

PSUEDOCODES

FUNCTION loadContacts():

TRY:

OPEN file "contacts.txt" FOR READING AS scanner

CLEAR contactListModel

WHILE (scanner HAS NEXT LINE) DO:

line = scanner.NEXT

parts = SPLIT line BY ","

IF (parts.length == 2)THEN:

contact = NEW Contact(parts[0], parts[1]) // Create a new Contact

contactListModel = contact.add(NEW Contact) // Add contact to the list

DISPLAY ("Contacts loaded successfully.") // Notify success

CATCH FileNotFoundException AS e: +

END FUNCTION

FUNCTION saveContacts ():

TRY:

OPEN file "contacts.txt" FOR WRITING AS writer

FOR (i == 0; i = contactListModel - 1;i++)DO:

contact = contactListModel.get(i)

line = contact.getName () + ";" + contact.getPhone()

writer.WRITE_LINE (line)

SHOW MESSAGE "Contacts saved successfully “

CATCH File Not Found Exception AS e:

DISPLAY "Error", "Unable to save contacts."

END FUNCTION

FUNCTION addContact():

Prompt for contact name

Get name

IF (name != null) AND (name != isEmpty) THEN

Prompt for contact phone

Get phone

IF phone IS NOT NULL AND phone IS NOT EMPTY THEN:

isDuplicate = FALSE

endif

endif

// Linear search for duplicates

FOR (i=0; i=contactListModel - 1)DO:

IF contactListModel.getName (i) = ignoreCase(name) THEN

isDuplicate = TRUE

endif

BREAK

```

    IF NOT isDuplicate THEN
        contactListModel.ADD NEW Contact(name, phone)
    ELSE
        DISPLAY "Duplicate Contact", "A contact with this name already exists."
    ELSE:
        DISPLAY "Invalid Input"; "Phone cannot be empty."
    ELSE:
        DISPLAY "Invalid Input" ; "Name cannot be empty."
endif
END FUNCTION

FUNCTION deletContact(){
    DECLARE confirmation
    Contact selectedContact= contactJList.getSelectedValue()
    IF (selectedContact != null)
        Prompt user if they are sure they want to delete this contact
        Get confirmation
    Endif
    IF (confirmation="yes") THEN
        contactListModel.removeElement(selectedContact)
    endif
END FUNCTION
}

```

```
FUNCTION updateContact(){  
    Contact selectedContact = contactJlist.getSelectedValue()  
    IF(selectedContact != null)  
        Prompt for new name  
        Get newName  
        Prompt for new phone  
        Get newphone  
    Endif  
    IF(newName != null) AND (newName != newName.isEmpty ) AND (newPhone != null)  
    AND (newPhone != newPhone.IsEmpty) THEN  
        selectedContact.Name = newName  
        selectedContact.setPhone = newPhone  
        REPAINT contactJList  
    ELSE  
        Display ("Invalid Input" , "Name and Phone cannot be empty")  
    Endif  
END FUNCTION}
```

```
FUNCTION searchContacts
```

```
    SET searchTerm = LOWERCASE(get text.searchField)
```

```
    CREATE filteredModel as new empty list of Contacts
```

```
    // Linear search for matching contacts
```

```
    FOR (i =0; i= contactListModel.length – 1) DO
```

```
        SET contact = get.contactListModel( i)
```

```
        IF contact.getName CONTAINS(LOWERCASE(searchTerm) OR  contact.getPhone
```

```
CONTAINS (searchTerm) THEN
```

```
            filteredModel.addElement = contact
```

```
        END IF
```

```
    END FOR
```

```
END FUNCTION}
```

```

FUNCTION bubbleSort(model, criteria)

    n = model.size()

    swapped = false

    FOR i=0;i= n - 2;i++) DO

        swapped = false

        FOR j = 0; j=n - 2 - i;i++) DO

            c1 = model.get(j)

            c2 = model.get(j + 1)

            IF (criteria = "name") THEN

                condition = c1.name == (IgnoreCase(c2.name) > 0 )

            ELSE

                condition = c1.phone == (ToIgnoreCase(c2.phone)) > 0

            ENDIF

            IF (condition) THEN

                model.set(j, c2)

                model.set(j + 1, c1)

                swapped = true

            ENDIF

        ENDFOR

    ENDIF

    IF (!swapped) THEN

        BREAK

    ENDIF

ENDFOR

```

END FUNCTION