

หลักสูตรฝึกอบรมฐานสมรรถนะ สาขาซอฟต์แวร์และการประยุกต์ อาชีพนักพัฒนาระบบ ระดับ 3 (Programmer)

วัตถุประสงค์ของหลักสูตร

1. ได้รับความรู้ ความเข้าใจเกี่ยวกับสัญลักษณ์ของลำดับงาน สัญลักษณ์ของแบบร่างหน้าจอ การจัดลำดับผังงาน การตีความหมายของลำดับงานเพื่อเขียนผังงาน และสัญลักษณ์ของหน้าจอ ตามวัตถุประสงค์
2. มีความรู้ ความเข้าใจในการวิเคราะห์จำนวน Functionality การตรวจสอบเงื่อนไขการทดสอบ การเขียน Test plan ระบุ Timeline คน resource และการวิเคราะห์ผลทดสอบเพื่อ go live หรือ no go live ได้ตามวัตถุประสงค์
3. นำความรู้ที่ได้รับจากการอบรมด้านเทคโนโลยีดิจิทัล ไปใช้ในการทำงานในอนาคตได้อย่างมีประสิทธิภาพ

คุณสมบัติผู้เข้ารับการประเมิน

1. นักศึกษาที่กำลังศึกษาในระดับประกาศนียบัตรวิชาชีพชั้นสูง (ปวส.) ถึงระดับอุดมศึกษา ในสถาบันการศึกษาภาครัฐ และเอกชน ในสาขาที่เกี่ยวข้อง หรือบุคคลทั่วไปที่มีประสบการณ์ทำงานด้านการพัฒนาโปรแกรม หรือที่เกี่ยวข้องไม่น้อยกว่า 2 ปี
2. มีบัตรประจำตัวประชาชน กรณีที่เป็นนักศึกษา จะต้องมียบัตรประจำตัวนักศึกษา และมีสัญชาติไทย
3. มีความรู้พื้นฐานในการใช้งานโปรแกรมที่เกี่ยวข้องกับหลักสูตร
4. สิทธิในการอบรม และประเมินมาตรฐานอาชีพได้ 1 อาชีพ/คน

หน่วยสมรรถนะที่ใช้ในอบรมและการประเมิน

1. ดำเนินการพัฒนาโปรแกรม
2. ทดสอบโปรแกรมแบบ Unit test

จำนวนผู้เข้ารับการประเมิน

บุคคลทั่วไป หรือนักศึกษา จำนวนไม่น้อยกว่า 40 คน/รุ่น

ระยะเวลาการอบรม พร้อมสอบประเมิน

ฝึกอบรม จำนวน 2 วัน (12 ชั่วโมง)

สอบประเมิน จำนวน 1 วัน

- ภาคทฤษฎี 45 นาที
- ภาคปฏิบัติ 3 ชั่วโมง 30 นาที

เกณฑ์การผ่านการฝึกอบรม

1. ผู้เข้าอบรม ต้องเข้ารับการฝึกอบรมไม่น้อยกว่าร้อยละ 80 ของระยะเวลาฝึกอบรมตลอดหลักสูตร
2. ผู้เข้าอบรม ต้องทดสอบประเมินความรู้ภาคทฤษฎีด้วยแบบประเมินผลก่อนการฝึกอบรม (Pre-Test)
3. ผู้เข้าอบรม ต้องทดสอบประเมินความรู้ภาคทฤษฎีด้วยแบบประเมินผลหลังการฝึกอบรม (Post-Test)

เกณฑ์ผ่านไม่น้อยกว่าร้อยละ 70

เกณฑ์การผ่านการประเมิน

- "ผ่านการประเมิน"

หมายถึง ผู้เข้ารับการประเมิน มีผลคะแนน ภาคทฤษฎีไม่น้อยกว่าร้อยละ 70 และภาคปฏิบัติไม่น้อยกว่าร้อยละ 80

- "ไม่ผ่านการประเมิน"

หมายถึง ผู้เข้ารับการประเมิน มีผลคะแนน ภาคทฤษฎีน้อยกว่าร้อยละ 70 และภาคปฏิบัติไม่น้อยกว่าร้อยละ 80

กำหนดการจัดกิจกรรมอบรม

วันที่ 1

09.00 – 12.00 น. บรรยายความรู้ หัวข้อ “การดำเนินการพัฒนาโปรแกรม”

- การอ่านแบบร่างหน้าจอ (GUI) ลำดับงาน (Work Flow)
- การทบทวนและแก้ไขแบบร่างหน้าจอ (GUI) และ/หรือ ลำดับงาน (Work Flow)
- ฝึกปฏิบัติ (Workshop)

13.00 – 16.00 น. บรรยายความรู้ หัวข้อ “การดำเนินการพัฒนาโปรแกรม” (ต่อ)

- การเขียนผังงาน (Flow Chart) ได้ตามที่กำหนด
- อ่านผังงาน
- การเขียนโปรแกรมตามผังงาน
- ฝึกปฏิบัติ (Workshop)

วันที่ 2

09.00 – 12.00 น. บรรยายความรู้ หัวข้อ “ทดสอบโปรแกรมแบบ Unit test”

- การวิเคราะห์จำนวน Functionality
- การตรวจสอบเงื่อนไขการทดสอบ
- ฝึกปฏิบัติ (Workshop)

13.00 – 16.00 น. บรรยายความรู้ หัวข้อ “ทดสอบโปรแกรมแบบ Unit test” (ต่อ)

- การเขียน Test plan ระบุ timeline, คน, resource
- การจัดทำรายงาน Unit test
- ฝึกปฏิบัติ (Workshop)

กำหนดการจัดกิจกรรมสอบประเมิน

วันที่ 1

09.00 – 10.00 น. แนะนำ และอธิบายกระบวนการประเมิน พร้อมแนวทางการสอบประเมิน

10.00 – 11.00 น. สอบภาคทฤษฎี

11.00 – 12.00 น. พักเที่ยง

12.00 – 15.30 น. สอบภาคปฏิบัติ

อุปกรณ์ที่ต้องใช้สำหรับอบรมและประเมิน อาชีพนักพัฒนาระบบ ระดับ 3 (Programmer)

ลำดับ	อุปกรณ์	จำนวน	หมายเหตุ
1	คอมพิวเตอร์	1 เครื่อง	<p>เครื่องคอมพิวเตอร์ 1 เครื่องต่อผู้เข้ารับการประเมิน 1 คน</p> <ul style="list-style-type: none"> • มีหน่วยความจำไม่น้อยกว่า 4 GB • ติดตั้งระบบปฏิบัติการไม่ต่ำกว่า Windows 7 64-bit หรือ Window เวอร์ชัน ใหม่กว่า
2	ซอฟต์แวร์ที่ใช้ในการประเมิน สำหรับสร้างฐานข้อมูล(Database) และภาษาที่ใช้เขียนโค้ด (Coding)	เลือกอย่างใด อย่างหนึ่ง	<ol style="list-style-type: none"> 1. โปรแกรม Text Editor เช่น EditPlus, Notepad ++, Sublime Text 3, NetBeans เป็นต้น 2. โปรแกรมจำลอง Web Server เช่น XAMPP, WAMP เป็นต้น โดยที่โปรแกรมจำลอง Web Server นั้น คือ โปรแกรมที่รวมเอาโปรแกรมที่จำเป็นในการพัฒนา เช่น Apache, MySQL, phpMyAdmin และอื่น ๆ เพื่อให้สามารถพัฒนาโปรแกรมได้สะดวกยิ่งขึ้น 3. โปรแกรม Web Browser เช่น Google Chrome หรือ Mozilla Firefox เป็นต้น 4. ภาษาที่ใช้เขียนโค้ด (Coding) เช่น PHP, Visual Studio Code, Java, C#, Python เป็นต้น 5. โปรแกรมสร้างฐานข้อมูล และสร้างหน้าฟอร์มติดต่อผู้ใช้งาน Microsoft Access <p>หมายเหตุ นักศึกษายังสามารถติดตั้งเพิ่มเติมก่อนสอบได้ หากต้องการพัฒนาโปรแกรมด้วย Tools อื่น ๆ เช่น ภาษาสคริปต์ อื่น ๆ เช่น ASP, JavaScript เป็นต้น</p>

****มหาวิทยาลัยฯ เลือกซอฟต์แวร์อย่างใด อย่างหนึ่ง ในการอบรมภาคปฏิบัติ และการสอบภาคปฏิบัติ****

แผนโครงสร้างหลักสูตร

หน่วยสมรรถนะ Unit of Competence : ดำเนินการพัฒนาโปรแกรม

ชื่อหัวข้อวิชา (Content Title)	เกณฑ์ในการปฏิบัติงาน (Performance Criteria)	รายละเอียดเนื้อหาวิชา (Content)
1. การอ่านแบบร่างหน้าจอ (GUI) ลำดับงาน (Work Flow) ได้	<ol style="list-style-type: none"> 1. ความสามารถในการอ่านลำดับงาน (Work Flow) 2. ความรู้เกี่ยวกับสัญลักษณ์ของลำดับงาน (Work Flow) 3. ความรู้เกี่ยวกับสัญลักษณ์ของแบบร่างหน้าจอ (GUI) 	<ol style="list-style-type: none"> 1. ความหมายของการออกแบบหน้าจอ (GUI) 2. การออกแบบส่วนติดต่อกับผู้ใช้งานแบบกราฟิก 3. รูปแบบการออกแบบส่วนติดต่อกับผู้ใช้แบบกราฟิก
2. การทบทวนและแก้ไขแบบร่างหน้าจอ (GUI) และ/หรือ ลำดับงาน (Work Flow) ได้	<ol style="list-style-type: none"> 1. ความสามารถในการแก้ไขแบบร่างหน้าจอ (GUI) 2. ความสามารถในการแก้ไขลำดับงาน (Work Flow) 	<ol style="list-style-type: none"> 1. ความหมายของผังงาน ประโยชน์ของผังงาน ประเภทของผังงาน หลักการเขียนผังงาน ความหมายของสัญลักษณ์ในผังงาน 2. ความหมายของอัลกอริทึม ประโยชน์ของอัลกอริทึม คุณลักษณะการเขียนอัลกอริทึม เครื่องมือช่วยในการเขียนอัลกอริทึม
3. การเขียนผังงาน (Flow Chart) ได้ตามที่กำหนด	<ol style="list-style-type: none"> 1. ความสามารถในการเขียนผังงาน (Flow Chart) 	<ol style="list-style-type: none"> 1. ความหมายการโปรแกรมโครงสร้างของผังงาน รูปแบบการโปรแกรมโครงสร้างของผังงาน 2. หลักเกณฑ์ในการวิเคราะห์งาน 3. ความหมายและหลักเกณฑ์ในเรื่องหลักทั่วไปในการเขียนผังงานระบบ
4. อ่านผังงานได้	<ol style="list-style-type: none"> 1. ความสามารถในการอ่านผังงาน (Flow Chart) 2. ความรู้เกี่ยวกับการตีความหมายของลำดับงาน (Work Flow) เพื่อเขียนผังงาน (Flow Chart) 	<ol style="list-style-type: none"> 1. แผนภาพกระแสข้อมูล (Data Flow Diagram : DFD) 2. สัญลักษณ์ต่างๆ ที่ใช้ในการเขียนแผนภาพการไหลของข้อมูล 3. ตรรกศาสตร์เบื้องต้น (Logical) 4. ลำดับการประมวลผลทางคณิตศาสตร์
5. การเขียนโปรแกรมตามผังงานได้	<ol style="list-style-type: none"> 1. ความสามารถในการนำผังงาน (Flow Chart) ไปเขียนโปรแกรม 	<ol style="list-style-type: none"> 1. ความหมายของการเขียนโปรแกรมเชิงวัตถุ 2. แนวคิดแบบการเขียนโปรแกรมเชิงวัตถุ

หน่วยสมรรถนะ Unit of Competence : ทดสอบโปรแกรมแบบ Unit test

ชื่อหัวข้อวิชา (Content Title)	เกณฑ์ในการปฏิบัติงาน (Performance Criteria)	รายละเอียดเนื้อหาวิชา (Content)
1. การวิเคราะห์จำนวน Functionality ได้	<ol style="list-style-type: none"> 1. ความสามารถในการวิเคราะห์จำนวน Functionality ได้ 2. ความรู้ในการการวิเคราะห์จำนวน Functionality ได้ 	<ol style="list-style-type: none"> 1. การใช้งานภาษา Structured Query Language (SQL) 2. ประเภทของคำสั่งภาษา SQL 3. ประเภทของข้อมูลที่ใช้ในฐานข้อมูล 4. ลักษณะการใช้งานและตัวอย่างของภาษา SQL 5. ความหมายและตัวอย่างของฟังก์ชัน
2. การตรวจสอบเงื่อนไขการทดสอบได้	<ol style="list-style-type: none"> 1. ความสามารถในการตรวจสอบเงื่อนไขการทดสอบได้ 2. ความรู้เกี่ยวกับการตรวจสอบเงื่อนไขการทดสอบได้ 	<ol style="list-style-type: none"> 1. ความหมายของการทดสอบซอฟต์แวร์ 2. วัตถุประสงค์หลักของการทดสอบ 3. ระดับการทดสอบซอฟต์แวร์
3. การเขียน Test plan ระบุ timeline, คน, resource	<ol style="list-style-type: none"> 1. ความสามารถในการเขียน Test plan ระบุ timeline, คน resource 2. ความรู้เกี่ยวกับการเขียน Test plan ระบุ timeline, คน, resource 	<ol style="list-style-type: none"> 1. แนวทางการทดสอบซอฟต์แวร์ 2. วิธีการทดสอบซอฟต์แวร์
4. การจัดทำรายงาน Unit test	<ol style="list-style-type: none"> 1. ความสามารถในการวิเคราะห์ผลทดสอบเพื่อ go live หรือ no go live ได้ 2. ความรู้เกี่ยวกับการวิเคราะห์ผลทดสอบเพื่อ go live หรือ no go live ได้ 	<ol style="list-style-type: none"> 1. การทดสอบโมดูลหรือทดสอบหน่วย (Module Testing หรือ Unit testing) 2. การทดสอบภาพรวม (Integration Testing) 3. การทดสอบฟังก์ชัน (Functional testing) 4. การทดสอบประสิทธิภาพ (Performance Testing) 5. การทดสอบการยอมรับ (Acceptance Testing)