**Admin Application:**

**main.c:**

function main():

initDisplay()

login()

displayTopSellers()

displayMainMenu()

function login():

print "Welcome to Phone-Dikhao Admin Application"

info(\_\_func\_\_, \_\_LINE\_\_, "Welcome to Phone-Dikhao Admin Application")

print "Enter Admin Username: "

read username

debug(\_\_func\_\_, \_\_LINE\_\_, "Entered username: " + username)

print "Enter Admin Password: "

read password

debug(\_\_func\_\_, \_\_LINE\_\_, "Entered password: " + password)

validateCredentials(username, password)

function validateCredentials(username, password):

if username is not "admin" or password is not "admin":

print "Invalid credentials. Please try again."

error(\_\_func\_\_, \_\_LINE\_\_, "Invalid credentials. Please try again.")

login()

else:

displayMainMenu()

function displayTopSellers():

info(\_\_func\_\_, \_\_LINE\_\_, "Start displaying top sellers")

file = openFile("mobileData.bin", "rb")

if file is NULL:

error(\_\_func\_\_, \_\_LINE\_\_, "Failed to open mobileData.bin file")

print "Error: Unable to open mobileData.bin file."

return

topSellersCount = 0

do:

mobile = readMobile(file)

if mobile != NULL and mobile.displayFlag == MostPurchased:

printMobileDetails(mobile)

topSellersCount = topSellersCount + 1

while mobile is not NULL and topSellersCount < 5

closeFile(file)

function displayMainMenu():

info(\_\_func\_\_, \_\_LINE\_\_, "Displaying Phone-Dikhao Admin Main Menu")

print "1. Add Mobile"

print "2. Delete Mobile"

print "3. Edit Mobile"

print "4. Search Mobile"

print "5. View Top Sellers"

print "6. Exit"

print "Enter Choice"

read choice

debug(\_\_func\_\_, \_\_LINE\_\_, "Entered choice: " + choice)

switch choice:

case 1:

addMobile()

case 2:

deleteMobile()

case 3:

editMobile()

case 4:

searchMobile()

case 5:

viewTopSellers()

case 6:

exitApplication()

default:

error(\_\_func\_\_, \_\_LINE\_\_, "Invalid choice. Please try again.")

print "Invalid choice. Please try again."

displayMainMenu()



**addMobile.c:**

function addMobile():

generateUniqueId()

askDetails()

if(confirm()):

if (saveToDatabase(mobile)):

displayMainMenu();

else:

displayMainMenu();

function generateUniqueId():

file = openFile("currentId.txt", "w+")

if file is NULL:

print "Error: Unable to open currentId.txt file."

return

fscanf(file, "%d", &id)

tempId = id + 1

fprintf(file, "%d", tempId)

closeFile(file)

function askDetails():

print "Enter Mobile Name: "

read name

if(checkIfMobileExists(name)):

print "Error: Mobile with this name already exists."

displayMainMenu()

print "Enter Brand: "

read brand

print "Enter Price: "

read price

print "Enter Discount: "

read discount

finalPrice = price - (price \* (discount / 100))

print "Enter RAM: "

read configurations.ram

print "Enter Storage: "

read configurations.storage

print "Enter Chipset: "

read configurations.chipset

print "Enter Camera: "

read configurations.camera

print "Enter Quantity: "

read quantity

displayFlagMenu()

function displayFlagMenu():

print "Select Display Flag:"

print "1. New"

print "2. Refurbished"

print "3. Outdated"

print "Enter Choice"

read choice

switch choice:

case 1:

displayFlag = New

break

case 2:

displayFlag = Refurbished

break

case 3:

displayFlag = Outdated

break

default:

print "Invalid choice. Please try again."

displayFlagMenu()

function checkIfMobileExists(name):

file = openFile("mobileData.bin", "rb")

if file is NULL:

print "Error: Unable to open mobileData.bin file."

return 0

while readMobile(file) is not NULL:

if mobile.name is equal to name:

closeFile(file)

return 1

closeFile(file)

return 0

function saveToDatabase(\*mobile):

file = openFile("mobileData.bin", "ab")

if file is NULL:

print "Error: Unable to open mobileData.bin file."

return 0

else:

fwrite(&mobile, sizeof(struct MobileData), 1, file);

print "Mobile added successfully!";

return 1

closeFile(file)



**deleteMobile.c:**

displayOutdatedMobiles():

outdatedCount = 0

do:

mobile = readMobile(file)

if mobile != NULL and mobile.displayFlag == Outdated:

printMobileDetails(mobile)

outdatedCount = outdatedCount + 1

while mobile is not NULL and outdatedCount < 5

deleteMobile():

function deleteMobile():

if isIdValid():

if (confirm()):

file = openFile("mobileData.bin", "rb")

if file is NULL:

print "Error: Unable to open mobileData.bin file."

return 0

tempFile = openFile("tempMobileData.bin", "wb")

if tempFile is NULL:

print "Error: Unable to open tempMobileData.bin file."

return

do:

mobile = readMobile(file)

if mobile != NULL and mobile.id != idToDelete:

fwrite(&mobile, sizeof(struct MobileData), 1, tempFile);

while mobile is not NULL

closeFile(file)

closeFile(tempFile)

remove("mobileData.bin")

rename("tempMobileData.bin", "mobileData.bin")

print "Mobile record with ID deleted successfully."



**searchMobile.c:**

function searchMobile():

displaySearchMenu()

function displaySearchMenu():

print "Search Options:"

print "1. Search by Mobile Name"

print "2. Search by Brand Name"

print "3. Return to Main Menu"

print "Enter your choice:"

read choice

switch choice:

case 1:

searchByMobileName()

case 2:

searchByBrandName()

case 3:

return

default:

print "Invalid choice. Please try again."

return

function searchByMobileName():

print "Enter Mobile Name to search:"

read mobileName

file = openFile("mobileData.bin", "rb")

if file is NULL:

print "Error: Unable to open mobileData.bin file."

return

foundMobiles = 0

do:

readMobile(file, &mobile)

if readMobile(file, &mobile) and strcasecmp (mobile.name, mobileName)==0:

printMobiledetails(mobile)

foundMobiles = foundMobiles + 1

while readMobile(file, &mobile)

closeFile(file)

mobileFound(foundMobiles)

function searchByBrandName():

print "Enter Brand Name to search:"

read brandName

file = openFile("mobileData.bin", "rb")

if file is NULL:

print "Error: Unable to open mobileData.bin file."

return

foundMobiles = 0

do:

readMobile(file, &mobile)

if readMobile(file, &mobile) and strcasecmp (mobile.brand, brandName)==0:

printMobiledetails(mobile)

foundMobiles = foundMobiles + 1

while readMobile(file, &mobile)

closeFile(file)

mobileFound(foundMobiles)

function mobileFound(foundMobiles):

if foundMobiles == 0:

print "No matching mobiles found."

return

else:

print "-- Mobiles Found."

escape()

function escape():

print "Enter 'x' to exit"

read escapeInput

if escapeInput is 'x' or 'X':

return

else:

print "Enter valid input."

escape()



**editMobile.c:**

function editMobile:

displaySearchMenu()

print "Enter the ID of the mobile to edit"

read idToEdit

file = openFile("mobileData.bin", "rb+")

if file is NULL:

print "Error: Unable to open mobileData.bin file."

return

idFound = 0

do:

mobile = readMobile(file)

if mobile != NULL and mobile.id == idToEdit:

idFound = 1

break

while mobile is not NULL

if !idFound:

print “Invalid ID.”

displaymainMenu()

displayEditMenu(mobile)

fseek(file, -sizeof(struct MobileData), SEEK\_CUR)

writeMobile(file, mobile)

print "Mobile record updated successfully."

closeFile(file)

function displayEditmenu(mobile):

printMobiledetails(mobile)

print "Edit Menu:"

print "1. Edit Price"

print "2. Edit Discount"

print "3. Edit Display Flag"

print "4. Edit Quantity"

print "Enter your choice"

read choice

switch choice:

case 1:

print "Enter new Price: "

read mobile.price

break

case 2:

print "Enter new Discount: "

read mobile.discount

break

case 3:

print "Enter new Display Flag: "

read mobile.displayFlag

break

case 4:

print "Enter new Quantity: "

read mobile.quantity

break

default:

defaultCount=1

while defaultCount < 3:

print "Invalid choice."

defaultCount++

displayEditMenu()

displayMainMenu()



**User Application:**

**searchMobile.c:**

function searchMobile:

filterByPrice()

function filterByPrice():

file = openFile("mobileData.bin", "rb")

if file is NULL:

print "Error: Unable to open mobileData.bin file."

return

tempFile = openFile("tempMobileData.bin", "wb")

if tempFile is NULL:

print "Error: Unable to open tempMobileData.bin file."

closeFile(file)

return

foundMobiles = 0

do:

mobile = readMobile(file)

if mobile != NULL and mobile.finalPrice >= minPrice and mobile.finalPrice <= maxPrice and mobile.display\_flag != Outofstock:

printMobileDetails(mobile)

fwrite(&mobile, sizeof(struct MobileData), 1, tempFile)

foundMobiles = foundMobiles + 1

while mobile is not NULL

closeFile(file)

closeFile(tempFile)

mobileFound(foundMobiles)

buyMobile()

function mobileFound(foundMobiles):

if foundMobiles == 0:

print "No matching mobiles found."

displayMainMenu()

else:

print "-- Mobiles Found."

function buyMobile():

print "Want to buy? (y/n): "

read buyChoice

if (buyChoice is 'n' or 'N'):

displayMainMenu()

else if (buyChoice is 'y' or 'Y'):

print "Enter mobile name: "

read mobName

tempFile = openFile("tempMobileData.bin", "rb")

if tempFile is NULL:

print "Error: Unable to open tempMobileData.bin file."

return

foundMobiles = 0

do:

mobile = readMobile(tempFile)

if mobile != NULL and strcasecmp(mobile.name, mobName) == 0:

foundMobiles = 1

print "Enter your name: "

read buyerName

print "Enter Quantity to buy: "

read buyQuantity

if buyQuantity <= mobile.quantity:

mobile.count = mobile.count + buyQuantity

mobile.quantity = mobile.quantity - buyQuantity

if mobile.quantity == 0:

mobile.display\_flag = Outofstock

buyFile = openFile("buyFile.txt", "a")

fprintf(buyFile, "%s,%s,%d\n", mobile.name, buyerName, buyQuantity)

fseek(tempFile, -sizeof(struct MobileData), SEEK\_CUR)

writeMobile(tempFile, mobile)

closeFile(tempFile)

closeFile(buyFile)

else:

print "Not Enough Quantity"

print "Operation Cancelled"

print "Max quantity mobile.quantity "

break

while mobile is not NULL

if !foundMobiles:

print "No matching mobiles found."

displayMainMenu()

