

DELIVERABLE 4

Group N

PROJECT TOPIC: HEALTHCARE MANAGEMENT SYSTEMS

Group members	CWID	Contribution
Dhanvanth Voona	A20543395	20%
Neeraj Vardhan Buneeti	A20545853	20%
Priyadarshani LNU	A20561249	20%
Sai Charan Gangili	A20543155	20%
Sai Pranay Yada	A20553636	20%

1. Programming Language and Application Type

To demonstrate our Team project on “Healthcare Management System”, We have used **PYTHON** programming language to implement basic Create, Read, Update, and Delete (CRUD) operations, which encompass adding, reading, modifying, and deleting database records related to our projects. At First, we have written program (**database.py**) to create Database “Healthcare.db”, in which we create all the tables (Patients, Hospital, MedicalStaff, Appointment, MedicalRecords, Billing, Medications, Hcare_provides, Hcare_plan and LabTests), then insert the data into all the tables for Healthcare Management System.

2. CRUD Implementation

We have successfully created a program, known as "del_four.py," which serves as a command line tool. This program is designed to seamlessly execute the fundamental Create, Read, Update, and Delete (CRUD) operations on our "Healthcare" database. It is meticulously structured to ensure code readability and maintainability, with explanatory comments strategically placed throughout to facilitate a better understanding of its functionality and implementation.

- **Program Menu**

- a) **Table Menu**

It lets the user select the table to perform CRUD operation. The menu table also consists quit option to let the user exit the program

```
Select a table to perform Create, Read, Update, Delete (CRUD) operations:
1. Patient
2. Healthcareprovider
3. Healthcareplan
4. Hospital
5. Medicalstaff
6. Appointment
7. MedicalRecord
8. Billing
9. Medication
10. LabTest
11. Quit
Enter the number of the table: 6
Selected table: Appointment
```

b) Operations Menu

It lets user select from the the Create, Read, Update, Delete (CRUD)

```
Selected table: Appointment

Options:
1. Create a record
2. Read a record
3. Update a record
4. Delete a record
Enter your choice: 2
```

● Test Case 1: Create Operation

```
# CREATE operation
def create_record(table_name, data):
    query = f'INSERT INTO {table_name} VALUES ({",".join(["?" * len(data)]})'
    cursor.execute(query, data)
    print(f'Record created successfully in the {table_name} table.')
```

While executing this Testcase:

```
Enter the number of the table: 1
Selected table: Patient

Options:
1. Create a record
2. Read a record
3. Update a record
4. Delete a record
Enter your choice: 1
Enter data in this order:
PatientID, Patient_firstname, Patient_lastname, Patient_DateofBirth, Patient_Gender, Patient_address, Patient_phonenumber, Patient_email
Enter values (comma-separated):
16, testname, testlastname, 2000-01-01, M, 3001 Shore drive, 8888888888, test@gmail.com
Record created successfully in the Patient table.
```

Verify result in table:

```
Records in the Patient table:
(1, 'Maxwell', 'Longstreeth', '2008-11-17', 'M', '2851 Talisman Drive', '845 866 3572', 'mlongstreeth@live.com')
(2, 'Johnathan', 'Hazlewood', '1992-05-07', 'M', '26 La Follette Point', '143 764 2212', 'jhazlewood1@skyrock.com')
(3, 'Natala', 'Shackell', '2009-09-23', 'F', '4 Duke Point', '379 692 6948', 'nshackell2@tuttocitta.it')
(4, 'Kareem', 'Gathercole', '2000-04-09', 'M', '0 Merchant Junction', '102 769 4217', 'kgathercole3@phpbb.com')
(5, 'Quillan', 'Pierrepont', '2001-06-19', 'M', '28 Heath Drive', '520 326 6798', 'qpierrepont4@phpbb.com')
(6, 'Shantee', 'Mainston', '1997-01-09', 'F', '4 Cascade Junction', '146 639 7710', 'smainston5@skype.com')
(7, 'Ronda', 'Easterfield', '1997-04-25', 'F', '1 Jenna Park', '194 204 4719', 'reasterfield6@tiny.cc')
(8, 'Mora', 'Harpham', '2010-06-19', 'F', '90 Kedzie Avenue', '394 921 9045', 'mharpham7@loc.gov')
(9, 'Mord', 'Batt', '2018-04-26', 'M', '449 Scofield Trail', '146 460 4194', 'mbatt8@timesonline.co.uk')
(10, 'Wilbur', 'Kordt', '1982-05-11', 'M', '51 Fulton Avenue', '621 892 3239', 'wkordt9@paypal.com')
(11, 'Ab', 'Macaulay', '1986-05-27', 'M', '54 Toban Lane', '260 921 9022', 'amacaulaya@nature.com')
(12, 'Rafaelita', 'Minger', '2012-09-11', 'F', '696 Stuart Court', '855 577 2785', 'rmingerb@mac.com')
(13, 'Daron', 'Gabitis', '2015-04-04', 'M', '6159 Novick Junction', '378 431 2001', 'dgabitisc@reddit.com')
(14, 'Ysabel', 'Purselow', '2005-05-28', 'F', '3865 Kropf Circle', '881 635 8524', 'ypurselowd@zdnnet.com')
(15, 'Tove', 'Grabam', '2004-09-03', 'F', '4 Green Alley', '766 609 3291', 'tgrabame@friendfeed.com')
(16, 'testname', 'testlastname', '2000-01-01', 'M', '3001 Shore drive', '8888888888', 'test@gmail.com')
```

- Test Case 2: Read Operation

```
# READ operation
def read_records(table_name):
    cursor.execute(f'SELECT * FROM {table_name}')
    records = cursor.fetchall()
    if records:
        print(f"\nRecords in the {table_name} table:")
        for record in records:
            print(record)
    else:
        print(f"No records found in the {table_name} table.")
```

While executing this Testcase:

```
Select a table to perform Create, Read, Update, Delete (CRUD) operations:
1. Patient
2. Healthcareprovider
3. Healthcareplan
4. Hospital
5. Medicalstaff
6. Appointment
7. MedicalRecord
8. Billing
9. Medication
10. LabTest
11. Quit
Enter the number of the table: 1
Selected table: Patient

Options:
1. Create a record
2. Read a record
3. Update a record
4. Delete a record
Enter your choice: 2
```

```
Records in the Patient table:
(1, 'Maxwell', 'Longstreeth', '2008-11-17', 'M', '2851 Talisman Drive', '845 866 3572', 'mlongstreeth0@live.com')
(2, 'Johnathan', 'Hazlewood', '1992-05-07', 'M', '26 La Follette Point', '143 764 2212', 'jhazlewood1@skyrock.com')
(3, 'Natala', 'Shackell', '2009-09-23', 'F', '4 Duke Point', '379 692 6948', 'nshackell2@tuttocitta.it')
(4, 'Kareem', 'Gathercole', '2000-04-09', 'M', '0 Merchant Junction', '102 769 4217', 'kgathercole3@phpbb.com')
(5, 'Quillan', 'Pierrepont', '2001-06-19', 'M', '28 Heath Drive', '520 326 6798', 'qpierrepont4@phpbb.com')
(6, 'Shantee', 'Mainston', '1997-01-09', 'F', '4 Cascade Junction', '146 639 7710', 'smainston5@skype.com')
(7, 'Ronda', 'Easterfield', '1997-04-25', 'F', '1 Jenna Park', '194 204 4719', 'reasterfield6@tiny.cc')
(8, 'Mora', 'Harpham', '2010-06-19', 'F', '90 Kedzie Avenue', '394 921 9045', 'mharpham7@loc.gov')
(9, 'Mord', 'Batt', '2018-04-26', 'M', '449 Scofield Trail', '146 460 4194', 'mbatt8@timesonline.co.uk')
(10, 'Wilbur', 'Kordt', '1982-05-11', 'M', '51 Fulton Avenue', '621 892 3239', 'wkordt9@paypal.com')
(11, 'Ab', 'Macaulay', '1986-05-27', 'M', '54 Toban Lane', '260 921 9022', 'amacaulaya@nature.com')
(12, 'Rafaelita', 'Minger', '2012-09-11', 'F', '696 Stuart Court', '855 577 2785', 'rmingerb@mac.com')
(13, 'Daron', 'Gabitis', '2015-04-04', 'M', '6159 Novick Junction', '378 431 2001', 'dgabitisc@reddit.com')
(14, 'Ysabel', 'Purselow', '2005-05-28', 'F', '3865 Kropf Circle', '881 635 8524', 'ypurselowd@zdnnet.com')
(15, 'Tove', 'Grabam', '2004-09-03', 'F', '4 Green Alley', '766 609 3291', 'tgrabame@friendfeed.com')
(16, 'testname', 'testlastname', '2000-01-01', 'M', '3001 Shore drive', '8888888888', 'test@gmail.com')
```

- Test Case 3: Update Operation

```
# UPDATE operation
def update_record(table_name, record_id, update_data):
    placeholders = ", ".join([f"{column} = ?" for column in update_data.keys()])
    query = f'UPDATE {table_name} SET {placeholders} WHERE {table_name}ID = ?'
    cursor.execute(query, list(update_data.values()) + [record_id])
    if cursor.rowcount > 0:
        print(f'Record with ID {record_id} updated in the {table_name} table.')
    else:
        print(f'Record with ID {record_id} not found in the {table_name} table.')
```

While executing this Testcase:

```
Select a table to perform Create, Read, Update, Delete (CRUD) operations:
1. Patient
2. Healthcareprovider
3. Healthcareplan
4. Hospital
5. Medicalstaff
6. Appointment
7. MedicalRecord
8. Billing
9. Medication
10. LabTest
11. Quit
Enter the number of the table: 1
Selected table: Patient

Options:
1. Create a record
2. Read a record
3. Update a record
4. Delete a record
Enter your choice: 3
Enter the ID of the record you want to update in Patient: 16
Enter the name of the column you want to update: Patient_email
Enter the new value for the Patient_email column: updatedmail@gmail.com
Record with ID 16 updated in the Patient table.
```

Verify update in Table:

```
Records in the Patient table:
(1, 'Maxwell', 'Longstreeth', '2008-11-17', 'M', '2851 Talisman Drive', '845 866 3572', 'mlongstreeth0@live.com')
(2, 'Johnathan', 'Hazlewood', '1992-05-07', 'M', '26 La Follette Point', '143 764 2212', 'jhazlewood1@skyyrock.com')
(3, 'Natala', 'Shackell', '2009-09-23', 'F', '4 Duke Point', '379 692 6948', 'nshackell2@tuttocitta.it')
(4, 'Kareem', 'Gathercole', '2000-04-09', 'M', '0 Merchant Junction', '102 769 4217', 'kgathercole3@phpbb.com')
(5, 'Quillan', 'Pierrepont', '2001-06-19', 'M', '28 Heath Drive', '520 326 6798', 'qpierrepont4@phpbb.com')
(6, 'Shantee', 'Mainston', '1997-01-09', 'F', '4 Cascade Junction', '146 639 7710', 'smainston5@skype.com')
(7, 'Ronda', 'Easterfield', '1997-04-25', 'F', '1 Jenna Park', '194 204 4719', 'reasterfield6@tiny.cc')
(8, 'Mora', 'Harpham', '2010-06-19', 'F', '90 Kedzie Avenue', '394 921 9045', 'mharpham7@loc.gov')
(9, 'Mord', 'Batt', '2018-04-26', 'M', '449 Scofield Trail', '146 460 4194', 'mbatt8@timesonline.co.uk')
(10, 'Wilbur', 'Kordt', '1982-05-11', 'M', '51 Fulton Avenue', '621 892 3239', 'wkordt9@paypal.com')
(11, 'Ab', 'Macaulay', '1986-05-27', 'M', '54 Toban Lane', '260 921 9022', 'amacauleya@nature.com')
(12, 'Rafaelita', 'Minger', '2012-09-11', 'F', '696 Stuart Court', '855 577 2785', 'rmingerb@mac.com')
(13, 'Daron', 'Gabbitis', '2015-04-04', 'M', '6159 Novick Junction', '378 431 2001', 'dgabbitisc@reddit.com')
(14, 'Ysabel', 'Purselow', '2005-05-28', 'F', '3865 Kropf Circle', '881 635 8524', 'ypurselowd@zdnnet.com')
(15, 'Tove', 'Grabbam', '2004-09-03', 'F', '4 Green Alley', '766 609 3291', 'tgrabbame@friendfeed.com')
(16, 'testname', 'testlastname', '2000-01-01', 'M', '3001 Shore drive', '8888888888', 'updatedmail@gmail.com')
```

- Test Case 4: Delete Operation

```
# DELETE operation
def delete_record(table_name, record_id):
    cursor.execute(f'DELETE FROM {table_name} WHERE {table_name}ID = ?', (record_id,))
    if cursor.rowcount > 0:
        print(f'Record with ID {record_id} deleted from the {table_name} table.')
    else:
        print(f'Record with ID {record_id} not found in the {table_name} table.')
```

While executing this Testcase:

```
Select a table to perform Create, Read, Update, Delete (CRUD) operations:
1. Patient
2. Healthcareprovider
3. Healthcareplan
4. Hospital
5. Medicalstaff
6. Appointment
7. MedicalRecord
8. Billing
9. Medication
10. LabTest
11. Quit
Enter the number of the table: 1
Selected table: Patient

Options:
1. Create a record
2. Read a record
3. Update a record
4. Delete a record
Enter your choice: 4
Enter the ID of the record you want to delete from Patient: 16
Record with ID 16 deleted from the Patient table.
```

Verify deleted row in table:

```
Records in the Patient table:
(1, 'Maxwell', 'Longstreeth', '2008-11-17', 'M', '2851 Talisman Drive', '845 866 3572', 'mlongstreeth0@live.com')
(2, 'Johnathan', 'Hazlewood', '1992-05-07', 'M', '26 La Follette Point', '143 764 2212', 'jhazlewood1@skyrock.com')
(3, 'Natala', 'Shackell', '2009-09-23', 'F', '4 Duke Point', '379 692 6948', 'nshackell2@tuttocitta.it')
(4, 'Kareem', 'Gathercole', '2000-04-09', 'M', '0 Merchant Junction', '102 769 4217', 'kgathercole3@phpbb.com')
(5, 'Quillan', 'Pierrepoint', '2001-06-19', 'M', '28 Heath Drive', '520 326 6798', 'qpierrepoint4@phpbb.com')
(6, 'Shantee', 'Mainston', '1997-01-09', 'F', '4 Cascade Junction', '146 639 7710', 'smainston5@skype.com')
(7, 'Ronda', 'Easterfield', '1997-04-25', 'F', '1 Jenna Park', '194 204 4719', 'reasterfield6@tiny.cc')
(8, 'Mora', 'Harpham', '2010-06-19', 'F', '90 Kedzie Avenue', '394 921 9045', 'mharpham7@loc.gov')
(9, 'Mord', 'Batt', '2018-04-26', 'M', '449 Scofield Trail', '146 460 4194', 'mbatt8@timesonline.co.uk')
(10, 'Wilbur', 'Kordt', '1982-05-11', 'M', '51 Fulton Avenue', '621 892 3239', 'wkordt9@paypal.com')
(11, 'Ab', 'Macauley', '1986-05-27', 'M', '54 Toban Lane', '260 921 9022', 'amacauleya@nature.com')
(12, 'Rafaelita', 'Minger', '2012-09-11', 'F', '696 Stuart Court', '855 577 2785', 'rmingerb@mac.com')
(13, 'Daron', 'Gabitis', '2015-04-04', 'M', '6159 Novick Junction', '378 431 2001', 'dgabitisc@reddit.com')
(14, 'Ysabel', 'Purselow', '2005-05-28', 'F', '3865 Kropf Circle', '881 635 8524', 'ypurselowd@zdnnet.com')
(15, 'Tove', 'Grabbam', '2004-09-03', 'F', '4 Green Alley', '766 609 3291', 'tgrabbame@friendfeed.com')
```

- Test Case 5: Quit Option

```
Select a table to perform Create, Read, Update, Delete (CRUD) operations:
1. Patient
2. Healthcareprovider
3. Healthcareplan
4. Hospital
5. Medicalstaff
6. Appointment
7. MedicalRecord
8. Billing
9. Medication
10. LabTest
11. Quit
Enter the number of the table: 11
Group N Program Exited, Goodbye!
```

3. Error Handling and User Experience

We have written code for error handling and to show clear error messages for Test cases while executing the program.

Error Handling Test cases:

Here we show the different edge cases / error handling test cases and how our program lets the user to handle edge cases with clear instructions

1. Invalid Table number (Integer and alphabet combination),
If a user enters an invalid table number a proper error message is displayed and gives the instruction to resolve the issue.

```
Select a table to perform Create, Read, Update, Delete (CRUD) operations:
1. Patient
2. Healthcareprovider
3. Healthcareplan
4. Hospital
5. Medicalstaff
6. Appointment
7. MedicalRecord
8. Billing
9. Medication
10. LabTest
11. Quit
Enter the number of the table: 4D5
Invalid table choice
Enter Again from 1 to 11

Select a table to perform Create, Read, Update, Delete (CRUD) operations:
1. Patient
2. Healthcareprovider
3. Healthcareplan
4. Hospital
5. Medicalstaff
6. Appointment
7. MedicalRecord
8. Billing
9. Medication
10. LabTest
11. Quit
Enter the number of the table: 4
Selected table: Hospital
```


2. While Selecting Invalid Operation Number,
The program clearly instructs the user for any edge case in invalid operation number.

```
Options:
1. Create a record
2. Read a record
3. Update a record
4. Delete a record
Enter your choice: 8
Invalid Operation Choice.
Please enter choice from 1 to 4
```

```
Options:
1. Create a record
2. Read a record
3. Update a record
4. Delete a record
Enter your choice: 2
```

3. Error in Create Operation, giving the incorrect data,
Example: While Creating a record in the Hospital Table, if the user enters partial data / incorrect data the program identifies and asks to enter data again.

```
Enter the number of the table: 4
Selected table: Hospital

Options:
1. Create a record
2. Read a record
3. Update a record
4. Delete a record
Enter your choice: 1
Enter data in this order:
HospitalID, Hospital_name, Hospital_address, HealthcareplanID
Enter values (comma-separated):
89, shine hospital
Invalid Entry!!!
Number of values entered does not match the number of columns
Enter Again
Enter data in this order:
HospitalID, Hospital_name, Hospital_address, HealthcareplanID
Enter values (comma-separated):
399, shine hospitals, 2013 Lake Shore Drive, 203
Record created successfully in the Hospital table.
```

4. Error in Update Operation, giving invalid ID,
Example: While updating the Patients Table if the user enters invalid ID it is shown as an error message and asks to enter again.

```
Enter the number of the table: 1
Selected table: Patient

Options:
1. Create a record
2. Read a record
3. Update a record
4. Delete a record
Enter your choice: 3
Enter the ID of the record you want to update in Patient: 7895
Record with ID 7895 does not exist in the Patient table. Please enter a valid ID.
```

5. Error in Update Operation, giving incorrect column name
Example: While updating the LabTest Table if the user enters an incorrect column name it is shown as an error message and asks to enter the valid column name by displaying the column names in the table.

```
Enter the number of the table: 10
Selected table: LabTest

Options:
1. Create a record
2. Read a record
3. Update a record
4. Delete a record
Enter your choice: 3
Enter the ID of the record you want to update in LabTest: 999
Enter the name of the column you want to update: hospital
Column 'hospital' does not exist in the LabTest table
Please enter a valid column name from the following

['LabtestID', 'PatientID', 'MedicalstaffID', 'Labtest_date', 'Labtest_name', 'Labtest_results', 'Labtest_refe
rnecerange']
Enter the name of the column you want to update: |
```

6. Error in Delete Operation, giving invalid ID
Example: While trying to delete a record in the Hospital table and put a random invalid ID we got a proper error message that the record is not found in the Hospital table.

```
Selected table: Hospital

Options:
1. Create a record
2. Read a record
3. Update a record
4. Delete a record
Enter your choice: 4
Enter the ID of the record you want to delete from Hospital: 4444444
Record with ID 4444444 not found in the Hospital table.
```

4. Code Quality and Documentation

We have tried to maintain well-documented code with clear comments and explanations including proper naming conventions and code organization. We are providing a documentation file that guides users in setting up and using the application.

5. Functionality Demonstration (Loom Video)

We are providing high-quality Loom video recording that clearly demonstrates the Healthcare Management System functionality which explains the CRUD operations, highlighting key features and functionality. The video is well-organized, engaging, and effectively conveys the application's usage through testing and showcases different scenarios.

Thank you!