# From Analysis to Deployment: Building & Publishing Data-Driven Software

## Course Overview

This semester-long course is designed for students from diverse backgrounds—including those with design or visualization experience and those with minimal coding skills. We focus on taking your data science outputs—whether they be analyses, visualizations, or other insights—and transforming them into interactive, online software products. Through a mix of theory and hands-on labs, you will learn basic web development (HTML, CSS, JavaScript), dynamic data integration via APIs, version control with Git, and a variety of deployment options (GitHub Pages, Vercel, Netlify, AWS). The course also explores the conceptual and practical differences between self-publishing your own projects and using pre-built platforms, emphasizing informed data stewardship and ethical sharing practices. Open Educational Resources (OER) will be leveraged throughout.

## Learning Outcomes

By the end of the course, students will be able to:

- **Bridge Data Science and Software Engineering:**
  - Integrate analysis, visualization, and interpretation into interactive web-based projects.
- **Develop Fundamental Web Skills:**
  - Build and style web pages using HTML and CSS; add interactivity with JavaScript.
- **Utilize Dynamic Data:**
  - Connect to and display live data via APIs, embedding dynamic content into web applications.
- **Deploy and Host Software:**
  - Publish projects online using a range of platforms, from GitHub Pages and Vercel to AWS.
- **Embrace Informed Data Stewardship:**
  - Critically evaluate the implications of self-hosting versus using pre-built platforms, addressing issues of privacy, scalability, and ethical data management.
- **Manage Projects Collaboratively:**
  - Use version control (Git) to track and collaborate on project development.
- **Deliver a Capstone Project:**
  - Design, develop, and deploy a comprehensive project that demonstrates the full cycle—from data science to published software.

---

# Weekly Schedule

### Week 1 – Introduction: From Analysis to Online Impact

- **Topics:**
  - Course orientation and overview of the data-to-deployment pipeline.
  - What it means to build, share, and publish data-driven software.
- **Readings:**
  - Selected OER readings on digital publishing and the evolving landscape of data science.
- **Lab:**
  - Setting up development environments, creating accounts (GitHub, code editors).
- **Discussion:**
  - What are the challenges and opportunities in sharing data science work online?

### Week 2 – Building Your Web Foundation: HTML & CSS Basics

- **Topics:**
  - Introduction to HTML structure and CSS styling.
  - Principles of accessible and responsive design.
- **Readings:**
  - OER modules (MDN Web Docs, W3Schools) on HTML/CSS fundamentals.
- **Lab:**
  - Create a basic webpage that outlines a data project idea.
- **Discussion:**
  - How does design influence the communication of data insights?

### Week 3 – Adding Interactivity: JavaScript Essentials

- **Topics:**
  - JavaScript fundamentals: syntax, DOM manipulation, and event handling.
  - Enhancing static pages with interactive elements.
- **Readings:**
  - Beginner-friendly tutorials (OER from freeCodeCamp, MDN).
- **Lab:**
  - Implement simple interactive features (e.g., buttons, toggles) on your webpage.
- **Discussion:**
  - In what ways can interactivity improve the user experience of data-driven software?

### Week 4 – Integrating Data Science Outputs into Web Projects

- **Topics:**
  - Strategies for embedding analysis results—charts, tables, and narrative summaries—into web pages.
  - Converting static reports into interactive experiences.

- **Readings:**
  - OER content on embedding dynamic content into websites.
- **Lab:**
  - Import a pre-generated data analysis (e.g., CSV data, a static chart) into your webpage.
- **Discussion:**
  - How can we best showcase data science work through software beyond simple visualizations?

**Week 5 – Dynamic Data & API Integration**

- **Topics:**
  - Understanding APIs and RESTful services.
  - Fetching and displaying live data from external sources.
- **Readings:**
  - Tutorials on JavaScript's fetch API and handling JSON data.
- **Lab:**
  - Build a mini-project that pulls real-time data from a public API and displays it.
- **Discussion:**
  - What are the benefits and challenges of incorporating dynamic data into your projects?

**Week 6 – The Publishing Landscape: Platforms & Paradigms**

- **Topics:**
  - Overview of web hosting, domain names, and DNS basics.
  - Conceptual differences between self-publishing your own projects versus using pre-built platforms (e.g., blogs, hosted notebooks).
- **Readings:**
  - OER articles on digital publishing strategies and platform comparisons.
- **Lab:**
  - Set up and publish a basic site using GitHub Pages.
- **Discussion:**
  - How do platform choices affect control, scalability, and data stewardship?

**Week 7 – Deployment Platforms I: GitHub Pages, Vercel, and Netlify**

- **Topics:**
  - Deep dive into free hosting platforms: features, limitations, and best-use cases.
- **Readings:**
  - Official documentation and case studies on GitHub Pages, Vercel, and Netlify.
- **Lab:**
  - Deploy your project on at least one of these platforms.
- **Discussion:**
  - What trade-offs exist when choosing between these services?

**Week 8 – Deployment Platforms II: Cloud Hosting with AWS**

- **Topics:**
  - Introduction to cloud services and AWS basics.
  - When to consider scalable, flexible hosting solutions.
- **Readings:**
  - OER guides on AWS deployment and cloud hosting best practices.
- **Lab:**
  - Deploy a simple application using AWS (e.g., AWS Amplify or Lightsail).
- **Discussion:**
  - How does cloud hosting influence your project's potential reach and maintenance?

**Week 9 – Version Control & Collaborative Development**

- **Topics:**
  - Introduction to Git and GitHub for managing code and collaboration.
  - Best practices for version control in data-driven projects.
- **Readings:**
  - OER tutorials on Git basics and collaborative workflows.
- **Lab:**
  - Create a repository, commit changes, and collaborate on code updates.
- **Discussion:**
  - How does effective version control support data stewardship and project integrity?

**Week 10 – Security, Privacy, and Ethical Data Stewardship**

- **Topics:**
  - Fundamentals of web security, protecting sensitive data, and ethical publishing.
  - Understanding the responsibilities of data stewardship when sharing software online.
- **Readings:**
  - Selected OER content on web security and data ethics.
- **Lab:**
  - Implement basic security measures in your web project (e.g., HTTPS, secure API calls).
- **Discussion:**
  - What ethical considerations must be addressed when publishing data-driven software?

**Week 11 – User Experience & Software Engineering Best Practices**

- **Topics:**
  - Refining user interfaces and ensuring a smooth user experience.
  - Principles of clean code, maintainability, and iterative development.

- **Readings:**
  - OER materials on UI/UX design and software engineering fundamentals.
- **Lab:**
  - Refine your project's design and interactivity based on user feedback.
- **Discussion:**
  - How do design and engineering practices work together to enhance project impact?

**Week 12 – Capstone Project: Planning & Prototyping**

- **Topics:**
  - Brainstorming and scoping your capstone project.
  - Integrating analysis, interactivity, and ethical considerations into a unified software product.
- **Readings:**
  - OER guides on project planning and agile development methodologies.
- **Lab:**
  - Workshop session: outline your project, draft a proposal, and receive peer feedback.
- **Discussion:**
  - What are the key components of a successful data-driven software project?

**Week 13 – Capstone Project Development & Iteration**

- **Topics:**
  - Focused development time for refining your capstone project.
  - Iterative testing and debugging across multiple platforms.
- **Lab:**
  - In-class work session to troubleshoot and polish projects.
- **Discussion:**
  - What challenges have arisen during development, and how can they be resolved?

**Week 14 – Final Presentations & Reflection**

- **Topics:**
  - Capstone project presentations.
  - Reflecting on the journey from data analysis to deployed software.
- **Lab/Presentation:**
  - Each student (or group) presents their fully deployed project, highlighting design decisions, technical challenges, and ethical considerations.
- **Discussion:**
  - How will you apply these skills in your future work as a data steward and software creator?

---

# Assignments & Grading

- **Participation & Weekly Reflections (20%):**
  - Engage actively in class discussions and submit brief reflective posts on weekly readings/labs.
- **Hands-On Labs & Mini-Projects (30%):**
  - Complete weekly lab assignments that build progressively toward your capstone project.
- **Capstone Project (40%):**
  - Develop and deploy a comprehensive project that integrates data analysis, interactive software, and ethical data stewardship.
  - Includes a project proposal, periodic progress updates, and a final presentation.
- **Peer Reviews & Workshops (10%):**
  - Provide constructive feedback during in-class workshops and project review sessions.

---

# Key Texts & Resources

- **OER Materials:**
  - MDN Web Docs, W3Schools, and freeCodeCamp for HTML, CSS, JavaScript, and API integration.
  - Open access articles on digital publishing, web hosting, and data ethics.
- **Platform Documentation:**
  - GitHub Pages, Vercel, Netlify, AWS, and Git/GitHub tutorials.
- **Supplementary Readings:**
  - Case studies on self-hosting vs. pre-built platforms and guides on ethical data stewardship.

---

# Concluding Vision

This course empowers you to transform data science outputs into robust, interactive, and ethically managed software projects. By learning how to build, share, and deploy your own digital products, you'll gain not only technical proficiency but also the critical perspective necessary for informed data stewardship in an increasingly connected world.