

TEST PLAN

ระบบยื่นคำร้องของคณะวิทยาศาสตร์

Faculty of Science Application Submission System

1 TEST PLAN IDENTIFIER G06-SASS1.0

2 REFERENCE

- Project proposal โครงการพัฒนาระบบยื่นคำร้องคณะวิทยาศาสตร์ฯ

3 INTRODUCTION

เอกสารนี้เป็นแผนการทดสอบที่อธิบายวิธีการทดสอบระบบยื่นคำร้องที่พัฒนาขึ้นสำหรับคณะวิทยาศาสตร์ วัตถุประสงค์หลักของแผนการทดสอบนี้คือเพื่อให้ระบบทำงานอย่างมีประสิทธิภาพและเชื่อถือได้โดยสอดคล้องกับความต้องการของผู้ใช้และผู้ดูแลระบบ

การทดสอบสำหรับโครงการนี้จะดำเนินการที่ระดับ Automated End2End Testing โดยระยะเวลาโครงการโดยประมาณคือหนึ่งสัปดาห์ ซึ่งต้องใช้ความพยายามและการประสานงานที่ดีระหว่างทีมพัฒนาและทีมทดสอบ ความล่าช้าในกระบวนการพัฒนาหรือการติดตั้งซอฟต์แวร์อาจมีผลกระทบต่อระหว่างการทดสอบ และการทดสอบการยอมรับจะต้องเสร็จสิ้นภายในระยะเวลาที่คาดเอาไว้

4 TEST ITEMS

- Frontend ของระบบยื่นถอนคำร้อง version 1.0

5 SOFTWARE RISK ISSUES

- FrontEnd อาจไม่สามารถเชื่อมต่อกับ Backend ได้
- Dependency และ External library อาจไม่ครบหรือเวอร์ชันที่ใช้อาจไม่เหมาะกับระบบ
- ฐานข้อมูลอาจมีข้อมูลสูญหายหรือมีข้อผิดพลาด
- ระบบไม่สามารถเรียกใช้ REST API ได้
- หลังจากการรีสตาร์ทระบบแล้ว ต้องตรวจสอบข้อมูล เนื่องจากอาจมีการเปลี่ยนแปลงภายในฐานข้อมูล

6 FEATURES TO BE TESTED

- เขียนคำร้อง
- การบันทึกคำร้องลง database

7 FEATURES NOT TO BE TESTED

- Login ระบบ
- Logout ระบบ
- ยกเลิกคำร้องลงทะเบียนล่าช้า
- แนบเอกสารเพิ่มเติม
- สืบค้นสถานะคำร้องโดยนักศึกษา
- ให้ความเห็นชอบโดยอาจารย์ที่ปรึกษา
- อนุมัติคำร้องโดยคนบดี
- อนุมัติคำร้องโดยอาจารย์ผู้สอน
- อนุมัติคำร้องโดยอธิการบดี
- ตรวจสอบความถูกต้องคำร้อง
- ชำระเงินค่าลงทะเบียน
- พิมพ์บันทึกคำร้องส่งคณะ
- พิมพ์บันทึกคำร้องส่งสำนักทะเบียนฯ
- แจ้งเตือนการเปลี่ยนสถานะคำร้อง
- Activate & Deactivate ผู้ใช้งานระบบ
- ค้นหารายละเอียดประวัติการใช้ระบบ
- เพิ่มข้อมูลรายละเอียดบุคลากรคณะฯ

- ลบข้อมูลผู้ใช้สัณฐานบุคคลากรคณะฯ
- สืบค้นคำร้องโดยผู้ดูแลระบบ
- ยกเลิกคำร้องโดยผู้ดูแลระบบ

8 APPROACH

8.1) Testing Levels

ในการทดสอบระบบยืนยันคำร้องของคณะวิทยาศาสตร์นั้นเราจะทดสอบโดยใช้การทดสอบแบบ Automated End2End Test ในการทดสอบ ซึ่งทางกลุ่มของเราวางแผนที่จะแบ่งหน้าที่ให้แต่ละสมาชิกเข้าไปทำการทดสอบแต่ละส่วนแยกกัน แต่ด้วยปัญหาด้านเวลานั้น ทำให้จากการแบ่งหน้าที่ลดลงมาเหลือให้สมาชิกบางส่วนไปทำการทดสอบ แล้วให้สมาชิกส่วนที่เหลือทำ Test Plan และดูแลการทดสอบร่วมกัน

ในการทำ Automated End2End Test นั้น จะทำการให้สมาชิกกลุ่มหนึ่งไปทำการเขียนโค้ดทดสอบขึ้นมา จากนั้นจะให้สมาชิกส่วนที่เหลือช่วยตรวจทานโค้ดที่สมาชิกเขียนขึ้นมา จากนั้นจึงเริ่มทำการทดสอบ

8.2) Configuration Management

ในการทดสอบโปรแกรมจะมีการกำหนดเวอร์ชันของโปรแกรมต่างๆเพื่อที่จะควบคุมสภาพแวดล้อมของอุปกรณ์ที่ใช้ในการทดสอบ

โดยโปรแกรมที่ใช้จะมีดังนี้:

- nodejs@21.7.1
- firefox@121.0
- npm@10.5.0
- selenium webdriver@12.1.9
- protractor@7.0.0
- apache maven@3.9.5
- java@17

8.3) Test Tools

อุปกรณ์ที่ใช้ในการสร้างและดำเนินการทดสอบมีดังนี้:

- protractor
- nodejs
- selenium webdriver

มีการใช้ Library ภายนอกดังนี้:

- mocha
- chai

8.4) Meetings

เมื่อได้รับงาน สมาชิกในกลุ่มประชุมหนึ่งครั้งเพื่อปรึกษาเพื่อทำ Test Strategies และ Test Plan หลังจากนั้นในหนึ่งสัปดาห์ก่อนส่งงานจะทำการประชุมกันอีกครั้งเพื่อตรวจสอบและจัดทำรายงานผลการทดสอบ

8.5) Measures and Metrics

กลุ่มของเรานั้นจะใช้มาตรวัดผลของการทดสอบดังต่อไปนี้ในการทดสอบ Automated End2End Test

1. จำนวนของ Defects
2. สาเหตุของ Defects

9 ITEM PASS/FAIL CRITERIA

PASS CRITERIA

- Test cases ทั้งหมดผ่าน
- Defect ที่พบไม่ส่งผลให้ระบบดำเนินการต่อไปไม่ได้
- Code Coverage ต้องได้มากกว่าหรือเท่ากับร้อยละ 80

FAIL CRITERIA

- Test cases บางกรณีไม่ผ่าน
- Defect ที่พบส่งผลให้ระบบล้มเหลว
- Code Coverage ได้น้อยกว่าร้อยละ 80

10 SUSPENSION CRITERIA AND RESUMPTION REQUIREMENTS

- พบ defect ที่ทำให้ระบบไม่สามารถดำเนินการต่อได้
 - ทำการทดสอบต่อก็คือเมื่อทำการแก้ไข defect แล้ว
- เวลาที่ระบบตอบสนองต่อการทดสอบนานเกินไป
 - ตรวจสอบการเชื่อมต่อระหว่างส่วนต่าง ๆ ระบบ ทำการรีสตาร์ทระบบและดำเนินการทดสอบต่อ
- ระบบไม่สามารถเชื่อมต่อกับเน็ตเวิร์คได้

- ตรวจสอบอินเทอร์เน็ตของเครื่องที่ใช้ในการทำทดสอบ
- และตรวจสอบการตั้งค่าสภาพแวดล้อมของตัวระบบ ทำการรีเซ็ตระบบแล้วดำเนินการทดสอบต่อ

11 TEST DELIVERABLES

- รายงานแผนการทดสอบ
- รายงาน defect หรือ bug ที่พบ

12 REMAINING TEST TASKS

เนื่องจากมีการทดสอบเพียงครั้งเดียว จึงไม่มีงานอื่น ๆ ที่ต้องทำเพิ่มอีก

13 ENVIRONMENTAL NEEDS

- ข้อมูลที่นำมาใช้ในการทดสอบเช่น ข้อมูลการกรอกแบบฟอร์ม จะถูกกำหนดไว้ล่วงหน้าก่อนการทดสอบ
- มีการใช้โปรแกรมจากภายนอกมาใช้ในการทดสอบโปรแกรม ดังนี้:
 - Protractor
- มีการใช้ framework กับ library มาใช้ในการทดสอบโปรแกรม ดังนี้:
 - Mocha
 - Chai

14 STAFFING AND TRAINING NEEDS

- Tester ต้องทำการศึกษา project proposal ของระบบยื่นคำร้องของคณะวิทยาศาสตร์
- Tester ต้องทำการศึกษาโค้ดของระบบยื่นคำร้องของคณะวิทยาศาสตร์
- Tester ต้องทำการศึกษาเครื่องมือ Protractor เพื่อใช้ในการทำ End2End Testing

15 RESPONSIBILITIES

	Test Leader	Testers
ศึกษา Project Proposal	X	X
หาหรือการออกแบบ Test Strategies และ Test Plan	X	X
จัดทำ Test Strategies และ Test Plan	X	

ตรวจสอบ Test Strategies และ Test Plan		X
เตรียมสภาพแวดล้อมก่อนทำการทดสอบ		X
เตรียมระบบที่ใช้ในการทดสอบ		X
ทำการทดสอบแบบ End2End		X
รายงานผลการทดสอบ		X
รายงาน defect และ bug ที่พบ		X
ตรวจสอบผลการรายงาน	X	X
จัดทำรายงานการทดสอบ	X	
จัดทำรายงาน defect และ bug ที่พบ	X	

16 SCHEDULE

1. ทีมทดสอบศึกษา Project Proposal ของระบบ
2. Testers นำระบบมาติดตั้งบนเครื่องแล้วทดลองรัน
3. ทีมทดสอบปรึกษารื้อเพื่อออกแบบ Test Strategies และ Test Plan
4. Test Leader จัดทำ Test Strategies และ Test Plan
5. Testers ในทีมตรวจสอบ Test Strategies และ Test Plan
6. Testers ทำการติดตั้งระบบและเครื่องมือที่ใช้ในการทดสอบ
7. Testers ทำการทดสอบแบบ End2End
8. Testers รายงานผลการทดสอบ defect และ bug ที่พบ
9. ทีมทดสอบตรวจสอบผลรายงาน
10. Test Leader จัดทำรายงานการทดสอบ defect และ bug ที่พบ

17 PLANNING RISKS AND CONTINGENCIES

- ทีมขาดประสบการณ์
- เวลาในการทำมีค่อนข้างจำกัดและไม่เพียงพอ

18 APPROVALS

นักศึกษาคณะวิทยาศาสตร์	
------------------------	--