Name: Paramprakash Makwana

Module 1 – Overview of IT Industry (Lab)

1.1) Write a simple "Hello World" program in two different programming languages of your choice. Compare the structure and syntax.

Ans:-C Language:

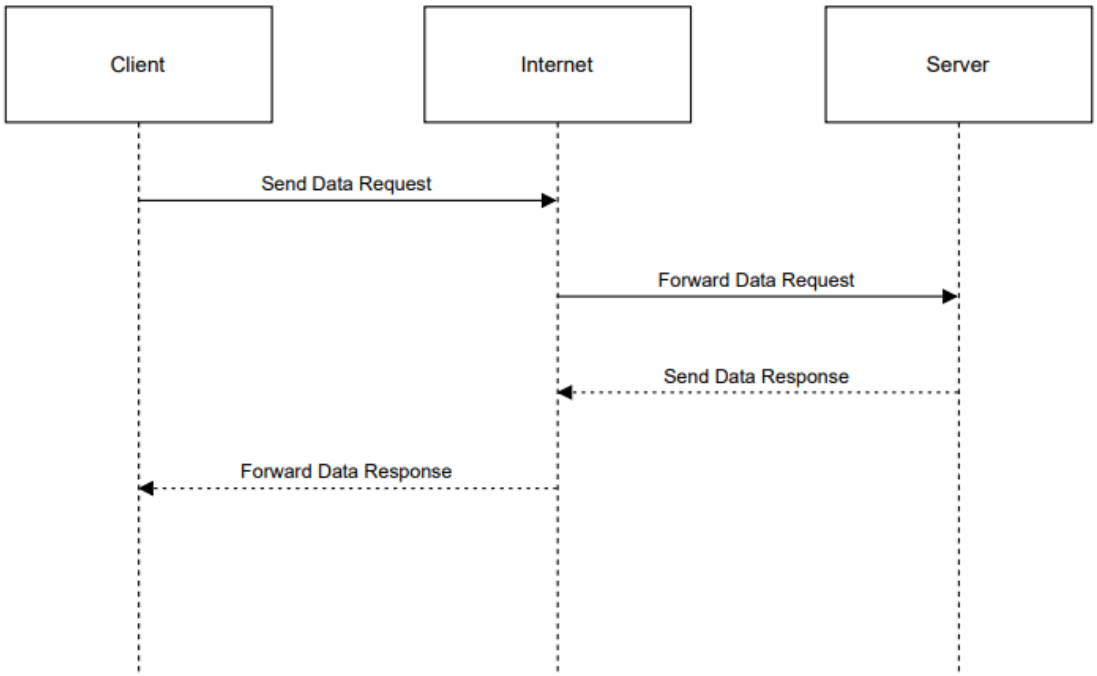
```
#include<stdio.h>
#include<conio.h>
void main()
{
   printf("Hello World");
   getch();
}
```

C++ Language:

```
#include<iostream>
using namespace std;
int main()
{
   cout<<"Hello World";
   return 0;
}
```

1.2) Research and create a diagram of how data is transmitted from a client to a server over the internet.

Ans:-



1.3) Research different types of internet connections (e.g., broadband, fiber, satellite) and list their pros and cons.

Ans:- Broadband (DSL/Cable):
Pros: Affordable, widely available.
Cons: Slower than fiber, speed drops at peak times.

Fiber Optic:
Pros: Very fast, reliable, low latency.
Cons: Limited availability, higher cost.

Satellite:
Pros: Works in remote areas.
Cons: High latency, weather issues, expensive.

Mobile Data (4G/5G):
Pros: Portable, easy setup, good speed (especially 5G).
Cons: Data limits, speed depends on signal.

1.4) Identify and explain three common application security vulnerabilities. Suggest possible solutions.

Ans:- SQL Injection:
Attackers inject SQL code through inputs.
Solution: Use prepared statements and input validation.

Cross-Site Scripting (XSS):
Malicious scripts run in users' browsers.
Solution: Escape output and validate input.

Weak Authentication:
Poor password or login security.
Solution: Strong passwords, hashing, and multi-factor authentication.

1.5) Identify and classify 5 applications you use daily as either system software or application software.

Ans:- Windows OS – System software – controls hardware and manages the system

Google Chrome – Application software – used for internet browsing

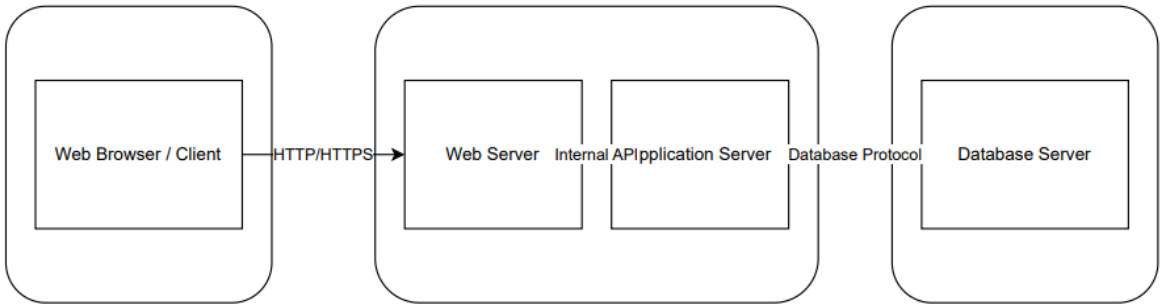
Microsoft Word – Application software – used to create documents

WhatsApp – Application software – used for messaging and calls

VLC Media Player – Application software – used to play media files

1.6) Design a basic three-tier software architecture diagram for a web application.

Ans:-



1.7) Create a case study on the functionality of the presentation, business logic, and data access layers of a given software system.

Ans:- Case Study : Student Management System

- Presentation Layer:
  User interface for input/output (login, forms, view results).
- Business Logic Layer:
  Processes data and applies rules (validate details, calculate grades, check permissions).
- Data Access Layer:
  Handles database operations (insert, update, fetch student records).

1.8) Create a list of software you use regularly and classify them into the following categories: system, application, and utility software.

Ans:- 1) System Software

- Windows 11 – Operating system that manages hardware and resources

2) Application Software

- Google Chrome – Web browsing
- Microsoft Word – Document creation
- WhatsApp – Messaging and calls
- VLC Media Player – Audio/video playback

3) Utility Software

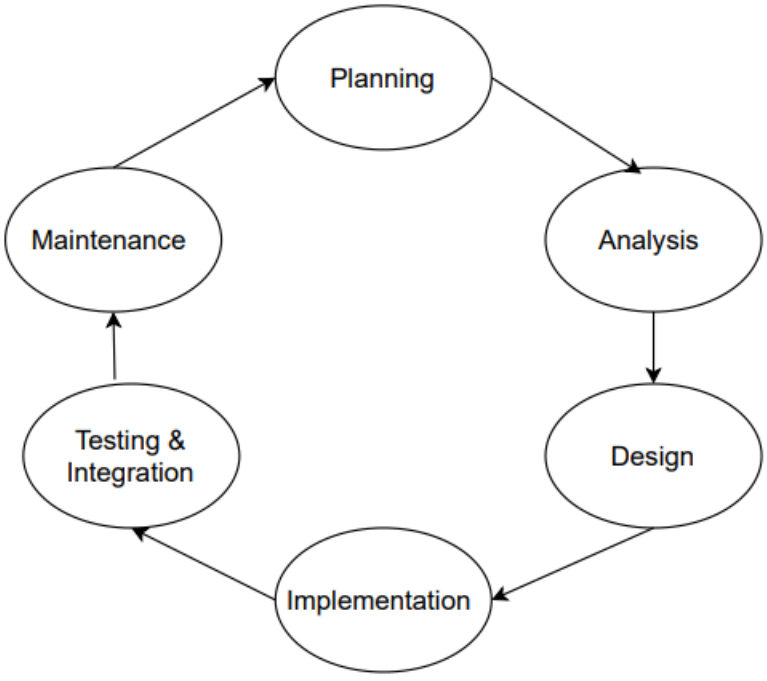
- WinRAR – File compression and extraction

1.9) Write a report on the various types of application software and how they improve productivity.

Ans:- 1. Word Processing: Microsoft Word, Google Docs – Speeds up document creation and collaboration.
2. Spreadsheet: Excel, Google Sheets – Simplifies data analysis, calculations, and reporting.
3. Presentation: PowerPoint, Keynote – Enhances communication of ideas visually.
4. Database: Access, MySQL – Quick data storage, retrieval, and management.
5. Communication: Slack, Zoom – Improves teamwork and reduces delays.
6. Multimedia: Photoshop, VLC – Supports content creation and professional presentations.

1.10) Create a flowchart representing the Software Development Life Cycle (SDLC).

Ans:-



1.11) Write a requirement specification for a simple library management system.

Ans:- Library Management System – Requirement Specification

Introduction**:** Manage books, members, and book issue/return efficiently.

Functional Requirements:

- User Management: Add, update, delete members.
- Book Management: Add, update, delete, and search books.
- Issue/Return: Record book issues, returns, and due dates.
- Inventory**:** Track available copies and notify if out of stock.
- Reports**:** Generate issued books, overdue books, and member activity reports.

Non-Functional Requirements:

- Easy-to-use interface
- Handle 500+ books and 200+ members
- Secure access for librarians
- Persistent database storage
- Scalable for future modules

1.12) Perform a functional analysis for an online shopping system.

Ans:- Functional Analysis: Online Shopping System

- User Management**:** Register, login, update profile.
- Product Management**:** Browse, search, filter products; track inventory.
- Shopping Cart & Wishlist**:** Add/remove items, manage quantities, view totals.
- Order Processing**:** Checkout, payment, order confirmation.
- Shipping & Delivery**:** Track orders, process returns/refunds.
- Reviews & Ratings**:** Rate and review products.
- Admin Functions**:** Manage products, orders, and users.
- Reports & Analytics**:** Sales, inventory, and user activity reports.

1.13) Develop test cases for a simple calculator program.

Ans:- Test Case 1:- Addition: 5 + 3 → Expected Output: 8

Test Case 2:- Addition with negative number: -2 + 7 → Expected Output: 5

Test Case 3:- Subtraction: 10 - 4 → Expected Output: 6

Test Case 4:- Subtraction resulting in negative: 5 - 10 → Expected Output: -5

Test Case 5:- Multiplication: 3 × 4 → Expected Output: 12

Test Case 6:- Multiplication with negative number: -3 × 6 → Expected Output: -18

Test Case 7:- Division: 12 ÷ 4 → Expected Output: 3

Test Case 8:- Division by zero: 5 ÷ 0 → Expected Output: Error / Cannot divide by zero

Test Case 9:- Division with negative number: -10 ÷ 2 → Expected Output: -5

Test Case 10:- Mixed operations: (5 + 3) × 2 → Expected Output: 16

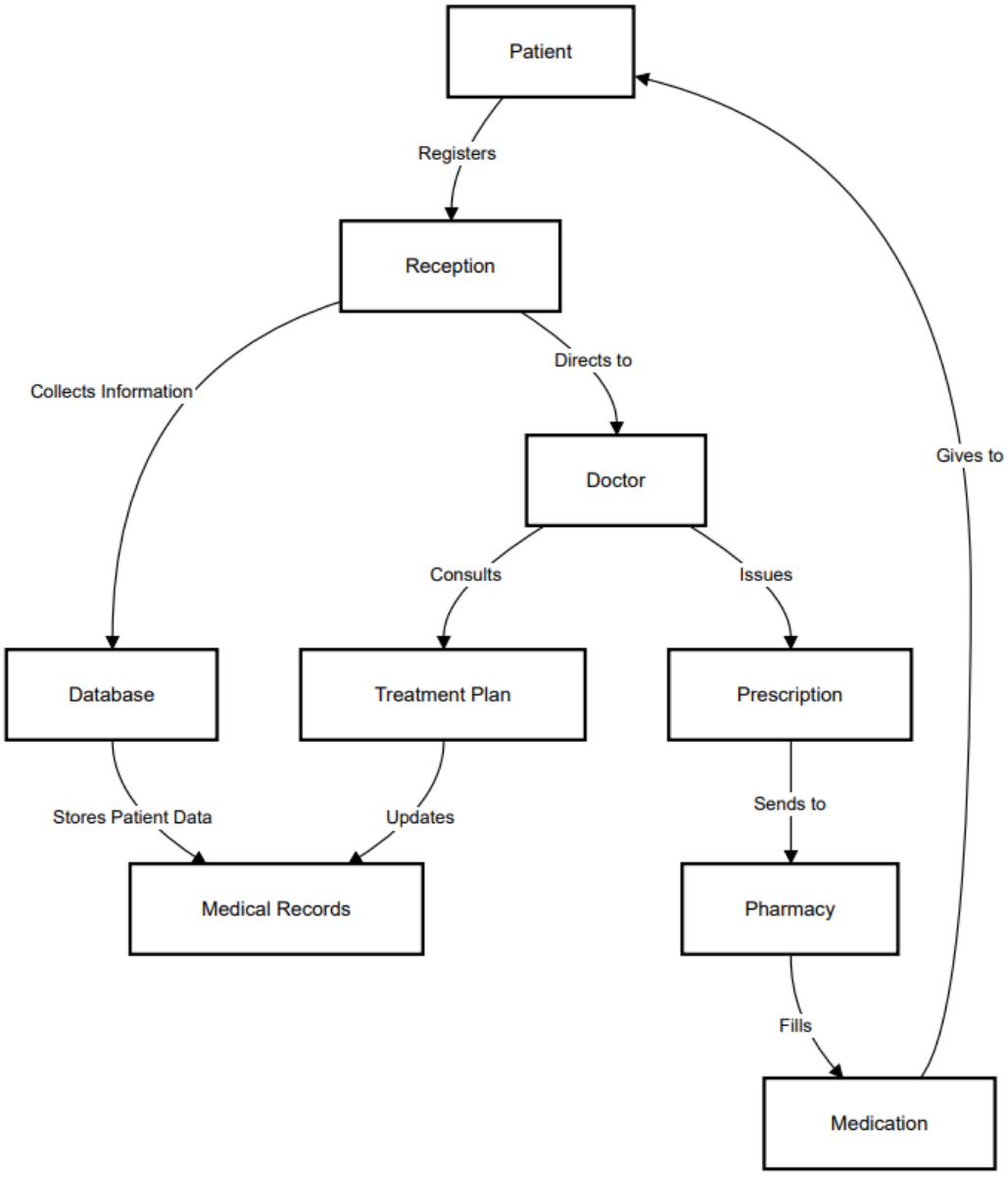
1.14) Document a real-world case where a software application required critical maintenance.

Ans:- Case Study: Healthcare.gov Critical Maintenance

- Background**:** Healthcare.gov, the U.S. health insurance portal, launched in 2013.
- Problem**:** Website crashed, had slow responses, and user registration errors due to integration issues and high traffic.
- Maintenance Actions**:** Audited the system, fixed code bugs, optimized servers and databases, performed load testing, and added real-time monitoring.
- Outcome**:** Website stabilized, handled large user traffic, and enrollment improved.
- Lesson**:** Critical maintenance is vital for high-traffic systems; proper testing and monitoring prevent failures.

1.15) Create a DFD for a hospital management system.

Ans:-



1.16) Draw a flowchart representing the logic of a basic online registration system.

Ans:-