Web Hosting & Testing Platform Hierarchy (with Examples & Diagrams)

1. **Web Server**

    - Serves static files or forwards requests.
    - Examples: `Apache`, `Nginx`, `Caddy`

    Diagram: `[User Browser] --> [Nginx] --> [Static HTML/CSS/JS Files]`

2. **Web Stack** (Software Stack)

    - Combines server, DB, and runtime.
    - Examples:
        - LAMP (Linux, Apache, MySQL, PHP)
        - MEAN (MongoDB, Express.js, Angular, Node.js)

    Diagram: `[Browser] --> [Apache] --> [PHP App] --> [MySQL DB]`

3. **Application Hosting Platform**

    - Adds management tools (e.g. Docker, systemd).
    - Example: Flask app in Docker with volume-mounted config.

    Diagram: `[Browser] --> [Nginx] --> [Docker: Flask App] --> [PostgreSQL]`

4. **Web Application Infrastructure**

    - Full stack with provisioning and monitoring.

    Example Tools:

    - Provisioning: `Ansible`, `Terraform`
    - Monitoring: `Prometheus`, `Grafana`
    - Load Balancer: `HAProxy`

    Diagram: `[Users] | [HAProxy] | [Nginx] --> [App Server] --> [Database] | [Prometheus/Grafana]`

5. **Staging/Testing Environment**

    - Replica of production with automated tests.

    Example:

    - Django staging server with pytest and CI/CD

    Diagram: `[Dev Push] --> [GitHub Actions] --> [Test: Pytest/Selenium] --> [Deploy: Docker Compose]`

6. **Security Testbed**

    - Adds vulnerability and config scanning.

    Example Tools:

    - Scanning: `OWASP ZAP`, `Nikto`
    - Audit: `Lynis`, `OpenSCAP`

    Diagram: `[Web App] <-- [ZAP Scan] <-- [OpenVAS/Nikto]`

7. **DevSecOps Platform**

    - CI/CD + IaC + Security + Compliance

    Toolchain:

    - IaC: Terraform, Ansible
    - CI/CD: GitLab CI, Jenkins
    - Security: Vault, Trivy, Bandit

    Diagram: `[Code Commit] --> [CI/CD Pipeline] --> [Lint/Test] --> [Security Scan] --> [Deploy to Staging]`

8. **Integrated Web Application Platform**

    - All-in-one infrastructure

    Example:

    - Kubernetes + GitOps + ArgoCD + Prometheus

    Diagram: `[Dev Commit] | [Git Repo] --> [ArgoCD] --> [K8s Cluster] | [Web App Pods] | [Prometheus/Grafana]`

Terms for Experimentation Environments

- **Testbed**: Structured for controlled testing
- **Web Lab**: General-purpose sandbox
- **Staging Env**: Pre-deployment replica
- **Sandbox**: Isolated for safe testing
- **Home Lab**: Personal or local experimentation

Suggested: Local Web Infrastructure Testbed

Frontend vs Backend Division

Frontend

- Languages: HTML, CSS, JS

- Tools: React, Vue, Svelte
- Runs in browser

Backend

- Languages: Python, Node.js, PHP
- Tools: Django, Express, Flask
- Runs on server

Shared Infra

- Reverse Proxy: Nginx, Traefik
- DB: PostgreSQL, MongoDB
- CI/CD: Jenkins, GitHub Actions

Example Flow: `[Browser] | [HTML/CSS/JS] --> [API Call] --> [Backend App] --> [Database]`