# ***full stack assignment set***

* **Module 1-overview of it industry**

**1.what is a program ?**

* A program is a set of instructions that a computer follows ot perform a specific task.
* **1.Lab exercise** : write a simle “hellow world” program in two different programming languages of your choice. Compare the structure and syntax.

**1.Html**

<html>

<head> <title>assignment program </title></head>

<body>

<H1> HELLO WORLD </H1>

</body>

</html>

**2.C languages**

#include<stdio.h>

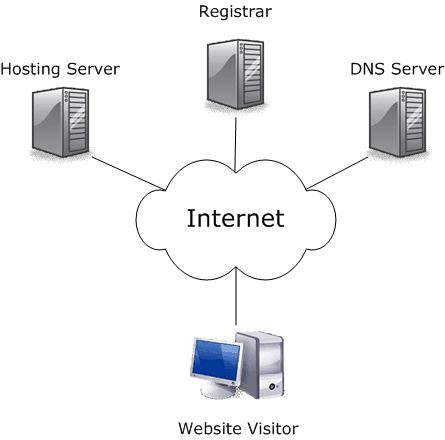
main()

{

printf("hello world");

}

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

****

**3. Simulate HTTP and FTP requests using command line tools.**

* **HTTP is primarily used for web communication, and tools**

**like curl and wget allow you to simulate requests to interact with web**

**servers.**

* **FTP is used for file transfers, and command line tools**

**like ftp and lftp enable you to connect to FTP servers to manage files.**

**4. Identify and explain three common application security**

**vulnerabilities. Suggest possible**

**1. SQL Injection**

**Explanation: Attackers insert malicious SQL code into input fields to access or manipulate the database.**

**Solution: Use prepared statements (parameterized queries), validate and sanitize all user inputs.**

**2. Cross-Site Scripting (XSS)**

**Explanation: Malicious scripts are injected into web pages, which execute in users' browsers and steal data or perform actions.**

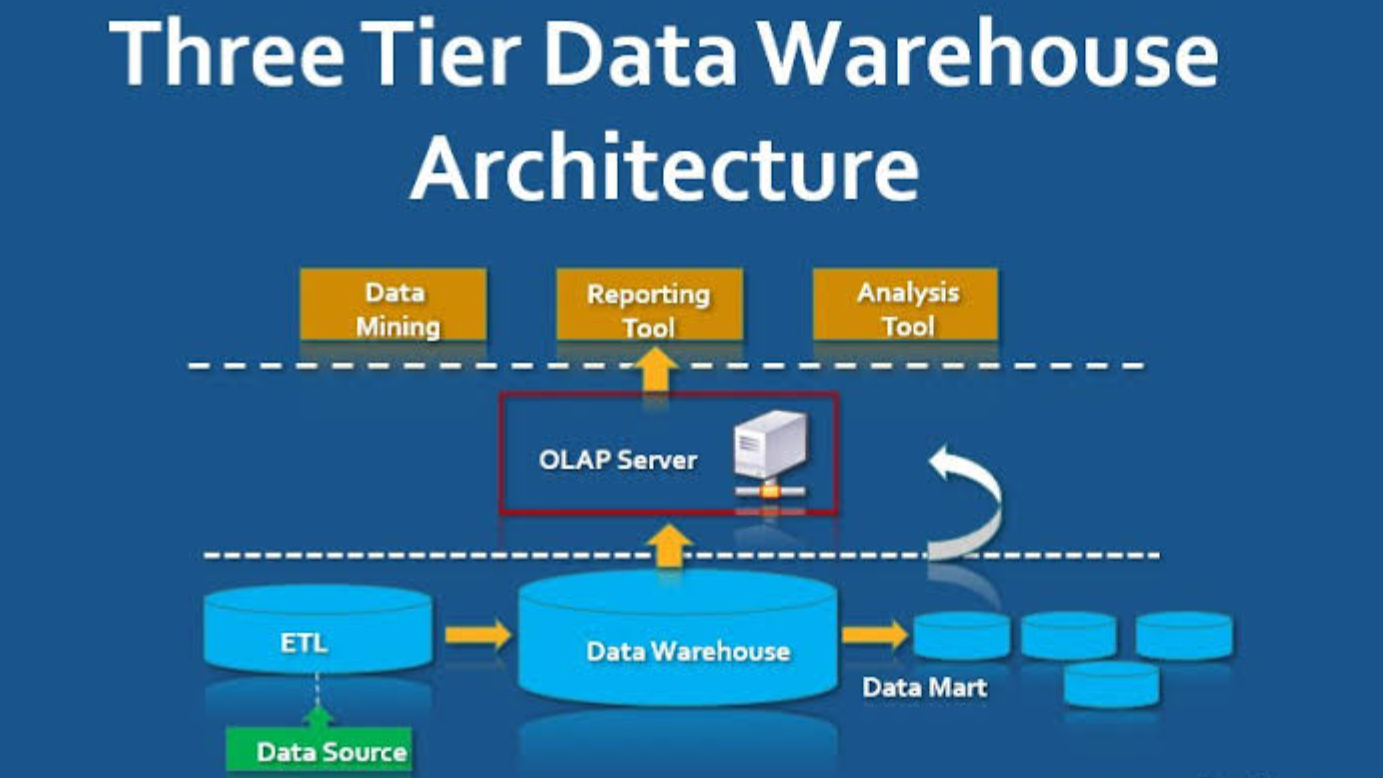
**Solution: Encode output data, use Content Security Policy (CSP), and validate input to block scripts.**

**3. Cross-Site Request Forgery (CSRF)**

**Explanation: An attacker tricks a user into performing unintended actions on a trusted website where they're logged in.**

**Solution: Use anti-CSRF tokens, verify request origins, and implement user authentication checks.**

**5.**

****

**6. Write and upload your first source code file to GitHub.**

* **Create GitHub Account: Sign up at GitHub.**
* **Create Repository: Set up a new repository on GitHub.**
* **Install Git: Download and install Git on your machine.**
* **Set Up Git: Configure your username and email in Git.**
* **Create Local Repository: Initialize a new Git repository locally.**
* **Add Source Code File: Create and edit your source code file.**
* **Stage Changes: Use git add to stage your file.**
* **Commit Changes: Commit your changes with a message.**
* **Link to GitHub: Add the remote repository URL.**
* **Push Changes: Push your local commits to GitHub.**
* **Verify Upload: Check your GitHub repository for the file.**

**7. Create a GitHub repository and document how to commit and push code**

**changes.**

* **Creating a GitHub Repository**
* **1. Log In: Access your GitHub account.**
* **2. New Repository: Click the "+" icon and select "New repository."**
* **3. Repository Details: Enter a name, description, and choose visibility.**
* **4. Create Repository: Click "Create repository" to finalize.**
* **Committing and Pushing Code Changes**
* **1. Clone Repository: Use git clone to copy the repository locally.**
* **2. Make Changes: Edit or create files in your local repository.**
* **3. Stage Changes: Use git add filename to stage files.**
* **4. Commit Changes: Commit with git commit -m "Your message here".**
* **5. Push Changes: Push to GitHub using git push origin main.**
* **6 Verify Changes: Refresh your GitHub repository to see updates.**

**8. Create a student account on Github and collaborate on a small project**

**with a classmate.**

* **Verify your email and complete your GitHub profile setup.**
* **Apply for GitHub Student Pack at education.github.com.**
* **Create a new repository for your project from your GitHub**

**dashboard.**

* **Initialize the repository with a README file and choose**
* **visibility (public/private).**
* **Add your classmate as a collaborator under Settings >**

**Collaborators.**

* **Clone the repository to your local system using git clone**

**<repo\_url>.**

* **Work on the project locally and commit changes using git add.,**

**git commit -m "", and git push.**

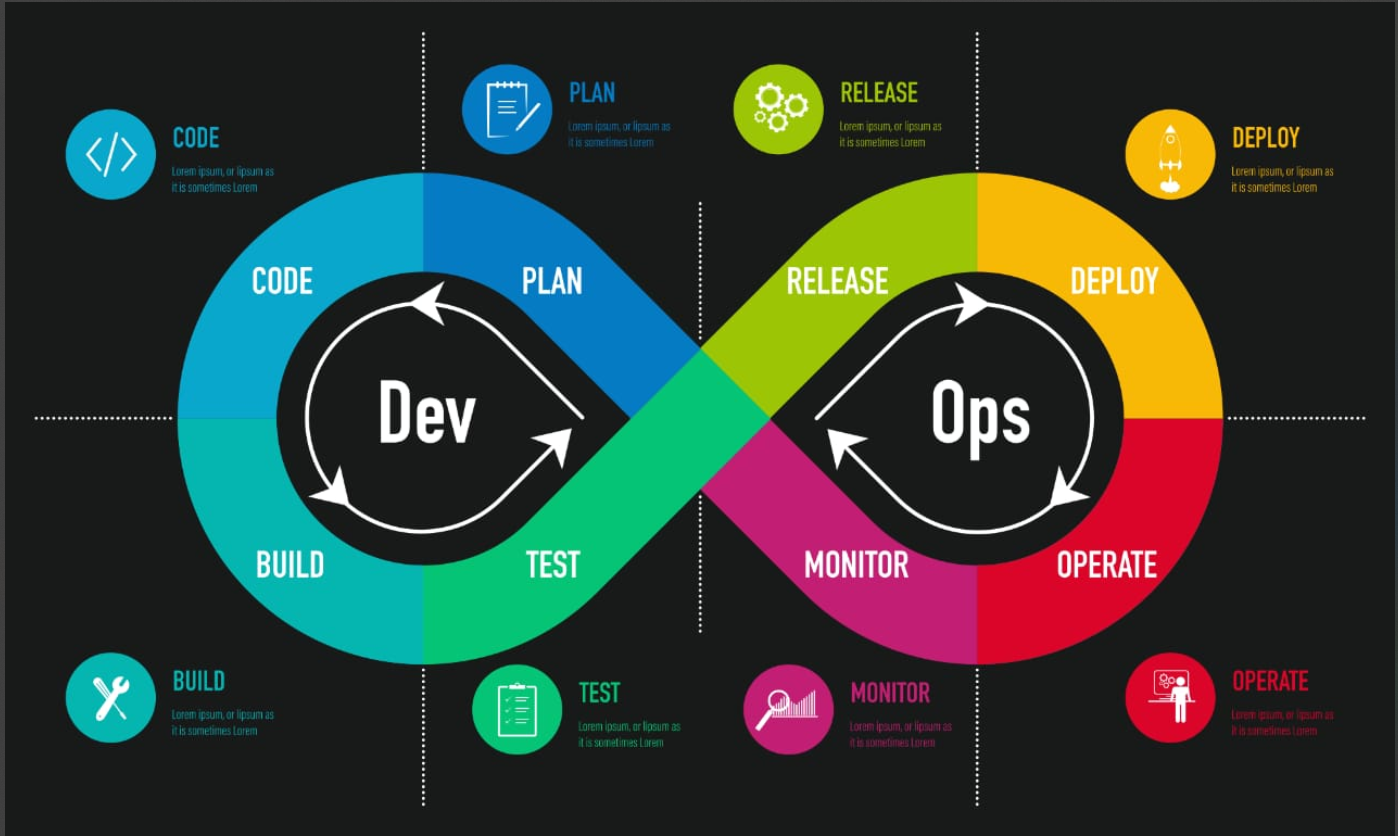
* **Your classmate can pull changes, add their contributions, and**

**push updates.**

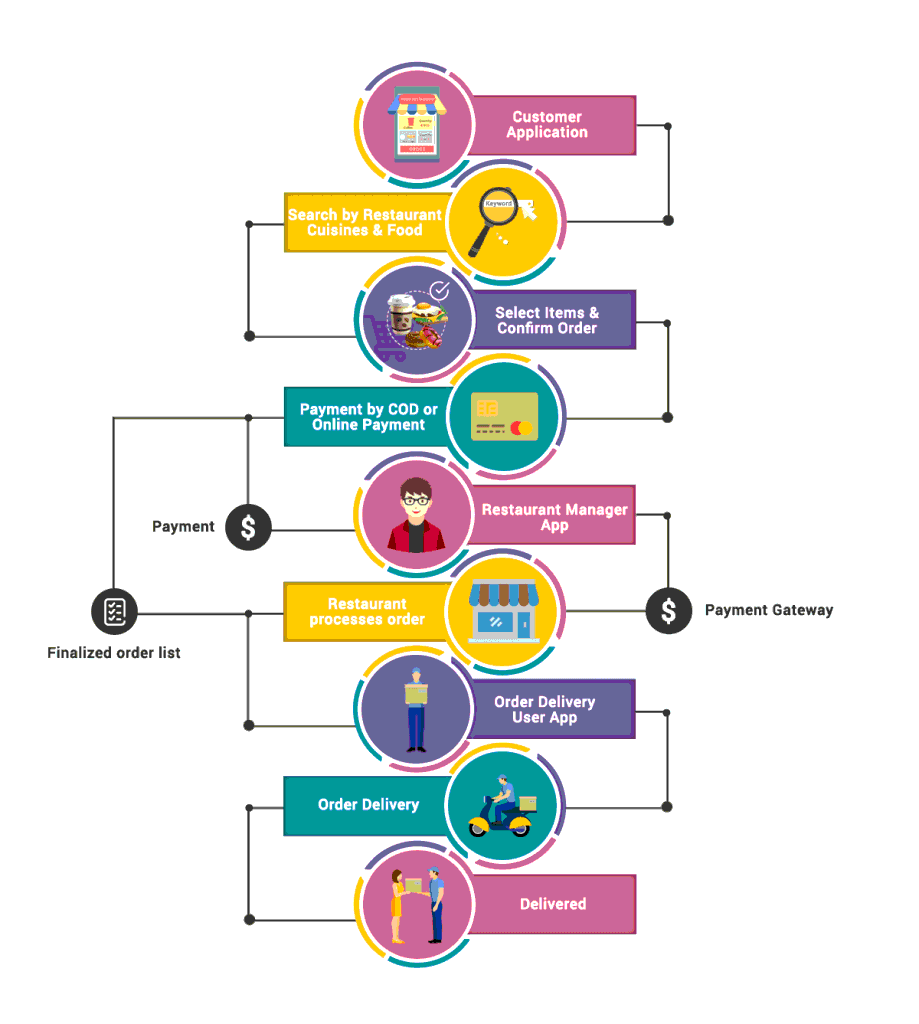
* **Use GitHub issues or pull requests to manage tasks and**

**collaborate effectively.**

**9.**

****

**10.**

****

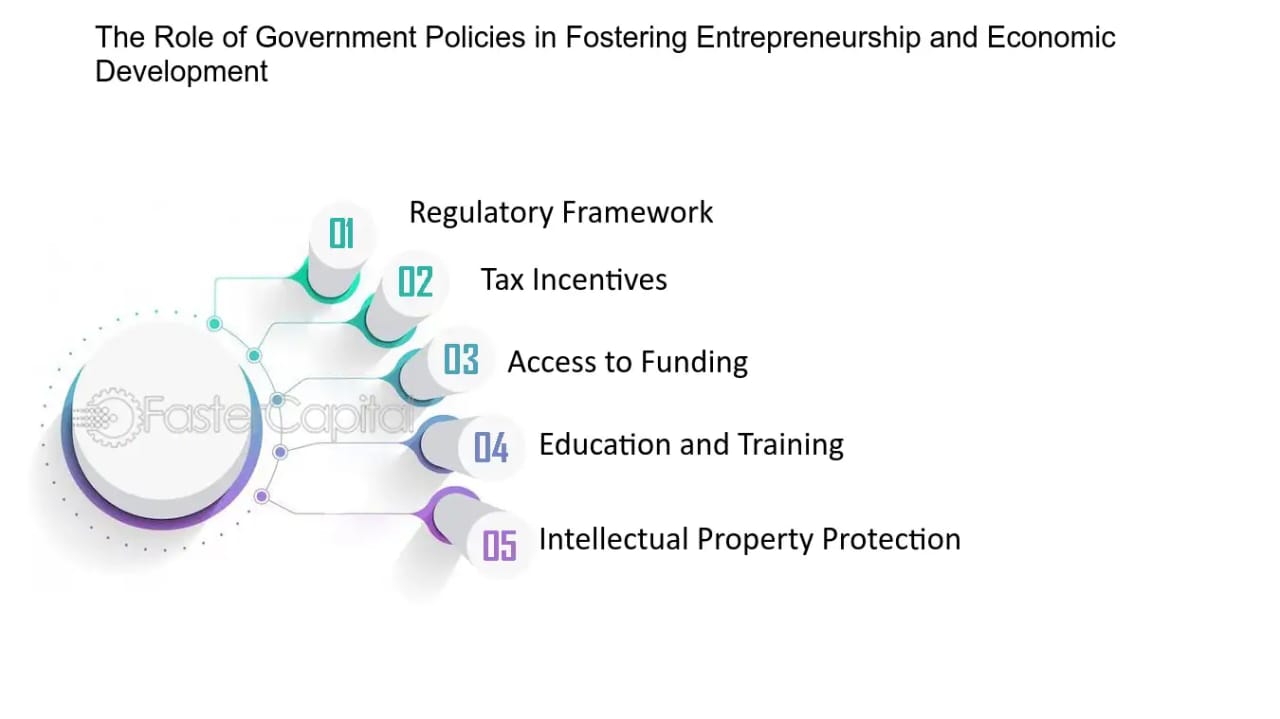
**11.**

****

**12.**

****

**13.**

****