**Software Engineering Bootcamp - Course Outline**

**"Gig Ready Developer Program"**

**Duration:** 16 weeks | 400 hours total  
**Schedule:** Mon-Thu: 6.5 hrs/day | Fri: 4 hrs/day  
**Start Date:** February 17, 2025  
**Languages:** JavaScript & Python (Equal Focus)

**MONTH 1: FOUNDATIONS & PROFESSIONAL TOOLS (100 Hours)**

**Week 1: "Setting Up Your Gear" - Professional Development Environment**

**Feb 17-21, 2025 | 30 hours**

* **Monday:** Gig Ready introduction, command line mastery, professional identity setup
* **Tuesday:** Git fundamentals - repos, commits, branching, collaboration
* **Wednesday:** IDE optimization (VS Code), productivity tools, professional shortcuts
* **Thursday:** Agile methodology, standups, sprints, user stories
* **Friday:** Project management tools (Jira, GitHub Issues), estimation basics

**Assignment:** Professional development environment setup with GitHub profile, IDE configuration, first repository, developer journey blog post, project management board

**Week 2: "Learning Your Instruments" - Programming Fundamentals (JavaScript & Python)**

**Feb 24-28, 2025 | 30 hours**

* **Monday:** Programming paradigms, Node.js and Python environments, REPL exploration
* **Tuesday:** Variables, data types, operators - side-by-side JS/Python examples
* **Wednesday:** Control structures (if/else, loops) - both languages, highlighting differences
* **Thursday:** Functions and scope - comparing function syntax and behavior
* **Friday:** Introduction to testing mindset, running simple tests in both languages

**Assignment:** Dual Language Programming Challenge - 10 problems in BOTH JavaScript and Python, document language differences, peer code review, present solution comparison

**Week 3: "Understanding Data Architecture" - Database & Data Fundamentals**

**Mar 3-7, 2025 | 30 hours**

* **Monday:** Data architecture fundamentals, data types, structures, why they matter
* **Tuesday:** Introduction to SQL databases - tables, relationships, basic queries
* **Wednesday:** Introduction to NoSQL concepts - documents, key-value stores
* **Thursday:** Data modeling fundamentals - designing schemas, normalization basics
* **Friday:** Data in real applications - API responses, JSON, CSV, how apps consume data

**Assignment:** Data Architecture Foundation Project - design database schema, write SQL queries (online sandbox), create JSON structures, document trade-offs, present decision with reasoning

**Week 4: "Data Structures & Storage" - Connecting Code to Data**

**Mar 10-14, 2025 | 30 hours**

* **Monday:** Arrays (JS) vs Lists (Python) - methods, iteration, comprehensions; relation to spreadsheet columns
* **Tuesday:** Objects (JS) vs Dictionaries (Python) - key-value data; Excel as database metaphor; CSV processing
* **Wednesday:** JSON structured data; reading/writing JSON; converting CSV to JSON
* **Thursday:** Relational database concepts - Excel sheets as tables, primary/foreign keys, basic SQL
* **Friday:** Algorithm basics - searching, sorting, filtering data; Month 1 Sprint Review & Retrospective

**Assignment:** Data Management System - Excel to Code: Create Excel workbook with 3 related sheets, build data processing app in BOTH languages reading CSV files, implement filtering/sorting/aggregation, convert between CSV and JSON, simulate JOIN operations, technical blog post, professional Git workflow

**MONTH 2: WEB DEVELOPMENT & LANGUAGE MASTERY (125 Hours)**

**Week 5: "Soundcheck & Setup" - Web Fundamentals**

**Mar 17-21, 2025 | 30 hours** *(Note: Mar 19 is holiday - Josefstag)*

* **Monday:** How the web works - HTTP, requests/responses, client-server model
* **Tuesday:** Semantic HTML, accessible CSS, professional standards
* **Wednesday:** Package management - npm (JavaScript) and pip (Python)
* **Thursday:** Project: Build professional portfolio site with Git workflow
* **Friday:** Code review session, documentation standards

**Assignment:** Professional Portfolio Website v1.0 - responsive site with HTML/CSS, semantic HTML, accessible design, mobile-responsive, host on GitHub Pages, complete Git workflow with feature branches/PRs, professional README, highlight both JavaScript and Python skills

**Week 6: "JavaScript for the Web" - DOM & Interactivity**

**Mar 24-28, 2025 | 30 hours**

* **Monday:** JavaScript in the browser - the Document Object Model (DOM)
* **Tuesday:** Element selection, manipulation, professional DOM patterns
* **Wednesday:** Event handling, event delegation, form validation
* **Thursday:** Accessibility considerations, keyboard navigation
* **Friday:** Professional project work and code review

**Assignment:** Interactive Dashboard Project - data dashboard with dynamic content (weather simulation, task tracker), form handling, data visualization, user feedback, professional error handling, accessibility features, performance optimization documentation, load data from JSON/CSV files

**Week 7: "Python for the Web" - Backend Basics with Flask**

**Mar 31 - Apr 4, 2025 | 30 hours**

* **Monday:** Introduction to Flask, routing, handling requests
* **Tuesday:** Working with JSON, building API endpoints
* **Wednesday:** Connecting frontend to backend - CORS, fetch API
* **Thursday:** Data persistence with files and SQLite
* **Friday:** Full-stack integration workshop

**Assignment:** Full-Stack Application (JS Frontend + Python Backend) - JavaScript frontend with Flask backend, backend with 3+ RESTful endpoints, frontend consumes endpoints, data persistence (SQLite or CSV), professional error handling on both ends, complete API documentation, environment variable management

**Week 8: Easter Break & Async Learning**

**Apr 7-11, 2025 | Self-Study** *(Note: Easter Holiday Week - Apr 14-21)*

* Complete online courses on async JavaScript and Python
* Read professional development blogs and documentation
* Contribute to open-source project (documentation, small bug fix)
* Review and refactor previous projects based on new learning
* Prepare for Month 2 capstone project

**Assignment:** Open Source Contribution - find and contribute to project (any language), document contribution process, reflect on professional collaboration, present contribution and learning in sprint review

**Week 9: "Asynchronous Programming" - Async in JavaScript & Python**

**Apr 22-25, 2025 | 35 hours**

* **Monday:** Callbacks and Promises (JavaScript), async foundations
* **Tuesday:** async/await in JavaScript, error handling
* **Wednesday:** Python asyncio, async/await in Python, comparing approaches
* **Thursday:** Working with real APIs - authentication, rate limiting, error handling
* **Friday:** Sprint Review - async project demonstrations

**Assignment:** Dual-Language API Integration - two versions (JavaScript/Node.js and Python/Flask), both consume 2+ external APIs, authentication, error handling, loading states, rate limiting, technical comparison document, deploy both versions

**MONTH 3: ADVANCED DEVELOPMENT & TESTING (100 Hours)**

**Week 10: "Testing & Quality" - Professional Testing Practices**

**Apr 28 - May 2, 2025 | 30 hours** *(Note: May 1 is holiday - Staatsfeiertag)*

* **Monday:** Testing fundamentals - unit, integration, e2e tests
* **Tuesday:** JavaScript testing with Jest, writing effective tests
* **Wednesday:** Python testing with pytest, comparing testing approaches
* **Thursday:** Test-Driven Development (TDD) workshop
* **Friday:** Code quality tools - linting, formatting, CI/CD basics

**Assignment:** Test-Driven Development Project - calculator/converter using TDD in both languages, write tests first then implementation, achieve 80%+ code coverage in both versions, professional CI/CD setup (GitHub Actions), document testing strategy and comparisons, performance benchmarking

**Week 11: "Modern Frontend" - React Fundamentals**

**May 5-9, 2025 | 30 hours** *(Note: May 9 is holiday - Christi Himmelfahrt)*

* **Monday:** React fundamentals, JSX, component architecture
* **Tuesday:** State and props, lifting state up, component communication
* **Wednesday:** Hooks - useState, useEffect, custom hooks
* **Thursday:** React with TypeScript introduction
* **Friday:** Professional React patterns and project work

**Assignment:** React + Python Backend Application - React frontend with Flask/FastAPI backend, complex state management, multiple components, API integration with Python backend, professional project structure, deployed to cloud platform, complete user documentation

**Week 12: "Advanced Backend & Databases" - Deep Dive**

**May 12-16, 2025 | 30 hours**

* **Monday:** Advanced Flask/FastAPI - middleware, authentication, security
* **Tuesday:** PostgreSQL deep dive - complex queries, joins, transactions
* **Wednesday:** SQLAlchemy (Python ORM) vs Sequelize (Node.js ORM)
* **Thursday:** Database optimization - indexes, query performance, N+1 problems
* **Friday:** Professional backend architecture patterns

**Assignment:** Production-Grade Backend API - RESTful API with Node.js/Express or Python/FastAPI, JWT authentication, role-based authorization, PostgreSQL database with proper relationships, comprehensive test suite, API documentation (Swagger/OpenAPI), deployed with proper environment management

**Week 13: "Month 3 Integration" - Full-Stack Capstone**

**May 19-23, 2025 | 30 hours**

* **Monday:** Sprint planning, architecture design
* **Tuesday:** Backend implementation, database setup
* **Wednesday:** Frontend development, integration
* **Thursday:** Testing, deployment, documentation
* **Friday:** Sprint Review, demonstrations, retrospective

**Assignment:** Full-Stack Professional Application - React frontend with complex UI, Python or Node.js backend with authentication, PostgreSQL database with proper schema, comprehensive testing (frontend and backend), deployed to production with CI/CD pipeline, professional documentation and performance monitoring

**MONTH 4: DATA ARCHITECTURE DEEP DIVE & SPECIALIZATION (75 Hours)**

**Week 14: "Data Architecture Deep Dive" - Advanced Database Concepts**

**May 26-30, 2025 | 30 hours**

* **Monday:** NoSQL deep dive - MongoDB document databases, schema design
* **Tuesday:** Graph databases (Neo4j) - when and why to use them
* **Wednesday:** Time-series databases - InfluxDB, real-time data scenarios
* **Thursday:** Caching strategies - Redis, memcached, application-level caching
* **Friday:** Data architecture decision framework, polyglot persistence patterns

**Assignment:** Multi-Database Architecture Project - design and implement application using 3 different database types (e.g., PostgreSQL + MongoDB + Redis), justify each database choice with documentation, implement data migration scripts, write comprehensive architectural decision records (ADRs), performance benchmarking comparing approaches

**Week 15: "Modern Development Tools" - Docker, TypeScript, & DevOps**

**June 2-6, 2025 | 30 hours**

* **Monday:** Docker fundamentals - containers, images, docker-compose
* **Tuesday:** TypeScript introduction - types, interfaces, type safety in production
* **Wednesday:** Monitoring and logging - application observability
* **Thursday:** Deployment strategies - CI/CD, blue-green deployments, feature flags
* **Friday:** Professional development workflow integration

**Assignment:** Containerized Full-Stack Application with TypeScript - convert previous project to use TypeScript (frontend and backend), Docker containers for all services (including databases), Docker Compose for orchestration, professional logging and monitoring setup, health checks and graceful shutdown, production-ready deployment configuration, complete infrastructure documentation

**Week 16: "Professional Capstone Project" - Part 1**

**June 9-13, 2025 | 30 hours**

* **Monday:** Project kick-off, requirements gathering, sprint planning
* **Tuesday:** Architecture design, technology selection (JS/Python/Both), database design
* **Wednesday:** Backend development, API implementation
* **Thursday:** Database setup, data modeling, migrations
* **Friday:** Sprint review, retrospective, planning Sprint 2

**Assignment:** Real-World Professional Application - Sprint 1: Must demonstrate both JavaScript AND Python skills, multiple database types used appropriately, professional testing suite setup, CI/CD pipeline configuration, Docker containerization, project plan and architecture documentation, technology stack justification

**Week 17: "Professional Capstone Project" - Part 2 & Career Readiness**

**June 16-20, 2025 | 15 hours**

* **Monday:** Frontend development, component architecture
* **Tuesday:** Frontend-backend integration, authentication flow
* **Wednesday:** Testing, security hardening, performance optimization
* **Thursday:** Final polish, documentation, deployment
* **Friday:** Final presentations, "Gig Ready Certification"

**Assignment:** Final Capstone Delivery & Portfolio Finalization - complete full-stack application with all features, comprehensive test coverage and professional documentation, deployed application with monitoring, post-mortem documentation, updated portfolio website with all 16 weeks of projects, GitHub profile showcasing professional work, LinkedIn profile optimization, technical presentation demonstrating professional communication, resume highlighting dual-language skills

**Assessment & Evaluation**

**Continuous Assessment (60%)**

* Weekly assignments and projects (both languages)
* Code review participation
* Professional Git workflow adherence
* Documentation quality
* Agile ceremony participation
* Peer collaboration

**Major Projects (30%)**

* Month 1 Integration (Week 4) - 5%
* Month 2 Full-Stack Project - 5%
* Month 3 Integration Project - 10%
* Final Capstone - 10%

**Professional Skills (10%)**

* Communication in standups/retrospectives
* Code review quality
* Documentation practices
* Presentation skills
* Professional conduct

**Graduate Outcomes**

Upon completion, graduates will have:

* 16+ projects in professional GitHub portfolio (JavaScript AND Python)
* 150+ commits demonstrating professional Git workflow
* Dual-language expertise - JavaScript and Python
* Data architecture knowledge - Excel to complex multi-database systems
* Professional portfolio website showcasing all work
* Full-stack development skills - frontend (HTML/CSS/React) + backend (Node.js/Flask)
* Database expertise - CSV, JSON, SQL, NoSQL, caching
* Testing experience - Jest and pytest
* Deployment experience - containerization and cloud deployment
* Professional practices - Agile, code review, documentation, CI/CD
* Interview preparation - technical and behavioral
* Real professional experience working like professional software engineers

*Digital Campus Vorarlberg | Feldkirch, Austria*  
*Instructors: Oliver Bedogg & Frank Blau*