UNIVERSITY OF THE COMMONWEALTH CARIBBEAN
A Member of the Commonwealth & OAS Consortia of Universities
School of Mathematics, Science & Technology
Department of Information Technology
Bachelor of Science Degree
in Information Technology

Course Code & Name	**ITT310 SYSTEMS ANALYSIS AND DESIGN**		
Unit Number	1,2		
Assignment Title	Assignment # 1		
Academic Year	2024		
Lecturer	Dr. Muhammad Zubair		
Issue Date	November 10, 2024	**Submission Date**	November 17, 2024
Weight	20%	**Presentation Time**	

Programme Learning Outcome and Course Learning Outcomes:

CLO-1	Explain the fundamentals of development methodologies and the business issues, processes, and techniques associated with organizational software analysis and design..
CLO-2	Compare current methods of analysis and design of information systems. Discuss the role of the systems development life cycle (SDLC) and its management. Discuss the attribute of a System Analyst

Assignment Brief and Guidance:

PERUSE THE INFORMATION BELOW BEFORE STARTING THE ASSIGNMENT.
The work done must solely and entirely be your own however using ChatGPT, other Artificial Intelligence tools or anyone else's work in any way is a violation of the code of academic integrity and will receive ZERO score. At the end of the assignment add the following statement with your electronic signature. "I CERTIFY THAT I HAVE NOT GIVEN OR RECEIVED ANY UNAUTHORIZED ASSISTANCE ON THIS ASSIGNMENT".

INTRODUCTION

This assignment is referred to the Unit 1,2 of this course. This assignment is comprising of 4 questions with different weightages. Some of the questions may have different valid solutions. You are expected to justify each of your answer with valid, clear, and accurate rationale.

INSTRUCTIONS

Please follow the given instructions carefully

- **Read the assignment carefully:** Carefully read the assignment instructions, including the hand-in date and any specific requirements or guidelines
- **Review the relevant course materials:** Discrete mathematics covers various topics such as sets, functions, sequences, proof techniques, and programming. Review your course notes, textbooks, and any additional resources provided by your instructor to ensure you have a solid understanding of the concepts.
- **Break down the assignment:** Break the assignment into smaller, manageable tasks. Identify the different types of questions or problems and prioritize them based on their difficulty and weightage
- **Start with the easier questions:** Begin by working on the easier questions first. This will help you build confidence and momentum for tackling the more challenging problems
- **Seek help if needed:** If you are struggling with a particular concept or problem, don't hesitate to seek help. You can reach out to your instructor, classmates, or online tutoring services for assistance
- **Manage your time:** Allocate sufficient time to complete the assignment, considering the complexity of the questions and your other commitments. Avoid procrastination and start working on the assignment well in advance
- **Check your work:** After completing the assignment, review your solutions for any errors or mistakes. Make sure your answers are clear, concise, and well-organized
- **Assignment Sheet.** MUST USE UCC STANDARD ASSIGNMENT SHEET
- **File Name convention:** FirstName_LastName_Assignment_1

By following these instructions, you can effectively approach your discrete mathematics assignment and improve your understanding of the subject.

Submission Format:	
Submission	Students must submit the assignment before the due date and late submission will incur 5% each working day. **The submission should be uploaded as PDF to the Moodle platform**.

Question 1:

You are a systems analyst at a company that is experiencing high customer dissatisfaction due to slow response times in their customer service system. The management wants to improve this system. Using your knowledge of Systems Analysis, outline the steps you would take to identify and resolve the problems in the current system. Justify your steps.

The steps in **SAD**, refer the system development life cycle framework

1. **Problem Definition**
2. **Requirements Gathering**
3. **System Analysis**
4. **System Design**
5. **System Development and Testing**
6. **System Implementation**
7. **System Maintenance**

By identifying these problems in the customer service system, particularly slow response times and a drop in customer satisfaction, a systems analyst would naturally start the process of determining what needs to be done to eliminate these issues. The steps would include:

1. Define the Problem

- Actions: Diagnostics Interviews with customer service personnel, management and customers on slow response times would reveal the(www), what, why and where. Review complaint logs and analyze performance metrics

- Justification: I would identify what causes the problem. Collect firsthand data so that I can better understand the problem.

2. Gather System Requirements

- Actions: I would conduct interviews with customers, service staff, and management, to identify desired outcomes; such as faster response times and improved customer satisfaction. Document their expectations and assess any specific features or functionalities needed.

- Justification: At this step, I would be building a clear picture of what the new system must achieve. By aligning the solution with both business goals and user needs to ensure that the problems are properly addressed and expectations are met.

3. Analyze the Current System

- Actions: Examine the existing customer service system by reviewing its hardware, software, workflows, and staffing. Pay special attention to identifying bottlenecks, performance issues, and recurring user complaints. Analyze system logs and performance reports to uncover areas of inefficiency.

- Justification: The goal is to now fully understand how the current system operates and why exactly it's falling short. By identifying the root of inefficiencies, I can now layout the groundwork for the best action plan in improvements for processes, technology, or resource allocation noting the (ppt framework).

4. Identify Causes of Delays

- Actions: Pinpoint the specific reasons behind slow response times by examining key areas such as:
  network performance, software functionality, database efficiency, staff training, and work processes. Use diagnostics tools, staff interviews, and performance data to identify where the system is falling short referencing the ITIL and ppt framework.

- Justification: The focus now is finding the root causes of the delays. By addressing these specific issues, you can implement targeted solutions that will have the greatest impact on improving response times and overall system efficiency.

5. Develop a Solution

- Actions: I would implement practical solutions to address the identified issues, such as upgrading outdated hardware if needed, streamlining software processes, automating repetitive tasks, or enhancing employee training programs. Include a mix of immediate fixes and strategies for sustained improvement. Collaborate with stakeholders to ensure proposed changes align with business goals.

- Justification: crafting solutions that will directly tackle the root causes of the problem. By ensuring the solutions are actionable, realistic, measurable and replicatable. Increasing the likelihood of achieving testable and replicable improvements in system stat performance and customer satisfaction.

6. Evaluate and Select Solutions

- Actions: I will have to analyze the proposed solutions to determine its proabbaility of success, check the cost-effectiveness, and potential impact. Conducting a cost-benefit analysis to compare the options and prioritize the most appropriate taking into consideration the urgency and or availability of resources. Collaborate with the company head to make sure these choices can alignment with business objectives.

- Justification: it is essential to choose a solution that is both practical and practoical within the company's budget and capacity. By carefully evaluating each option, I can ensure that the selected approach delivers the max value and also addresses the most critical issues as effectively as possible.

7. Implement the Solution

- Actions: I would beguin the roll out the selected solution with careful planning to minimize disruptions. This per the solution could include the upgrading of systems, deploying new software, or modifying customer service workflows. The conducting of pilot tests if necessary, and provide clear instructions to all users involved in the new transition.

- Justification: A structured implementation ensures the solution is integrated smoothly without significant downtime or negative impacts on the customer and employee experience.

8. Test and Monitor the System

- Actions: After implementing the solution, I would closely monitor the system's performance. stress testing is key in ensuring it can handle increased loads, gather user feedback, and track kpi metrics to check improvements in response time and customer satisfaction.

- Justification: Monitoring is critical to verify that the solution is functioning as expected. It also provides valuable insights into any potential adjustments or further optimization.

9. Train Staff and Provide Ongoing Support

- Actions: Provide training for customer service staff on the newly implemented systems and processes. Offer continuous support and make necessary adjustments based on staff feedback and performance metrics.

- Justification: training is needed to make sure the staff can use the system efficiently, that will directly improve response times and service quality. Support management helps maintain system performance and makes sure that any issues that may arise are remedied quickly.

Question 2:

As a newly appointed systems analyst at a mid-sized manufacturing company, you are tasked with overseeing the development of a new inventory management system. What key roles and functions should you perform to ensure the success of this system? Provide a scenario where these roles are applied in the development phase.

.

1. Requirements Gathering

- Role: to serve as the bridge between management, employees and the IT team; to make sure all the requirements for the inventory system are identified and understood.

- Function: using methods like interviews, surveys, and on-site observations to gain direct insights into the desired functionality of the system. Focus on areas such as real-time inventory tracking, barcode scanning, and reporting frame work

- Scenario Application: a customer service department at a medium-sized retail chain struggling with slow response times and declining customer satisfaction. While conducting diagnostics interviews, customer service representatives express that response delays often stem from outdated ticketing software and unclear workflows. Management highlights the lack of real-time performance tracking, while customers express frustration over repeated follow-ups and unresolved issues.

.

2. System Analysis and Design

- Role: I would analyze existing processes and identify problems, such as inefficiencies in customer service systems, to design solutions that improve performance.

- Function: to evaluate current systems to identify the key issues, the Development of targeted technical specifications to address noticed problems and align with organizational goals.

- Scenario Application:
  A large e-commerce company facing a spike in customer complaints, primarily due to slow response times and unresolved issues. Customer service representatives reveal that much of their time is spent manually sorting and assigning tickets, causing delays. Management notes the absence of tools to monitor response efficiency, while customers express frustration over repeated follow-ups and extended wait times for resolutions. Using "what is causing the delays," "why are they happening," and "where in the workflow are issues occurring."

3. Feasibility Study and Cost-Benefit Analysis

- Role: I would make the decision to implement if the proposed system is practical and worth the investment by weighing its costs against its benefits.

- Function: Looking at the financial, technical, and operational aspects of implementing the system. Considering costs of the needed software, hardware, and training, and compare them to the expected benefits, such as saving time, reducing blunders.

Scenario Application:
A customer service team is dealing with a slow response times and unhappy /irate customers. To see if a new ticketing system is a good idea, I would:
1. Work with the finance team to estimate savings from faster response times and fewer mistakes.
2. Compare buying ready-made software to building a custom system, considering costs, how long it takes to set up, and how well it can grow with the company.
3. Think about how training the team and switching systems might impact daily work.

4. Vendor Selection (if applicable)

- Role: Help choose the right vendor or software solution if the company decides to use an off-the-shelf product.

- Function: Research and compare potential vendors to ensure they fit the company's technical needs, budget, and long-term goals.

Scenario Application:
A company is looking to improve its inventory management system but wanting to avoid the costs of developing a custom solution. I would:
1. Inspect and research for the right vendors that offer inventory software that integrates well with the company's current ERP system.
2. Check for options noting on factors like ease of use, cost, scalability and quality
3. Launch product demos and trials to get feedback from key managers and IT staff.

5. System Development/Customization

- Role: to guide the development and tailoring of the system to make sure it aligns with the requirements identified earlier.
- Function: to work closely with the dev team and or external vendors to build & adjust the system. Insuring functionality and quality
- Scenario Application:

A growing retail company struggling with inefficiencies in its manual inventory tracking. As the business expands, they decide to implement a new inventory management system.

In the system dev phase, Id coordinate closely with the development team using a collaborative platforms like GitHub or GitLab. This helps us to keep track of the progress and ensure that key features; such as barcode scanning for inventory updates.

---

6. Testing and Quality Assurance

- Role: to oversee the testing phase to make sure everything works as expected and there are no bugs.
- Function: Creating testing cases to validate all features of the system. Test it in different environments to ensure it works correctly under real world conditions.
- Scenario Application: an organize round of user acceptance testing with warehouse staff, who test the system by performing tasks like tracking stock, creating orders, and generating reports. I would ensure any issues that pop up are resolved before the system launches or goes live.

---

7. Training and Documentation

- Role: Training the users and provide documentation for how the system should be used and maintained.
- Function: Prepare training materials and lead sessions for the users make sure they know how to use the system effectively.
- Scenario Application: After the system is deployed, I would have created detailed user manuals and host training sessions for warehouse staff to show them how to use barcode scanners, track inventory, and generate reports.

---

8. Implementation and Go Live

- Role: to oversee the deployment of the new system into the live environment.
- Function: Planning the transition from the old system to the new one, with minimal disruptions.
- Scenario Application: a situation where managing the go-live process, ensuring all data is accurately migrated from the old system and that staff is ready to use the new one. And a back up on standby in case any unforeseen situations.

---

9. Post-Implementation Support and Evaluation

- Role: Providing ongoing support and assess the system's performance after it goes live.
- Function: Monitor the system's performance, while fixing any issues that could arise, making noted activity allocation to collect feedback from users.
- Scenario Application: After the system is live, I would collect feedback from the warehouse team about how it's performing. If issues arise (like glitches in reporting), close work with the tech team would be underway to resolve them

---

10. Continuous Improvement

- Role: evaluate the system to make improvements as needed.
- Function: Gather feedback from users and assess system performance to identify areas for improvement.
- Scenario Application: let's say a few months after the system is up and running, an assessment is done whether additional features, like predictive analytics or automated reporting, could improve inventory management. Immediately future planning would be on ground for updates and scalability.

Question 3:

Your company is deciding between implementing the SDLC (System Development Life Cycle) or another development approach like Agile for a new project. Given the organization's structured environment and long-term goals, recommend which development system would be more appropriate. Justify your choice and highlight potential impacts of this decision on information systems within the company.

Why SDLC is the Right Approach???

1. Structured Environment:
   - The SDLC is a traditional approach that works well for organizations with clear, hierarchical structures. SDLC is ideal. It allows everyone to follow the same process with well-defined phases; requirement gathering, design, testing, implementation, and maintenance, which helps with the maintainance of consistency and control throughout the projects lifespan.

2. Long-Term Goals:
   - SDLC is well geared for projects with clear, long-term objectives and limited changes once the requirements are set. It's basically particularly useful for large, complex systems (like ERP or CRM systems) where every detail needs to be carefully planned and controlled

3. Emphasis on Documentation:
   - With SDLC, thorough documentation is key. You'll have very very clear records for each phase, from requirement documentation to design specs and testing. This helps maintain accountability, facilitates audits, and supports training for employees.

4. Control Over Scope:
   - The frameworks linear process helps keep the project's scope clearly defined, reducing the risk of scope creep. This is crucial in a structured environment where deviating from the original plan can cause coordination issues.

Impact of SDLC on information systems

- Integration with existing systems:

- The SDLC's detailed implementation process and analysis ensures that the new system will fit seamlessly into your company's existing infrastructure. This reduces the risk of compatibility issues with legacy systems. This allows for a smoother transition and better integration as you deploy new systems.

- Improve risk management:

- The SDLC's structured approach helps identify potential risks early. This is especially true during the planning or design stages. Addressing these risks early It will reduce surprises later in the process. This leads to smoother implementation. and reduce the chance of disruption in business operations...

- Long Term Support:

- The SDLC's emphasis on proper documentation and comprehensive testing facilitates long-term system maintenance. As your system evolves This structured approach allows for efficient updates and troubleshooting. Ensuring sustainability without causing major disruption to day-to-day operations...

- Compliance and Audit:

- For industries with strict regulatory requirements, the SDLC prioritizes documentation to simplify the process to meet legal standards. With a step-by-step development process ones company can provide clear evidence of compliance. This ensures a smooth inspection. and reduce regulatory risk.

My argument is why while AGILE may not be the best option,

Although Agile is suitable for projects that require rapid changes, But it is not suitable for environments where consistency is required. Detailed documents and careful boundary control In highly structured settings with complex systems; Agile's flexibility can lead to scope creep, inconsistent documentation and unpredictable timelines

In compact :
For companies with a structured environment and clear long-term goals Systems Development Lifecycle (SDLC) is the right way to go. Provide a predictable and well-documented process to ensure the new system meets the company's needs. Integration with existing systems and easy to maintain over time Compliance with the SDLC helps companies pursue strategic goals and avoid risks associated with More flexible development processes such as Agile.

Question 4:

A client has requested a detailed understanding of their system's structure. As a systems analyst, you are tasked with creating graphical representations of systems. Describe how you would use Data Flow Diagrams (DFD) and Entity-Relationship Models (ERM) to map out the system. Present a scenario where these diagrams would help in identifying system issues or designing improvements.

To provide a detailed understanding of a system's structure for a client, I would use **Data Flow Diagrams (DFD)** and **Entity-Relationship Models (ERM)** as key tools.

Level 0

Zero Level DFD for Online Mobile Store Management System provides an overview of the entire mobile store project. It is the highest level of abstract vision of the entire system. It is also called a reference diagram of an online mobile store management system. At this level, the entire system is represented as a single process and has relationships with external agencies.

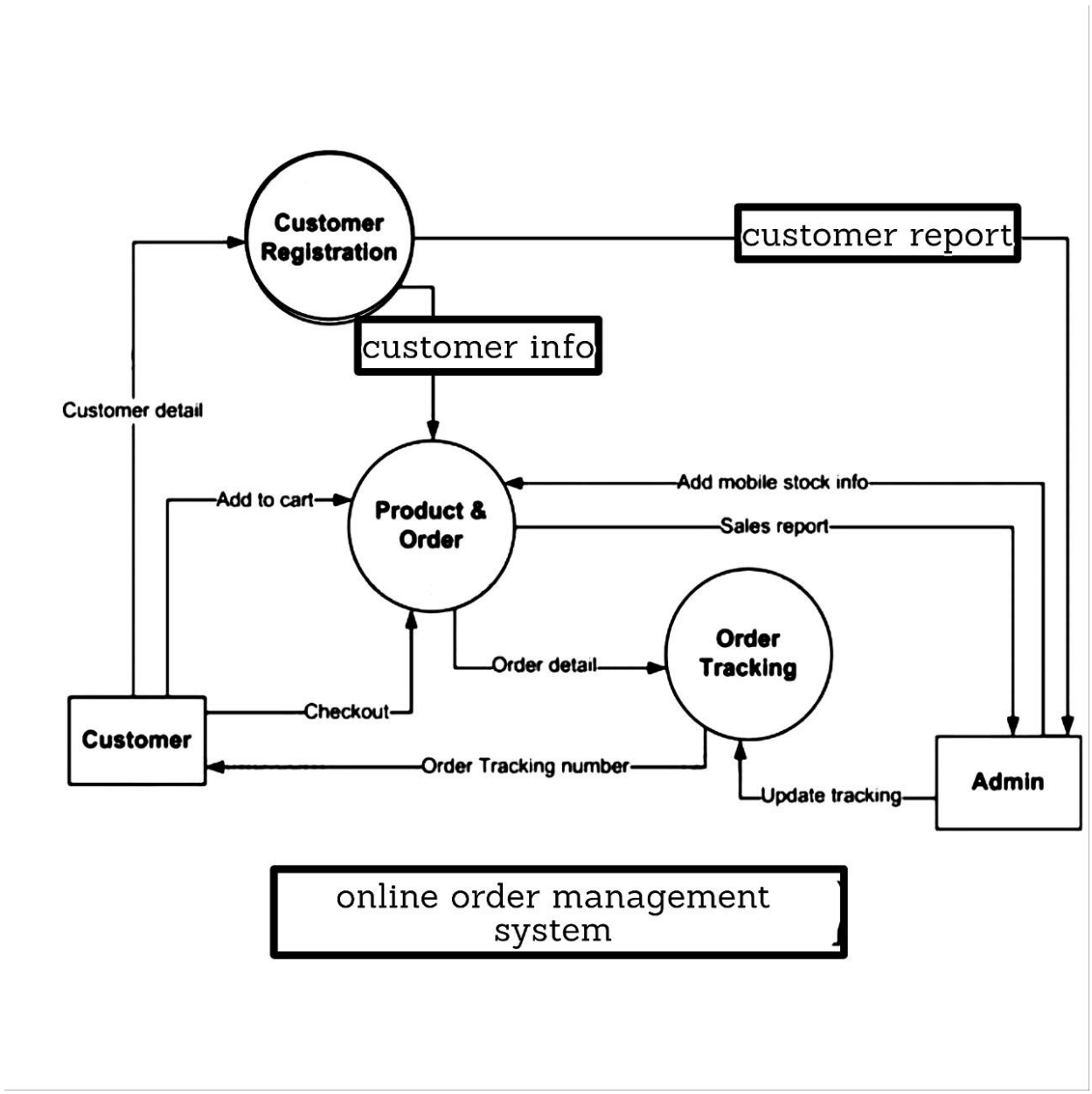
Level 1 (Decomposed DFD):

Break the main system process into smaller sub-processes, each showing how data is transformed.

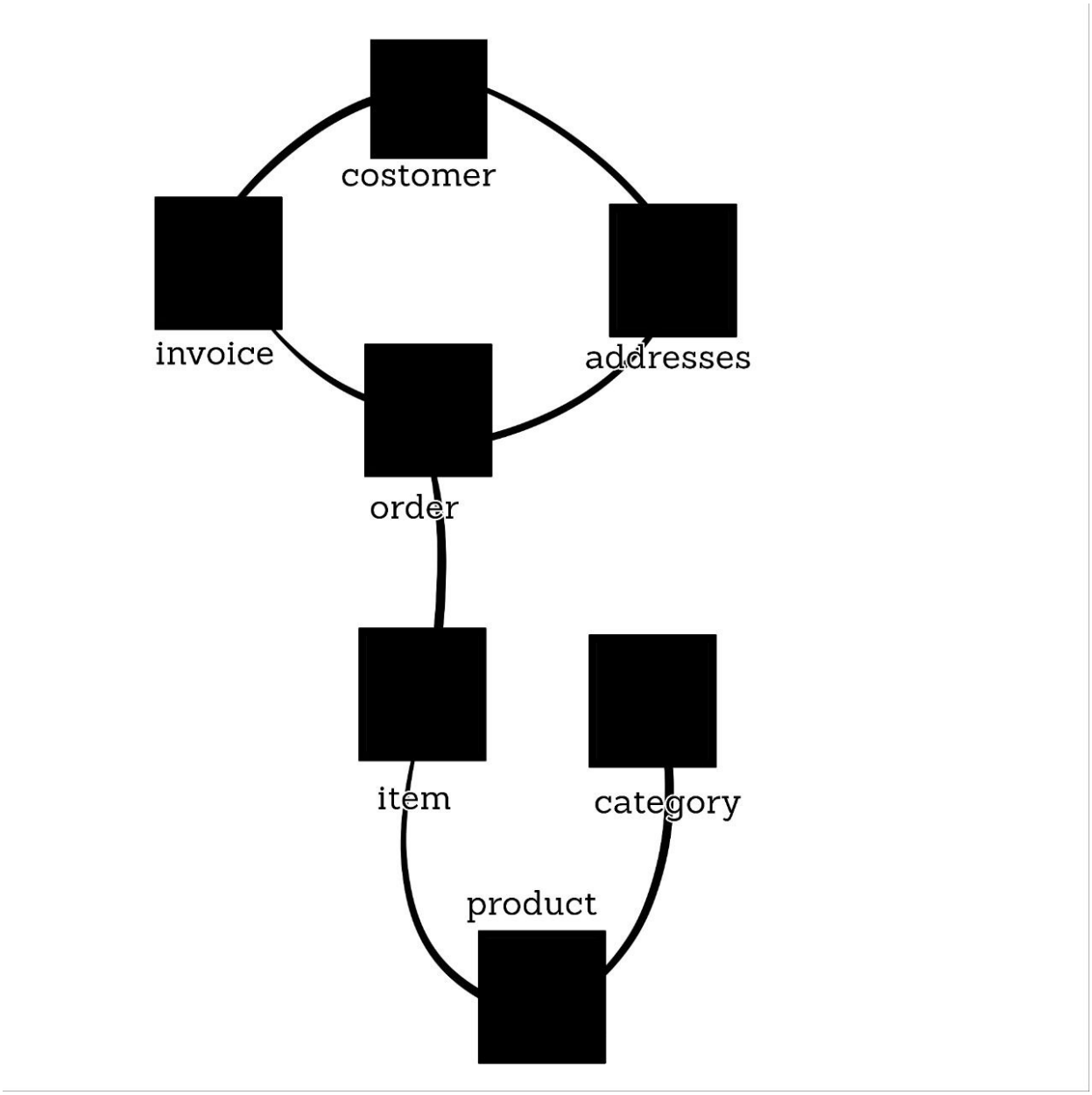
- Customer Registration: Keeps track of customer information.
- Product & Order: Handles products and orders.
- Order Tracking: Follows orders from start to finish.
- Customer Info: Stores customer details like contact information and purchase history.

The system also interacts with:

- Customers: Who buy products.
- Admins: Who manage the shop.


Customer Registration
customer report
customer info
Customer detail
Add to cart
Product & Order
Add mobile stock info
Sales report
Order detail
Order Tracking
Checkout
Customer
Order Tracking number
Update tracking
Admin
online order management system

ERM :



- **DFD Level 0**: This Level 0 Data Flow Diagram (DFD) provides a high-level overview of the online order management system. It presents the system as a process and its interactions with two external entities: the customer and the system administrator. Customers can register Add product to cart and can pay Administrator can manage the system by adding stocks. Generate sales reports and update order tracking details. This DFD basically shows the scope of the system and its interaction with the outside world.
- **DFD Level 1**:. This figure shows the flow of information in an online order management system. It shows how customers can register. Add products to cart and can pay The system then processes the order, tracks its status, and creates reports for customers and admins.
- **ERM**: The diagram shows how customers can order different items. Each order can have multiple items. And each item belongs to a specific product category. An invoice is created for each order. and customer information including address It is linked to their orders and invoices.

MARKING SCHEME– 20%

Questions #	Criteria	Exceeds Standard (5)	Meets Standard (4)	Acceptable Standard (3)	Below Standard (2)	No Outcome or Unacceptable Standard (1)
1	**Understanding of Systems Analysis**	Demonstrates deep understanding, effectively outlines and justifies key steps.	Outlines most key steps, with clear justifications, minor gaps in detail.	Covers basic steps, with some justification but lacks depth in some areas.	Mentions steps but lacks proper explanation or justification.	Little to no understanding of systems analysis or justification.
2	**Systems Analyst Roles and Functions**	Provides a comprehensive list of roles with appropriate application scenarios.	Most roles are identified, with relevant but somewhat unclear scenarios.	Lists basic roles, provides a scenario with limited depth or detail.	Roles are vaguely mentioned, and scenario application is unclear.	Roles are either missing or poorly applied to the scenario.
3	**Recommendation of Development System**	Thoroughly justified recommendation with well-explained impacts on the system.	Recommendation is mostly well-justified, with minor gaps in understanding.	Offers a valid recommendation, though with limited justification or impact.	Recommendation is unclear or lacks proper reasoning, impact not addressed.	No clear recommendation or reasoning, lacks connection to the scenario.
4	**Use of Graphical Representation Tools**	Correctly describes DFD and ERM with detailed scenario application.	Provides a good explanation of DFD and ERM, with some minor application gaps.	Basic understanding of DFD and ERM, limited or unclear application.	Limited understanding of the tools, scenario application lacks clarity.	Inadequate or incorrect understanding of DFD/ERM, no valid scenario given.