

Web Application Programming and Hacking – Lab 1 Report

Instructor: Dr. Phu Phung

Student Name: **Revanth Kumar Mallem**

Email: mallemr1@udayton.edu

Short-bio: Motivated and proactive student with a strong foundation in electronics and communication engineering, currently pursuing a Master's in Computer Science. Passionate about web development, cloud services, and AI.

LinkedIn: <https://www.linkedin.com/in/revanthmallem>

Repository URL: <https://github.com/Revanth-0606>

## 1. Overview

This project involved creating a comprehensive, professional portfolio website from the ground up and deploying it to the cloud using GitHub Pages. The core of the assignment was to build a dynamic and interactive front-end experience by leveraging modern web technologies. This included structuring the site with HTML5, styling it with a responsive CSS framework (Bootstrap), and bringing it to life with advanced JavaScript functionalities.

**Key learning outcomes:**

- **Front-End Frameworks:** Practical experience using Bootstrap to create a professional, mobile-first, and responsive layout.
- **Advanced JavaScript:** Implemented several dynamic features, including real-time clocks and interactive UI elements, reinforcing skills in DOM manipulation.
- **Asynchronous Operations:** Mastered async/await with the fetch API to integrate third-party web services, handling asynchronous data fetching, parsing JSON, and updating webpage content.
- **Client-Side State Management:** Used JavaScript cookies to store data on the client-side, allowing the website to “remember” users and personalize their experience across sessions.
- **Cloud Deployment:** Hands-on experience with the development-to-deployment workflow using Git and GitHub Pages, making a web application publicly accessible.

- **Live Deployed Website:** <https://Revanth-0606.github.io>
- **Project GitHub Repository:** <https://github.com/Revanth-0606/Revanth-Final-Website-Enhanced>

## 2. Project Tasks and Implementation

### Task 1: Professional Website and Non-Technical Requirements

#### 1.1 Use of an Open-Source CSS Framework

Bootstrap 5 was used to ensure the website is fully responsive and visually appealing.

#### 1.2 Page Tracker

A Flag Counter is embedded in the footer to track the number and origin of visitors.

```
<a href="https://info.flagcounter.com/YgqF">

```

## Task 2: WAPH Course Page

A separate `course.html` page introduces the “Web Application Programming and Hacking” course, with instructor, institution, and a link to the project repository.

## Task 3: Technical Requirements - JavaScript Functionality

### 3.1 Use of JavaScript Libraries

- jQuery: For simplified DOM manipulation and event handling.
- Native JS only used for clarity.

### 3.2 Live Digital Clock

```
const clock = document.getElementById("digital-clock");
setInterval(() => {
  clock.textContent = new Date().toLocaleTimeString();
}, 1000);
```

### 3.3 Show/Hide Email Address

A toggle feature is included for privacy, implemented using native JavaScript.

## Task 4: Technical Requirements - Web API Integration

### 4.1 JokeAPI Integration

```
async function fetchJoke() {
  const res = await fetch('https://v2.jokeapi.dev/joke/Programming,Pun?type=single');
  const data = await res.json();
  document.getElementById("joke-text").textContent = data.joke;
}
fetchJoke();
setInterval(fetchJoke, 60000);
```

### 4.2 XKCD API Integration

```
async function fetchComic() {
  const comicBox = document.getElementById("comic-container");
  const randomId = Math.floor(Math.random() * 2500) + 1;
  const url = `https://api.allorigins.win/get?url=${encodeURIComponent(`https://xkcd.com/${randomId}`)}`;
  const res = await fetch(url);
  const json = await res.json();
  const data = JSON.parse(json.contents);
  comicBox.innerHTML = `<strong>${data.title}</strong><br>`;
}
```

## Task 5: Technical Requirements - JavaScript Cookies

### 5.1 Remembering the Client

```
const lastVisit = document.cookie.split('; ').find(row => row.startsWith('lastVisit='));
let message = "Welcome to my portfolio!";
if (lastVisit) {
  const lastDate = new Date(decodeURIComponent(lastVisit.split('=')[1]));
  message = `Welcome back! Last visit: ${lastDate.toLocaleString()}`;
}
document.getElementById("welcome-banner").textContent = message;
document.cookie = `lastVisit=${encodeURIComponent(new Date().toISOString())}; path=/; max-age`;
```

### Features Implemented

- Responsive Design with Bootstrap 5
- Sections: About, Education, Projects, Internships, Contact, WAPH Course
- Digital Clock and Cookie-based Greeting
- Joke of the Minute using JokeAPI
- Random XKCD Comic
- Flag Counter Visitor Tracker

### Technologies Used

- HTML5
- CSS3
- JavaScript (ES6+)
- Bootstrap 5
- JokeAPI
- XKCD API

### Setup and Usage

Open `index.html` in a web browser or host on GitHub Pages to see the deployed site.

**Live Site:** <https://Revanth-0606.github.io>

### Credits

Website theme designed with Bootstrap. Additional APIs used from JokeAPI and XKCD.