) from the description. Determine the same things with different names. Identify the things that are NOT part of the System and those that are part of the System. Determine the things that need to be stored. Determine the things that will be used during execution of the programs only. Determine the behaviors of things that need to be modeled.

Q11. Relate the elements of "**Entries**" coulmn with the elements of "**Mismatched Entries**" coulmn. Write the most appropriate match in the coulmn named "**Correct Match**". Write the corresponding letter instead of writing full entry. For example, see that **Agile** in "**Entries**" is matched with **M** in "**Mismatched Entries**".

Note: "**Mismatched Entries**" coulmn contains more elements than "**Entries**" coulmn. You do not need to map every element of "**Mismatched Entries**" onto "**Entries**"

Entries	Correct Match	Mismatched Entries	
ER Diagram		A.	Legitimate/Allowed transitions in a system
Fence Diagram		B.	Quality requirements
Event Traces		C.	Event Management
Testing		D.	Static description of a system
Stakeholder conflicts		E.	Unification of process
Functional requirements		F.	Requirements prioritization
Non-functional requirements		G.	Interaction between system and its environment
Agile		H.	A sequence of steps
Unified Process		I.	A stage in waterfall model
Process		J.	Inception, Elaboration, Construction, Transition
		K.	Decision Tree
		L.	Composition
		M.	Scrum
		N.	Message Sequence Diagram

Q12. Consider a grocery store where customers can shop using a cart. The customers view a grocery list, select items of desire, fill their cart, and checkout. Checkout means payment is received at cash counter and the groceries in the cart are handed over to the customer. The sales data is then sent to Sales Data Store. The Store Manager monitors the sales data to identify weekly needs of the store and calls the suppliers to procure the needed items. An order is sent to the supplier during the call to the supplier. Weekly orders are delivered by the suppliers. Each weekly delivery is unloaded at first. Afterwards, individual items are sent to stock shelves.

You need to develop a website for the grocery store such that customers can shop remotely. Checkout in case of remote shopping means that the customer is asked to enter shipment address and the groceries are sent to the shipment department who is responsible for shipping the groceries. Remote payments are made through credit card. Checkout requires that there is at least one item in the cart. The website should also enable the Store Manager to identify weekly needs and place order for the required items. Place Order scenario involves the monitoring of sales data and identification of weekly needs.

To do:

   a. Identify Actors
   b. Develop a use case model
   c. Write a use case for Checkout
   d. Develop a sequence diagram for Checkout scenario of part c

Q13. Describe some ways to identify ambiguity in requirements

Q14. For an airport baggage handling system, generate a user story for dealing with baggage that is to be diverted to another flight.

Q15. What sort of problems might you find through traceability matrix that you might not see without one?

Q16. Solve the following problems from chapter 4 of Pfleeger's book: 1, 2, 4, 5, 8, 15, 16, 17, 19, 20

Q17. Solve the following problems from the uploaded book of Pressman: 8.2, 8.3, 8.9, 8.10, 8.11, 8.12,

Q18. Draw a context-level DFD for the following systems, take whatever assumptions you want:

   a. A network-based course registration system for your university
   b. A web-based order processing system for a computer store
   c. A simple invoicing system for a small business
   d. An automated cookbook that is built into an electric cooking range or microwave

Q19. From the list of systems given in Q18, write a high level narrative for one of the systems of your interest. Using the context level DFD developed in Q31 and the narrative written in this question, develop level 1 and level 2 DFDs. Use 'grammatical parse' on the narrative to get

yourself started (to develop level 1 DFD). Specify all information flows by labelling all arrows. Use meaningful names for each transform.

Q20. Consider the following description for a web-based Pothole Tracking and Repair System (PTRS):

Citizens can log onto a website and report the location and severity of potholes. As potholes are reported they are logged within a 'public works department repair system' and are assigned an identifying number, stored by street address, size, (on scale of 1 to 10), location (middle, curb, etc.), district (determined from street address), and repair priority (determined from the size of the pothole). Work order data are associated with each pothole and include pothole location and size, repair crew identifying number, number of people on a crew, equipment assigned, hours applied to repair, hole status (work in progress, repaired, temporary repair, not repaired), amount of filler material used, and cost of repair (computed from hours applied, number of people, material, and equipment used). Finally, a damage file is created to hold information about reported damage due to the pothole and includes the citizen's name, address, phone number, type of damage, dollar amount of damage. PTRS is a web-based system; all queries are to be made interactively.

To do:

   a. Develop a complete set of CRC cards for the PTRS
   b. Develop a class diagram for the PTRS
   c. Develop a use case diagram for the PTRS

Q21.

Waiters on Call (WoC) is a restaurant meal-delivery service started in 2003 by Sue and Tom Bickford. Both the Bickford worked for restaurants while in college and always dreamed of opening their own restaurant. But unfortunately, the initial investment was always out of reach. The Bickford notices that many restaurants offer takeout food, and some restaurant - primarily pizzerias - offer home delivery service. Many people they met, however, seemed to want home delivery service with a wider food selection.

Sue and Tom conceived Waiters on Call as the best of both worlds: a restaurant service without the high initial investment. The Bickford contracted with a variety of well-known restaurants in town to accept orders from customers and to deliver the complete meals. After preparing the meal to order, the restaurant charges Waiters on Call a wholesale price, and the customer pays retail plus a service charge and tip. Waiters on Call started modestly, with only two restaurants and one delivery driver working the dinner shift. Business rapidly expanded, and the Bickford realized they needed a custom computer system to support their operations. They hired a consultant, Sam Wells, to help them define what sort of a system they needed.

"What sort of events happen when you are running your business that make you want to reach for a computer?" asked Sam. "Tell me about what usually goes on."

"Well," answered Sue, "when a customer calls in wanting to order, I need to record it and get the information to the right restaurant. I need to know which driver to ask to pick up the order, so I need drivers to call in and tell me when they are free.

Sometimes customers call back wanting to change their orders, so I need to get my hands on the original order and notify the restaurant to make the change.

"Okay, how do you handle the money?" queried Sam.

Tom Jumped in, "The drivers get a copy of the bill directly from the restaurant when they pick up the meal. The bill should agree with our calculations. The drivers collect that amount plus a service charge. When drivers report in at closing, we add up the money they have and compare it with the records we have. After all drivers report in, we need to create a deposit slip for the bank for the day's total receipts. At the end of each week, we calculate what we owe each restaurant at the agreed-to wholesale price and send each a statement and a cheque."

"What other information do you need to get from the system?" continued Sam.

"It would be great to have some information at the end of each week about orders by restaurant and orders by area of town – things like that," added Sue. "That would help us decide about advertising and contracts with restaurants. Then we need monthly statements for our accountant."

Sam made some notes and sketched some diagrams as Sue and Tom talked. Then after spending some time thinking about it, he summarized the situation for Waiters on Call.

"It sounds to me like you need a system that does some processing when these events occur:

- A customer calls in to place an order, so you need to *record an order*.
- A driver is finished with a delivery, so you need to *record delivery completion*.
- A customer calls back to change an order, so you need to *update an order*.
- A driver reports for work, so you need to *sign in the driver*.
- A driver submits the day's receipts, so you need to *reconcile driver receipts*.

"Then you need the system to produce information at specific points in time – for example, when it is time to produce:

- an end-of-day deposit slip
- end-of-week restaurant payments
- weekly sales reports
- monthly financial reports

"Based on the way you have described your business operations, I am assuming you will need a database to store information about these types of things:

- Restaurant

- Menu items
- Customers
- Orders
- Order payment
- Drivers

"Then I suppose you are going to maintain the information in a database about restaurants and drivers. You will need to do some processing when you add a new restaurant, a restaurant changes the menu, you drop a restaurant, you hire a new driver

"Am I on the right track?"

Sue and Tom quickly agreed that Sam was talking about the system in a way they could understand. They were confident they had found the right consultant for the job.

To do: For the Waiters on Call system discussed above develop diagrams for all the applicable elaboration techniques.

Q22. Explain why is it important for a requirements engineer to understand the nature of stakeholders?

Q23.

Determine the type of flow (transaction or transform) in the following DFD with labelled boundaries. Map the flow to architecture and provide the corresponding architecture diagram for the modeled portion of the system. Name the architecture style used.
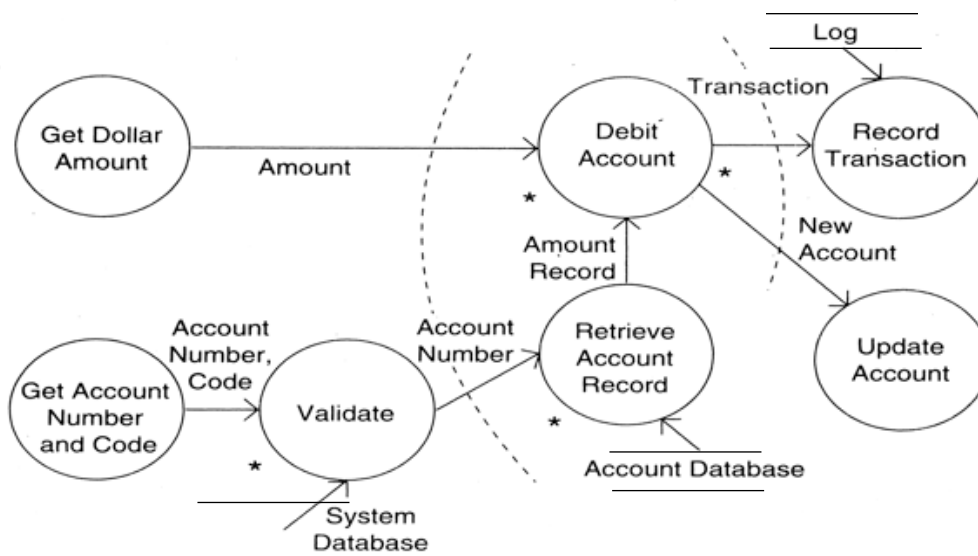
Q24.

In a chat application there can multiple chat groups. A user can be the participant of any chat group by sending joining request to the admin. The admin will then approve the request and the requesting user will become participant of the chat group. Once a user becomes participant of a group, (s)he will receive all the future messages (or updates) of that group instantly. The participant can choose to read a received message at the time of his/her choice.

    a. Which architecture style(s) seem(s) the most obvious to be used for this application? _____

    b. Provide the architecture diagram:

Q25.

You are a software development organization. A bank named JS approaches you to get its Internet Banking System (IBS) developed. You note that it is a web based application. You start gathering requirements for the IBS. In initial phases of requirements gathering a couple of more banks namely ABL and BAHL contact your company for the same reason. You start the requirements gathering phase for the IBS of ABL and BAHL too.

You notice during the process that the IBS of all banks has to deal with existing ATMs and the ATM transactions have to be recorded in addition to the daily routine banking (aka general banking) transactions such as cash receipt and payment transactions at cash counters, funds transfers or cheque clearing, demand draft and pay order management, loan repayment services, new loan approval and management(including personal loan management, car-financing, home-financing), managing cheque-book requests (receiving requests and issuing cheque-books, discarding cheque-books etc.), managing debit card related activities (managing new debit cards, managing complaints and blocking of the existing cards etc.), managing customer accounts (like creating new accounts, updating account details, biometric verification of accounts, managing bank account statements etc.). In addition the IBS also has to track the return of ATM cards captured at the local ATM (recording the return to the customer, or return to the parent bank). The IBS also deals with the utility bills, this module interacts with different utility billers and marks the billed paid upon receiving the billed amount from customer (through cash or cheque).

During the requirements gathering, you find out that all 3 banks have similar requirements in terms of banking operations (for example Online Account Statement Request, Funds Transfer, Bill Payment, Chequebook Request etc.). However, there are minor detailed level differences in the operations (for example, who updates the status of chequebook request, transfer to which banks is allowed etc). The other difference between the requirements lies in User Interface (UI). ABL requires very simplistic interface while the other two require the state of the art UI trends. Information that needs to be displayed on bank websites also differs, one bank has a plan to have a mobile application for their bank customers. Also all 3 banks would like to keep their data in separate data bases.

To Do:

    a. Provide the functional decomposition of the system (modules, sub-modules, sub-sub-modules etc.).

    b. Provide an architecture diagram and highlight the architecture styles used. The diagram should be detailed enough to indicate the placement of system's modules. The diagram should not be a high level (and generic) diagram only.

    c. Provide a component diagram

Hint: Design the architecture for the system such that you can reuse your work, and use the same application in future if more banks with similar requirements contact you? (i.e. design the architecture of the application keeping in mind the similarity in requirements, the objective of reusability and difference in interfaces)