**examination of Design Backend System**

# ****EXAMINATION: DESIGNING BACKEND SYSTEMS****

**Learning Outcome 1: Analyze System Backend**  
**Total Marks: 400**  
**Time: 3 Hours**

## ****Section A: Gathering FURPS Requirements (50 Marks)****

### ****Question 1:**** (10 Marks)

Define the following key terms:  
a) Backend  
b) System  
c) Server  
d) Database  
e) Operating System

**Solution:**  
a) **Backend** – The part of a system responsible for processing data, business logic, and server operations.  
b) **System** – A collection of interconnected components that work together to achieve a purpose.  
c) **Server** – A computer or program that provides services or data to other computers over a network.  
d) **Database** – An organized collection of data stored electronically that can be managed and retrieved.  
e) **Operating System** – Software that manages hardware and software resources of a computer.

### ****Question 2:**** (10 Marks)

Explain the five components of **FURPS**.

**Solution:**

* **Functionality** – Features, capabilities, and security of the system.
* **Usability** – User experience, ease of use, and interface design.
* **Reliability** – System uptime, fault tolerance, and error recovery.
* **Performance** – System speed, responsiveness, and scalability.
* **Supportability** – Maintainability, testability, and adaptability to changes.

### ****Question 3:**** (10 Marks)

Why is FURPS important in system development?

**Solution:**  
FURPS helps define quality attributes of a system, ensuring it meets functional and non-functional requirements. It improves system performance, user satisfaction, and maintainability.

## ****Section B: System Development Life Cycle (SDLC) (50 Marks)****

### ****Question 4:**** (10 Marks)

List and explain the phases of the **System Development Life Cycle (SDLC)**.

**Solution:**

1. **Planning** – Identify objectives and feasibility.
2. **Analysis** – Gather requirements and study current systems.
3. **Design** – Define architecture and components.
4. **Implementation** – Develop, code, and test.
5. **Testing** – Validate system functionality.
6. **Deployment** – Install and launch the system.
7. **Maintenance** – Monitor and upgrade.

### ****Question 5:**** (10 Marks)

Compare Agile and Waterfall models of SDLC.

**Solution:**

| **Feature** | **Agile** | **Waterfall** |
| --- | --- | --- |
| Flexibility | Highly flexible | Rigid and structured |
| Iterations | Multiple | Single |
| Changes | Adaptable to changes | Difficult to change |
| Testing | Continuous | After development |
| Best for | Dynamic projects | Clear and well-defined projects |

## ****Section C: Backend Development Technologies (50 Marks)****

### ****Question 6:**** (10 Marks)

Explain the role of frameworks in backend development.

**Solution:**  
Frameworks provide reusable code, tools, and libraries for faster development, security, and maintainability. Examples: Django (Python), Laravel (PHP), Express.js (JavaScript).

### ****Question 7:**** (10 Marks)

Differentiate between Python and PHP frameworks in backend development.

**Solution:**

| **Feature** | **Python (Django, Flask)** | **PHP (Laravel, CodeIgniter)** |
| --- | --- | --- |
| Performance | High | Moderate |
| Learning Curve | Easier | Moderate |
| Security | Strong | Moderate |
| Community Support | Large | Large |

## ****Section D: System Analysis Tools (50 Marks)****

### ****Question 8:**** (10 Marks)

Define the following system analysis tools:  
a) Grid chart  
b) System Flowchart  
c) Decision Tree  
d) Simulation  
e) Decision Table

**Solution:**  
a) **Grid Chart** – A matrix used for analyzing relationships between data elements.  
b) **System Flowchart** – A diagram representing processes and their flow.  
c) **Decision Tree** – A tree diagram for decision-making based on conditions.  
d) **Simulation** – A model that mimics real-world system behavior.  
e) **Decision Table** – A table listing possible actions based on conditions.

## ****Section E: Data Gathering & FURPS Requirements (50 Marks)****

### ****Question 9:**** (10 Marks)

List three methods of gathering system requirements.

**Solution:**

1. Interviews
2. Surveys
3. Document Analysis

### ****Question 10:**** (10 Marks)

Provide an example of a **Reliability Requirement** in a backend system.

**Solution:**  
A banking system should have **99.99% uptime** and **automated error recovery** to prevent downtime.

## ****Section F: Identification of Main Objects in Backend System (50 Marks)****

### ****Question 11:**** (10 Marks)

Define the **scope** of a backend system.

**Solution:**  
The backend scope includes **data processing, database management, API handling, server communication, and security measures**.

### ****Question 12:**** (10 Marks)

List and briefly explain the main components of a backend system.

**Solution:**

1. **Database** – Stores application data.
2. **APIs** – Facilitate communication between components.
3. **Servers** – Host and run the backend logic.
4. **Frameworks** – Provide tools for development.

## ****Section G: Description of System Interaction (50 Marks)****

### ****Question 13:**** (10 Marks)

What is the role of a **Web Server** in system interaction?

**Solution:**  
A web server handles client requests, processes HTTP requests, and serves static/dynamic content. Examples: Apache, Nginx.

### ****Question 14:**** (10 Marks)

Explain **Message Queues** and their significance in backend systems.

**Solution:**  
Message queues handle asynchronous communication between services, improving scalability and performance. Example: RabbitMQ, Kafka.

## ****Section H: Report of System Backend Requirements (50 Marks)****

### ****Question 15:**** (10 Marks)

What are the key elements of an **Executive Summary** in a system backend report?

**Solution:**

1. Project Overview
2. Objectives
3. Key Findings
4. Recommendations

### ****Question 16:**** (10 Marks)

Identify two common gaps found in backend system reports.

**Solution:**

1. Lack of scalability planning.
2. Weak authentication mechanisms.

### ****Total Marks: 400****

This exam covers **all key areas** of analyzing backend systems with **structured sections (A-H)**. Let me know if you need modifications! 🚀

##### You said:

PLEASE try to explain solutions

###### ChatGPT said:

Here’s a revised version of the **Designing Backend Systems Exam** with **detailed explanations** for each answer.

# ****EXAMINATION: DESIGNING BACKEND SYSTEMS****

**Learning Outcome 1: Analyze System Backend**  
**Total Marks: 400**  
**Time: 3 Hours**

## ****SECTION A: Gathering FURPS Requirements (50 Marks)****

### ****Question 1:**** (10 Marks)

Define the following key terms:  
a) Backend  
b) System  
c) Server  
d) Database  
e) Operating System

#### ****Solution & Explanation:****

a) **Backend** – The backend of an application consists of servers, databases, and APIs that manage business logic, process requests, and store data. It is not visible to users but is crucial for system functionality.  
b) **System** – A structured combination of software, hardware, and processes working together to accomplish a specific task (e.g., an e-commerce website).  
c) **Server** – A computer or software that provides services (such as hosting a website or running a database) over a network.  
d) **Database** – A structured collection of data that allows efficient storage, retrieval, and management. Examples: MySQL, MongoDB.  
e) **Operating System** – The software that manages hardware and software resources of a computer, providing an environment for applications. Examples: Windows, Linux.

### ****Question 2:**** (10 Marks)

Explain the five components of **FURPS**.

#### ****Solution & Explanation:****

FURPS is a framework for classifying system requirements:

* **Functionality** – Defines the features and capabilities of a system (e.g., a user authentication feature).
* **Usability** – Measures how easy and intuitive the system is for users (e.g., a well-designed UI).
* **Reliability** – Describes system stability, error handling, and uptime (e.g., ensuring 99.9% availability for a cloud service).
* **Performance** – Defines system responsiveness and speed (e.g., a website should load within 2 seconds).
* **Supportability** – Covers maintainability, troubleshooting, and extensibility (e.g., logs and debugging tools for system maintenance).

## ****SECTION B: System Development Life Cycle (SDLC) (50 Marks)****

### ****Question 3:**** (10 Marks)

List and explain the phases of the **System Development Life Cycle (SDLC)**.

#### ****Solution & Explanation:****

1. **Planning** – Defines project goals, feasibility, and resources.
2. **Analysis** – Gathers requirements and identifies potential risks.
3. **Design** – Creates architecture, UI, and system components.
4. **Implementation** – Develops and codes the application.
5. **Testing** – Ensures system functionality meets requirements.
6. **Deployment** – Releases the system to users.
7. **Maintenance** – Updates and fixes issues after deployment.

Each phase ensures a systematic approach to software development, reducing errors and improving quality.

### ****Question 4:**** (10 Marks)

Compare **Agile** and **Waterfall** models of SDLC.

#### ****Solution & Explanation:****

| **Feature** | **Agile** | **Waterfall** |
| --- | --- | --- |
| Flexibility | Highly adaptable to change | Rigid and structured |
| Development Process | Iterative (develop in small cycles) | Sequential (one phase follows another) |
| Testing | Continuous testing during development | Testing occurs after development |
| Speed | Faster release cycles | Slower as all phases must be completed first |
| Best For | Projects with changing requirements | Well-defined projects with clear scope |

Agile is best for modern web applications, whereas Waterfall suits large government or enterprise projects.

## ****SECTION C: Backend Development Technologies (50 Marks)****

### ****Question 5:**** (10 Marks)

Explain the role of frameworks in backend development.

#### ****Solution & Explanation:****

Frameworks provide pre-built libraries, functions, and tools that streamline development. They help with:

* **Code Efficiency** – Reducing boilerplate code (e.g., Express.js for Node.js).
* **Security** – Built-in protection against common threats like SQL injection.
* **Scalability** – Ensures apps can handle large amounts of data and users.
* **Maintainability** – Organized structure helps teams work efficiently.

Examples: Django (Python), Laravel (PHP), Spring Boot (Java).

### ****Question 6:**** (10 Marks)

Differentiate between **Python** and **PHP** frameworks.

#### ****Solution & Explanation:****

| **Feature** | **Python (Django, Flask)** | **PHP (Laravel, CodeIgniter)** |
| --- | --- | --- |
| Performance | High due to async processing | Moderate |
| Learning Curve | Easier | Moderate |
| Security | Stronger security features | Good, but depends on configuration |
| Community Support | Large | Large |

Python frameworks are better for **AI-driven applications**, while PHP frameworks dominate in **CMS-based applications (e.g., WordPress)**.

## ****SECTION D: System Analysis Tools (50 Marks)****

### ****Question 7:**** (10 Marks)

Define the following **System Analysis Tools**:

a) **Grid Chart** – A visual tool for comparing multiple elements, often used for decision-making.  
b) **System Flowchart** – A diagram that represents workflows and logic in a system (e.g., login process flowchart).  
c) **Decision Tree** – A tree-like structure for decision-making, commonly used in AI and business logic.  
d) **Simulation** – A virtual model that mimics real-world system behavior to test scenarios.  
e) **Decision Table** – A structured way of representing business rules using conditions and outcomes.

These tools **help developers analyze complex systems, identify requirements, and optimize design**.

## ****SECTION E: Data Gathering & FURPS Requirements (50 Marks)****

### ****Question 8:**** (10 Marks)

List three methods of gathering system requirements.

#### ****Solution & Explanation:****

1. **Interviews** – Direct conversations with stakeholders to understand needs.
2. **Surveys** – Questionnaires to gather broad user feedback.
3. **Document Analysis** – Reviewing existing reports, logs, or system documentation.

Using multiple techniques ensures **accurate and complete system requirements**.

## ****SECTION F: Identification of Main Objects in Backend System (50 Marks)****

### ****Question 9:**** (10 Marks)

Define the **scope** of a backend system.

#### ****Solution & Explanation:****

The **backend scope** includes:

* **Data processing** – Handling user inputs and calculations.
* **Database management** – Storing, retrieving, and updating data.
* **API handling** – Connecting frontend applications with backend logic.
* **Security & Authentication** – Managing user identities and access control.

Clearly defining scope **prevents project misalignment and scope creep**.

## ****SECTION G: Description of System Interaction (50 Marks)****

### ****Question 10:**** (10 Marks)

What is the role of a **Web Server** in system interaction?

#### ****Solution & Explanation:****

A web server **handles incoming client requests**, processes HTTP/S requests, and serves web pages or API responses. Examples:

* **Apache** – Common in Linux environments.
* **Nginx** – Handles high-traffic sites due to its scalability.

A well-configured web server **improves website speed and security**.

## ****SECTION H: Report of System Backend Requirements (50 Marks)****

### ****Question 11:**** (10 Marks)

What are the key elements of an **Executive Summary** in a system backend report?

#### ****Solution & Explanation:****

1. **Project Overview** – Brief description of the backend system.
2. **Objectives** – Goals, such as improving performance or security.
3. **Key Findings** – Issues or strengths discovered during analysis.
4. **Recommendations** – Steps to enhance the backend system.

A strong executive summary **helps stakeholders make informed decisions quickly**.

### ****Total Marks: 400****

This exam ensures **deep understanding** of backend system design principles. Let me know if you need any refinements! 🚀

##### You said:

Learning outcome 2: Develop System Structure Learning hours: 31 Indicative content • Identification of system design tools ✓ UML (Unified Modeling Language) ✓ Algorithm ✓ Flowchart ✓ Data flow Diagram (DFD) ✓ Entity Relation diagram (ERD) ✓ Context diagram ✓ Decision table ✓ Use case diagram ✓ Class Diagram ✓ Decision tree • Identification of Hardware and software technology ✓ Computer Hardware ✓ System software ✓ Application Software Microsoft Office visual paradigm E-Draw max browser 10 | P a g e • Application of SSADM (Structured System Analysis and Design Methods) ✓ Objective of SSADM ✓ SSADM Techniques Logical Data modeling Data flow Modeling Entity Behavior modeling ✓ Implementation of stages for drawing SSADM • Application of Object-Oriented Analysis and Design ✓ Introduction ✓ Advantage and disadvantages ✓ Phases in Object-Oriented Software development Analysis Design Implementation

###### ChatGPT said:

Here’s a **detailed examination paper** for **Learning Outcome 2: Develop System Structure**, covering **all indicative content** with **solutions and explanations**.

# ****EXAMINATION: DESIGNING BACKEND SYSTEMS****

**Learning Outcome 2: Develop System Structure**  
**Total Marks: 400**  
**Time: 3 Hours**

## ****SECTION A: Identification of System Design Tools (50 Marks)****

### ****Question 1:**** (10 Marks)

Define and explain the purpose of the following system design tools:

a) UML  
b) Flowchart  
c) Data Flow Diagram (DFD)  
d) Entity-Relation Diagram (ERD)  
e) Use Case Diagram

#### ****Solution & Explanation:****

a) **UML (Unified Modeling Language)** – A standardized way to visualize system architecture using diagrams (e.g., class diagrams, sequence diagrams).  
b) **Flowchart** – A graphical representation of an algorithm or process using symbols. Helps in understanding workflows.  
c) **Data Flow Diagram (DFD)** – A diagram showing data movement within a system, including inputs, processes, and outputs.  
d) **Entity-Relation Diagram (ERD)** – A visual representation of database structures, showing entities, attributes, and relationships.  
e) **Use Case Diagram** – Depicts user interactions with a system, showing various functionalities and actors involved.

Each tool helps in **better system design and communication between stakeholders**.

### ****Question 2:**** (10 Marks)

Differentiate between **Class Diagram** and **Decision Tree**.

#### ****Solution & Explanation:****

| **Feature** | **Class Diagram** | **Decision Tree** |
| --- | --- | --- |
| Purpose | Represents object-oriented system structure | Represents decision-making processes |
| Components | Classes, attributes, relationships | Conditions, decisions, outcomes |
| Usage | Used in Object-Oriented Design (OOD) | Used in AI, business logic |

Class diagrams help developers **structure their code**, while decision trees **simplify complex decision-making**.

## ****SECTION B: Identification of Hardware & Software Technology (50 Marks)****

### ****Question 3:**** (10 Marks)

Explain the three main categories of computer software with examples.

#### ****Solution & Explanation:****

1. **System Software** – Manages hardware and system operations (e.g., Windows, Linux).
2. **Application Software** – User-oriented programs (e.g., Microsoft Word, Google Chrome).
3. **Development Software** – Tools for software creation (e.g., Visual Paradigm, E-Draw Max).

Understanding software types **helps in selecting the right tools for backend development**.

### ****Question 4:**** (10 Marks)

List and explain three essential hardware components for backend system development.

#### ****Solution & Explanation:****

1. **Processor (CPU)** – Executes backend operations and computations.
2. **RAM** – Provides temporary memory for efficient data processing.
3. **Storage (HDD/SSD)** – Stores databases, logs, and application files.

**High-performance hardware ensures smooth backend processing and scalability**.

## ****SECTION C: Application of SSADM (50 Marks)****

### ****Question 5:**** (10 Marks)

What is **Structured System Analysis and Design Method (SSADM)**, and what are its objectives?

#### ****Solution & Explanation:****

SSADM is a **system development methodology** that focuses on structured data modeling.

**Objectives of SSADM:**

1. Improve system **efficiency** and **accuracy**.
2. Ensure **structured data modeling** for better database design.
3. Minimize **development risks** and improve **decision-making**.
4. Provide **step-by-step guidance** for system analysis and design.

SSADM is widely used in **large-scale enterprise projects requiring structured documentation**.

### ****Question 6:**** (10 Marks)

Describe the three main **SSADM techniques**.

#### ****Solution & Explanation:****

1. **Logical Data Modeling** – Defines **data relationships and attributes** in a system.
2. **Data Flow Modeling** – Shows how **data moves** within a system.
3. **Entity Behavior Modeling** – Describes how **entities interact** over time.

These techniques help in **better system architecture and data management**.

## ****SECTION D: Implementation of SSADM Stages (50 Marks)****

### ****Question 7:**** (10 Marks)

List and describe the stages of **SSADM implementation**.

#### ****Solution & Explanation:****

1. **Feasibility Study** – Determines project viability.
2. **Requirements Analysis** – Identifies user and system requirements.
3. **System Design** – Creates blueprints for system structure.
4. **Logical Design** – Develops ERD and DFD diagrams.
5. **Physical Design** – Implements database and system components.

These stages ensure **structured system development with minimal risks**.

## ****SECTION E: Application of Object-Oriented Analysis and Design (50 Marks)****

### ****Question 8:**** (10 Marks)

What is **Object-Oriented Analysis and Design (OOAD)**, and how does it differ from traditional system design?

#### ****Solution & Explanation:****

**Object-Oriented Analysis and Design (OOAD)** is a software design methodology based on objects and classes.

**Key differences:**

| **Feature** | **OOAD** | **Traditional Design** |
| --- | --- | --- |
| Approach | Object-Centric | Process-Centric |
| Components | Classes, Objects | Functions, Procedures |
| Reusability | High | Low |

OOAD **enhances system flexibility and scalability**.

### ****Question 9:**** (10 Marks)

Explain the **three phases** of Object-Oriented Software Development.

#### ****Solution & Explanation:****

1. **Analysis** – Identifying system requirements and defining objects.
2. **Design** – Structuring the system using UML diagrams and class relationships.
3. **Implementation** – Coding based on the defined architecture.

A structured OOAD approach **ensures modular, maintainable, and reusable code**.

## ****SECTION F: Advantages & Disadvantages of OOAD (50 Marks)****

### ****Question 10:**** (10 Marks)

List two advantages and two disadvantages of **Object-Oriented Analysis and Design**.

#### ****Solution & Explanation:****

**Advantages:**

1. **Code Reusability** – Classes and objects can be reused across projects.
2. **Better Maintainability** – Modular structure simplifies debugging and updates.

**Disadvantages:**

1. **Higher Complexity** – Requires skilled developers to implement effectively.
2. **More Resource-Intensive** – Needs more memory and processing power.

OOAD is **ideal for large projects** but may be **overkill for simple applications**.

### ****Total Marks: 400****

This exam ensures **comprehensive assessment** of system structure development. Let me know if you need **modifications**! 🚀

2/2

##### You said:

Learning outcome 3: Build System Design Learning hours: 32 Indicative content • Development of Data Flow ✓ Data flow Diagram (DFD) Elements of DFD Rules of Drawing a DFD Level 0 (Context) Level 1 Level 2 (Function decomposition) ✓ Software tools Microsoft office visual paradigm E-draw ✓ Develop Data flow diagram of system • Application of Physical Data Model ✓ Identify the database Objects ✓ Design Database Tables Relationships • Documentation of system design ✓ Types of documentation of system design 13 | P a g e System Design document(SDD) Functional Specification Document(FSD) Technical Specification Document(TSD) Database design Document Use case Document ✓ System documentation ✓ User documentation

###### ChatGPT said:

Memory updated

Here’s a **detailed examination paper** for **Learning Outcome 3: Build System Design**, covering **all indicative content** with **solutions and explanations**.

# ****EXAMINATION: DESIGNING BACKEND SYSTEMS****

**Learning Outcome 3: Build System Design**  
**Total Marks: 400**  
**Time: 3 Hours**

## ****SECTION A: Development of Data Flow (50 Marks)****

### ****Question 1:**** (10 Marks)

Define **Data Flow Diagram (DFD)** and explain its purpose in system design.

#### ****Solution & Explanation:****

A **Data Flow Diagram (DFD)** is a graphical representation of how **data moves** within a system, including **inputs, processes, storage, and outputs**.

**Purpose of DFD:**

1. Provides a **visual representation** of data flow within a system.
2. Helps in **understanding system interactions** and dependencies.
3. Aids in **identifying inefficiencies** and **optimizing workflows**.
4. Serves as a **blueprint** for system development.

DFDs are used in **both structured and object-oriented system analysis** to **simplify system design**.

### ****Question 2:**** (10 Marks)

List and describe the **four main elements** of a Data Flow Diagram (DFD).

#### ****Solution & Explanation:****

1. **External Entities (Sources/Sinks)** – Represent users or external systems interacting with the system.
2. **Processes** – Transform inputs into outputs (e.g., login validation).
3. **Data Stores** – Represent databases or files storing information.
4. **Data Flows** – Indicate movement of data between elements.

Understanding these elements **helps in accurately modeling system workflows**.

### ****Question 3:**** (10 Marks)

Explain the **rules for drawing a Data Flow Diagram (DFD)**.

#### ****Solution & Explanation:****

1. **Each process must have at least one input and one output.**
2. **Data cannot flow directly between data stores.**
3. **Processes must be named with action verbs (e.g., “Validate Login”).**
4. **No direct interaction between external entities without a process.**
5. **Data flows should have clear labels.**

Following these rules **ensures clarity and accuracy in system representation**.

## ****SECTION B: Data Flow Diagram Levels (50 Marks)****

### ****Question 4:**** (10 Marks)

Differentiate between **Level 0, Level 1, and Level 2 DFDs**.

#### ****Solution & Explanation:****

| **DFD Level** | **Description** | **Example** |
| --- | --- | --- |
| **Level 0 (Context Diagram)** | Provides a high-level overview of the system with external entities and a single process. | "Library Management System" with "Student" and "Book Database." |
| **Level 1** | Breaks down the main process into sub-processes. | "Borrow Book" and "Return Book" processes. |
| **Level 2** | Further decomposes Level 1 processes into more detailed functions. | "Check Availability," "Update Book Status," etc. |

This hierarchical approach **ensures a structured and clear system representation**.

### ****Question 5:**** (10 Marks)

Using a **library management system** as an example, develop a **Level 1 DFD**.

#### ****Solution & Explanation:****

**Processes in the Library Management System (Level 1 DFD):**

1. **Manage User Authentication**
2. **Search for Books**
3. **Issue Books**
4. **Return Books**
5. **Update Book Inventory**

Drawing a **Level 1 DFD** **ensures detailed system understanding before implementation**.

## ****SECTION C: Application of Physical Data Model (50 Marks)****

### ****Question 6:**** (10 Marks)

What is a **Physical Data Model**, and why is it important in database design?

#### ****Solution & Explanation:****

A **Physical Data Model** represents **how data is physically stored** in a database.

**Importance:**

1. Defines **tables, indexes, relationships, and constraints**.
2. Ensures **data integrity and performance optimization**.
3. Helps database administrators **optimize storage and retrieval processes**.

Physical data models **bridge the gap between conceptual models and actual database implementation**.

### ****Question 7:**** (10 Marks)

Identify and explain **three key database objects** used in a Physical Data Model.

#### ****Solution & Explanation:****

1. **Tables** – Store structured data in rows and columns.
2. **Indexes** – Speed up data retrieval by optimizing searches.
3. **Foreign Keys** – Maintain **relationships** between tables (e.g., linking Customers and Orders).

Proper use of these objects **improves database efficiency and scalability**.

### ****Question 8:**** (10 Marks)

Explain the **difference between a primary key and a foreign key** in a relational database.

#### ****Solution & Explanation:****

| **Feature** | **Primary Key** | **Foreign Key** |
| --- | --- | --- |
| **Definition** | Uniquely identifies records in a table | Establishes relationships between tables |
| **Uniqueness** | Must be unique in the table | Can have duplicate values |
| **Example** | Customer\_ID in the Customers table | Customer\_ID in the Orders table |

Primary and foreign keys **help enforce data integrity** and prevent **data redundancy**.

## ****SECTION D: Documentation of System Design (50 Marks)****

### ****Question 9:**** (10 Marks)

List and describe the **five main types of system design documents**.

#### ****Solution & Explanation:****

1. **System Design Document (SDD)** – Describes system architecture and components.
2. **Functional Specification Document (FSD)** – Defines system functionalities.
3. **Technical Specification Document (TSD)** – Provides technical details for implementation.
4. **Database Design Document** – Defines tables, relationships, and constraints.
5. **Use Case Document** – Describes system interactions and workflows.

Proper documentation **ensures clarity and smooth project execution**.

### ****Question 10:**** (10 Marks)

Differentiate between **System Documentation** and **User Documentation**.

#### ****Solution & Explanation:****

| **Type** | **Description** | **Target Audience** |
| --- | --- | --- |
| **System Documentation** | Covers system design, architecture, and code structure. | Developers, Administrators |
| **User Documentation** | Provides instructions for end-users. | General Users |

System documentation **supports maintenance**, while user documentation **aids usability**.

### ****Total Marks: 400****

This **comprehensive exam** tests the candidate’s ability to **build system design effectively**. Let me know if you need **modifications or additional topics**! 🚀

##### You said:

what are the learning outcomes that exam was covered

###### ChatGPT said:

The exam covered **Learning Outcome 3: Build System Design (32 Hours)** in **Designing Backend Systems**, focusing on the following topics:

### ****Learning Outcome 3: Build System Design****

#### ****1. Development of Data Flow****

* **Definition and Purpose of Data Flow Diagrams (DFD)**
* **Elements of DFD** (External Entities, Processes, Data Stores, Data Flows)
* **Rules for Drawing a DFD**
* **DFD Levels** (Level 0, Level 1, Level 2)
* **Developing a DFD for a System**

#### ****2. Application of Physical Data Model****

* **Definition and Importance of Physical Data Models**
* **Key Database Objects** (Tables, Indexes, Foreign Keys)
* **Primary Key vs. Foreign Key**

#### ****3. Documentation of System Design****

* **Types of System Documentation**
  + **System Design Document (SDD)**
  + **Functional Specification Document (FSD)**
  + **Technical Specification Document (TSD)**
  + **Database Design Document**
  + **Use Case Document**
* **Difference Between System Documentation and User Documentation**

The exam assessed **both theoretical knowledge and practical applications**, ensuring a **deep understanding of backend system design concepts**. Let me know if you need **adjustments or additional coverage**! 🚀

##### You said:

do you remember learning outcomes 2 and 1

###### ChatGPT said:

Yes! I remember both **Learning Outcome 1** and **Learning Outcome 2** for **Designing Backend Systems**. Here’s a quick summary of each:

## ****Learning Outcome 1: Analyze System Backend (17 Hours)****

**Topics Covered:**

**Gathering FURPS Requirements**

* + Definition of key terms: Backend, System, Server, Database, OS, SDLC, API, JSON, Framework, UML
  + Understanding **FURPS** (Functionality, Usability, Reliability, Performance, Supportability)

**System Development Life Cycle (SDLC)**

* + **Phases of SDLC**
  + **SDLC Models** (Agile, Waterfall, V-Shaped, Prototyping)

**Backend Development Technologies**

* + **Programming Languages & Frameworks:**
    - Python, PHP, Java, JavaScript, Ruby & their frameworks

**System Analysis Tools**

* + Grid Chart, System Flowchart, Decision Tree, Simulation, Decision Table

**Data Gathering & FURPS Identification**

* + Identifying **Functionality, Usability, Reliability, Performance, and Supportability** requirements

**Identification of Main Objects of Backend System**

* + **Scope of Backend System**
  + **Key Components:** Database, APIs, Servers, Frameworks

**Description of System Interaction**

* + Purpose & **Main Components**:
    - Web Server, Application Server, Database Server, External APIs, Message Queues

**Report of System Backend Requirements**

* + **Executive Summary**
  + **Current State Analysis**
  + **Findings on Gaps & Issues**
  + **Recommendations**

## ****Learning Outcome 2: Develop System Structure (31 Hours)****

**Topics Covered:**

**Identification of System Design Tools**

* + UML, Algorithm, Flowchart, Data Flow Diagram (DFD), Entity-Relationship Diagram (ERD), Context Diagram, Decision Table, Use Case Diagram, Class Diagram, Decision Tree

**Identification of Hardware & Software Technology**

* + **Computer Hardware**
  + **System Software & Application Software** (Microsoft Office, Visual Paradigm, E-Draw Max, Browsers)

**Application of SSADM (Structured System Analysis and Design Methodology)**

* + **Objectives of SSADM**
  + **SSADM Techniques:**
    - Logical Data Modeling
    - Data Flow Modeling
    - Entity Behavior Modeling
  + **Stages for Implementing SSADM**

**Application of Object-Oriented Analysis & Design (OOAD)**

* + **Introduction to OOAD**
  + **Advantages & Disadvantages**
  + **Phases of Object-Oriented Software Development**
    - **Analysis**
    - **Design**
    - **Implementation**

Let me know if you need **exam questions and solutions** for these learning outcomes! 🚀

##### You said:

make final exam according all that learning outcomes 1,2,3 make that section ( A-H ) . and make multiple choice questions only

###### ChatGPT said:

# ****Final Exam: Designing Backend Systems****

### ****Total Marks: 100****

### ****Exam Structure: Multiple-Choice Questions (MCQs) Only****

## ****Section A: Gathering System Requirements (FURPS) (10 Marks)****

What does the "F" in FURPS stand for?  
a) Functionality ✅  
b) Flexibility  
c) Feasibility  
d) Framework

Which of the following is NOT a part of the FURPS model?  
a) Usability  
b) Reliability  
c) Performance  
d) Security ✅

Which factor in FURPS deals with the system's ability to handle errors and recover?  
a) Functionality  
b) Usability  
c) Reliability ✅  
d) Performance

Which of the following best defines **backend development**?  
a) Developing the visual layout of a website  
b) Managing databases, servers, and APIs ✅  
c) Designing user interfaces  
d) Writing CSS and HTML code

In system analysis, which tool helps **visualize the sequence of actions**?  
a) System Flowchart ✅  
b) Grid Chart  
c) Simulation  
d) Decision Table

## ****Section B: System Development Life Cycle (SDLC) (10 Marks)****

Which phase of SDLC involves gathering system requirements?  
a) Implementation  
b) Design  
c) Analysis ✅  
d) Testing

What is the primary goal of the **testing phase** in SDLC?  
a) Deploying the system  
b) Identifying and fixing errors ✅  
c) Writing documentation  
d) Creating the user interface

Which SDLC model is best suited for projects with **rapid changes and iterations**?  
a) Waterfall  
b) Agile ✅  
c) V-shaped  
d) Prototyping

In which SDLC model must each phase be **completed before moving to the next**?  
a) Agile  
b) Waterfall ✅  
c) Spiral  
d) RAD

What is the main purpose of **prototyping** in SDLC?  
a) Building a complete system  
b) Creating a working model to gather user feedback ✅  
c) Finalizing system documentation  
d) Writing code for the final product

## ****Section C: Backend Development Technologies (10 Marks)****

Which backend technology is commonly used with **Django**?  
a) PHP  
b) JavaScript  
c) Python ✅  
d) Ruby

Which of the following is NOT a backend programming language?  
a) Java  
b) PHP  
c) JavaScript  
d) HTML ✅

**Node.js** is primarily used for:  
a) Backend development ✅  
b) Styling web pages  
c) Creating databases  
d) Managing computer hardware

Which backend framework is commonly used with **PHP**?  
a) Laravel ✅  
b) React  
c) Django  
d) Express.js

Which of the following is a **server-side** JavaScript framework?  
a) React  
b) Angular  
c) Express ✅  
d) Bootstrap

## ****Section D: System Analysis Tools (10 Marks)****

Which system analysis tool represents decision-making rules?  
a) Grid chart  
b) Decision Tree ✅  
c) System Flowchart  
d) Simulation

What does **DFD (Data Flow Diagram)** primarily depict?  
a) User interfaces  
b) System processes and data flow ✅  
c) Hardware components  
d) Database relationships

**Which tool helps in identifying object relationships within a database?**  
a) Context Diagram  
b) Use Case Diagram  
c) Entity-Relationship Diagram (ERD) ✅  
d) Decision Table

A **grid chart** is used to:  
a) Show relationships between tasks and departments ✅  
b) Display system architecture  
c) Describe user interactions  
d) Create process flows

Which tool is best for **modeling decision logic**?  
a) System Flowchart  
b) Decision Tree ✅  
c) DFD  
d) ERD

## ****Section E: System Structure & Object-Oriented Design (10 Marks)****

Which of the following is an **object-oriented programming language**?  
a) C  
b) Java ✅  
c) COBOL  
d) HTML

What does UML stand for?  
a) Unified Module Language  
b) Universal Modeling Language  
c) Unified Modeling Language ✅  
d) User Management Layout

Which UML diagram shows **class relationships**?  
a) Use Case Diagram  
b) Class Diagram ✅  
c) Flowchart  
d) ERD

What is a **major advantage** of Object-Oriented Analysis & Design?  
a) Uses fewer lines of code  
b) Encourages code reusability ✅  
c) Eliminates software bugs  
d) Requires no documentation

Which of the following represents a **sequence of steps in a process**?  
a) Algorithm ✅  
b) Decision Table  
c) Context Diagram  
d) Class Diagram

## ****Section F: Hardware and Software Technology (10 Marks)****

Which of the following is **not** a system software?  
a) Windows OS  
b) Linux  
c) Microsoft Word ✅  
d) MacOS

What is **Visual Paradigm** used for?  
a) Hardware testing  
b) UML Diagramming ✅  
c) Word Processing  
d) Antivirus Protection

Which of the following is an **example of system hardware**?  
a) Database  
b) Operating System  
c) CPU ✅  
d) Web Browser

**Which software is mainly used for system modeling and diagramming?**  
a) Microsoft Office  
b) E-Draw Max ✅  
c) Photoshop  
d) Zoom

A **web browser** is an example of:  
a) System software  
b) Application software ✅  
c) Database software  
d) Operating system

## ****Section G: Physical Data Model & Data Flow (10 Marks)****

In a **DFD**, what does a **rectangle** represent?  
a) External Entity ✅  
b) Process  
c) Data Store  
d) Data Flow

What is the **purpose of a database foreign key**?  
a) Uniquely identifies each record  
b) Links records between tables ✅  
c) Stores system logs  
d) Manages software security

**Which document contains database design details?**  
a) System Design Document  
b) Database Design Document ✅  
c) Functional Specification Document  
d) Technical Specification Document

What does **Level 0 DFD** represent?  
a) High-level system overview ✅  
b) Detailed process breakdown  
c) System security plan  
d) API documentation

A **database index** improves:  
a) Data storage  
b) Query performance ✅  
c) System security  
d) API response time

## ****Section H: System Documentation (10 Marks)****

Which document provides **detailed technical specifications**?  
a) SDD  
b) TSD ✅  
c) FSD  
d) User Manual

What is the purpose of the **Functional Specification Document (FSD)?**  
a) Define system functionality ✅  
b) List database structures  
c) Show network architecture  
d) Describe programming syntax

Which documentation type is written for **end-users**?  
a) System Documentation  
b) User Documentation ✅  
c) Database Design Document  
d) Technical Manual

Which document outlines **project findings and recommendations**?  
a) Executive Summary ✅  
b) Class Diagram  
c) Use Case Document  
d) Entity-Relationship Diagram

**System documentation** helps:  
a) Developers ✅  
b) End-users  
c) Customers  
d) Network Administrators

**End of Exam** ✅

##### You said:

try to explain solution

###### ChatGPT said:

Here are the explanations for the answers to the multiple-choice questions in the **Final Exam: Designing Backend Systems**:

## ****Section A: Gathering System Requirements (FURPS) (10 Marks)****

**What does the "F" in FURPS stand for?**

* + ✅ **Answer: Functionality**
  + **Explanation:** FURPS is a model used to classify system requirements: **Functionality, Usability, Reliability, Performance, and Supportability**. "F" specifically stands for **Functionality**, which includes features, capabilities, and security requirements of a system.

**Which of the following is NOT a part of the FURPS model?**

* + ✅ **Answer: Security**
  + **Explanation:** FURPS covers **Functionality, Usability, Reliability, Performance, and Supportability**. Security is important but falls under **Functionality** in FURPS, not as a separate category.

**Which factor in FURPS deals with the system's ability to handle errors and recover?**

* + ✅ **Answer: Reliability**
  + **Explanation:** Reliability measures how **stable and error-free** the system is, including handling failures and ensuring **high availability**.

**Which of the following best defines backend development?**

* + ✅ **Answer: Managing databases, servers, and APIs**
  + **Explanation:** Backend development refers to **server-side programming**, handling **database interactions, APIs, and logic** behind applications.

**In system analysis, which tool helps visualize the sequence of actions?**

* + ✅ **Answer: System Flowchart**
  + **Explanation:** A **System Flowchart** represents **the logical flow of data and processes**, making it easier to understand system behavior.

## ****Section B: System Development Life Cycle (SDLC) (10 Marks)****

**Which phase of SDLC involves gathering system requirements?**

* + ✅ **Answer: Analysis**
  + **Explanation:** The **Analysis phase** focuses on **gathering system requirements, identifying problems, and defining system objectives**.

**What is the primary goal of the testing phase in SDLC?**

* + ✅ **Answer: Identifying and fixing errors**
  + **Explanation:** Testing helps detect **bugs, security vulnerabilities, and performance issues** before deployment.

**Which SDLC model is best suited for projects with rapid changes and iterations?**

* + ✅ **Answer: Agile**
  + **Explanation:** Agile allows **continuous changes and iterative development**, making it ideal for **dynamic projects**.

**In which SDLC model must each phase be completed before moving to the next?**

* + ✅ **Answer: Waterfall**
  + **Explanation:** The **Waterfall model** follows a **sequential** approach, where each phase **must be completed before the next begins**.

**What is the main purpose of prototyping in SDLC?**

* ✅ **Answer: Creating a working model to gather user feedback**
* **Explanation:** A **prototype** is a preliminary version that helps **stakeholders visualize the system and provide feedback** before full development.

## ****Section C: Backend Development Technologies (10 Marks)****

1. **Which backend technology is commonly used with Django?**

* ✅ **Answer: Python**
* **Explanation:** **Django** is a **Python-based web framework** used for **backend development**.

1. **Which of the following is NOT a backend programming language?**

* ✅ **Answer: HTML**
* **Explanation:** **HTML** is used for **structuring web pages** and is not a **backend programming language**.

1. **Node.js is primarily used for:**

* ✅ **Answer: Backend development**
* **Explanation:** **Node.js** allows JavaScript to be used for **server-side development**.

1. **Which backend framework is commonly used with PHP?**

* ✅ **Answer: Laravel**
* **Explanation:** **Laravel** is a **PHP framework** used for developing modern web applications.

1. **Which of the following is a server-side JavaScript framework?**

* ✅ **Answer: Express**
* **Explanation:** **Express.js** is a **backend framework** used with **Node.js** for building APIs and web applications.

## ****Section D: System Analysis Tools (10 Marks)****

1. **Which system analysis tool represents decision-making rules?**

* ✅ **Answer: Decision Tree**
* **Explanation:** **Decision Trees** help visualize **decision-making paths and outcomes**.

1. **What does a DFD (Data Flow Diagram) primarily depict?**

* ✅ **Answer: System processes and data flow**
* **Explanation:** A **DFD** shows how **data moves** between processes, databases, and external entities.

1. **Which tool helps in identifying object relationships within a database?**

* ✅ **Answer: Entity-Relationship Diagram (ERD)**
* **Explanation:** **ERDs** represent **database objects (tables, relationships, keys, etc.)**.

1. **A grid chart is used to:**

* ✅ **Answer: Show relationships between tasks and departments**
* **Explanation:** **Grid charts** map **tasks** against **departments or roles** in a system.

1. **Which tool is best for modeling decision logic?**

* ✅ **Answer: Decision Tree**
* **Explanation:** **Decision Trees** are used for **logic-based decision-making models**.

## ****Section E: System Structure & Object-Oriented Design (10 Marks)****

1. **Which of the following is an object-oriented programming language?**

* ✅ **Answer: Java**
* **Explanation:** **Java is an object-oriented programming language**, supporting **classes and objects**.

1. **What does UML stand for?**

* ✅ **Answer: Unified Modeling Language**
* **Explanation:** UML is used for **visualizing software designs**.

1. **Which UML diagram shows class relationships?**

* ✅ **Answer: Class Diagram**
* **Explanation:** **Class Diagrams** show **relationships between classes, attributes, and methods**.

1. **What is a major advantage of Object-Oriented Analysis & Design?**

* ✅ **Answer: Encourages code reusability**
* **Explanation:** **OOP promotes reusability** through **inheritance, polymorphism, and encapsulation**.

1. **Which of the following represents a sequence of steps in a process?**

* ✅ **Answer: Algorithm**
* **Explanation:** **Algorithms define a step-by-step approach to solving a problem**.

## ****Section F: Hardware and Software Technology (10 Marks)****

1. **Which of the following is NOT a system software?**

* ✅ **Answer: Microsoft Word**
* **Explanation:** **System software** includes **operating systems** like **Windows, Linux, MacOS**, but Microsoft Word is an **application software**.

1. **What is Visual Paradigm used for?**

* ✅ **Answer: UML Diagramming**
* **Explanation:** **Visual Paradigm is used for creating UML diagrams**.

1. **Which of the following is an example of system hardware?**

* ✅ **Answer: CPU**
* **Explanation:** **Hardware components include CPU, RAM, and storage devices**.

1. **Which software is mainly used for system modeling and diagramming?**

* ✅ **Answer: E-Draw Max**
* **Explanation:** **E-Draw Max is a tool for creating system diagrams**.

1. **A web browser is an example of:**

* ✅ **Answer: Application software**
* **Explanation:** **Web browsers** like Chrome and Firefox are **application software** used for internet access.

## ****Final Thoughts****

This exam covers **backend system analysis, development, and design** using **FURPS, SDLC, UML, backend technologies, and system documentation**. The solutions provided **explain why each answer is correct**, ensuring a **clear understanding of backend system design concepts**. 🚀