

เอกสารการออกแบบระบบ

แอปพลิเคชัน Unknown Bookstore

นำเสนอ

อาจารย์เกียรติณรงค์ ทองประเสริฐ

อาจารย์คณัฐ ตั้งติสานนท์

ผู้จัดทำโครงการ :

1. นางสาวกนิษฐา ศศิปรียจันทร์ 61010014
2. นายชวกร เหลาแก้ว 62010175
3. นายณพวัฒน์ คำยอด 62010243
4. นางสาวนภาพร ตั้งใจ 62010462
5. นายอภิรักษ์ อูลิศ 62011019

ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ภาคเรียนที่ 2 ปีการศึกษา 2563

สารบัญ

บทที่ 1 Data Structure	1
ความหมาย	1
โครงสร้างพื้นฐาน	1
โค้ดตัวอย่าง	3
การประยุกต์ใช้	7
บทที่ 2 Unknown Bookstore.....	8
ประวัติความเป็นมา.....	8
จุดประสงค์	8
รูปแบบโปรแกรมและการใช้งาน	1
การใช้โครงสร้างข้อมูลในการจัดสรร	17
โค้ดตัวอย่างของฟังก์ชันหลัก	18
บทที่ 3 การทำงานในกลุ่ม	22
การวางแผนและดำเนินงาน	22
การมีส่วนร่วมในกลุ่ม	23
ปัญหาที่เกิดขึ้นภายในกลุ่ม.....	25
แนวทางการแก้ไข	25
ปัญหาที่เกิดขึ้นของตัวโปรแกรม.....	26
แนวทางการแก้ไขตัวโปรแกรม	26
แนวคิดต่อยอดของโปรแกรม	26
สรุปผลการทำงาน.....	26

บทที่ 4 เอกสารที่เกี่ยวข้อง	27
Tkinter.....	27
การใช้งาน csv.....	28
เว็บไซต์ร้านหนังสือต้นแบบ.....	28
อ้างอิง.....	29

บทที่ 1

Data Structure

1. ความหมาย

Data Structure (โครงสร้างข้อมูล) เป็นรูปแบบพิเศษสำหรับการจัดและเก็บข้อมูล ประเภทโครงสร้างข้อมูลทั่วไปรวมถึง array, ไฟล์, เรคคอร์ด, ตารางข้อมูล, tree และอื่นๆ โครงสร้างข้อมูลต่างๆได้รับการออกแบบให้จัดข้อมูลให้เหมาะสมกับวัตถุประสงค์เฉพาะ ดังนั้นสามารถเข้าถึงและทำงานภายในวิธีการที่เหมาะสม ในโปรแกรมคอมพิวเตอร์โครงสร้างข้อมูลอาจจะได้รับการเลือกหรือออกแบบให้เก็บข้อมูลสำหรับวัตถุประสงค์ของการทำงานด้วยอัลกอริทึมที่หลากหลาย

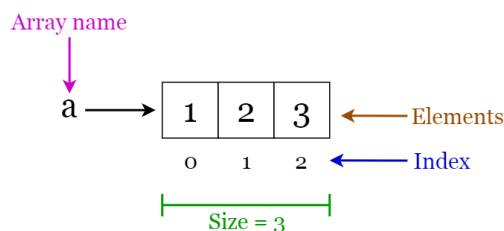
คำว่าโครงสร้างข้อมูล เกิดจากคำสองคำคือ คำว่า โครงสร้าง(Structure) และคำว่า ข้อมูล (Data) สำหรับข้อมูล หมายถึง ข้อเท็จจริงหรือสิ่งที่ถือหรือยอมรับว่าเป็นข้อเท็จจริงสำหรับใช้เป็นหลักฐานหาความจริงหรือการคำนวณ และโครงสร้าง หมายถึง ส่วนประกอบสำคัญๆ ซึ่งนำมาเชื่อมเข้าด้วยกัน ให้เป็นรูปร่างเดียวกัน ดังนั้นโครงสร้างข้อมูลจึงหมายถึง การนำเอาข้อมูลหรือส่วนย่อยๆ ที่ได้ทำการรวบรวมมาให้อยู่ในรูปแบบหรือโครงสร้างที่เหมาะสม เพื่อให้เครื่องคอมพิวเตอร์ประมวลผล ข้อมูลได้อย่างรวดเร็ว และมีประสิทธิภาพ จะเห็นว่าสิ่งพื้นฐานในการประมวลผลข้อมูลด้วยคอมพิวเตอร์ก็คือข้อมูล ดังนั้นการศึกษาถึงความสัมพันธ์ของข้อมูลจึงมีความสำคัญอย่างมากในศาสตร์คอมพิวเตอร์ (Computer Science)

โดยโครงสร้างข้อมูลจะมีการเก็บซึ่งมีอยู่หลากหลายรูปแบบ เช่น เขตข้อมูล(Field), แถวลำดับ(Array), ระเบียบ(Record), ต้นไม้(Tree), ลิงค์ลิสต์(Link List) เป็นต้น

2. โครงสร้างพื้นฐาน

a) แถวลำดับ (Array)

แถวลำดับเป็นโครงสร้างข้อมูลชนิดตรึงขนาดข้อมูล ซึ่งสามารถจัดเก็บข้อมูลที่มีประเภทข้อมูลเดียว กันไว้ โดยประเภทของข้อมูลสามารถเป็นจำนวนเต็ม จำนวนทศนิยม สตริงหรือแม้กระทั่งตัวแถวลำดับเอง และเนื่องจากแถวลำดับมีการจัดเรียงเป็นดัชนี ทำให้การเข้าถึงแบบสุ่มเป็นไปได้



ภาพที่ 1.1 ภาพการทำงานของแถวลำดับอย่างง่าย

b) ลิงค์ลิสต์ (Linked list)

ลิงค์ลิสต์เป็นโครงสร้างลำดับที่ประกอบไปด้วยข้อมูลที่เรียงต่อกันเป็นเส้นตรง โดยข้อมูลนั้นเชื่อมต่อซึ่งกันและกัน ดังนั้นการจะเข้าถึงแบบสุ่มจึงเป็นไปได้ไม่ได้และผู้ใช้จะต้องเข้าถึงข้อมูลตามลำดับชั้น กล่าวคือจากข้อมูลหนึ่งไปอีกข้อมูลหนึ่งตามลำดับก่อนหลัง เช่น หากผู้ใช้ต้องการเข้าถึงข้อมูลที่ 3 จากตำแหน่งข้อมูล ณ ปัจจุบัน ผู้ใช้จะต้องผ่านข้อมูลข้างหน้าก่อน 2 ตัว

c) สแต็ค (Stack)

สแต็คคือโครงสร้างข้อมูลแบบ LIFO (Last In First Out) กล่าวคือ ข้อมูลที่ถูกใส่เข้าโครงสร้างไปเป็นลำดับสุดท้ายจะเป็นข้อมูลแรกที่ถูกดึงออก เป็นโครงสร้างข้อมูลที่พบได้ทั่วไปในการเขียนโปรแกรมหลายๆ ภาษา โดยโครงสร้างข้อมูลนี้ถูกให้ชื่อว่าสแต็คเนื่องจากการทำงานคล้ายคลึงกับการวางจานหรือสิ่งของอื่นๆ เรียงซ้อนกัน

d) คิว (Queue)

คิวคือโครงสร้างข้อมูลแบบ FIFO (First In First Out) กล่าวคือ ข้อมูลที่ถูกใส่เข้าโครงสร้างไปเป็นลำดับแรกจะถูกนำออกมาทำงานก่อน เป็นโครงสร้างข้อมูลที่พบได้ทั่วไปในการเขียนโปรแกรมหลายๆ ภาษา เช่นเดียวกับสแต็ค โดยโครงสร้างข้อมูลนี้จะมีการทำงานคล้ายคลึงกับการรอคอยแถวหรือคิวในชีวิตจริง

e) Hash Table

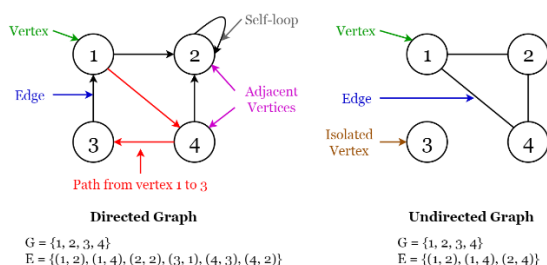
Hash table เป็นอาร์เรย์ที่มีขนาดคงที่และมีหน่วยข้อมูล เรียกว่า keys บรรจุอยู่ เช่น หน่วยข้อมูลอาจจะประกอบด้วยสตริง (ซึ่งถูกกำหนดให้ใช้เป็น key) และฟิลด์อื่น ๆ (เช่น ชื่อลูกจ้างที่เป็นส่วนประกอบหนึ่งของโครงสร้างข้อมูลของลูกจ้าง) แต่ละค่าของ key จะถูก mapped ไปเป็นตัวเลข (หมายเลขเซลล์) ที่อยู่ในช่วง 0 ถึง Table Size – 1 และค่าของ key นั้นก็จะถูกบรรจุลงในเซลล์ตามหมายเลขดังกล่าว

f) Tree

โครงสร้างข้อมูลที่ออกแบบมาให้มีลักษณะไม่เป็นเชิงเส้น มีการจัดเก็บข้อมูลเชื่อมโยงกันเป็นระดับชั้น โดยเริ่มจากโหนดแรกที่อยู่บนสุดเรียกว่า Root Node เชื่อมโยงไปยังโหนดระดับรองลงไป แต่ละระดับก็มีการเชื่อมโยงโหนดระดับต่อไป ซึ่งเรียกว่า Subtree

g) Graph

กราฟเป็นเป็นโครงสร้างข้อมูลแบบไม่เป็นเชิงเส้น เช่นเดียวกับโครงสร้างข้อมูลทรี กราฟเป็นโครงสร้างข้อมูลที่นิยมใช้ในการจัดเก็บข้อมูลและถูกนำไปประยุกต์ใช้ในงานด้านต่างๆ เช่น การหาระยะทางที่สั้นที่สุดจากเมืองหนึ่งไปยังเมืองหนึ่ง การคำนวณค่าใช้จ่ายในการเดินสายเชื่อมต่อระบบเครือข่ายคอมพิวเตอร์ หรือการเดินสายโทรศัพท์ รวมถึงการประยุกต์ใช้ในสาขาวิชาปัญญาประดิษฐ์ (Artificial Intelligence) โครงสร้างทางเคมีของแก้วเจียรนัย (Crystals) วงจรไฟฟ้า (Electrical Circuitry) และการวิเคราะห์ของภาษาโปรแกรม เป็นต้น



ภาพที่ 1.2 การทำงานของกราฟ

3. โค้ดตัวอย่าง

a) แอวลำดับ (Array)

```
cars = ["Ford", "Volvo", "BMW"]
```

```
print(cars)
```

```
cars[0] = "Toyota"
```

```
print(\ncars[0])
```

Output: Ford Volvo BMW

Toyota

b) ลิงค์ลิสต์ (Linked list)

```
class Node:
    def __init__(self, dataval=None):
        self.dataval = dataval
        self.nextval = None

class SLinkedList:
    def __init__(self):
        self.headval = None

    def listprint(self):
        printval = self.headval
        while printval is not None:
            print (printval.dataval)
            printval = printval.nextval

list = SLinkedList()
list.headval = Node("Mon")
e2 = Node("Tue")
e3 = Node("Wed")

# Link first Node to second node
list.headval.nextval = e2

# Link second Node to third node
e2.nextval = e3

list.listprint()
```

Output: Mon

Tue

Wed

c) สแต็ค (Stack)

```
class Stack:

    def __init__(self):
        self.stack = []

    def add(self, dataval):
        # Use list append method to add element
        if dataval not in self.stack:
            self.stack.append(dataval)
            return True
        else:
            return False
        # Use peek to look at the top of the stack

    def peek(self):
        return self.stack[-1]

AStack = Stack()
AStack.add("Mon")
AStack.add("Tue")
AStack.peek()
print(AStack.peek())
AStack.add("Wed")
AStack.add("Thu")
print(AStack.peek())
```

Output: Tue

Thu

d) คิว (Queue)

```
class Queue:

    def __init__(self):
        self.queue = list()

    def addtoq(self,dataval):
        # Insert method to add element
        if dataval not in self.queue:
            self.queue.insert(0,dataval)
            return True
        return False

    def size(self):
        return len(self.queue)

TheQueue = Queue()
TheQueue.addtoq("Mon")
TheQueue.addtoq("Tue")
TheQueue.addtoq("Wed")
print(TheQueue.size())
```

Output: 3

e) Hash Table

```
# Declare a dictionary
dict = {'Name': 'Zara', 'Age': 7, 'Class': 'First'}

# Accessing the dictionary with its key
print "dict['Name']: ", dict['Name']
print "dict['Age']: ", dict['Age']
```

Output: dict['Name']: Zara

dict['Age']: 7

f) Tree

```
class Node:

    def __init__(self, data):

        self.left = None
        self.right = None
        self.data = data

    def insert(self, data):
# Compare the new value with the parent node
        if self.data:
            if data < self.data:
                if self.left is None:
                    self.left = Node(data)
                else:
                    self.left.insert(data)
            elif data > self.data:
                if self.right is None:
                    self.right = Node(data)
                else:
                    self.right.insert(data)
            else:
                self.data = data

# Print the tree
    def PrintTree(self):
        if self.left:
            self.left.PrintTree()
        print( self.data),
        if self.right:
            self.right.PrintTree()

# Use the insert method to add nodes
root = Node(12)
root.insert(6)
root.insert(14)
root.insert(3)

root.PrintTree()
```

Output: 3 6 12 14

g) graph

```
class graph:
    def __init__(self,gdict=None):
        if gdict is None:
            gdict = []
        self.gdict = gdict

# Get the keys of the dictionary
    def getVertices(self):
        return list(self.gdict.keys())

# Create the dictionary with graph elements
graph_elements = { "a" : ["b","c"],
                    "b" : ["a", "d"],
                    "c" : ["a", "d"],
                    "d" : ["e"],
                    "e" : ["d"]
                  }

g = graph(graph_elements)

print(g.getVertices())
```

Output: ['d', 'b', 'e', 'c', 'a']

4. การเอาไปประยุกต์ใช้

โดยทั่วไปแล้วโครงสร้างข้อมูลถูกนำไปใช้ในรูปแบบของข้อมูลชนิดนามธรรม (ADT: Abstract Data Type) โครงสร้างข้อมูลเป็นส่วนสำคัญในการพัฒนาซอฟต์แวร์ นอกจากนี้ยังมีบทบาทสำคัญในการออกแบบอัลกอริทึมและการนำอัลกอริทึมไปใช้กับโปรแกรม

ตัวอย่างของโครงสร้างข้อมูลที่ถูกนำไปใช้มีดังนี้

- 1) จัดเก็บข้อมูล - โครงสร้างข้อมูลถูกนำไปใช้กับ persistent data เช่น การเจาะจงไปยังกลุ่มข้อมูลที่คุณลักษณะใดลักษณะหนึ่ง หรือใช้จัดเก็บบันทึกในระบบจัดเก็บข้อมูล
- 2) เรียงลำดับและจัดหมวดหมู่ - โครงสร้างข้อมูลอย่างไบนารีทรี หรือรู้จักกันในชื่อทรีที่ถูกเรียงลำดับ มีฟังก์ชันที่จำเป็นในการจัดหมวดหมู่ของแอปพลิเคชัน ยกตัวอย่างเช่นอักษรหรือสตริง จากการใช้โครงสร้างข้อมูลเช่นทรีหรือคิว โปรแกรมเมอร์สามารถเรียงลำดับข้อมูลให้เรียงตามความสำคัญได้
- 3) การปรับขนาด - แอปพลิเคชันที่มีข้อมูลจำนวนมากใช้โครงสร้างข้อมูลในการจัดสรรและจัดการข้อมูลที่ถูกจัดเก็บในที่เก็บที่มีความกระจัดกระจายโดยไม่ทำให้ประสิทธิภาพลดน้อยลง

บทที่ 2

Unknown Bookstore

1. ประวัติความเป็นมา

สืบเนื่องจากสถานการณ์โควิด-19 ที่ได้แพร่ระบาดอย่างแพร่หลายทั่วประเทศไทยจนถึงทุกวันนี้ซึ่งยังคงย่ำแย่ลงต่อเนื่องเนื่องจากการจัดหาวัดขึ้นที่ล่าช้า ทำให้ประชาชนเป็นจำนวนมากต่างก็หวาดกลัวที่จะออกมาจับจ่ายใช้สอยสิ่งของเครื่องใช้ในบ้าน ดังนั้นการซื้อของผ่านทางช่องทางออนไลน์ ไม่ว่าจะเป็นแอปพลิเคชัน เช่น Shopee Lazada ฯลฯ เว็บไซต์ หรือตามโซเชียลมีเดียได้รับความนิยมเป็นอย่างสูงในช่วงการกักกันตัว ถึงกระนั้นหนังสือกลับเป็นหนึ่งในสินค้าที่ไม่ได้รับความนิยมในการสั่งซื้อออนไลน์มากนัก เหตุผลคือผู้อ่านมักจะกังวลกับการจัดส่งพัสดุของขนส่ง หรือการไม่ได้รับเคลมสินค้าจากทางร้านหากหนังสือชำรุดหรือไม่อยู่ในสภาพที่ดี รวมไปถึงการติดต่อหาพนักงานร้านค้าเพื่อขอความช่วยเหลือที่เป็นไปอย่างยากลำบาก

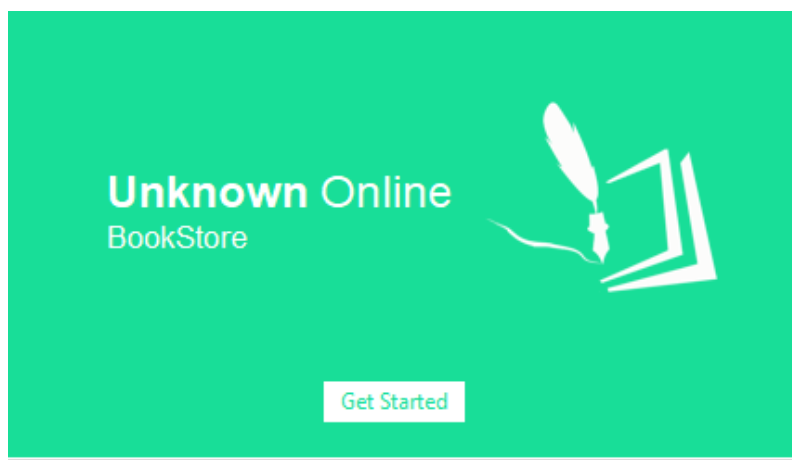
คณะผู้จัดทำจึงออกแบบแอปพลิเคชัน Unknown Bookstore เพื่อตอบสนองต่อความต้องการเหล่านี้ เพื่อให้ลูกค้ามั่นใจได้ว่าลูกค้าจะได้รับพัสดุอย่างรวดเร็ว ปลอดภัย และสามารถติดต่อพนักงานได้ทุกเมื่อหากต้องการความช่วยเหลือเกี่ยวกับสินค้า

2. จุดประสงค์

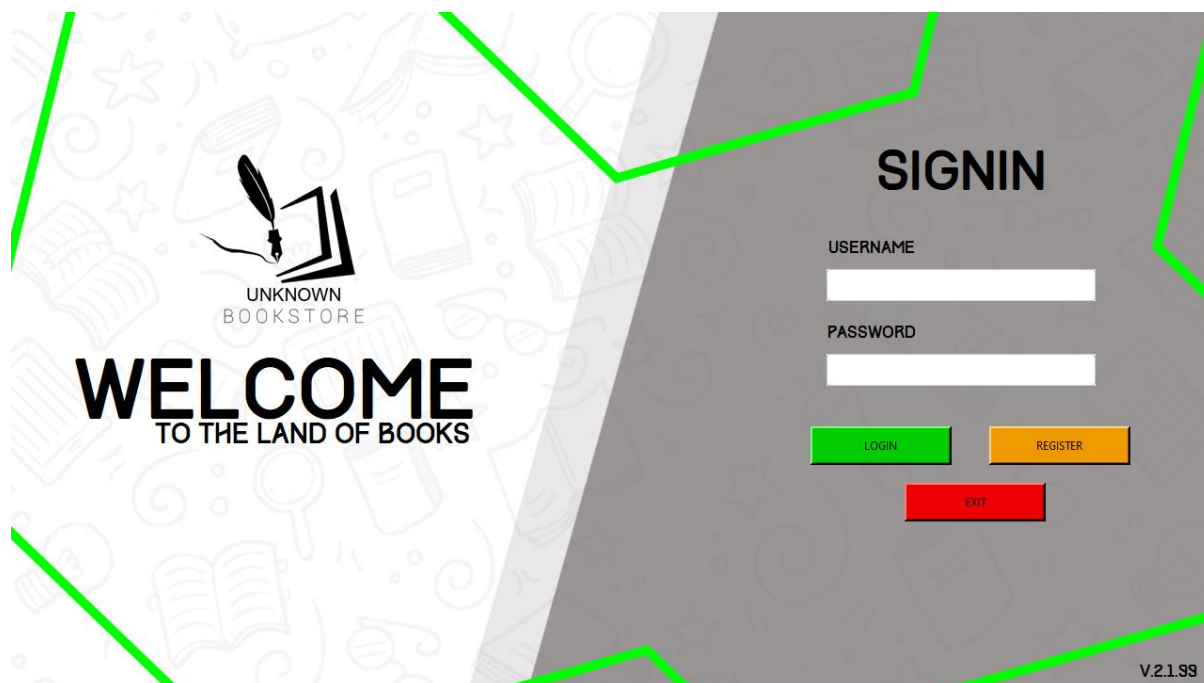
เพื่อตอบสนองความต้องการของผู้ใช้ที่ต้องการความสะดวก รวดเร็ว และปลอดภัยในการสั่งซื้อผ่านแอปพลิเคชัน

3. รูปแบบโปรแกรม และรายละเอียดการใช้งาน

1) หน้า Loading เข้าตัวโปรแกรม



2) หน้า Login เข้าสู่แอปพลิเคชัน



3) หน้าลงทะเบียนผู้ใช้ (กรณีไม่ได้เป็นสมาชิก)

Register

USER INPUT

USER

PASSWORD

CONFIRM PASSWORD

NAME

LAST NAME

GENDER
☐ MALE ☐ FEMALE

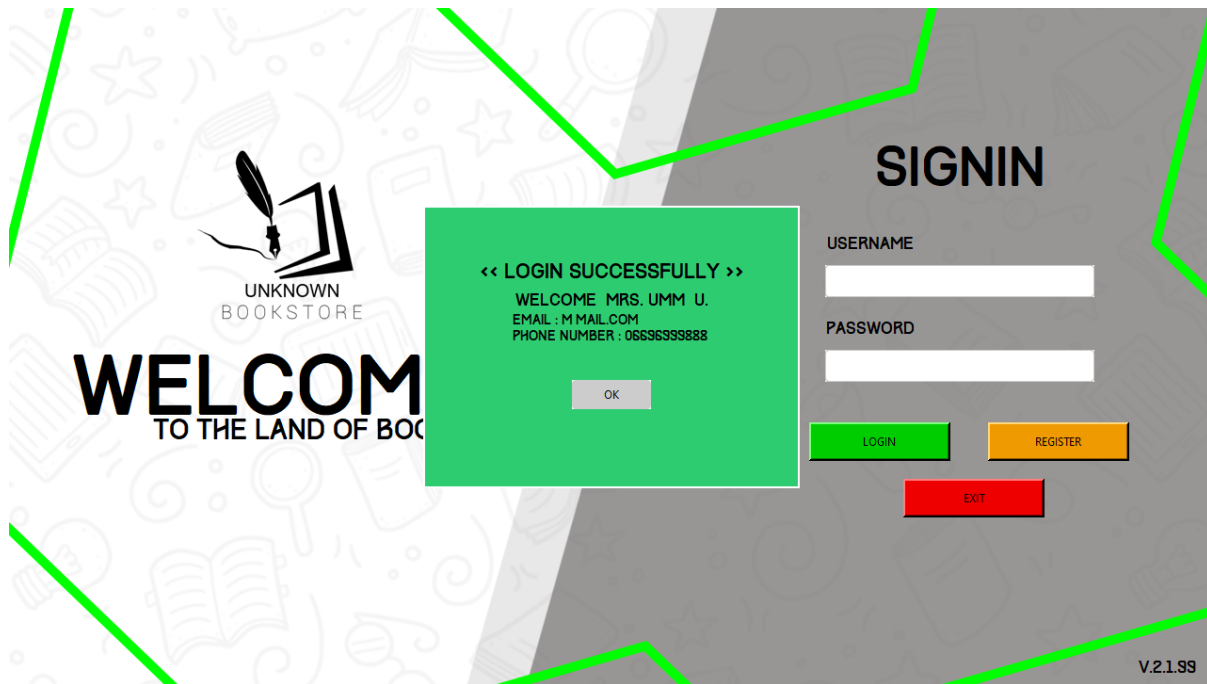
BIRTHDAY

PHONE NUMBER (+66)

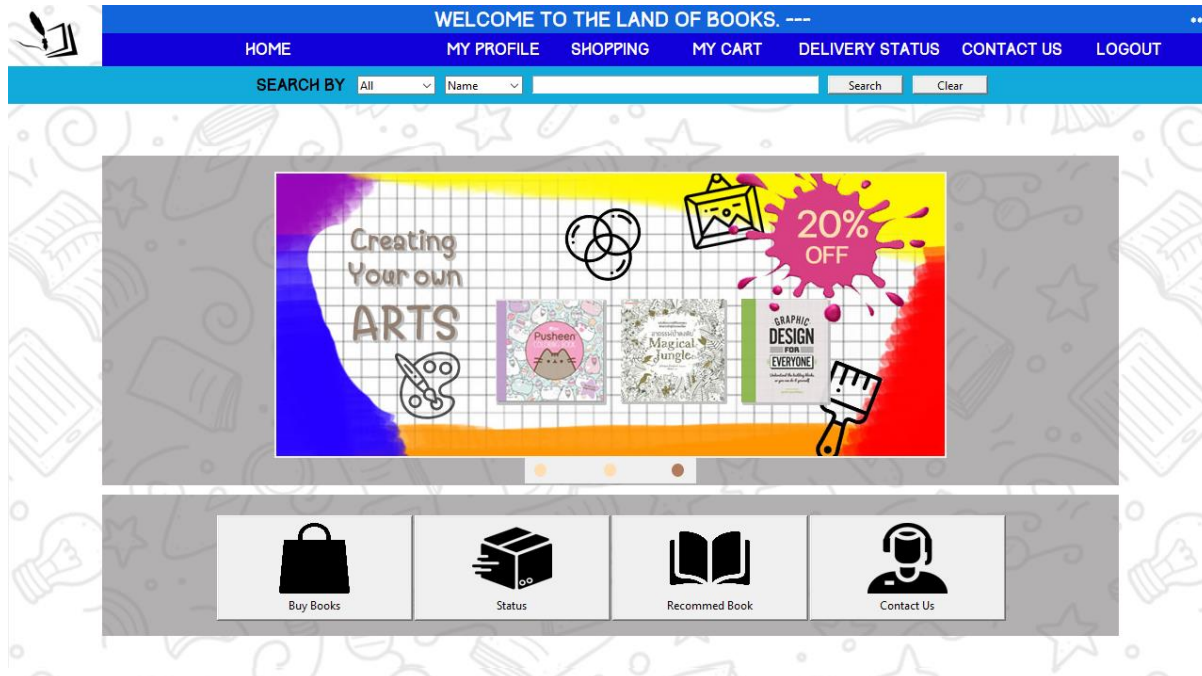
EMAIL ADDRESS

REGISTER CLEAR CANCEL

4) หน้าแสดงผลว่าผู้ใช้ได้เข้าสู่ระบบสำเร็จ



5) หน้าหลัก - หน้าแรกที่ผู้ใช้งานเจอเมื่อล็อกอินเข้าสู่แอปพลิเคชันแล้ว



6) หน้าแสดงรายละเอียดข้อมูลหนังสือแต่ละเล่ม

WELCOME TO THE LAND OF BOOKS. ---

HOME MY PROFILE SHOPPING MY CART DELIVERY STATUS CONTACT US LOGOUT

SEARCH BY Children's £ Name The Midnight Search Clear

from the NUMBER ONE bestselling author
David Walliams
Illustrated by Tony Stone
THE MIDNIGHT GANG

CODE CHL_WA009
NAME THE MIDNIGHT GANG
AUTHOR WALLIAMS, DAVID
CATEGORY CHILDREN'S BOOK
PRICE 286.0
RATING 5

WELCOME TO THE MIDNIGHT GANG! MIDNIGHT IS THE TIME WHEN A

1 +

No	Code	Name	Author	Category	Price
49	CHL_WA009	The Midnight Gang	Walliams, David /	Children's Books	286.0

7) หน้าเลือกประเภทหนังสือ โดยแบ่งตามสี ดังนี้

Orange - Arts / Design / Decoration, Blue – Literature, Green - Administration / Management,
Black - Humanities / Science, Gray - Children's Books, Red - Career Academic Textbooks, Purple
– Psychology

WELCOME TO THE LAND OF BOOKS. ---

HOME MY PROFILE SHOPPING MY CART DELIVERY STATUS CONTACT US LOGOUT

SEARCH BY Children's £ Name The Midnight Search Clear

-- CHOOSE TYPE --

ORANGE

8) หน้าแจกแจงออเดอร์และรวมยอดที่ผู้ใช้จะต้องจ่าย

WELCOME TO THE LAND OF BOOKS. ---

HOME MY PROFILE SHOPPING MY CART DELIVERY STATUS CONTACT US LOGOUT

SEARCH BY All Name the Search Clear

Welcome to the land of books

no.	Code	Name	Item (s)	Price	Total amount
1	ADM_CA010	STARTING A BUSINESS WHEN YOU'RE AFRAID TO: THE	1	441.75	441.75
2	CHI_MA008	The Wonderful Things You Will Be	4	503.0	503.0
3	ART_ZO005	MICHELANGELO: THE COMPLETE PAINTINGS, SCULPT	4	674.25	674.25
4	CHI_WA009	The Midnight Gang	6	286.0	286.0
5	HUM_HA004	Brief Answers to the Big Questions	1	453.0	453.0
6	PSY_HO014	Tao of Pooh & the Te of Piglet	1	500.0	500.0

< Edit > < Del > < Confirm Order > < Next >

RECEIPT

Sales : 2858.0

Member : -15.25 ฿

Promotion : -12.5 ฿

Shipping : 40.0 ฿

Total(s) : 0.00 ฿

Total(s) : 2870.25

< Payment > < Cancel Order >

9) หน้า Payment ให้ผู้ใช้เลือกวิธีการชำระค่าหนังสือและแจ้งที่อยู่จัดส่ง

WELCOME TO THE LAND OF BOOKS. ---

HOME MY PROFILE SHOPPING MY CART DELIVERY STATUS CONTACT US LOGOUT

SEARCH BY All Name the Search Clear

Payment

Detail

Address 228/66 Rimsa, Ochaya Rd., LA. USA 45000

Payment Method

Promptpay (selected)

Cash On Delivery

Promptpay

Upload File

Promptpay Upload

Preview

Cancel Place Order

Welcome to the lar

RECEIPT

Sales : 2858.0

Member : -15.25 ฿

Promotion : -12.5 ฿

Shipping : 40.0 ฿

Total(s) : 0.00 ฿

Total(s) : 2870.25

< Payment > < Cancel Order >

10) หน้าแสดงสถานะจัดส่งและให้คะแนนหนังสือที่ผู้ใช้ได้สั่งซื้อ

WELCOME TO THE LAND OF BOOKS. ---

HOME MY PROFILE SHOPPING MY CART DELIVERY STATUS CONTACT US LOGOUT

SEARCH BY All Name the Search Clear

Status

2021-05-27 23:08:18 : Payment confirmed
 2021-05-27 23:08:18 : Waiting for shipment
 2021-05-27 23:09:21 : Shipped
 2021-05-27 23:09:35 : Delivered

Button Status

< Back > < Next >

Review Book

Picture

STARTING A BUSINESS
 WHEN YOU'RE AFRAID TO
 MONEY

Databook

Code ADM_CA010
 Name STARTING A BUSINESS
 Author CAVALLARO, MICHAEL
 Category Administration / Mana
 Price 441.75
 Rating 5

COMMENT :

BEST BOOKS FOREVER. I LIKE SIA AUM MAKMAK.

Review Button

< Send > < Clear >
 < Review >

11) (ส่วนของแอดมิน) หน้าจัดการและติดตามสถานะออเดอร์

Unknown Book Store // Admin

2021-05-27 23:13:18

Order Management

Order Time	Order ID	Customer	Address	Quantity	Total Amount	Status
2021-05-22 18:14:20	4	Ummmmmmmm	123 LA srgrs	5	250	Cancelled order
2021-05-22 18:14:20	5	Ummmmmmmmmm	123 LA srgrs	5	250	Cancelled order
2021-05-23 18:14:20	6	Ummmmmmmmmm	123 LA srgrs	5	250	Cancelled order
2021-05-26 22:24:54	7	Ummmmmm	sdfvsdfsdfsdfsdcvxczvczvczvczsdvs	27	10501.93	Payment confirmed - Prorr
2021-05-26 22:27:05	8	Ummmmmm	erafgresfwaerf	2	870.0	Payment confirmed - Cash
2021-05-27 09:45:05	9	Ummmmmm	123	2	2113.25	Payment confirmed - Cash
2021-05-27 09:48:09	10	Ummmmmm	test	1	274.35	Payment confirmed - Cash
2021-05-27 09:49:25	11	Ummmmmm	a	1	367.35	Payment confirmed - Cash
2021-05-27 09:51:06	12	Ummmmmm	aa	1	553.35	Payment confirmed - Cash
2021-05-27 09:53:58	13	Ummmmmm	a	1	309.0	Payment confirmed - Cash
2021-05-27 09:55:52	14	Ummmmmm	00	1	1390.35	Payment confirmed - Cash
2021-05-27 10:41:27	15	Ummmmmm	aa	4	3591.95	Payment confirmed - Cash
2021-05-27 21:58:02	16	Ummmmmm	asdasdasdasf	13	PV_VAR10	Delivered
2021-05-27 23:08:18	17	Ummmmmm	228/66 Rimsa, Ochaya Rd., LA. USA 45000	17	2858.0	Delivered

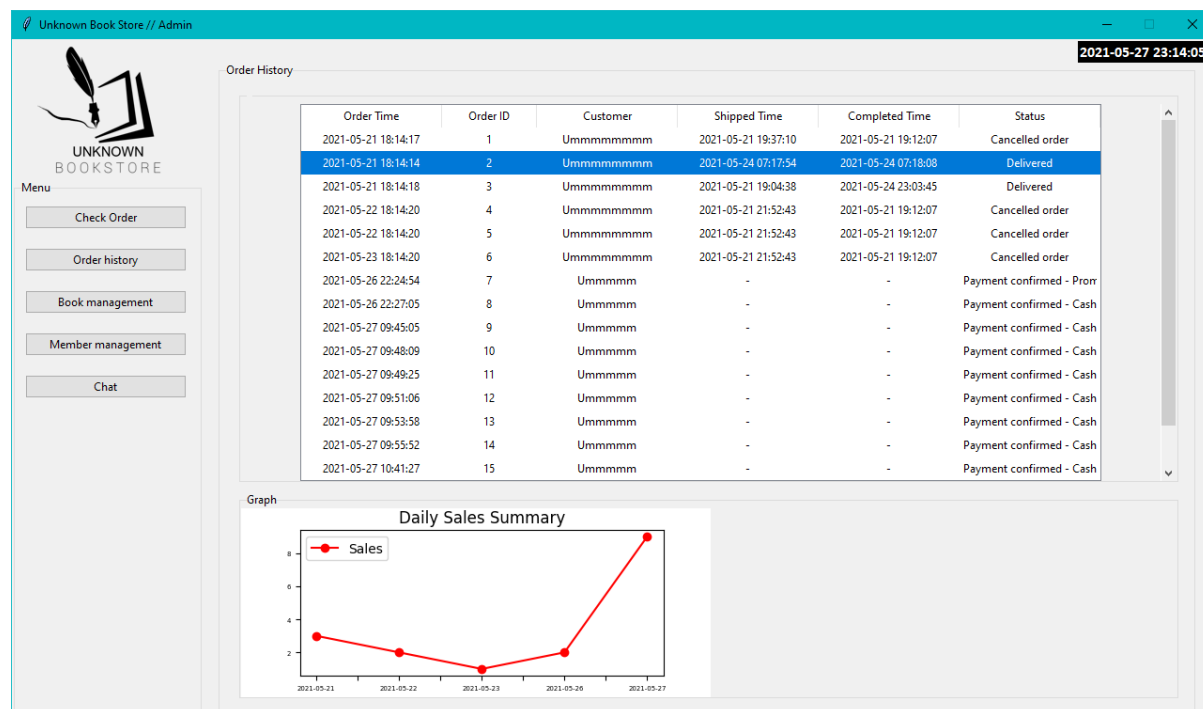
Details

Order ID# 17 Address: 228/66 Rimsa, Ochaya Rd., LA. USA 45000
 Customer: Ummmmmmmm
 Order Time: 2021-05-27 23:08:18
 Shipped Time: 2021-05-27 23:12:59 Order Status: Delivered
 Completed Time: 2021-05-27 23:13:14

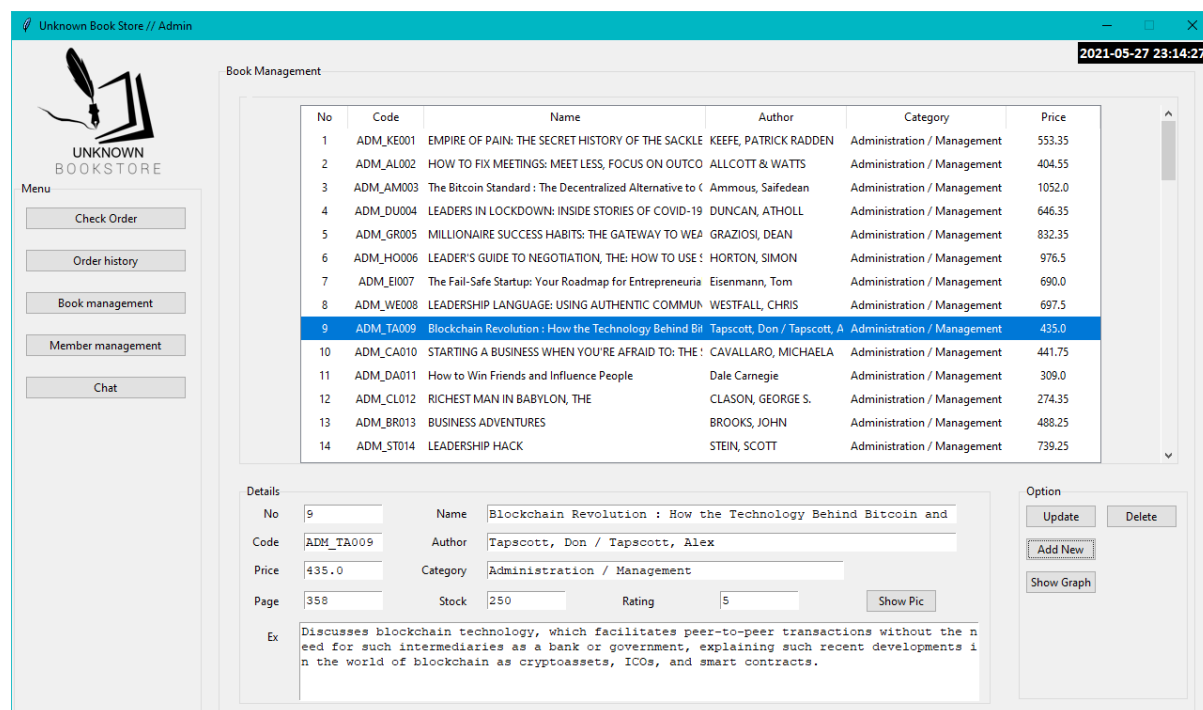
Option

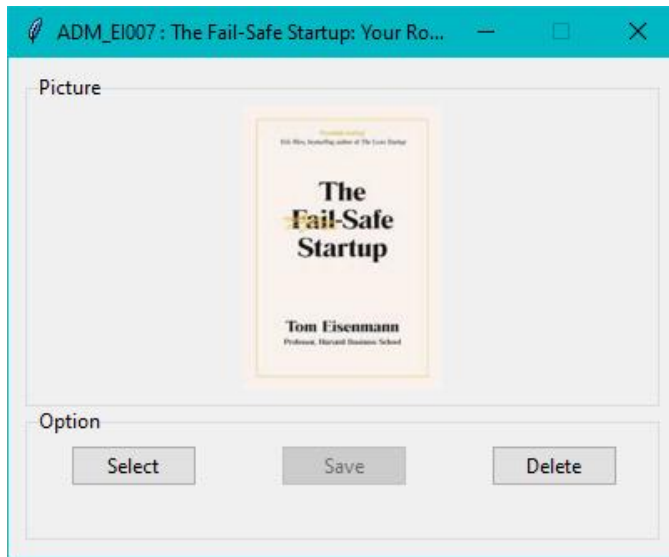
Update Order Details
 Order ID Search Order
 View Slip

12) (ส่วนของแอดมิน) หน้ารายงานผลการขายแต่ละวัน



13) (ส่วนของแอดมิน) หน้าอัพเดทชื่อ คัด และรายละเอียดของหนังสือ สามารถเพิ่ม ถอน และแก้ไขรายละเอียดหนังสือแต่ละเล่ม รวมถึงมีกราฟแสดงผลสัดส่วนของหนังสือแต่ละประเภท





Add Book

Picture

Book Details

No Name

Code Author

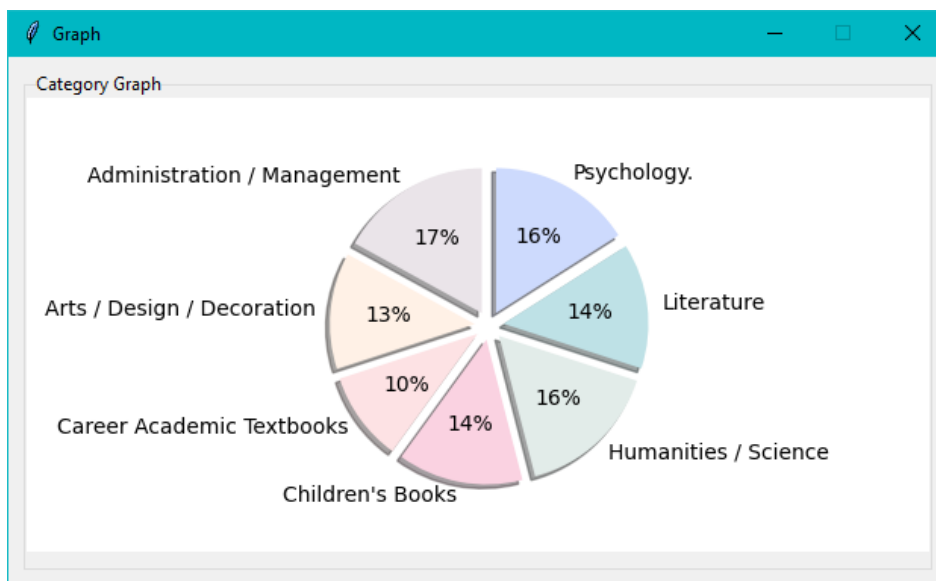
Price Category

Page Ex

Stock

Rating

Select Delete Clear Save



14) (ส่วนของแอดมิน) หน้าเพิ่มผู้ใช้ที่ได้ทำการสมัครเป็นสมาชิก

Unknown Book Store // Admin

2021-05-27 23:15:54

Member Management

Logged in	Username	Name	LastName	Gender	Birthday	Email	PhoneNumber
T	aaa	Ummmmmm	Umm	FEMALE	2/May/1925	m@mail.com	6696999888
T	pppppppp	Pluem	Zzzz	MALE	25/Aug/2000	p.apirak2000@hotmail.com	959495028

Menu

- Check Order
- Order history
- Book management
- Member management
- Chat

Details

Username: ppppppppp Gender: MALE

Password: ***** Birthday: 25/Aug/2000

Firstname: Pluem Email: p.apirak2000@hotmail

Lastname: Zzzz Telephone: 959495028

Option

Update Delete Register

15) หน้าแะระหว่างแอดมินและผู้ใช้

Unknown Book Store // Admin

2021-05-27 23:15:54

Member Management

Logged in	Username	Name	LastName	Gender	Birthday	Email	PhoneNumber
T	aaa	Ummmmmm	Umm	FEMALE	2/May/1925	m@mail.com	6696999888
T	pppppppp	Pluem	Zzzz	MALE	25/Aug/2000	p.apirak2000@hotmail.com	959495028

Menu

- Check Order
- Order history
- Book management
- Member management
- Chat

Details

Username: ppppppppp Gender: MALE

Password: ***** Birthday: 25/Aug/2000

Firstname: Pluem Email: p.apirak2000@hotmail

Lastname: Zzzz Telephone: 959495028

Option

Update Delete Register

4. การใช้โครงสร้างข้อมูลในการจัดสรร

1) Stack

- จัดเก็บและนำออกหนังสือที่ผู้ใช้ได้ทำการสั่งซื้อลงในตะกร้าสินค้า
- เก็บรีวิวหนังสือจากผู้ใช้
- อัปเดตสถานะจัดส่งเพื่อให้ผู้ใช้ติดตามสถานะสินค้าที่ตัวเองสั่ง

2) Sort

- จัดหมวดหมู่หนังสือ แบ่งเป็น 7 หมวดหมู่ดังนี้ Art/Decoration, Literature, Administration/Management, Humanities/Science, Children's Book, Career/Academic Textbook, Psychology
- จัดเรตติ้งที่ผู้ใช้ได้ให้คะแนนต่อหนังสืออื่นๆ
- จัดเรียงคะแนน membership ของสมาชิกแต่ละคน เพื่อหาสมาชิกที่มีแต้มสูงที่สุด

3) Search

- ค้นหาหนังสือแบ่งตามประเภท ชื่อ ผู้เขียน และอื่นๆ

4) Linked list

- ตกแต่งหน้าแอปพลิเคชัน
- เลือกหมวดหมู่ของหนังสือทั้ง 7 หมวดหมู่

5) Recursive

- คำนวณรวมของสินค้าทั้งหมดที่ผู้ใช้ทำการสั่งซื้อเพื่อออกใบเสร็จและแสดงผลราคาที่ใช้ต้องจ่าย

5. โค้ดตัวอย่างของฟังก์ชันหลัก

1) Stack

```
class Stack:
    def __init__(self):
        self.items = []
    def push(self,data):
        self.items.append(data)
    def pop(self):
        return self.items.pop()
    def __str__(self):
        return str(self.items)
    def isEmpty(self):
        return self.items == []
    def peek(self):
        return self.items[len(self.items)-1]
    def size(self):
        return len(self.items)
self.order_to_review = Stack()
for i in reversed(order_detail):
    if str(i[0]) == str(self.order_id_review):
        self.order_to_review.push(i[1])
    if i[5] != '' or i[6] != '':
        self.Next_bottomn.config(state = NORMAL)
```

2) Sort

```
def QuickSort(arr):
    elements = len(arr)
    #Base case
    if elements < 2:
        return arr
    current_position = 0
    for i in range(1, elements):
        if arr[i] > arr[0]:
            current_position += 1
            temp = arr[i]
            arr[i] = arr[current_position]
            arr[current_position] = temp
    temp = arr[0]
    arr[0] = arr[current_position]
    arr[current_position] = temp
    left = QuickSort(arr[0:current_position])
    right = QuickSort(arr[current_position+1:elements])
    arr = left + [arr[current_position]] + right
    return arr

loadorder_deatil = pandas.read_csv('UnknownShop\\database\\order_detail.csv')
order_detail = loadorder_deatil.values.tolist()
to_sort = []
duplicate = False
for i in order_detail:
    if str(i[5]) != 'nan':
        for j in to_sort:
            if i[1] in j[1]:
                j[0] = (float(j[0]) + float(i[5]))/2
                round(j[0],2)
                duplicate = True
            if not duplicate:
                to_sort.append([i[5],i[1],i[2]])
recommend_sorted = QuickSort(to_sort)
```



```

def search(arr, n, x):
    for i in range(0, n):
        if (arr[i] == x):
            return i
    return -1

# Driver Code
arr = [2, 3, 4, 10, 40]
x = 10
n = len(arr)

# Function call
result = search(arr, n, x)
if(result == -1):
    print("Element is not present in array")
else:
    print("Element is present at index", result)

```

4) Linked list

```

class Node:
    def __init__(self, data):
        self.next = None
        self.prev = None
        self.data = data

class CircularLinkedList:
    def __init__(self):
        self.head = None

    def is_empty(self):
        return self.head is None

    def add(self, data):
        if self.head is None:
            new_node = Node(data)
            new_node.next = new_node.prev = new_node
            self.head = new_node
            return

        last = self.head.prev
        new_node = Node(data)

        new_node.next = self.head
        self.head.prev = new_node
        new_node.prev = last
        last.next = new_node

```

```

self.ll = CircularLinkedList()
self.ls = CircularLinkedList()
self.ls.add('orange')
self.ls.add('blue')
self.ls.add('green')
self.ls.add('black')
self.ls.add('gray')
self.ls.add('red')
self.ls.add('purple')
self.ll.add('UnknownShop/Picture/ShopPage/orange.jpg')
self.ll.add('UnknownShop/Picture/ShopPage/blue.png')
self.ll.add('UnknownShop/Picture/ShopPage/green.jpg')
self.ll.add('UnknownShop/Picture/ShopPage/black.png')
self.ll.add('UnknownShop/Picture/ShopPage/gray.jpg')
self.ll.add('UnknownShop/Picture/ShopPage/red.png')
self.ll.add('UnknownShop/Picture/ShopPage/purple.png')

```

5) Recursive

```
def listsum(numList):
    if len(numList) == 1:
        return float(numList[0][4])
    else:
        return float(numList[0][4]) + listsum(numList[1:])
if self.usercart != []:
    temp = round(listsum(self.usercart),2)
    self.total_amount.set(temp)
```

```
def listsum(numList):
    if len(numList) == 1:
        return float(numList[0][4])
    else:
        return float(numList[0][4]) + listsum(numList[1:])
```


บทที่ 3

การทำงานในกลุ่ม

1. การวางแผนและดำเนินงาน

วันที่	รายการ	กำหนดส่ง	ผู้รับผิดชอบ	หมายเหตุ
12/04/2021	ลงข้อมูล DATA หนังสือแต่ละประเภท	13/04/2021	ทุกคน	
13/04/2021	หน้า login register - เคลียร์และจัดโฟลเดอร์ หน้า Shop - รูปแบบ การออกแบบ - แถบเมนู - ปุ่มต่างๆ	16/04/2021	ทุกคน	Login + register -> อภिरักษ์, ณพวัฒน์ Shop -> กนิษฐา ชวกร นภาพร
16/04/2021	ดึงฐานข้อมูลมาแสดง ลิงค์หน้าเชื่อมต่อกัน	19/04/2021	ณพวัฒน์ อภिरักษ์ ชวกร	
19/04/2021	ทำหน้า Shop - ออกแบบ UI - เขียนตัวโปรแกรม	23/04/2021	ทุกคน	ตัวโปรแกรม-> ณพวัฒน์ ชวกร อภिरักษ์ UI -> อภिरักษ์ กนิษฐา นภาพร
23/04/2021	[Font] ทำหน้า UXUI ทุกหน้า [Back] ดึง Data Base + เปลี่ยนโค้ด โปรแกรมเป็นแบบคลาส	26/04/2021	ทุกคน	
26/04/2021	แก้ไขและจัดเรียงโฟลเดอร์ใหม่ แก้ไข UI อีกรอบ	27/04/2021	ทุกคน	

27/04/2021	แก้ไขการดึงข้อมูลจาก Data base ใหม่ เปลี่ยนข้อมูลหนังสือและรูปเป็นภาษาอังกฤษทั้งหมด	03/05/2021	ทุกคน	ดึงข้อมูล -> อภิรักษ์ ชวกร ณพวัฒน์ แก้ไขข้อมูล -> นภาพร กนิษฐา
03/05/2021	เพิ่มหนังสือให้ครบ 100 เล่ม หน้าแชทติดต่ ออกแบบ UI ต่อ ทำให้ผู้ใช้ซื้อสินค้าได้แล้วยังตาราง หน้า payment แสดงรายละเอียดออ เดอร์ แอปพลิเคชันส่วนของแอดมิน	07/05/2021	ทุกคน	เพิ่มหนังสือ+UI -> นภาพร กนิษฐา หน้าแชท -> อภิรักษ์ Payment -> ชวกร ส่วนแอดมิน -> ณพวัฒน์
07-18/05/2021	หยุดพักเตรียมสอบปลายภาค			
19/05/2021	แก้ไข UI อีกรอบ (ล่าสุด) เพิ่มโค้ดส่วน Sort, Search แก้บัคต่างๆในโปรแกรม	24/05/2021	ทุกคน	แก้ไข UI -> อภิรักษ์ กนิษฐา นภาพร Sort, Search -> ชวกร ณพวัฒน์ อภิรักษ์
22/05/2021	เตรียมทำ PowerPoint	27/05/2021	นภาพร	
25/05/2021	ทำให้ผู้ใช้ติดต่อกับแอดมินในหน้า แชท	27/05/2021	อภิรักษ์	
26/05/2021	หาข้อมูลและจัดทำรายงาน	27/05/2021	ทุกคน	
26/05/2021	แก้ไขแอปพลิเคชันในส่วนที่ไม่ สมบูรณ์และเกิดบัค	27/05/2021	ชวกร ณพวัฒน์ อภิรักษ์	
27/05/2021	ตรวจเช็คก่อนส่งไฟล์งาน	27/05/2021	ทุกคน	

2. การมีส่วนร่วมในกลุ่ม

หมายเหตุ: [4]=เป็นกำลังสำคัญของทีม; [3]=ให้ความร่วมมือสม่ำเสมอ; [2]=ช่วยงานบางครั้ง; [1]=ไม่ตั้งใจ
ช่วยงาน; [0]=ไม่ช่วยงาน

1) นางสาวกนิษฐา ศศิปรียจันทร์ รหัส 61010014

	4	3	2	1	0
การวางแผน		✓			
การวิเคราะห์ Requirements		✓			
การออกแบบระบบ		✓			
การพัฒนาระบบ			✓		
การทดสอบระบบ			✓		
งานเอกสาร		✓			

2) นายชวกร เหลาแก้ว รหัส 62010175

	4	3	2	1	0
การวางแผน		✓			
การวิเคราะห์ Requirements		✓			
การออกแบบระบบ		✓			
การพัฒนาระบบ	✓				
การทดสอบระบบ		✓			
งานเอกสาร		✓			

3) นายณพวัฒน์ คำยอด รหัส 62010243

	4	3	2	1	0
การวางแผน		✓			
การวิเคราะห์ Requirements		✓			
การออกแบบระบบ		✓			
การพัฒนาระบบ	✓				
การทดสอบระบบ	✓				
งานเอกสาร		✓			

4) นางสาวนภาพร ตั้งใจ รหัส 62010462

	4	3	2	1	0
การวางแผน		✓			
การวิเคราะห์ Requirements		✓			
การออกแบบระบบ		✓			
การพัฒนาระบบ			✓		
การทดสอบระบบ		✓			
งานเอกสาร		✓			

5) นายอภิรักษ์ อูลิศ รหัส 62011019

	4	3	2	1	0
การวางแผน	✓				
การวิเคราะห์ Requirements		✓			
การออกแบบระบบ	✓				
การพัฒนาระบบ		✓			
การทดสอบระบบ		✓			
งานเอกสาร	✓				

3. ปัญหาที่เกิดขึ้นภายในกลุ่ม

- 1) การทำงานผ่านช่องทางออนไลน์ทำให้เกิดปัญหาในการสื่อสาร อัปโหลดโค้ดชนกันหรือแม้กระทั่งต่างคนต่างทำงานในส่วนเดียวกันโดยไม่รู้ว่ามีเพื่อนอีกคนก็ทำส่วนนี้อยู่
- 2) ไม่สามารถทำตามแผนการทำงานได้เนื่องจากเกิดปัญหาในตัวโปรแกรมและบางขั้นตอนใช้เวลานานในการจัดทำ รวมถึงภาระงานวิชาอื่นทำให้คณะผู้จัดทำมีเวลาน้อยในการพัฒนาแอปพลิเคชัน

4. แนวทางการแก้ไข

- 1) คณะผู้จัดทำได้ใช้ระบบ GitHub Desktop ที่สามารถแชร์ไฟล์โค้ด เอกสาร รูปภาพและอื่นๆที่จำเป็นต่อการพัฒนาแอปพลิเคชัน โดยผู้ที่อัปโหลดเข้าระบบจะอัปเดตผ่านกรุปแชทเพื่อแจ้งให้สมาชิกคนอื่นๆทราบว่าตนได้อัปโหลดอะไรเข้าไปและให้ที่เหลือดาวน์โหลดเข้ามายังคอมพิวเตอร์ของตนเอง
- 2) ขยายระยะเวลาการจัดทำ และมีการเรียกประชุมในดิสคอร์ดเพื่ออัปเดตงานแต่ละส่วนอย่างสม่ำเสมอ

5. ปัญหาที่เกิดขึ้นของตัวโปรแกรม

- 1) โปรแกรมมีความล่าช้าเนื่องจากไฟล์ที่ใช้โหลดมีขนาดใหญ่
- 2) โค้ดในส่วนของUX/UIทำตามที่ยกแบบได้ไม่สมบูรณ์เนื่องจาก tkinter ไม่รองรับ

6. แนวทางการแก้ไขตัวโปรแกรม

- 1) ใช้แอปพลิเคชันหรือเว็บไซต์ที่สามารถย่อขนาดไฟล์ให้เล็กลง
- 2) ใช้ Photoshop หรือโปรแกรมตัดต่อภาพอื่นๆเพื่อค้นหาหรือตัดแต่งภาพในส่วนของUIแทนการใช้โค้ดของtkinter

7. แนวคิดต่อยอดของโปรแกรม

คณะผู้จัดทำมีแนวทางต่อยอดโปรแกรมให้การยกเลิกและแก้ไขออเดอร์ที่ส่งไปแล้วเป็นไปได้อย่างอิสระ โดยได้รับแรงบันดาลใจจากแอปพลิเคชันสั่งซื้อสินค้าออนไลน์อย่าง Shopee ที่ผู้ใช้สามารถยกเลิกออเดอร์เพื่อสั่งใหม่ได้โดยทางแอปจะคืนเงินให้เต็มจำนวน

นอกจากนี้เพื่อลดภาระงานในส่วนของแอดมิน คณะผู้จัดทำมีแนวคิดที่จะทำให้สถานะการส่งอัปเดตอย่างอัตโนมัติโดยที่พนักงานส่วนจัดส่งไม่ต้องพึ่งพาแอดมิน ซึ่งเป็นไปได้ว่าต้องจัดทำแอปพลิเคชันในส่วน of พนักงานจัดส่งเพิ่มเติม

8. สรุปผลการทำงาน

แอปพลิเคชัน Unknown Bookstore ถูกจัดทำขึ้นตรงตามวัตถุประสงค์ที่ได้คาดหวังไว้ หากแต่มีปัญหาในส่วน of โปรแกรมและUIเพียงเล็กน้อย ซึ่งถ้ามีเวลาและโอกาสคณะผู้จัดทำจะปรับปรุงและแก้ไขให้แอปพลิเคชันทำงานได้อย่างสมบูรณ์แบบรวมถึงต่อยอดให้สะดวกต่อการใช้งานมากขึ้น

บทที่ 4

เอกสารที่เกี่ยวข้อง

1. Tkinter

Tkinter GUI framework เป็นไลบรารีที่บิวท์อินอยู่ในตัวไพทอนเวอร์ชัน 4.0 และใช้ได้หลายแพลตฟอร์มทั้งวินโดวส์ แมค และลินุกซ์ และเวอร์ชัน Tkinter จะเรนเดอร์ให้แอปพลิเคชันเข้ากับระบบปฏิบัติการที่ผู้ใช้งานใช้อยู่ แต่ถึงกระนั้น Tkinter ก็มีข้อเสียในเรื่องของระบบ GUIs ที่ล้าหลัง ดังนั้นหากผู้พัฒนาโปรแกรมต้องการสร้างโปรแกรมที่ทันสมัย GUI อื่นอาจจะเป็นทางเลือกที่เหมาะสมกว่า

อย่างไรก็ตาม Tkinter กินพื้นที่จัดเก็บน้อยและส่งผลกระทบต่อคอมพิวเตอร์น้อยกว่า GUI framework ตัวอื่น ทำให้เป็นตัวเลือกรายแรกที่ผู้ใช้เลือกใช้ในการพัฒนาแอปพลิเคชันที่ไม่ต้องการการตกแต่งที่ทันสมัยมากนัก และให้ความสำคัญกับฟังก์ชันในตัวโปรแกรมมากกว่า

A) สร้างแอปพลิเคชันด้วย Tkinter

โครงสร้างพื้นฐานแรกของ Tkinter คือ window โดย window จะประกอบไปด้วย กล่องข้อความ ปุ่มกด และอื่นๆ ส่วนนี้จะถูกเรียกว่า widgets และ widgets จะอยู่ใน window อีกทีหนึ่ง

ตัวอย่างโค้ด:

```
import tkinter as tk #อิมพอร์ตtkinterเข้ามาใช้งานในตัวแปร tk
```

```
window = tk.Tk() #เรียกใช้windowผ่านตัวแปร window
```

B) เริ่มสร้าง widgets

Widgets เป็นส่วนสำคัญของการพัฒนาโปรแกรมและสร้าง UI เนื่องจากเป็นส่วนที่ผู้ใช้ใช้งานผ่านทางแอปพลิเคชันโดยตรง โดยแต่ละ widgets จะถูก tkinter แยกออกเป็นคลาส ตัวอย่างเช่น

Label: แสดงผลข้อความผ่านหน้าจอ

Button: ปุ่มที่ใส่ข้อความลงไปและแสดงผลต่างๆตอนที่ถูกกดใช้งาน

Entry: widgetที่สามารถใส่ข้อความได้เพียง1แถว

Text: widgetที่สามารถใส่ข้อความได้มากกว่า1แถว

Frame: พื้นที่ที่เปลี่ยนมุมมองจากที่สามารถใส่ widget ที่เกี่ยวข้องและขยายผลไป widget อื่นๆ ได้

ตัวอย่างโค้ด:

```
label = tk.Label( #เรียกใช้ Label ผ่านตัวแปร label

    text="Hello Tkinter" #กำหนด text ที่จะแสดง

    foreground="white" #กำหนดสี text

    background="black" #กำหนดสีพื้นหลัง

)
```

2. การใช้งาน csv

CSV (Comma-separated values) คือรูปแบบไฟล์เก็บข้อมูลที่เป็นรูปแบบตาราง เช่น spread sheet และ database ไฟล์ในรูปแบบ CSV สามารถนำเข้าและส่งออกจากโปรแกรมที่เก็บข้อมูลเป็นตาราง เช่น Excel, OpenOffice, Calc.

A) จุดเด่นของไฟล์ CSV

- รองรับการใช้งานกับโปรแกรมฐานข้อมูลต่างๆ รวมทั้ง Microsoft Excel
- ไฟล์ที่ได้มีขนาดเล็กมาก
- รองรับการเปิดไฟล์ด้วยโปรแกรม Text Editor รวมทั้ง Microsoft Word

B) การใช้งาน csv ผ่าน Microsoft Excel

การสร้างไฟล์ csv ผ่าน Microsoft Excel ทำได้โดยการเปิด Microsoft Excel ขึ้นมาและเปิดไฟล์ข้อมูลที่ต้องการแปลง (ทางคณะผู้จัดทำจัดเก็บข้อมูลใน Google Sheet) และเซฟให้รูปแบบไฟล์เป็น .csv

3. เว็บไซต์ร้านหนังสือต้นแบบ

- UI: <https://www.betterworldbooks.com/>
- แบนเนอร์+หมวดหมู่หนังสือ: <https://www.asiabooks.com/>
- รายชื่อหนังสือ: <https://www.asiabooks.com/> และ <https://thailand.kinokuniya.com/>

- รายละเอียดหนังสือแต่ละเล่ม: <https://www.naiin.com/>

4. อ้างอิง

David Loshin. (2021). Data Structure. Retrieved 27 May 2021 from <https://searchsqlserver.techtarget.com/definition/data-structure#:~:text=How%20are%20data%20structures%20used,are%20used%20within%20computer%20programs>

David Amos. (2020). Python GUI Programming With Tkinter. From <https://realpython.com/python-gui-tkinter/>

ShadowCal20017. (2021). Tkinter. Retrieved 27 May 2021 from <https://wiki.python.org/moin/TkInter>

Python Software Foundation. (2021). tkinter – Python interface to Tcl/Tk. Retrieved 27 May 2021 from <https://docs.python.org/3/library/tkinter.html>

ไอที 24 ชั่วโมง. (2018). ไฟล์ CSV คืออะไร ใช้ทำอะไรได้บ้าง และเปิดอย่างไร. สืบค้นเมื่อวันที่ 27 พ.ค. 2564, จาก <https://www.it24hrs.com/2018/what-is-csv-file-import-contact-csv/>

Computer Hope. (2021). How to create CSV file. Retrieved 27 May 2021 from <https://www.computerhope.com/issues/ch001356.htm#:~:text=CSV%20is%20a%20simple%20file,%20comma%20separated%20values%22>