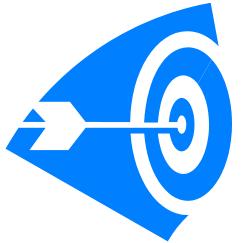


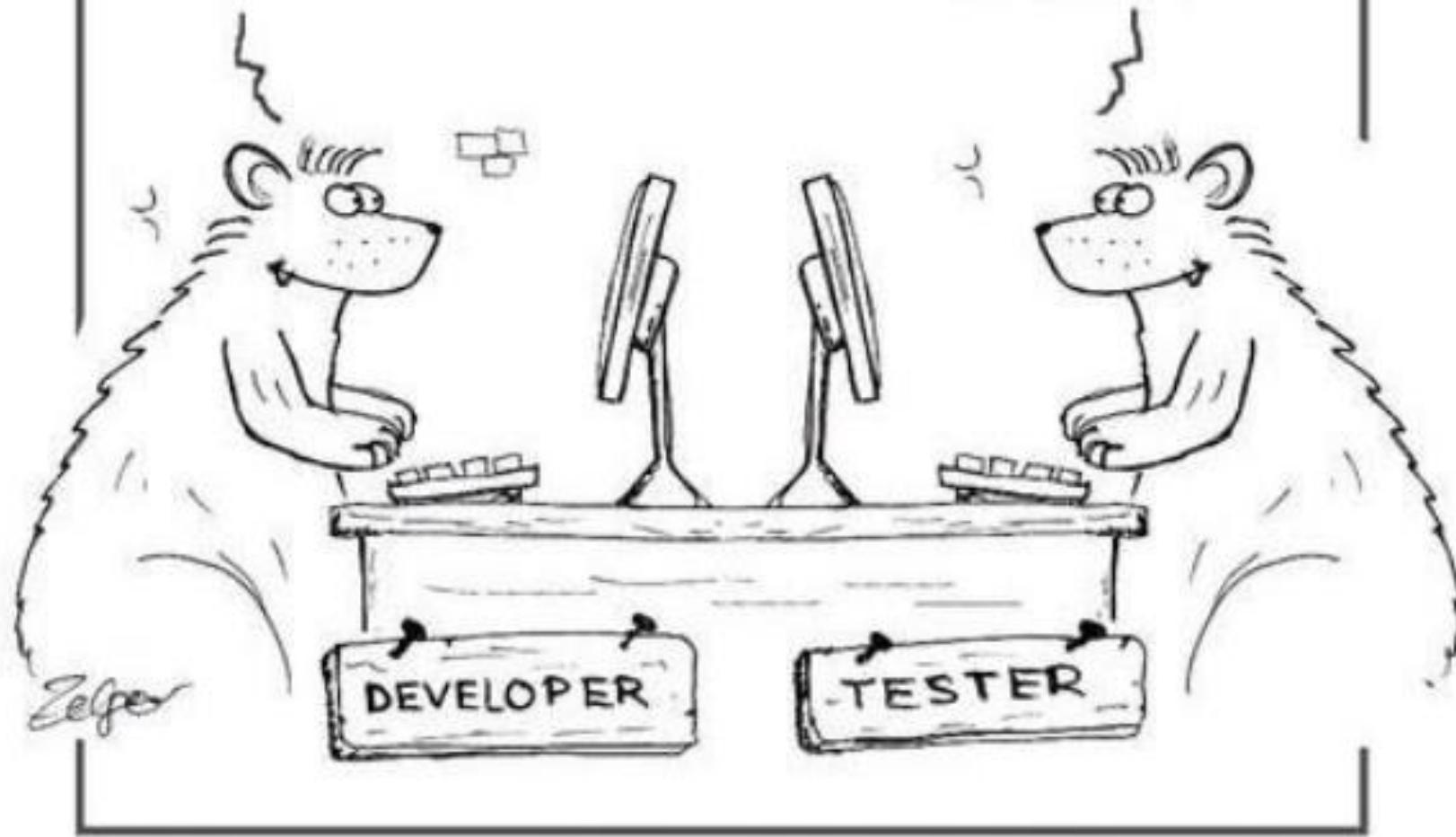
BASIC SOFTWARE TESTING

Guide for starting of tester career



"HOW CAN I
MAKE IT ?"

"HOW CAN I
BREAK IT ?"



Agenda

- What is software testing?
- WHAT, WHY, AND WHO
- Fundamentals—Concepts, Roles, and Terminology
- Test Planning and Estimation
- Test Requirements
- Test Cases/Scripts and Test Data
- Test Execution and defect capturing
- Test Measurement
- Test Automation
- Test Preparation and Process Summary
- Quality Assurance and Quality Control
- Improvement
- Class Activities

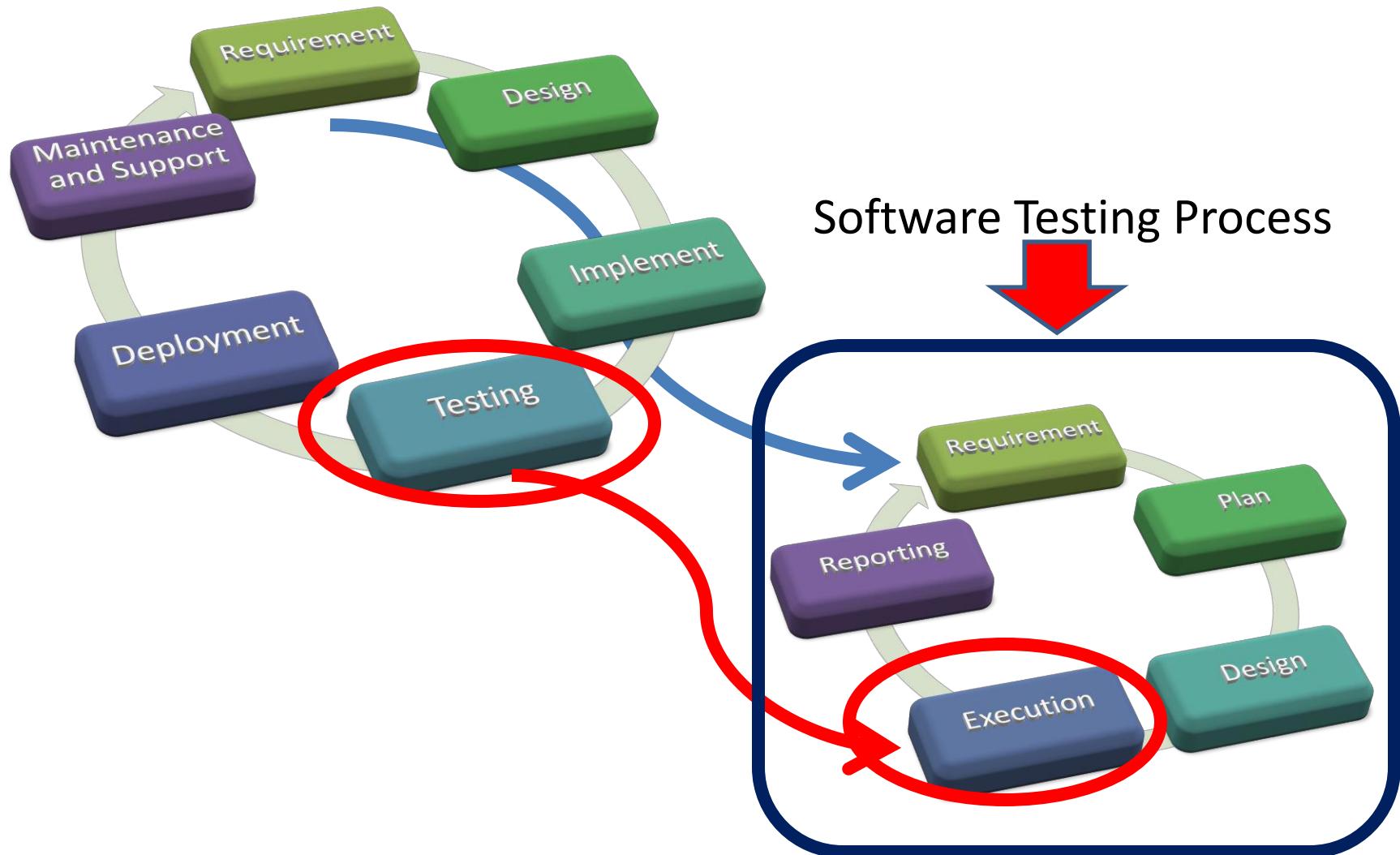


1. WHAT IS SOFTWARE TESTING?

WHAT IS SOFTWARE TESTING?

- Software testing is a process of *verifying and validating that a software application or program*
 - 1. *Meets the business and technical requirements that guided its design and development, and*
 - 2. *Works as expected.*
- Software testing also identifies important *defects, flaws, or errors in the application code that must be fixed.*

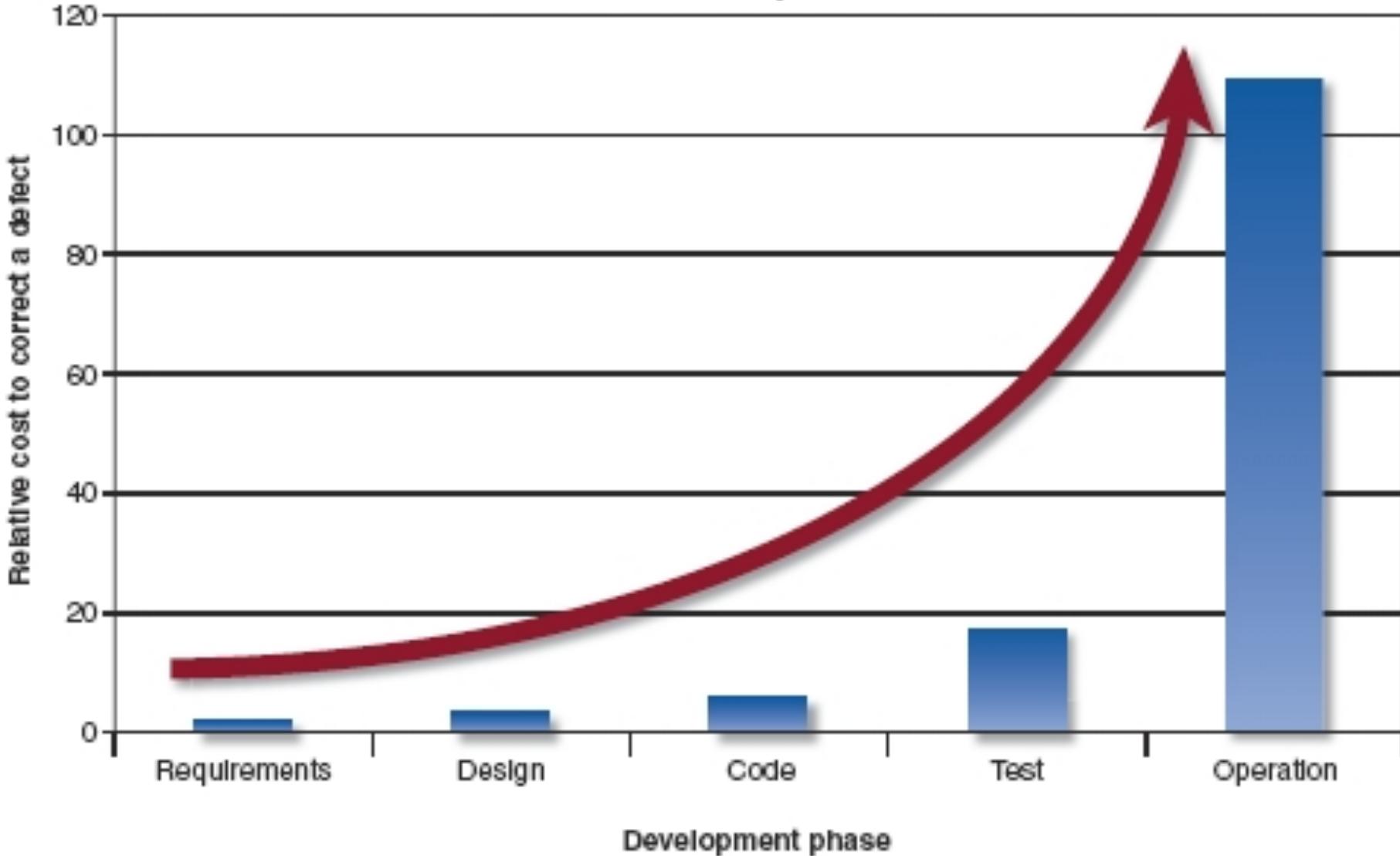
Software Development Process & Software Testing Process





2. WHAT, WHY, AND WHO

Cost of fixing a defect



Source: Control Engineering with data from Robert Grady, "Applications of Software Measurement Conference," 1999.

Why does Software have bugs?

- Most common reason is human mistakes in software design and coding.
 - Miscommunication or no communication
 - Software complexity
 - Programming errors
 - Changing requirements
 - Time pressures
 - Egotistical or overconfident people
 - Poorly documented code
 - Software development tools
 - Obsolete automation scripts
 - Lack of skilled testers

Sources of Problems

- **Requirements Definition:** Erroneous, incomplete, inconsistent requirements.
- **Design:** Fundamental design flaws in the software.
- **Implementation:** Mistakes in chip fabrication, wiring, programming faults, malicious code.
- **Support Systems:** Poor programming languages, faulty compilers and debuggers, misleading development tools.
- **Inadequate Testing of Software:** Incomplete testing, poor verification, mistakes in debugging.
- **Evolution:** Sloppy redevelopment or maintenance, introduction of new flaws in attempts to fix old flaws, incremental escalation to inordinate complexity.

Adverse Effects of Faulty Software

- **Communications:** Loss or corruption of communication media, non delivery of data.
- **Space Applications:** Lost lives, launch delays.
- **Defense and Warfare:** Misidentification of friend or foe.
- **Transportation:** Deaths, delays, sudden acceleration, inability to brake.
- **Safety-critical Applications:** Death, injuries.
- **Electric Power:** Death, injuries, power outages, long-term health hazards (radiation).
- **Money Management:** Fraud, violation of privacy, shutdown of stock exchanges and banks, negative interest rates.
- **Control of Elections:** Wrong results (intentional or non-intentional).
- **Control of Jails:** Technology-aided escape attempts and successes, accidental release of inmates, failures in software controlled locks.
- **Law Enforcement:** False arrests and imprisonments.

Why do Software Testing?

- software testing is focused on finding *defects in the final product*.
 - Gaining confidence about the level of quality
 - Providing information for decision-making
 - Preventing defects
- Software testing answers questions that development testing and code reviews can't.
 - Does it really work as expected?
 - Does it meet the users' requirements?
 - Is it what the users expect?
 - Do the users like it?
 - Is it compatible with our other systems?
 - How does it perform?
 - How does it scale when more users are added?
 - Which areas need more work?
 - Is it ready for release?



What do we Test?

- Focus on the core functionality—the parts that are critical or popular—before looking at the ‘nice to have’ features.
- Concentrate on the application’s capabilities in common usage situations before going on to unlikely situations.
- Testing provides an opportunity to validate and verify things like the assumptions that went into the requirements, the appropriateness of the systems that the application is to run on, and the manuals and documentation that accompany the application.

Some Testing Factors

- Business requirements
- Functional design requirements
- Technical design requirements
- Regulatory requirements
- Programmer code
- Systems administration standards and restrictions
- Corporate standards
- Professional or trade association best practices
- Hardware configuration
- Cultural issues and language differences

Who does Testing?

- Software testing is not a one person job. It takes a team.
- Programmers?
- Testers must be cautious, curious, critical but non-judgmental, and good communicators.

The Role of Testing

- Role of testing in software development, maintenance and operations
 - Reduce the risk of problems
 - Reduce long-term defect-related costs
 - Contribute to the quality of the software
 - Help meeting standards:
 - Contractual or legal requirements
 - Industry-specific standards

Who's involved in testing?

- Requirements Analysts – Inspections, Peer Reviews
- Developers – Code Inspection, Unit Testing
- Testers – System & Integration Testing
- Trainers – Training materials production
- Users – User Acceptance Testing
- Project Managers – Scheduling, Resourcing, Risks, Issues, Defect Status
- Everybody is responsible for quality - NASA



Tester

Why these Bugs are not solved yet??

Developer

I am working on it,
will solve ASAP....





3. FUNDAMENTALS

Testing Principles

- All tests should be traceable to customer requirements
- Tests should be planned long before testing begins
- Testing should begin “in the small” and progress toward testing “in the large.”
- Exhaustive testing is not possible
- To be most effective, testing should be conducted by an independent third party

Types of Testing

- Unit Testing
- Integration Testing
- Functional and System Testing
- Acceptance Testing
- Regression Testing
- Beta Testing

Unit Testing

Topics	Explanation
Meaning	<i>Unit testing is the testing of individual hardware or software units or groups of related Units.</i>
Test Approach	White Box Testing
Specification	Low-level design and/or code structure
Performing	Developer

Integration Testing

Topics	Explanation
Meaning	<i>Integration test is testing in which software components, hardware components, or both are combined and tested to evaluate the interaction between them.</i>
Test Approach	Black- and white-box testing
Specification	Low- and high-level design
Performing	Developer

Functional and System Testing

Topics	Explanation
Meaning	<p>Functional testing involves ensuring that the functionality specified in the requirement specification works.</p> <p>System testing involves putting the new program in many different environments to ensure the program works in typical customer environments with various versions and types of operating systems and/or applications.</p>
Test Approach	Black-box testing
Specification	high-level design, requirements specification
Performing	Tester

System Testing

- *System testing is testing conducted on a complete, integrated system to evaluate the system compliance with its specified requirements*
 - Stress testing – testing conducted to evaluate a system or component at or beyond the limits of its specification or requirement
 - Performance testing – testing conducted to evaluate the compliance of a system or component with specified performance requirements
 - Usability testing – testing conducted to evaluate the extent to which a user can learn to operate, prepare inputs for, and interpret outputs of a system or component.

Acceptance Testing

Topics	Explanation
Meaning	Acceptance testing is formal testing conducted to determine whether or not a system satisfies its acceptance criteria (the criteria the system must satisfy to be accepted by a customer) and to enable the customer to determine whether or not to accept the system.
Test Approach	Black-box testing
Specification	requirements specification
Performing	Customers/Users

Regression Testing

- The purpose of running the regression test case is to make a “spot check” to examine whether the new code works properly and has not damaged any previously-working functionality by propagating unintended side effects.

Regression Testing

Topics	Explanation
Meaning	Regression testing is selective retesting of a system or component to verify that modifications have not caused unintended effects and that the system or component still complies with its specified requirements
Test Approach	Black- and white-box testing
Specification	Any changed documentation, high-level design
Performing	Developer/Tester

Guideline choosing Regression Test

- Choose a representative sample of tests that exercise all the existing software functions;
- Choose tests that focus on the software components/functions that have been changed; and
- Choose additional test cases that focus on the software functions that are most likely to be affected by the change.

Smoke Test

- Demonstrate stability, not to find bugs with the system
- All major functionality is present and works under “normal” conditions
- Often automated
- Test cases are broad in scope
- Run before deciding to proceed with further testing

Beta Testing

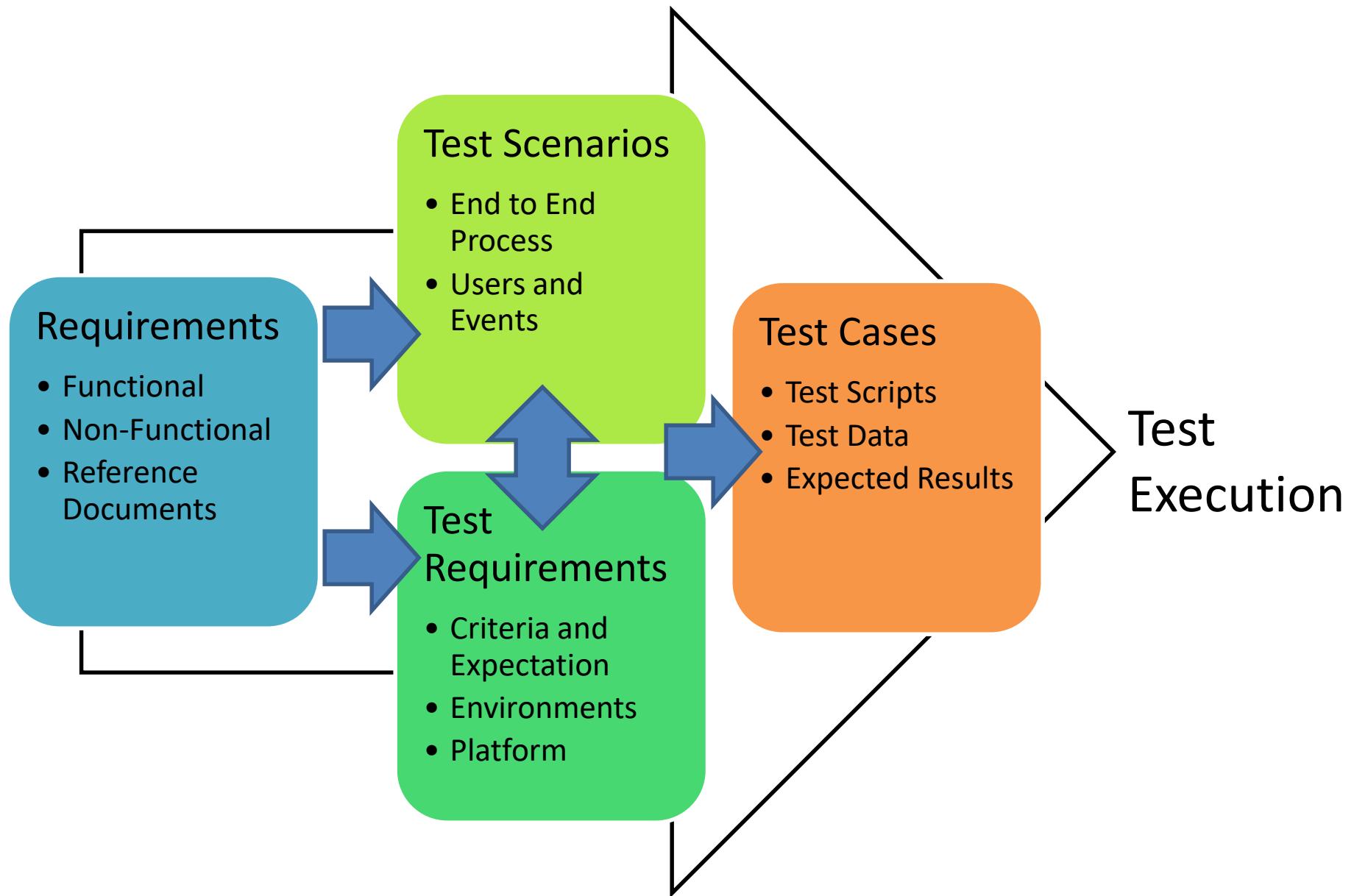
Topics	Explanation
Meaning	A test for a software product prior to commercial release. Beta testing is the last stage of testing, and normally can involve sending the product to <i>beta test sites</i> outside the company for real-world exposure or offering the product for a free trial download over the Internet.
Test Approach	Black-box testing
Specification	None
Performing	Experience Users

Advantage and Disadvantage

- Advantage
 - Identification of unexpected errors because the beta testers use the software in unexpected ways.
 - A wider population search for errors in a variety of environments (different operating systems with a variety of service releases and with a multitude of other applications running).
 - Low costs because the beta testers generally get free software but are not compensated.
- Disadvantages
 - Lack of systematic testing because each user uses the product in any manner they choose.
 - Low quality error reports because the users may not actually report errors or may report errors without enough detail.
 - Much effort is necessary to examine error reports particularly when there are many beta testers.

Summary

Testing Type	Specification	General Scope	Approach	Who generally does it?
Unit	Low-Level Design	Actual Code Structure Small unit of code no larger than a class	White Box	Programmer who wrote code
Integration	Low-Level Design High-Level Design	Multiple classes	White Box Black Box	Programmers who wrote code
Functional	High Level Design	Whole product	Black Box	Independent tester
System	Requirements Analysis	Whole product In representative Environments	Black Box	Independent tester
Acceptance	Requirements Analysis	Whole product in customer's Environment	Black Box	Customer
Beta	Ad hoc	Whole product in customer's Environment	Black box	Customer
Regression	Changed Documentation High-Level Design	Any of the above	Black Box White Box	Programmer(s) or independent testers



4. TEST REQUIREMENTS

I built it exactly to the spec.

You're using the app wrong.

It works on my machine!

Well, it doesn't work on mine.

No, it is *not* fixed yet!

Did you even read the requirements?

What exactly is a Test Requirement?

- Identifies the “WHAT” of testing
 - What needs to be tested, AND
 - What are you going to validate about it
- Includes both ***normal*** and ***error*** conditions
- Covers business rules, functionality, non-functional standards
- Do NOT have case specific data values assigned to them yet (data appears in test cases, the “How” of testing) *examples...*

Test Requirements Procedure

Business Functions

Find tasks in the functions

Find transactions for the identified task

Find the data type of each transaction

Identify field Validation

Example : Insert a record to a table

Task	Transaction	Data Type	Field
<ul style="list-style-type: none">• Open Application• Insert 1st record• Insert duplicate record• Insert without authorized• Insert damage data• Insert...	<ul style="list-style-type: none">• Record in the table	<ul style="list-style-type: none">• Any (as specified in the specification)	<ul style="list-style-type: none">• Any (as specified in the specification)

Example : Insert a record to a table

Test Requirements Identified (among others):

“Validate that you can insert an entry”

“Validate that insertion fails if entry already present”

“Validate that insertion fails if table already full”

“Validate that you can insert an entry to an empty table (initial)”

- *These are test requirements NOT tests because they do not describe the data element being inserted*
- *The data is irrelevant at this level, it will appear in the test cases used to cover these test requirements*
- *“Validate you can insert ‘John Doe’” is a test case not a test requirement*

Name Card Data Entry

Name *:

Surname* :

Title* :

Address :

Contact Phone* :

Contact Email *:

Clear

Save

Example : Add Name Card Information

Task	Transaction	Data Type	Field
<ul style="list-style-type: none">• Select Add function• Fill in correct Name card information• Fill in duplicated information• Fill in incorrect name card information• Click Save• Click Clear	<ul style="list-style-type: none">• Name card record	<ul style="list-style-type: none">• Correct Data Type is allowed to insert.• Incorrect Data Type is Fail to insert.• Telephone must be numeric number.	<p>Email is in correct format.</p> <p>Name length is at least 50 characters long.</p> <p>Surname is at least 100 characters long.</p> <p>Address is at least 1000 characters long.</p> <p>Name and Surname must be Thai and English Characters.</p>

Class Example : Add Name Card Information

Test Requirements Identified (among others):

“Validate that you can insert Name Card information”

“Validate that insertion fails if Name Card is duplicated”

“Validate that insertion fails if these required fields are empty;
name, surname, company, title, contact phone and email”

“Validate that you can insert an entry to an empty data file
(initial)”

...

- *These are test requirements NOT tests because they do not describe the data element being inserted*

Function : เพิ่มข้อมูลนามบัตร

- ตรวจสอบว่ามีฟังก์ชันเพิ่มข้อมูลนามบัตรให้เลือกใช้
- ตรวจสอบว่าสามารถใส่ข้อมูลที่ถูกต้องได้
- ตรวจสอบว่า Authorized Person สามารถใส่ข้อมูลที่ถูกต้องได้
- ตรวจสอบว่าเมื่อกด Save และข้อมูลถูกจัดเก็บเรียบร้อย
- ตรวจสอบว่าถ้าไม่ใส่ข้อมูลที่เป็น Required Fields ปุ่ม Save จะไม่ Enable ให้กด
- ตรวจสอบว่าเมื่อกด Clear ต้องล้างข้อมูลทุก Fields ที่อยู่บนจอภาพ

Why identify Test Requirements?

- It's in our workbench: Requirements-based or Function-based testing
- It's the basis for establishing the completion of testing
- Helps determine the scale of the testing effort
- Governs the types of resources you will need
- Serves to identify automation strategies you can use
- Becomes a roadmap for your testing effort
- Can be a tool for leverage and dialog with developers and business analysts
- Dev Team can sign off on them (Verification!)

Where does a TR come from?

- Business Requirements, functionality, internal logic...
 - Marketing specs, Functional specs, Technical specs
- “Interview Analysis”, Non-Functional Checklists (standards & compliance), Use Cases (from business scenarios and users), Discovery during testing, any other deliverables from previous workbenches (diagrams, modeling, flowcharts, etc.)
- Analyze Requirements:
 - What's missing?
 - Will it work?
 - Will it make sense for the user?
 - Can I test this?
 - Can I measure them?
 - How will I find out if it works?
 - Are there any contradictions?
- Talk to the developer(s)
 - Ask them about the risky parts
 - Is there anything that they might want to share with you?

Entry Criteria for Business Requirements to generate Test Requirements

- Visible ?
- Clear? (unambiguous)
- Complete?
- Consistent? (conflicting requirements must be prioritized)
- Reasonable? (achievable)
- Measurable? (quantifiable)
- Modifiable? (will it change or is it stable?)
- Traceable? (the source is known)
- Dependent requirements identified?
- Testable? (given current environment, resources, skills)

Exit Criteria for Test Requirements

- Can another tester create test cases/scenarios from these?
- Does a Test Requirement specify what is being tested and what about it we are validating? (Clear?)
- Are the Test Requirements...
 - Complete?
 - Consistent? (conflicting requirements must be prioritized)
 - Reasonable? (achievable)
 - Measurable? (quantifiable for measuring test coverage)
 - Modifiable? (will it change or is it stable?)
 - Traceable? (the source is known)
 - Testable? (given current environment, resources, skills)
- Do the test requirements cover the complete scope of the project?
- Are all the test requirements verified and signed off by the Development Team?

ATM : Practice Writing Test Requirements

Business Requirements:

- “ATM must do withdrawals”
- “Withdrawals are between 100-20,000”
- “Withdrawals are in 100 multiples”

Group Exercise!

1. Limit the scope to these 3 requirements.
2. What will you validate (test for)?
3. Are there any implied requirements that may not be written out?

Testing Withdrawals on an ATM

Test Requirements Identified (among others):

"Validate that a withdrawal option is available"

"Validate that a withdrawal of a multiple of 100, between 100-20000 can be done"

"Validate that <100 is not allowed"

"Validate that >20000 is not allowed"

"Validate that 100 multiples >20000 is not allowed"

"Validate that non-100 multiples between 100-20000 not allowed"

"Validate strange numeric amounts/combinations not allowed (all zero's, all 9's, 10.0000)"

"Validate that the withdrawal received is equal to the amount requested"

"Validate that a valid withdrawal amount must be below the account balance"

"Validate that a valid withdrawal amount must be below or equal amount in ATM"

- *These are test requirements NOT tests because they do not describe the data element being used (like 100, 40, 60, 1)*
- *The data is irrelevant at this level, it will appear in the test cases used to cover these test requirements*

Test Requirements

- ตรวจสอบว่าใส่บัตรและรหัสผ่านถูกต้องจะให้ทำการ
- ตรวจสอบว่าใส่บัตรและรหัสผ่านไม่ถูกต้องไม่ให้ทำการ
- ตรวจสอบว่าถ้าใส่บัตรถูกต้องและรหัสผ่านผิดเกิน 3 ครั้ง ^{ไม่}ให้ทำการและกินบัตร
- ตรวจสอบว่า **Network** ต้องเปิดใช้งานจะให้ทำการได้
- ตรวจสอบว่ามีปุ่ม หรือฟังก์ชันให้ลูกค้าเลือกถอนเงินปรากฏให้เลือก
- ตรวจสอบว่าลูกค้าสามารถถอนเงินได้ระหว่าง 100 ถึง 20000 เท่านั้น
- ตรวจสอบว่าลูกค้าไม่สามารถถอนเงินน้อยกว่า 100 บาทได้
- ตรวจสอบว่าลูกค้าไม่สามารถถอนเงินมากกว่า 20000 บาทได้
- ตรวจสอบว่าถอนเงินเป็นทศนิยมไม่สามารถทำได้
- ตรวจสอบว่าถอนเงินมากกว่า **Balance** ไม่ได้

Add new payment method - Credit Card

Business Requirements:

- “Allow to pay with credit card”
- “Only valid card is accepted.”
- “Must get approval before order confirmation”

Group Exercise!

1. What will you validate (test for)?
2. Are there any implied requirements that may not be written out?

E-commerce Add new payment method - Credit Card

Expected GUI-Desktop

The screenshot shows a desktop application window for adding a new payment method. At the top, a navigation bar indicates the current step: "PAYMENT" (step 2). Below the navigation, the title "PAYMENT DETAILS" is centered. On the left, a section titled "Choose Payment Method" lists three options: VISA, MasterCard, and PayPal. On the right, input fields are provided for "Cardholder's Name" (containing "William Williamson" with a green checkmark), "Card Number" (containing "4793" with red dots), "Expiry Date" (containing "May 2023" with a red X), and "CVV/CVC*" (containing "933" with a note "* 3-4 Digits Code"). A large green button at the bottom right is labeled "COMPLETE ORDER".

SHIPPING

PAYMENT

REVIEW ORDER

PAYMENT DETAILS

Choose Payment Method

VISA MasterCard PayPal

Cardholder's Name

William Williamson

Card Number

4793

Expiry Date

May 2023

CVV/CVC*

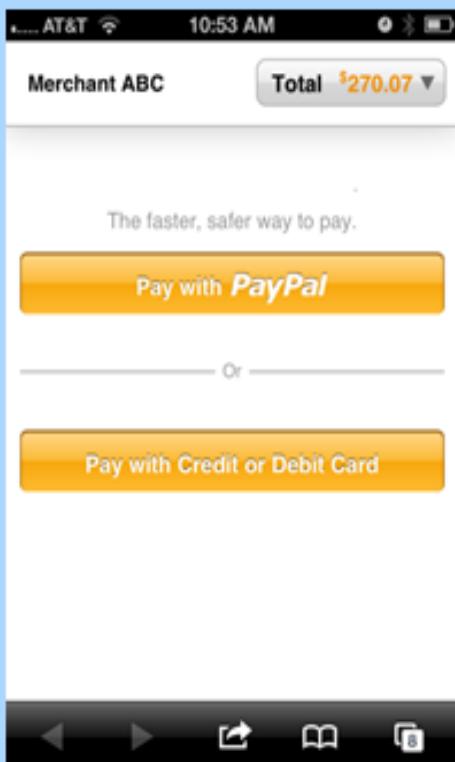
933

* 3-4 Digits Code

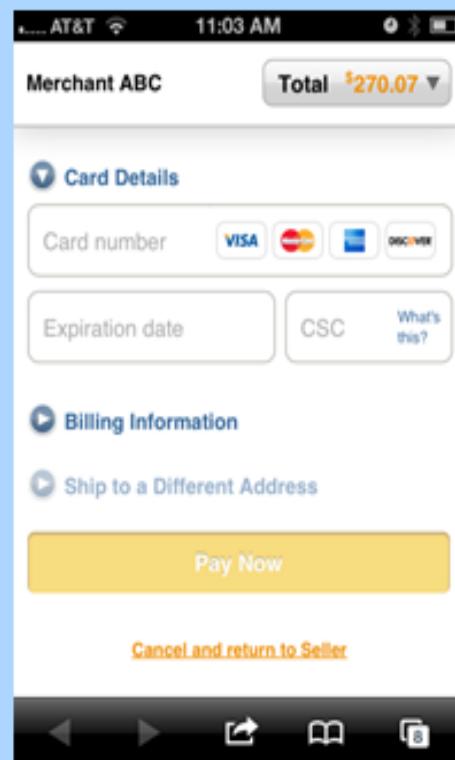
COMPLETE ORDER

Expected GUI-Mobile

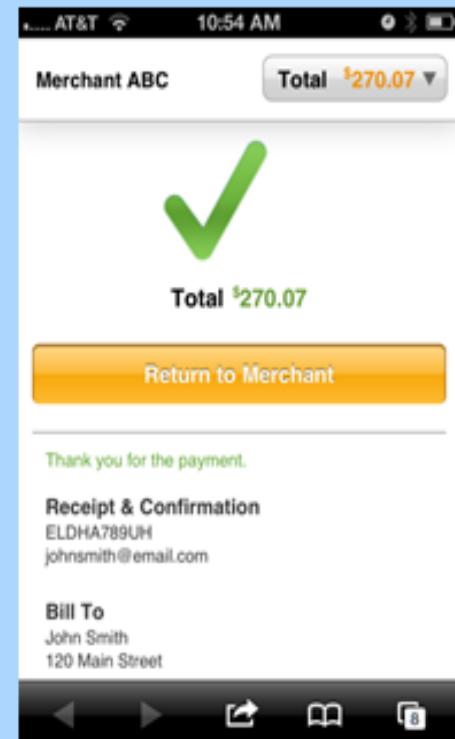
Entry



Credit Card



Done (Optional)



Testing E-commerce Credit Card Payment

Test Requirements Identified (among others):

"Validate that a payment with credit card is available"

"Validate that only allowed cards are accepted"

"Validate that card number must be 16 digits"

"Validate that 4 digits of card number matched with card type is accepted"

"Validate that only card number with valid check digit is accepted"

"Validate that expiration month is between 1 to 12 or Jan to Dec"

"Validate expiration year must be started from current year until 7 years plus from current year"

"Validate that CVV code is numeric and only 3 digits"

"Validate that confirm payment is enable after all card information is verified and corrected."

"Validate that display message pop up if customer input valid card and did not get approve code"

- *These are test requirements NOT tests because they do not describe the data element being used (like 4552 0112 0111 4576)*
- *The data is irrelevant at this level, it will appear in the test cases used to cover these test requirements*

Test Requirements

- ตรวจสอบว่ามีปุ่มให้เลือกรับชำระด้วยบัตรเครดิตให้เลือก
- ตรวจสอบว่าบัตรเครดิตในรายการที่กำหนดไว้จะให้ทำการได้
- ตรวจสอบว่าต้องใส่เลขบัตร 16 หลักที่มี **Check Digit** ถูกต้องจึงจะให้ทำการ
- ตรวจสอบว่าต้องใส่เลขบัตร 16 หลักที่มี **Check Digit** ไม่ถูกต้องจึงจะให้ทำการ
- ตรวจสอบว่าต้องใส่เลขบัตรไม่ครบ 16 หลักไม่ให้ทำการ
- ตรวจสอบว่าใส่เดือนปีที่ไม่หมดอายุจะให้ทำการได้
- ตรวจสอบว่าใส่เดือนปีที่หมดอายุจะแสดงข้อความเตือนและให้ใส่ใหม่
- ตรวจสอบว่าใส่ข้อมูล **CVV** ตรงตามที่คำนวนตามสูตรในระบบจะให้ทำการได้
- ตรวจสอบว่าใส่ **CVV** เป็นตัวเลขเท่านั้น
- ตรวจสอบว่าหลังจากใส่ข้อมูลทั้งหมดแล้วกดส่งระบบจะไปที่หน้าจออีเมลผ่าน **Third Party Software**
- ตรวจสอบว่าได้รับ **Approval Code** จาก **Third Party Software** จึงจะให้ยืนยันการทำรายการ
- ตรวจสอบว่าถ้าไม่ได้รับ **Approval Code** จาก **Third Party Software** ในช่วงเวลาที่กำหนด
- ต้องแสดงข้อความเตือน

When creating Test Requirements (“Do”)...

- Use “action” verbs & words
 - “Validate that...”
 - “Check for...”
 - “Test that...”
- Trace them back to the source
- Remember that different applications arrange in different ways
 - Think of MF, batch, C/S, web, e-commerce, GUI, etc.
 - Use “testing considerations” checklists that generally cover what kinds of things should be considered when testing your specific situation
- Make your Test Requirements document a “living document”

Test Requirements : Class Practice

1 Inventory Control:

- “Maintain product information and location of the warehouse”
- “Maintain quantity in/out”
- “Know the status of the product in the inventory”
- “Adjust the quantity and reasons”

2 New Customer Registration:

- “Customer inputs personal information”
- “Email Validation and activate the customer”
- “Customer log in”
- “Customer updates profile”

3 Sales Order (SO):

- “Issue Sales Order (SO)”
- “Print SO”
- “Only available quantity can issue SO”
- “SO with issued invoice is not allowed to change”

4 Chat Room Activities:

- “Send Voice, Text, images and videos”
- “Leave the chat room”
- “Mention persons in the room”
- “Notify the members if the new message arrived”
- “Trace back to previous chat”

Inventory Control

- ตรวจสอบว่ามีการพังก์ชัน **Maintain product information** ให้เลือก
- ตรวจสอบว่าเมื่อเลือก **Edit** และใส่รหัสสินค้าที่ถูกต้องแล้วมีในระบบจะแสดงข้อมูลสินค้าให้แก้ไขบนจอภาพ
- ตรวจสอบว่าเมื่อเลือก **Edit** และใส่รหัสสินค้าที่ไม่ถูกต้องและไม่มีในระบบจะแสดงข้อความเตือน
- ตรวจสอบว่าเมื่อเลือก **Add** และใส่รหัสสินค้าที่มีอยู่แล้วจะแสดงข้อความเตือน
- ตรวจสอบว่ามีการพังก์ชัน **Maintain warehouse location** ให้เลือก
- ตรวจสอบว่ามีการพังก์ชัน **Maintain quantity in/out** ให้เลือก
- ตรวจสอบว่าในหน้าจอข้อมูล **Product** มี **Status** ปรากฏให้เห็น
- ตรวจสอบว่าแก้ไข **Quantity** ของสินค้าได้
- ตรวจสอบว่าแก้ไข **Quantity** เป็นค่าลบไม่ได้
- ตรวจสอบว่าใส่เหตุผลที่แก้ไข **Quantity** ได้

New Customer Registration

- ตรวจสอบว่ามีเมนูให้ลูกค้าเลือกลงทะเบียนที่หน้าจอได้
- ตรวจสอบว่าลูกค้าใส่ข้อมูลลูกค้าเพื่อลงทะเบียนได้
- ตรวจสอบว่าลูกค้าใส่ **email** ที่ลงลงทะเบียนไว้แล้วจะแสดงข้อความเตือน
- ตรวจสอบว่าลูกค้าใส่ **email** ที่ไม่มีในระบบจะให้ลูกค้าลงลงทะเบียนได้
- ตรวจสอบว่า **email** จัดส่งไปที่ mail box ลูกค้าพร้อม activation link
- ตรวจสอบว่าสถานะของลูกค้าต้องไม่ activate ถ้าลูกค้ายังไม่กด link
- ตรวจสอบว่าลูกค้า login เข้ามาแก้ไข profile ได้ถ้า activate แล้วเท่านั้น
- ตรวจสอบว่าลูกค้าแก้ไขข้อมูล profile และ update ได้

Sales Order (SO)

- ตรวจสอบว่ามีฟังก์ชัน Issue Sales Order (SO) ให้เลือก
- ตรวจสอบว่ามีฟังก์ชัน Print SO ให้เลือก
- ตรวจสอบว่าสามารถใส่เลขที่ SO ที่ต้องการค้นหาได้
- ตรวจสอบว่าถ้าใส่ SO ที่ไม่มีในระบบต้องแสดงข้อความเตือน
- ตรวจสอบว่าเมื่อเลือก SO ที่ยังไม่มี invoice พังก์ชัน edit enable ให้เลือก
- ตรวจสอบว่าเมื่อเลือก SO ที่มี invoice แล้วจะให้ดูได้อย่างเดียว
- ตรวจสอบว่าเมื่อเลือก product และใส่ QTY ที่มีในสต็อกจึงจะให้ใส่ใน SO ได้
- ตรวจสอบว่าเมื่อเลือก product และใส่ QTY ที่น้อยกว่าที่มีในสต็อกจะแสดงข้อความเตือน

Chat Room Activities

- ตรวจสอบว่าสามารถเปิด Chat room กับเพื่อนได้
- ตรวจสอบว่าสามารถส่ง **text** ไปให้เพื่อนได้
- ตรวจสอบว่าเพื่อนเห็น **text** ตรงกับที่ส่งไป
- ตรวจสอบว่าสามารถส่ง **video** ไปให้เพื่อนได้
- ตรวจสอบว่าเพื่อนเห็น **video** ตรงกับที่ส่งไป
- ตรวจสอบว่าสามารถส่ง **image** ไปให้เพื่อนได้
- ตรวจสอบว่าเพื่อนเห็น **image** ตรงกับที่ส่งไป
- ตรวจสอบว่าสามารถส่ง **leave chat room** ได้
- ตรวจสอบว่าถ้าเพื่อนอ่านแล้วให้แสดง **Read** ที่หน้าจอในกล่อง **text, image, video**
- ตรวจสอบว่าถ้ามี **message** 送来 และ **Notification on** ให้แสดงข้อความเตือน
- ตรวจสอบว่าสามารถเลื่อนหน้าจอไปดู **Chat history** ได้

5. TEST SCENARIOS

Scenarios

- A scenario is any functionality.
- Test scenarios are selected among all real world and possible scenarios. They can be tested. It is also called **Test Condition or Test Possibility**.
- As a tester, you may put yourself in the end user's shoes and figure out the real-world scenarios and use cases of the Application Under Test.
 - We create personas to identify the scenarios.

Scenario Testing

- Scenario Testing is a variant of Software Testing where Scenarios are used for Testing. Scenarios help in an Easier Way of Testing of the more complicated Systems

Objectives

- Creating Test Scenarios ensures complete Test Coverage
- Test Scenarios can be approved by various stakeholders like Business Analyst, Developers, Customers to ensure the Application Under Test is thoroughly tested. It ensures that the software is working for the most common use cases.
- They serve as a quick tool to determine the testing work effort and accordingly create a proposal for the client or organize the workforce.
- They help determine the most important end-to-end transactions or the real use of the software applications.
- For studying the end-to-end functioning of the program, Test Scenario is critical.

How to create a Test Scenario?

- **Step 1:** Read the Requirement Documents like BRS, SRS, FRS, of the System Under Test (SUT). You could also refer uses cases, books, manual, etc. of the application to be tested.
- **Step 2:** For each requirement, figure out possible users actions and objectives. Determine the technical aspects of the requirement. Ascertain possible scenarios of system abuse and evaluate users with hacker's mindset.
- **Step 3:** After reading the Requirements Document and doing your due Analysis, list out different test scenarios that verify each feature of the software.
- **Step 4:** Once you have listed all possible Test Scenarios, a Traceability Matrix is created to verify that each & every requirement has a corresponding Test Scenario
- **Step 5:** The scenarios created are reviewed by your supervisor. Later, they are also reviewed by other Stakeholders in the project.

Example 1: Test Scenario for Email

- **Test Scenario 1:** Check the Login Functionality
- **Test Scenario 2:** Check that a New mail can be created
- **Test Scenario 3:** Check that an existing mail can be opened
- **Test Scenario 4:** Check that a user can send mail
- **Test Scenario 5:** Check that the information displayed in receiver user is correct

Example 1: Test Scenario for Email

- More Scenario?

Example 2: ATM Scenarios

- Check authentication Function (PIN)
 - Check Wrong PIN entered
 - Check Valid PIN entered with active account
 - Check Valid PIN entered with suspend account
- Check Withdraw Function
 - Check withdraw less than balance with active account
 - Check withdraw less than balance with Suspend account
 - Check withdraw more than balance with active account
- Check ATM machine features
 - Check ATM alert when no money
 - Check ATM alert when no connection

Practice : Online Shopping

https://www.lazada.co.th

LAZADA CO.TH

ค้นหาในลาซาด้า

CHANGE LANGUAGE

ข้อเสนอคุณภาพเด่นนั้นแอบ
ขายสินค้ากับลาซาด้า
ช่วยเหลือ
ติดตามสินค้า
ลงชื่อเข้าใช้
สมัครสมาชิก

สูบบีบ 100.-

อุปกรณ์ อิเล็กทรอนิกส์
อุปกรณ์เสริม อิเล็กทรอนิกส์
ทีวีและเครื่องใช้ไฟฟ้าในบ้าน
สุขภาพและความงาม
เด็กอ่อน และของเล่น
ชุดเครื่องแต่งกาย เก็ตติ และสัตว์เลี้ยง
บ้านและไลฟ์สไตล์
แฟชั่นผู้หญิง
แฟชั่นผู้ชาย
เครื่องประดับ
กีฬาและ การเดินทาง
ยานยนต์ และอุปกรณ์

มากรรสมหกรรม!
ลดสูงสุด 80%
ช้อปเลย 15 - 17 ธ.ค. 61

LazMall GlobalCollection เติมเงิน&เติลออนไลน์ ใส่โค้ดลดเพิ่ม

เทرنดยิด Ipad 2018 GTX 1050Ti Pipper Standard ข้อความ

Example 1: Test Scenario online

- **Test Scenario 1:** ดู Promotion ประจำวัน
- **Test Scenario 2:** ดูสินค้าที่อยู่ใน categories
- **Test Scenario 3:** ค้นหาสินค้าตามชื่อสินค้า
- **Test Scenario 4:** ตรวจสอบรายการซื้อสินค้า
- **Test Scenario 5:** ลงทะเบียนลูกค้าใหม่
- **Test Scenario 6:** เปลี่ยนภาษาไทย/อังกฤษ
- **Test Scenario 7:** ขอความช่วยเหลือ

Practice : LIVE TV

Not secure | www.workpointtv.com

workpoint ช่อง 23

((LIVE STREAM)) รายการสด ดูรายการย้อนหลัง พั้งรายการ ศิลปิน dara เพลง CONTENT LICENSING

ข่าวสารกิจกรรม ข้อมูลรีวิว ติดต่อสอบถาม

Facebook Twitter Instagram YouTube

บัดดี้ ตีบี

ทุกวันจันทร์ เวลา 20.05 น.

รายการสด

• ถ่ายทอดสด 8 2797

LIVE

workpoint

WorkpointOfficial

Workpoint กดหมายเลข 23

บัดดี้ ตีบี

workpoint ช่อง 23

NOW SHOWING

ใบค่ากองคำ 7 ((LIVE STREAM))

18:20 น. ใบค่ากองคำ 7

20:00 น. ข่าวในพระราชสำนัก

20:15 น. ใบค่ากองคำเด็ก 4

21:15 น. I Can See Your Voice Thailand

workpoint | 23

workpoint | 23

Practice : facebook

https://www.facebook.com

Ray Wannasuk

News Feed

Messenger

Watch

Shortcuts

- yanwongkons - Fre... 20+
- RayMon's Garden 20+
- Extra income
- สำหรับนักพัฒนา Ba...
- DeepQuant 3

See More...

Explore

- Events 16
- Groups
- Pages
- Fundraisers
- Friend Lists

See More...

Create Post

What's on your mind, Ray?

Photo/Video Tag Friends Feeling/Activ...

Somkiat Puisungnoen and 4 others commented on this.

Tanjai TJ Kongyuen is 😊 feeling fantastic. 4 hrs ·

สอบผ่านแล้วจ้าาาาา
I pass the ISTQB Exam with 30/40
Now I'm Certified Tester ~~~~

Stories

Add to Your Story Share a photo, video or write something

ฟ้อมดราม เสกในเรื่องมารักกัน โอลิ... 2 minutes ago

LIVE

Nonthanun Sudlapa about an hour ago

Tanjai TJ Kongyuen 12 minutes ago

See More...

Peerasan Buranasanti and 2 others

Your Page

MIDAS Software

Messages Notifications 1

Chat (124)

Practice : Buy Travel Insurance

https://travel.cigna.co.th/CIMS/?gclid=Cj0KCQiA6dLgBRDoARIsAJgoM4vYCDpH8Fz0rZ6lakQZm4m9WZUyA0Dd2fBE-ec3VNNjwIPg0FOGg6laAgbIEALw_wcB...

ເຖິງກັບເສັກນໍາ ຄ່າດາມພບບ່ອຍ ຕິດຕ່ອສັກນໍາ

Cigna

ຄົ້ນຫາ

ສອບຕາມຜັດກັບໂທ. 0-2099-3999

ແພນປະກັນ ເຄລີດ(ໄໝ)ລັບ
ຍກຮະດັບຄວາມພົດ ບ້າວແລະກົງກຽມ ອຸນຍົດແລລູກຄ້າ ເຊີກເບື້ຍປະກັນ

ເຂົ້າປະກັນທີ່ນີ້

ແພນປະກັນກ້າຍກາຣເດີນກາງຕ່າງປະເທດ

ວິນເຮັນເຕັນກາງ ເຊີກເພນປະກັນ ຂົ້ນມູສພູ່ຂອງເຕັມປະກັນ ບໍາຮາເຈັນ

ແບບແພນປະກັນ

ຮາຍເກີຍວ່າ

ຮາຍເປີ
(ໄປຈ່າກົດຈຳນວນກັ້ນ)

ຈະຫມາຍປລາຍກາງ

ເລືອກໄດ້ສູງສຸດ 10 ປະເທດ

Leave a Message

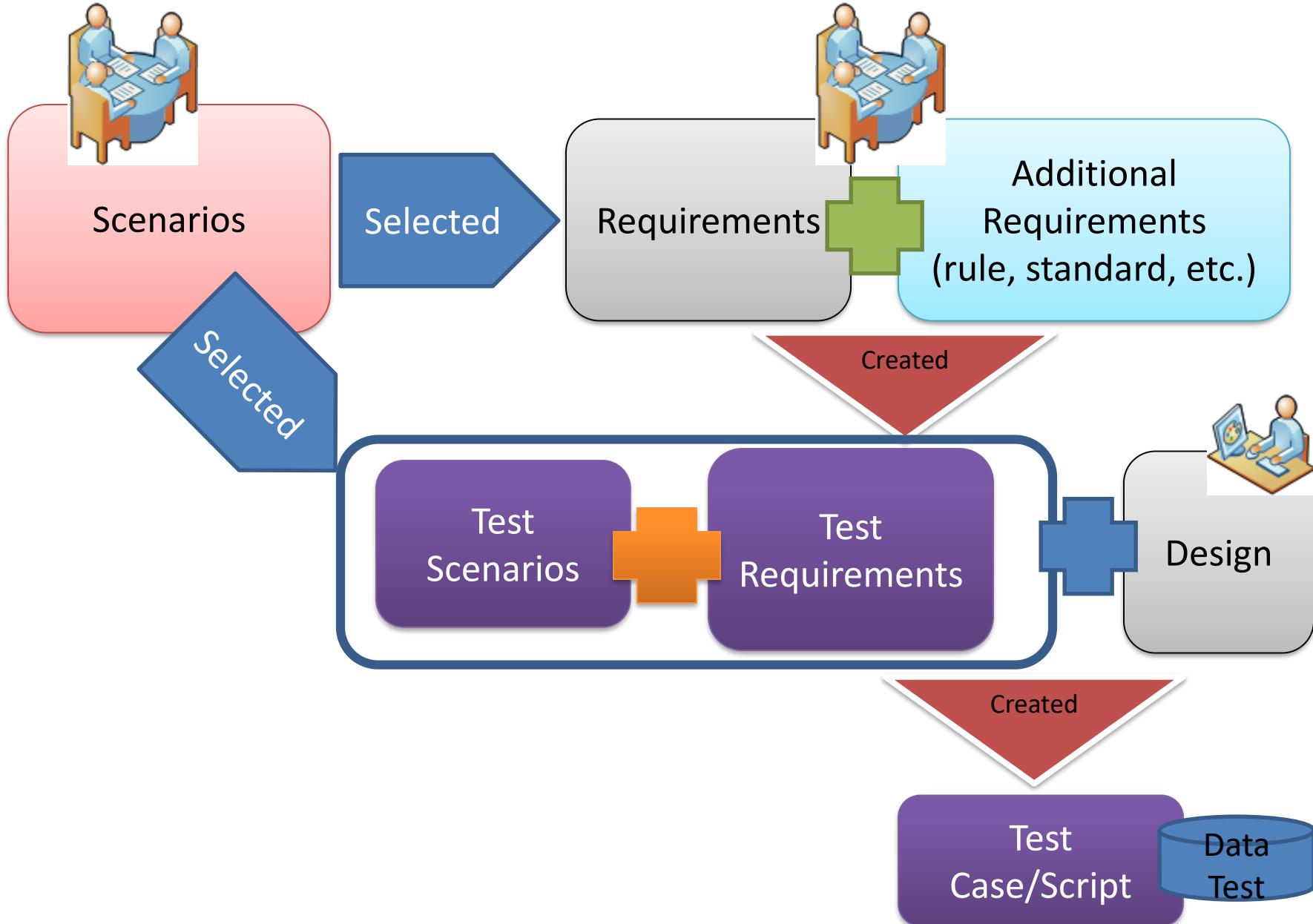
Practice : Book Hotel

The image shows the Booking.com website interface. At the top, there's a header bar with the URL 'Booking.com BV [NL] | https://www.booking.com/index.th.html?label=gen173nr-1FCAEoggI46AdIM1gEaN0BiAEBmAEmuAExYAE2AEB6AEB-AELiAIB...'. Below the header, the Booking.com logo is on the left, followed by currency 'THB' and a flag icon. To the right are buttons for 'จองห้องพักของท่าน' (Book your room), 'สมัคร' (Sign up), and 'เข้าสู่ระบบ' (Log in). The main search area has tabs for 'ที่พัก' (Accommodation), 'ตัวเครื่องบิน' (Flight), 'บริการร่วมเดินทาง' (Travel services), and 'บริการแท็กซี่ส่วนบุคคล' (Personal taxi service). The search bar contains the text 'ค้นหาข้อเสนอสำหรับทุกฤดูกาล' (Search for offers for all seasons) and 'ด้วยแต่ละวันพักตากอากาศและสนับสนุนไปจนถึงห้องชุดสุดชิป' (With each vacation day and support until the end). Below the search bar are three input fields: 'ท่านจะไปที่ไหน?' (Where will you go?), 'เช็คอิน — เช็คเอาท์' (Check-in — Check-out), and 'ผู้ใหญ่ 2 • 0 เด็ก' (Adults 2 • Children 0). A large blue 'ค้นหา' (Search) button is on the far right. The background features a scenic image of a coastal town with colorful buildings and a stone tower overlooking the sea.

Scenarios

Requirements

Functions



6. PREPARATION OF TEST DATA

Test Approach

- Black box testing (also called functional testing) is testing that ignores the internal mechanism of a system or component and focuses solely on the outputs generated in response to selected inputs and execution conditions.
- White box testing (also called structural testing and glass box testing) is testing that takes into account the internal mechanism of a system or component.

Compare two types of Testing

Black Box Testing

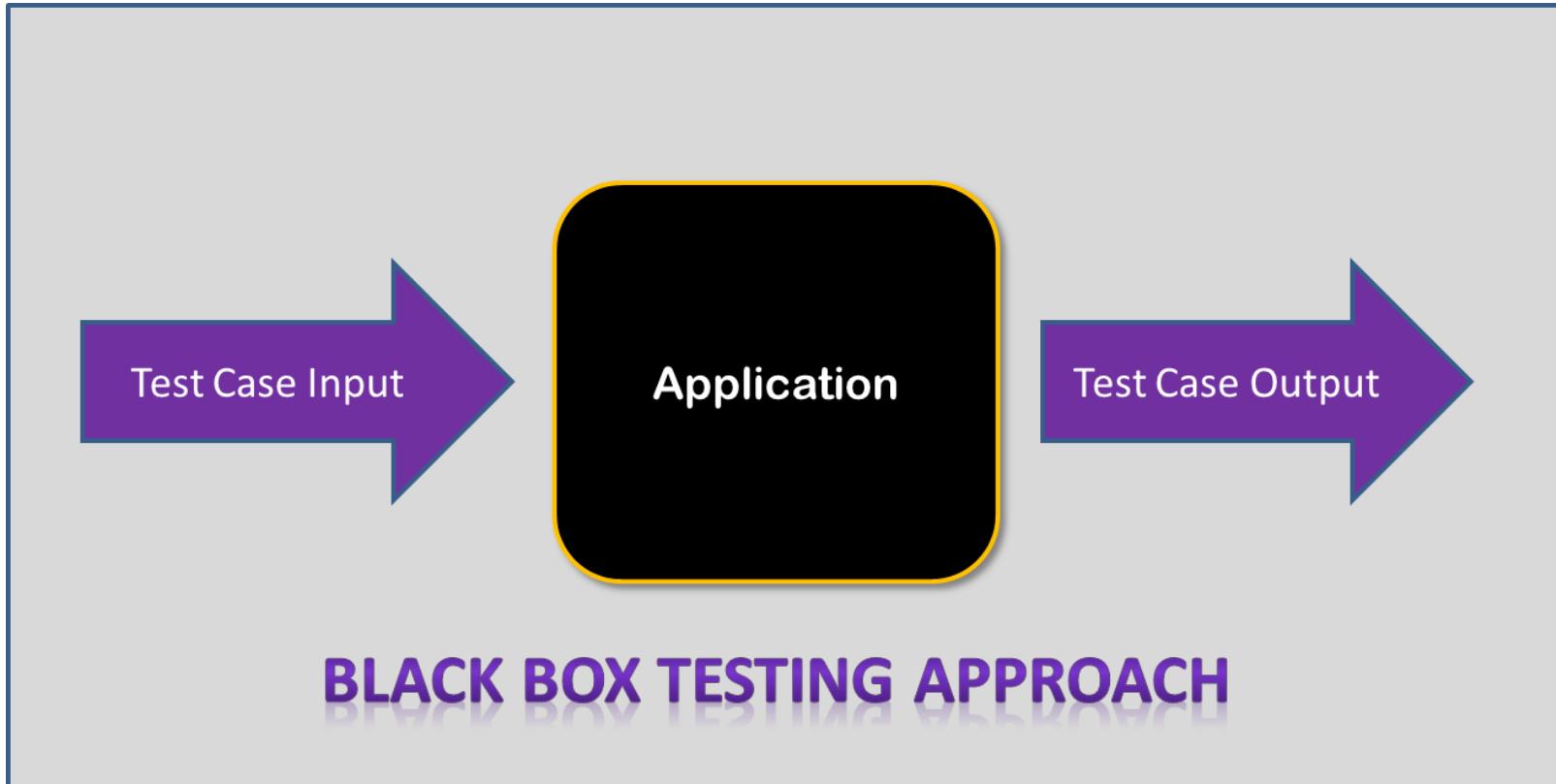
- Testing that UUT satisfies requirements
- Focus: I/O behavior
- No knowledge of the internals of the UUT
- Goal: Cover all the requirements

White-box testing

- Testing control structures
- Focus: Thoroughness (coverage)
- Knowledge of the internal structure of the UUT
- Goal: Cover all the code

➤ UUT = “Unit under test”

Black-box testing



Aims of Black Box Testing

- Find
 - incorrect or missing functionality
 - interface errors
 - errors in data structures used by interfaces
 - behavior or performance errors
 - initialization and termination errors

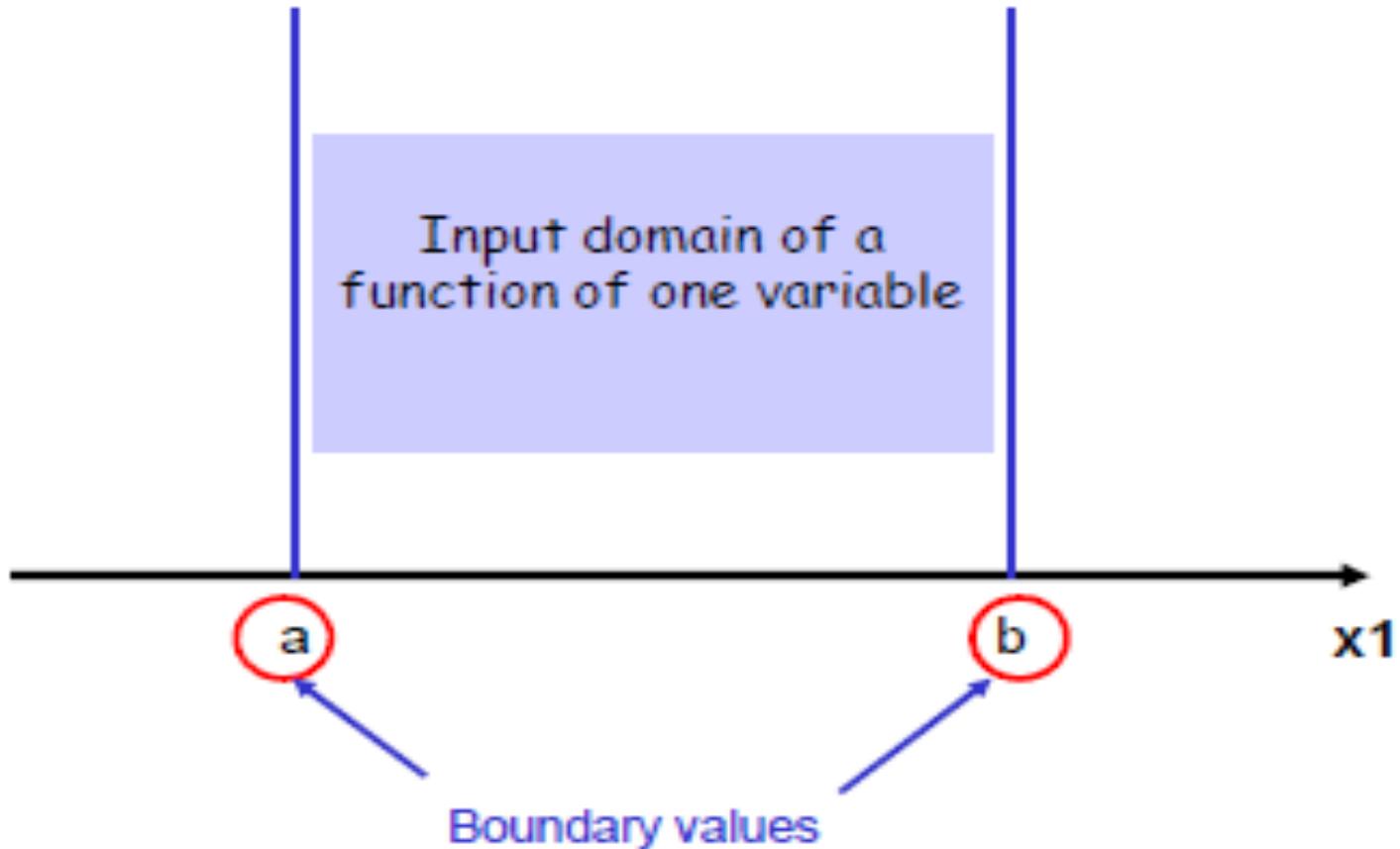


Methodologies

- Boundary testing
- Equivalence classes
- Decision tables
- State transitional diagrams



Boundary value testing



Sample : Age input field

AGE

Enter Age

*Accepts value 18 to 56

BOUNDARY VALUE ANALYSIS		
Invalid (min -1)	Valid (min, +min, -max, max)	Invalid (max +1)
17	18, 19, 55, 56	57

Minimum boundary value is 18

Maximum boundary value is 56

Valid Inputs: 18,19,55,56

Invalid Inputs: 17 and 57

Test Data 1: Enter the value 17 (18-1) = Invalid

Test Data 2: Enter the value 18 = Valid

Test Data 3: Enter the value 19 (18+1) = Valid

Test Data 4: Enter the value 55 (56-1) = Valid

Test Data 5: Enter the value 56 = Valid

Test Data 6: Enter the value 57 (56+1) = Invalid

Sample : Name input field

Name

Enter Name

*Accepts characters length (6 - 12)

BOUNDARY VALUE ANALYSIS		
Invalid (min -1)	Valid (min, +min, -max, max)	Invalid (max +1)
5 characters	6, 7, 11, 12 characters	13 characters

Minimum boundary value is 6

Maximum boundary value is 12

Valid text length is 6, 7, 11, 12

Invalid text length is 5, 13

Test Data 1: Text length of 5 (min-1) = Invalid

Test Data 2: Text length of exactly 6 (min) = Valid

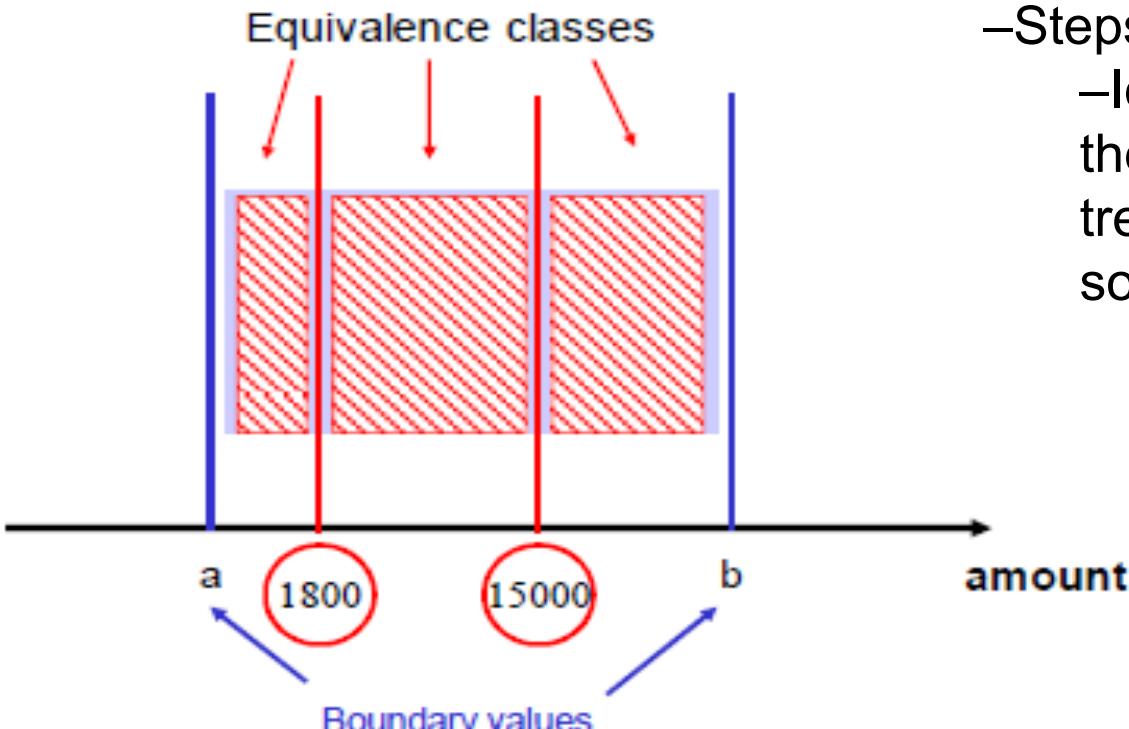
Test Data 3: Text length of 7 (min+1) = Valid

Test Data 4: Text length of 11 (max-1) = Valid

Test Data 5: Text length of exactly 12 (max) = Valid

Test Data 6: Text length of 13 (max+1) = Invalid

Equivalence class partitioning



–Steps:

- Identify equivalence classes, the input values which are treated the same by the software:

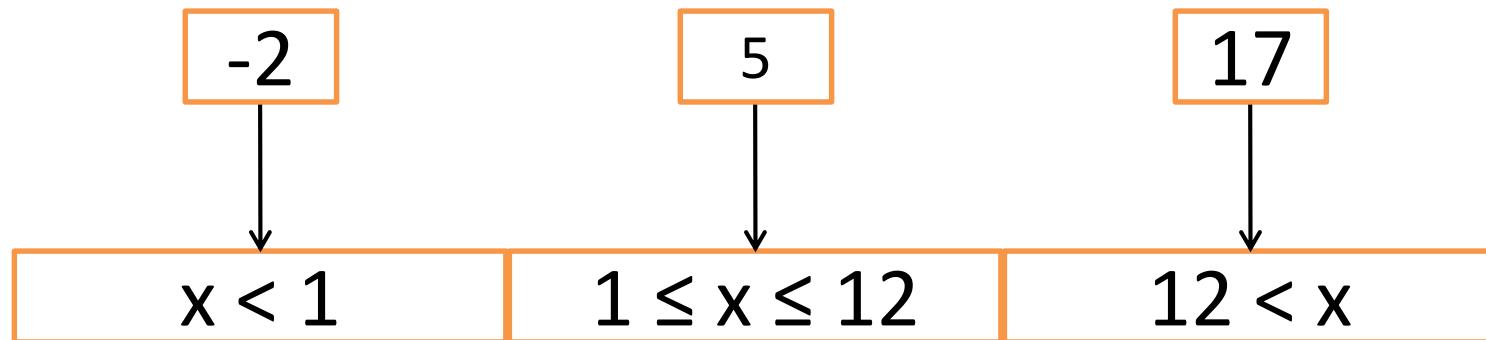
- Valid classes: legal input values;

- Invalid classes: illegal or unacceptable input values;

- Create a test case for each equivalence class.

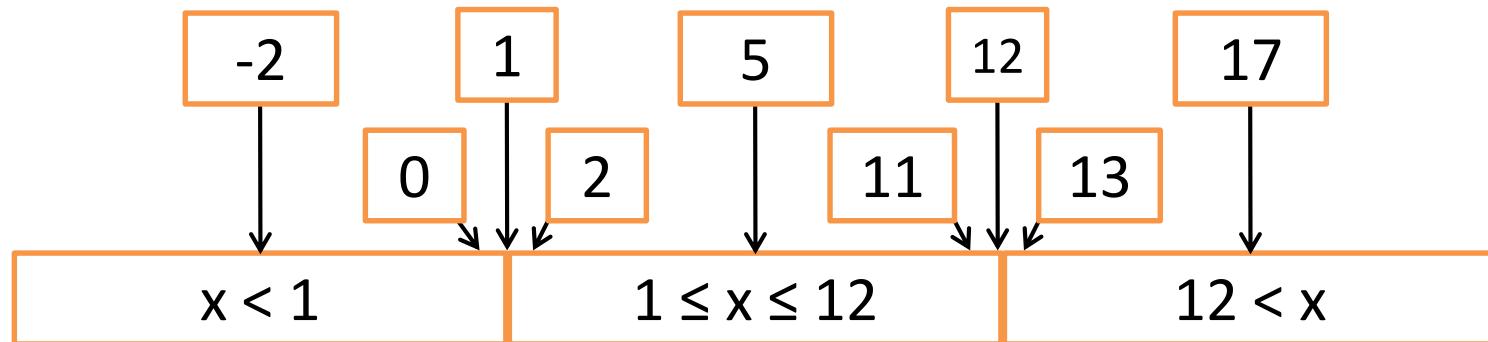
Equivalence partitioning example

- Test cases are chosen so that each partition would be tested.



Equivalence partitioning with boundary value analysis

- We use the same example as before.
- Test cases are supplemented with ***boundary values***.



Sample : Age input field

AGE

Enter Age

*Accepts value 18 to 56

BOUNDARY VALUE ANALYSIS		
Invalid (min -1)	Valid (min, +min, -max, max)	Invalid (max +1)
17	18, 19, 55, 56	57

Valid Input: 18 – 56

Invalid Input: less than or equal to 17 (≤ 17), greater than or equal to 57 (≥ 57)

Valid Class: 18 – 56 = Pick any one input test data from 18 – 56

Invalid Class 1: ≤ 17 = Pick any one input test data less than or equal to 17

Invalid Class 2: ≥ 57 = Pick any one input test data greater than or equal to 57

Sample : Mobile input field

MOBILE NUMBER

Enter Mobile No.

*Must be 10 digits

EQUIVALENCE PARTITIONING		
Invalid	Valid	Invalid
987654321	9876543210	98765432109

Valid input: 10 digits

Invalid Input: 9 digits, 11 digits

Valid Class: Enter 10 digit mobile number = 9876543210

Invalid Class Enter mobile number which has less than 10 digits
= 987654321

Invalid Class Enter mobile number which has more than 11
digits = 98765432109

Decision table

- composed of rows and columns, separated into quadrants:

Conditions	Condition Alternatives
Actions	Action Entries

This test technique is appropriate for functionalities which has logical relationships between inputs (if-else logic). We take conditions as inputs and actions as outputs.



Decision table

Condition Stubs	Conditions/ Courses of Action	Rules					
		1	2	3	4	5	6
	Employee type	S	H	S	H	S	H
	Hours worked	<40	<40	40	40	>40	>40
Action Stubs	Pay base salary	X		X		X	
	Calculate hourly wage		X		X		X
	Calculate overtime						X
	Produce Absence Report		X				

Decision Table Practice

Business Rules

- Company X sells merchandise to wholesale and retail outlets. Wholesale customers receive a two percent discount on all orders.
- The company also encourages both wholesale and retail customers to pay cash on delivery by offering a two percent discount for this method of payment.
- Another two percent discount is given on orders of 50 or more units.

Decision Table

Condition

Order Less than 50 Units	Y	Y	Y	Y	N	N	N	N
Cash on Delivery	Y	Y	N	N	Y	Y	N	N
Whole Sales Customer	Y	N	Y	N	Y	N	Y	N

Action

Discount 0%				X				
Discount 2%		X	X					X
Discount 4%	X					X	X	
Discount 6%				X				

Decision Table Practice

Business Rules

- login page validation.
 - Allow user to login only when both the ‘User ID’ and ‘Password’ are entered correct.
 - The Actions performed are Displaying home page and Displaying an error message that User ID or Password is wrong.

Decision Table

Condition

Valid User ID	Y	Y	N	N				
Valid Password	Y	N	Y	N				

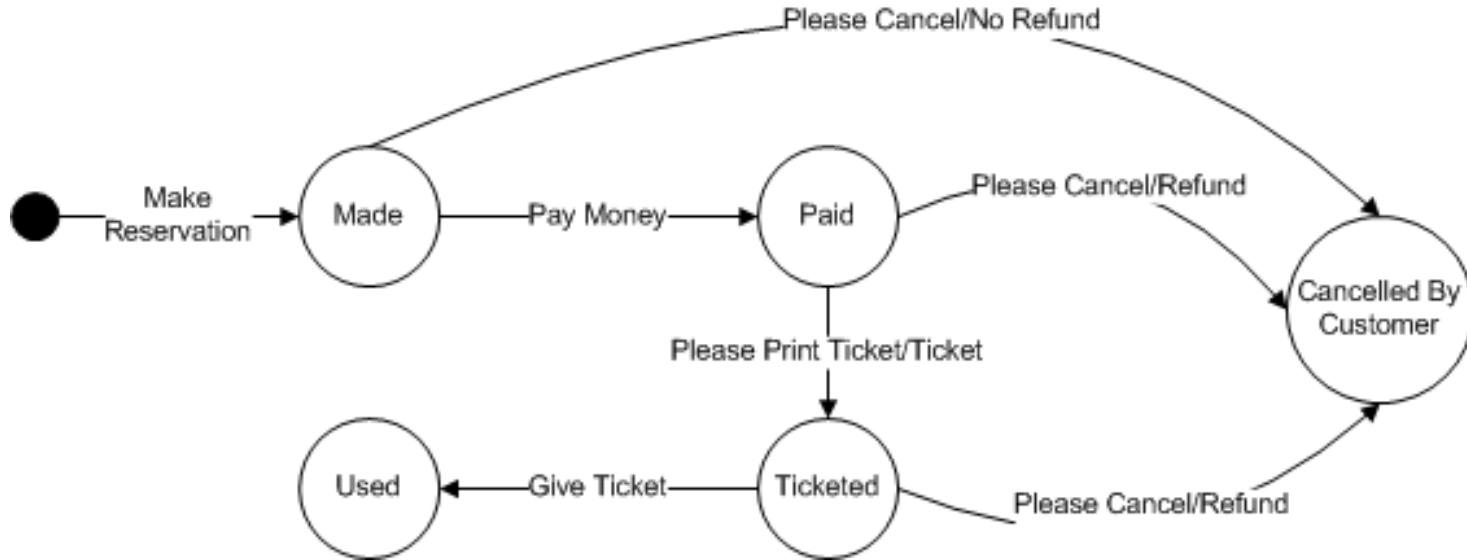
Action

Display Home Page	X							
Display Error Message		X	X	X				

State transitional diagrams

- Identify a finite number of *states* the model execution goes through
- Create a state *transition* diagram showing how the model transitions from one state to the other
- Assess the model accuracy by *analyzing* the conditions under which a state change occurs
- **State transition:** A transition between two states of a component or system.

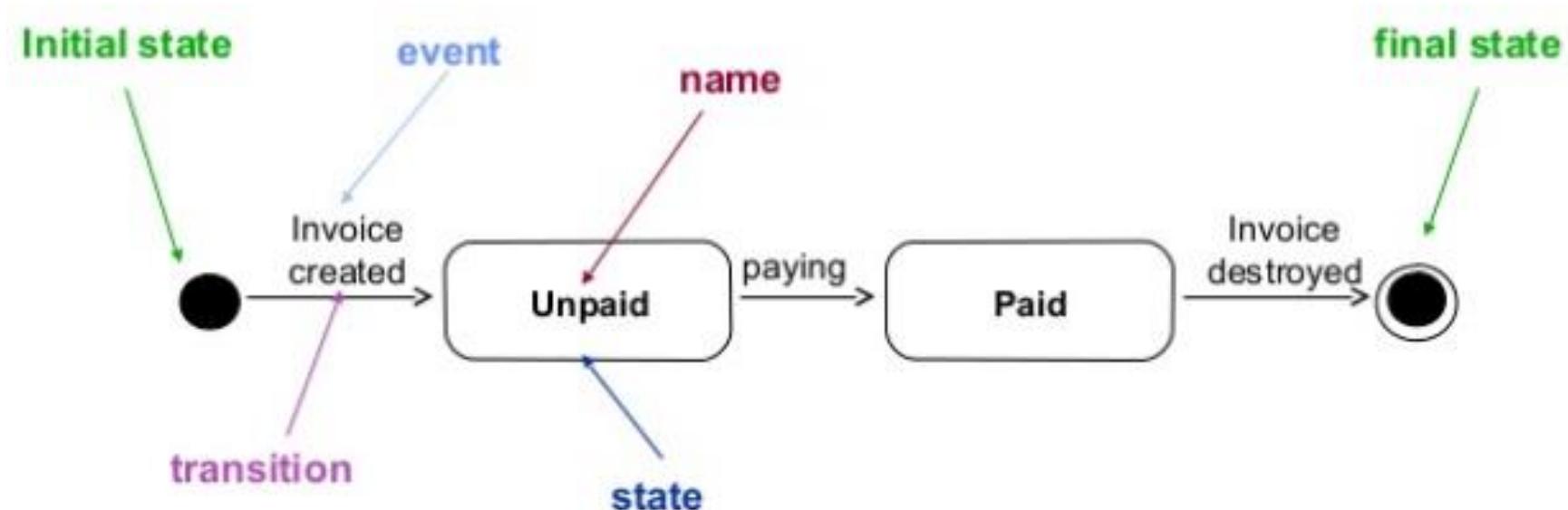
State transitional diagrams (cont'd)



- Circles/ellipses are **states**
- Lines represent **transitions** between states
- Text represents the **events** that cause transitions
- The solid circle represents an **initial state**
- A solid circle or ellipses/circles with no exit lines (transitions) are **final states**
- In the example above, minimal number of test cases to cover each state is two

Invoice Business Rule

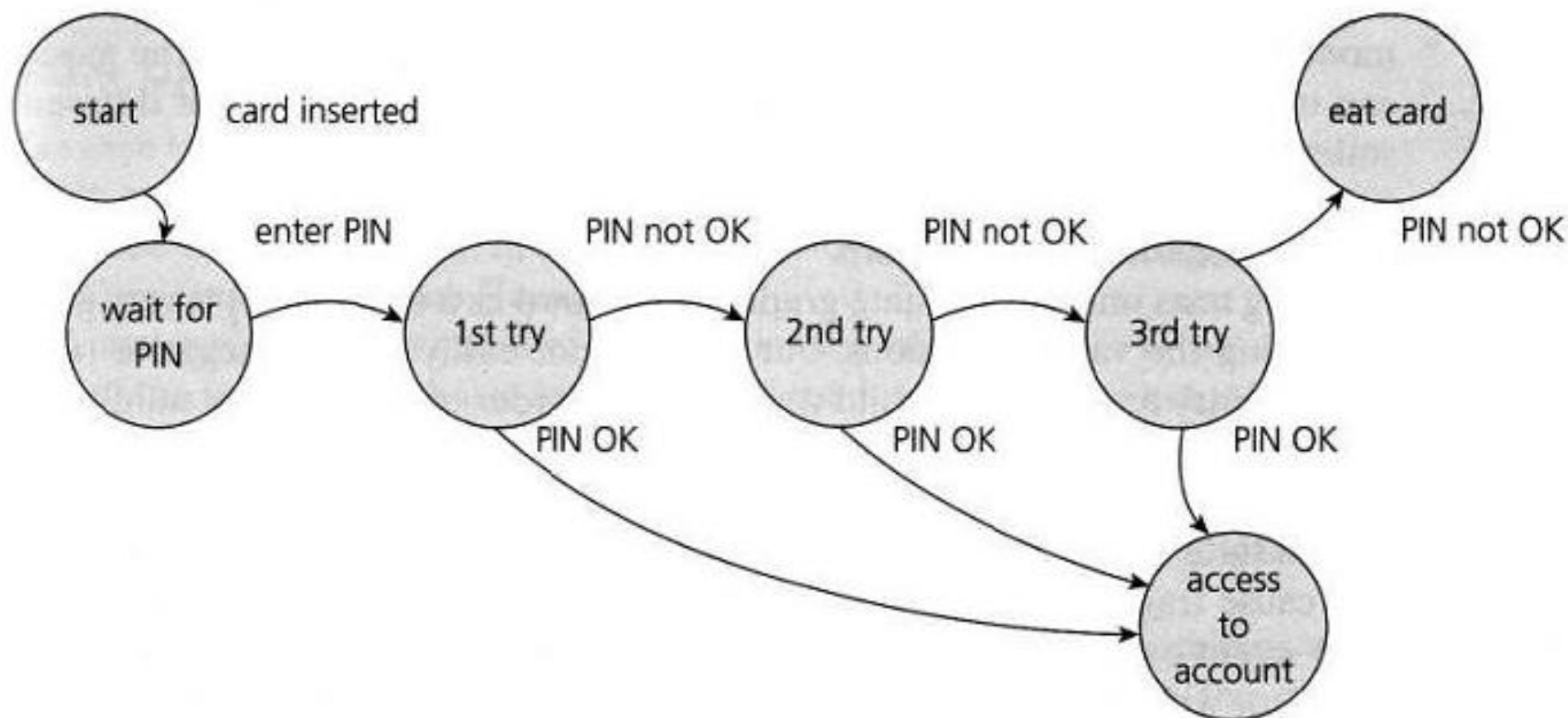
- Create invoice with status Unpaid.
- After the customer paid then update to Paid and pull invoice to the destroyed mode



Authentication Business Rules

- Request card to withdraw money
- After insert card the system wait for customer to enter PIN
- If the customer enter wrong PIN more than 3 times the ATM must eat card
- If the customer enter correct PIN the system allow to access the account and continue

State Diagram for PIN Entry



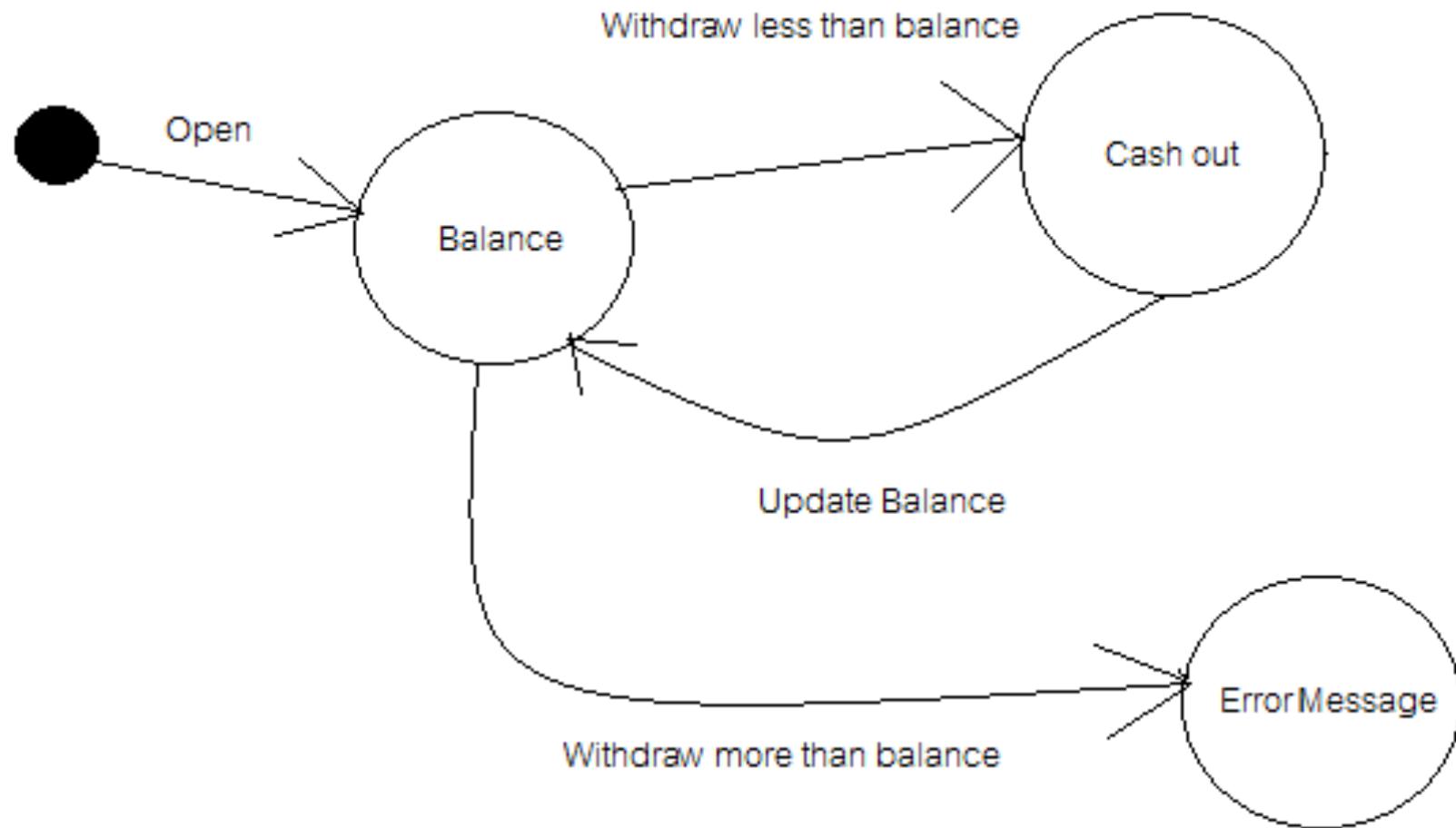
Identify Test Paths with Test Data Set

State	Authentication	Correct PIN	Incorrect PIN
S1	Wait for PIN		
S2	First Attempt	S5	S3
S3	Second Attempt	S5	S4
S4	Third Attempt	S5	S6
S5	Access Account		
S6	Eat Card		

Withdraw Business Rules

- Request to withdraw less than balance from a bank ATM, you may be given cash.
- Later you may make exactly the same request but be refused the money (because your balance is insufficient).

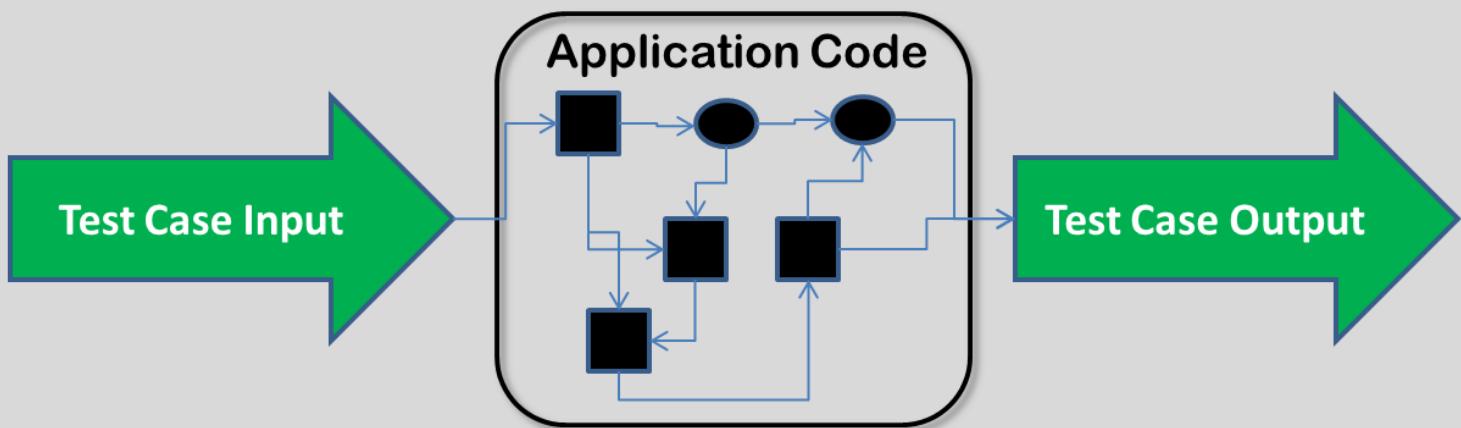
State Diagram



Identify Test Paths with Test Data Set

State	Withdrawal	Less Than Balance	More than balance
S1	Withdraw Balance	S2	S3
S2	Cash Out		
S3	Error Message		

White-box testing



WHITE BOX TESTING APPROACH

Aims of White Box Testing

- Internal security holes
- Broken or poorly structured paths in the coding processes
- The flow of specific inputs through the code
- Expected output
- The functionality of conditional loops
- Testing of each statement, object and function on an individual basis



White Box Testing Techniques

- A major White box testing technique is **Code Coverage analysis**.
 - **Statement Coverage** This technique requires **every possible statement in the code to be tested at least once** during the testing process.
 - **Branch Coverage** - This technique **checks every possible path** (if-else and other conditional loops) of a software application.
- Using **Statement and Branch coverage you generally attain 80-90% code coverage** which is sufficient.

Sample Code

```
1 int foo (int a, int b, int c, int d, float e) {  
2     float e;  
3     if (a == 0) {  
4         return 0;  
5     }  
6     int x = 0;  
7     if ((a==b) OR ((c == d) AND bug(a) )) {  
8         x=1;  
9     }  
10    e = 1/x;  
11    return e;  
12 }
```

We again examine our bug() information, which informs us that the bug method should return a false value if fed any integer greater than 1.

Statement Coverage

Test Cases	Coverage
Test Case 1 foo(0, 0, 0, 0, 0.), return 0	Statement 1-5
Test Case 2 foo(1, 1, 1, 1, 1.), return 1	Statement 6-12

Branch Coverage

Test Cases	Branch
Test Case 1 foo(0, 0, 0, 0, 0) return 0	True ($a == 0$)
Test Case 2 foo(1, 1, 1, 1, 1) return 1	False ($a == 0$), True ($(a==b)$ OR ($(c == d)$ AND $\text{bug}(a)$))
Test Case 3 foo(1, 2, 1, 2, 1) division by zero!	False ($(a==b)$ OR ($(c == d)$ AND $\text{bug}(a)$))

Other Coverage

- Method coverage is a measure of the percentage of methods that have been executed by test cases.
- Condition coverage is a measure of percentage of Boolean sub-expressions of the program that have been evaluated as both true or false outcome [applies to compound predicate] in test cases.

Method Coverage

- Test Case 1: the method call **foo(0, 0, 0, 0, 0.)**,
expected return value of 0.

Condition Coverage

Condition	True	False
a==b	Test Case 2 foo(1, 1, x, x, 1) return value 0	Test Case 3 foo(1, 2, 1, 2, 1) division by zero!
c==d	Test Case 4 foo(1, 2, 1, 1, 1) return value 1	Test Case 3 foo(1, 2, 1, 2, 1) division by zero!
bug(a)	Test Case 4 foo(1, 2, 1, 1, 1) return value 1	Test Case 5 foo(3, 2, 1, 1, 1) division by zero!

Test Data

- The minimum size of data set to identify all the application errors
- Prepare test data that will incorporate all application functionality, but not exceeding cost and time constraint for preparing test data and running tests
- ***Poorly designed testing data may not test all possible test scenarios which will hamper the quality of the software.***
- Good test data will help us;
 - Rely on the result of testing
 - Reduce the testing schedule and effort

Test Data can be Generated

- Manually
- Mass copy of data from production to testing environment
- Mass copy of test data from legacy client systems
- Automated Test Data Generation Tools

first_name	last_name	company_name	address	city	county	state	zip	phone1	phone2	email	web
James	Butt	Benton, John B Jr	6649 N Blue Gum St	New Orleans	Orleans	LA	7011504-621-68927	504-845-1427	jbutt@gmail.com	http://www.bentonjohnbjr.com	
Josephine	Darakjy	Chanay, Jeffrey A Esq	4 B Blue Ridge Blvd	Brighton	Livingston	MI	4811810-292-69388	810-374-9840	josephine_darakjy@darakjy.org	http://www.chanayjeffreyaesq.com	
Art	Venere	Chemel, James L Cpa	8 W Cerritos Ave #54	Bridgeport	Gloucester	NJ	80148749	856-636-4130	art@venere.org	http://www.chemeljameslcpa.com	
Lenna	Paprocki	Feltz Printing Service	639 Main St	Anchorage	Anchorage	AK	9950907-385-14412	907-921-2010	lpaprocki@hotmail.com	http://www.feltzprintingservice.com	
Donette	Foller	Printing Dimensions	34 Center St	Hamilton	Butler	OH	4480419-503-11893	513-549-4561	donette.foller@cox.net	http://www.printingdimensions.com	
Simona	Morasca	Chapman, Ross E Esq	3 Mcauley Dr	Ashland	Ashland	OH	52484	419-800-6759	simona@morasca.com	http://www.chapmanrosseesq.com	
Mitsue	Tollner	Morlong Associates	7 Eads St	Chicago	Cook	IL	5710605-414-26914	8565	mitsue_tollner@yahoo.com	http://www.morlongassociates.com	
Leota	Dilliard	Commercial Press	7 W Jackson Blvd	San Jose	Santa Clara	CA	9511408-752-13500	408-813-1105	leota@hotmail.com	http://www.commercialpress.com	
Sage	Wieser	Truhlar And Truhlar Attys	5 Boston Ave #88	Sioux Falls	Minnehaha	SD	52147	4895	sage_wieser@cox.net	http://www.truhlarandtruhlarattys.com	
Kris	Marrier	King, Christopher A Esq	228 Runamuck Pl #2808	Baltimore	Baltimore City	MD	1944215-874-48723	2122410-655-410-804-4694	kris@gmail.com	http://www.kingchristopheraesq.com	
Minna	Amigon	Dorl, James J Esq	2371 Jerrold Ave	Kulpsville	Montgomery	PA	31229	215-422-8694	minna_amigon@yahoo.com	http://www.dorljamesjesq.com	
Abel	Maclead	Rangoni Of Florence	37275 St Rt 17m M	Middle Island	Suffolk	NY	9003310-498-33414	310-254-3675	amaclead@gmail.com	http://www.rangoniofflorence.com	
Kiley	Caldarera	Feiner Bros	25 E 75th St #69	Los Angeles	Los Angeles	CA	4402440-780-45651	440-579-3084	kiley.caldarera@aol.com	http://www.feinerbros.com	
Graciela	Ruta	Buckley Miller & Wright	98 Connecticut Ave Nw	Chagrin Falls	Geauga	OH	7804956-537-38425	7763	gruta@cox.net	http://www.buckleymillerwright.com	
Cammy	Albares	Rousseaux, Michael Esq	56 E Morehead St	Laredo	Webb	TX	8501602-277-56195	956-841-7216	calbares@gmail.com	http://www.rousseauxmichaelsaesq.com	
Mattie	Poquette	Century Communications	73 State Road 434 E	Phoenix	Maricopa	AZ	34385	602-953-6360	mattie@aol.com	http://www.centurycommunications.com	
Meaghan	Garufi	Bolton, Wilbur Esq	69734 E Carrillo St	Mc Minnville	Warren	TN	3711931-313-09635	931-235-7959	meaghan@hotmail.com	http://www.boltonwilburesq.com	
Gladys	Rim	T M Byxbe Company Pc	322 New Horizon Blvd	Milwaukee	Milwaukee	WI	4818313-288-79598	414-377-2880	gladys.rim@rim.org	http://www.tmbxbeecompanypc.com	
Yuki	Whobrey	Farmers Insurance Group	1 State Route 27	Taylor	Wayne	MI	6110815-828-07937	313-341-4470	yuki_whobrey@aol.com	http://www.farmersinsurancegroup.com	
Fletcher	Flosi	Post Box Services Plus	394 Manchester Blvd	Rockford	Winnebago	IL	1901610-545-92147	610-492-5657	fletcher.flosi@yahoo.com	http://www.postboxservicesplus.com	
Bette	Nicka	Sport En Art	6 S 33rd St	Aston	Delaware	PA	43615	610-492-4643	bette_nicka@cox.net	http://www.sportenart.com	
Veronika	Inouye	C 4 Network Inc	6 Greenleaf Ave	San Jose	Santa Clara	CA	9511408-540-11785	815-426-408-813-4592	vinouye@aol.com	http://www.cnetworkinc.com	
Willard	Kolmetz	Ingalls, Donald R Esq	618 W Yakima Ave	Irving	Dallas	TX	7506972-303-29197	972-896-4882	willard@hotmail.com	http://www.ingallsdonaldresq.com	
Maryann	Royster	Franklin, Peter L Esq	74 S Westgate St	Albany	Albany	NY	1220518-966-47987	518-448-8982	mroyster@royster.com	http://www.franklinpeterlesq.com	
Alisha	Slusarski	Wtlz Power 107 Fm	3273 State St	Middlesex	Middlesex	NJ	732-658-88463154	732-635-3453	alisha@slusarski.com	http://www.wtlzpowerfm.com	
Allene	Iturbide	Ledecky, David Esq	1 Central Ave	Stevens Point	Portage	WI	5448715-662-16764	715-530-9863	allene_iturbide@cox.net	http://www.ledeckydavidesq.com	
Chanel	Caudy	Professional Image Inc	86 Nw 66th St #8673	Shawnee	Johnson	KS	6621913-388-82079	913-899-1103	chanel.caudy@caudy.org	http://www.professionalimageinc.com	
Ezekiel	Chui	Sider, Donald C Esq	2 Cedar Ave #84	Easton	Talbot	MD	2160410-669-11642	410-235-8738	ezekiel@chui.com	http://www.siderdonaldcesq.com	
Willow	Kuskosko	U Pull It	90991 Thorburn Ave	New York	New York	NY	1001212-582-14976	212-934-5167	wkuskosko@yahoo.com	http://www.upullit.com	

Test Data for White Box Testing

- Test data may be selected by taking into account the following things:

- เป็นที่พึงปรารถนาที่จะครอบคลุมสาขาให้ได้มากที่สุด
สามารถสร้างข้อมูลการทดสอบเพื่อให้ทุกสาขาในชอร์สโค้ดของโปรแกรมได้รับการทดสอบอย่างน้อยหนึ่งครั้ง
- การทดสอบเส้นทาง: เส้นทางทั้งหมดในชอร์สโค้ดของโปรแกรมได้รับการทดสอบอย่างน้อยหนึ่งครั้ง - ข้อมูลการทดสอบสามารถออกแบบให้ครอบคลุมกรณีต่างๆ ได้มากที่สุดเท่าที่จะเป็นไปได้
- Negative API testing: *Data ผิด + call method ที่ถูก*
 - Testing data may contain invalid parameter types used to call different methods *Data ผิด + call method ที่ถูก*
 - Testing data may consist in invalid combination's of arguments which are used to call the program's methods

Test Data for Black Box Testing

୨ୟମ୍ବ

- **No data:** Check system response when no data is submitted
- **Valid data :** Check system response when Valid test data is submitted
- **Invalid data :**Check system response when *InValid* test data is submitted
- **Illegal data format:** Check system response when test data is in invalid format
- **Boundary Condition Data set:** Test data meeting bounding value conditions
- **Equivalence Partition Data Set :** Test data qualifying your equivalence partitions.
- **Decision Table Data Set:** Test data qualifying your decision table testing strategy
- **State Transition Test Data Set:** Test data meeting your state transition testing strategy
- **Use Case Test Data:** Test Data in-sync with your use cases.

Test Data for Security Testing

- The set of data that need to be designed in order to fully test a software security must cover the following topics:

การรักษาความลับ: ข้อมูลทั้งหมดที่ได้รับจากลูกค้าจะถูกเก็บไว้เป็นความลับอย่างเข้มงวดที่สุดและจะไม่เปิดเผยกับบุคคลภายนอก ดังตัวอย่าง ส้นๆ หากแอปพลิเคชันใช้ SSL คุณสามารถออกแบบชุดข้อมูลทดสอบซึ่งยืนยันว่าการเข้ารหัสทำอย่างถูกต้อง

- ความสมบูรณ์: ตรวจสอบว่าข้อมูลที่ระบบให้มานั้นถูกต้อง ในการออกแบบข้อมูลทดสอบที่เหมาะสม คุณสามารถเริ่มต้นด้วยการดึงลิ๊กเก็ย์กับการออกแบบโค้ด ฐานข้อมูล และโครงสร้างไฟล์
- การพิสูจน์ตัวตน: แสดงถึงกระบวนการสร้างตัวตนของผู้ใช้ ข้อมูลการทดสอบสามารถออกแบบเป็นชุดซึ่งผู้ใช้และรหัสผ่านที่แตกต่างกัน และมีวัตถุประสงค์เพื่อตรวจสอบว่าเฉพาะบุคคลที่ได้รับอนุญาตเท่านั้นที่สามารถเข้าถึงระบบซอฟต์แวร์ได้

การอนุญาต: บอกสิทธิ์ของผู้ใช้เฉพาะ ข้อมูลการทดสอบอาจประกอบด้วยผู้ใช้ บทบาท และการดำเนินการที่แตกต่างกัน เพื่อตรวจสอบเฉพาะผู้ใช้ที่มีสิทธิ์เพียงพอเท่านั้นที่สามารถดำเนินการบางอย่างได้

Test Data for Performance Testing

- **Data set for performance, load and stress testing:** This data set should be large in volume. You can get data from users. They will help you estimate the volume of data that will be generated to the system at the specific period.
- In case you are in a **maintenance testing** project you could copy data from the production environment into the testing bed. It is a good practice to **anonymize** (scramble) sensitive customer data like Social Security Number , Credit Card Numbers , Bank Details etc while the copy is made.

7. REQUIREMENTS TRACEABILITY MATRIX

Requirements Traceability Matrix (RTM)

- A table that links requirements to their origins and traces them throughout the project life cycle. Developing the RTM helps to ensure that each requirement adds business value and that approved requirements are delivered.
- Goal
 - Establish a baseline for requirements change control, design, and testing
 - Help tester to check coverage of the test cases through requirements

Benefits of using Traceability Matrix

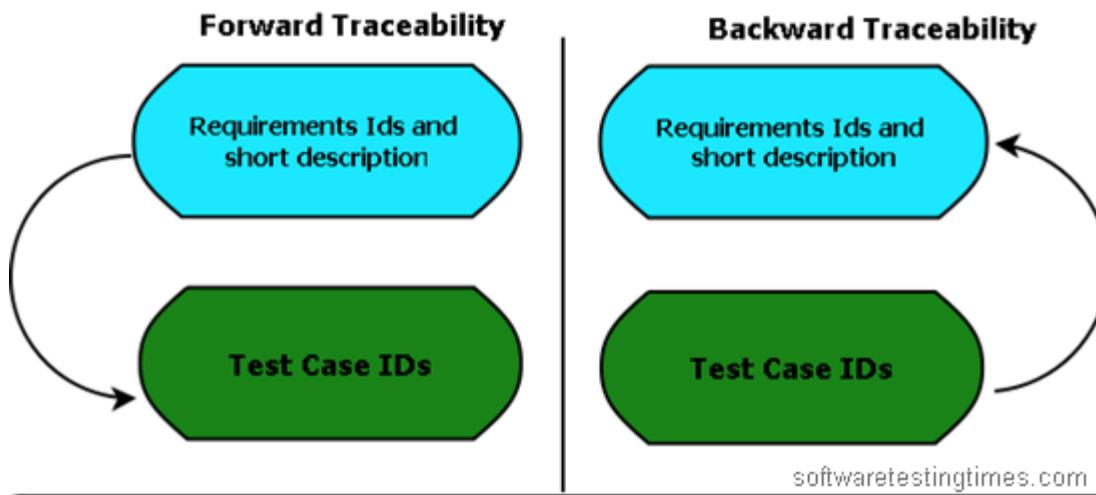
- Make obvious to the client that the software is being developed as per the requirements.
- To make sure that all requirements included in the test cases
- To make sure that developers are not creating features that no one has requested
- Easy to identify the missing functionalities.
- If there is a change request for a requirement, then we can easily find out which test cases need to update.
- The completed system may have “Extra” functionality that may have not been specified in the design specification, resulting in wastage of manpower, time and effort.

Types of Traceability Matrix

Forward Traceability – Mapping of Requirements to Test cases

Backward Traceability – Mapping of Test Cases to Requirements

Bi-Directional Traceability - A Good Traceability matrix is the References from test cases to basis documentation and vice versa.



Forward Traceability Matrix ensures – We are building the Right Product.

Backward Traceability Matrix ensures – We the Building the Product Right.

No traceability or Incomplete Traceability Results into:

- Poor or unknown test coverage, more defects found in production
- It will lead to miss some bugs in earlier test cycles which may arise in later test cycles. Then a lot of discussions arguments with other teams and managers before release.
- Difficult project planning and tracking, misunderstandings between different teams over project dependencies, delays, etc

Traceability matrix (example)

Requirement Section	Function Section	Test case	Notes
1.1. Validation of user login credentials.	4.1. User login validation.	6.1.4. User login with proper credentials. 6.1.5. User login with invalid username. 6.1.6. User login with invalid password.	
1.2. Validation of credit card information.	7.2.4. Credit card information verification.	10.1.1. Valid credit card information input. 10.1.2. Invalid credit card number. 10.1.3. Invalid credit card name. ...	

Traceability matrix (example)

Requirement ID	Requirement description	TC 001	TC 002	TC 003
SR-1.1	User should be able to do this	x		
SR-1.2	User should be able to do that	x		
SR-1.3	On clicking this, following message should appear		x	
SR-1.4			x	
SR-1.5		x		x
SR-1.6				x
SR-1.7			x	

8. TEST CASES/SCRIPTS

Test Requirements, Test Scenarios and Test Cases

- Test Requirements → Tell us What to be tested and expected results?
 - Cover all specific requirements and use to validate the requirements
- Test Scenario → Tell us What to be tested?
 - Cover a wide range of possibility
- Test Cases → Tell us How to test?
 - Test Procedures, test steps and data test must be added.
 - Test Requirements and Test Scenarios are reference documents.

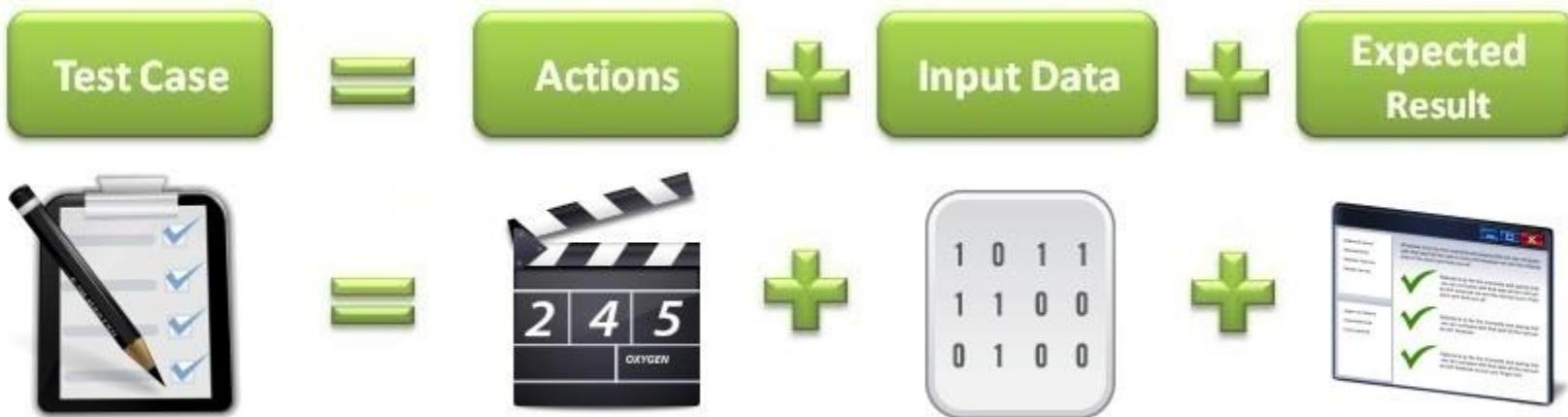
What is Test Case?

- A set of test steps, execution conditions and expected results developed for a particular objective, such as to exercise a particular program path or to verify compliance with a specific requirement
- Test cases are specific inputs and procedures that you will follow when you test software.
- A Test Case is like a recipe. You follow the steps to produce an end result.

Test Cases Objectives

- Find defects
- Maximize bug count
- Block premature product releases
- Help managers make ship / no-ship decisions
- Assess conformance to specification
- Conform to regulations
- Verify correctness of the product
- Assure quality

Test Case



A good Test Case is one which is more realistic, validates requirements and intends to find maximum defects.

Preparation

- Read Specification and Requirement Carefully
- Understand Test Requirements thoroughly
- Be clear with the design and implementation details
- Analyze and Identify all possible scenarios
- Identify the test environments and test types
- Should be clear of behavior under failure condition (invalid input, boundary values etc)

Basic Parts of a Test Case

- A Test Case is multiple Steps which are comprised of these 4 basics parts
 - Description of the Test Case
 - Description of each Step
 - Expected result each Step
 - Actual result each Step

Test Step

- A Test Step is the smallest portion of a Test Case. It typically describes a single action in a chain of actions which go on to comprise a Test Case:
- Usually start with a verb such as:
 - Verify, Validate, Navigate
 - Usually are doing something very discrete.
 - Validate color, Verify SSN, Navigate to account
 - Examples:
 - Step 1. Navigate to SSN field.
 - Step 2. Enter a valid ID.
 - Step 3. Validate that the SSN field will only accept a 9 digit number

Writing Good Test Case

- As far as possible, write test cases in such a way that you test only one thing at a time. Do not overlap or complicate test cases. Attempt to make your test cases ‘atomic’.
- Ensure that all positive scenarios and negative scenarios are covered.
- Language:
 - Write in simple and easy to understand language.
 - Use active voice: Do this, do that.
 - Use exact and consistent names (of forms, fields, etc).
- Characteristics of a good test case:
 - *Accurate*: Exacts the purpose.
 - *Economical*: No unnecessary steps or words.
 - *Traceable*: Capable of being traced to requirements.
 - *Repeatable*: Can be used to perform the test over and over.
 - *Reusable*: Can be reused if necessary.

Keep in mind during writing

- Document Sign off
- Test Case Naming Convention
- Track Test Case changes
- Test Case Priority
- Others will execute your test cases
- Test Case review is a must
- Test Data needs
- Environment needs
- “Log on” (user ID) needs
- Hardware needs
- Batch requirements
- Interfacing applications (up and down stream)
- Positive and negative scenarios
- Boundary Test (how big/small can I go and what happens if I do)

Test Scenario : Check Login Functionality

Test Case No.	Description	Expected Result
1	Check results on entering valid User Id & Password	Success Login and show home screen
2	Check results on entering Invalid User ID & Password	Error message pop up, not allow to login
3	Check response when User ID is Empty & Login Button is pressed	Error message pop up, not allow to login

TEST CASE TEMPLATE AND EXAMPLE

Test Cases, Scripts and Data Test

- A test case includes:
 - The purpose of the test.
 - Special hardware requirements, such as a modem.
 - Special software requirements, such as a tool.
 - Specific setup or configuration requirements.
 - A description of how to perform the test.
 - A data or set of test data.
 - The expected results or success criteria for the test.

Pre-conditions/Information

- Browsers – IE, Firefox, Safari
- O/S – Linux, Windows
- Access Control – Logins, Roles
- Test Data requirements
- Date/Time considerations
- Other document references

Formal Test Case Contents

Test Suit ID	The ID of the test suite to which this test case belongs.
Test Case NO#	The ID of the test case.
Test Case Summary	The summary / objective of the test case.
SRS Ref/TR Ref	The ID of the requirement this test case relates/traces to.
Precondition	Any prerequisites or preconditions that must be fulfilled prior to executing the test.
Test Procedure	Step-by-step procedure to execute the test.
Test Data	The test data, or links to the test data, that are to be used while conducting the test.
Expected Result	The expected result of the test.
Actual Result	The actual result of the test; to be filled after executing the test.
Pass/Fail Status	Pass or Fail. Other statuses can be 'Not Executed' if testing is not performed and 'Blocked' if testing is blocked.
Remarks	Any comments on the test case or test execution.
Created By	The name of the author of the test case.
Date of Creation	The date of creation of the test case.
Executed By	The name of the person who executed the test.
Date of Execution	The date of execution of the test.
Test Environment	The environment (Hardware/Software/Network) in which the test was executed.
Post-conditions:	What will be happened after all steps are done?

Sample Test Case

Test Suite ID	TS001
Test Case NO	TS001-01
Test Case Summary	To verify that clicking the Generate Coin button generates coins.
SRS Ref/ TR Ref	RS001 බෙංකාගැ කොරෝ මි තේස් සේවීතුනු
Precondition	<p>1. User is authorized. 2. Coin balance is available.</p>
Test Procedure	<p>1. Select the coin denomination in the Denomination field. 2. Enter the number of coins in the Quantity field. 3. Click Generate Coin.</p>
Test Data	<p>1. Denominations: 0.05, 0.10, 0.25, 0.50, 1, 2, 5 2. Quantities: 0, 1, 5, 10, 20</p>
Expected Result	<p>1. Coin of the specified denomination should be produced if the specified Quantity is valid (1, 5) 2. A message 'Please enter a valid quantity between 1 and 10' should be displayed if the specified quantity is invalid.</p>
Actual Result	<p>1. If the specified quantity is valid, the result is as expected. 2. If the specified quantity is invalid, nothing happens; the expected message is not displayed</p>
Pass/Fail Status	Fail
Remarks	This is a sample test case.
Created By	John Doe
Date of Creation	01/14/2020
Executed By	Jane Roe
Date of Execution	02/16/2020
Test Environment	<ul style="list-style-type: none"> • OS: Windows Y • Browser: Chrome N
Post-conditions:	None

Sample Test Cases Template

Why test cases we need documents?



Applied Test Cases

- Create test case match with testing activities or testing features
 - Test only one condition at a time such as personal tax deduction, fee calculation, or date time checking
 - Test specific function or feature such as login, access authorization, search book and etc.
- The unique template must be set and informed all team members.

Test cases Example

Test Ref.	Reqs Ref.	Scenarios	Inputs Data	Expected Result	Actual Result	Pass/ Fail
2.001	REF003	Search Researchers	1. Forenames = John 2. Surname = <Blank> 3. eMail = <Blank>	All UCL researchers with forenames starting Pete displayed in alphabetic order, 23 records per page List comprises Name, Department, Occupation Type All data items hyperlinked	427 matches - paging working correctly, data displayed correctly and in reasonable time (5 secs)	
2.002	REF003	Search Researchers	1. Forenames = <Blank> 2. Surname = Smith 3. eMail = <Blank>	All UCL researchers with surnames starting Smith displayed in alphabetic order, 23 records per page List comprises Name, Department, Occupation Type All data items hyperlinked	61 matches - paging working correctly, data displayed correctly and in reasonable time (5 secs)	

Test Cases Example

- System Test of input of numeric month into data field

Ref.	Field/Button	Action	Input	Expected Result	Pass/Fail
001	Month	Enter Data	0	Data rejected. Error Message 'Invalid Month'	
002	Month	Enter Data	1	Data Accepted, January Displayed	
003	Month	Enter Data	06	Data Accepted, June Displayed	
004	Month	Enter Data	12	Data Accepted, December Displayed	
005	Month	Enter Data	13	Data rejected. Error Message 'Invalid Month'	

Test Cases and Test Data Example

TC ID#	Scenario / Condition	PIN	Account #	Amount Entered (or chosen)	Amount in Account	Amount in ATM	Expected Result
CW1.	Scenario 1 - Successful Cash Withdraw	4987	809 - 498	50.00	500.00	2,000	Successful cash withdrawal. Account balance updated to 450.00
CW2.	Scenario 2 - ATM out of Money	4987	809 - 498	100.00	500.00	0.00	Cash Withdraw option unavailable, end of use case
CW3.	Scenario 3 - Insufficient funds in ATM	4987	809 - 498	100.00	500.00	70.00	Warning message, return to Basic Flow - Enter Amount

CLASS PRACTICE

TEST CASES GENERATION

Scenario

ลูกค้าซื้อสินค้า และกำลังตัดสินใจเลือกประเภทการส่งสินค้า

- ลูกค้าอาจเลือกส่งสินค้าจากตัวเลือกต่อไปนี้เพื่อให้ส่งของ
- ถ้าลูกค้าไม่เลือกวิธีการส่ง ระบบจะแจ้งเตือน “ไม่สามารถทำการต่อได้ถ้าไม่เลือกวิธีการส่งของ”

ราคา	น้อยกว่า \$20	\$20 – \$29.99	\$30 – 39.99	\$40 และมากกว่า
วิธีการส่ง				
Overnight	\$12.00	\$16.00	\$22.00	\$27.00
2-day air	\$8.00	\$8.00	\$4.00	\$0
USPS	\$6.00	\$6.00	\$6.00	\$0

Test Cases	Expected Result
ลูกค้าเลือกซื้อสินค้าที่ยอดรวมน้อยกว่า \$20 และเลือก overnight shipping	Pay 12 US\$
ลูกค้าเลือกซื้อสินค้าที่ยอดรวมน้อยกว่า \$20 และเลือก 2-day air	Pay 8 US\$
ลูกค้าเลือกซื้อสินค้าที่ยอดรวมน้อยกว่า \$20 และเลือก USPS	Pay 6 US\$
ลูกค้าเลือกซื้อสินค้าที่ยอดรวมระหว่าง \$20-\$29.99 และเลือก overnight shipping	Pay 16 US\$
ลูกค้าเลือกซื้อสินค้าที่ยอดรวมระหว่าง \$20-\$29.99 และเลือก 2-day air	Pay 8 US\$
ลูกค้าเลือกซื้อสินค้าที่ยอดรวมระหว่าง \$20-\$29.99 และเลือก USPS	Pay 6 US\$
ลูกค้าเลือกซื้อสินค้าที่ยอดรวมระหว่าง \$30-\$39.99 และเลือก overnight shipping	Pay 22 US\$
ลูกค้าเลือกซื้อสินค้าที่ยอดรวมระหว่าง \$30-\$39.99 และเลือก 2-day air	Pay 4 US\$
ลูกค้าเลือกซื้อสินค้าที่ยอดรวมระหว่าง \$30-\$39.99 และเลือก USPS	Pay 6 US\$
ลูกค้าเลือกซื้อสินค้าที่ยอดรวมมากกว่าเท่ากับ \$40 และเลือก overnight shipping	Pay 27 US\$
ลูกค้าเลือกซื้อสินค้าที่ยอดรวมมากกว่าเท่ากับ \$40 และเลือก 2-day air	Pay 0 US\$
ลูกค้าเลือกซื้อสินค้าที่ยอดรวมมากกว่าเท่ากับ \$40 และเลือก USPS	Pay 0 US\$
ลูกค้าไม่เลือกวิธีการจัดส่งสินค้า	แสดงข้อความเตือนให้เลือก ก่อนที่จะดำเนินการต่อไป

สร้าง Data Test : รายการสินค้า

Product	Description	Unit Price
1. 	Etekcity Ultra Bright Portable LED Camping Lantern Flashlights (Black, Collapsible)	\$7.99
2. 	Microsoft Band 2 – Large	\$174.99
3. 	Fire, 7" Display, Wi-Fi, 8 GB - Includes Special Offers, Black	\$49.99
4. 	SanDisk 64 GB micro SD Memory Card for All-New Fire Tablets and All-New Fire TV	\$22.99

Test Cases	Data Test	Expected Result
ลูกค้าเลือกซื้อสินค้าที่ยอดรวมน้อยกว่า \$20 และเลือก overnight shipping	Product 1	Pay 12 US\$
ลูกค้าเลือกซื้อสินค้าที่ยอดรวมน้อยกว่า \$20 และเลือก 2-day air	Product 1	Pay 8 US\$
ลูกค้าเลือกซื้อสินค้าที่ยอดรวมน้อยกว่า \$20 และเลือก USPS	Product 1	Pay 6 US\$
ลูกค้าเลือกซื้อสินค้าที่ยอดรวมระหว่าง \$20-\$29.99 และเลือก overnight shipping	Product 4	Pay 16 US\$
ลูกค้าเลือกซื้อสินค้าที่ยอดรวมระหว่าง \$20-\$29.99 และเลือก 2-day air	Product 4	Pay 8 US\$
ลูกค้าเลือกซื้อสินค้าที่ยอดรวมระหว่าง \$20-\$29.99 และเลือก USPS	Product 4	Pay 6 US\$
ลูกค้าเลือกซื้อสินค้าที่ยอดรวมระหว่าง \$30-\$39.99 และเลือก overnight shipping	Product 1 & 4	Pay 22 US\$
ลูกค้าเลือกซื้อสินค้าที่ยอดรวมระหว่าง \$30-\$39.99 และเลือก 2-day air	Product 1 & 4	Pay 4 US\$
ลูกค้าเลือกซื้อสินค้าที่ยอดรวมระหว่าง \$30-\$39.99 และเลือก USPS	Product 1 & 4	Pay 6 US\$
ลูกค้าเลือกซื้อสินค้าที่ยอดรวมมากกว่าเท่ากับ \$40 และเลือก overnight shipping	Product 3	Pay 27 US\$
ลูกค้าเลือกซื้อสินค้าที่ยอดรวมมากกว่าเท่ากับ \$40 และเลือก 2-day air	Product 3	Pay 0 US\$
ลูกค้าเลือกซื้อสินค้าที่ยอดรวมมากกว่าเท่ากับ \$40 และเลือก USPS	Product 2	Pay 0 US\$
ลูกค้าไม่เลือกวิธีการจัดส่งสินค้า	Product 1 & 2 & 3 & 4	แสดงข้อความเตือนให้เลือก ก่อนที่จะดำเนินการต่อไป

Test Cases Expedia Search

The screenshot shows the Expedia search interface for flight + hotel bookings. At the top, there are tabs for Hotels, Flights, Flight + Hotel (selected), Cars, Things to Do, and Discover. Below the tabs, there are fields for Origin and Destination, both labeled "City or airport". Under "Departing" and "Returning", there are date input fields. A checkbox "I only need a hotel for part of my stay" is checked. Advanced options and a dropdown for travel class ("Economy") are also present. A yellow "Search" button is on the left, and a promotional message "Save up to 59% Book Flight + Hotel at the same time*" is on the right. In the center, there are dropdown menus for "Rooms" (set to 1), "Adults (18+)" (set to 2), and "Children (0-17)" (set to 0). The background features a blurred image of a bridge at night.

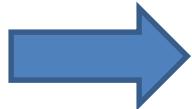
This screenshot shows the "Advanced options" dropdown menu. It includes a dropdown for travel class with "Economy" selected, and other options like First class, Business, Premium economy, and Economy.

This screenshot shows a date selection calendar for June and July 2017. The calendar allows users to select dates for their trip. The month of June starts on Monday, June 5th. The month of July starts on Saturday, July 1st. A blue arrow points to the "Close" button at the bottom right of the calendar.

This screenshot shows the "Destination" dropdown menu. It lists various cities and airports with their respective codes and locations, such as Barcelona (BCN-All Airports) near Barcelona, Spain; Berlin (BER-All Airports) near Berlin, Germany; Denpasar (DPS-Ngurah Rai Intl.) near Bali, Indonesia; Bangkok (BKK-All Airports) near Bangkok, Thailand; Birmingham (BHX-All Airports) near Birmingham, England, UK; Budapest (BUD-Ferenc Liszt Intl.) near Budapest, Hungary; Buenos Aires (BUE-All Airports) near Buenos Aires, Argentina; Brussels (BRU-All Airports) near Brussels, Belgium; Boston (BOS-All Airports) near Boston, Massachusetts, USA; and Beijing (BJS-All Airports) near Beijing, China. A blue arrow points to the "Close" button at the bottom right of the dropdown.

Expected Results

ถ้าอยู่ในระหว่างดำเนินการ
ค้นหาข้อมูลให้แสดงผลตามรูป



ถ้าหาไม่พบตามที่เลือก หรือไม่ match
ให้แสดงข้อความตามรูป

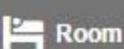
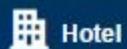


The screenshot shows the Expedia.co.th logo at the top. Below it, there's a large, light gray circular placeholder icon. To the left of the placeholder are three circular icons with checkmarks: one with a yellow airplane, one with a building, and one with a dollar sign. To the right of each icon is a corresponding text description: "Exclusive Member Deals", "Book from over 321,000 Hotels and 400 Airline partners", and "Price Guarantee".

We were unable to find any flights for your package. Please adjust your search or [continue booking your hotel separately.](#)

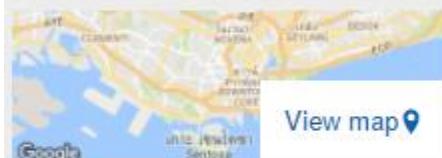
តារាងប្រចាំខែមុនកីឡាដែលត្រូវបាន

ORIGIN	DESTINATION	DATES	ROOMS	Change search 
 Bangkok, Thailand (BKK)	Singapore (all), Singapore	Sun, 18 Jun - Sun, 25 J...	1	



Start by choosing your hotel

The trip prices shown below include Flight + Hotel, taxes and fees, but do not include baggage fees or other fees charged directly by the airline for the included flight.



[View map !\[\]\(80aafe61bc345dfeeca14a737b6ec333_img.jpg\)](#)

Search by property name



[Go](#)

Filter properties by

Property Class

 5 Stars

 4 Stars

Distance from City Centre

Price

Guest Rating

Recommended

More 

Deal of the Day



Raffles Singapore 



City Hall

02 105 5728 • Expedia Rate



Economy direct flights

included in price

7 people booked this property in the last 48 hours

Booked in the last 9 hours

Book this and save 100% on your flight

Exceptional! 4.7/5

(614 reviews)

We have 3 left at

~~฿76,257~~

฿63,990

per person includes flight + hotel, taxes and fees



Earn 6,399 Expedia+ points

Test Case : Search hotel with found data

Action Steps	Test Data	Expected Results
1. ลูกค้าเลือกปุ่มค้นหา Hotels		แสดงข้อมูลที่ลูกค้าต้องการ ปรากฏขึ้นมา
2. ลูกค้าใส่ข้อมูลที่ต้องการค้นหาครบถ้วนถูกต้อง และมีข้อมูลที่ตรงกันในระบบ	From 1/12/2017 to 10/12/2017 Chiang Mai, Thailand, 2 guests	สามารถใส่ข้อมูลได้
3. ลูกค้ารอประเมินผลการค้นหา		ปรากฏจอภาพแสดงให้ลูกค้าทราบว่ากำลังค้นหาข้อมูลให้
4. ตรวจสอบการแสดงผลข้อมูล		แสดงข้อมูลที่ตรงกับที่ลูกค้าค้นหานบนจอภาพ โดยมีที่พักใน Chiang Mai ปรากฏ และมีช่วงเวลาตรงกับที่เลือก

Test Case : Search hotel with not found data

Action Steps	Test Data	Expected Results
1. ลูกค้าเลือกปุ่มค้นหา Hotels		แสดงข้อมูลที่ลูกค้าต้องการ ปรากฏขึ้นมา
2. ลูกค้าใส่ข้อมูลที่ต้องการค้นหาครบถ้วนถูกต้อง และมีข้อมูลที่ตรงกันในระบบ	From 1/12/2017 to 10/12/2017 Chiang Khong, Thailand, 2 guests	สามารถใส่ข้อมูลได้
3. ลูกค้ารอประมาณผลการค้นหา		ปรากฏเฉพาะการแสดงให้ลูกค้าทราบว่ากำลังค้นหาข้อมูลให้
4. ตรวจสอบการแสดงผลข้อมูล		แสดงข้อความให้ลูกค้าเลือกใส่ข้อมูลค้นหาใหม่

Practice : Online Shopping

https://www.lazada.co.th

LAZADA CO.TH

ค้นหาในลาซาด้า

CHANGE LANGUAGE

ข้อปุ่นกาวเด่นนนนแนป
ขายสินค้ากับลาซาด้า
ช่วยเหลือ
ติดตามสินค้า
ลงชื่อเข้าใช้
สมัครสมาชิก

สูบเงิน 100.-

อุปกรณ์ อิเล็กทรอนิกส์
อุปกรณ์เสริม อิเล็กทรอนิกส์
ทีวีและเครื่องใช้ไฟฟ้าในบ้าน
สุขภาพและความงาม
เด็กอ่อน และของเล่น
ชุดเครื่องแต่งกายเด็ก และสัตว์เลี้ยง
บ้านและไลฟ์สไตล์
แฟชั่นผู้หญิง
แฟชั่นผู้ชาย
เครื่องประดับ
กีฬาและ การเดินทาง
ยานยนต์ และอุปกรณ์

มหกรรม
ของเล่น
ลดสูงสุด
80%

ช้อปเลย 15 - 17 ธ.ค. 61

LazMall GlobalCollection เติมเงิน&เติลออนไลน์ ใส่โค้ดลดเพิ่ม

เทรนด์ยิ่ต Ipad 2018 GTX 1050Ti Pipper Standard ข้อความ

Example 1: Test Scenario online

- **Test Scenario 1:** ดู Promotion ประจำวัน
- **Test Scenario 2:** ดูสินค้าที่อยู่ใน categories
- **Test Scenario 3:** ค้นหาสินค้าตามชื่อสินค้า
- **Test Scenario 4:** ตรวจสอบรายการซื้อสินค้า
- **Test Scenario 5:** ลงทะเบียนลูกค้าใหม่
- **Test Scenario 6:** เปลี่ยนภาษาไทย/อังกฤษ
- **Test Scenario 7:** ขอความช่วยเหลือ

Test Scenario 1: ดู Promotion ประจำวัน

Test Case No.	Description	Expected Result
1	Check การแสดงผล Promotion บน จอภาพ	Promotion image ของวันเปลี่ยน ทุก ๆ 3 วินาที และตรงกับที่ตั้งไว้
2	Check การ Link Promotion	เมื่อกด Promotion แสดงหน้า Page Detail ของ Promotion นั้น
3	Check รายละเอียดการแสดง Promotion	ในหน้า Page จะมีเฉพาะข้อมูลของ Promotion Product นั้น ๆ เท่านั้น
4	Check การแสดงผล Promotion บน จอภาพของวันถัดไป	Promotion image ของวันเปลี่ยน ทุก ๆ 3 วินาที และตรงกับที่ตั้งไว้

Practice : LIVE TV

Not secure | www.workpointtv.com

workpoint ช่อง 23

((LIVE STREAM)) รายการสด ดูรายการย้อนหลัง พั้งรายการ ศิลปิน dara เพลง CONTENT LICENSING

ข่าวสารกิจกรรม ข้อมูลรีวิว ติดต่อสอบถาม

Facebook Twitter Instagram YouTube

บัดดี้ ตีบี

ทุกวันจันทร์ เวลา 20.05 น.

รายการสด

• ถ่ายทอดสด 8 2797

LIVE

workpoint

WorkpointOfficial

Workpoint กดหมายเลข 23

play button

workpoint | 23

workpoint ช่อง 23

NOW SHOWING

ใบค่ากองคำ 7 ((LIVE STREAM))

เวลา	รายการ
18:20 น.	ใบค่ากองคำ 7
20:00 น.	ข่าวในพระราชสำนัก
20:15 น.	ใบค่ากองคำเด็ก 4
21:15 น.	I Can See Your Voice Thailand

workpoint | 23

Practice : facebook

https://www.facebook.com

Ray Wannasuk

News Feed

Messenger

Watch

Shortcuts

- yanwongkons - Fre... 20+
- RayMon's Garden 20+
- Extra income
- สำหรับนักพัฒนา Ba...
- DeepQuant 3

See More...

Explore

- Events 16
- Groups
- Pages
- Fundraisers
- Friend Lists

See More...

Create Post

What's on your mind, Ray?

Photo/Video Tag Friends Feeling/Activ...

Somkiat Puisungnoen and 4 others commented on this.

Tanjai TJ Kongyuen is 😊 feeling fantastic. 4 hrs ·

สอบผ่านแล้วจ้าาาาา
I pass the ISTQB Exam with 30/40
Now I'm Certified Tester ~~~~

Stories

Add to Your Story Share a photo, video or write something

ฟ้อมดราม เสกในเรื่องมารักกัน โอลิ... 2 minutes ago **LIVE**

Nonthanun Sudlapa about an hour ago

Tanjai TJ Kongyuen 12 minutes ago

See More...

Peerasan Buranasanti and 2 others

Your Page

MIDAS Software

Messages Notifications 1

Chat (124)

Practice : Buy Travel Insurance

https://travel.cigna.co.th/CIMS/?gclid=Cj0KCQiA6dLgBRDoARIsAJgoM4vYCDpH8Fz0rZ6lakQZm4m9WZUyA0Dd2fBE-ec3VNNjwIPg0FOGg6laAgbIEALw_wcB...

ເຖິງກັບເສັກນໍາ ຄ່າດາມພບບ່ອຍ ຕິດຕ່ອສັກນໍາ

Cigna

ຄົ້ນຫາ

ສອບຕາມຜັດຕັບທີ່ໄດ້. 0-2099-3999

ແພນປະກັນ ເຄລືດ(ໄໝ)ລັບ
ຍກຮະດັບຄວາມພົດ ບ້າວແລະກົງກຣນ ອຸນຍົດແລລູກຄ້າ ເຊີກເບື້ຍປະກັນ

ເຊື້ອປະກັນທີ່ນີ້

ແພນປະກັນກ້າຍກາຣເດີນກາງຕ່າງປະເທດ

ວິນເຮັນເຕັນກາງ ເຊີກເພນປະກັນ ຂົ້ນມູສພູ່ຂອງເຕາປະກັນ ບໍາຮາເຈັນ

ແບບແພນປະກັນ

ຮາຍເກີຍວ່າ

ຮາຍເປີ
(ໄປຈ່າກົດຈຳນວນກັ້ນ)

ຈະຫມາຍປລາຍກາງ

ເລືອກໄດ້ສູງສຸດ 10 ປະເທດ

Leave a Message

Practice : Book Hotel

← → C Booking.com BV [NL] | https://www.booking.com/index.th.html?label=gen173nr-1FCAEoggI46AdIM1gEaN0BiAEBmAEmuAExyAEM2AEB6AEB-AELiAIB... ⌂ ☆ ⌂ | ⌂ | ⌂ | ⌂

Booking.com

THB ลงทะเบียนที่พักของท่าน

สมัคร เว็บสู่ระบบ

ที่พัก ตัวเครื่องบิน บริการรับยนต์เช่า บริการแท็กซี่สนา�บิน

ค้นหาข้อเสนอสำหรับทุกฤดูกาล
ดังแต่บ้านพักตามอากาศและสนับสนุนไปจนถึงห้องชุดสุดอึดใจกลางเมือง

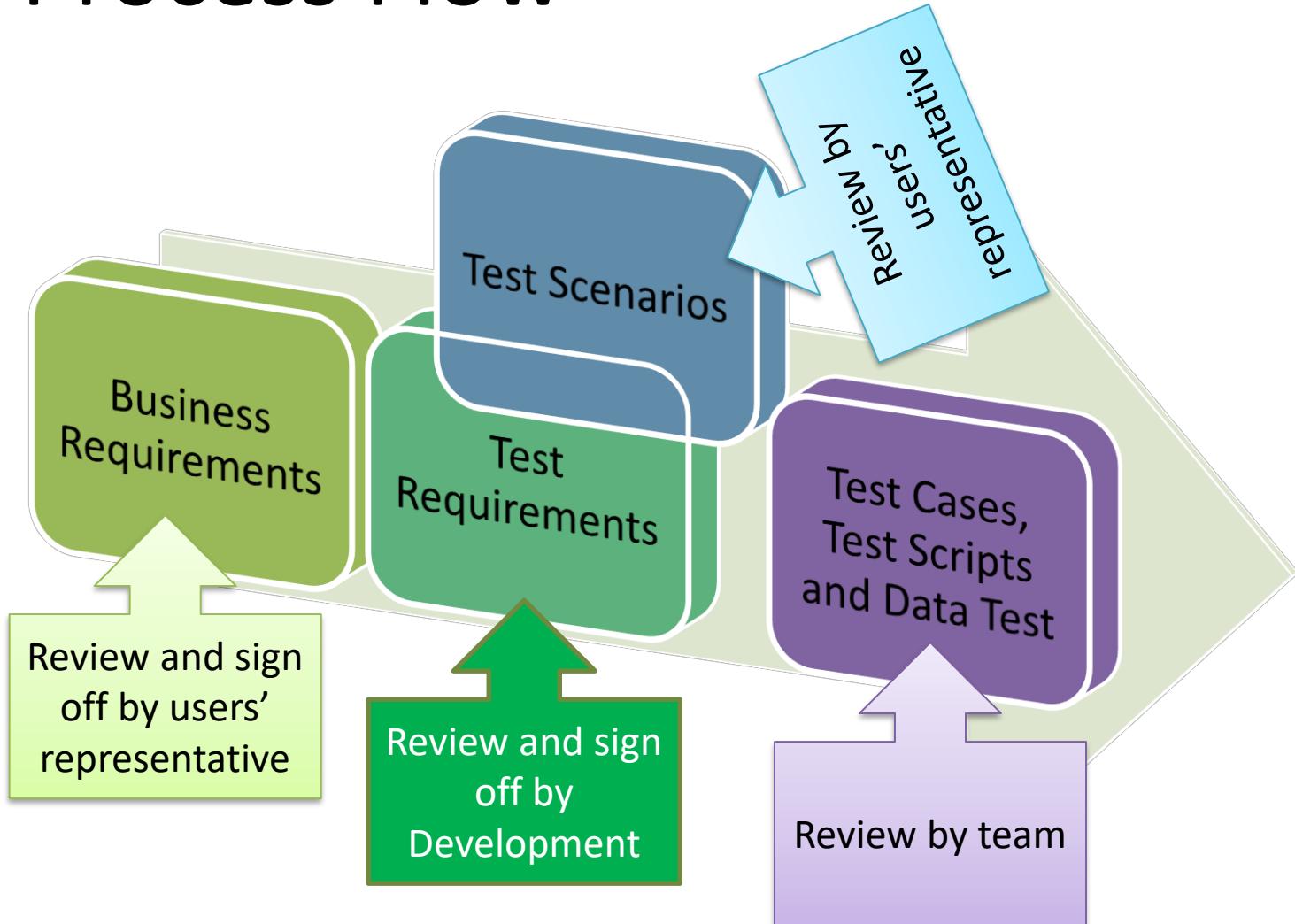
ทำงานจะไปที่ไหน? เช็คอิน — เช็คเอาท์ ผู้ใหญ่ 2 • 0 เด็ก ค้นหา

เดินทางไปทำงาน

ข้อเสนอช่วงเทศกาลปีใหม่ X

ข้อเสนอช่วงตรุษจีน X

Process Flow



9. TEST EXECUTION AND CAPTURE DEFECTS

Test Execution

- Execute the test cases
- Logging Defects
- Re-execute test cases after every change
 - **Automate as much as possible**
 - For instance, after each refactoring
- Regression testing
 - Testing that everything that used to work **still works after** changes are made to the system
 - rerunning test cases from existing test suites to build confidence that software changes have no unintended side-effects

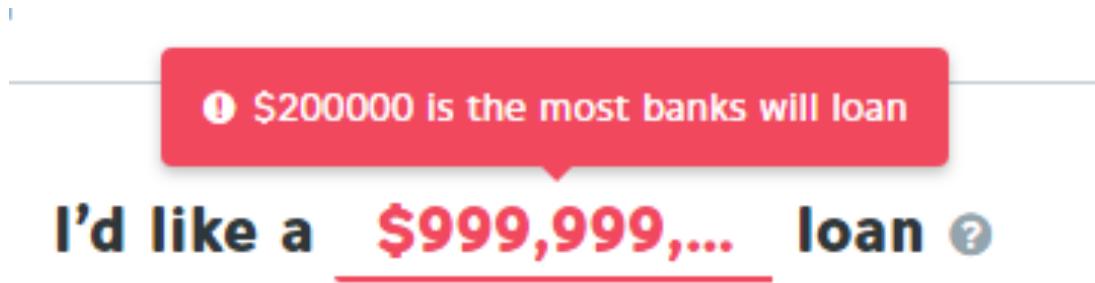
Test Case : Login with valid account

Test Scenario	Test Case	Pre Conditions	Test Step	Test Data	Expected Result	Actual Results	Pass/Fail
Check Login Functionality	Check response on Entering valid Agent Name & Password	Flight Reservation Application must be installed	<ol style="list-style-type: none">1. Launch Application2. Enter Agent Name3. Enter Password4. Click OK button	Agent Name : guru99 Password : MERCURY	Login must be successful.	Login Successful	Pass

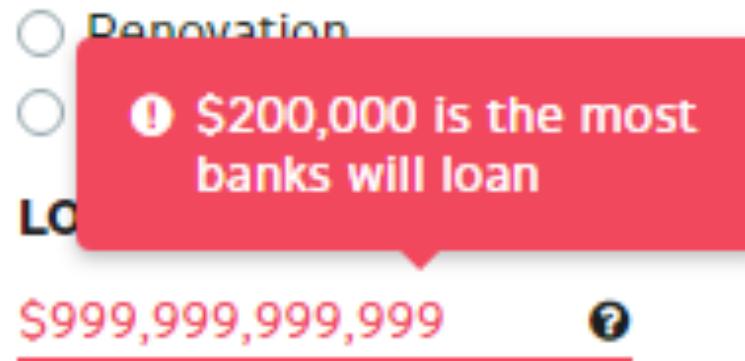
Class Practices : Is it defect?

Message Format

<https://www.gobear.com/sg/personal-loan>



<https://www.gobear.com/sg/personal-loan/quote-online?tenure=3&loanAmount=10000&income=3000&purpose=8b9ceabf-f4bd-4b83-a488-e8d61347c44c>



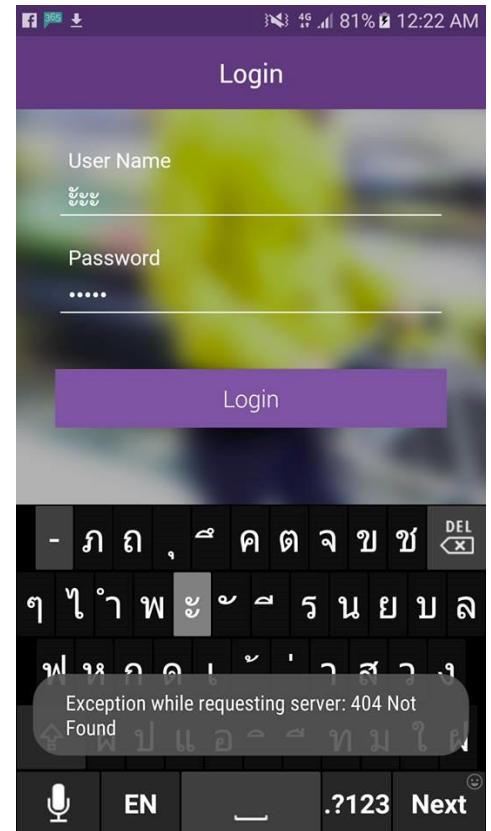
Calendar

SUN	MON	TUE	WED	THU	FRI	SAT
1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28
29	30	31				

					1	2	3
4	5	6	7	8	9	10	
11	12	13	14	15	16	17	
18	19	20	21	22	23	24	
25	26	27	28	29	30	31	

SUN	MON	TUE	WED	THU	FRI	SAT
1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28
29	30	31				

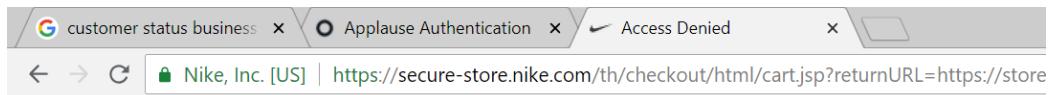
Sunday	Monday	Tuesday	Wednesday	Thursday	Friday	Saturday
26	27	28	1	2	3	4
5	6	7	8	9	10	11
12	13	14	15	16	17	18
19	20	21	22	23	24	25
26	27	28	29	30	31	1



Link to page

:(
Oops.
500 Internal Server Error

An error occurred while processing this request.



Access Denied

You don't have permission to access "http://secure-store.nike.com/th/checkout/html/cart.jsp?" on this server.

Reference #18.e2d613a.1536588792.1feabc0

Error: 404 - Article not found

You may not be able to visit this page because of:

1. an out-of-date bookmark/favourite
2. a search engine that has an out-of-date listing for this site
3. a mistyped address
4. you have no access to this page
5. The requested resource was not found.
6. An error has occurred while processing your request.

Something went wrong

Try that again, and if it still doesn't work, let us know.

Our status page is currently reporting a status of [All Systems Operational](#).

[VIEW STATUS PAGE](#)[LET US KNOW](#)

GUI

PRIMARY MEMBER NUMBER: 5503016004440

Members

test Test (primary member)

[view membership details](#)

STANDINGS

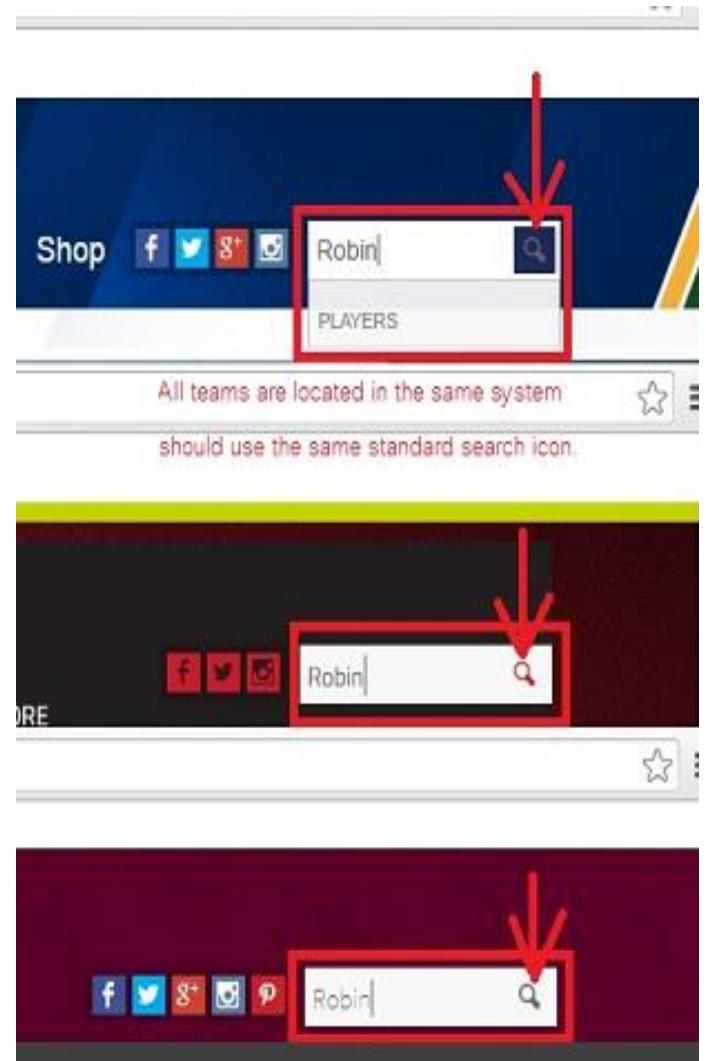
CATEGORY

Streaks and Last 10 ▾

2015-2016 CONFERENCE REGULAR SEASON STANDINGS - STREAKS AND LAST 10

EASTERN CONFERENCE

Eastern	W-L	CURRENT STRK	HOME STRK	ROAD STRK	WIN STRK	LOSS STRK	L 10	L 10 HOME	L 10 ROAD
---------	-----	--------------	-----------	-----------	----------	-----------	------	-----------	-----------



GUI

Ninja Van @ Platinum

จุดรับพัสดุนินจา ? 



กลับ

Address: 222 Phetchaburi Rd,
Khwaeng Thanon Phetchaburi,
Khet Ratchathewi, Bangkok
10400

Address: 222 Phetchaburi Rd,
Khwaeng Thanon Phetchaburi,
Khet Ratchathewi, Bangkok
10400

เวลาทำการ

Monday 08:00 AM-08:00 PM
Tuesday 08:00 AM-08:00 PM
Wednesday 08:00 AM-08:00
PM
Thursday 08:00 AM-08:00 PM
Friday 08:00 AM-08:00 PM
Saturday 08:00 AM-08:00 PM
Sunday 08:00 AM-08:00 PM

Feature

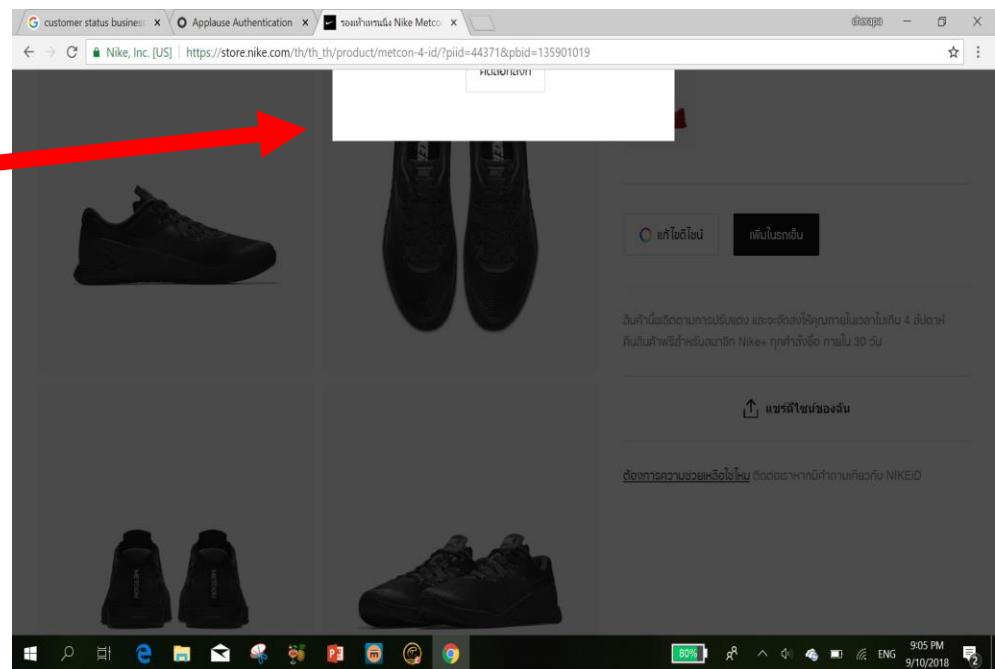
แชร์ดีไซน์ของคุณ

รองเท้า Nike Metcon 4 iD

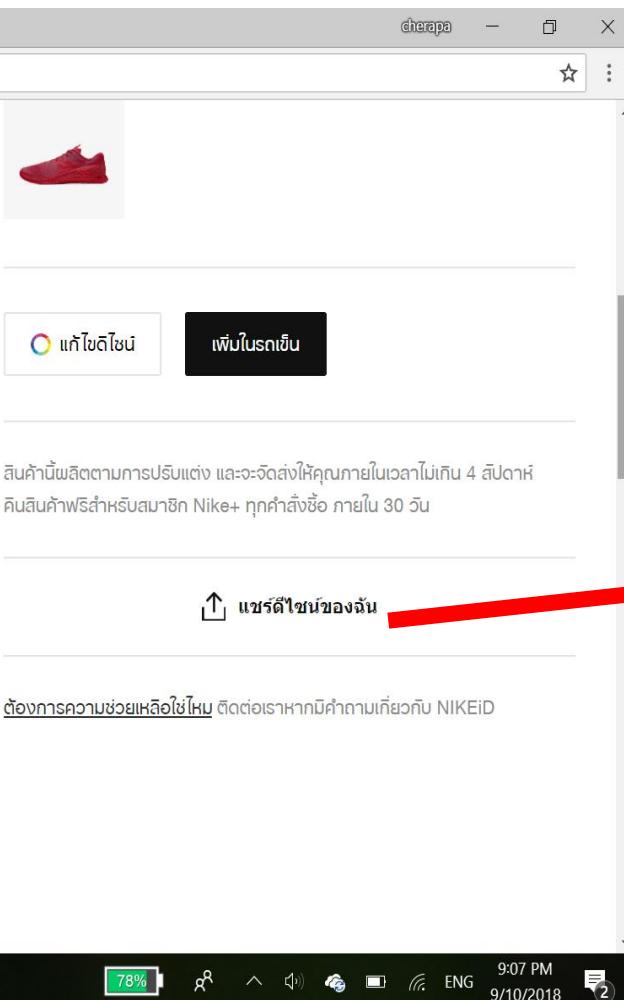


คลิกลอกลิ้งก์

Click ได้ แต่ไม่ copy link ให้



35901019



Search

คำนวณค่าจัดส่ง

จันทบุรี

กรุงเทพมหานคร

คำนวณ

น้ำหนักพัสดุ

การจัดส่งแบบวันถัดไป

การจัดส่งแบบปกติ (1-3 วันทำการ)

1 kg

฿35

฿33

5 kg

฿55

฿53

10 kg

฿85

฿83

ต้นทาง (ผู้ส่ง)
กรุงเทพมหานคร

ปลายทาง (ผู้รับ)
กรุงเทพมหานคร

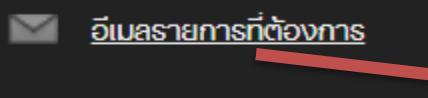
กลับ

Share Feature

X

แชร์ผลิตภัณฑ์ของคุณ

ถ้ามีความคิดเห็นจากเพื่อน หรือบอกรับชมภาพเช่าซื้อว่าดีจริงๆ แล้วคุณต้องการอะไร



เลือก
ทั้งหมด / ไม่

บันทึกเพิ่ม



10/9/2018

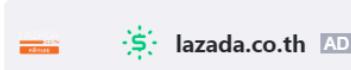
จาก : cherapa@yahoo.com

cherapaw@gmail.com

เพิ่มข้อความ

ลองดูสโตร์ของ Nike ที่ฉันบันทึกไว้ในรายการที่ต้องการ

Today



bosny ขาวชีลิโคน 300 มล. รุ่น B330 (สีใส) ติดไม...

Lazada - #1 Online Shopping



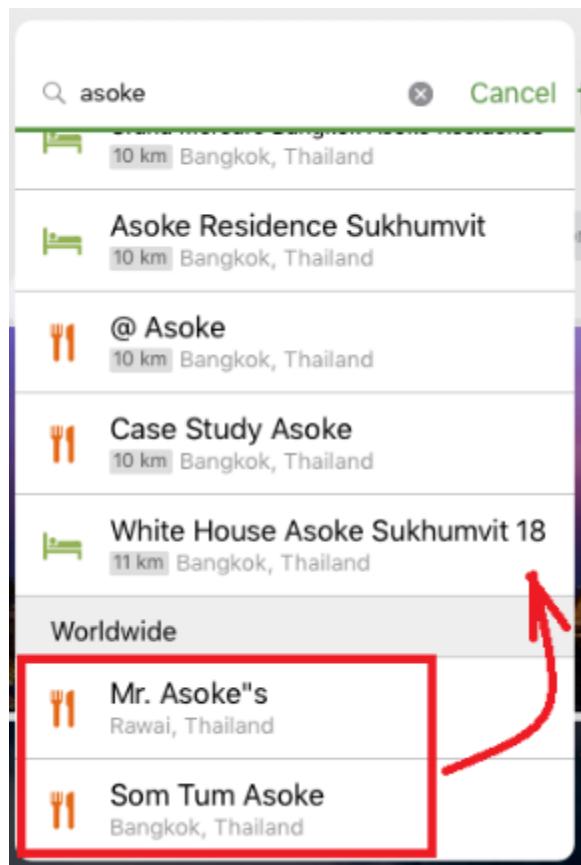
Facebook

Weekly Facebook Page update for RayMon's Garden See

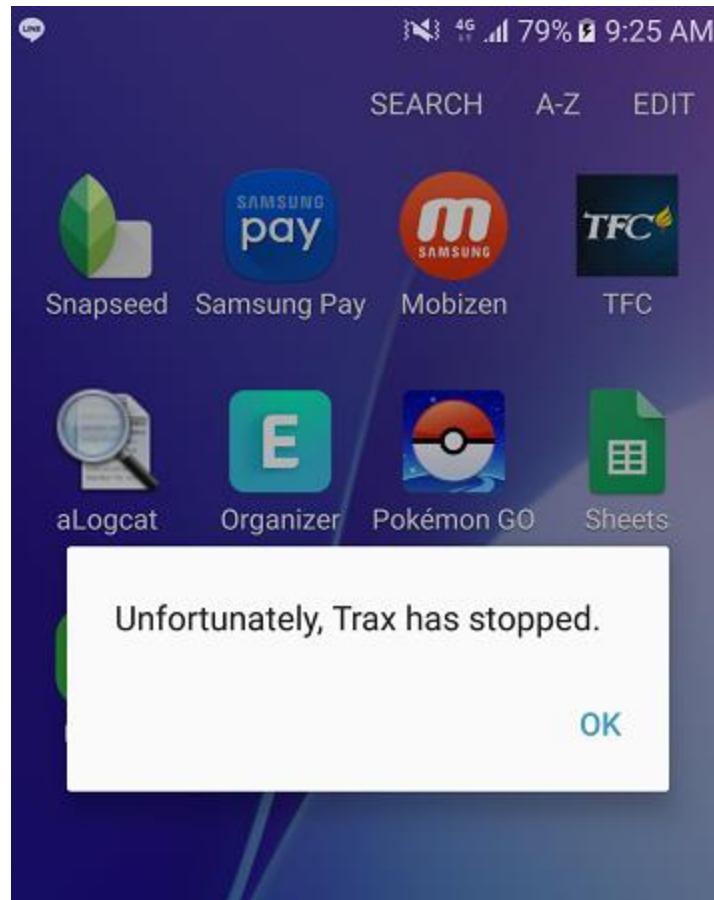
ยกเลิก

ลบ

Display Information



Terminated Application



Defect Log

- A log or database of all defects that were uncovered during the testing and maintenance phase of development. It categorizes defects into severity and similarity in an attempt to identify areas requiring special attention.

General Information

1. **Project Name:** The name assigned to the project.
2. **Project ID:** The ID number for the project.
3. **System Name:** The name of the system under test.
4. **System Version:** The version number for the system under test.
5. **Patch #:** The identifier assigned to the patch.
6. **Recorder:** The person responsible for submitting, tracking, and closing defects in the Test Defect Log.
7. **Test Category:** Nature of the testing performed. Place an X in the space provided beside the appropriate category.

Defect Information

8. **Defect ID:** A unique, sequential number used to identify the defect.
9. **Test Cycle:** The end-to-end execution of tests, per phase. Multiple tests may be executed during each phase. Tests may be repeated in subsequent cycles.
10. **Test Case ID:** The reference number of the specific test case or test script.
11. **Description:** A brief description of the problem.
12. **Reported by:** The name of the person who discovered the defect.
13. **Date Reported:** The date the defect was reported.
14. **Reproduce:** This defect is happened very time testing. Tester found the pattern to make the defect occurred.
15. **Severity:** The classification of defects within a system examines both the severity and priority of the defect. **Severity** is a measure of how great the impact is on the user's ability to complete the documented actions within the system.

- 16. Priority:** A priority set up to fix the defect when it was occurred.
- 17. Defect Codes:** CM=Configuration Management, CO=Coding, CS=Coding Standards, DC=Documentation Content, DE=Design, DP=Documentation Presentation, IA=Integration Agreement, PE=Performance, RE=Requirements, TR=Traceability, TP=Test Plan. See the Detailed Defect Report Instructions for defect code definitions.
- 18. Status:** A state through which a defect passes from identification to closure.
- 19. Resolution:** The resolution is the solution identified for the defect. There are four possible solutions: duplicate, enhancement request, fixed, fixed indirectly, and functions as designed. If needed, the description field can be used to record comments on the nature of the resolution.
- 20. Date Resolved:** Date the defect is resolved.

Sample Template

Test Defect Log

Fill in Defect Log

Sample Test Defect Log

1. Project Name			Bar Code Medication Administration			2. Project ID			XXXX				
3. System Name			Bar Code Medication Administration			4. System Version #			3.0				
5. Patch #			N/A			6. Recorder			Alan Allenby				
7. Test Category <input checked="" type="checkbox"/> System Test <input type="checkbox"/> UAT <input type="checkbox"/> IOC Testing <input type="checkbox"/> Iteration Test <input type="checkbox"/> Other													
8. ID	9. Test Cycle	10. Test Case ID	11. Description	12. Reported By	13. Date Reported	14. Reproduce	15. Severity	16. Priority	17. Defect Code	18. Status	19. Resolution	20. Date Resolved	
1.	3	3.1.4.4.3	BCMA failed to confirm administered dose and accurately document the selected fractional dose. ReqPro reference: SPEC 18-29	HH	04/12/03	Yes	High	High	RSD	Open	Work around exists.		
2.	3	3.1.4.1.1	Patient Transfer notification sent and changes to the VDL's last action column occurred. Expected result – no notification and no changes. ReqPro reference: SPEC 03-17	BW	04/12/03	Yes	Critical	High	RSD	Open			

Priority

- Help in defect management
 - Priority first will be set by the tester and later on can be updated by the authorized person
 - Priority is set before test execution. It will help in measurement such as high priority must be fixed in 4 hours.
 - The sample of priority;
 - High : It must be fixed with in 4 hours even thought the test cannot continue
 - Medium : It must be fixed with in 8 hours. It's value is not correct and is not effected to the next test execution. Tester can modify the value in the system by using another feature.
 - Low : It is cosmetic error such as small character, incorrect color. It must be fixed before UAT.

Severity

- Help in defect management in term of business impact
- The sample of Severity;
 - Critical>Show Stopper
 - Major/High
 - Average/Medium/Moderate
 - Minor/Low



Minor



Moderate



Major



Critical

Defect Stages

- Help in tracking
 - They are set and agreed within the working team.
 - They may automatic updated or manually updated depend on the tool
- The different states of a bug can be summarized as follows:
 1. New
 2. Assign
 3. Fix
 4. Test
 5. Reopened
 6. Rejected (New requirements, not an error, duplicated)
 7. Deferred
 8. Closed

Defect Management

- Thorough understanding
- Adhere to processes
- Verify defects before distribute
- Capture Screenshot and steps
- Raise every bugs in the defect tracking tool
- Always use the facts
- Keep track of impact of bugs and associate to the risks
- Some defects may not resolve, prepare for these
- Never close the defects because of your feeling or not finding by developer

Defect Tracking Tools

- <http://www.testingfaqs.org/t-track.html>
- For example
 - Open source
 - [Mantis \(freeware\)](#)
 - [Bugzilla \(freeware\)](#)
 - Etc.
 - Commercial
 - [ClearQuest \(IBM Rational\)](#)
 - [TestDirector](#)
 - SpiraPlan®
(<http://www.inflectra.com/SpiraPlan/Highlights.aspx?feature=Defect-Tracking>)
 - Etc.

10. TEST MEASUREMENT

Why Measure?

- “Software bugs cost the U.S. economy an estimated \$59.5 billion per year. An estimated \$22.2 billion could be eliminated by improved testing that enables earlier and more effective identification and removal of defects.”
- - US Department of Commerce (NIST)

Why Measure?

- It is often said, “You cannot improve what you cannot measure.”

„When you can measure what you are speaking about, and express it in numbers, you know something about it, but when you cannot measure it, when you can not express it in numbers, then your knowledge is of a measure and unsatisfactory kind.“

from Lord Kelvin
British physicist, 1882

Definition

Test Metrics:

- Are a standard of measurement.
- Gauge the effectiveness and efficiency of several software development activities.
- Are gathered and interpreted throughout the test effort.
- Provide an objective measurement of the success of a software project.

Metrics Philosophy

- Keep it simple
- Make it Meaningful
- Track it
- Use it
 - When tracked and used properly, test metrics can aid in software development process improvement by providing pragmatic & objective evidence of process change initiatives.

Base Metrics (Direct Measurement)

- ▶ Raw data gathered by Test Analysts
- ▶ Tracked throughout test effort
- ▶ Used to provide project status and evaluations/feedback
- ▶ Example
 - ▶ # Test Cases
 - ▶ # Executed
 - ▶ # Passed
 - ▶ # Failed
 - ▶ # Under Investigation

Base Metrics and Testing Phases

Metric	Testing Phase
# Test Cases	Test Development
# Executed	Test Execution
# Passed	Test Execution
# Failed	Test Execution
# Under Investigation	Test Development
# Blocked	Test Development/Execution
# 1 st Run Failures	Test Execution
# Re-Executed	Regression
Total Executions	Test Reporting
Total Passes	Test Reporting
Total Failures	Test Reporting

Calculate Metrics

- Convert Base metrics data into useful information
- Prepare by Test lead and used to track project's progress
- Leads to significant improvement in the overall SDLC

Calculated Metrics and Phases

- Normally created at Test Reporting or Post Test Analysis Phase:
 - % of Test cases Passed
 - % of Test Coverage
 - % of Defects corrected
 - % of Test cases Blocked
 - % of Rework
 - % of Test Effectiveness
 - % 1st Run Fail rate

% 1st Run Failures

- The percentage of executed test cases that failed on their first execution.
- Used to determine the effectiveness of the analysis and development process.
Comparing this metric across projects shows how process changes have impacted the quality of the product at the end of the development phase.

Test case Defect Density

- The number of errors found in test cases v/s test cases developed and executed
 - $\rightarrow (\text{Defective Test cases} / \text{Total Test cases}) * 100$
- Example : Total no of test cases developed is 1360, total test cases executed is 1280, total no of test cases passed is 1065, total no of test scripts failed is 215 So
 - Test case Defect Density is : $(215 \times 100) / 1280 = 16.8\%$
- The 16.8 % value can also be called as Test Case Efficiency % which depends upon the total number of Test cases which found defects

Calculated Formula

Value	How (x100)
% Complete	#Passed/Total # of TCs
% of Test cases Passed	#Passed/Total # of TCs
% of Test Coverage	#Executed/Total # of TCs
% of Defects corrected	(Total Failures-#Failed)/Total Failures
% of Test cases Blocked	#Blocked/Total # of TCs
% of Rework	(Total Executions-#Executed-#Re-executed)/#Re-executed
% of Test Effectiveness	#Failed/#Executed
% 1 st Run Fail rate	1 st Run Failures/#Executed

Sample Run Log

Sample System Test

TC ID	Run Date	Actual Results	Run Status			Current Status	# of Runs
SST-001	01/01/04	Actual results met expected results.	P			P	1
SST-002	01/01/04	Sample failure	F			F	1
SST-003	01/02/04	Sample multiple failures	F	F	P	P	3
SST-004	01/02/04	Actual results met expected results.	P			P	1
SST-005	01/02/04	Actual results met expected results.	P			P	1
SST-006	01/03/04	Sample Under Investigation	U			U	1
SST-007	01/03/04	Actual results met expected results.	P			P	1
SST-008		Sample Blocked	B			B	0
SST-009		Sample Blocked	B			B	0
SST-010	01/03/04	Actual results met expected results.	P			P	1
SST-011	01/03/04	Actual results met expected results.	P			P	1
SST-012	01/03/04	Actual results met expected results.	P			P	1
SST-013	01/03/04	Actual results met expected results.	P			P	1
SST-014	01/03/04	Actual results met expected results.	P			P	1
SST-015	01/03/04	Actual results met expected results.	P			P	1
SST-016							0
SST-017							0
SST-018							0
SST-019							0
SST-020							0

Sample Run Log

Base Metrics		Calculated Metrics	
Metric	Value	Metric	Value
Total # of TCs	100	% Complete	11.0%
# Executed	13	% Test Coverage	13.0%
# Passed	11	% TCs Passed	84.6%
# Failed	1	% TCs Blocked	2.0%
# UI	1	% 1st Run Failures	15.4%
# Blocked	2	% Failures	20.0%
# Unexecuted	87	% Defects Corrected	66.7%
# Re-executed	1	% Rework	100.0%
Total Executions	15		
Total Passes	11		
Total Failures	3		
1st Run Failures	2		

11. TEST AUTOMATION

Automated testing tool

- Automated software testing is an alternative to manual testing, where software tools, not human testers, execute pre-scripted tests on a software application before it is released into production.
 - Able to playback pre-recorded and predefined actions
 - Compare the results to the expected behavior
 - Report the success or failure

Why automated?

- A product as designed cannot technologically be manufactured manually
- Boring tedious work that humans will not do or repetitive and dangerous work that can lead to human injury
- To replicate processes consistently and with better precision or quality than a human

12. TEST ESTIMATION AND PLANNING

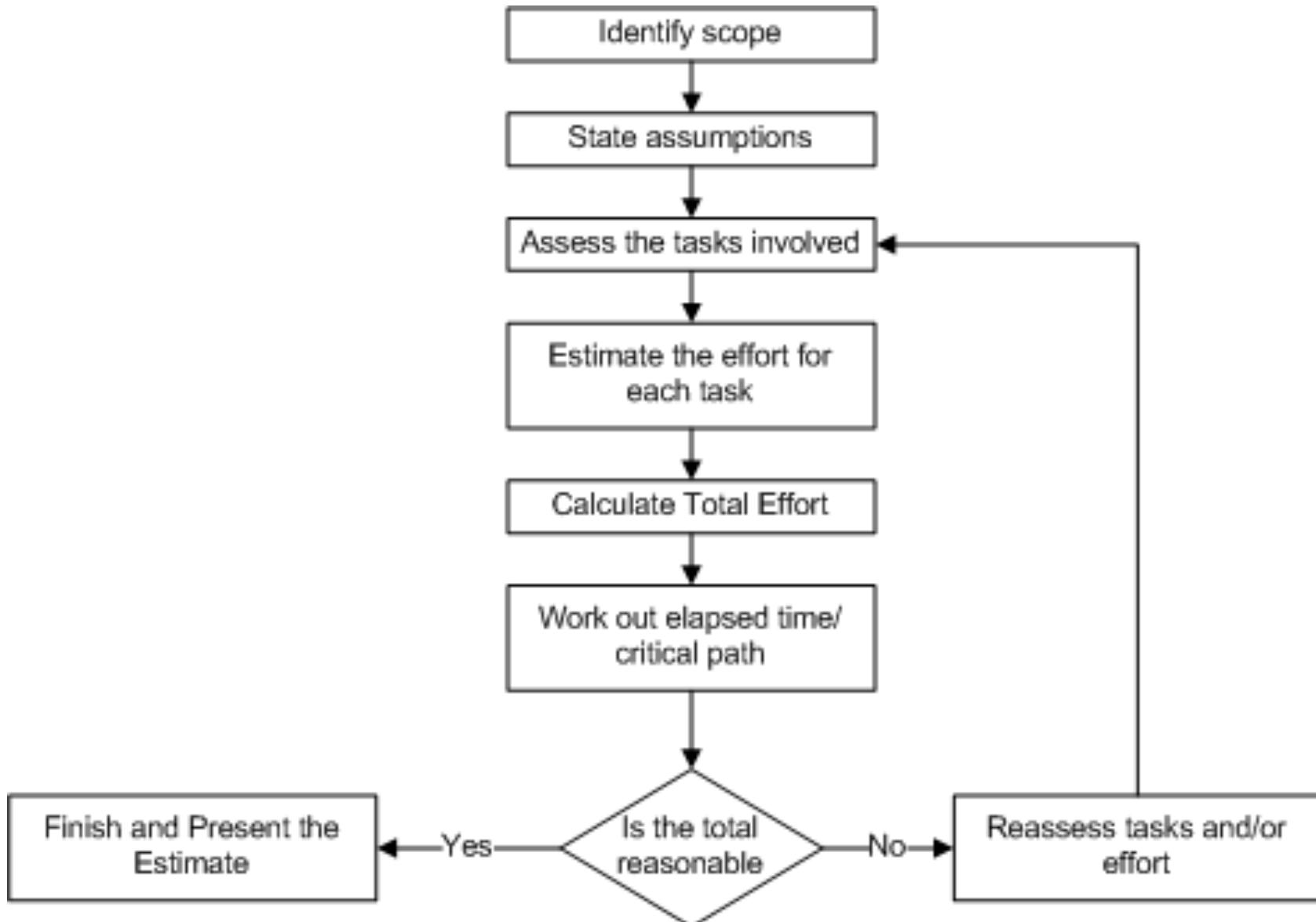
Test Estimation

- Test Estimation is the estimation of the testing size, testing effort, testing cost and testing schedule for a specified software testing project in a specified environment using defined methods, tools and techniques.

Estimation for Testing

- Questions to be asked
 - Are the requirements finalized?
 - If not then, How frequency the requirements change?
 - What are the Types of Testing required?
 - Are Test Assets already available?
 - If Test Assets are available, What Stage they are in?
 - Is the Infrastructure for Testing ready?
 - Have the Assumptions/Risks been documented?
 - Are the Responsibilities clear?

Steps involved in Estimation



Functional Testing Estimation

- Number of test cases required is based on:
 - Test Requirements := Testing all functions and features in the Requirements
 - Test Scenarios := Including an appropriate number of ALAC (Act Like A Customer) tests including:
 - Do it wrong
 - Use wrong or illegal combination of inputs
 - Don't do enough
 - Do nothing
 - Do too much
 - Achieving some test coverage goal
 - Achieving a software reliability goal

Considerations in Test Estimation

- Test Complexity – It is better to have many small tests than a few large ones.
- Different Platforms – Does testing need to be modified for different platforms, operating systems, etc.
- Automated or Manual Tests – Will automated tests be developed? Automated tests take more time to create but do not require human intervention to run.

Estimating Tests Required

SRS Reference	Test Requirements	Estimated Number of Tests Required	Notes
4.1.1	2	3	2 positive and 1 negative test
4.1.2	1	2	2 automated tests
4.1.3	2	4	4 manual tests
4.1.4	2	5	1 boundary condition, 2 error conditions, 2 usability tests
....			
Total		165	

Estimated Test Development Time

Estimated Number of Tests: 165

Average Test Development Time: 3.5
(person-hours/test)

Estimated Test Development Time: 577.5
(person-hours)

Estimated Test Execution Time

Estimated Number of Tests: 165

Average Test Execution Time: 1.5
(person-hours/test)

Estimated Test Execution Time: 247.5
(person-hours)

Estimated Regression Testing (50%): 123.75
(person-hours)

Total Estimated Test Execution Time: 371.25
(person-hours)

3 Points Estimation Techniques

= Best Case + Worst Case + (4 * Normal Case) / 6

- **Sum up the –**
 - Expected times to get Expected effort estimate for the project
 - Best-Case times to obtain best-case effort estimate
 - Worst-Case times to obtain worst-case effort estimate
 - Normal-Case times to obtain normal-case effort estimate

3 Points Estimated Example

Test Case Id	Test Case Description	Effort in PH			
		Best Case	Worst Case	Normal Case	Expected
US1	Setup Test Environment				
US1.1	Check Test Environment	1	2	1.5	1.500
US2	install Screen recorder	0.75	1.5	1	1.042
US1.2	Ensure Defect Reporting mechanism	1.25	3	2	2.042
UI1	Login Screen on IE				0.000
UI1.1	Correct Login	0.05	0.2	0.1	0.108
UI2	Wrong id and Correct Passsword	0.07	0.2	0.1	0.112
UI1.2	Correct Id and wrong Password	0.07	0.2	0.1	0.112
UI3	Forgot Password Functionality	0.15	0.3	0.2	0.208
UF2	Login Screen on Firefox				0.000
UF2.1	Correct Login	0.05	0.2	0.1	0.108
UF3	Wrong id and Correct Passsword	0.07	0.2	0.1	0.112
UF2.2	Correct Id and wrong Password	0.07	0.2	0.1	0.112
UF4	Forgot Password Functionality	0.15	0.3	0.2	0.208
	Total Effort Estimate	3.680	8.300	5.500	5.663

Test Planning

- The Test Plan – defines the scope of the work to be performed
- The Test Procedure – a container document that holds all of the individual tests (test scripts) that are to be executed
- The Test Report – documents what occurred when the test scripts were run

Test Plan

- Questions to be answered:
 - How many tests are needed?
 - How long will it take to develop those tests?
 - How long will it take to execute those tests?
- Topics to be addressed:
 - Test estimation
 - Test development and informal validation
 - Validation readiness review and formal validation
 - Test completion criteria

Test Plan Contents

- Version History
- INTRODUCTION
- TEST SCOPE AND STRATEGY
- TEST APPROACH
- TESTING CRITERIA
- TESTING PROCESS AND DELIVERABLES
- TESTING RESOURCES AND REQUIREMENTS
- RISKS AND DEPENDENCIES
- Approve Section

How to Write a Test Plan

1. **Write an introduction.** An introduction includes a general description and schedule of a test, as well as any related documents.
2. **For Scope and Strategy, write a section on what you are going to test and not test.** List what new aspects you will be and will not be testing and what old aspects you will be re-testing.
3. **Write testing types and Techniques on what you are going to use during testing.** Apply unit, integration, system, and UAT testing in your test plan.
4. **Write testing process and a list of documents that will be produced during testing.**
5. **Write a section on required resources.** This section describes all of the resources needed to complete the testing, including hardware, software, testing tools, and staff.
6. **Identify risks and dependencies that will effect testers to follow test plan.**

Main Test Activities

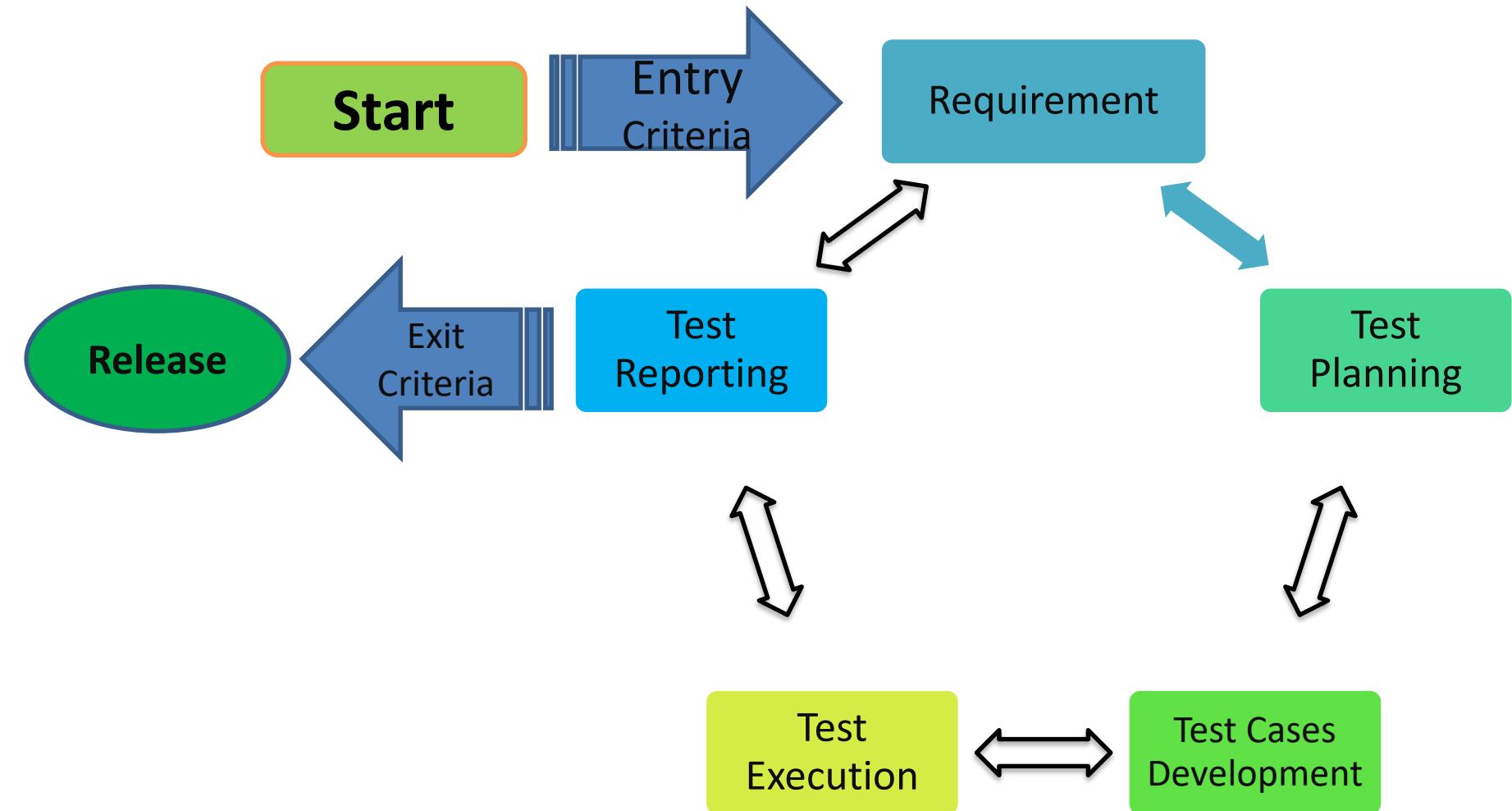
- Testing is not just running tests, but also:
 - Choosing test requirements and scenarios
 - Estimation of testing activities
 - Planning and control
 - Designing and executing test cases
 - Checking results
 - Evaluating exit criteria
 - Reporting on the testing process and system under test
 - Finalizing or completing closure activities

13. TEST PREPARATION AND PROCESS SUMMARY

Are you ready?

- Know the testing process and templates
- Know the business process under test
- Understand requirements
- Test Environment is ready
 - Define deliverables version
- Tester is ready
 - Skill and Technique
 - Train
- Deliverables are ready
 - Pass the entry criteria of testing agreement

Testing Processes



Testing Process

- Entry Criteria
 - Requirements analysis: Testing should begin in the requirements phase of the software development life cycle. During the design phase, testers work with developers in determining what aspects of a design are testable and with what parameters those tests work.
- Exit Criteria
 - Test Closure: Once the test meets the exit criteria, the activities such as capturing the key outputs, lessons learned, results, logs, documents related to the project are archived and used as a reference for future projects.

14. QUALITY ASSURANCE AND QUALITY CONTROL

QA & QC

- Quality Assurance: A set of activities designed to ensure that the development and/or maintenance process is adequate to ensure a system will meet its objectives.
- Quality Control: A set of activities designed to evaluate a developed work product.
- In **CMMI SQA\SQC is defined as Process and Product Quality Assurance.**
 - CMMI is an approach to process improvement, in which SQA\SQC play a major but not exclusive role.

Quality Assurance (QA)

- QA is planned and systematic way to evaluate quality of process used to produce a quality product.
- The goal of a QA is to provide assurance that a product is meeting customer's quality expectations.
- QA deals with how to prevent bugs from occurring in a product being developed.
- Software Quality Assurance Engineer's main responsibility is to create and implement methods and standards to improve development process.
- QA is associated with activities like measuring the quality of process used to develop a product, process improvement and defect prevention. It consists of auditing and reporting procedures related to development and testing.

Quality Control (QC)

- Quality control name comes from manufacturing industry where QC inspector evaluate sample products taken from the manufacturing line, test them & if the products fail the test they have the authority to shut down the whole production line.
 - QC in software industry is evaluating software product, find the defects & suggest improvements.
 - QC implements the process established by QA.
 - Software tester is responsible for QC.
 - If required, personnel involve in QC have to carry out QA tasks as well.

15. BASIC QUESTIONS IN TESTING



Common Questions

- "When do you stop testing?",
- "Have you tested thoroughly?",
- "How can you prevent defects?" and
- the always popular "Can we ship yet?"

When do you stop testing?

- Reach the exit criteria
 - All test cases were executed and pass.
 - There is no high severity defects.
 - All defects are close.
- Time is running out. We have to ship the product now.
- No budget left, we have to stop testing and release the product.
- Project Manager told us to stop.
- The product is not pass the minimum quality requirements, such as show stopper immediately after deployment.

Have you tested thoroughly?

- The correct release is installed in the test environment.
- All functions have test cases associated to them. See traceability metric (RTM).
- Test cases pass team review and get feedback. See review report.
- Finally, we execute all of them and they are pass. See test results reporting.
- All risks associated to the critical functions were removed.

How can you prevent defects?

- Use Causal Analysis and Resolution (CAR) to detect the root cause of the defects and find solution
- Fully understanding the scope of the software development project
- Getting involved at the earliest possible stage in the development cycle
- Reviewing the project plan for development and offering feedback and
- Maintaining a presence in the development process before the first release to test.

Can we ship yet?

- All test cases that we defined are conformed to User Acceptance Criteria.
 - We believe that after we executed all test cases and no defect found we can ship it.
- The defects found are under control limited.
- We already prepare the solution for any incidents. Make sure that all concerned staffs are well plan.
- The user accept the product ship with some defects. We have commitment to fix them.
 - Keep your promise in this case.

What if there isn't enough time for testing?

- Here are some points to be considered when you are in such a situation:
 - Find out Important functionality is your project?
 - Find out High-risk module of the project?
 - Which functionality is most visible to the user?
 - Which functionality has the largest safety impact?
 - Which functionality has the largest financial impact on users?
 - Which aspects of the application are most important to the customer?
 - Which parts of the code are most complex, and thus most subject to errors?
 - Which parts of the application were developed in rush or panic mode?
 - What do the developers think are the highest-risk aspects of the application?
 - What kinds of problems would cause the worst publicity?
 - What kinds of problems would cause the most customer service complaints?
 - What kinds of tests could easily cover multiple functionalities?
- Considering these points you can greatly reduce the risk of project releasing under less time constraint.

16. IMPROVEMENT

KPI

- A key performance indicator(KPI) is a measure of performance, commonly used to help an organization define and evaluate how successful it is, typically in terms of making progress towards its long-term organizational goals.



Tester's KPI

- Be in the top three of bug issuing
- Find the highest severity bugs of the projects/team
- Prepare test cases and test data as plan
- Good estimation
- Test Execution as plan
- Etc.

Become a Better Tester

- Test for quality over quantity
- Practice and improve
- Learn from your own mistakes – and from others too
- Be objective and professional
- Don't be humble with software... think out of the box
- Question. Everything
- Think like the user
- Increase the effectiveness of bug reports
- Be Passionate!

Testing skills and knowledge

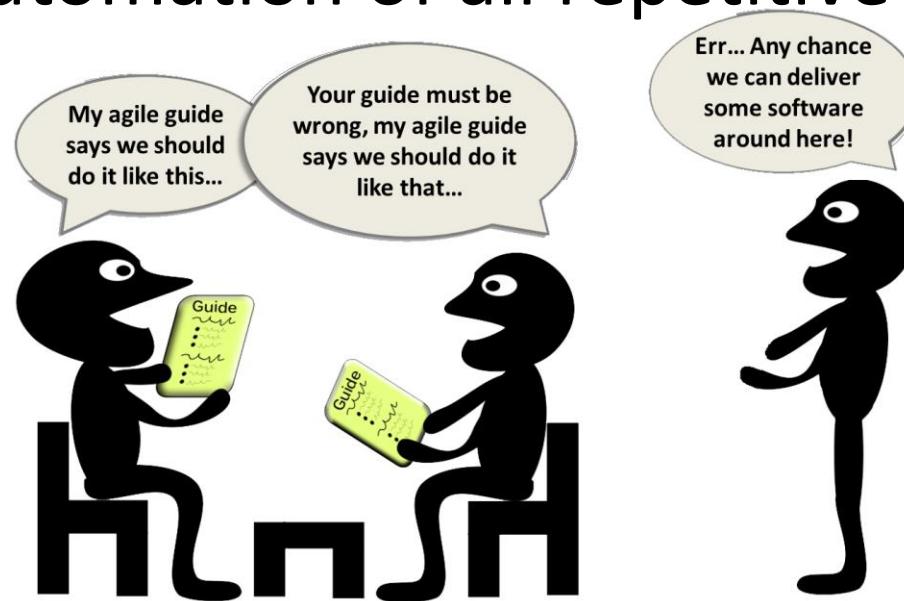
Test knowledge <ul style="list-style-type: none">▪ test principles▪ techniques▪ tools, etc.	IT knowledge <ul style="list-style-type: none">▪ software development▪ requirements▪ configuration management
Domain knowledge <ul style="list-style-type: none">▪ business process▪ user characteristics	Soft skills <ul style="list-style-type: none">▪ communication▪ critical mindset▪ presentation and reporting

Personal motivation model for software testers

<p><u>Assessment</u></p> <p>How do I rate myself as a tester? How do my peers rate me as a tester? Have I been successful as a tester? What is my contribution to my team? Who is a good role model?</p>	<p><u>Ends</u></p> <p>What do I want to achieve within the next 5 years? How do I define success? What do I need to achieve to feel that I have realized my true potential? What do I desire in my career as a tester? What is my true north?</p>
<p><u>Influencers</u></p> <p>Why did I become a software tester? Why am I good at what I am good at? What skills can I leverage to the max? What are my primary motivations for working towards improving my performance?</p>	<p><u>Means</u></p> <p>How do I get the results I want? Who can help or coach me? Do I need to learn new tools, techniques or skills? How can I run or walk towards the desired end state (my true north)?</p>

What testing principles should teams working in agile methodologies follow ideally?

- Early and continuous engagement
- Co-location of Development and Testing teams
- First Time Yield
- Parallel automation of all repetitive tasks



Testing certifications

- Certified Associate in Software Testing (CAST) offered by the Quality Assurance Institute (QAI)
- CATe offered by the International Institute for Software Testing
- Certified Manager in Software Testing (CMST) offered by the Quality Assurance Institute (QAI)
- Certified Software Tester (CSTE) offered by the Quality Assurance Institute (QAI)
- Certified Software Test Professional (CSTP) offered by the International Institute for Software Testing
- CSTP (TM) (Australian Version) offered by K. J. Ross & Associates
- ISEB offered by the Information Systems Examinations Board
- ISTQB Certified Tester, Foundation Level (CTFL) offered by the International Software Testing Qualification Board
- ISTQB Certified Tester, Advanced Level (CTAL) offered by the International Software Testing Qualification Board

17. TEST PLAN EXAMPLE

Test Plan Example

TEST PLAN

Version	Date	Description	By
1.0	20/9/2012	Initial Test Plan	Cherapa Wannasuk

PROJECT/SYSTEM/MODULE

ABC VISA Card withdrawal at ABC's ATMs

TEST SCOPE AND STRATEGY

The test is to ensure that all ABC's ATMs can operate the withdrawal process of ABC Visa and only for a valid Visa. The system must provide transaction logs and sufficient reports to ATM operator. The system must update accurate data and issue the right slip at the ATM. Every period the system should print the withdrawal amount in the customer statement along with charging amount, if any.

Tester will get involved early in the development process and prepare the test cases to development team.

In scope:

- All features defined in VISA Requirement document

Out scope:

- Link between ABC and VISA System
- All VISA reports stored in VISA System
- The validation of data exchange to VISA System

TEST APPROACH

The testing need real ATM machine to execute for the test environment. The testing will start with unit, integration and UAT. The performance test is required to ensure that the system can handle concurrent process.

During development process team must follow these processes.

- Organization development process and use standard templates
- Control version
- Follow Configuration Management Process
- Follow Change Control Process

During Test Execution tester will capture and report these information.

Measurement metrics:

- Number of Defect separated by Severity
- Number of Defects by phase
- Time used to fix the defects

Meeting and Reports

- Update test status every 2 weeks

TESTING CRITERIA

Criteria	Pass	Fail
Approval Criteria	All high priority test cases are executed and no high and medium severity defects occurred.	High and Medium severity defects are not fixed.

If any defects are found which seriously impact the test progress the test lead may choose to suspend testing.

The criteria which are considered for suspension or resumption are;

- [a] hardware / software not available at the time indicated in the project schedule
- [b] the build contains many serious defects which seriously prevent or limit testing progress
- [c] Assigned test resources are not available when needed by the test team

Resumption Criteria:

If testing is suspended, resumption will only occur when the problem(s) that caused the suspension have been resolved. When a critical defect is the cause of the suspension, the "FIX" must be verified by the testing team before testing is resumed.

TESTING PROCESS AND DELIVERABLES

Tester will follow the schedule defined in Project Management Plan. The table below shown the task that team will perform.

Tasks	Test Manager	Project Manager	Dev Team	Test Team	User
Acceptance test Documentation & Execution	X	X		X	X
System/Integration test Documentation & Exec.	X		X	X	
Unit test documentation & Execution	X		X	X	
System Design Reviews	X	X	X	X	X
Detail Design Reviews	X	X	X	X	
Test procedures and rules	X	X	X	X	
Screen & Report prototype reviews			X	X	X
Change Control and regression testing	X	X	X	X	X

During the testing process tester will develop these deliverables.

- Test Plan
- Test Cases and Script
- Defect Logs
- Test Results Report

TESTING RESOURCES AND REQUIREMENTS

To start testing, tester will need additional training from team to understand the requirements.

The test environment must be ready before start execution.

The VISA card must be sent to Tester in advanced and register in the VISA organization.

RISKS AND DEPENDENCIES

If integration network between ABC and VISA System is not ready before development starting time. The test execution period may be changed.

The Credit Approval System and Accounting system are in the development process now. If the development time is moved testers may not be available to join this project. If this situation happen the priority must be defined and discuss among project manager and top management.

Signature of PM/PL

Date

<https://starwest.techwell.com/program/virtual-conference>

The screenshot shows the homepage of the STARWEST Virtual Conference. At the top, there's a navigation bar with the TECHWELL logo, a search bar, and links for MORE EVENTS and TECHWELL LOGIN. Below the bar, the STARWEST logo is displayed, followed by the text "A TECHWELL EVENT". To the right, the event date "OCTOBER 1-6, 2017 | ANAHEIM, CA" is shown. A horizontal menu bar below the logo includes links for PROGRAM, SPEAKERS, TRAINING, EXPO, VENUE, and a prominent orange "REGISTER" button. Underneath the menu, a breadcrumb trail shows "Home / Program / Free STARWEST Virtual Conference". The main content area features a large orange header "Free STARWEST Virtual Conference" and a sub-section titled "Virtual Conference" with the text "Wednesday, October 4 | 8:30am–5:30pm PT and Thursday, October 5 | 8:30am–5:30pm PT". A large orange "REGISTER NOW" button is centered below this information. At the bottom, a paragraph describes the virtual conference as streaming live from Anaheim, California, featuring presentations and interviews straight to the desktop. A "Leave a message" button is visible on the right, and a small window for "flashplayer27pp_g....exe" is open at the bottom left.

TECHWELL™

MORE EVENTS ▾ TECHWELL LOGIN

STAR WEST
A TECHWELL EVENT

OCTOBER 1-6, 2017 | ANAHEIM, CA

PROGRAM SPEAKERS TRAINING EXPO VENUE **REGISTER**

Home / Program / Free STARWEST Virtual Conference

Free STARWEST Virtual Conference

Virtual Conference

Wednesday, October 4 | 8:30am–5:30pm PT
and Thursday, October 5 | 8:30am–5:30pm PT

REGISTER NOW

The STARWEST Virtual Conference will be streaming live on October 4-5 from Anaheim, California. The Virtual Conference features presentations and interviews straight to your desktop—including the latest testing solutions and networking opportunities. Watch keynote sessions and industry technical presentations as they are happening live at STARWEST. The STARWEST Virtual Conference features:

Leave a message

flashplayer27pp_g....exe

Show all X

Freelance Jobs

- www.bugfinders.com
- www.atest.com
- <https://www.usertesting.com/>
- <https://www.testbirds.com/>
- <https://www.trymyui.com/>
- <http://www.testerwork.com/>
- <https://testlio.com/>
- <https://usertest.io/Tester/Signup>
- <https://crowdtap.com/>

Question?
Question?
Cherapa@yahoo.com

