



Software Testing Foundation

พื้นฐานการทดสอบซอฟต์แวร์ ตั้งแต่ Requirement ถึง Test Scenario

● 12 ชั่วโมง

● 2 วัน

● ไม่ต้องเขียน โปรแกรม

● ระดับ Beginner - Inter



ระยะเวลา: 2 วัน

12 ชั่วโมง (2 วัน วันละ 6 ชม.) · 09:00-16:00 น.



รูปแบบ: Classroom / Hybrid

Onsite หรือ Online ผ่าน Microsoft Teams · Public / In-house



รายละเอียดหลักสูตร

www.itgenius.co.th

ซอฟต์แวร์ที่มีปัญหาหลังส่งมอบส่วนใหญ่ไม่ได้ฟังเพราะไม่มีการทดสอบ แต่ฟังเพราะทดสอบไม่ตรงจุด Requirement ที่ตกลงหรือขัดกันเองไม่ถูกจับได้ตั้งแต่ต้น Test Case เขียนกว้างจนคนอื่นทำตามไม่ได้ และไม่มีใครรู้ว่าสิ่งที่ทดสอบไปครอบคลุมการใช้งานจริงของผู้ใช้แล้วหรือยัง การทดสอบที่ดีจึงเริ่มตั้งแต่การทำความเข้าใจ Requirement และต้องมีวิธีคิดที่เป็นระบบ ไม่ใช่แค่ลองกดดูว่าใช้ได้หรือไม่

หลักสูตรนี้ปูพื้นฐานการทดสอบซอฟต์แวร์ตั้งแต่บทบาทของการทดสอบในแต่ละเฟสของ SDLC การรวบรวม Requirement จากผู้เกี่ยวข้อง การหา Requirement ที่ขาดหายหรือขัดแย้งกันแล้วแปลงเป็น Test Case ประเภทของการทดสอบและวิธีเลือกใช้ ความต่างระหว่าง Test-Last กับ Test-First และการเขียน Test Case ที่แม่นยำมีโครงสร้างครบ วันที่สองต่อยอดไปสู่การจัดลำดับ Test Case ด้วย Risk Analysis การใช้ Generative AI ช่วยออกแบบ Test Case เทคนิคการออกแบบ Test Scenario ให้ครอบคลุม Requirement การวิเคราะห์ User Scenario และ System Flow การออกแบบ Scenario แบบ End-to-End ตามกระบวนการธุรกิจ และการจัดการ Defect อย่างเป็นระบบ เรียนผ่าน Workshop ที่ใช้โจทย์ต่อเนื่องกันตลอดสองวัน (อบรม 2 วัน วันละ 6 ชั่วโมง รวม 12 ชั่วโมง ระดับ Beginner ถึง Intermediate)

เครื่องมือ

SD SDLC

TC Test Case

RB Risk-based

E2E E2E Scenario

AI Generative AI

ราคาค่าอบรม · PUBLIC / IN-HOUSE

4,900 บาท / ท่าน

ราคายังไม่รวมภาษีมูลค่าเพิ่ม (VAT 7%) · รวมเอกสารประกอบการอบรม แบบฝึกหัดทุก Workshop และ Template สำหรับ Test Case, Test Scenario และ Defect Report อาหารกลางวันและอาหารว่าง และ Certificate · รับจัด In-house ขอใบเสนอราคาได้



เพิ่มเพื่อน
LINE



01 วัตถุประสงค์ Objectives

- เข้าใจบทบาทและความสำคัญของการทดสอบในทุกเฟสของ SDLC
- ประยุกต์ใช้เทคนิคการทดสอบเพื่อยกระดับคุณภาพของผลิตภัณฑ์
- วิเคราะห์ปัญหาและออกแบบกระบวนการทดสอบที่ตอบสนองทั้งความต้องการด้านเทคนิคและธุรกิจ
- มีพื้นฐานที่แน่นพอสำหรับต่อยอดสู่งานทดสอบขั้นสูง

02 กลุ่มเป้าหมาย Target Audience

- Tester และ QA ที่ต้องการวางพื้นฐานการทดสอบให้เป็นระบบ
- Developer และ Dev Lead ที่ใส่ใจคุณภาพของซอฟต์แวร์
- Product Owner, Business Analyst และ System Analyst
- ผู้ที่ต้องการเปลี่ยนสายงานมาเป็น Software Tester
- ผู้ที่สนใจการทดสอบซอฟต์แวร์ทุกคน

03 จุดเด่นของหลักสูตร Highlights

- **เริ่มจาก Requirement** จับจุดตกหล่นและขัดแย้งก่อนเขียน Test
- **Test Case ที่แม่นยำ** โครงสร้างครบ คนอื่นทำตามได้
- **จัดลำดับด้วยความเสี่ยง** ทดสอบสิ่งสำคัญก่อนเมื่อเวลาจำกัด
- **ใช้ Generative AI ช่วย** ออกแบบ Test Case ได้เร็วและครบขึ้น
- **ไม่ต้องเขียนโปรแกรม** เน้นวิธีคิดและเทคนิคการออกแบบ

04 ความรู้พื้นฐาน Prerequisites

- ไม่จำเป็นต้องมีพื้นฐานการเขียนโปรแกรม
- เคยใช้งานเว็บไซต์หรือแอปพลิเคชันทั่วไป และเข้าใจภาพรวมการพัฒนาซอฟต์แวร์เบื้องต้นจะเป็นประโยชน์
- มีบัญชี Generative AI สำหรับใช้ใน Workshop วันที่ 2 (แบบใช้ฟรีได้)

05 สิ่งที่ต้องเตรียมและเครื่องมือ Setup & Tools

รายการ	รายละเอียด
อุปกรณ์	Notebook เชื่อมต่ออินเทอร์เน็ต
เอกสารทดสอบ	Spreadsheet สำหรับ Test Case, Test Scenario และ Defect Log
AI ช่วยออกแบบ	เครื่องมือ Generative AI สำหรับช่วยออกแบบ Test Case และ Scenario

หมายเหตุ: หลักสูตรนี้เน้นเทคนิคและวิธีคิดในการทดสอบ ไม่ผูกกับเครื่องมือทดสอบตัวใดตัวหนึ่ง Workshop ทั้งหมดทำบน Spreadsheet และ Template ที่สถาบันไอทีจีเนียสเตรียมให้ จึงนำไปใช้ต่อกับเครื่องมือบริหารงานทดสอบที่องค์กรใช้อยู่ได้ทันที ส่วนเครื่องมือ Generative AI ใช้แบบฟรีได้

06 ภาพรวมการเรียนรู้ 2 วัน Big Picture

วัน	หัวข้อหลัก
Day 1 Section 1-6	บทบาทของการทดสอบใน SDLC, การรวบรวมและวิเคราะห์ Requirement, ประเภทของการทดสอบ, Test-Last กับ Test-First และการเขียน Test Case ที่แม่นยำและมีโครงสร้างครบ
Day 2 Section 7-12	จัดลำดับ Test Case ด้วย Risk Analysis, ใช้ Generative AI ช่วยออกแบบ, เทคนิค Test Scenario, User Scenario และ System Flow, Scenario แบบ End-to-End และการจัดการ Defect



DAY
1

พื้นฐานการทดสอบและการออกแบบ Test

เนื้อหาการอบรม · Section 1-6

รวม
6 ชั่วโมง

1 SDLC และบทบาทของการทดสอบ

- ภาพรวมของ Software Development Life Cycle ทีละเฟส
- การทดสอบมีบทบาทอย่างไรตั้งแต่ Requirement จนถึงการส่งมอบ
- ต้นทุนของ Defect ที่ถูกพบช้ากับที่ถูกพบตั้งแต่ต้น
- ความรับผิดชอบด้านคุณภาพของ Tester, Developer และ BA

2 การพัฒนาและรวบรวม Requirement

- Requirement ที่มีลักษณะอย่างไร
- ระบุผู้เกี่ยวข้อง (Stakeholder) และความต้องการของแต่ละฝ่าย
- เทคนิคการสอบถามและรวบรวม Requirement จากผู้เกี่ยวข้อง
- แยก Functional กับ Non-functional Requirement

3 วิเคราะห์ Requirement สู่ Test Case

- ตรวจสอบ Requirement ที่ขาดหาย คลุมเครือ หรือขัดแย้งกัน
- แปลง Requirement เป็นเงื่อนไขที่ทดสอบได้
- เชื่อม Requirement กับ Test Case ให้ตรวจสอบย้อนกลับได้
- Workshop: วิเคราะห์ Requirement จากโจทย์จริงแล้วร่าง Test Case

4 แนวคิดและประเภทของการทดสอบ

- หลักการพื้นฐานของการทดสอบซอฟต์แวร์
- ระดับการทดสอบ: Unit, Integration, System และ Acceptance
- Functional, Non-functional, Regression และ Smoke Testing
- เลือกประเภทการทดสอบให้เหมาะกับงานและทรัพยากร

5 Test-Last กับ Test-First

- แนวทางการทดสอบหลังพัฒนา (Test-Last)
- แนวทางการคิด Test ก่อนพัฒนา (Test-First)
- ข้อดีข้อจำกัดของแต่ละแนวทาง
- เลือกแนวทางให้เข้ากับทีมและรูปแบบโครงการ

6 เขียน Test Case ที่แม่นยำและมีประสิทธิภาพ

- องค์ประกอบของ Test Case ที่มีโครงสร้างครบ
- เขียน Precondition, Test Step และ Expected Result ให้ชัด
- หลีกเลี่ยง Test Case ที่กว้างหรือกำกวมจนทำตามไม่ได้
- Workshop: เขียนและทบทวน Test Case ร่วมกันในกลุ่ม



DAY
2

การนำไปใช้เชิงลึกและเครื่องมือ

เนื้อหาการอบรม · Section 7-12

รวม
6 ชั่วโมง

7 จัดลำดับ Test Case ด้วย Risk Analysis

- ประเมินความเสี่ยงจากโอกาสเกิดและผลกระทบ
- ระบุส่วนของระบบที่เสี่ยงสูงต่อธุรกิจและผู้ใช้
- จัดลำดับ Test Case เมื่อเวลาทดสอบมีจำกัด
- Workshop: จัดลำดับ Test Case ของโจทย์ตามระดับความเสี่ยง

8 ใช้ Generative AI ออกแบบ Test Case

- ใช้ Generative AI ช่วยแตก Requirement เป็น Test Case
- เขียน Prompt ให้ได้ Test Case ที่ครบและตรงรูปแบบ
- ตรวจสอบผลจาก AI หาสิ่งที่ตกหล่นหรือไม่ถูกต้อง
- Workshop: เทียบ Test Case ที่เขียนเองกับที่ AI ช่วยออกแบบ

9 เทคนิคการออกแบบ Test Scenario

- ความแตกต่างระหว่าง Test Scenario กับ Test Case
- ออกแบบ Scenario ให้ครอบคลุมทุก Requirement
- Scenario ทั้งกรณีปกติ กรณีผิดพลาด และกรณีขอบเขต
- ใช้ตารางตรวจสอบความครอบคลุมของ Requirement

10 User Scenario และ System Flow

- มอง Scenario จากมุมมองของผู้ใช้
- วิเคราะห์ System Flow เพื่อหาเส้นทางที่ต้องทดสอบ
- ตรวจสอบความครบถ้วนของ Scenario
- Workshop: เขียน User Scenario และตรวจสอบความครบถ้วนจาก Flow

11 Scenario แบบ End-to-End ตามกระบวนการธุรกิจ

- ใส่กระบวนการธุรกิจตั้งแต่ต้นจนจบ
- จุดเชื่อมต่อระหว่างระบบที่มักเกิดปัญหา
- ออกแบบ Scenario แบบ End-to-End ที่ใกล้เคียงการใช้งานจริง
- Workshop: ออกแบบ Scenario End-to-End ของกระบวนการตัวอย่าง

12 การจัดการและวิเคราะห์ Defect

- วงจรชีวิตของ Bug และ Defect
- เขียน Defect Report ที่ Developer ทำซ้ำและแก้ไขได้
- กำหนด Severity และ Priority ให้เหมาะสม
- วิเคราะห์แนวโน้มของ Defect เพื่อปรับปรุงกระบวนการ
- สรุปแนวทางนำความรู้ไปใช้ในทีมและต่อยอดสู่งานทดสอบขั้นสูง

07 เรียนจบแล้วได้อะไร Outcomes

- เข้าใจการทดสอบใน SDLC รู้ว่าแต่ละเฟสต้องทดสอบอะไร
- วิเคราะห์ Requirement เป็น จับจุดตกหล่นและขัดแย้งได้ก่อน
- เขียน Test Case ได้ดี แม่นยำ มีโครงสร้างครบ ทำตามได้
- จัดลำดับด้วยความเสี่ยง ทดสอบสิ่งสำคัญก่อนเสมอ
- ออกแบบ Scenario ครบ ทั้ง User Scenario และ End-to-End
- จัดการ Defect เป็นระบบ รายงานชัด วิเคราะห์ต่อได้

08 สิ่งที่คุณเรียนจะได้รับ Deliverables

- เอกสารประกอบการอบรมฉบับสมบูรณ์ครบทั้ง 2 วัน พร้อมแบบฝึกหัดทุก Workshop
- Template สำหรับ Test Case, Test Scenario และ Defect Report ที่นำไปใช้ได้ทันที
- ตัวอย่าง Prompt สำหรับใช้ Generative AI ช่วยออกแบบ Test Case
- ประกาศนียบัตรจบหลักสูตร (Certificate of Completion) และช่องทางสอบถามวิทยากรหลังจบหลักสูตร

09 ข้อมูลผู้สอน Instructor

อ.สามิตร โขยม ผู้สอนที่ทำงานด้านการทดสอบและประกันคุณภาพซอฟต์แวร์ในโครงการจริง จุดเด่นคือการสอนให้เห็นว่าการทดสอบเริ่มตั้งแต่การอ่าน Requirement ไม่ใช่หลังเขียนโค้ดเสร็จ ใช้โจทย์ต่อเนื่องที่ใกล้งานจริงให้ผู้เรียนฝึกวิเคราะห์ เขียน Test Case และออกแบบ Scenario ด้วยตนเอง พร้อมให้ Feedback ทีละขั้นจนผู้เรียนนำวิธีคิดกลับไปใช้กับทีมได้ทันที

ราคาค่าอบรม 4,900 บาท/ท่าน · สนใจ IN-HOUSE ขอใบเสนอราคา

4,900 บาท / ท่าน (ยังไม่รวม VAT 7%)

☎ 02-570-8449

📞 @itgenius

☎ 088-807-9770

✉ contact@itgenius.co.th



เพิ่มเพื่อน
LINE