Final Report

Functional Programing

**เนื้อหา**

จากที่ได้ Sample ไปเมื่อวันที่ 11/3 จะเห็นได้ว่า C# ได้มีการนำเอา Functional Programming เข้ามาประยุกต์ใช้มากขึ้น ซึ่งจากตัวอย่างเราสามารถสร้างตัวแปรที่ชื่อว่า Func ที่เก็บค่า function เอาไว้ได้โดยตัวอย่างจะแสดง function ที่คำนวณค่าเวลาเพื่อใช้ในการแสดงผล UI เป็นวินาที จะเห็นได้ว่าเราจะประกาศตัวแปรที่ชื่อว่า Func โดยมี parameter คือ <float,float,float> ซึ่งคล้ายกับการประกาศ type ในภาษา functional เช่น Haskell (รับ Input เป็น float 2 ตัว แล้ว output เป็น float หรือก็คือ a -> a -> a)



จากตัวอย่าง นี่คือ function ที่คำนวณเวลาโดยแสดงผลออกเป็นวินาทีแบบปกติทั่วไปของภาษา c#

Text

Description automatically generated

ซึ่งเราไม่ต้องประกาศเป็น function แยกเหมือนตัวอย่างข้างบนแล้ว สามารถสร้างตัวแปรที่เป็น function ใน บรรทัดนั้นๆได้เลย

Graphical user interface, text

Description automatically generated

Project ที่ผู้พัฒนาได้เลือกมาทำ functional programming คือ project เดิม (เกม Escape) โดยที่ผู้พัฒนาได้หยิบยก feature ใน to do list ซึ่งอันนี้ได้มีการ pospond ไปหลายรอบมากๆ กับ feature หลักๆคือ ผู้พัฒนาคิดวิธีในการ implement ไม่ออก มันคือ การสร้าง map เกมด้วยการ Input file .txt เข้าไป โดยใช้หลักการของ functional programming ผ่าน classlib ที่ชื่อ Option หรือถ้าในภาษา Haskell ก็คือ Maybe นั่นเอง

Shape

Description automatically generated with medium confidence

จากการที่ได้เรียนรู้ในการใช้ตัว option นี้คือ ได้เรียนรู้การสร้าง library .net ของตัวเองรวมไปถึงการ import เข้าไปใน unity ผ่านไฟล์ .dll ซึ่งเป็นองค์ความรู้ใหม่ที่ผู้พัฒนาได้ลองทำเลยทำให้ใช้เวลาพอสมควรในการศึกษาข้อมูลในการ import โดยปกติแล้วใน Unity พวก package ต่างๆ สามารถ download และ import มาได้เลยใน package manager ของ unity แต่ว่าถ้าเมื่อเราจะใช้ package ที่เป็นของตัวเองหล่ะ?(class option) ผู้พัฒนาเลยได้สืบค้นวิธีการเพื่อสามารถนำ classlib ของตัวเองไปใช้ในตัว unity ผ่านไฟล์ .dll

ซึ่งในคลาส option จะมีอยู่ 2 ตัวแปรหลักๆ คือ none กับ some เลยทำให้ผู้พัฒนาได้นำเอา pattern matching ซึ่งในที่นี่คือ method ที่ชื่อว่า match ซึ่งจากตัวอย่างคือ สร้างตัวแปรที่ชื่อว่า someblock และใช้ method SomeWhen เมื่อ asset.text.Length มากกว่า 2 จะทำให้ someblock มีค่าเป็น some ที่มีขนาดของ asset.text.length อยู่ มิเช่นนั้นเป็น none และหลังจากนั้นใช้ someblock.matchnone คือถ้า someblock เป็น none จะไปเรียก function wrongInput

Text

Description automatically generated with medium confidence

และผู้ใช้ก็ได้มีการใช้หลักการประมาณนี้ซึ่งคล้ายๆ กันในการสร้าง feature

Text

Description automatically generated

และนี่คือ code ทั้งหมด

Text

Description automatically generated

Text

Description automatically generatedText

Description automatically generated

ในตัว code เองอาจจะดูน้อยใช้ครับมันน้อย แต่ว่าใช้เวลาคิดเยอะมากกกกก โดยปกติแล้วถ้าเป็นแนวนี้ผู้พัฒนาจับใส่  if else หมดแล้วมันจะอิงรุงตุงนังไปหมด ยกตัวอย่าง code

Text

Description automatically generated

และนี่คือผลลัพธ์ของตัว code

Input: Text, application

Description automatically generated

Output:Shape

Description automatically generated with medium confidence

**สรุป**

จากที่ได้ลองใช้ตัว option มาทำให้รู้ได้ว่า เราจะไม่ปวดหัวกับการทำ if else ซ้อน ๆ จากการ implement code ข้างต้น แต่ผู้พัฒนาคิดว่ามันอาจจะทำได้ดีกว่านี้ clean ตัว code ได้ดีกว่านี้ แต่โดยที่ว่าผู้พัฒนาเพิ่งได้มาเจอ classlib นี้ รวมไปถึงทุ่มเทเวลาให้กับการ import classlib ไปใช้เองใน unity ทำให้อาจจะไม่มีเวลาได้ศึกษาตัว classlib option นี้ทั้งหมดแต่ก็ได้เปิดโลกในการเขียน code ของผู้พัฒนาในอีกซีกนึง โดยproject นี้เป็น project ที่ต่อเนื่องและเป็น  project จบของผู้พัฒนาเองดังนั้น ผู้พัฒนาเล็งเห็นว่าในอนาคตจะมีการ clean code โดยการใช้ ofunctional programming เข้ามาเพื่อลดความซับซ้อนในการอ่านของแต่ละ feature ที่ได้ implement ไปหรือที่กำลังจะ implement ในอนาคต