KWARA STATE POLYTECHNIC, ILORIN

INSTITUTION OF INFORMATION COMMUNICATION TECHNOLOGY

DEPARTMENT OF COMPUTER SCIENCE

**A TECHNICAL REPORT OF THE STUDENTS INDUSTRIAL
WORK EXPERIENCE SCHEME
(SIWES)**

UNDERTAKEN AT:

**G-TECHNOLOGY TRAINING INSTITUTE
ILORIN KWARA STATE**

SUBMITTED BY:

ADEBIYI TUNJI OLUWATAYO

ND/23/COM/PT/0075

SUBMITTED TO:

THE DEPARTMENT OF COMPUTER SCIENCE

PERIOD OF ATTACHMENT:

AUGUST - NOVEMBER 2024

DEDICATION

I dedicate my industrial training report to almighty god, who has given me the grace to participate in the SIWES program, to my parents and as many that have contributed greatly to the success of my industrial training.

ACKNOWLEDGEMENT

I thank God who has seen me throughout my SIWES program and also thank my Industrial based supervisor who guided me through My Industrial training. I also send out my appreciation to my lecturers, friends and Co-workers for their moral support. My special thanks to my wonderful and lovely parents   Mr. & Mrs. ADEBIYI   who were there for me in terms of care, prayers, financial support and others.

ABSTRACT

This report presents a comprehensive overview of the implementation and challenges of web development and cyber-security during the Student Industrial Work Experience Scheme (SIWES). As digital platforms continually evolve, the need for robust web applications combined with effective security measures has never been more critical.

The internship experience focused on acquiring practical skills by developing web applications while simultaneously addressing potential cyber-security threats, such as data breaches and unauthorized access. Various web development technologies, including HTML, CSS, and JavaScript were utilized to create dynamic and responsive user interfaces. Concurrently, cyber-security principles were integrated into the development process, employing measures such as encryption, secure coding practices, and vulnerability assessments to safeguard applications.

This report outlines the methodologies employed, the integration of cyber-security practices during the development process, and highlights key findings and recommendations for best practices in the web development life cycle. Ultimately, the experience underscored the importance of a holistic approach that encompasses both development and security, aiming to create resilient web applications in an ever-changing threat landscape.

TABLE OF CONTENT

<div align="center">**CHAPTER ONE**</div>

1.0   INTRODUCTION

The Student Industrial Work Experience Scheme (SIWES) is program which forms part of the academic standards in the degree program for Nigerian Universities. The Federal Government of Nigeria introduced the policy on Industrial training, called the Student Industrial Work Experience Scheme (SIWES) IN 1974. The Industrial Training Fund (ITF) is in charge of this program which is under the umbrella of the Ministry of Education. SIWES was designed to help students acquire the necessary practical education/experience in their fields of study and other related professions.

This is an effort which was created in order to compliment the theory taught in the classrooms of the Nigerian tertiary institutions. This objective of the program is exposing students to the use of various machines and equipment's, professional work methods and ways of safeguarding the work areas in industries as well as other organizations. SIWES was established to impact practical knowledge to students with respect to their various disciplines.

This training is funded by the Federal Government of Nigeria and coordinated by the Industrial Training Fund (ITF) and the National Universities Commission (NUC). The SIWES program involves the student, the Universities and the industries.

1.1   OBJECTIVES OF SIWES:

- The program teaches the student on how to interact effectively with other workers and supervisors under various conditions in the organization
- It will help students to gain increased maturity and understanding of the workplace
- The students will have chance to evaluate companies for which they might wish to work
- It exposes students to work methods and techniques in handling equipment and machines that may not be available in educational institution.
- The program provides students with an opportunity to apply their knowledge in real work and actual practice.
- SIWES increases a student sense of responsibility.
- SIWES provides students the opportunity to test their interest in a particular career before permanent commitments are made.

CHAPTER TWO

2.0   DESCRIPTION OF THE ESTABLISHMENT OF ATTACHMENT

G-tech is a fully indigenous Nigerian company that provides clients with high-quality, cost-effective, and innovative IT and security solutions. G-tech prioritizes ultimate client satisfaction. The Company has established a respected standard over the years due to her extensive technical knowledge and ability in project management and implementation.

2.1   VISION OF THE COMPANY

To be a leading institution in technology education, empowering individuals with the skills, knowledge, and confidence to innovate and excel in a rapidly evolving digital landscape. We envision a future where our graduates are at the forefront of technological advancement, contributing to sustainable development and impactful solutions for global challenges.

2.2 THE OBJECTIVES OF THE COMPANY

1. To offer a comprehensive and up-to-date curriculum that encompasses the latest technologies and industry trends to ensure our students acquire relevant competencies.
2. To provide practical, hands-on training experiences through labs, workshops, and projects that enable students to apply theoretical knowledge in real-world scenarios.
3. To promote accessibility in technology education by offering diverse programs that cater to individuals from various backgrounds, ensuring an inclusive learning environment.

2.3 CORE VALUES

- Honesty
- Services
- Commitment
- Excellence
- Professionalism

2.4 COMPANY AREA OF SPECIALIZATION

With a team of professional Computer programmers, web development, data science, Graphic designers, UI/UX, and cyber-security, the Company has a reputable recognition in the following areas:

1. Software Development

Software development involves the process of designing, coding, testing, and maintaining software applications. This field encompasses a variety of programming languages, frameworks, and methodologies. Software developers work on applications that run on various platforms, including desktop, mobile, and cloud environments. Key concepts include software engineering principles, version control, agile methodologies, and user experience design. Developers may specialize in different areas, such as front-end development (user interface) or back-end development (server-side logic).

2. Web Development

Web development focuses specifically on the creation and maintenance of websites and web applications. It includes both front-end development, which involves the design and interactivity of the website that users see, and back-end development, which deals with server-side logic, databases, and application integration. Technologies such as HTML, CSS, JavaScript, and various frameworks (e.g., React, Angular, and Node.js) are commonly used.

Web developers must ensure websites are functional, responsive, and accessible across different devices and browsers.

3. Cyber Security

Cyber-security is the field dedicated to protecting systems, networks, and data from cyber threats and attacks. It involves identifying vulnerabilities, implementing security measures, and responding to incidents. Cyber-security specialists work to safeguard information through various practices, including encryption, firewalls, intrusion detection systems, and security protocols. They also focus on compliance with regulations and standards to ensure data privacy. With the growing prevalence of cyber threats, the demand for cyber-security professionals continues to rise.

4. Data Science

Data science is the study and analysis of complex data sets to derive insights, make informed decisions, and drive business strategy. This interdisciplinary field combines statistics, mathematics, programming, and domain expertise to extract meaning from structured and unstructured data. Data scientists use tools and programming languages like Python, R, and SQL, along with machine learning techniques and data visualization tools, to analyze trends and patterns. The profession is crucial in various industries, enabling organizations to harness the power of data for predictive modeling, customer segmentation, and more.

5. Graphic Design

Graphic Design is a field of computer science that involves the visual communication of ideas and messages through the use of typography, color, and images. It involves the creation of visual elements such as logos, graphics, icons, and graphics for websites, magazines, newspapers, and other media.

6. UI/UX DESIGN

Our company specializes in creating intuitive and engaging User Interface (UI) and User Experience (UX) designs that enhance digital interactions. We focus on understanding user behavior, preferences, and needs to develop solutions that are not only aesthetically pleasing but also highly functional.

2.5   STRUCTURE CHART OF G-TECHNOLOGY TRAINING INSTITUTE

```
                          ┌─────────────┐
                          │     CEO     │
                          └──────┬──────┘
                                 │
                          ┌──────┴──────┐
                          │   MANAGER   │
                          └──────┬──────┘
                                 │
┌──────────────┐ ┌──────────────┐ │ ┌──────────────┐ ┌──────────────┐
│   HEAD OF    │ │   HEAD OF    │ │ │   HEAD OF    │ │   HEAD OF    │
│     WEB      │─│   GRAPHICS   │─┼─│    CYBER     │─│   SOFTWARE   │
│ DEVELOPMENT  │ │     AND      │ │ │   SECURITY   │ │  DEPERTMENT  │
│  DEPERTMENT  │ │   ADVERTING  │ │ │  DEPARTMENT  │ └──────────────┘
└──────────────┘ └──────────────┘ │ └──────────────┘
                                 │
                          ┌──────┴──────┐
                          │ RECEPTIONIST│
                          └──────┬──────┘
                                 │
                          ┌──────┴──────┐
                          │   INTERN    │
                          └─────────────┘
```
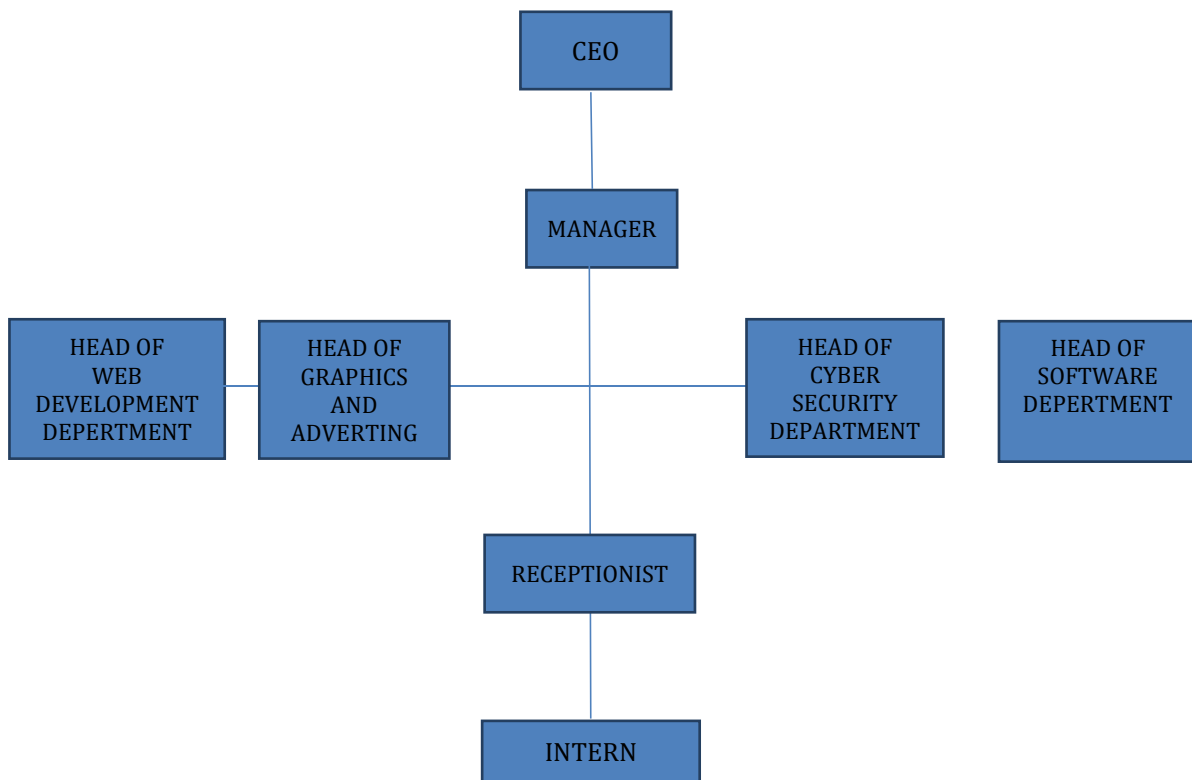
CHAPTER THREE

3.0 INDUSTRIAL EXPERIENCE

During my four-month SIWES training at [G-Technology], I worked as a web development and cyber-security intern. My responsibilities are to assisting in the design and development of web application using [HTML, CSS and JAVASCRIPT] in web development and the operating system (Linux), Basic command, file permission, and Virtualization machine that we use to partnering with our main operating system (window).

3.1 INTRODUCTION TO WEB DEVELOPMENT

Web Development is the process of building and maintaining website, web application and mobile application using various programming language, framework and tools. It is divided into (3) which involve:

1.   Frontend development
2.   Backend development
3.   Full – stack development

✓ **Frontend development:** creating user interface user experience and client side logic using html, CSS, and JavaScript. And framework likes react angular, or value

✓ **Back-end development:** building server-side logic database integration and API connectivity using language like java, python, ruby, and framework like node.js.

✓ **Full- stack development:** combining frontend and backend development to create a complete web application

Web development involves various aspects such as:

1.   Web design
2.   User experience (ux) design
3.   Web security
4.   Mobile application development
5.   Web assembly

Web developer uses various tools such as:

1.   Text editor (e.g. visual studio code)
2.   Integrated development environment (IDE)
3.   Database management system
4.   Testing framework

STATUS CODE

A status code is a three- digit number that a web server user to communicate the outcome of a request to a client (usually a web browser). It's a way for the server to say "here's what happened with your request ".

Five categories of status codes

1.   **1xx :** informational (request received, processing)

2.   **2xx:** success (request successful data returned)

3.   **3xx:** redirection (resource moved, redirect to another URL)

4.   **4xx:** client error (request invalid, access denied)

5.   **5xx:** server error (server-side error, request failed)

Some common status code includes:

*   **200:** ok (request successful)

*   **404:** not found (resource not found)

*   **500:** internal server error (server-side error)

*   **301:** moves permanently (resource moved, new URL)

*   **400:** bad request (invalid request error in syntax)

3.2   FRONTEND DEVELOPMENT

HTML (HYPERTEXT MARKUP LANGUAGE)

Html (Hypertext Markup Language): is the standard markup language used to create web page it's the backbone of a website, providing the structure and content that the web browser render the user. It defines the structure and layout of a web page, html consists of series of element, represented tags (< >), these elements include:

1.   Heading (h1 – h6)

2.   Image (img)

3.   Span (span)

4.   Title (title)

5.   Paragraph (p)

6.   Form (form input, select) etc

HTML DOM [DOCUMENT OBJECT MODEL]

3.3   CSS (CASCADING STYLE SHEETS)

CSS (Cascading Style Sheets): is a styling language used to control the layout and visual appearance of web page written in html. It also, used in target specific html element to apply style and consist of properties and value example (color: red; padding: 10px;,) etc.

HOW TO CONNECT YOUR CSS TO YOUR HTML

We can connect CSS to html in 3 ways:

1.   Inline CSS

2.   Internal CSS
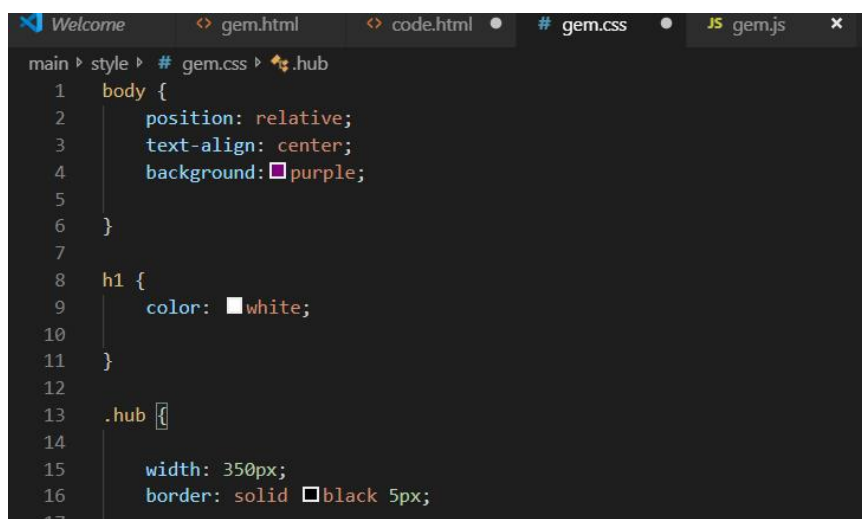
3.   External CSS

1. Inline CSS: written directly within an html element using the style attribute.

Example; <p style = "color: blue ;"> this is inline CSS </p>

2. Internal CSS: written within an html file using the <style> tag in the <head> section.

Example: <style> /* style here */ </style>

3. External CSS: is separate from html while internal and inline CSS are embedded within html. It generally preferred for maintainability and scalability.

CSS SELECTOR

CSS (Cascading Style Sheets) selectors are patterns used to select elements in an HTML document that you want to style.

TYPE OF CSS SELECTOR

1.   Type Selector: Targets all elements with a specific tag name (e.g., h1, p).

2.   Class Selector: Targets all elements with a specific class (e.g., .nav, .header).

3.   ID Selector: Targets a single element with a specific ID (e.g., #header, #footer).
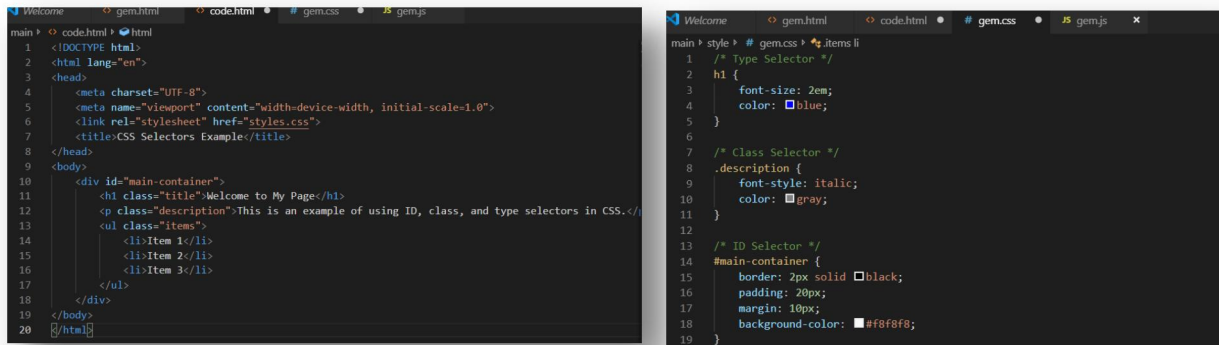
3.4   JAVASCRIPT

JavaScript: is a high-level dynamic and interpreted programming language used for client-side scripting on the web. It allows developer to create interactive web page, web application and mobile application. It is used for:

1.   Client – side scripting

2.   Server – side programming

3.   Game development

4.   Mobile development

WAY TO DECLARE A VARIABLE IN JAVASCRIPT

1.   Var.
2.   Let.
3.   Const.

1. Var.: is a global object that can be access anywhere. It didn't have a scope.

Example: var name: "mercy";

Console.log (name)

2.   Let : is a block scope ,

Anything inside curly bracket define scope

Example: {

Let name = "john";

}

Console.log (name);

CONCATENATION

Concatenation: mean joining two things together to become one. Using template string to concatenate:

1.   Back tick
2.   Dollar sign
3.   Curly bracket

Example:

Let first name = "jerry"

Let second name = "Akanke"

let full name = $`{first name}${second name}`

Console.log (full name);

•   **Console:** this is a feature in almost all browser it work with java Script to output a value to the console.

•   **Alert [ ] method:** is a built in function that display a message box with a specified message and an ok button. it is often used to alert or notify the user of something important

DATA TYPE IN JAVASCRIPT PROGRAMMING

Data type is a classification of data based on its format, size, and set of values, it can hold. JavaScript

Is a dynamically-typed language, which means that you don't need to declare the data type of a variable before using it?

TYPE DATA TYPE IN JAVASCRIPT

1.   **Number:** represents a numeric value e.g., 42,3.14
2.   **String:** represent a sequence of character e.g., "hello"
3.   **Boolean:** represent a true or false value.
4.   **Null:** represent the absence of any value.
5.   **Variable:** Are containers that hold a value think of it like a labeled box where you can store value. E.g.,

Age =20,

console.log (age)

6.   **Array:** represent a collection of value, e.g., [1,2,3,4,5,6]

Method of array

1.   **Push:** it is use to push information into array

Example:

Let user = [

"Samuel",

"Jade",

]

ser. Push ("favor")

Console.log (user)

2.   Pop : is used for delete item from the back

Example:

Let user = [ ]

User. push ("Tom")

User. push ("Sunday")

User.pop ("favor")

Console.log (user)

3.   Shift () method: is used to remove the first element from an array and return that element. It change the length of the array and shift all the element down by one position

/*

Let color = ["red", "green", "blue", "yellow",]

Console.log (color. shift ()); // output: "red",

Output ["green", "blue", "yellow",]   */

4.   Un-shift () method: is a built-in JavaScript array method that adds one or more elements to the beginning of an array. It modifies the original array and returns the new length of the array after the elements have been added.

Example:

Let fruits = ['banana', 'orange', 'apple'];

fruits. Un-shift ('mango'); // Adding a single element

console.log (fruits); // Output: ['mango', 'banana', 'orange', 'apple'].

7.   **Object:** is a complex data type that allows you to store collections of data and more complex entities. Objects can hold multiple values in the form of key-value pairs, where each key (also known as a property) is a string (or a Symbol) that map to a value, which can be of any data type, including another object or even a function.

Example;

```
Const person = {
Name: "Alice",
Age: 25,
Is Student: false
};
console.log (person)
```

8. **Function:** is a reusable block of code that performs a specific task. Functions allow you to encapsulate logic, making it easier to write, manage, and reuse code. Functions can take inputs (called parameters), perform operations, and return outputs (values).

2 Common Way To Declare A Function

1. **Function Declaration:** is a way to define a named function using the function keyword. This is one of the most common ways to create functions in JavaScript. Function declarations are hoisted, meaning they can be called before they are defined in the code, making them flexible for organizing code.

Example:

```
Function functionName (parameter1, parameter2) {
// Code to be executed
Return result; // Optional
}
```

Example:

```
Functions add (a, b) {
Return a + b;
}
console.log (add (3, 5)); // Output: 8
```

2. **An arrow function:** is a feature in JavaScript introduced with ECMAScript 6 (ES6) that provides a more concise syntax for writing function expressions.

```
Const function Name = (parameter1, parameter2) => {
// Code to be executed
Return result; // Optional
};
```

Example:

Const subtract = (a, b) => a - b; // Single expression, implicit return

console.log (subtract (10, 4)); // Output: 6

CONDITIONALS/DECISION MAKING IN JAVASCRIPT

Condition or decision-making in JavaScript allows you to execute different blocks of code based on certain conditions. This is a fundamental aspect of programming that enables dynamic and flexible application behavior. JavaScript provides several statements for decision-making, including if, else, switch, and the ternary operator.

1. if Statement

If statement: evaluates a condition and executes a block of code if the condition is true.

Example:

Let temperature = 30;

If (temperature > 25) {

console.log ("It's a hot day.");

}

2. if...else Statement

You can provide an alternative block of code to execute if the condition is false using else.

Example:

Let temperature = 20;

If (temperature > 25) {

console.log ("It's a hot day.");

} else {

console.log ("It's not a hot day.");

}

3. else if Statement

For multiple conditions, you can use else if to test additional conditions.

Example:

Let temperature = 10;

If (temperature > 25) {

console.log ("It's a hot day.");

} else if (temperature < 15) {

console.log ("It's a cold day.");

} else {

console.log ("It's a mild day.");

3.5   INTRODUCTION TO CYBER-SECURITY

Cyber-security refers to the practices, technologies, and processes designed to protect computer systems, networks, and sensitive information from unauthorized access, use, disclosure, disruption, modification, or destruction. This includes protection against malware, viruses, phishing, and other types of cyber threats.

Cyber-security is a critical component of modern life, as more and more aspects of our lives are dependent on technology and the internet. With the increasing reliance on digital technologies, the risk of cyber attacks and data breaches has also grown.

Some common types of cyber-security threats include:

1. Malware: Software designed to harm or exploit a computer system.
2. Viruses: Types of malware that replicate and spread to other computers.
3. Phishing: Attempts to trick individuals into revealing sensitive information, such as login credentials or credit card numbers.
4. Ransom ware: Malware that demands payment in exchange for restoring access to encrypted data.
5. Denial of Service (DoS) attacks: Attempts to overwhelm a system with traffic in order to render it unavailable.
6. SQL Injection: A type of attack that targets databases by injecting malicious code.
7. Cross-Site Scripting (XSS): A type of attack that involves injecting malicious code into websites.

To protect against these threats, cyber-security professionals use a variety of techniques, including:

1. Firewalls: Network security systems that control incoming and outgoing network traffic.
2. Antivirus software: Programs that detect and remove malware.
3. Encryption: The practice of converting plaintext data into unreadable cipher text.
4. Secure authentication: Mechanisms that verify the identity of users and devices.
5. Regular backups: Creating copies of important data to ensure its recovery in case of a disaster.
6. Patches and updates: Applying security fixes to software and operating systems.
7. User education: Educating individuals on how to identify and prevent cyber-security threats.

Some of the key areas of cyber-security include:

1. Network security: Protecting computer networks from unauthorized access.
2. Application security: Securing software applications from vulnerabilities.
3. Data security: Protecting sensitive information from unauthorized access.
4. Cloud security: Securing cloud-based data and applications.
5. Endpoint security: Protecting endpoints, such as laptops and mobile devices.
6. Internet of Things (IOT) security: Securing connected devices from vulnerabilities.
7. Penetration Testing: This involves simulating cyber attacks to test an organization's security defenses.

3.6 OPERATING SYSTEM (OS)

An operating system (OS) is a software program or set of programs that manage computer hardware and provide services for computer programs. It serves as an intermediary between the user and the computer hardware, enabling the user to perform tasks without needing to understand the complexities of the underlying hardware components.

Types of Operating Systems

1. Single-user OS: Designed for one user at a time (e.g., Windows, macOS).
2. Multi-user OS: Allows multiple users to access the system concurrently (e.g., UNIX, Linux).
3. Real-time OS: Designed to process data as it comes in, with a strict time constraint (e.g., embedded systems in medical devices, automotive controls).
4. Distributed OS: Manages a group of independent computers and makes them appear to the users as a single coherent system (e.g., Google's Chrome OS).

Examples of Operating Systems

1. Windows: Developed by Microsoft, widely used on personal computers and laptops.
2. Mac OS: Developed by Apple, used on Macintosh computers.
3. Linux: An open-source operating system with various distributions (like Ubuntu, Cent OS, and Fedora) used on servers, desktops, and embedded systems.
4. UNIX: A powerful, multiuser operating system often used in servers and workstations.
5. Android: A mobile operating system based on a modified version of the Linux kernel, designed primarily for touch screen devices like smart phones and tablets.

KALI LINUX

Kali Linux is a Debi an-based Linux distribution specifically designed for penetration testing, security research, and digital forensics. It is widely used by cyber-security professionals and ethical hackers for its extensive collection of security tools and applications.

Kali Linux is a powerful and versatile operating system tailored for cyber-security tasks. Its rich feature set and extensive toolset make it a preferred choice for professionals in the field of information security. However, it is important to use Kali Linux responsibly and ethically, as its tools can also be used maliciously. Always ensure that you have permission before testing or scanning any systems or networks.

KALI LINUX INTERFACE
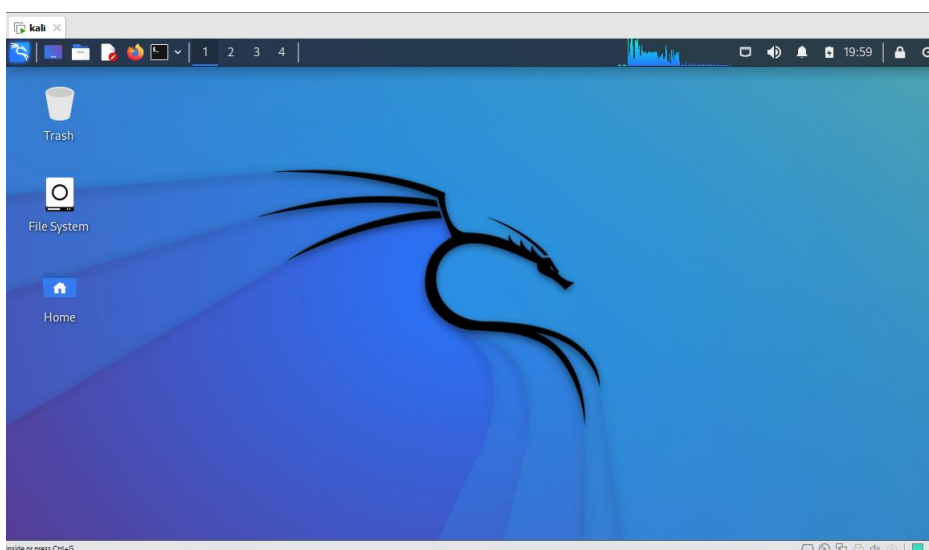
Use Cases for Kali Linux

1. Ethical Hacking: Professionals use Kali to perform penetration testing on systems to identify vulnerabilities before malicious hackers can exploit them.

2. Security Research: Researchers use Kali to develop and test new security tools and techniques.

3. Training and Education: Kali is often used in cyber-security training programs to teach students about penetration testing and ethical hacking.

4. Digital Forensics: Kali includes tools that can help in recovering data and analyzing systems involved in security incidents

2 WAY OF INTERACTING WITH KALI

1.   We can interact with GUI (graphic user interface)

2.   Command prompt

WE CAN RUN KALI LINUX IN 3 WAYS:

1.   It can be run has the Main operating system

2.   It can run through pen drive (Rufus software we allow this)

3.   Through using virtual machine (VM-ware).

3.7 VIRTUALIZATION MACHINE

A virtualization machine, often referred to as a virtual machine (VM), is a software-based emulation of a physical computer. It allows users to run multiple operating systems (OS) on a single physical machine by creating isolated environments that function like separate computers. Virtual machines utilize virtualization technology, which enables the abstraction of hardware resources and management of multiple operating systems on a single host machine.

Key Concepts of Virtual Machines

1. **Hypervisor**: This is the software layer that creates and manages VMs. It sits between the hardware and the operating systems. There are two types of hypervisors:

•   **Type 1 Hypervisor (Bare-metal):** This runs directly on the hardware and does not require a host operating system. Examples include VMware ESXi, Microsoft and Hyper-V.

•   **Type 2 Hypervisor (Hosted):** This runs on top of an existing operating system. Examples include VMware Workstation, Oracle Virtual Box, and Parallels Desktop.

2. **Guest Operating System:** The OS installed and running on a VM. A single physical machine can run multiple guests OSs at the same time, allowing for diverse environments.

4.   **Virtual Hardware:** Each VM has virtualized versions of hardware components (CPU, memory, storage, network interfaces) that are allocated by the hypervisor from the physical resources of the host machine.

VM WARE WORKSTATION AND VIRTUAL BOX

VMware Workstation: A Type 2 hypervisor for desktop virtualization that allows users to run multiple operating systems on their desktops or laptops. It's commonly used by developers and IT professionals for testing and development purposes.

Virtual Box: is an open-source Type 2 hypervisor that allows users to create and manage virtual machines on their desktop or laptop computers. It supports a wide range of host and guest operating systems and is known for its user-friendly interface.

DIFFERENT BETWEEN VM WARE AND VIRTUAL BOX

1.   VMware Workstation and Fusion are commercial products with licensing fees, whereas Virtual Box is free and open-source.

2.   VMware products are often considered to have a more polished interface, while Virtual Box is user-friendly but may have fewer advanced features.

        Both VMware and Virtual Box provide good performance, but VMware has historically offered better performance and features for enterprise use cases due to its Type 1 and advanced offerings.

BASIC COMMAND IN KALI LINUX

Command refers to instructions or instructions entered in the terminal (command-line interface) to perform specific actions, like managing files, running scripts, installing software, or configuring system settings. Kali Linux, being a security-focused distribution, includes many specialized commands for penetration testing, vulnerability scanning, network analysis, and more.

Common types of commands you'll find in Kali Linux

Basic Commands

1.   **`ls`** – List Directory Contents

It displays files and directories in the current directory. Adding options like `-l` or `-a` can show more details.

ls   # Basic list

ls -l   # Detailed listing

ls -a   # Show hidden files

2.   **`cd`** – Change Directory

This command allows you to navigate between directories.

```bash
cd /path/to/directory
Cd   # Go back one directory
```

3.   **`mk-dir`** – Make Directory

This command is use to create a new directory.

```bash
Mk-dir new directory
```

4. **`rm-dir`** – Remove Directory

This command is use to deletes an empty directory.

```bash
rm-dir directory name
```

5. **`rm`** – Remove Files or Directories

To remove files or directories, be careful with this one as it deletes permanently (no trash bin).

```bash
rm filename      # Remove a file
rm -r directory name  # Remove a directory and its contents
```

6. **`cp`** – Copy Files or Directories

This command is use to copies files or directories.

```bash
Cp source destination
Cp -r source dir destination dir # Copy directories
```

7. **`mv`** – Move or Rename Files or Directories

Used to move or rename files/directories.

```bash
mv old name new name
mv filename /path/to/new/directory
```

8. **`man`** – Manual Pages

Displays the manual pages for a command (useful for understanding how to use a command)

```bash
Man command name.
```

9. **`cat`**: Displays the contents of a file.
10. **`cat > `**: allow you to write something inside your file.
11. **`man`**: Displays manual for command.
12. **`touch`**: his use to create a file.
13. **'nano' or 'vim'**: Text editors to create or modify files.

KALI LINUX

TERMINAL COMMAND

System Commands

- **`sudo`**: Runs commands with super user (administrator) privileges.

- **`apt update`**: Updates the list of available packages from the repositories.

- **`apt upgrade`**: Upgrades installed packages to the latest version.

- **`apt install <package>`**: Installs a software package.

- **`ifconfig` or `ip a`**: Displays network interface information.

- **`ps`**: Displays currently running processes.

- **`top`**: Displays real-time system resource usage.

- **`shutdown`**: Shuts down or reboots the system.

FILE PERMISSION IM KALI LINUX

File permissions in Kali Linux, like in other UNIX and Linux-based operating systems, govern who can read, write, or execute a file or directory. Understanding file permissions is crucial for ensuring the security and proper functioning of a system. In Kali Linux, file permissions are set using three categories: the owner (user), the group, and others.

Each file and directory has three types of permissions:

1. Read (r): Permission to read the file or list the contents of a directory.
2. Write (w): Permission to modify the file or add, delete, or modify files within directory.
3. Execute (x): Permission to execute a file (if it's a script or binary) or traverse (enter) directory.

Permission Representation

Permissions can be represented in two ways:

1. Symbolically: Using letters (r, w, x)
2. Numerically: Using octal numbers

1. The first character (-) indicates whether it's a regular file (-) or a directory (d).
2. The next three characters (rwx) are permissions for the owner of the file.
3. The next three characters (r-x) are permissions for the group associated with the file.
4. The last three characters (r--) are permissions for others (everyone else).

Numeric Permissions

Permissions can also be set numerically using a three-digit octal number:

Read = 4

Write = 2

Execute = 1

For example:

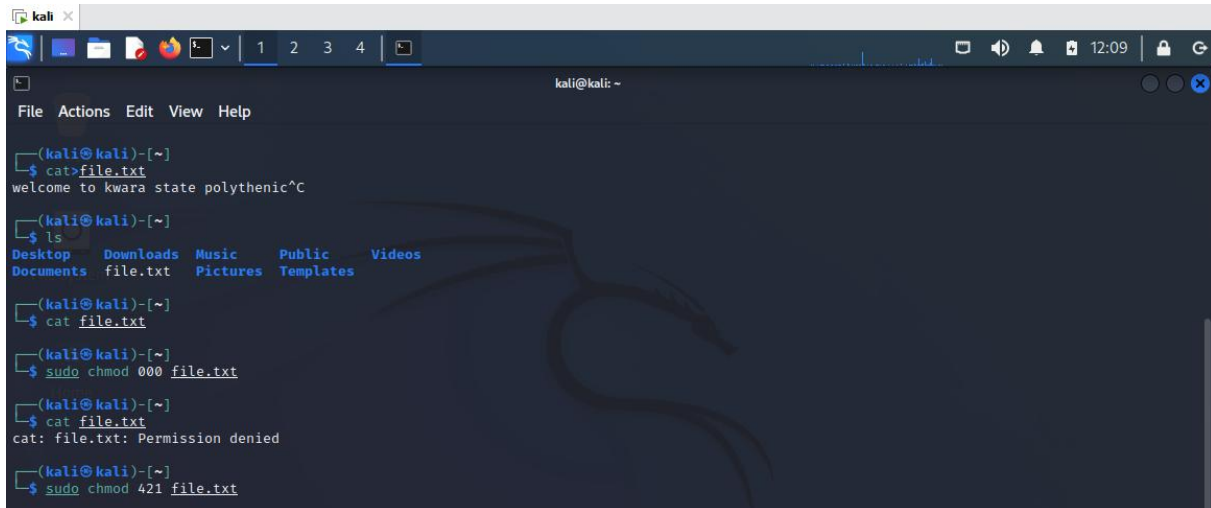rwx is represented as 4 + 2 + 1 = 7

r-x is represented as 4 + 0 + 1 = 5

r-- is represented as 4 + 0 + 0 = 4

Thus, the permissions rwxr-xr-- can be represented as 754.

You can manage file permissions using the CHMOD command. Here are a few examples:

1.   To give execution permission to the owner: CHMOD u+x filename

2.   To remove write permission from a group: CHMOD g-w filename

3.   To set permissions numerically: CHMOD 755 filename

HOW TO LOCK YOUR TEXT FILE USING FILE PERMISSION "SUDO CHMOD"

CHAPTER FOUR

4.0   CHALLENGES

Understanding and effectively using version control systems like Git is difficult for me, leading to potential issues with code management and collaboration. Developing a habit of writing test cases and ensuring code quality might be overlooked, leading to bugs in production. Troubleshooting bugs and errors can be frustrating, especially if you lack prior experience with similar problems. Long hours of coding and problem-solving can lead to burnout if not managed with proper breaks and self-care. Despite these challenges I gained a lot of experience from the program

While these challenges can be daunting, they also present valuable learning opportunities that can enhance my skills and resilience as a web developer . Embracing these challenges, seeking help when needed, and maintaining a growth mindset help me navigate my SIWES program more effectively.

4.1   RECOMMENDATIONS

I urge the Federal Government to take the SIWES program more seriously especially for I.T students and Applied Sciences because the world is drifting towards I.T. The Government should ensure a proper supervision of SIWES students so that the purpose of the program will be archived. The Federal Government should also make adequate provision in the annual budget for proper funding of SIWES in view of the potential of the program to contribute to enhancing the quality of the pool I.T skills available to the country. A comprehensive and detailed directory of who accepts students for SIWES is urgently required to facilitate placement for SIWES student in industry. The ITF also should ensure that the payment of the student allowance is paid on time in other to keep the SIWES students zealous and motivated.

I recommend that the Nigerian educational system should be reviewed as what is taught in the classroom totally contradicts with what is needed the in the actual work place.

CHAPTER FIVE

5.0   CONCLUSION

My SIWES was a very successful one, I had an insight of the information technology world. I have now known the basic knowledge of web development and Cyber-Security. With this, I will be able set a goal for myself to build a complete website and write codes for different programs.

SIWES as a course has truly exposed me to the challenges faced in a growing I.T world that is dependent on computers.