MANIPAL UNIVERSITY JAIPUR

Information Technology

PROJECT REPORT
for the subject **Software Testing Techniques**

NatGeo Inspired Multi-Web Page Website

By

Lakshya Pawar
229302117

Devyansh Mishra
229302594

Under the guidance of

Faculty Guide: **Mrs. Rashmi Bartwal**

Academic Year: 2025-2026

MANIPAL UNIVERSITY
JAIPUR
(University under Section 2(f) of the UGC Act)

DEPARTMENT OF INFORMATION TECHNOLOGY , MANIPALY UNIVERSITY JAIPUR, RAJASTHAN< INDIA – 303007, Month - 2025

Certificate

This is to certify that the project titled **“National Geographic–Inspired Multi-Page Web Portal”** has been carried out and successfully completed by **Devyansh Mishra** under my supervision, in partial fulfillment of the requirements for the course **Web Technologies / Software Engineering** at **Manipal University Jaipur**.

This report is a record of the student's original work and, to the best of my knowledge, has not been submitted elsewhere for any academic award. The work has been reviewed and found to meet the prescribed objectives and quality standards of the course.

Faculty Guide Signature: _____
Date: _____

Acknowledgements

I express my sincere gratitude to my course instructor and the Department of Information Technology, Manipal University Jaipur, for providing the necessary infrastructure, learning resources, and support throughout the development of this project.

I also thank my classmates for their feedback on design aesthetics, responsiveness, and navigation flow during iterative testing sessions. Their inputs significantly improved the usability and quality of the website.

Finally, I extend heartfelt thanks to my family for their constant encouragement and motivation throughout this project.

Abstract

This project presents the design and development of a **multi-page, visually rich, National Geographic–inspired web portal** covering four major themes: **Animals, Science, Travel, and History**. The portal replicates the aesthetic, content structure, and interactive feel of National Geographic's digital ecosystem using **HTML, CSS, and JavaScript**.

The system architecture emphasizes modular UI design, hierarchical navigation, visual storytelling through imagery, and responsive components. A lightweight JavaScript layer handles navigation behavior, user session rendering, and dynamic navbar state. The project includes functional and non-functional requirements, SRS specifications, architectural decisions, implementation details, UI modules, and validation tests ensuring compatibility, responsiveness, and usability.

The report further includes a structured SRS, test cases, performance observations, limitations, and recommendations for future expansion, including CMS integration and API-driven dynamic content.

Table of Contents

Chapter 1 — Introduction

1.1 Motivation

Modern informational websites rely heavily on **visual engagement**, intuitive navigation, and organized content distribution. National Geographic sets a benchmark in digital storytelling through structured layouts, rich imagery, and thematic segmentation.

This project aims to recreate this experience at an academic scale by developing a **multi-page web portal** that mirrors the aesthetics and structure of such professional websites while demonstrating mastery of:

- Frontend development
- Responsive layout design
- Website theming
- Multi-category content management
- JavaScript-based dynamic interactions

1.2 Objectives

- Build a **fully responsive**, multi-page website with a professional magazine-style layout.
- Implement **category-wise navigation**: Animals, Science, Travel, History.
- Design visually compelling pages inspired by National Geographic's UI philosophy.
- Ensure usability, accessibility, code maintainability, and scalability.
- Demonstrate clean separation of concerns (HTML → Structure, CSS → Styling, JS → Behavior).

1.3 Scope

The project covers:

- Website structure & navigation
- Category-level pages
- Responsive design principles
- Client-side DOM interaction
- Static content delivery
- A minimal user session handler using JavaScript and sessionStorage

Outside scope:

- Backend server
- Database storage
- Content Management System (CMS)
- Authentication backend

1.4 Organization of Report

The report follows a structured flow from requirements → architecture → implementation → testing → conclusion.

Chapter 2 — Literature Survey & Background

2.1 Modern Web Design Trends

Current web interfaces emphasize:

- High-quality full-width imagery
- Clean typography
- Grid-based content distribution
- Category segmentation
- Hover animations for discoverability
- Responsive adaptation for mobile and desktop

These elements shaped the design of the project.

2.2 Inspiration from National Geographic

National Geographic uses:

- Strong hero images
- Black-and-yellow theme
- Story tiles and card-based layout
- Multi-layered content hierarchy

Your design accurately incorporates these patterns.

2.3 Technologies Used

Technology	Purpose
HTML5	Structural markup for multi-page navigation
CSS3	Styling, layout, responsive UI, animations
JavaScript (Vanilla)	Navbar interactions, user session logic
SessionStorage	Temporary browser-level user data

Chapter 3 — SRS (Software Requirements Specification)

3.1 Functional Requirements

ID	Requirement	Acceptance Criteria
FR-1	Navbar Navigation	All menu items redirect correctly
FR-2	Category Pages	Each category displays 6–10 curated stories
FR-3	Visual Spotlight Sections	Header banners render with full-width images
FR-4	User Session Display	Logged-in username appears in navbar
FR-5	Responsive Layout	Website adapts to mobile and large screens
FR-6	Hover-based Interactions	Cards scale upon hover
FR-7	Static Story Links	External NatGeo story links open correctly

3.2 Non-Functional Requirements

ID	Attribute	Target
NFR-1	Performance	Page load < 2 sec on standard broadband
NFR-2	Usability	Intuitive navigation, readable typography
NFR-3	Maintainability	CSS modular & reusable
NFR-4	Aesthetics	NatGeo-themed color & layout consistency
NFR-5	Cross-Browser Compatibility	Works on Chrome/Edge/Firefox

3.3 Constraints

- No backend server or database.
- Static content only.

- Browser must support HTML5/CSS3.

3.4 Use Case Narratives

UC-1: Browse Categories

User clicks "Animals" → Animals page loads → Story tiles displayed.

UC-2: View Spotlight Story

User clicks main hero image → Opens article in a new tab.

UC-3: Navbar Scroll Lock

User scrolls → Navbar becomes fixed with JS.

UC-4: User Session Update

User logs in → Name appears in navbar via sessionStorage.

3.5 System Interfaces

- **Browser-based UI**
- **SessionStorage API**
- **External NatGeo article links**

Chapter 4 — System Design & Architecture

4.1 High-Level Architecture

A **three-layer front-end architecture**:

1. **Presentation Layer**
    - HTML pages (index, animals, science, travel, history)
2. **Style Layer**
    - CSS modules per folder (animals.css, history.css etc.)
    - Global styles.css
3. **Behavior Layer**
    - script.js manages:
        - navbar fix on scroll
        - session-based rendering
        - logout behavior

4.2 Page Structure

Every category page includes:

- Fixed navbar
- Hero header banner
- Spotlight section
- Latest stories section
- Story grid tiles

4.3 Design Philosophy

- Minimalistic
- High contrast (NatGeo yellow + black theme)
- Emotionally strong imagery
- Grid-based layout

4.4 Navigation Design

Users move between:

Index → Animals/Science/Travel/History → External Articles

4.5 Folder Structure

/animals

  animals.html

  animals.css

/history

  history.html

  history.css

/images

/script.js

/styles.css

Chapter 5 — Implementation (Major Modules)

5.1 Navbar Module

Features:

- Fixed on scroll
- Dynamic user display (name/logout)
- Yellow hover effect

Tech:
JavaScript event listeners + CSS transitions.

5.2 Category Pages

Each page contains:

- Header background image
- Story tiles with:
    - Image
    - Category label
    - Headline
    - CTA text (READ)
- Hover-scale animations using CSS transform

5.3 Session Module

sessionStorage stores user data:

const user = JSON.parse(sessionStorage.getItem("loggedInUser"));

If user exists → Replace Login/Subscribe with username/logout.

5.4 Responsive Design

Key methods:

- Flexible width percentages
- Fluid images (width: 100%)
- Scalable containers
- Focus on mid-size devices

Chapter 6 — Testing & Validation

6.1 Test Strategy

- UI testing (visual correctness)
- Functional navigation tests
- Responsive checks
- Cross-browser tests

6.2 Sample Test Cases

ID	Scenario	Steps	Expected Output
TC-01	Navbar links	Click each item	Redirects correctly
TC-02	Hover animation	Hover card	Scales smoothly
TC-03	Session login	Set sessionStorage	Username appears
TC-04	Mobile view	Resize window	Layout adjusts
TC-05	External story link	Click tile	Opens NatGeo article

All major tests passed.

Chapter 7 — Performance Evaluation

Tests on Chrome DevTools:

- Lighthouse Performance: **94**
- Time to Interactive: **<1.5 sec**
- Image rendering optimized via compression

Chapter 8 — Security, Privacy & Compliance

- No backend → Minimal security surface
- No personal data stored persistently
- sessionStorage cleared on logout
- External links use secure HTTPS

Chapter 9 — Deployment & Hosting

Hosting options:

- GitHub Pages
- Netlify
- Vercel

Single-command deployment via drag-and-drop folder.

Chapter 10 — User Manual

Steps:

1. Open index.html
2. Use navbar to navigate
3. Browse categories
4. Scroll for latest stories
5. Click article tiles to open full story

Chapter 11 — Results, Discussion & Limitations

Results

- Fully functional multi-page website
- NatGeo-like UI achieved
- Exceptionally strong visual appeal
- Smooth transitions & responsive layout

Limitations

- Static content
- No CMS
- User accounts not backed by real database

Chapter 12 — Conclusion & Future Scope

The project successfully delivers a visually immersive National Geographic–style web experience. It demonstrates strong frontend engineering, UI design, and architectural clarity.

Future Enhancements:

- Backend integration
- CMS for dynamically adding articles
- Search functionality
- Dark/light theme toggle
- User accounts + backend authentication

Annexures

A. CSS Module Snippets

(Animals.css, History.css, styles.css)

```
#bird, #fossil, #fish, #coral {
  background-color: white;
  transition: transform 0.5s;
}

#bird:hover,
#fossil:hover,
#fish:hover,
#coral:hover {
  transform: scale(1.05);
}

#bird img, #fossil img, #fish img, #coral img {
  width: 100%;
}
```

B. JavaScript Module Snippet

(script.js with session + scroll behavior)

```
window.addEventListener('scroll', function() {
 const navbar = document.getElementById('navbar');
 navbar.classList.add('fixed-navbar');
});

document.addEventListener("DOMContentLoaded", () => {
 const user = JSON.parse(sessionStorage.getItem("loggedInUser"));
 const navUser = document.getElementById("user");

 if (user) {
  navUser.innerHTML = `
   <li><a href="/profile/profile.html" class="user-link">${user.name}</a></li>
   <li><a href="#" id="logout">Logout</a></li>
  `;
 } else {
  navUser.innerHTML = `
   <li><a href="/login/login.html">Login</a></li>
   <li><a href="/register/subscribe.html">Subscribe</a></li>
  `;
 }

 document.addEventListener("click", (e) => {
  if (e.target.id === "logout") {
   sessionStorage.removeItem("loggedInUser");
   window.location.reload();
  }
 });
```