

# Advanced Web Development Syllabus

---



## Contact Info

While going through the program, if you have questions about anything, you can reach us at [support@udacity.com](mailto:support@udacity.com). For help from Udacity Mentors and your peers visit the Udacity Classroom.

## Nanodegree Program Info

**Version:** 1.0.0

**Length of Program:** 77 Days\*

*\* This is a self-paced program and the length is an estimation of total hours the average student may take to complete all required coursework, including lecture and project time. Actual hours may vary.*

## Part 1: Introduction to the Program

Welcome to Egypt "Future Work is Digital" initiative. Learn more about the program and how to navigate the classroom.

## Part 2: SQL and Data Modeling for the Web

In this part, you'll be building out the data models and database for an artist/venue booking application. The fictitious startup Fy-yur is building a website that facilitates bookings between artists who can play at venues, and venues who want to book artists.

### Project: Fyyur: Artist Booking Site

In this project, you'll demonstrate your new SQL and data modeling skills by creating a site to help coordinate bookings between artists and venues.

### Supporting Lessons

Lesson	Summary
<b>Course Introduction</b>	Understand what this course will cover and the learning objectives achieved.
<b>Interacting with Databases</b>	Interact with relational databases. Review SQL and the Client-Server Model. Use Postgres, understand DBAPIs, and use psycopg2.
<b>SQLAlchemy Basics</b>	Learn to use SQLAlchemy and SQLAlchemy ORM to work with a relational database in Python.
<b>SQLAlchemy ORM in Depth</b>	Get to know Model.query and the SQLAlchemy Object Lifecycle to master how to query for and change records in a database.
<b>Build a CRUD App with SQLAlchemy - Part 1</b>	Build out the ability to read and create todo items in our To-do app, handling changes from the database to the views.
<b>Migrations</b>	Handle changes to your database schema over time using a version control system involving migration files.
<b>Build a CRUD App with SQLAlchemy ORM - Part 2</b>	Finish developing our To-Do app with update and delete functionality. Model relationships with To-Do lists.

## Part 3: API Development and Documentation

In this part, you will use the skills you've developed to build a Trivia API. The goal of this project is to use APIs to control and manage a web application using existing data models. You'll be given a set of data models and the application front end. Your task will be to implement the API in Flask to make the Trivia game functional.

### Project: Trivia API

In this project, you'll use your new API and API documentation skills to build a trivia API.

### Supporting Lessons

## Lesson

## Summary

---

### Introduction to APIs

In this lesson, you'll learn what APIs are, how they're implemented, and why they're important. We'll also dive into the concepts of Internet Protocols and REST.

---

### HTTP and Flask Basics

In this lesson, you'll learn about HTTP—including methods, requests, responses and status codes— as well as how to set up a Flask app, and implement and test endpoints using Curl.

---

### Endpoints and Payloads

In this lesson, you'll learn about how Flask is extensible in itself to handle different kinds of methods, more complex endpoints, and to return formatted data to the client.

---

### API Testing

This lesson teaches students the benefits and purposes of testing an API, Test-Driven Development for APIs and implementing the tests using unittest.

---

### API Documentation

To round out the course, students will consider and write API documentation and project documentation so their projects can be hosted, shared, and used by other developers with clarity.

---

## Part 4: Identity and Access Management

In this part, you will build the backend for a coffee shop application. You'll add user accounts and authentication to your application and use role-based access management strategies to control different types of user behavior in the app.

### Project: Coffee Shop Full Stack

In this project, you'll demonstrate your new authentication and authorization skills by creating a full-stack application for a coffee shop menu.

### Supporting Lessons

Lesson	Summary
<b>Foundation</b>	Set the groundwork for understanding information security and refresh your understanding of the technologies used in future lessons.
<b>Identity and Authentication</b>	Explore frequently used methods of identifying who is making requests on web systems. Implement modern software patterns to accomplish this goal across the stack.
<b>Passwords</b>	Understand and overcome common pitfalls of the ubiquitous password authentication design pattern.
<b>Access and Authorization</b>	Limit access to specific resources or actions by restricting requests only to authorized request to particular users and groups of users. Implement role-based access controls (RBAC) across the stack.
<b>Thinking Adversarially</b>	Stay one step ahead of attackers by implementing a secure development process and knowing how to keep informed on the cutting edge of security research.

## Part 5: Digital Freelancing - Intro

Get introduced to the world of digital freelancing.

## Part 6: Digital Freelancing - Pro

Get started with digital freelancing and submit your first proposal.

### Project: Conquer the freelancing marketplaces

#### Supporting Lessons

## مخطط البرنامج

## المنصات التخصصية

ستتعرف على أهمية مواقع المنصات التخصصية مثل Github, Behance, Dribbble, Stack Overflow, Kaggle للاستفادة من تجارب الآخرين و أيضا للترويج لعملك كفري لأنسرفي المكان المناسب.

## إنشاء مدونة

## إدارة العملاء

## عرض العمل (proposal)

## مهارات التواصل

## التسويق

## إدارة المشاريع

## الإدارة الذاتية

## تطوير المهارات

## الإنضمام للوكالات

## Part 7: Congratulations and Next Steps



Udacity

Generated Sat Jan 9 04:00:31 PST 2021