Subscribe to
ISM - IGNOU STUDY MENTOR
Youtube Channel and
Get a Complete Solution
for IGNOU Students

www.ignoustudymentor.com

Follow : 🐦 ▶ f 📷 💬
ism_ignoustudymentor_

No. of Printed Pages : 2   BCS-051

BACHELOR OF COMPUTER APPLICATIONS
(BCA) (Revised)
Term-End Examination
December, 2022

BCS-051 : INTRODUCTION TO SOFTWARE
ENGINEERING

Time : 3 hours   Maximum Marks : 100
                                 (Weightage : 75%)

Note : *Question number* **1** *is* **compulsory** *and carries 40 marks. Attempt any* **three** *questions from the rest.*

1. (a) Develop SRS for *Online Examination Form Submission System (OEFSS)*. SRS should be in IEEE format. Make necessary assumptions.   25

   (b) Explain *Prototype Model,* with the help of an example. What are its advantages and disadvantages over Waterfall Model ?   10

   (c) What is a Structure Chart ? Explain with the help of an example.   5

BCS-051   1   P.T.O

2. (a) Draw the zero and first level DFDs for OEFSS. Make necessary assumptions.   10

   (b) Draw GANTT Chart for the development of OEFSS.   10

3. (a) Draw ERD for OEFSS. Make necessary assumptions.   10

   (b) Briefly explain different levels of SEI-CMM.   10

4. (a) Explain the terms "Black Box Testing" and "White Box Testing".   10

   (b) How will you ensure that the software developed by you meets the Quality benchmarks ? Define the term "Software Quality".   10

5. (a) In Object Oriented Design, list the common utility objects and criteria for identifying utility objects.   10

   (b) Explain various Debugging strategies.   10

BCS-051   2

Ques. 1(a) Develop SRS for Online Examination Form Submission System (OEFSS). SRS should be in IEEE format. Make necessary assumptions.

1. Introduction

1.1 Purpose

The purpose of this document is to provide a detailed specification for the development of an Online Examination Form Submission System. This system will allow users to submit examination forms online, streamlining the process and reducing manual paperwork.

1.2 Scope

The Online Examination Form Submission System will be a web-based application that enables users to submit their examination forms electronically. The system will handle form submission, validation, storage, and retrieval of examination forms. It will also provide administrative functionalities to manage user accounts, forms, and generate reports.

1.3 Assumptions

• Users have access to a computer or mobile device with internet connectivity.

• Users are registered students or candidates eligible for examination form submission.

• The system will support multiple types of examination forms.

• The system will be accessible through standard web browsers.

• The system will be developed using a web application framework and a relational database management system.

• Adequate security measures, such as user authentication and data encryption, will be implemented to protect user data.

• The system will be hosted on a reliable web server.

1.4 Overview

The rest of the document provides an overview of the system's functional and non-functional requirements, system features, external interfaces, and constraints.

2. Overall Description

2.1 Product Perspective

The Online Examination Form Submission System will be a standalone application that interacts with external systems for authentication and storage. It will integrate with the existing student database system for user authentication and retrieve information related to examination forms.

2.2 Product Features

• User registration and login: Users can create an account and log in to the system.

• Examination form submission: Users can fill out and submit examination forms online.

• Form validation: The system will validate the submitted forms for completeness and correctness.

• Form storage: Submitted forms will be stored securely in a database for future reference.

• Form retrieval: Users can view and download their submitted forms.

• Administrative functions: Admin users can manage user accounts, examination forms, and generate reports.

2.3 User Classes and Characteristics

• Students: Registered students eligible for examination form submission.

• Administrators: System administrators responsible for managing user accounts, forms, and generating reports.

2.4 Operating Environment

• The system should be compatible with commonly used web browsers (e.g., Chrome, Firefox, Safari).

• The system should be hosted on a web server with appropriate hardware and software requirements.

3. System Features and Requirements

3.1 User Registration and Login

3.1.1 Description

Users should be able to create a new account and log in to the system.

3.1.2 Functional Requirements

• FR 3.1.2.1: The system shall provide a registration form to collect user information.

• FR 3.1.2.1: The system shall provide a registration form to collect user information.

• FR 3.1.2.3: The system shall store user account information securely.

• FR 3.1.2.4: The system shall provide a login form to authenticate registered users.

• FR 3.1.2.5: The system shall validate user credentials during login.

3.2 Examination Form Submission

3.2.1 Description

Users should be able to fill out and submit examination forms online.

3.2.2 Functional Requirements

Ques. 1(c) What is a Structure Chart ? Explain with the help of an example.

A structure chart is a graphical representation that depicts the modular structure of a software system or application. It shows the hierarchical organization of modules or components and their relationships within the system. The structure chart helps in understanding how different modules interact with each other and how the system's functionality is divided and allocated to different components.

```
        +----------------+
        |  E-commerce    |
        |    System      |
        +----------------+
                |
    +--------------+-------------
        ----+
    |            |            |
    V            V            V
```

```
+---------------+ +---------------+ +---------------+
| User          | | Product       | | Order         |
| Management    | | Catalog       | | Processing    |
+---------------+ +---------------+ +---------------+
        |                 |                 |
+--------|------+ +-----|-------+ +-----|-------+


    |       |      | |      |      | |     |       |
    V       V      V V      V      V V     V       V

+------------+ +-------------- +----------------+
| User       | | Product      | | Order         |
| Registration| | Search      | | Placement     |
| Login      | | Listing      | | Payment       |
+------------+ +---------------+ +----------------+
```
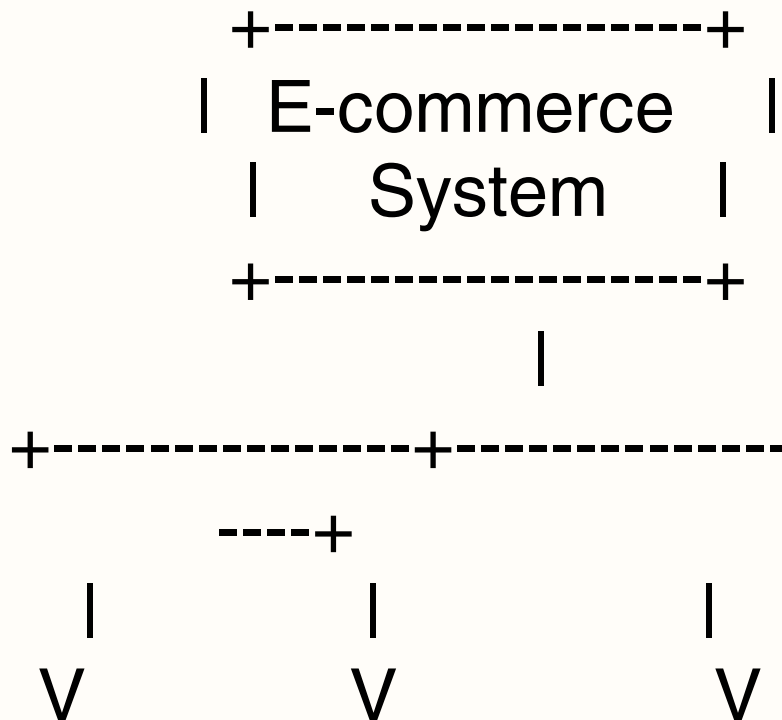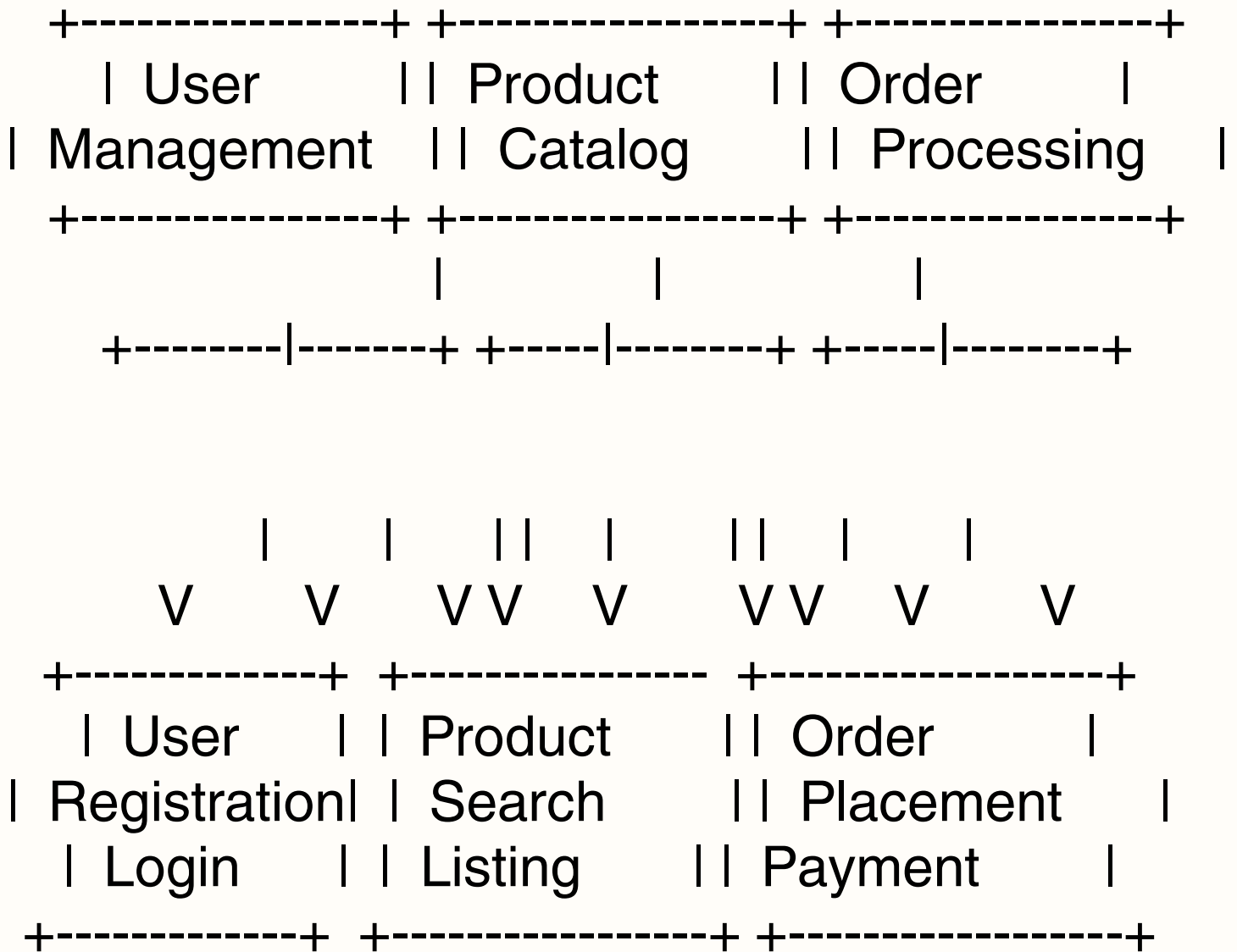
In this structure chart example, the top-level module is the "E-commerce System," representing the entire system. The system is divided into three major modules: User Management, Product Catalog, and Order Processing. Each module is represented by a rectangle.

The User Management module can be further divided into sub-modules: User Registration and Login. These sub-modules handle user-related operations, such as new user registration and authentication.

The Product Catalog module is divided into sub-modules: Product Search and Listing. These sub-modules handle product-related functionalities, such as searching for products and displaying product listings.

The Order Processing module is divided into sub-modules: Order Placement and Payment. These sub-modules handle the processing of orders and payment transactions.

The structure chart provides a clear visual representation of the system's hierarchical structure, showing how the different modules and sub-modules are organized and interact with each other.

Ques. 2(b) Draw GANTT Chart for the development of OEFSS.

|---------------------------------|
Development Timeline

Task	Duration
1. Requirements Gathering	5 days
2. System Design	7 days
3. Database Design	4 days
4. User Registration	5 days
5. User Login	4 days
6. Form Submission	6 days

```
| 7. Form Validation              |   4 days   |
|----------------------------------|
| 8. Form Storage and Retrieval   |   6 days   |
|----------------------------------|
| 9. Administrative Functions     |   7 days   |
|----------------------------------|
| 10. Testing and Bug Fixing      |   10 days  |
|----------------------------------|
| 11. Documentation               |   5 days   |
|----------------------------------|
| 12. Deployment and Launch       |   3 days   |
|----------------------------------|
```

Ques. 3(a) Draw ERD for OEFSS. Make necessary assumptions.

```
        +----------------+
        |     User       |
        +----------------+
        | UserID (PK)    |
        | Name           |
        | Email          |
        | Password       |
        +----------------+
                |    ^
                |    |
                |    |
        +----------------+
        |  Examination   |
        |     Form       |
        +----------------+
        | FormID (PK)    |
        | UserID (FK)    |
        | FormType       |
        | SubmissionDate |
        | Status         |
        +----------------+
                |    ^
                |    |
                |    |
```

```
        +------------------+
        |   Form Question   |
        +------------------+
        | QuestionID (PK)  |
        | FormID (FK)      |
        | Question         |
        +------------------+
                |      ^
                |      |
                |      |
        +------------------+
        |   Form Response   |
        +------------------+
        | ResponseID (PK)  |
        | FormID (FK)      |
        | QuestionID (FK)  |
        | Answer           |
```
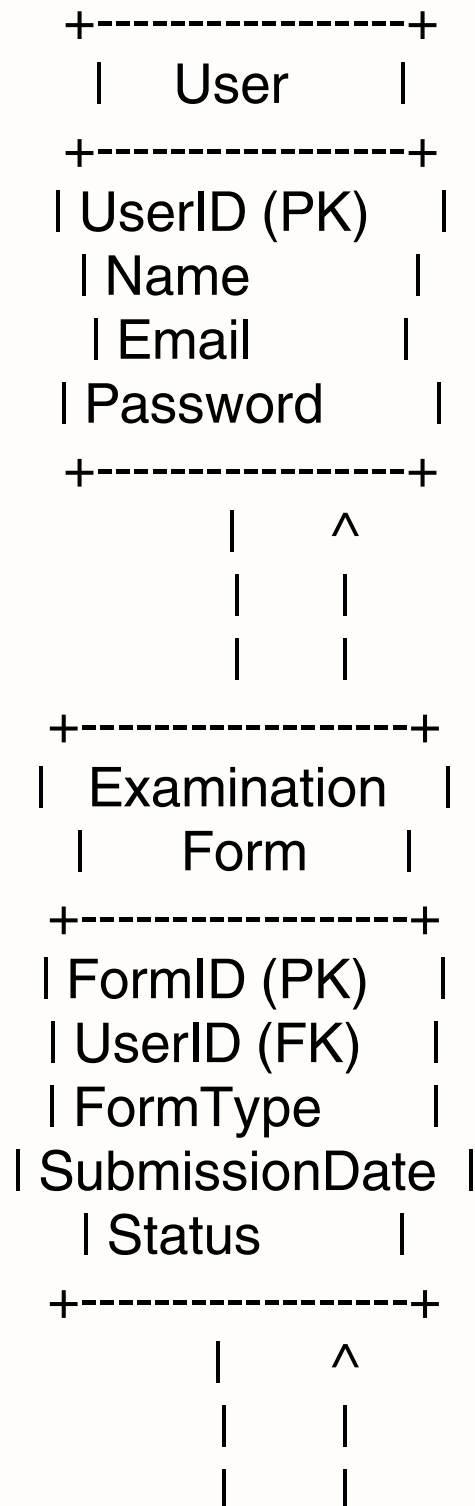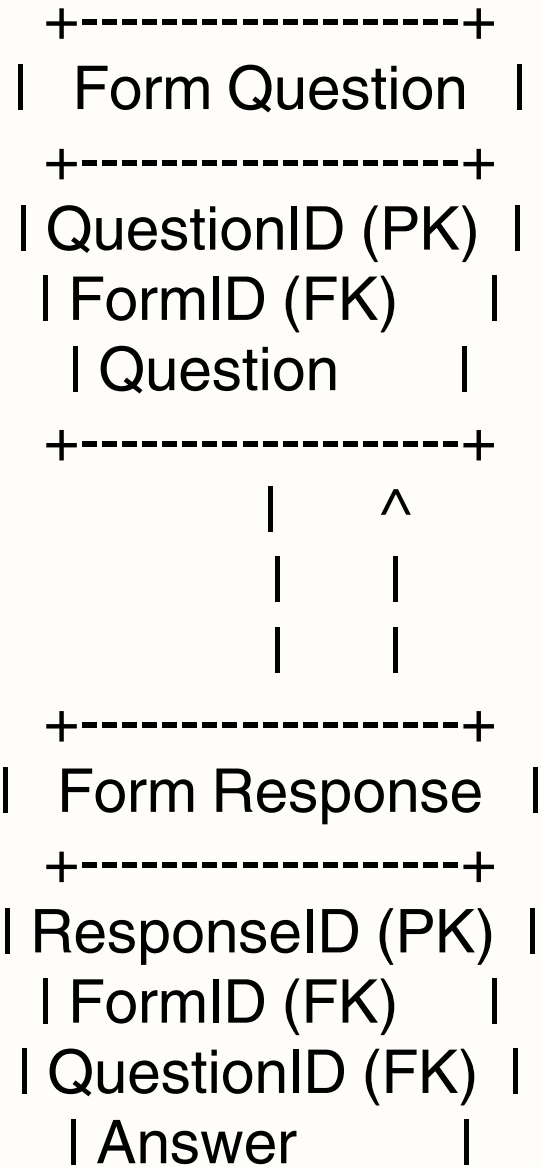
Ques. 4(a) Explain the terms "Black Box Testing" and "White Box Testing".

White Box Testing: This method, also known as glass box testing, is performed early in the testing process. Using this, the software engineer can derive a tests that guarantees that all independent paths within the module have been exercised at least once.

It has the following features:

(i) Exercise all logical decisions on their true and false sides

(ii) Execute all loops at their bouridaries and within their operational bounds.

(iii) Exercise internal data structures to assure their validity.

Blackbox testing:

This is applied during the later stage of testing: It enables the software developer to derive a set of input conditions that will fully exercise the functional requirements of a program. It enables him to find errors like incorrect or missing functions, interface errors,' data structures or external' data base access errors and performance errors.

It has the following features:

(i) To rectify the errors which are encountered during the operation of software.

(ii) To change the program function to interface with new hardware or software.

(iii) To change the program according to increased requirements.

Follow :

ism_ignoustudymentor_

 Youtube Channel

ISM - IGNOU Study MENTOR

Website

www.ignoustudymentor.com

730-339-3555