WILFRED GITHUKA
BIT2019/44664

BIT2206 SYSTEM ANALYSIS & DESIGN
CAT

1. **Explanation of 2 types and 4 functions performed by various documentation produces during the design and development of a computerized computerized information processing system.**

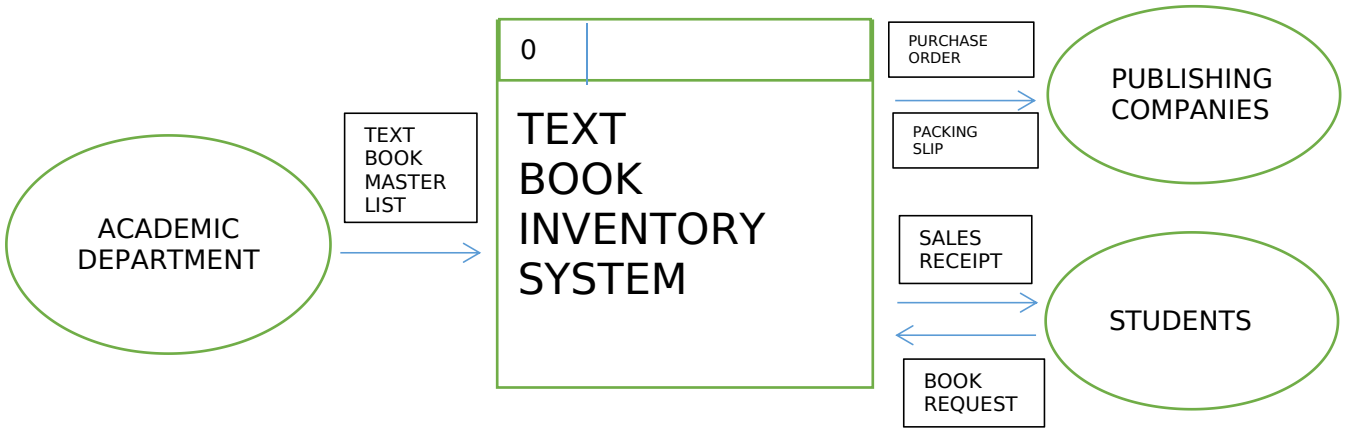
Operations Documentation.
Operations documentation contains all the information needed for processing and distributing online and printed output. Operations documentation should be clear, concise, and available online if possible.

Functions:
- E-mail and report distribution lists.
- Special forms required, including online forms.
- Error and informational messages to operators and restart procedures.
- Special instructions, such as security requirements.

User Documentation.
It includes instructions and information to the users who will interact with the system. For example, user manuals, help guides, and tutorials.

Functions:
- A system overview that clearly describes all major system features, capabilities, and limitations.
- Description of source document content, preparation, processing, and, samples.
- Overview of menu and data entry screen options, contents, and processing instructions.

Context High Level DFD for a Text Book Inventory System.



2. **Explanation of 5 features of the structured approach to system analysis and design**

- Logical Data Modeling: This involves the process of identifying, modeling and documenting data as a part of system requirements gathering. The data are classified further into entities and relationships.

- Data Flow Modeling: This involves tracking the data flow in an information system. It clearly analyzes the processes, data stores, external entities and data movement.

- Entity Behavior Modeling: This involves identifying and documenting the events influencing each entity and the sequence in which these events happen.
- How information is changed in its life - Entity Life Histories.
- How the information is stored and how it is structured - Logical Data Models

3. **Difference between Validation and Verification**

   Verification is the process of evaluating software to determine whether the products of a given development phase satisfy the conditions imposed at the start of that phase. Its is a static practice of verifying documents, design, code and program. It also includes all the activities associated with producing high quality software: inspection, design analysis and specification analysis. It is a relatively objective process.

   Validation is he process of evaluating software during or at the end of the development process to determine whether it satisfies specified requirements. It looks at final product to check whether the software meets the customer expectations and requirements. It is a dynamic mechanism of validating and testing the actual product.

4. **Distinguishing between functional and non functional requirements, citing an example in each.**

   A functional requirement is describes the behavior of the system as it relates to the system's functionality. The non-functional requirement elaborates a performance characteristic of the system.

   A functional requirement describes what a software system should do, while non-functional requirements place constraints on how the system will do so. Example: A system must send an email whenever a certain condition is met (e.g. an order is placed, a customer signs up, etc).

   A non-functional requirement for the system may be: Emails should be sent with a latency of no greater than 12 hours from such an activity.

5. **Description of tests in system implemented**

- **Parallel Conversion**
  A type of conversion in which both new and old systems operate together for a period of time is called parallel conversion. It is the safest type of conversion. The results of both systems are compared. The old system can be used until all problems in the new system are removed.

- **Pilot Conversion**
  In this type of conversion, one part of the organization uses the new system and the rest of the organization uses the old system. When one part of the organization is satisfied with the new system, the rest of the organization can start using it.

- **Phased conversion**
  A type of conversion in which individual components of new system are implementation one by one is called phased conversion.

- **Direct conversion**

In this type of conversion, the old system is directly replaced by the new system. It is the most risky conversion. It may be necessary when time is very short. It is also called crash conversion.