

ภาคผนวก

ภาคผนวก ก

**คู่มือและวิธีการใช้งานซอฟต์แวร์การหาเส้นทางที่สั้นที่สุดของการหยิบสินค้า
ตามใบสั่งในคลังสินค้าด้วยข้อบังคับระยะทางและน้ำหนัก**

การออกแบบหน้าจอของระบบ

หน้าจอหลักของระบบ มี 2 สิทธิการใช้งาน คือ หน้าจอสิทธิการใช้งานของผู้ใช้งาน และ หน้าจอสิทธิการใช้งานของแอดมิน

1. หน้าสิทธิการใช้งานของผู้ใช้งาน



ภาพที่ ก.1 หน้าสิทธิการใช้งานของผู้ใช้งาน

จากภาพที่ ก.1 หน้าสิทธิการใช้งานของผู้ใช้งาน หน้านี้คือหน้าที่จำเป็นที่สุดที่ผู้ใช้งานต้องกำหนดรายละเอียดการสั่งซื้อสินค้าก่อน เพื่อจะได้บันทึกรายละเอียดการสั่งซื้อที่ผู้ใช้งานได้กำหนดไว้ไปแสดงในหน้าข้อมูลการสั่งซื้อเพื่อนำไปแสดงเส้นทาง ซึ่งมีทั้งหมด 2 ประเภทการใช้งาน ดังนี้

1.1 หน้าจอแสดงส่งสินค้า

| ชื่อบริษัท | ชื่อสินค้า | น้ำหนัก | จำนวนสินค้า | ราคา | โซน | จำนวนที่ซื้อ | ราคา | วันที่ |
|------------|------------|---------|-------------|------|-----|--------------|------|--------|
| | | | | | | | | |

ภาพที่ ก.2 หน้าส่งสินค้า

1.2 หน้าจอแสดงรายละเอียดข้อมูลการสั่งซื้อสินค้า

เมนู - user

สั่งซื้อสินค้า ข้อมูลการสั่งซื้อสินค้า **ออกจากระบบ**

ข้อมูลการสั่งซื้อสินค้า

รายละเอียดการสั่งซื้อ ข้อมูลการสั่งซื้อ

| เลขที่สั่งซื้อ | ชื่อรหัสสินค้า | ชื่อสินค้า | น้ำหนัก | จำนวน สินค้า | ราคา สินค้า | โซน | จำนวน สั่งซื้อ | ราคา รวม | วันที่ |
|----------------|----------------|------------------------------|---------|-----------------|----------------|-----|-------------------|-------------|-----------------------|
| 400031 | 300007 | หมาป่า | 60 | 995 | 100 | A1 | 1 | 100 | Sunday, April 30, ... |
| 400031 | 300004 | ข้าวสาร | 1000 | 994 | 72 | B2 | 1 | 72 | Sunday, April 30, ... |
| 400031 | 300004 | ข้าวสาร | 1000 | 993 | 72 | B2 | 1 | 72 | Sunday, April 30, ... |
| 400031 | 300006 | ผงซักฟอก BREEZE เพลารจ... | 3000 | 980 | 135 | C2 | 1 | 135 | Sunday, April 30, ... |
| 400032 | 300001 | จอยสติ๊กเกที | 250 | 963 | 24 | A1 | 1 | 24 | Sunday, April 30, ... |
| 400032 | 300004 | ข้าวสาร | 1000 | 992 | 72 | B2 | 1 | 72 | Sunday, April 30, ... |
| 400032 | 300008 | หมอลิ้น | 50 | 999 | 85 | B1 | 1 | 85 | Sunday, April 30, ... |
| 400032 | 300003 | เคสม้า ตุ๊กกี้ออลใส่ รสมลไม้ | 380 | 979 | 110 | B1 | 1 | 110 | Sunday, April 30, ... |
| 400033 | 300001 | จอยสติ๊กเกที | 250 | 962 | 24 | A1 | 1 | 24 | Tuesday, May 2, ... |
| 400033 | 300004 | ข้าวสาร | 1000 | 991 | 72 | B2 | 1 | 72 | Tuesday, May 2, ... |
| 400033 | 300007 | หมาป่า | 60 | 994 | 100 | A1 | 1 | 100 | Tuesday, May 2, ... |
| 400033 | 300004 | ข้าวสาร | 1000 | 990 | 72 | B2 | 1 | 72 | Tuesday, May 2, ... |
| 400033 | 300006 | ผงซักฟอก BREEZE เพลารจ... | 3000 | 979 | 135 | C2 | 1 | 135 | Tuesday, May 2, ... |

ภาพที่ ก.3 หน้าข้อมูลการสั่งซื้อสินค้า

2. หน้าสิทธิการใช้งานของแอดมิน



ภาพที่ ก.4 หน้าโปรแกรมสิทธิการใช้งานของแอดมิน

จากภาพที่ ก.4 หน้าสิทธิการใช้งานของแอดมิน หน้านี้คือหน้าที่จำเป็นที่สุดที่ผู้ใช้งานต้องกำหนดรายละเอียดการสั่งซื้อสินค้าก่อน เพื่อจะได้บันทึกรายละเอียดการสั่งซื้อที่ผู้ใช้งานได้กำหนดไว้ไปแสดงในหน้าแสดงเส้นทาง ซึ่งมีทั้งหมด 4 ประเภทการใช้งาน ดังนี้

2.1 หน้าจอแสดงรายข้อมูลตั้งสินค้า

เมนู - admin

ข้อมูลการตั้งสินค้า แนะนำเส้นทาง สมุดรสมาชิก คลังสินค้า **ออกจากระบบ**

ข้อมูลการตั้งสินค้า

รายละเอียดการสั่งซื้อ ข้อมูลการสั่งซื้อ

| เลขที่สั่งซื้อ | วันที่สั่งซื้อ | ยอดรวม | สถานะ |
|----------------|-----------------------------|--------|-----------------|
| 400020 | Sunday, February 19, 2017 | 1408 | ลบข้อมูล |
| 400021 | Saturday, February 25, 2017 | 643 | ลบข้อมูล |
| 400022 | Saturday, February 25, 2017 | 681 | ลบข้อมูล |
| 400023 | Saturday, February 25, 2017 | 634 | ลบข้อมูล |
| 400024 | Saturday, February 25, 2017 | 305 | ลบข้อมูล |
| 400025 | Saturday, February 25, 2017 | 408 | ลบข้อมูล |
| 400026 | Saturday, April 29, 2017 | 436 | จัดการลบเส้นทาง |
| 400027 | Saturday, April 29, 2017 | 978 | จัดการลบเส้นทาง |
| 400028 | Sunday, April 30, 2017 | 166 | จัดการลบเส้นทาง |
| 400029 | Sunday, April 30, 2017 | 255 | จัดการลบเส้นทาง |
| 400030 | Sunday, April 30, 2017 | 947 | จัดการลบเส้นทาง |
| 400031 | Sunday, April 30, 2017 | 403 | จัดการลบเส้นทาง |
| 400032 | Sunday, April 30, 2017 | 291 | จัดการลบเส้นทาง |
| 400033 | Tuesday, May 2, 2017 | 403 | จัดการลบเส้นทาง |

ภาพที่ ก.5 หน้าข้อมูลการตั้งสินค้า

2.2 หน้าจอแสดงแนะนำเส้นทาง

เมนู - admin

ข้อมูลการตั้งสินค้า **แนะนำเส้นทาง** สมุดรสมาชิก คลังสินค้า ออกจากระบบ

แนะนำเส้นทาง

เลขที่สั่งซื้อ: 400033 นำหนักรวม: 5310 ระยะทางรวม: 5 พลังงาน: 80834.4 **ลบข้อมูล**

| เลขที่สั่งซื้อ | ชื่อจุดสินค้า | ชื่อสินค้า | น้ำหนัก | จำนวนสินค้า | ราคาสินค้า | โซน | จำนวนตู้ | ราคา | วันที่ |
|----------------|---------------|------------------------------|---------|-------------|------------|-----|----------|------|---------------------|
| 400033 | 300007 | มะม่วง | 60 | 994 | 100 | A1 | 1 | 100 | Tuesday, May 2, ... |
| 400033 | 300001 | อ.จันต๊ะ | 250 | 962 | 24 | A1 | 1 | 24 | Tuesday, May 2, ... |
| 400033 | 300004 | ข้าวสาร | 1000 | 990 | 72 | B2 | 1 | 72 | Tuesday, May 2, ... |
| 400033 | 300004 | ข้าวสาร | 1000 | 991 | 72 | B2 | 1 | 72 | Tuesday, May 2, ... |
| 400033 | 300006 | หมวกผ้า BREEZE เพลิดเพลิน... | 3000 | 979 | 135 | C2 | 1 | 135 | Tuesday, May 2, ... |

แนะนำเส้นทาง: A1> Pick A1 ||A2>B2> Pick B2 ||C2> Pick C2 ||OUT สรุป: A1>B2>C2

ภาพที่ ก.6 หน้าแนะนำเส้นทาง

ภาคผนวก ข

โปรแกรมการหาเส้นทางที่สั้นที่สุดของการหยิบสินค้าตามใบสั่งในคลังสินค้า
ด้วยข้อบังคับระยะทางและน้ำหนัก

```

1: Public Class Fnavigator

2: Dim graph2D(,) As Integer

3: Dim countArray As Integer = 0

4: Dim r As Integer = 0

5: Private Shared Function MinimumDistance(distance As Integer(), shortestPathTreeSet
    As Boolean(), verticesCount As Integer) As Integer

6: Dim min As Integer = Integer.MaxValue

7: Dim minIndex As Integer = 0

8: For v As Integer = 0 To verticesCount - 1

9: If shortestPathTreeSet(v) = False AndAlso distance(v) <= min Then

10: min = distance(v)

11: minIndex = v

12: End If

13: Next

14: Return minIndex

15: End Function

16: Private Shared Sub Print(distance As Integer(), verticesCount As Integer)

17: For i As Integer = 0 To verticesCount - 1

18: Console.WriteLine("{0}" & vbTab & " {1}", i, distance(i))

19: Next

20: End Sub

21: Public Shared Sub Dijkstra(graph As Integer(,), source As Integer, verticesCount As
    Integer)

22: Dim distance As Integer() = New Integer(verticesCount - 1) {}

```

```
23: Dim shortestPathTreeSet As Boolean() = New Boolean(verticesCount - 1) {}

24: For i As Integer = 0 To verticesCount - 1
25:     distance(i) = Integer.MaxValue
26:     shortestPathTreeSet(i) = False
27: Next

28: distance(source) = 0

29: For count As Integer = 0 To verticesCount - 2
30:     Dim u As Integer = MinimumDistance(distance, shortestPathTreeSet, verticesCount)
31:     shortestPathTreeSet(u) = True

32:     For v As Integer = 0 To verticesCount - 1
33:         If Not shortestPathTreeSet(v) AndAlso Convert.ToBoolean(graph(u, v)) AndAlso
           distance(u) <> Integer.MaxValue AndAlso distance(u) + graph(u, v) < distance(v) Then
34:             distance(v) = distance(u) + graph(u, v)
35:         End If
36:     Next
37: Next

38: Print(distance, verticesCount)
39: End Sub

40: Class Way
41:     Public Property way As String
42: End Class

43: Class Item
44:     Public Property itemName As String
```

```
45: Public Property ItemZone As String
46: Public Property ItemWeight As Double
47: Public Property ItemQuantity As Double
48: Public Property ItemDistance As Double
49: Public Property ItemWay As String
50: Public Property ValueOfSolution As Double

51: End Class

52: Private Sub testcal()
53: Dim total_distance As Double = 0
54: Dim total_weight As Double = 0
55: Dim total_energy As Double = 0
56: Dim total_recommend As String = " "
57: Dim total_zone As String = " "

58: Dim i As Integer = 0
59: Dim ItemList = New List(Of Item)

60: For i = 0 To DataGridView1.Rows.Count - 1
61: Dim Item = New Item()
62: Item.ItemName = DataGridView1.Rows(i).Cells(2).Value
63: Item.ItemZone = DataGridView1.Rows(i).Cells(6).Value
64: Item.ItemWeight = CDbI(DataGridView1.Rows(i).Cells(3).Value)
65: Item.ItemQuantity = CDbI(DataGridView1.Rows(i).Cells(7).Value)
66: ItemList.Add(Item)
67: Next
```

```

68: Dim conn As New
      OleDb.OleDbConnection("Provider=Microsoft.ACE.OLEDB.12.0;Data Source=" &
      Application.StartupPath & "\dbpill.accdb")
69: conn.Open()
70: If ItemList.Count - 1 > 0 Then

71: ItemList.RemoveAt(ItemList.Count - 1) 'Remove Emthy Array

72: Dim ResultOfItem = New List(Of Item)
73: Dim CurrentZone As String = "IN"

74: If ItemList.Count - 1 > 0 Then
75: While ItemList.Count >= 1

76: Dim BestItemWayInRound = New List(Of Item)
77: Dim IndexOfItem_OnSameZone = New List(Of Int32)
78: For i = 0 To ItemList.Count - 1
79: Dim CheckSameZone As Boolean = True
80: For q = 0 To IndexOfItem_OnSameZone.Count - 1
81: If IndexOfItem_OnSameZone(q) = i Then
82: CheckSameZone = False
83: End If
84: Next

85: If CheckSameZone = True Then
86: Dim Way = New List(Of Way) 'เส้นทางการเก็บตัวแรก
87: 'Dim strshow As String = "SELECT From T_ways"
88: Dim strshow As String = "SELECT way FROM T_ways WHERE CurrentZone = " &
      CurrentZone & " And AnotherZone= " & ItemList(i).ItemZone & ""
89: Using cmd = New OleDb.OleDbCommand(strshow, conn)
90: Using rdr = cmd.ExecuteReader()

```

```

91: While rdr.Read()
92: Dim user = New Way()
93: user.way = rdr.GetString(0)
94: Way.Add(user)
95: End While
96: End Using
97: End Using

98: Dim ValueOfItem = New List(Of Double) 'ระยางทางของเส้นทางที่ไปได้ในแต่ละไอเทม
99: Dim ValueOfSumSolution = New List(Of Double) 'เก็บค่าที่ใช้คำนวณจากสูตร
100: Dim ValueWeight = New List(Of Double)
101: For j = 0 To Way.Count - 1
102: Dim str As String
103: Dim strArr() As String
104: str = Way(j).way
105: strArr = str.Split(",")
106: Dim sumDistance As Double = 0
107: For count = 0 To strArr.Length - 1
108: If count + 1 < strArr.Length Then
109: Dim query As String = "SELECT Distance FROM T_distance WHERE startzone = '" &
    strArr(count) & "' And targetzone = '" & strArr(count + 1) & "'"
110: Using cmd = New OleDb.OleDbCommand(query, conn)
111: Using rdr = cmd.ExecuteReader()
112: While rdr.Read()
113: sumDistance += rdr.GetInt32(0)
114: End While
115: End Using
116: End Using
117: End If
118: Next

```

```
119:   Dim Sumsolution As Double = (sumDistance * 0.3) + (ItemList(i).ItemWeight *
      ItemList(i).ItemQuantity * 0.7)
120:   Dim Weight As Double = ItemList(i).ItemWeight
121:   For o = 0 To ItemList.Count - 1
122:     If o > i Then
123:       Dim CheckTrue = True
124:       For t = 0 To IndexOfItem_OnSameZone.Count - 1
125:         If ItemList(IndexOfItem_OnSameZone(t)).ItemZone = ItemList(i).ItemZone Then
126:           CheckTrue = False
127:         End If
128:       Next
129:       If CheckTrue = True Then
130:         If ItemList(o).ItemZone = ItemList(i).ItemZone Then
131:           Sumsolution += (ItemList(o).ItemWeight * ItemList(o).ItemQuantity * 0.7)
132:           Weight += ItemList(o).ItemWeight
133:           IndexOfItem_OnSameZone.Add(o)
134:           ItemList(i).ItemWeight = Weight
135:         End If
136:       End If
137:     End If
138:   Next
139:   ValueOfSumSolution.Add(Sumsolution)
140:   ValueOfItem.Add(sumDistance)
141:   ValueWeight.Add(Weight)
142: Next
```

```

143: Dim BestWayIndex As Int32 =
        ValueOfSumSolution.IndexOf(ValueOfSumSolution.Min()) 'หาค่าที่น้อยที่สุด
144: Dim Item = New Item()
145: Item.ItemName = ItemList(i).ItemName
146: Item.ItemQuantity = ItemList(i).ItemQuantity
147: Item.ItemWay = Way(BestWayIndex).way
148: Item.ItemWeight = ValueWeight(BestWayIndex)
149: Item.ItemDistance = ValueOfItem(BestWayIndex)
150: Item.ValueOfSolution = ValueOfSumSolution(BestWayIndex)
151: Item.ItemZone = ItemList(i).ItemZone
152: BestItemWayInRound.Add(Item)
153: End If

154: Next
155: Dim BestWayIndexonRound As Int32
156: For x = 0 To BestItemWayInRound.Count - 1
157: Dim DistanceofItem As Double
158: If x = 0 Then
159: If BestItemWayInRound(x).ValueOfSolution < 100000 Then
160: BestWayIndexonRound = x
161: End If
162: Else
163: DistanceofItem = BestItemWayInRound(BestWayIndexonRound).ValueOfSolution
164: If BestItemWayInRound(x).ValueOfSolution < DistanceofItem Then
165: BestWayIndexonRound = x
166: End If
167: End If
168: Next
169: Dim strBestWay() As String
170: strBestWay = BestItemWayInRound(BestWayIndexonRound).ItemWay.Split(",")

```

```

171: CurrentZone = strBestWay(strBestWay.Length - 1)

172: Dim ItemChoose = New Item()

173: ItemChoose.ItemName = BestItemWayInRound(BestWayIndexonRound).ItemName
174: ItemChoose.ItemWeight = BestItemWayInRound(BestWayIndexonRound).ItemWeight
175: ItemChoose.ItemQuantity =
    BestItemWayInRound(BestWayIndexonRound).ItemQuantity
176: ItemChoose.ItemWay = BestItemWayInRound(BestWayIndexonRound).ItemWay
177: ItemChoose.ItemDistance =
    BestItemWayInRound(BestWayIndexonRound).ItemDistance
178: ItemChoose.ValueOfSolution =
    BestItemWayInRound(BestWayIndexonRound).ValueOfSolution
179: ItemChoose.ItemZone = BestItemWayInRound(BestWayIndexonRound).ItemZone
180: ResultOfItem.Add(ItemChoose)

181: For e As Double = IndexOfItem_OnSameZone.Count - 1 To 0 Step -1
182: ItemList.RemoveAt(IndexOfItem_OnSameZone(e))
183: Next
184: ItemList.RemoveAt(BestWayIndexonRound) 'DeleteItemChoose

185: End While

186: Dim EndZone As String = "OUT"
187: Dim WayEndZone = New List(Of Way) 'เส้นทางกการเก็บตัวแรก
188: 'Dim strshow As String = "SELECT From T_ways"
189: Dim QueryFindEnZone As String = "SELECT way FROM T_ways WHERE
    CurrentZone = '" & CurrentZone & "' And AnotherZone= '" & EndZone & """"
190: Using cmd = New OleDb.OleDbCommand(QueryFindEnZone, conn)
191: Using rdr = cmd.ExecuteReader()

```

```

192: While rdr.Read()
193: Dim user = New Way()
194: user.way = rdr.GetString(0)
195: WayEndZone.Add(user)
196: End While
197: End Using
198: End Using

199: Dim ValueOfItemEndZone = New List(Of Double) 'ระยะทางช่องเส้นทางที่ไปได้ในแต่ละไอเทม
200: Dim ValueOfSumSolutionEndZone = New List(Of Double) 'เก็บค่าที่ใช้คำนวณจากสูตร
201: For j = 0 To WayEndZone.Count - 1
202: Dim str As String
203: Dim strArr() As String
204: str = WayEndZone(j).way
205: strArr = str.Split(",")
206: Dim sumDistance As Double = 0
207: For count = 0 To strArr.Length - 1
208: If count + 1 < strArr.Length Then
209: Dim query As String = "SELECT Distance FROM T_distance WHERE startzone = '" &
strArr(count) & "' And targetzone = '" & strArr(count + 1) & "'"
210: Using cmd = New OleDb.OleDbCommand(query, conn)
211: Using rdr = cmd.ExecuteReader()
212: While rdr.Read()
213: sumDistance += rdr.GetInt32(0)
214: End While
215: End Using
216: End Using
217: End If
218: Next

```

```

219: Dim Sumsolution As Double = (sumDistance * 0.3)

220: ValueOfSumSolutionEndZone.Add(Sumsolution)
221: ValueOfItemEndZone.Add(sumDistance)
222: Next

223: Dim BestWayIndexEndZone As Int32 =
    ValueOfSumSolutionEndZone.IndexOf(ValueOfSumSolutionEndZone.Min()) 'หาค่าที่
    น้อยที่สุด
224: Dim IT = New Item()
225: IT.ItemWay = WayEndZone(BestWayIndexEndZone).way
226: IT.ValueOfSolution = ValueOfSumSolutionEndZone(BestWayIndexEndZone)
227: IT.ItemDistance = ValueOfItemEndZone(BestWayIndexEndZone)
228: ResultOfItem.Add(IT)
229: End If
230: Dim mx = New Matrix()

231: For m = 0 To ResultOfItem.Count - 1
232: Console.Write(" ชื่อสินค้า " & ResultOfItem(m).ItemName)
233: Console.WriteLine()
234: Console.Write(" เส้นทาง " & ResultOfItem(m).ItemWay)
235: Console.WriteLine()
236: Console.Write(" ระยะทาง " & ResultOfItem(m).ItemDistance)
237: Console.WriteLine()
238: Console.Write(" น้ำหนัก " & ResultOfItem(m).ItemWeight)
239: Console.WriteLine()
240: Console.Write(" Utility " & ResultOfItem(m).ValueOfSolution)
241: Console.WriteLine()
242: Console.Write("=====")
243: Console.WriteLine()

```

```
244: Dim strSplitWay As String
245: Dim strSplitWayArr() As String
246: strSplitWay = ResultOfItem(m).ItemWay
247: strSplitWayArr = strSplitWay.Split(",")
248: For h = 1 To strSplitWayArr.Length - 1

249: total_recommend += strSplitWayArr(h) + ">"
250: Next
251: total_zone += ResultOfItem(m).ItemZone + ">"
252: total_recommend += " Pick " & ResultOfItem(m).ItemZone & " ||"
253: Dim sum_weight_quantity = ResultOfItem(m).ItemWeight *
    ResultOfItem(m).ItemQuantity
254: If (m <> 0) Then
255: total_energy += ResultOfItem(m).ItemDistance * total_weight * 9.81

256: End If

257: If ResultOfItem(m).ItemZone = "A1" Then
258: mx.A1 += ResultOfItem(m).ValueOfSolution
259: ElseIf ResultOfItem(m).ItemZone = "A2" Then
260: mx.A2 += ResultOfItem(m).ValueOfSolution
261: ElseIf ResultOfItem(m).ItemZone = "B1" Then
262: mx.B1 += ResultOfItem(m).ValueOfSolution
263: ElseIf ResultOfItem(m).ItemZone = "B2" Then
264: mx.B2 += ResultOfItem(m).ValueOfSolution
265: ElseIf ResultOfItem(m).ItemZone = "C1" Then
266: mx.C1 += ResultOfItem(m).ValueOfSolution
267: ElseIf ResultOfItem(m).ItemZone = "C2" Then
268: mx.C2 += ResultOfItem(m).ValueOfSolution
```

```
269: End If

270: total_weight += sum_weight_quantity
271: total_distance += ResultOfItem(m).ValueOfSolution
272: Next
273: Dim SplitOutArr As String
274: SplitOutArr = total_recommend.Substring(0, total_recommend.Length - 10)
275: txt_recommend.Text = SplitOutArr
276: txt_zone.Text = total_zone.Substring(0, total_zone.Length - 2)
277: txt_weight.Text = total_weight.ToString
278: txttotalway.Text = total_distance.ToString
279: txt_energy.Text = total_energy.ToString
280: Console.Write(mx.A1 & " " & mx.B1 & " " & mx.C1)
281: Console.WriteLine()
282: Console.Write(mx.A2 & " " & mx.B2 & " " & mx.C2)
283: Console.WriteLine()

284: End If

285: End Sub

286: Class Matrix
287: Public Property A1 As Double
288: Public Property B1 As Double
289: Public Property C1 As Double
290: Public Property A2 As Double
291: Public Property B2 As Double
292: Public Property C2 As Double
293: End Class
```