

(Web Programming)

รหัสรายวิชาเดิม : ENGCE176 การเขียนโปรแกรมบนเว็บ

วิชาบังคับก่อน : ENGCE126 ระบบฐานข้อมูล

ศึกษาและฝึกปฏิบัติการเกี่ยวกับ เทคโนโลยีของอินเทอร์เน็ต โพรโทคอล HTTP กลไกจัดการการร้องขอในเว็บเซิร์ฟเวอร์ การเขียนโปรแกรมแบบ CGI และการสร้างหน้าเว็บแบบพลวัต การใช้งานคุกกี้ การติดต่อเชื่อมโยงกับฐานข้อมูล การปรับแต่งประสิทธิภาพและความปลอดภัยในการใช้งาน การเขียนโปรแกรมในฝั่งของ Client เช่น HTML, CSS, JavaScript และ ฝั่งของ Server เช่น PHP, ASP, JSP, AJAX เป็นต้น

Study and practice of an Internet technology and how the computers are communication between client and server take place using the HTTP. We will explore the basic mechanisms and concepts of process of web servers such as how to use cookies in JavaScript, database and security management for web developers. Students will gain hands-on experience with tools and techniques that simplify the creation and maintenance of dynamic web page, client program: HTML, CSS and JavaScript, and server program such as CGI, PHP, ASP, JSP, and AJAX.

ตารางที่ 1 แผนการสอนรายวิชา การเขียนโปรแกรมบนเว็บ (Web Programming)

สัปดาห์	หัวข้อการเรียนการสอน	กิจกรรม / การบ้าน / การประเมินผล
1	แนะนำรายวิชา + พื้นฐานอินเทอร์เน็ตและ HTTP	<ul style="list-style-type: none">- แนะนำตัวและระบบการเรียน- ติดตั้งเครื่องมือพัฒนา (VS Code, Git)- แบบทดสอบก่อนเรียน (Pre-test)- Quiz: ความเข้าใจเบื้องต้นเกี่ยวกับเว็บและ HTTP
2	HTML5 และโครงสร้างเว็บไซต์	<ul style="list-style-type: none">- ปฏิบัติ: สร้างหน้าโปรไฟล์ด้วย HTML5- การบ้าน: ออกแบบ Resume เป็นหน้า HTML- ให้ความรู้ด้าน Semantic Elements และ Accessibility
3	CSS3 และการออกแบบให้รองรับทุกอุปกรณ์	<ul style="list-style-type: none">- ปฏิบัติ: จัดเลย์เอาต์ด้วย Flexbox และ Grid- การบ้าน: สร้าง Portfolio ที่ Responsive- แนะนำ Tailwind CSS สำหรับจัดรูปแบบ

ตารางที่ 1 แผนการสอนรายวิชา การเขียนโปรแกรมบนเว็บ (Web Programming) (ต่อ)

สัปดาห์	หัวข้อการเรียนการสอน	กิจกรรม / การบ้าน / การประเมินผล
4	JavaScript พื้นฐาน (ES6+)	<ul style="list-style-type: none"> - ปฏิบัติ: เขียน JavaScript จัดการเหตุการณ์ (Events) และ DOM - Quiz: พื้นฐาน JavaScript
5	JavaScript ขั้นสูง + การจัดเก็บข้อมูลฝั่งผู้ใช้	<ul style="list-style-type: none"> - ปฏิบัติ: ตรวจสอบฟอร์ม + เก็บข้อมูลด้วย Local Storage - การบ้าน: สร้างแอป Todo List ที่บันทึกข้อมูลได้
6	AJAX และการเชื่อมต่อกับ API ภายนอก	<ul style="list-style-type: none"> - ปฏิบัติ: ดึงข้อมูลจาก OpenWeatherMap API - การบ้าน: สร้างแดชบอร์ดแสดงข้อมูลสภาพอากาศแบบเรียลไทม์
7	การเขียนโปรแกรมฝั่งเซิร์ฟเวอร์: Node.js + Express	<ul style="list-style-type: none"> - ปฏิบัติ: เขียน Routing เบื้องต้นด้วย Express.js - Quiz: แนวคิดการทำงานฝั่งเซิร์ฟเวอร์ - แจกโจทย์ Mini Project สำหรับสอบกลางภาค
8	การเชื่อมต่อฐานข้อมูล (MySQL หรือ MongoDB)	<ul style="list-style-type: none"> - ปฏิบัติ: สร้างระบบ CRUD และเชื่อมต่อฐานข้อมูล - ทดสอบ API ด้วย Postman
9	การสร้าง RESTful API + Postman	<ul style="list-style-type: none"> - ปฏิบัติ: พัฒนา API สำหรับระบบ Blog หรือสินค้า - การบ้าน: สร้าง REST API พร้อมใช้งานจริง
10	การยืนยันตัวตนและความปลอดภัยเว็บไซต์	<ul style="list-style-type: none"> - ปฏิบัติ: ระบบล็อกอินด้วย JWT หรือ Session - Quiz: แนวคิดความปลอดภัยเว็บ - ส่งงาน Mini Project (สอบกลางภาค)
11	เครื่องมือสำหรับนักพัฒนา + Git & GitHub	<ul style="list-style-type: none"> - ปฏิบัติ: สร้าง GitHub Repo และใช้งาน Git Workflow - ฝึกทำ Pull Request และ Code Review แบบทีม
12	การนำเว็บไซต์ขึ้นออนไลน์ (Deploy)	<ul style="list-style-type: none"> - ปฏิบัติ: นำเว็บไซต์ขึ้น Netlify, Vercel หรือ Railway - การบ้าน: นำระบบของตนเองขึ้นออนไลน์จริง
13	เวิร์กช็อปโปรเจกต์ปลายภาค (Final Project)	<ul style="list-style-type: none"> - นักศึกษาลงมือพัฒนาโปรเจกต์จริงตามกลุ่ม - ตรวจสอบความคืบหน้าโค้ดและ Deployment
14	นำเสนอโปรเจกต์ปลายภาค + สรุปรวิชา	<ul style="list-style-type: none"> - นักศึกษา Demo ผลงานเว็บแอป + ตอบคำถาม - ทำแบบประเมินหลังเรียน (Post-test) - ให้ Feedback และแนะแนวการต่อยอด

ตารางที่ 2 การประเมินผลรายวิชา การเขียนโปรแกรมบนเว็บ (Web Programming)

ประเภทการประเมินผล	รายละเอียด	สัดส่วน (%)
งานเดี่ยว / การบ้าน (Assignment)	<ul style="list-style-type: none"> - งานเดี่ยวรายสัปดาห์ เช่น HTML Resume, Portfolio, Todo App, REST API, Deploy - ตรวจสอบความถูกต้องของโค้ด, ความครบถ้วน, UX/UI และการส่งงานผ่าน GitHub 	25%
แบบทดสอบย่อย / กิจกรรมในห้อง (Quiz & Lab)	<ul style="list-style-type: none"> - Quiz 3-4 ครั้งในหัวข้อสำคัญ เช่น HTTP, JS, Security - กิจกรรมในห้อง เช่น Pair Programming, Debugging, Code Review 	15%
มินิโปรเจกต์ (สอบกลางภาค)	<ul style="list-style-type: none"> - พัฒนาแอปเล็ก เช่น Blog API หรือ CRUD App - ต้องใช้ Express.js + DB และ Deploy จริง 	20%
โปรเจกต์ปลายภาค (Final Project + การนำเสนอ)	<ul style="list-style-type: none"> - พัฒนาเว็บแอปเต็มรูปแบบเป็นกลุ่ม (3-5 คน) - มี Frontend + Backend + Database + REST API + Deployment - นำเสนอหน้าชั้น พร้อมตอบคำถาม 	30%
จิตพิสัย / วินัยการเรียน	<ul style="list-style-type: none"> - การมาเรียนตรงเวลาและสม่ำเสมอ - การส่งงานตรงเวลา - การมีส่วนร่วมในห้องเรียน/กลุ่ม - ความมีวินัย ความรับผิดชอบ 	10%
รวม		100%