

การวิเคราะห์และบทสรุปในการเลือก

Open-source software project

โดย

633020556-0 นายนรบดี เดชขันธ์

633021009-4 นายอัครเดช วิทยาวุฒิรัตน์

เสนอ

อาจารย์ที่ผศ.ดร.ชิตสุธา สุ่มเล็ก

รายงานนี้เป็นส่วนหนึ่งของวิชาศึกษาวิชา SC313504 Software Quality Assurance

ภาคเรียนที่ 1 ปีการศึกษา 2565

ภาควิชาวิทยาการคอมพิวเตอร์ คณะวิทยาลัยการคอมพิวเตอร์

มหาวิทยาลัยขอนแก่น

**คำนำ**

รายงานเล่มนี้จัดทำขึ้นเพื่อเป็นส่วนหนึ่งของรายวิชา SC313504 SOFTWARE QUALITY ASSURANCE

เพื่อศึกษาหาความรู้ในเรื่อง การวิเคราะห์คุณภาพของซอฟต์แวร์ด้วย Software metric/Code metric โดยใช้เครื่องมือ ประเภท Static หรือ Dynamic analysis tools และได้ศึกษาอย่างเข้าใจเพื่อเป็นประโยชน์กับการเรียน

ผู้จัดทำหวังว่า รายงานเล่มนี้จะเป็นประโยชน์กับผู้อ่าน หรือนักเรียน นักศึกษา ที่กำลัง หาข้อมูลเรื่องนี้อยู่ หากมีข้อแนะนำหรือข้อผิดพลาดประการใด ผู้จัดทำขอน้อมรับไว้และขออภัย ณ ที่นี้ด้วย

ผู้จัดทำ

นายอัครเดช วิทยาวุฒิรัตน์

นายนรบดี เดชขันธ์

**สารบัญ**

**เรื่อง หน้า**

Purpose 1

Application Overview 1

Testing Scope 1

Test Environments and Tools 1

Conclusion 2

Functional Testing 2

Cyclomatic Complexity 3

Class Lines of Code 4

Lack of Cohesion 5

Response for a class 6

Quality Gates 7

อ้างอิง 8

1. **Purpose**

จุดประสงค์เพื่อทดสอบการทำงานของ Source Code ว่าทำงานถูกต้องตามที่ควรจะเป็นหรือไม่ และ ทดสอบคุณภาพของ Source Code รวมถึงตรวจสอบความซับซ้อนของ Method ต่าง ๆ ที่อยู่ภายในตัวโปรแกรม

1. **Application Overview**

“BlackJack” เป็นเกมไพ่รูปแบบหนึ่งที่ผู้เล่นต้องวางเดิมพัน Dealer ก็จะเริ่มแจกไพ่ให้ฝั่งละ 2 ใบ หน้าที่ต่อไปคือการทำการจั่วไพ่เพิ่มเติมไปเรื่อย ๆ จนกว่าแต้มของคุณจะใกล้เคียงหรือเท่ากับ 21 แต้ม แต่หากเกิน 21 แต้ม เมื่อไร จะถือว่าแพ้ทันที

1. **Testing Scope**

* Functional Testing
* Cyclomatic Complexity
* Class Lines of Code
* Lack of Cohesion
* Response for a class

1. **Test Environment & Tools**

* Eclipse IDE for Java Developers
* JavaSE-18
* Junit 4
* CodeMR
* JArchitect

1. **Conclusion**

* Table

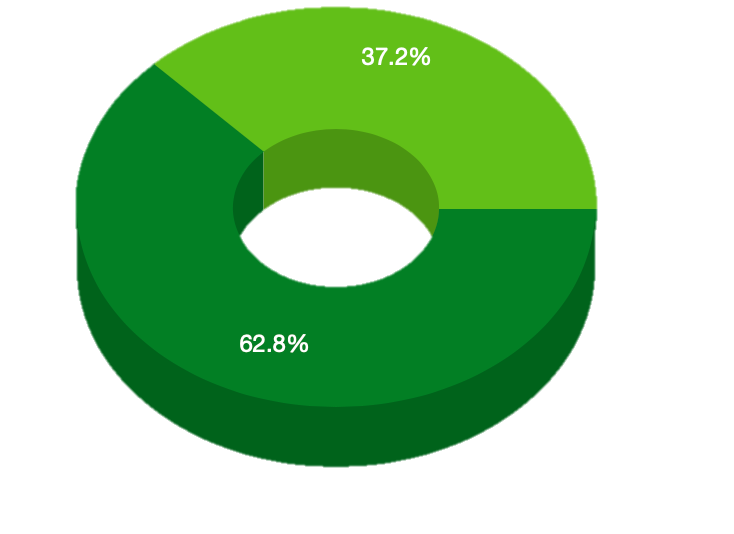
  Description automatically generatedGraphical user interface, application, Word

  Description automatically generatedTable

  Description automatically generatedFunctional Testing

การทดสอบเฉพาะหน่วย (Unit Testing) ผู้ทดสอบได้แบ่ง Test Scenario เป็น 3 ส่วน โดยหลักการสร้าง Test Scenario ยึดหลักกติกาเกม Black Jack ที่ผู้พัฒนาได้สร้างขึ้น ซึ่งเหตุการณ์ที่เกิดขึ้น สามารถแบ่งได้ หลัก ๆ 3 เหตุการณ์ คือ เหตุการณ์ที่ Player ชนะ, Player แพ้ และ เหตุการณ์ที่แต้มไพ่ของทั้ง Player และ Dealer เท่ากัน

จากการทดสอบสามารถสรุปได้ว่า โปรแกรมเกม BGS มีการทำงานที่ในส่วนของการประมวลผลผู้ชนะในเกมถูกต้อง 100% ซึ่งผู้ทดสอบได้ใส่ค่าเข้าไปใน Method whoWon( ) และ Method ได้ Return ค่าออกมาตรงตามความคาดหวังของผู้ทดสอบทั้ง 11 Testcase ซึ่งผลลัพธ์ที่คาดหวัง ยึดหลักความถูกต้องตามกติกาของเกมทั้งหมด



Cyclomatic Complexity

* Cyclomatic Complexity

ความซับซ้อนในการเขียนโปรแกรม (หรือความซับซ้อนของซอฟแวร์) เป็นคุณสมบัติอย่างหนึ่งของซอฟแวร์

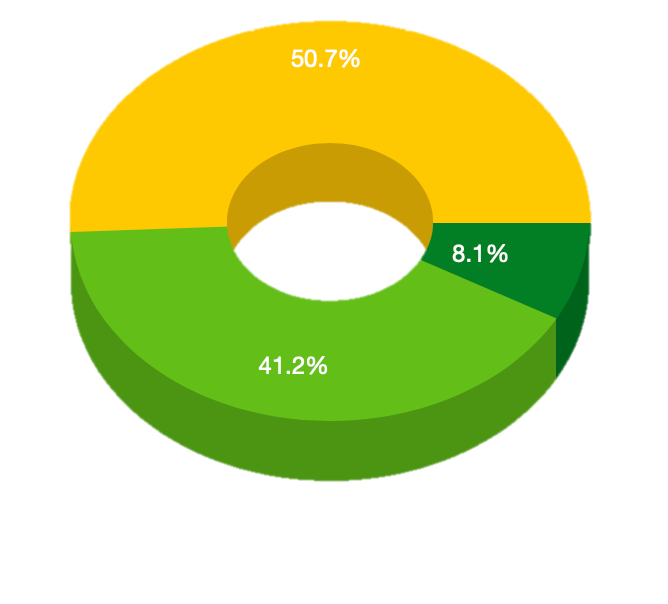
ซึ่งทั้งหมดนี้ส่งผลต่อการโต้ตอบภายในตัวโปรแกรมโดยตรงความซับซ้อนอธิบายการได้จากการที่มีเอนทิตี้ตัวหนึ่งมีความเชื่อมโยงกันเอนทิตี้อีกตัวหนึ่ง ซึ่งอาจจะอยู่คนละคลาสกัน หรือ ต่าง Method เมื่อจำนวนของเอนทิตี้เพิ่มขึ้น จำนวนของการโต้ตอบและการทำงานของโปรแกรมก็จะเพิ่มขึ้นแบบทวีคูณระดับความซับซ้อนที่สูงขึ้นในซอฟต์แวร์จะเพิ่มความเสี่ยงที่จะเกิดความผิดพลาดในการประมวลผลได้

จากการวิเคราะห์พบว่าในตัวโปรแกรม BGS มีค่า Cyclomatic Complexity อยู่ในระดับต่ำ ถึง ปานกลาง ซึ่งวัดจากการตอบสนองกันระหว่างเอนทิตี้ (the interactions between a number of entities) สามารถสรุปได้ว่าตัวโปรแกรม BGS นั้นมีโอกาสเกิดข้อผิดพลาดจากการทำงานของ Software อยู่ในระดับต่ำ เพราะจาการทดสอบ ค่าเปอร์เซ็นต์ของคลาสที่มีความซับซ้อนนั้น พบว่าเอนทิตี้ในแต่ละคลาสมีการโต้ตอบระหว่างกันค่อนข้างน้อย

Total 26 classes

37.2 % 2 classes with low-medium Complexity

62.8 % 24 classes with low Complexity



Class Lines of Code

* Class Lines of Code

เป็นวิธีการที่นิยมใช้ในการประมาณซอฟต์แวร์ใช้ในการวัดขนาดของโปรแกรม

คอมพิวเตอร์ โดยการนับจำนวนบรรทัดนั้นโดยจะไม่นับรวมบรรทัดว่าง, บรรทัดที่มีข้อความเป็นอธิบาย โดยทั่วไปแล้วจะใช้เพื่อคาดการณ์ปริมาณความจำเป็น เช่น โปรแกรมเมอร์ใช้กี่คนใช้เวลานานประมาณเท่าไหร่ และ สามารถบ่อบอกถึง Class Complexity ได้

จากการทดสอบโปรแกรม BGS ด้วยเครื่องมือ CodeMR ทำให้ทราบถึงจำนวนบรรทัดทั้งหมดของโปรแกรม และ จำแนกออกมาเป็นจำนวนบรรทัดของแต่ละ Class สามารถแสดงเป็นเปอร์เซ็นต์ได้ดังกราฟ ซึ่งสามารถสรุปได้ว่า โปรแกรม BGS นั้นออกแบบมาได้ค่อนข้างดี เพราะจำนวนบรรทัดไม่เยอะมาก และ สามารถทำงานตามวัตถุประสงค์ได้ ส่งผลดีต่อการ Maintenance ตัวโปรแกรมในอนาคต รวมถึง ช่วยลดแนวโน้มที่จะเกิดข้อผิดพลาดของการทำงานภายในตัวโปรแกรมอีกด้วย

Total 26 classes

50.7 % 2 classes with medium-hight Class Lines of Code

41.2 % 10 classes with low-medium Lines of Code

8.1 % 14 classes with low Class Lines of Code

A picture containing chart

Description automatically generated

Lack of Cohesion

* Lack of Cohesion

การขาดการประสานกันของคลาสเป็นการวัดจำนวนคู่ของคลาสที่ไม่ได้เชื่อมกันที่แสดงชิ้นส่วนอิสระที่ไม่มีการติดต่อกัน แสดงถึงความแตกต่างระหว่างจำนวนของคู่คลาสที่ไม่มีตัวแปรเหมือนกัน และจำนวนของคู่คลาสที่มีตัวแปรเหมือนกัน

จากการทดสอบโปรแกรม BGS ด้วยเครื่องมือ CodeMR ทำให้ทราบว่าภายในตัวโปรแกรม BGS มีคลาสที่เชื่อมต่อกัน และ มีการสืบทอดตัวแปร หรือ ฟังชั่นต่อกันเป็นส่วนมาก ซึ่งเป็นผลดีเพราะ ทำให้ Developer ใหม่ทำงานได้เร็วขึ้น – ถ้า Code เป็น Structure อยู่แล้ว ทำให้คนใหม่ที่เข้ามาก็สามารถเข้าใจ Code ได้ง่ายขึ้น ลดความผิดพลาดในการทำงาน ยังรวมถึง ความปลอดภัยที่มากขึ้น เนื่องจากการสืบทอดนั้นเราสามารถกำหนดได้ว่าจะให้ใครบ้างที่สามารถเข้าถึง Attriabute หรือ Method นั้นได้บ้าง ช่วยให้ลดโอกาสที่ผิดพลาดจากการให้ Object อื่นภายนอกที่ไม่ได้รับอนุญาติให้เข้าถึงข้อมูล

Total 26 classes

37.7 % 2 classes with high Lack of Cohesion of Methods

2.8 % 1 classes with medium-high Lack of Cohesion of Methods

37.7 % 7 classes with medium-high Lack of Cohesion of Methods

21.8 % 16 classes with low-medium Lack of Cohesion of Methods

Diagram

Description automatically generated

Response For A Class

* Response For A Class

คือการทดสอบการตอบสนองของคลาสแต่ละคลาสว่ามีการเรียกใช้งานไปแล้วสามารถตอบสนองส่งค่ากลับออกมา หรือสามารถทำงานได้ตามความต้องการของผู้ใช้ได้หรือไม่ โดยคลาสที่มีการตอบสนองดีจะส่งผลให้ Software มีความลื่นไหล และ มีโอกาสเกิดข้อผิดพลาดได้น้อย ซี่งการทดสอบนั้นจะทำการเรียกคลาสขึ้นมา และ ทดลองสร้าง Model หรือ Object แล้วเรียกใช้ Method ดูการตอบสนองว่ามีการตอบสนองอยู่ในระดับใด โดยอาจวัดได้หลายด้าน เช่น ความเร็วในการตอบสนอง หรือ ความถูกต้องของการตอบสนอง

จากการทดสอบโปรแกรม BGS ด้วยเครื่องมือ CodeMR ทำให้ทราบว่า ภายในตัวโปรแกรมนั้น คลาสในแต่ละคลาสมีการตอบสนองอยู่ในระดับค่อนข้างดี โดยผลการทดสอบพบว่ามี 1 คลาสในโปรแกรม BGS ตอบสนองอยู่ในระดับสูงเมื่อใช้เครื่องมือทดสอบคิดออกมาเป็น 32.5% และ ระดับ Medium-High คิดเป็น 23.3% ซึ่งคิดเป็นส่วนมากของการตอบสนอง

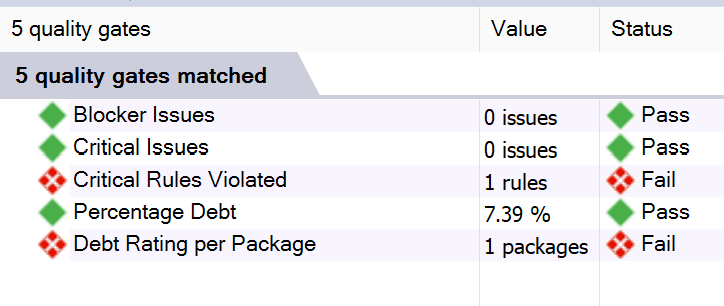
Total Classes

32.5% 1 with high response for a class

23.3% 2 with medium-high response for a class

16% 4 with low medium response for a class

28.2% 19 with low response for a class



Chart, scatter chart

Description automatically generated with medium confidence

* Quality Gates

คือการทดสอบคุณภาพของโปรแกรมก่อนการส่งต่อถึงผู้ใช้ Quality Gates ใช้วัดการทดสอบซึ่งจะส่งผลกลับออกมาเป็น Pass, Warn และ Fail ซึ่งจะมีการตั้งมาตฐานของแต่ละGate เป็นเปอร์เซ็นต์ แล้วทดสอบออกมาและแจงรายละเอียดว่ามีกี่ Gates ที่ทดสอบผ่าน

จากการทดสอบโปรแกรม BGS ด้วยเครื่องมือ JArchitect ทำให้ทราบว่า มีจำนวน 2 Gates ที่ไม่ผ่าน Quality Gates และ 3 Gates ที่ผ่าน Quality Gates ซึ่งจุดที่ Fail คือส่วนของ Avoid types too big และ Debt Rating per Package ที่แพ็คเกจ View เนื่องจากภายในคลาสเขียน Method ที่เป็นลักษณะ Unmaintanaceable เพราะคลาสส่วนมากเป็นคลาสโครงสร้างของหน้าแอพพลิเคชั่นจึงสรุปได้ว่าซึ่งสรุปได้ว่า ตัวโปรแกรมนั้นมีคุณภาพของ Software ที่ยังมีข้อบกพร่องอยู่ ในส่วนของการออกแบบหน้าต่างการใช้งาน

**อ้างอิง**

Hamilton, T. (2022). Test Environment for Software Testing. Guru99. https://www.guru99.com/test-environment-software-testing.html

O. (2021). Software Tester ( Test script | Test case | Test scenario). medium. https://medium.com/@srivichai29/test-ee39fb6772cb

(n.d.). Development and Test Environments: Understanding the Different Types of Environments. unitrends. https://www.unitrends.com/blog/development-test-environments

(2022). วิธีเล่นไพ่แบล็คแจ็คสำหรับมือใหม่ 2022. aviancethailan. https://www.aviancethailand.com/วิธีเล่นไพ่แบล็คแจ็คสำ/

(2022). How To Write An Effective Test Summary Report. softwaretestinghelp. https://www.softwaretestinghelp.com/test-summary-report-template-download-sample/