

ISTEP

ระบบจัดบันทึกรายรับรายจ่าย

MONEY MOVEMENT

ชื่อของ GitHub repository

Money-Movement

นายพัสกร จุลพล 54010907

นายสุรพงศ์ เท่าเทียมตน 54011423

Cloud Computing

ภาคการศึกษาที่ 1/ 2557

สารบัญ

บทคัดย่อ.....	1
บทนำและรายละเอียดการวิเคราะห์หัวข้อ.....	1
งานที่เกี่ยวข้อง	2
สถาปัตยกรรมของระบบ.....	2
ออกแบบระบบ	2
รายละเอียดการพัฒนาซอฟต์แวร์.....	3
Deployment.....	3
Implement plan.....	5
ผลการทดสอบซอฟต์แวร์.....	6
Unit Test.....	6
Evaluation	7
บทสรุป	13
บรรณานุกรม	14

บทคัดย่อ

การออมเงินเป็นสิ่งสำคัญเพราะการทำธุรกรรมต่างๆ เช่น การลงทุน การเริ่มต้นเปิดกิจการ การซื้อบ้าน การซื้อรถยนต์ ต้องใช้เงินจำนวนมาก ซึ่งเป็นที่มาว่าหากสามารถสร้างตัวช่วยในการออมขึ้นมาจะมีประโยชน์ให้คนจำนวนมากได้ ดังนั้นจึงเป็นเหตุผลที่ทำให้ Money Movement ขึ้นมา Money Movement เป็น web application เกี่ยวกับการจดบันทึกรายรับรายจ่ายของผู้ใช้เพื่อให้ผู้ใช้สามารถรู้ว่าตนเองใช้เงินไปกับอะไรบ้างมีจำนวนสัดส่วนอย่างไร จะได้นำไปใช้ประกอบการตัดสินใจในการลดรายจ่ายหรือหาวิธีเพิ่มรายได้เพื่อให้สามารถออมเงินได้มากขึ้น

บทนำและรายละเอียดการวิเคราะห์หัวข้อ

1. คนจำนวนมากเมื่อจะทำธุรกรรมบางอย่างที่ต้องใช้เงินจำนวนมาก เช่น การลงทุน การเปิดกิจการ การซื้อรถยนต์ การซื้อบ้าน มักจะไม่สามารถทำได้เนื่องจากไม่มีเงินออมพอจึงทำให้เกิดการขาดสภาพคล่องทางการเงิน ซึ่งหากสามารถสร้างตัวช่วยที่ช่วยให้ผู้ใช้รู้ว่ามีการรับรายจ่ายอะไรบ้าง มีสัดส่วนระหว่างหมวดหมู่ต่างๆอย่างไรและทำให้เห็นเป็นภาพเพื่อให้เข้าใจง่าย ผู้ใช้ก็จะสามารถใช้ข้อมูลเหล่านั้นเพื่อใช้ประกอบการตัดสินใจว่าจะลดรายจ่ายส่วนใดหรือจะหารายได้เพิ่มเพื่อให้ตนเองมีเงินออมได้

2. ปัญหาอย่างหนึ่งของคนจำนวนหนึ่งคือขาดการวางแผนในการใช้เงินตัวอย่างเช่น นักศึกษาในสถาบันอุดมศึกษาที่ไม่ได้วางแผนการใช้เงินทำให้ในไม่มีเงินเหลือใช้ในช่วงปลายเดือนทำให้ต้องหยิบยืมเงินจากเพื่อน หรือรบกวนผู้ปกครองส่งเงินมาให้เพิ่ม ซึ่งปัญหานี้สามารถแก้ไขได้ถ้าหากผู้ใช้รู้ว่าในวันนี้ผู้ใช้เหลือเงินอยู่เท่าไรและตนใช้เงินจำนวนมากไปกับส่วนใด ผู้ใช้จะสามารถวางแผนได้ว่าตนเองจะใช้เงินจำนวนเท่าไรลดรายจ่ายส่วนไหนจนกว่าจะสิ้นเดือน

3. ปัญหาการทำรายรับรายจ่ายบนสมุดบันทึกซึ่งยุ่งยากไม่สามารถทำได้ทันทีเมื่อทำการใช้จ่ายหรือได้รับเงินเพิ่ม ซึ่งต้องไปเขียนลงสมุดบันทึกซึ่งอาจจะไม่ได้อยู่ติดกับตัว ซึ่งกว่าจะได้กลับไปเขียนก็ลืมไปแล้วว่าตนใช้เงินไปกับอะไร อีกทั้งต้องทำการคำนวณรายรับรายจ่ายสรุปด้วยตัวเอง และเมื่อต้องการทราบว่าหมวดหมู่ใดมีรายรับรายจ่ายเท่าใดก็จะต้องยุ่งยากค้นหาไปมาดูรายละเอียดว่ารายจ่ายนี้อยู่ในหมวดหมู่ใด ซึ่งปัญหานี้สามารถแก้ไขได้เมื่อมีตัวช่วยในการเก็บข้อมูลให้เป็นระเบียบ สามารถเรียกดูข้อมูลตามเป็นหมวดหมู่ตามต้องการได้และสามารถคำนวณให้โดยอัตโนมัติ

ปัญหาต่างเหล่านี้สามารถแก้ไขได้โดยตัว MONEY MOVEMENT เป็น web application เกี่ยวกับการจดบันทึกรายรับรายจ่ายของผู้ใช้เพื่อให้ผู้ใช้สามารถรู้ว่าตนเองใช้เงินไปกับอะไรบ้างมีจำนวนสัดส่วนอย่างไร

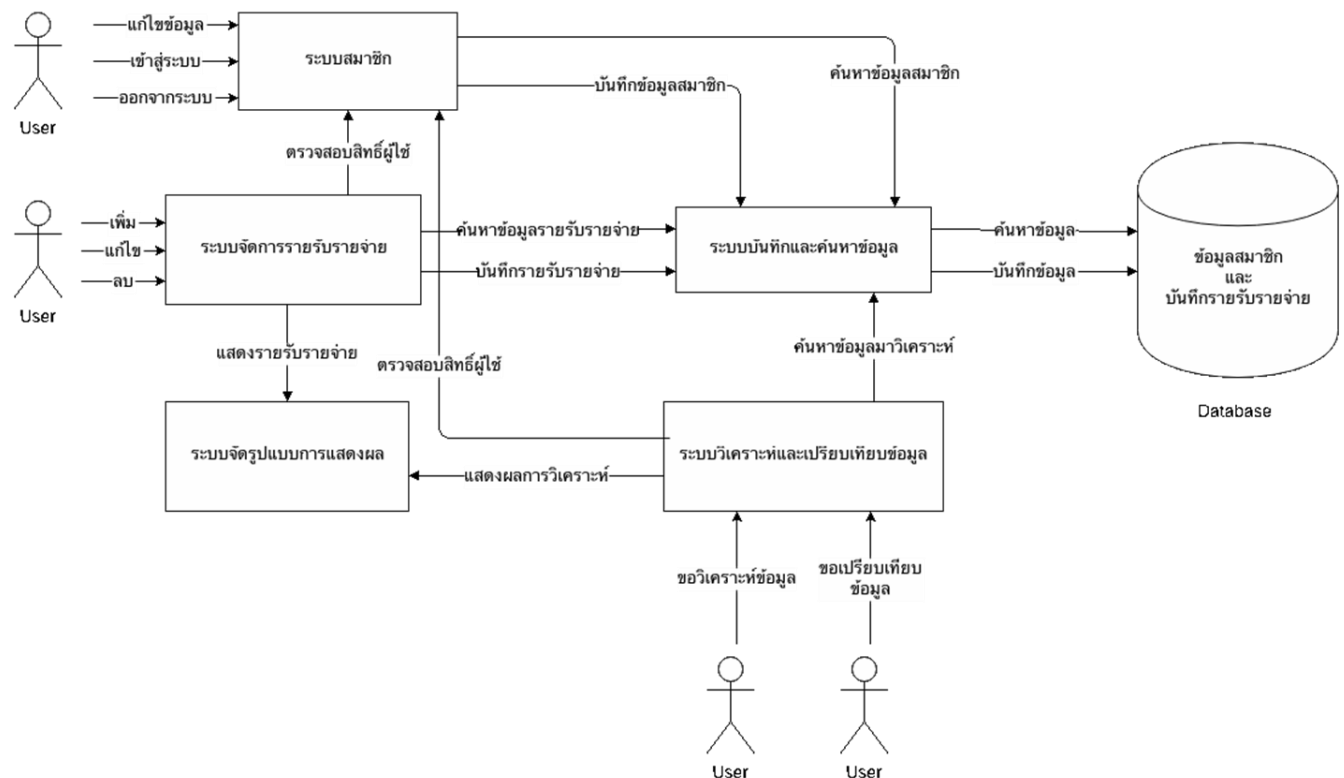
งานที่เกี่ยวข้อง

- MINT เป็น web application ที่สามารถบันทึกรายรับรายจ่ายได้เหมือนกัน แต่ของ MINT นั้นสามารถทำงานได้หลากหลาย สามารถตั้งเป้าหมายการเก็บเงินได้ว่าจะเก็บเงินไปทำอะไร หรือ สามารถกำหนดเป้าหมายรายจ่ายได้ว่าจะใช้เงินกับส่วนนี้จำนวนเท่าไร

- Spendee เป็น application บนสมาร์ตโฟน (smartphone) ที่สามารถบันทึกรายรับรายจ่ายได้ โดยในส่วนของรายรับรายจ่ายที่ทำการบันทึกจะสามารถเลือกประเภทของที่มาได้ ซึ่งได้แบ่งเป็นหมวดหมู่ เช่น ค่ารถ ค่าท่องเที่ยว ค่าพักผ่อน เงินเดือน เป็นต้น และสามารถแสดงกราฟผลการใช้จ่ายออกเป็นรายอาทิตย์ รายเดือน รายไตรมาส รายปี

สถาปัตยกรรมของระบบ

ออกแบบระบบ



ระบบสมาชิก

มีหน้าที่ทำเกี่ยวกับการสมัครสมาชิก เข้าสู่ระบบ ออกจากระบบ การทำการตรวจสอบว่าผู้ใช้ทำการเข้าสู่ระบบแล้วหรือยัง เมื่อเรียกใช้ส่วนที่จำเป็นต้องผ่านการเข้าสู่ระบบจึงจะสามารถทำได้

ส่วนจัดการรายรับรายจ่าย

มีหน้าที่ทำการจัดการเกี่ยวกับ เพิ่มข้อมูลรายรับรายจ่าย แก้ไขข้อมูลรายรับรายจ่าย ลบข้อมูลรายรับรายจ่าย

ส่วนวิเคราะห์ข้อมูล

มีหน้าที่จัดการการวิเคราะห์ข้อมูลรายรับรายจ่ายของผู้ใช้และเปรียบเทียบข้อมูลรายรับรายจ่ายกับค่าเฉลี่ยของผู้ใช้คนอื่นๆ

ส่วนบันทึกและค้นหาข้อมูล

เป็นส่วนที่มีหน้าที่ทำการไปค้นหาข้อมูลและบันทึกข้อมูลจากที่ที่เก็บข้อมูลไว้

ส่วนจัดรูปแบบการแสดงผล

เป็นส่วนที่มีหน้าที่จัดการข้อมูลให้ตรงตามเงื่อนไขกับการนำไปแสดงผล

รายละเอียดการพัฒนาซอฟต์แวร์

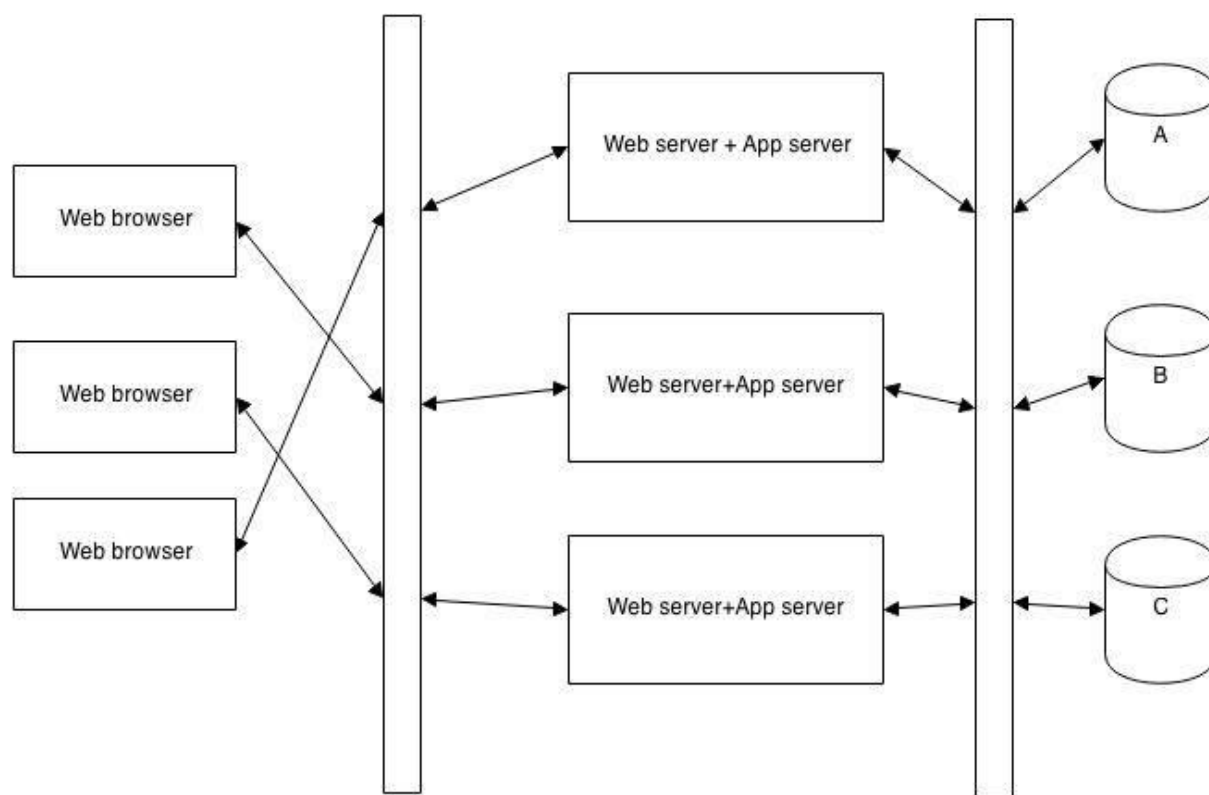
Deployment

ส่วนที่เป็นหน้าแสดงผล

ในส่วนนี้จะใช้ HTML, CSS, Java Script ในการทำส่วนการแสดงผล(Presentation) ที่จะนำไปใช้กับ Web- browser ของผู้ใช้โดยจะเลือกใช้ Bootstrap เพื่อช่วยในการจัดหน้าจอเพื่อให้ใช้แสดงได้ในหลายอุปกรณ์ เช่น คอมพิวเตอร์ มือถือ ที่มีขนาดหน้าจอแตกต่างกัน ใช้ Angular ที่เป็น API ของภาษา Java Script ซึ่งช่วยให้เขียนโปรแกรมได้สะดวกและง่ายขึ้นไม่ต้องทำการเขียนในหลายๆ ส่วนเอง

Web server

ในส่วนนี้จะใช้บริการ Web site ของ Microsoft Azure ที่มีไว้ให้ โดยทาง Azure จะสร้างเครื่องที่ทำการลงโปรแกรมที่จำเป็นต่างๆ ที่จำเป็นในการเป็นเครื่อง Server ไว้ให้ ตัวเราเพียงแค่ทำการส่งชุดคำสั่งของโปรแกรมของเราขึ้นไปบนตัวเครื่องเท่านั้น อีกทั้งตัวบริการนี้สามารถทำการเพิ่ม-ลดขนาด(Scale) โดยสร้างเครื่องใหม่ขึ้นมาทำงานเมื่อเครื่อง Server ตัวเดียวเริ่มรับงานไม่ไหว ซึ่งสามารถตั้งกฎในการเพิ่มจำนวนเครื่องที่เป็น Instances ได้ง่ายโดยไม่จำเป็นต้องเขียนโปรแกรม โดยเราเลือกให้ตัว Web server ทำการลงตัว Apache tomcat และใช้ภาษา Java ในการเขียนส่วนที่เป็น Business logic และใช้ตัว Spring framework เป็นตัวช่วยในการติดต่อ



ส่วนของ Database

ในส่วนนี้จะใช้บริการของ SQL Database ของ Microsoft Azure ซึ่งมีให้อยู่แล้วอีกทั้งยังรับประกันความสามารถในการให้บริการ (Availability) อีกทั้งยังสามารถทำการ Scale up หรือ Scale down ได้ง่าย โดยจะเอา SQL Database ที่ได้นั้นมาทำเป็น Distribute database โดยกระจายข้อมูลแบบ Horizontal ไปให้ทุก Database เพื่อแบ่งภาระงานในการเขียนและอ่านข้อมูล โดยส่วนที่ทำการเลือกที่จะส่งข้อมูลลง Database ว่าจะลง Database ไหนนั้นจะทำการดำเนินการสร้าง(Implement) ขึ้นมาเอง

Implement plan

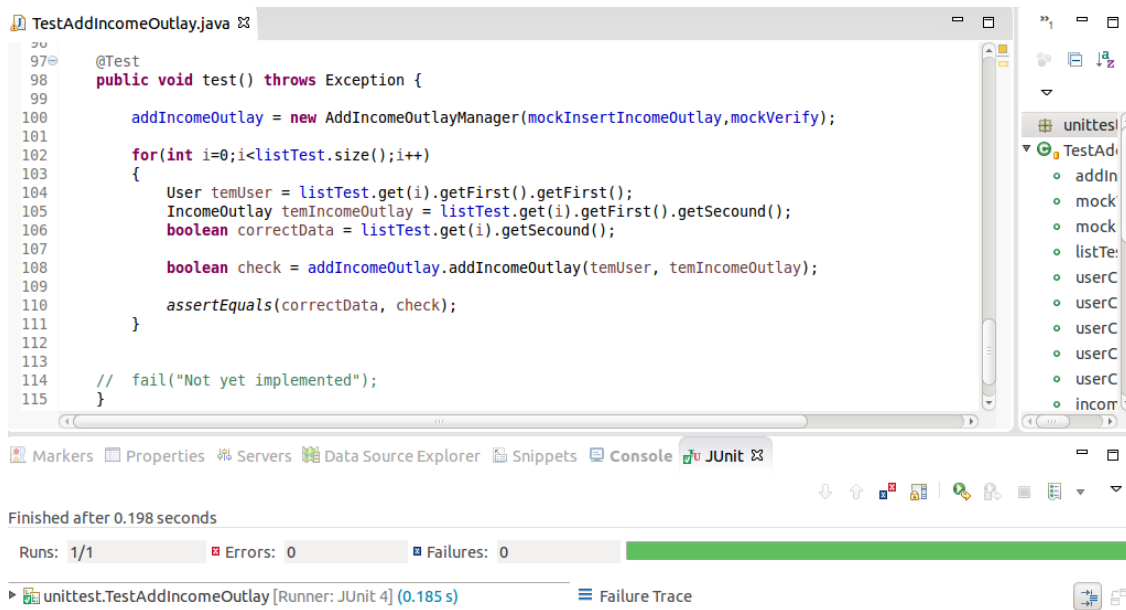
งานที่ทำ	ระยะเวลา/วัน	ผู้รับผิดชอบ
1. ออกแบบส่วนที่เป็น business logic ว่าแต่ละระบบนั้นจะมี Class อะไรบ้าง แต่ละ Class ติดต่อกันอย่างไร	5 (22-26 ก.ย. 57)	นายสุรพงศ์ เท่าเทียมตน
2. ออกแบบส่วนที่เป็นส่วนแสดงผลให้กับผู้ใช้ว่าควรจัดวางอย่างไร และทดลองใช้ Bootstrap สร้างส่วนแสดงผลขึ้นมา	5 (27 ก.ย. -1 ต.ค. 57)	นายพัสกร จุลพล
3. ทดลองใช้บริการ Web site ของ Microsoft Azure ว่าสามารถตั้งค่าอะไรได้บ้าง การติดต่อกับตัว Web site ต้องติดต่อโดยวิธีไหน ตัวชุดคำสั่งจะต้องเก็บไว้ที่ตำแหน่งใดของตัว Web site จึงสามารถทำงานได้	2 (2-3 ต.ค. 57)	นายพัสกร จุลพล
4. ทดลองใช้บริการ SQL Database ของ Microsoft Azure ว่าจะสร้างฐานข้อมูลอย่างไร จะติดต่อกับฐานข้อมูลต้องใช้ตัวติดต่อตัวไหน	2 (9-10 ต.ค. 57)	นายสุรพงศ์ เท่าเทียมตน
5. สร้างและออกแบบฐานข้อมูล	2 (11-12 ต.ค. 57)	นายสุรพงศ์ เท่าเทียมตน
6. เขียนส่วนที่เป็นส่วนจัดรูปแบบแสดงผล และ ส่วนระบบสมาชิก	5 (13-17 ต.ค. 57)	นายพัสกร จุลพล
7. เขียนส่วนแสดงผลที่เกี่ยวกับระบบสมาชิก เช่น หน้าเข้าสู่ระบบ หน้าประวัติสมาชิก	5 (18-22 ต.ค. 57)	นายสุรพงศ์ เท่าเทียมตน
8. เขียนส่วนจัดการรายรับรายจ่าย ส่วนบันทึกและค้นหาข้อมูล	5 (24-28 ต.ค. 57)	นายสุรพงศ์ เท่าเทียมตน
9. เขียนส่วนแสดงผลที่เกี่ยวกับการจัดการรายรับรายจ่าย เช่น หน้าบันทึกรายรับรายจ่าย หน้าค้นหา หน้าลบบรายรับรายจ่าย	5 (29 ต.ค.-2 พ.ย. 57)	นายพัสกร จุลพล
10. เขียนส่วนวิเคราะห์และเปรียบเทียบข้อมูล	5 (3-7 พ.ย. 57)	นายสุรพงศ์ เท่าเทียมตน
11. เขียนส่วนแสดงผลการวิเคราะห์และเปรียบเทียบ	5 (8-12 พ.ย. 57)	นายพัสกร จุลพล
12. เริ่มทำการสร้าง Distribute database และทำส่วนที่ทำการแยกว่า จะเลือกค้นหาหรือบันทึกลง Database ตัวไหน	3 (13-15 พ.ย. 57)	นายสุรพงศ์ เท่าเทียมตน
13. การ Test แต่ละส่วนของระบบ	5 (15-19 พ.ย. 57)	นายพัสกร จุลพล

ผลการทดสอบซอฟต์แวร์

Unit Test

TestAddIncomeOutlay.java ไฟล์นี้ใช้ตรวจสอบเกี่ยวกับการเพิ่มบันทึกรายรับรายจ่ายว่ามีข้อผิดพลาดหรือไม่ โดย file test จะอยู่ใน directory ดังนี้

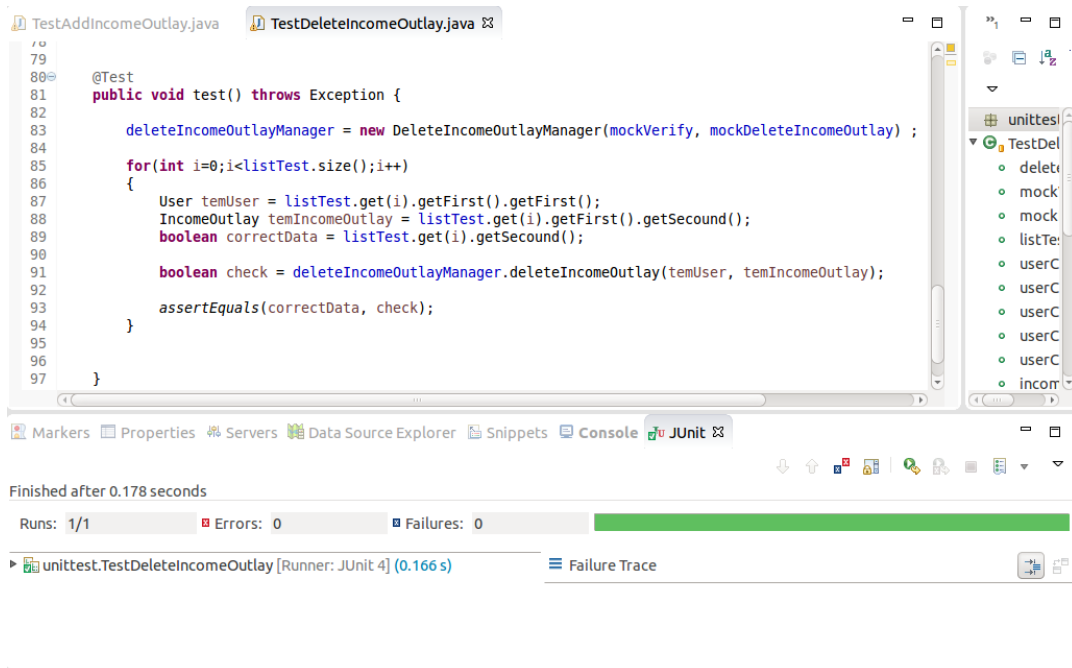
/ROOT/src/unittest/TestAddIncomeOutlay.java



จากการทดลอง unit test พบว่า Class `AddIncomeOutlayManager` สามารถผ่านการ unit test ได้ทุก case ที่ได้มีกำหนดไว้

TestDeleteIncomeOutlay.java ไฟล์นี้ใช้ตรวจสอบเกี่ยวกับการลบบันทึกรายรับรายจ่ายว่ามีข้อผิดพลาดหรือไม่ โดย file test จะอยู่ใน directory ดังนี้

/ROOT/src/unittest/TestDeleteIncomeOutlay.java



จากการทดลอง unit test พบว่า Class DeleteIncomeOutlayManager สามารถผ่านการ unit test ได้ทุก case ที่ได้กำหนดไว้

Evaluation

การทดลองที่ 1

การบันทึกการรับรายจ่าย

- จุดประสงค์: เพื่อดูว่าการบันทึกการรับ-รายจ่ายของผู้ใช้นั้นถูกบันทึกลงระบบจริงๆ และเมื่อนำข้อมูลกลับมาแสดงผลนั้นถูกต้องตามที่บันทึกไปหรือไม่
- สิ่งที่จะวัด: ความถูกต้องของข้อมูลการรับรายจ่าย เช่น วันที่บันทึก จำนวนเงิน ชนิดของข้อมูลการรับรายจ่ายที่ใช้ เป็นต้น
- วิธีทำและสิ่งที่ต้องใช้ในการทดลอง:
 1. ทำการบันทึกข้อมูลการรับรายจ่ายให้กับระบบ
 2. ตรวจสอบการแสดงผลที่หน้าจอว่าแสดงผลหลังจากบันทึกถูกต้องตามที่ต้องการหรือไม่
 3. ทำการออกจากระบบจากนั้นเข้าสู่ระบบอีกครั้งและเข้าไปดูส่วนแสดงผลการรับรายจ่ายว่าข้อมูลที่บันทึกไปนั้นยังอยู่และมีค่าถูกต้องหรือไม่
 4. บันทึกผลการทดลอง

5. ทำการทดลองซ้ำอีกตามจำนวนครั้งที่ต้องการ

- ผลที่ได้จากการทดลอง:

จากการทดลองที่จะทำการบันทึกรายรับรายจ่ายโดยจะทดลองบันทึกข้อมูลรายรับรายจ่าย 5 รายรับรายจ่ายโดยผลออกมาว่าสามารถบันทึกได้

<input type="checkbox"/>	2014-10-31	food1	food	outcome	100	del
<input type="checkbox"/>	2014-10-31	food2	food	outcome	200	del
<input type="checkbox"/>	2014-10-31	food3	food	outcome	300	del
<input type="checkbox"/>	2014-10-31	food4	food	outcome	400	del

รูปที่ 1 ภาพก่อนการบันทึกข้อมูลรายรับรายจ่ายที่ 5

<input type="checkbox"/>	2014-10-31	food1	food	outcome	100	del
<input type="checkbox"/>	2014-10-31	food2	food	outcome	200	del
<input type="checkbox"/>	2014-10-31	food3	food	outcome	300	del
<input type="checkbox"/>	2014-10-31	food4	food	outcome	400	del
<input type="checkbox"/>	2014-10-31	food5	food	outcome	500	del

รูปที่ 2 ภาพการบันทึกรายรับรายจ่ายที่ 5 ที่ทำการบันทึกได้สำเร็จ

รูปที่ 3 ภาพผลการทดลองการเข้ามาระบบอีกครั้งเพื่อดูข้อมูลรายรับรายจ่าย

และการทดลองออกจากระบบและเข้ามาดูข้อมูลใหม่ว่าข้อมูลที่บันทึกลงไปนั้นยังอยู่หรือไม่ซึ่งจากการทดลองพบว่าข้อมูลยังอยู่ครบ

และจากการทดลองซ้ำอีก 2 ครั้งเพื่อทดสอบว่าจะสามารถบันทึกข้อมูลได้หรือไม่นั้นการทดลองทุกครั้งก็สามารถบันทึกข้อมูลได้อย่างถูกต้องแม้ออกจากระบบกลับมาข้อมูลก็ยังอยู่

- สรุปผลการทดลอง:

จากผลการทดลองพบว่าการบันทึกรายรับรายจ่ายนั้นสามารถทำได้ถูกต้อง และเมื่อออกจากระบบและกลับเข้ามาอีกครั้งข้อมูลยังสามารถแสดงได้อย่างถูกต้อง

การทดลองที่ 2

การวิเคราะห์รายรับรายจ่าย

- จุดประสงค์: เพื่อดูว่าการวิเคราะห์ข้อมูลรายรับรายจ่ายเป็นไปตามสูตรการวิเคราะห์ที่กำหนดให้หรือไม่

- สิ่งที่จะวัด: ความถูกต้องของผลลัพธ์การวิเคราะห์

- วิธีทำและสิ่งที่ต้องใช้การทดลอง:

1. ทำการวิเคราะห์ข้อมูลรายรับรายจ่ายที่กำหนดไว้เพื่อหาผลลัพธ์การวิเคราะห์

2. ทำการเปรียบเทียบผลลัพธ์จากการทดลองที่ได้จากระบบกับผลการวิเคราะห์ที่ถูกต้องว่าตรงกันหรือไม่
3. บันทึกผลการทดลอง
4. ทำการทดลองซ้ำอีกตามจำนวนครั้งที่ต้องการ

- ผลที่ได้จากการทดลอง:

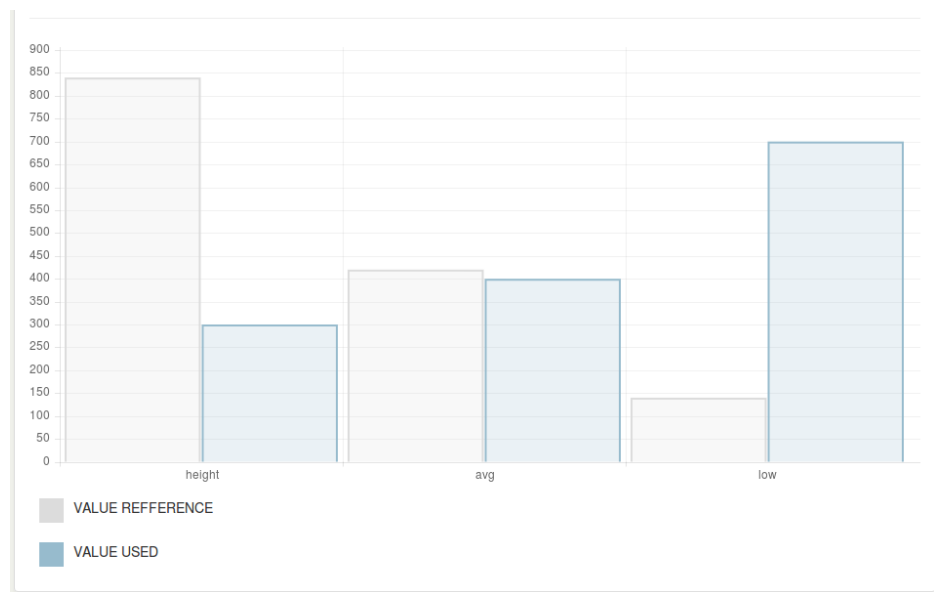
ในการทดลองนี้จะทำการทดลองวิเคราะห์ข้อมูลรายรับรายจ่ายโดยใช้หลักการคำนวณโดยอาศัยหลักการใช้เงินตามความสำคัญ โดยหลักการคือเราควรใช้เงินให้กับรายจ่ายที่สำคัญ 6 ส่วน รายจ่ายที่จำเป็นรองลงมา 3 ส่วน และรายรับรายจ่ายที่จำเป็นน้อย 1 ส่วน เช่น ถ้ารายจ่ายทั้งหมดเป็น 1000 รายจ่ายควรเป็นดังตาราง

	รายจ่ายที่มีความสำคัญ	รายจ่ายที่มีความสำคัญรองลงมา	รายจ่ายที่มีความจำเป็นน้อย
การแบ่งที่ควรเป็น	600	300	100

ในการทดลองนี้จะกำหนดให้ผู้ใช้จ่ายเงินเป็นจำนวน 1400 โดยแบ่งเป็นตามตารางดังนี้

	รายจ่ายที่มีความสำคัญ	รายจ่ายที่มีความสำคัญรองลงมา	รายจ่ายที่มีความจำเป็นน้อย
รายจ่ายที่จ่ายจริง	300	400	700
การแบ่งที่ควรเป็น	840	420	140

ซึ่งจากทดลองพบว่าผลออกมาเป็นดังที่ต้องการดังภาพ

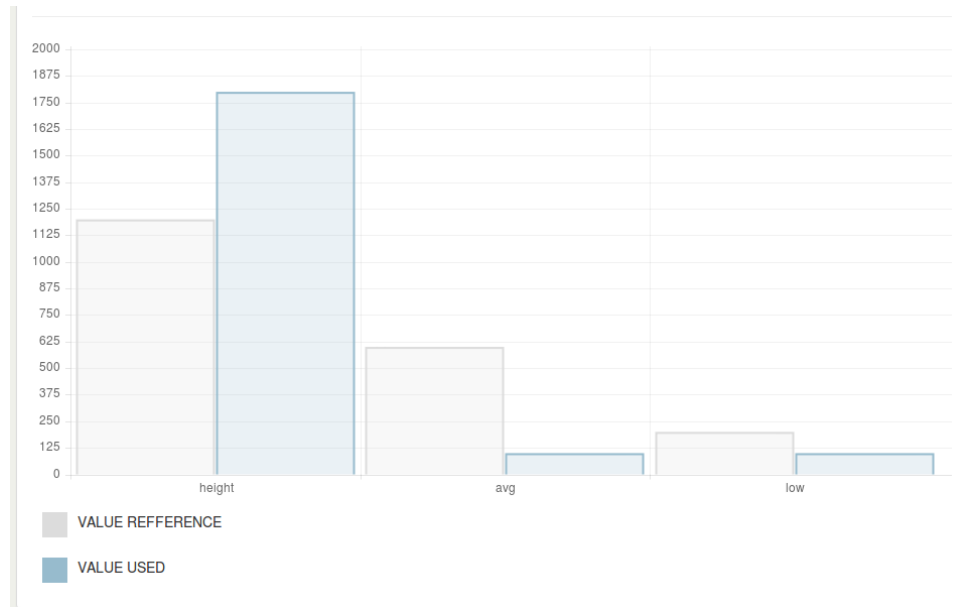


รูปที่ 4 ภาพผลการวิเคราะห์ข้อมูลการทดลองครั้งที่ 1

โดยจากภาพกราฟแท่งสีขาวยุหมายถึงรายจ่ายที่ควรจะเป็น กราฟแท่งสีฟ้าหมายถึงรายจ่ายที่ใช้ไปจริงๆซึ่งเป็นไปตามที่คาดไว้ และจากการทดลองอีก 2 ครั้งเพื่อตรวจสอบอีกครั้ง

ผลที่คาดหวังในการทดลองครั้งที่ 2

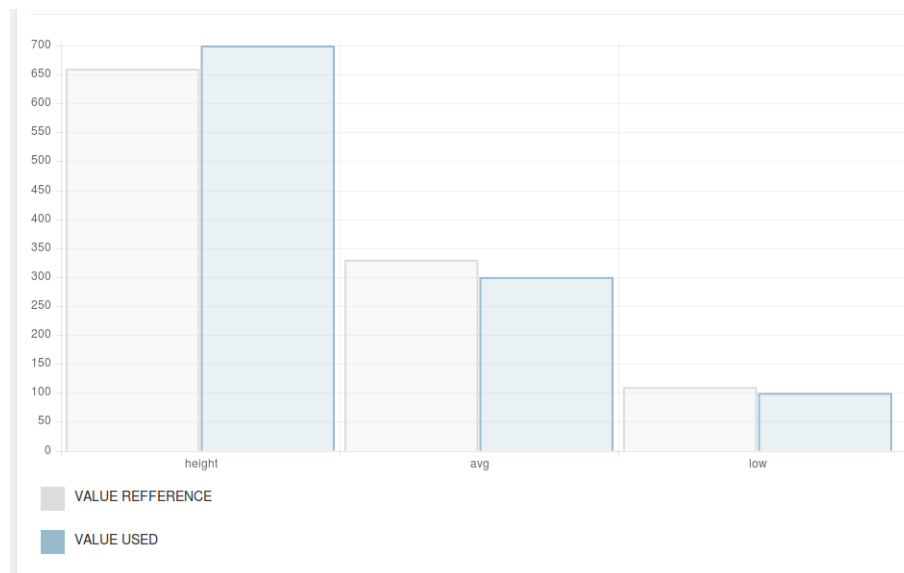
	รายจ่ายที่มีความสำคัญ	รายจ่ายที่มีความสำคัญลงมา	รายจ่ายที่มีความจำเป็นน้อย
รายจ่ายที่จ่ายจริง	1800	100	100
การแบ่งที่ควรเป็น	1200	600	200



รูปที่ 5 ภาพผลการวิเคราะห์การทดลองครั้งที่ 2

ผลที่คาดหวังในการทดลองครั้งที่ 3

	รายการที่มีความสำคัญ	รายการที่มีความสำคัญลงมา	รายการที่มีความจำเป็นน้อย
รายการที่จ่ายจริง	700	300	100
การแบ่งที่ควรเป็น	660	600	200



รูปที่ 6 ภาพผลการวิเคราะห์การทดลองครั้งที่ 3

- สรุปผลการทดลอง:

จากผลการทดลองสามารถสรุปได้ว่าการวิเคราะห์การทดลองเป็นไปตามสูตรการทดลองที่กำหนด

บทสรุป

Money Movement เป็น web application เกี่ยวกับการจดบันทึกรายรับรายจ่ายของผู้ใช้เพื่อให้ผู้ใช้สามารถรู้ตัวเองใช้เงินไปกับอะไรบ้างมีจำนวนสัดส่วนอย่างไร สามารถวิเคราะห์รายรับรายจ่ายของผู้ใช้งานได้ สามารถเปรียบเทียบรายรับรายจ่ายของตนกับค่าเฉลี่ยรายรับรายจ่ายของคนอื่นๆได้ ซึ่งจะนำมาแสดงผลในรูปแบบกราฟ แผนภูมิ เพื่อให้ผู้ใช้เข้าใจง่ายขึ้น และด้วยความสามารถที่ได้กล่าวไปผู้ใช้สามารถนำข้อมูลการวิเคราะห์และเปรียบเทียบ นำไปใช้ประกอบการตัดสินใจในการลดรายจ่าย หาวิธีเพิ่มรายได้ หรือ วางแผนการใช้เงิน

โดยในการพัฒนานั้น Money Movement ถูกออกแบบโดยใช้หลักการออกแบบเชิงวัตถุซึ่งเป็นความรู้ใหม่ที่ผู้จัดทำไม่มีความรู้มากนัก อีกทั้งยังต้องใช้เทคโนโลยีใหม่ๆ เช่น Cloud service ในการพัฒนา ซึ่งทำให้ต้องใช้เวลาในการศึกษาและทดลองเป็นอย่างมาก ทำให้การพัฒนาทำได้ช้าไม่ตรงตามเวลาที่ได้วางแผนไว้ในตอนแรก แต่ท้ายที่สุดผู้พัฒนาก็สามารถทำการพัฒนา Money Movement จนสามารถมาใช้งานได้จริงได้ ซึ่งจากการพัฒนา Money Movement นั้นทำให้ผู้ใช้ได้ความรู้ใหม่มากมาย เช่น การออกแบบเชิงวัตถุ ,Cloud service, json , angularjs , spring framework,การทำ unit test สุดท้ายนี้ผู้พัฒนาหวังเป็นอย่างยิ่งว่า Money Movement นั้นจะสามารถนำไปใช้ในชีวิตประจำวัน เพื่อช่วยแก้ปัญหาในการใช้จ่ายได้

บรรณานุกรม

Google Inc. 2010-2012. “AngularJS.”

[ระบบออนไลน์]. แหล่งที่มา <https://angularjs.org/> (20 ตุลาคม 2557)

Twitter Inc. 2010. “Bootstrap.”

[ระบบออนไลน์]. แหล่งที่มา <http://getbootstrap.com/> (4 ตุลาคม 2557)

Microsoft Corporation 2010 “Azure.”

[ระบบออนไลน์]. แหล่งที่มา <http://azure.microsoft.com/en-us/> (2 กันยายน 2557)

W3shcool.com. 2012. “JSON.”

[ระบบออนไลน์]. แหล่งที่มา <http://www.w3schools.com/json/default.asp> (25 กันยายน 2557)

Pivotal Software, Inc. 2014 “spring framework”

[ระบบออนไลน์]. แหล่งที่มา <http://spring.io/> (30 กันยายน 2557)