Pseudocode

# **1. Function To add Items to the cart**

FUNCTION addRecord()

WHILE True

OUTPUT "Enter Product code"

code <-- USER INPUT

IF isValidCode(code)

IF code is "END"

showRecords(cart)

BREAK

ELSE:

CONTINUE

ENDIF

OUTPUT "Enter Quantity"

quantity <-- USER INPUT

OUTPUT "Enter shipping method"

shipping <-- USER INPUT

IF isValidCode(code) and isValidQuantity(quantity) and isValidShipping(shipping)

item <-- [code ,productList[code] , priceList[code] , quantity , shipping]

cart <-- cart + item

ELSE

RAISE ERROR AS AN ALERT

ENDIF

REPEAT

END FUNCTION

# **2. Function To validate code input.**

FUNCTION isValidCode(code)

IF code =="END"

RETURN True

ELSE

TRY

code <-- int(code)

EXCEPT

RETURN False

ELSE

IF code>=0 and code <= 39

RETURN True

ELSE

False

ENDIF

ENDIF

END FUNCTION

# **3. Function to validate quantity value.**

FUNCTION isValidQuantity(quantity)

TRY

quantity <-- int(quantity)

EXCEPT

RETURN False

ELSE

IF quantity >0 AND quantity<=99

RETURN True

ELSE

False

ENDIF

END FUNCTION

# **4. Function to validated the shipping method.**

FUNCTION isValidShipping(shipping)

IF shipping !="Pick-up" and shipping != "Delivery":

RETURN False

ELSE:

RETURN True

ENDIF

END FUNCTION

# **5. Function to calculate total cost of items in the cart.**

FUNCTION totalCost(cart)

all\_cost <-- 0

FOR each\_item <-- IN cart

IF each\_item[4] == "Delivery"

sub\_cost <-- each\_item[2]\*each\_item[3] + 10/100(each\_item[2]\*each\_item[3])

ELSE

sub\_cost <-- each\_item[2]\*each\_item[3]

ENDIF

all\_cost <-- all\_cost + sub\_cost

ENDFOR

RETURN all\_cost

END FUNCTION

# **6. Function to search all items with a specific query.**

FUNCTION searchRecord(cart)

OUTPUT "Enter item information to search"

query <-- USER INPUT

found <-- []

FOR item <-- IN cart:

FOR val <-- IN item

IF query IN val

found <-- found + item

ENDIF

ENDFOR

ENDFOR

RETURN sortRecords(found)

END FUNCTION

# **7. Function to sort items in ascending order from cart.**

FUNCTION sortRecords(cart)

i <-- 0

FOR item <-- IN cart

if item[0][i] > item[0][i+1]

SWAP(item[:][i], item[:][i+1])

ENDIF

i <-- i +1

ENDFOR

RETURN cart

END FUNCTION

# **8. Function to display all formated items from the cart.**

FUNCTION showRecords(cart)

OUTPUT "Code Product Price $ Quantity Shopping Method"

OUTPUT "---- ------- ------- -------- ---------------"

FOR item <-- IN cart

OUTPUT item[0] item[1] item[2] item[3] item[4]

ENDFOR

OUTPUT "Total Cost is $ totalCost(cart)

END FUNCTION

# **9. Function to call all other functions.**

FUNCTION Controller()

OUTPUT "WELCOME TO Toowoomba Shopping System"

OUTPUT "Select an option to continue"

OUTPUT "1 . Add Items to Cart 2.ShowRecords 3. Sourt You Cart 4.Search for Items Else Exit"

choice <-- USER INPUT

IF choice == 1

addRecord()

ELIF choice == 2

showRecords(cart)

ELIF choice == 3

sortRecords(cart)

ELIF choice == 4

searchRecord(cart)

ELSE

EXIT

END FUNCTION

# **10. Function to read file and create product details.**

FUNCTION ReadFileAndCreateCart()

cart <-- []

details <--[]

file <-- OPEN(filepath)

FOR line <-- in file

details = details + line.split("=")[1]

ENDFOR

CLOSE file

price\_list <--details[1]

iten\_list <--details[0]

RETURN item\_list , price\_list , cart