

1. หากต้องการ Design Test-case มี Technique อะไรบ้างที่สามารถช่วยให้การ Design Test-case เพื่อทดสอบระบบได้ครอบคลุม

Answer :: Test Design Technique จะมีหลักๆอยู่ 3 เทคนิค คือ

1. Black Box Testing เป็นการทดสอบระบบโดยมองจากมุมมองผู้ใช้ โดยไม่จำเป็นต้องทราบว่ารระบบ ออกแบบมาอย่างไร หรือ code ข้างในเป็นอย่างไร แต่อาจจะต้องทราบเงื่อนไขของระบบที่จะทำการทดสอบ ข้อมูลที่จะนำเข้าสู่ระบบ การส่งออกของข้อมูลจากระบบ และผลลัพธ์ที่คาดหวัง โดยในส่วนของ Black Box Testing ก็จะมีเทคนิคที่ใช้กันบ่อยๆ ดังนี้

- **Equivalence Partitioning (EP)** เทคนิคนี้ จะเป็นการแบ่งแยกกลุ่มข้อมูลออกเป็นช่วงๆ และนำตัวแทนข้อมูลในแต่ละช่วง นำมาทดสอบ
- **Boundary Value Analysis (BVA)** เทคนิคนี้ จะใช้ขอบเขตของข้อมูล 3 ส่วน คือ ค่าที่เท่ากันกับตัวมันเอง ค่าที่มากกว่าตัวมันเอง และค่าที่น้อยกว่าตัวมันเอง ส่วนใหญ่จะใช้กับกลุ่มข้อมูลที่เป็นตัวเลขหรือเวลา
- **State Transition Testing** เทคนิคนี้ เป็นการทดสอบการทำงานของระบบในการเปลี่ยนแปลงจากสถานะหนึ่งไปยังอีกสถานะหนึ่ง หรือ Flow ของระบบ เช่น สถานะเริ่มต้น จนถึงสถานะสุดท้าย
- **Use Cases** เทคนิคนี้ จะเป็นการออกแบบตามการใช้งานของผู้ใช้งานที่อาจเกิดขึ้นได้จริงในระบบ โดยจะออกแบบความเป็นไปได้ทั้งหมดที่สามารถเกิดขึ้นได้ ภายใต้เงื่อนไขของระบบ
- **Decision Table Test Design** เทคนิคนี้ จะเป็นการทดสอบที่ใช้กับระบบที่มีเงื่อนไขหลายอย่าง โดยเงื่อนไขและค่าที่ใช้สำหรับ Input , Output หรือ การสร้างสถานการณ์การทดสอบที่เป็นไปได้ จะแสดงในตาราง

2. White Box Testing เป็นการทดสอบระบบที่เน้นการทำงานของภายในของระบบ ซึ่งโดยส่วนใหญ่ Developer จะใช้ในการทำ Unit test โดยจะออกแบบ Test case ให้ครอบคลุมทั้ง เงื่อนไข และ input ต่างๆ โดยในส่วนของ Black Box Testing ก็จะมีเทคนิคที่ใช้กันบ่อยๆ เช่น

- **Statement coverage** เทคนิคนี้ จะทดสอบระบบเฉพาะเคสที่ค่าเป็นจริง (true) และ 1 test case ต่อ 1 เงื่อนไข
- **Decision coverage** เทคนิคนี้ จะทดสอบให้ครอบคลุมทั้งเคสที่เป็น true และ false
- **Condition coverage** จะทดสอบให้ครอบคลุมทั้งเคสที่เป็น true และ false และผลลัพธ์ที่เป็นไปได้ทั้งหมดอย่างน้อยหนึ่งครั้ง โดยแยกเป็นเทสเคส
- **Decision and condition coverage** เทคนิคนี้ เป็นวิธีที่ละเอียดและครอบคลุมกับทุกๆ เงื่อนไข รวมไปถึงถึงเคสที่เป็น true และ false โดยการนำเอาเทคนิค Decision coverage รวม Condition coverage

- **Branch coverage** เทคนิคนี้ใช้ทดสอบ Path Testing โดยจะทดสอบเส้นทางเท่าที่จำเป็นเพื่อให้แน่ใจว่าทุกๆ ทางเลือก (branch) เช่น if-then-else หรือ case statement ได้ถูกทำการทดสอบแล้วอย่างน้อยหนึ่งครั้ง

3. Experience-based test เป็นเทคนิคในการทดสอบ โดยที่ใช้ความรู้ ทักษะ และประสบการณ์ ของผู้มีส่วนร่วมในการทดสอบ ในการสร้าง condition หรือ สร้าง test case ใช้ประสบการณ์ ในการออกแบบทดสอบ

- **Error guessing** การคาดเดาจากประสบการณ์ ว่าตรงจุดไหนที่ให้เกิดข้อผิดพลาดของระบบอยู่บ่อยๆ

- **Exploratory testing** เป็นวิธีในการทดสอบ ที่มีเวลาน้อย ไม่ได้มีการวางแผนไว้ล่วงหน้า ทดสอบจากประสบการณ์

- **Checklist-based testing** เป็นการทดสอบตามรายการตรวจสอบที่วางแผนไว้ล่วงหน้า

Design Test-case จากโจทย์ต่อไปอย่างน้อย 5 Case พร้อมระบุ Technique ที่ใช้นั้น ๆ

- [โจทย์] : ผู้ใช้ต้องการโอน Point จากบัญชีตัวเอง ไปยังบัญชีปลายทาง โดยเงื่อนไขคือ
 - ขั้นต่ำในการโอน Point คือ 100 / การทำรายการ
 - สูงสุดในการโอน Point คือ 3,000 / การทำรายการ
 - หากโอน < ขั้นต่ำ ระบบคิดค่า Fee 8 Point โดยบวกเพิ่มจากค่าที่กรอก
 - ต้องกรอก Passcode 4 หลัก ให้ถูกต้องจึงทำการสำเร็จ
 - บัญชีปลายทางต้องถูกต้อง จึงจะสามารถกรอก Passcode ได้

Test Case ID	Function	Sub Function	Positive/ Negative	Data Test	Test Step	Expected Details	Technique
1	การโอน Point	จำนวน Point ขั้นต่ำในการโอน 100 Point	P	Point ในการโอน จำนวน 100 Point	1. เข้าสู่ระบบการโอน Point 2. ระบุจำนวน Point ที่ต้องการโอน	ระบบสามารถตรวจสอบ Point ที่จะทำการโอนได้ถูกต้อง (ถ้าการโอนต่ำกว่า 100 Point จะต้องเสียค่าธรรมเนียม) และสามารถทำขั้นตอนการโอน Point ต่อไปได้	Boundary Value Analysis (BVA)
2	การโอน Point	จำนวน Point ขั้นต่ำในการโอนเกิน 100 Point	P	Point ในการโอน จำนวน 101 Point	1. เข้าสู่ระบบการโอน Point 2. ระบุจำนวน Point ที่ต้องการโอน	ระบบสามารถตรวจสอบ Point ที่จะทำการโอนได้ถูกต้อง (ถ้าการโอนต่ำกว่า 100 Point จะต้องเสียค่าธรรมเนียม) และสามารถทำขั้นตอนการโอน Point ต่อไปได้	Boundary Value Analysis (BVA)
3	การโอน Point	จำนวน Point ขั้นต่ำในการโอนไม่เกิน 100 Point	N	Point ในการโอน จำนวน 99 Point	1. เข้าสู่ระบบการโอน Point 2. ระบุจำนวน Point ที่ต้องการโอน	ระบบสามารถตรวจสอบ Point ที่จะทำการโอนได้ถูกต้อง (ถ้าการโอนต่ำกว่า 100 Point จะต้องเสียค่าธรรมเนียม) และแสดงข้อความ "มีค่าธรรมเนียมในการโอนขั้นต่ำ 8 Point"	Boundary Value Analysis (BVA)
4	การโอน Point	จำนวน Point ในการโอนไม่พอกับ Point ที่มี	N	Point ในการโอน จำนวน 100 Point แต่มี Point เหลือ จำนวน 50	1. เข้าสู่ระบบการโอน Point 2. ระบุจำนวน Point ที่ต้องการโอน	ระบบสามารถตรวจสอบ Point ที่จะทำการโอนได้ถูกต้อง และแสดงข้อความ "คุณมีจำนวน Point ไม่พอที่จะทำการโอน"	State Transition Testing
5	การโอน Point	จำนวน Point สูงสุดในการโอน 3000 Point	P	Point ในการโอน จำนวน 3000 Point	1. เข้าสู่ระบบการโอน Point 2. ระบุจำนวน Point ที่ต้องการโอน	ระบบสามารถตรวจสอบ Point ที่จะทำการโอนได้ถูกต้อง (ต้องไม่เกิน 3000 Point) และสามารถทำขั้นตอนการโอน Point ต่อไปได้	Boundary Value Analysis (BVA)
6	การโอน Point	จำนวน Point สูงสุดในการโอนไม่เกิน 3000 Point	P	Point ในการโอน จำนวน 2999 Point	1. เข้าสู่ระบบการโอน Point 2. ระบุจำนวน Point ที่ต้องการโอน	ระบบสามารถตรวจสอบ Point ที่จะทำการโอนได้ถูกต้อง (ต้องไม่เกิน 3000 Point) และสามารถทำขั้นตอนการโอน Point ต่อไปได้	Boundary Value Analysis (BVA)
7	การโอน Point	จำนวน Point สูงสุดในการโอนเกิน 3000 Point	N	Point ในการโอน จำนวน 4000 Point	1. เข้าสู่ระบบการโอน Point 2. ระบุจำนวน Point ที่ต้องการโอน	ระบบสามารถตรวจสอบ Point ที่จะทำการโอนได้ถูกต้อง (ต้องไม่เกิน 3000 Point) และแสดงข้อความ "จำนวน Point สูงสุดในการโอน ไม่เกิน 3000 Point"	Boundary Value Analysis (BVA)
8	การโอน Point	การกรอก บัญชีปลายทางถูกต้อง	P	1. Point ในการโอน จำนวน 300 Point 2. บัญชีผู้รับโอน 12345678	1. เข้าสู่ระบบการโอน Point 2. ระบุจำนวน Point ที่ต้องการโอน 3. กรอกเลขบัญชีปลายทาง	ระบบสามารถทำขั้นตอนต่อไปได้ (ขั้นตอนการกรอก Passcode)	State Transition Testing
9	การโอน Point	การกรอก บัญชีปลายทางไม่ถูกต้อง	N	1. Point ในการโอน จำนวน 300 Point 2. บัญชีผู้รับโอน 123Wnr456	1. เข้าสู่ระบบการโอน Point 2. ระบุจำนวน Point ที่ต้องการโอน 3. กรอกเลขบัญชีปลายทาง	ระบบขึ้นข้อความแจ้งเตือน "กรุณากรอกตัวเลขบัญชีผู้รับโอน 8 หลักให้ถูกต้อง"	State Transition Testing
10	การโอน Point	การกรอก Passcode 4 หลักถูกต้อง	P	Point ในการโอน จำนวน 300 Point	1. เข้าสู่ระบบการโอน Point 2. ระบุจำนวน Point ที่ต้องการโอน 3. กรอกเลขบัญชีปลายทาง	ระบบทำการโอน Point สำเร็จ	State Transition Testing
11	การโอน Point	การกรอก Passcode 4 หลัก ไม่ถูกต้อง	N	Point ในการโอน จำนวน 300 Point	1. เข้าสู่ระบบการโอน Point 2. ระบุจำนวน Point ที่ต้องการโอน 3. กรอกเลขบัญชีปลายทาง	ระบบขึ้นข้อความแจ้งเตือน "กรุณากรอก Passcode 4 หลักให้ถูกต้อง"	State Transition Testing

3. หากทีมต้องการทดสอบ Feature ในข้อ 2 จะต้องมีการ Test Plan อย่างไร?

แผนการทดสอบระบบการโอน Point (Test Plan)

โดย Wasin Winyarat

บทนำ (Introduction)

จุดประสงค์หลักของเอกสารแผนการทดสอบนี้ ได้ทำการอธิบายรายละเอียดเกี่ยวกับวิธีการทดสอบระบบการโอน Point และใช้เป็นแนวทางในการทำการทดสอบตามแผนงานที่วางไว้

วัตถุประสงค์ (Objective)

เป้าหมายของการทดสอบระบบการโอน Point คือ ตรวจสอบการทำงานของระบบการโอน Point ที่กำหนดไว้ในขอบเขตของการทดสอบ

ขอบเขตของการทดสอบ (Scope)

- การ Login เข้าสู่ระบบของผู้ใช้งาน
- การตรวจสอบจำนวน Point คงเหลือ
- ขั้นตอนและข้อจำกัดของการโอน Point ไปยังบัญชีปลายทาง

วิธีการทดสอบ (Test Approach)

- การทดสอบแบบ Manual Test และใช้วิธีการทดสอบแบบ Black box testing โดยใช้เทคนิค
- Boundary Value Analysis (BVA)
 - State Transition Testing

สิ่งที่นำส่งหลังจากดำเนินการทดสอบตามแผนงาน (Test Deliverables)

- รายงานผลการทดสอบ
- รายงานข้อผิดพลาดของระบบที่พบ

สภาพแวดล้อมในการทดสอบระบบ (Testing Environment)

- Test Environment : Function test
- Browser : Chrome
- Device : Desktop
- Testing Tools : MS Word , MS Excel , Trello

เกณฑ์ในการดำเนินการทดสอบ (Entry Criteria)

- แผนการทดสอบได้รับการอนุมัติ
- ข้อมูลสำหรับการทดสอบพร้อมใช้งาน

เกณฑ์ในการประเมินผลการทดสอบ (Exit Criteria)

- ข้อผิดพลาดที่พบในระบบ ถูกแก้ไข และทดสอบแล้วผ่าน

ทรัพยากรบุคคลที่ต้องใช้ (Resource)

- Tester 1 คน
- Programmer support 1 คน

ระยะเวลาของการทดสอบ (Estimation and Timeline)

- ระยะเวลาที่ใช้ในการทดสอบ 4.5 วัน โดยแบ่งเป็น
 1. จัดทำ Test Planning : 1 วัน
 2. ออกแบบการทดสอบระบบ : 1 วัน
 3. ทดสอบระบบ : 1.5 วัน
 4. จัดทำรายงานสรุปผลการทดสอบ

แผนการทดสอบ		ระยะเวลาการดำเนินการ (วัน)				
		1	2	3	4	5
1	จัดทำ Test Planning					
2	ออกแบบ Test Case					
3	ทดสอบระบบ					
4	จัดทำรายงานสรุปผลการทดสอบ					

4. Software Testing มีความสำคัญอย่างไรในการพัฒนาระบบ

- ช่วยในการตรวจสอบขั้นตอนการทำงานของระบบให้มีความถูกต้องตาม requirement
- ช่วยตรวจสอบประสิทธิภาพในการทำงานของระบบ
- ช่วยในการตรวจสอบข้อผิดพลาดที่เกิดขึ้นของระบบ
- ช่วยตรวจสอบระบบในกรณีที่มีการเพิ่มเติม หรือเปลี่ยนแปลง ความต้องการของระบบ ให้สามารถใช้งานได้ปกติ
- ช่วยสร้างความพึงพอใจให้กับผู้ใช้งาน ลดปัญหาการ complain จาก user ในกรณีที่ลูกค้าพบ Bug หรือ error ต่างๆ