UTM
UNIVERSITI TEKNOLOGI MALAYSIA

UNIVERSITI TEKNOLOGI MALAYSIA

FACULTY OF COMPUTING, UTM

SEMESTER I, SESION 2023/2024

PROJECT: PHASE 3

SECD2523 : DATABASE

SECTION 08

NAME :

1. ADAM IMRAN BIN ALWI B23CS0017
2. CHONG WEI YANG A23CS5027
3. ISKANDAR HAKIMI BIN ZULKIPPLI B23CS0040
4. MUHAMMAD IZMEER BIN ZULKIFLEE B23CS0058
5. TAHNUSSRI A/P MORTHY B23CS0071

COURSE : BACHELOR OF COMPUTER SCIENCE

SUBMISSION DATE : 14 - JANUARY - 2024

LECTURER'S NAME : DR NOOR HIDAYAH BINTI ZAKARIA

TABLE OF CONTENTS

1.0 Introduction

In the ever-evolving healthcare environment, offering high-quality patient care requires efficient clinic and appointment management. Appointment scheduling, patient interactions, and healthcare service organization are the cornerstones of a well-functioning medical facility. It is common knowledge that advancements in technology have enhanced medicine in many ways, such as how doctors recognise and treat patients, how accurate and efficient medical procedures are, how simple it is to obtain information, and how easy it is for patients to receive treatment.

Our project's goal is to develop an appointment scheduling system for the Johor Bahru dialysis center Pusat Dialisis FN. With a commitment to enhancing patient experiences, improving operational efficiency, and ensuring the highest standards of care, our system is poised to revolutionize the way dialysis centers operate.Making medical bookings and arranging appointments has traditionally taken a lot of effort due to errors and inefficiencies. Prior to the widespread use of technology, a few challenging and traditional ways were manually arranging and canceling appointments, keeping records on paper, and making appointment calls. Pusat Dialisis FN is still using almost all of these issues.

The DialysisXCare system that we created can help the patients to easily schedule appointments through our user-friendly interface that is available 24/7 during clinic operating hours. The system provides real-time availability information, allowing patients to choose suitable time slots with confidence. We understand that life is unpredictable. Patients can modify or cancel appointments within a reasonable timeframe before the scheduled date, ensuring flexibility in managing their healthcare commitments. The system ensures a seamless experience by sending confirmation emails for successfully scheduled, modified, or canceled appointments. This feature enhances communication, keeping patients informed and engaged in their healthcare journey.Patients can securely view their upcoming appointments, access medical records, and review shared feedback. This transparency empowers individuals to actively participate in their healthcare decisions. Besides,there are so many benefits that can get from the system that Pusat Dialisis can use in their management. In conclusion, at the intersection of cutting-edge technology and compassionate healthcare, our system is designed to enhance operational efficiency, streamline patient care, and ensure the highest standards of safety and convenience.

2.0 Overview of Project

The DialysisXCare system represents a revolutionary advancement in the management and delivery of renal healthcare services. This comprehensive report aims to provide an insightful overview of the database conceptual design which includes business rules, the conceptual ERD is presented, offering a visual representation of the database's entities and relationships in their preliminary state and Enhanced ERD that provides a more detailed and accurate portrayal of the database structure. Next, we can see the database logical design,which is Logical ERD, illustrating the refined relationships and attributes based on logical considerations, data dictionary that contain entity, relationship and the attributes of the system we designed and also normalization that includes 1NF, 2NF, 3NF and BCNF (Boyce Codd's normal form). In this report also we have included relational database schemas for DialysisXCare and SQL statements that are divided into two which is Data Definition Language (DDL) and Data Manipulation Language (DML). The system represents a transformative step forward in the delivery of renal healthcare. This report concludes by summarizing the key benefits and outcomes associated with the implementation of this system, highlighting its potential to redefine the dialysis experience for both healthcare providers and patients alike.

3.0 Database Conceptual Design

3.1 Updated Business Rule

1. Patients should have the ability to schedule appointments through the system during clinic operating hours.
2. The system should provide real-time availability information for doctors and appointment slots.
3. Patients should be allowed to modify or cancel appointments within a reasonable timeframe before the scheduled appointment.
4. The system should send confirmation emails for successfully scheduled, modified, or canceled appointments.
5. Implement robust security measures to ensure the confidentiality and integrity of patient records.
6. The patients can securely view their upcoming appointments, medical records, and any shared feedback.
7. The doctors should be able to send message, share test results, and records, ensuring accountability and compliance,
8. The patients can request and receive their medical records in a portable format, ensuring data portability in compliance with regulations.
9. The feedback form ensures that it is comprehensive but does not overburden the patient.

3.2 Conceptual ERD

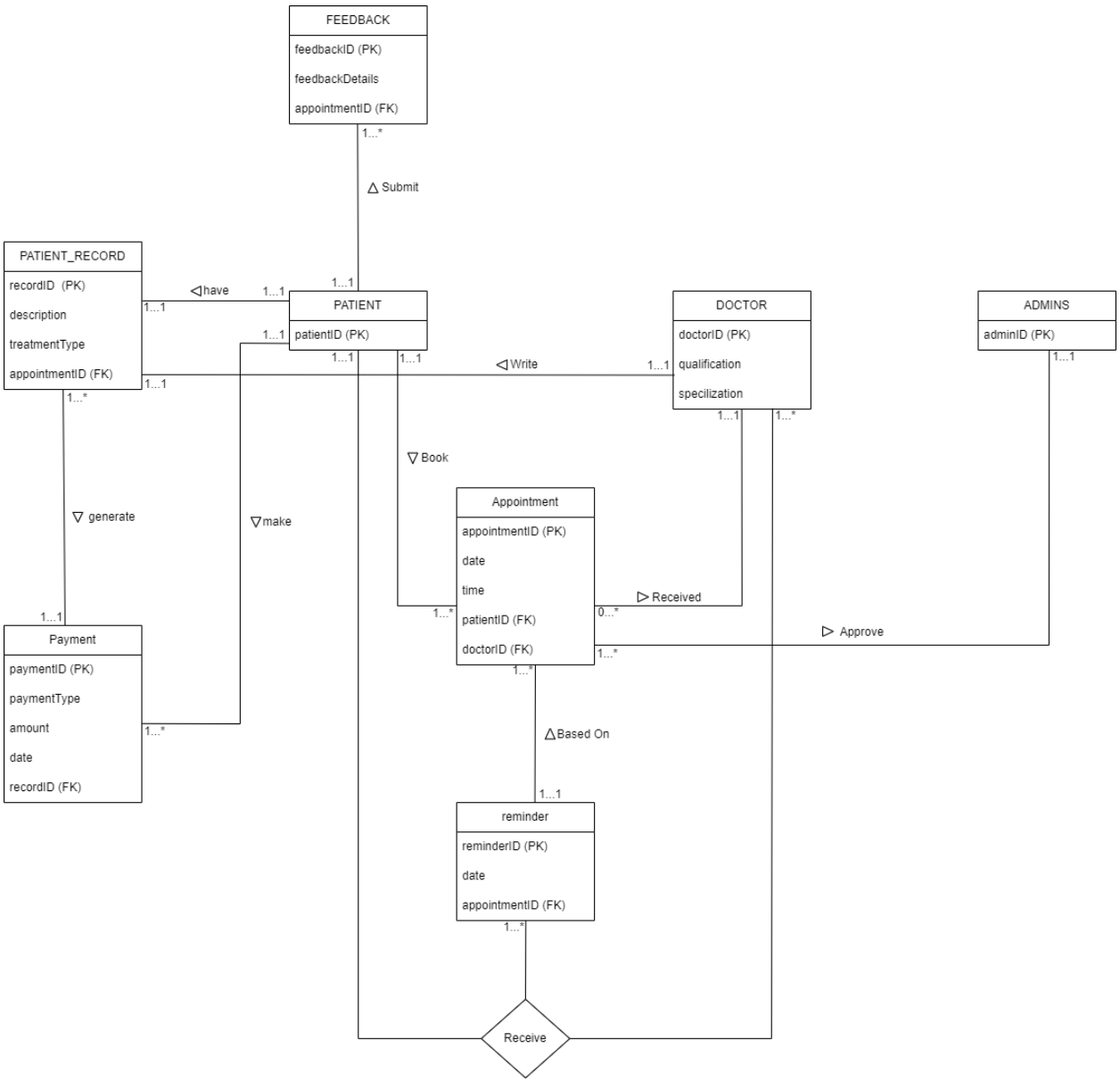


Figure 1: Conceptual ERD

3.3 Enhanced ERD

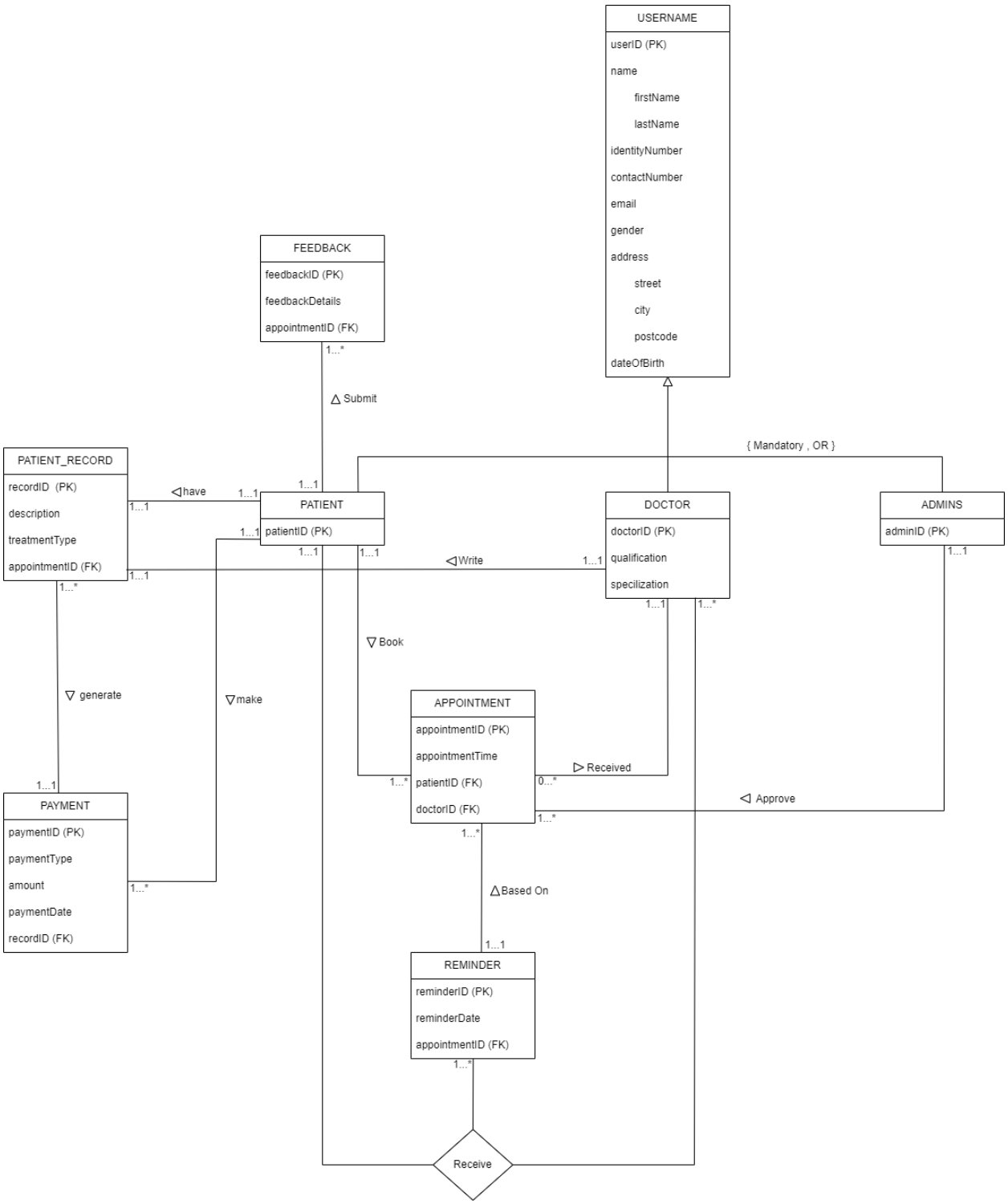


Figure 2: Enhanced ERD

4.0 DB Logical Design

4.1 Logical ERD

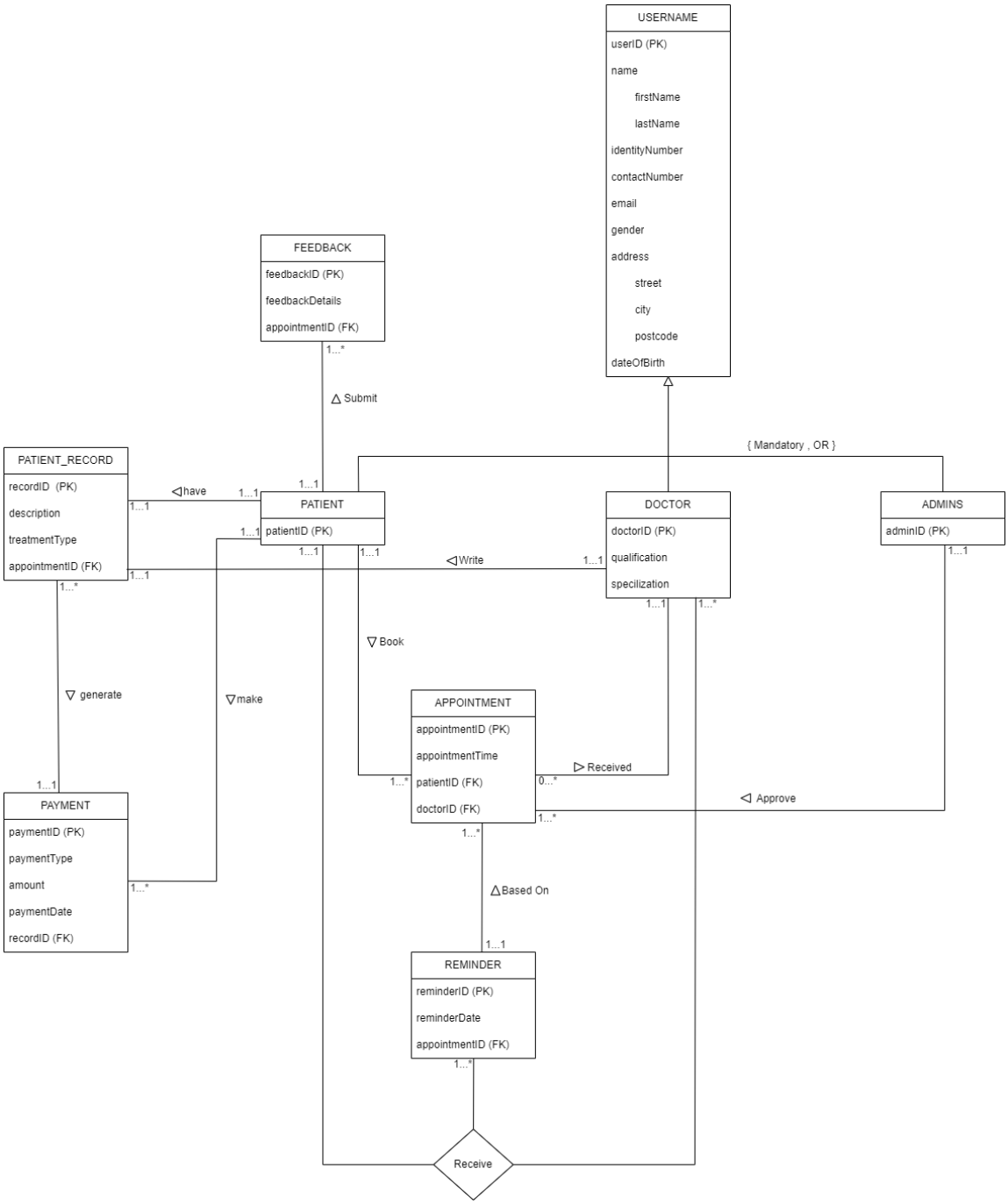


Figure 3: Logical ERD

4.2 Updated Data Dictionary

4.2.1 Description Of Entity

Entity	**Description**	**Occurrence**
USERNAME	Holds the data of users.	Users consist of admins, patients and doctors.
ADMINS	Holds data of admins.	Admin approves appointments and sends reminders to patients and doctors.
PATIENT	Holds the data of patients.	Patient book appointments and make payments.
DOCTOR	Holds the data of doctors.	Doctors receive appointments, write patient records and submit prescriptions.
APPOINTMENT	Holds the data of appointments made by patients.	Appointments are made by patients, and are approved by the admin.
REMINDER	Holds the data of appointment reminders.	Reminders will be sent to the patient and doctor.
PATIENT_RECORD	Holds the data of a patient's medical record.	Patient medical record is written by the doctor.
FEEDBACK	Holds the data of feedback given.	Feedback is given by patients.
PAYMENT	Holds the data of payments made.	Payment is made and proceed by patients.

Table 1: Table of Desciption of Entity.

4.2.2 Description Of Relationship

Entity	**Multiplicity**	**Relationship**	**Multiplicity**	**Entity**
USERNAME	1…1	Has	1…*	Admin
	1…1	Has	1…*	Patient
	1…1	Has	1…*	Doctor
ADMINS	1…1	Approve	1…*	Appointment
PATIENT	1...1	Book	1…*	Appointment
	1…1	Have	1…1	PatientRecord
	1…1	Submit	1…*	Feedback
	1…1	Make	1…*	Payment
	1…1	Receive	1…*	Reminder
PATIENT_RECORD	1…1	Generate	1…*	Payment
DOCTOR	1…1	Accept	0…*	Appointment
	1…1	Write	1…*	PatientRecord
	1…1	Receive	1…*	Reminder
REMINDER	1…1	Based on	1…1	Appointment

Table 2: Table of Description of Relationship.

4.2.3 Description Of Attributes

Entity	Attribute	Description	Data Type	Constraint
USERNAME	userID	User's ID	VARCHAR(10)	PRIMARY KEY
	password	Password	VARCHAR(64)	NOT NULL
	firstName	First name	VARCHAR(20)	NOT NULL
	lastName	Last name	VARCHAR(20)	NOT NULL
	identityNumber	Ic Number	VARCHAR(20)	NOT NULL, UNIQUE
	contactNumber	User's contact number	VARCHAR(15)	NOT NULL, UNIQUE
	email	Email	VARCHAR(30)	NOT NULL
	gender	Gender	VARCHAR(10)	NOT NULL
ADMINS	adminID	Admin's ID	VARCHAR(10)	PRIMARY KEY
	userID	User's ID	VARCHAR(10)	FOREIGN KEY
PATIENT	patientID	Patient's ID	VARCHAR(10)	PRIMARY KEY
	dateOfBirth	Date of birth	DATE	NOT NULL
	address	Address of patient	VARCHAR(30)	NOT NULL
	street	Street name	VARCHAR(20)	NOT NULL
	city	City name	VARCHAR(20)	NOT NULL
	postcode	Postcode number	VARCHAR(10)	NOT NULL
	userID	User's ID	VARCHAR(10)	FOREIGN KEY
DOCTOR	doctorID	Doctor's ID	VARCHAR(10)	PRIMARY KEY
	qualification	Doctor's qualification	VARCHAR(50)	NOT NULL
	specialization	Doctor's specialization	VARCHAR(30)	NOT NULL

	userID	User's ID	VARCHAR(10)	FOREIGN KEY
APPOINTMENT	appointmentID	Appointment's ID	VARCHAR(10)	PRIMARY KEY
	appointmentTime	Appointment Time	TIME	NOT NULL
	patientID	Patient's ID	VARCHAR(10)	FOREIGN KEY
	doctorID	Doctor's ID	VARCHAR(10)	FOREIGN KEY
REMINDER	reminderID	Reminder's ID	VARCHAR(10)	PRIMARY KEY
	reminderDate	Date before appointment	DATE	NOT NULL
	appointmentID	Appointment's ID	VARCHAR(10)	FOREIGN KEY
PATIENT_RECORD	recordID	Record's ID	VARCHAR(10)	PRIMARY KEY
	description	Description	VARCHAR(200)	NOT NULL
	treatmentType	Type of treatment	VARCHAR(50)	NOT NULL
	appointmentID	Appointment's ID	VARCHAR(10)	FOREIGN KEY
	patientID	Patient's ID	VARCHAR(10)	FOREIGN KEY
	doctorID	Doctor's ID	VARCHAR(10)	FOREIGN KEY
FEEDBACK	feedbackID	Feedback's ID	VARCHAR(10)	PRIMARY KEY
	feedbackDetails	Details of the feedback	VARCHAR(200)	NOT NULL
	appointmentID	Appointment's ID	VARCHAR(10)	FOREIGN KEY
	patientID	Patient's ID	VARCHAR(10)	FOREIGN KEY
PAYMENT	paymentID	Payment's ID	VARCHAR(10)	PRIMARY KEY
	paymentType	Type of payment	VARCHAR(10)	NOT NULL
	amount	Amount of	NUMBER(10,2)	NOT NULL

		payment		
	date	Date of payment done	TIMESTAMP	NOT NULL
	recordID	Record's ID	VARCHAR(10)	FOREIGN KEY

Table 3: Table of Description of Attributes.

4.3 Normalization

1. USERNAME (userID, password, firstName, lastName, identityNumber, contactNumber, email, gender)
   fd1: userID → password, firstName, lastName, identityNumber, contactNumber, email, gender
   1NF & 2NF & 3NF & BCNF:
   User (<u>userID</u>, password, firstName, lastName, identityNumber, contactNumber, email, gender)

2. ADMINS (adminID, userID)
   fd1: adminID → userID
   1NF & 2NF & 3NF & BCNF:
   Admin (<u>adminID</u>, userID)

3. PATIENT (patientID, dateOfBirth, address, street, city, postcode, userID)
   fd1: patientID → dateOfBirth, address, street, city, postcode, userID
   1NF & 2NF & 3NF & BCNF:
   Patient (<u>patientID</u>, dateOfBirth, address, street, city, postcode, userID)

4. DOCTOR (doctorID, qualification, specialization, UserID)
   fd1: doctorID → qualification, specialization, UserID
   1NF & 2NF & 3NF & BCNF:
   Doctor (<u>doctorID</u>, qualification, specialization, UserID)

5. APPOINTMENT (appointmentID, appointmentTime, patientID, doctorID)
   fd1: appointmentID → appointmentTime, patientID, doctorID
   1NF & 2NF & 3NF & BCNF:
   Appointment (<u>appointmentID</u>, appointmentTime, patientID, doctorID)

6. REMINDER (reminderID, reminderDate, appointmentID)
   fd1: reminderID → reminderDate, appointmentID
   1NF & 2NF & 3NF & BCNF:
   Reminder (<u>reminderID</u>, reminderDate, appointmentID)

7. PATIENT_RECORD (recordID, description, treatmentType, appointmentID, patientID, doctorID)
   fd1: recordID → description, treatmentType, appointmentID, patientID, doctorID
   1NF & 2NF & 3NF & BCNF:
   PatientRecord (<u>recordID</u>, description, treatmentType, appointmentID, patientID, doctorID)

8. FEEDBACK (feedbackID, feedbackDetails, appointmentID, patientID)
   fd1: feedbackID → feedbackDetails, appointmentID, patientID
   1NF & 2NF & 3NF & BCNF:
   Feedback (feedbackID, feedbackDetails, appointmentID, patientID)

9. PAYMENT (paymentID, paymentType, amount, paymentDate, recordID)
   fd1: paymentID → paymentType, amount, paymentDate, recordID
   1NF & 2NF & 3NF & BCNF:
   Payment (<u>paymentID</u>, paymentType, amount, paymentDate, recordID)

5.0 Relational DB Schemas (After Normalization)

These are the set of relation schemas in relational database schema for DialysisXCare database.

USERNAME (userID, password, firstName, lastName, identityNumber, contactNumber, email, gender)

ADMINS (adminID, userID)

PATIENT (patientID, dateOfBirth, address, street, city, postcode, userID)

DOCTOR (doctorID, qualification, specialization, userID)

APPOINTMENT (appointmentID, appointmentTime, patientID, doctorID)

REMINDER (reminderID, reminderDate, appointmentID)

PATIENT_RECORD (recordID, description, treatmentType, appointmentID, patientID, doctorID)

FEEDBACK (feedbackID, feedbackDetails, appointmentID, patientID)

PAYMENT (paymentID, paymentType, amount, paymentDate, recordID)

** Remark: Underline word is primary key

USERNAME

userID	password	firstName	lastName	identityNumber	contactNumber	email	gender

Table 4: Table of Username.

ADMINS

adminID	userID

Table 5: Table of Admins.

PATIENT

patientID	dateOfBirth	address	street	city	postcode	userID

Table 6: Table of Patient.

DOCTOR

doctorID	qualification	specialization	userID

Table 7: Table of Doctor.

APPOINTMENT

appointmentID	appointmentTime	patientID	doctorID

Table 8: Table of Appointment.

REMINDER

reminderID	reminderDate	appointmentID

Table 9: Table of Reminder.

PATIENT_RECORD

recordID	description	treatmentType	appointmentID	patientID	doctorID

Table 10: Table of Patient Record.

FEEDBACK

feedbackID	feedbackDetails	appointmentID	patientID

Table 11: Table of Feedback.

PAYMENT

paymentID	paymentType	amount	paymentDate	recordID

Table 12: Table of Payment.

6.0 SQL Statements

6.1 Data Definition Language (DDL)

```
--DDL: CREATE & ALTER

CREATE TABLE USERNAME (
    userID VARCHAR2(10) CONSTRAINT userID_PK PRIMARY KEY,
    password VARCHAR2(100) NOT NULL,
    firstName VARCHAR2(20) NOT NULL,
    lastName VARCHAR2(20) NOT NULL,
    identityNumber VARCHAR2(20) NOT NULL,
    contactNumber VARCHAR2(15) NOT NULL,
    email VARCHAR2(30) NOT NULL,
    gender VARCHAR2(10) NOT NULL,
    CONSTRAINT identity_UQ UNIQUE (identityNumber),
    CONSTRAINT contact_UQ UNIQUE (contactNumber)
);

-------------------------------------------------------------

CREATE TABLE ADMINS (
     adminID VARCHAR2(10) CONSTRAINT adminID_PK PRIMARY
     KEY,PRIMARY KEY,
     userID VARCHAR2(10),
     CONSTRAINT fk_admin_user FOREIGN KEY (userID) REFERENCES
USERNAME(userID)
);

-------------------------------------------------------------

CREATE TABLE PATIENT(
    patientID VARCHAR2(10) PRIMARY KEY,
    dateOfBirth DATE NOT NULL,
```

```
    address VARCHAR2(30) NOT NULL,
    street VARCHAR2(20) NOT NULL,
    city VARCHAR2(20) NOT NULL,
    postcode VARCHAR2(10) NOT NULL,
    userID VARCHAR2(10),
   CONSTRAINT fk_patient_user FOREIGN KEY (userID) REFERENCES
USERNAME(userID)
);

-----------------------------------------------------------------

CREATE TABLE DOCTOR(
    doctorID VARCHAR2(10) PRIMARY KEY,
    qualification VARCHAR2(50) NOT NULL,
    specialization VARCHAR2(30) NOT NULL,
    userID VARCHAR2(10),
      CONSTRAINT fk_doctor_user FOREIGN KEY (userID) REFERENCES
USERNAME(userID))
);

-----------------------------------------------------------------

CREATE TABLE APPOINTMENT(
   appointmentID VARCHAR2(10) PRIMARY KEY,
   appointmentTime TIMESTAMP NOT NULL,
   patientID VARCHAR2(10),
   doctorID VARCHAR2(10),
   CONSTRAINT fk_appointment_patient FOREIGN KEY (patientID)
     REFERENCES PATIENT(patientID),
   CONSTRAINT fk_appointment_doctor FOREIGN KEY (doctorID)
     REFERENCES DOCTOR(doctorID)
);

-----------------------------------------------------------------
```

```
CREATE TABLE REMINDER(
    reminderID VARCHAR2(10) PRIMARY KEY,
    reminderDate DATE NOT NULL,
    appointmentID VARCHAR2(10),
    CONSTRAINT fk_reminder_appointment FOREIGN
KEY(appointmentID)
       REFERENCES APPOINTMENT(appointmentID)
);
-----------------------------------------------------------------

CREATE TABLE PATIENT_RECORD(
    recordID VARCHAR2(10) PRIMARY KEY,
    description VARCHAR2(200) NOT NULL,
    treatmentType VARCHAR2(50) NOT NULL,
    appointmentID VARCHAR2(10),
    patientID VARCHAR2(10),
    doctorID VARCHAR2(10),
    CONSTRAINT fk_patient_record_appointment FOREIGN KEY
(appointmentID) REFERENCES APPOINTMENT(appointmentID),
    CONSTRAINT fk_patient_record_patient FOREIGN KEY (patientID)
REFERENCES PATIENT(patientID),
    CONSTRAINT fk_patient_record_doctor FOREIGN KEY (doctorID)
REFERENCES DOCTOR(doctorID)
);
-----------------------------------------------------------------
```

```
CREATE TABLE FEEDBACK(
    feedbackID VARCHAR2(10) PRIMARY KEY,
    feedbackDetails VARCHAR2(200) NOT NULL,
    appointmentID VARCHAR2(10),
    patientID VARCHAR2(10),
    CONSTRAINT fk_feedback_appointment FOREIGN KEY
(appointmentID) REFERENCES APPOINTMENT(appointmentID),
    CONSTRAINT fk_feedback_patient FOREIGN KEY (patientID)
REFERENCES PATIENT(patientID)
);
-----------------------------------------------------------------
CREATE TABLE PAYMENT(
    paymentID VARCHAR2(10) PRIMARY KEY,
    paymentType VARCHAR2(10) NOT NULL,
    amount number(10,2) NOT NULL,
    paymentDate TIMESTAMP NOT NULL,
    recordID VARCHAR2(10),
    CONSTRAINT fk_payment_record FOREIGN KEY (recordID)
REFERENCES PATIENT_RECORD(recordID)
);
-----------------------------------------------------------------
ALTER TABLE USERNAME
     ADD CONSTRAINT uk_user_identityNumber
     UNIQUE (identityNumber);
-----------------------------------------------------------------
ALTER TABLE USERNAME
     ADD CONSTRAINT uk_user_contactNumber
     UNIQUE (contactNumber);
```

6.2 Data Manipulation Language (DML)

```
--DML: INSERT

--USERNAME

INSERT INTO USERNAME(userId, password, firstName, lastName,
identityNumber, contactNumber, email, gender)
VALUES('US0001',
'4331803d82bcacd1217e08314a5c8c01f4be34322956de53c1d42979d6bdebf
5', 'Oliver', 'Mitchell', '567-23-8901', '987-6543-2101',
'olivermitchell@gmail.com', 'MALE');

INSERT INTO USERNAME(userId, password, firstName, lastName,
identityNumber, contactNumber, email, gender)
VALUES ('US0002',
'065bf7f9700a59a766e23a9289cfd177aa6c9bf32c58ab12e6265731953dc64
d', 'Olivia', 'Thompson', '987-45-2310', '987-6543-2121',
'oliviathompson@gmail.com', 'FEMALE');

INSERT INTO USERNAME(userId, password, firstName, lastName,
identityNumber, contactNumber, email, gender)
VALUES ('US0003',
'23b624b0f420f783d300bcb1f6dfa6df8c2346ed6f2984f0be47219e609bb05
9', 'Liam', 'Rodriguez', '123-67-8902', '456-8901-2345',
'liamrodriguez@gmail.com', 'MALE');

INSERT INTO USERNAME(userId, password, firstName, lastName,
identityNumber, contactNumber, email, gender)
VALUES ('US0004',
'800460f08e392aeaa98dd0cddfd908395aa657f3fbfc0129677e7e4a42537d3
c', 'Emma', 'Mitchell', '876-12-3456', '321-5678-9012',
'emmamitchell@gmail.com', 'FEMALE');
```

```
INSERT INTO USERNAME(userId, password, firstName, lastName,
identityNumber, contactNumber, email, gender)
VALUES ('US0005',
'acf3684ef248b824eb4fd3af2e438346c6bebb706e7e1e28a11c6d2d77d762c
9', 'Noah', 'Sanchez', '345-89-0123', '789-0123-4567',
'noahsanchez@gmail.com', 'MALE');

INSERT INTO USERNAME(userId, password, firstName, lastName,
identityNumber, contactNumber, email, gender)
VALUES ('US0006',
'99e4217ae0788fd2ad411f248a9d7e9a12b66c74d229844f8f853c53964ede7
8', 'Ava', 'Campbell', '654-32-1098', '234-5678-9010',
'avacampbell@gmail.com', 'FEMALE');

INSERT INTO USERNAME(userId, password, firstName, lastName,
identityNumber, contactNumber, email, gender)
VALUES ('US0007',
'e046ceb020636e25a9e7c3e6be66f907f74b24ff4565976d057adb949fe2931
a', 'Ethan', 'Baker', '210-98-7654', '890-1234-5678',
'ethanbaker@gmail.com', 'FEMALE');

INSERT INTO USERNAME(userId, password, firstName, lastName,
identityNumber, contactNumber, email, gender)
VALUES ('US0008',
'027b0be5905d915473f428c0298e38c5562ddfb99a9a5af2b91fd8bce096e3d
f', 'Sophia', 'Carter', '432-56-7890', '345-6789-0121',
'sophiacarter@gmail.com', 'FEMALE');

INSERT INTO USERNAME(userId, password, firstName, lastName,
identityNumber, contactNumber, email, gender)
```

```
VALUES ('US0009',
'00ba3fe2a5136ca2f1d5cf0385852ccc814df00b27cfcfb692b0544736f1f4a
f', 'Jackson', 'Flores', '789-01-2345', '678-9012-3456',
'jacksonflores@gmail.com', 'MALE');

INSERT INTO USERNAME(userId, password, firstName, lastName,
identityNumber, contactNumber, email, gender)
VALUES ('US0010',
'1f844c8f1c55b2468096a6a1bb1169a1893c7c31c2daf3c91a1f20cd4a63958
3', 'Isabella', 'Martinez', '345-67-8901', '432-1098-7654',
'isabellamartinez@gmail.com', 'MALE');

INSERT INTO USERNAME(userId, password, firstName, lastName,
identityNumber, contactNumber, email, gender)
VALUES ('US0011',
'4ee32cadef2800d428eb0a6d5e93db075680f1bdff012a1a065d081a898c376
c', 'Aiden', 'Wright', '890-12-3456', '567-8901-2345',
'aidenwright@gmail.com', 'MALE');

INSERT INTO USERNAME(userId, password, firstName, lastName,
identityNumber, contactNumber, email, gender)
VALUES ('US0012',
'9aa5b5aae5deb20120ab6afedc83a1b57e9a5a04b14e5af3685cd1565f11af1
8', 'Mia', 'Turner', '456-78-9012', '901-2345-6789',
'miaturner@gmail.com', 'FEMALE');

INSERT INTO USERNAME(userId, password, firstName, lastName,
identityNumber, contactNumber, email, gender)
VALUES ('US0013',
'2dca5dc0387afa17fb930e2be214aa12a706fb957127cf9ea2042cf0c33d0db
e', 'Lucas', 'Johnson', '321-09-8765', '210-9876-5432',
'lucasjohnson@gmail.com', 'MALE');
```

```
INSERT INTO USERNAME(userId, password, firstName, lastName,
identityNumber, contactNumber, email, gender)
VALUES ('US0014',
'67cbe52c30cda798f763d7792adb2ba7319d249e31c240f05eb7447c908f7b0
4', 'Harper', 'Taylor', '567-89-0123', '543-2109-8765',
'harpertaylor@gmail.com', 'MALE');

INSERT INTO USERNAME(userId, password, firstName, lastName,
identityNumber, contactNumber, email, gender)
VALUES ('US0015',
'909eed3b64d30dc98418bf3cf4003afd147dbe02d648bd8cf6e082d428f23d3
3', 'Amelia', 'Davis', '901-23-4567', '876-5432-1098',
'ameliadavis@gmail.com', 'FEMALE');

INSERT INTO USERNAME(userId, password, firstName, lastName,
identityNumber, contactNumber, email, gender)
VALUES ('US0016',
'c699151080506dbdb8bef287e2188da9661f6c9ba536f83ba014db61132d4b0
4', 'Oliver', 'Williams', '234-56-7890', '109-8765-4323',
'oliverwilliams@gmail.com', 'MALE');

INSERT INTO USERNAME(userId, password, firstName, lastName,
identityNumber, contactNumber, email, gender)
VALUES ('US0017',
'7fd1217f4b5eef82c5d196c369c6e60512633b799aae82cf953c150cb31d2e1
c', 'Hannah', 'Kamila', '789-90-1234', '765-4321-0987',
'hannahkamila@gmail.com', 'FEMALE');

INSERT INTO USERNAME(userId, password, firstName, lastName,
identityNumber, contactNumber, email, gender)
```

```
VALUES ('US0018',
'6325bbd2de1ed11ba283d942454623c4499466aca9c289730d1ac9df26864bf
a', 'Abigail', 'Scott', '543-21-0987', '432-1098-1111',
'abigailscott@gmail.com', 'MALE');

INSERT INTO USERNAME(userId, password, firstName, lastName,
identityNumber, contactNumber, email, gender)
VALUES ('US0019',
'784416209fcc22a477f3a2ffa998120c3cd4ab676bd676fb342691409b916bd
3', 'Mason', 'White', '876-54-3210', '876-5432-1098',
'masonwhite@gmail.com', 'MALE');

INSERT INTO USERNAME(userId, password, firstName, lastName,
identityNumber, contactNumber, email, gender)
VALUES ('US0020',
'a4c422e5c99de9838fe50576625fa4668cdc62ee315eba568c521f322c1bdce
4', 'Evelyn', 'Lewis', '109-87-6543', '109-8765-4321',
'evelynlewis@gmail.com', 'MALE');
```

```
--ADMINS

INSERT INTO ADMINS(adminID, userID)
VALUES('AD0001', 'US0001');


INSERT INTO ADMINS(adminID, userID)
VALUES('AD0002', 'US0002');


--PATIENT

INSERT INTO PATIENT(patientID, dateOfBirth, address, street,
city, postcode, userID)
VALUES('PI0001', TO_DATE('1989-03-15', 'YYYY-MM-DD'), '123',
'Maple Avenue', 'Elm City', '56789', 'US0011');

INSERT INTO PATIENT(patientID, dateOfBirth, address, street,
city, postcode, userID)
VALUES('PI0002', TO_DATE('1975-11-05', 'YYYY-MM-DD'), '456',
'Oak Lane', 'Pine City', '12345', 'US0012');

INSERT INTO PATIENT(patientID, dateOfBirth, address, street,
city, postcode, userID)
VALUES('PI0003', TO_DATE('1992-04-28', 'YYYY-MM-DD'), '789',
'Cedar Road', 'Birch City', '67890', 'US0013');

INSERT INTO PATIENT(patientID, dateOfBirth, address, street,
city, postcode, userID)
VALUES('PI0004', TO_DATE('1983-09-12', 'YYYY-MM-DD'), '321',
'Pine Drive', 'Maple City', '23456', 'US0014');
```

```
INSERT INTO PATIENT(patientID, dateOfBirth, address, street,
city, postcode, userID)
VALUES('PI0005', TO_DATE('1978-06-07', 'YYYY-MM-DD'), '654',
'Elmm Bouvelard', 'Oak City', '78901', 'US0015');

INSERT INTO PATIENT(patientID, dateOfBirth, address, street,
city, postcode, userID)
VALUES('PI0006', TO_DATE('1996-01-23', 'YYYY-MM-DD'), '987',
'Birch Court', 'Cedar City', '34567', 'US0016');

INSERT INTO PATIENT(patientID, dateOfBirth, address, street,
city, postcode, userID)
VALUES('PI0007', TO_DATE('1985-08-09', 'YYYY-MM-DD'), '210',
'Pine Lane', 'Pine City', '89012', 'US0017');

INSERT INTO PATIENT(patientID, dateOfBirth, address, street,
city, postcode, userID)
VALUES('PI0008', TO_DATE('1970-12-14', 'YYYY-MM-DD'), '543',
'Oak Avenue', 'Elm City', '45678', 'US0018');

INSERT INTO PATIENT(patientID, dateOfBirth, address, street,
city, postcode, userID)
VALUES('PI0009', TO_DATE('1990-05-31', 'YYYY-MM-DD'), '876',
'Cedar Road', 'Birch City', '12345', 'US0019');

INSERT INTO PATIENT(patientID, dateOfBirth, address, street,
city, postcode, userID)
VALUES('PI0010', TO_DATE('1982-10-18', 'YYYY-MM-DD'), '109',
'Birch Drive', 'Oak City', '12345', 'US0020');
```

```
--DOCTOR

INSERT INTO DOCTOR(doctorID, qualification, specialization,
userID)dr0001
VALUES('DR0001', 'Doctor of Medicine (MD)', 'Nephrology',
'US0003');

INSERT INTO DOCTOR(doctorID, qualification, specialization,
userID)
VALUES('DR0002', 'Master of Medicine (MMed)', 'Dialysis
Medicine', 'US0004');

INSERT INTO DOCTOR(doctorID, qualification, specialization,
userID)
VALUES('DR0003', 'Doctor of Medicine (MD)', 'Internal Medicine',
'US0005');

INSERT INTO DOCTOR(doctorID, qualification, specialization,
userID)
VALUES('DR0004', 'Doctor of Medicine (MD)', 'Internal Medicine',
'US0006');

INSERT INTO DOCTOR(doctorID, qualification, specialization,
userID)
VALUES('DR0005', 'Doctor of Medicine (MD)', 'Clinical
Nephrology', 'US0007');


INSERT INTO DOCTOR(doctorID, qualification, specialization,
userID)
VALUES('DR0006', 'Doctor of Philosophy (PhD) in Medicine',
'Renal Medicine', 'US0008');
```

```
INSERT INTO DOCTOR(doctorID, qualification, specialization,
userID)
VALUES('DR0007', 'Doctor of Philosophy (PhD) in Medicine',
'Renal Medicine', 'US0009');

INSERT INTO DOCTOR(doctorID, qualification, specialization,
userID)
VALUES('DR0008', 'Doctor of Philosophy (PhD) in Medicine',
'Renal Medicine', 'US0010');

--APPOINTMENT

INSERT INTO APPOINTMENT(appointmentID, appointmentTime,
patientID, doctorID)
VALUES('AP0001', TO_TIMESTAMP('2024-10-04 09:00:00', 'YYYY-MM-DD
HH24:MI:SS'), 'PI0001', 'DR0001');

INSERT INTO APPOINTMENT(appointmentID, appointmentTime,
patientID, doctorID)
VALUES('AP0002', TO_TIMESTAMP('2024-04-08 09:00:00', 'YYYY-MM-DD
HH24:MI:SS'), 'PI0002', 'DR0002');

INSERT INTO APPOINTMENT(appointmentID, appointmentTime,
patientID, doctorID)
VALUES('AP0003', TO_TIMESTAMP('2024-07-26 09:00:00', 'YYYY-MM-DD
HH24:MI:SS'), 'PI0003', 'DR0003');

INSERT INTO APPOINTMENT(appointmentID, appointmentTime,
patientID, doctorID)
VALUES('AP0004', TO_TIMESTAMP('2024-09-14 09:00:00', 'YYYY-MM-DD
HH24:MI:SS'), 'PI0004', 'DR0004');
```

```
INSERT INTO APPOINTMENT(appointmentID, appointmentTime,
patientID, doctorID)
VALUES('AP0005', TO_TIMESTAMP('2024-11-30 09:00:00', 'YYYY-MM-DD
HH24:MI:SS'), 'PI0005', 'DR0005');

INSERT INTO APPOINTMENT(appointmentID, appointmentTime,
patientID, doctorID)
VALUES('AP0006', TO_TIMESTAMP('2024-06-17 09:00:00', 'YYYY-MM-DD
HH24:MI:SS'), 'PI0006', 'DR0001');

INSERT INTO APPOINTMENT(appointmentID, appointmentTime,
patientID, doctorID)
VALUES('AP0007', TO_TIMESTAMP('2024-08-10 09:00:00', 'YYYY-MM-DD
HH24:MI:SS'), 'PI0007', 'DR0002');

INSERT INTO APPOINTMENT(appointmentID, appointmentTime,
patientID, doctorID)
VALUES('AP0008', TO_TIMESTAMP('2024-01-28 09:00:00', 'YYYY-MM-DD
HH24:MI:SS'), 'PI0008', 'DR0003');

INSERT INTO APPOINTMENT(appointmentID, appointmentTime,
patientID, doctorID)
VALUES('AP0009', TO_TIMESTAMP('2024-03-22 09:00:00', 'YYYY-MM-DD
HH24:MI:SS'), 'PI0009', 'DR0004');

INSERT INTO APPOINTMENT(appointmentID, appointmentTime,
patientID, doctorID)
VALUES('AP0010', TO_TIMESTAMP('2024-10-04 09:00:00', 'YYYY-MM-DD
HH24:MI:SS'), 'PI0010', 'DR0005');
```

```sql
--REMINDER

INSERT INTO REMINDER(reminderID, reminderDate, appo  intmentID)
VALUES('RM0001', TO_DATE('2024-02-16', 'YYYY-MM-DD'), 'AP0001');

INSERT INTO REMINDER(reminderID, reminderDate, appointmentID)
VALUES('RM0002', TO_DATE('2024-04-07', 'YYYY-MM-DD'), 'AP0002');

INSERT INTO REMINDER(reminderID, reminderDate, appointmentID)
VALUES('RM0003', TO_DATE('2024-07-25', 'YYYY-MM-DD'), 'AP0003');

INSERT INTO REMINDER(reminderID, reminderDate, appointmentID)
VALUES('RM0004', TO_DATE('2024-09-13', 'YYYY-MM-DD'), 'AP0004');

INSERT INTO REMINDER(reminderID, reminderDate, appointmentID)
VALUES('RM0005', TO_DATE('2024-11-29', 'YYYY-MM-DD'), 'AP0005');

INSERT INTO REMINDER(reminderID, reminderDate, appointmentID)
VALUES('RM0006', TO_DATE('2024-06-16', 'YYYY-MM-DD'), 'AP0006');

INSERT INTO REMINDER(reminderID, reminderDate, appointmentID)
VALUES('RM0007', TO_DATE('2024-08-09', 'YYYY-MM-DD'), 'AP0007');

INSERT INTO REMINDER(reminderID, reminderDate, appointmentID)
VALUES('RM0008', TO_DATE('2024-01-27', 'YYYY-MM-DD'), 'AP0008');

INSERT INTO REMINDER(reminderID, reminderDate, appointmentID)
VALUES('RM0009', TO_DATE('2024-03-21', 'YYYY-MM-DD'), 'AP0009');

INSERT INTO REMINDER(reminderID, reminderDate, appointmentID)
VALUES('RM0010', TO_DATE('2024-10-03', 'YYYY-MM-DD'), 'AP0010');
```

```
--PATIENT_RECORD

Remarks: Acronym for HD is Hemodialysis  and PD is Peritoneal
dialysis

INSERT INTO PATIENT_RECORD(recordID, description, treatmentType,
appointmentID, patientID, doctorID)
VALUES('PR0001', 'Aiden Wright sleeps through nocturnal
hemodialysis sessions, providing a restful night and promoting
overall well-being.', 'HD', 'AP0001', 'PI0001', 'DR0001');

INSERT INTO PATIENT_RECORD(recordID, description, treatmentType,
appointmentID, patientID, doctorID)
VALUES('PR0002', 'Mia Turner undergoes HVHDF, a specialized
hemodialysis technique for advanced toxin clearance, supporting
his health goals.', 'PD', 'AP0002', 'PI0002', 'DR0002');

INSERT INTO PATIENT_RECORD(recordID, description, treatmentType,
appointmentID, patientID, doctorID)
VALUES('PR0003', 'Lucas Johnson attends her routine
hemodialysis, where a machine filters her blood to manage kidney
failure by removing waste and excess fluids.', 'HD', 'AP0003',
'PI0003', 'DR0003');

INSERT INTO PATIENT_RECORD(recordID, description, treatmentType,
appointmentID, patientID, doctorID)
VALUES('PR0004', 'Seeking efficient toxin removal, Harper Taylor
opts for daily hemodialysis, maintaining better fluid balance
for an improved quality of life.', 'PD', 'AP0004', 'PI0004',
'DR0004');
```

```
INSERT INTO PATIENT_RECORD(recordID, description, treatmentType,
appointmentID, patientID, doctorID)
VALUES('PR0005', 'Amelia Davis experiences pediatric
hemodialysis, with equipment and strategies designed for
children''s comfort and effectiveness.', 'HD', 'AP0005',
'PI0005', 'DR0005');

INSERT INTO PATIENT_RECORD(recordID, description, treatmentType,
appointmentID, patientID, doctorID)
VALUES('PR0006', 'Oliver Williams enjoys the independence of
peritoneal dialysis at home, effectively managing kidney failure
with a cleansing solution.', 'PD', 'AP0006', 'PI0006',
'DR0001');

INSERT INTO PATIENT_RECORD(recordID, description, treatmentType,
appointmentID, patientID, doctorID)
VALUES('PR0007', 'Hannah Kamila chooses CCPD, nightly peritoneal
dialysis providing effective waste removal with flexibility for
her day.', 'HD', 'AP0007', 'PI0007', 'DR0002');

INSERT INTO PATIENT_RECORD(recordID, description, treatmentType,
appointmentID, patientID, doctorID)
VALUES('PR0008', 'Abigail Scott opts for APD, automated
peritoneal dialysis that fits well into her daily routine,
offering effective treatment without constant manual
involvement.', 'PD', 'AP0008', 'PI0008', 'DR0003');
```

```
INSERT INTO PATIENT_RECORD(recordID, description, treatmentType,
appointmentID, patientID, doctorID)
VALUES('PR0009', 'Mason White prefers intermittent peritoneal
dialysis for flexibility, adapting his routine to daily life
while managing kidney failure.', 'HD', 'AP0009', 'PI0009',
'DR0004');

INSERT INTO PATIENT_RECORD(recordID, description, treatmentType,
appointmentID, patientID, doctorID)
VALUES('PR0010', 'Evelyn Lewis undergoes peritoneal dialysis
tailored for pediatric patients, ensuring a comfortable and
age-appropriate experience.', 'PD', 'AP0010', 'PI0010',
'DR0005');
```

```
--FEEDBACK

INSERT INTO FEEDBACK(feedbackID, feedbackDetails, appointmentID,
patientID)
VALUES('FB0001', 'I appreciate the caring and attentive staff.
They made me feel comfortable and at ease.', 'AP0001',
'PI0001');


INSERT INTO FEEDBACK(feedbackID, feedbackDetails, appointmentID,
patientID)
VALUES('FB0002', 'The dialysis team is fantastic! Always
friendly and supportive during my sessions.', 'AP0002',
'PI0002');

INSERT INTO FEEDBACK(feedbackID, feedbackDetails, appointmentID,
patientID)
VALUES('FB0003', 'I appreciate the convenience of hemodialysis,
making it easier to integrate into my daily life.', 'AP0003',
'PI0003');

INSERT INTO FEEDBACK(feedbackID, feedbackDetails, appointmentID,
patientID)
VALUES('FB0004', 'Nocturnal hemodialysis allows me to sleep
through the sessions, making my nights more restful. Appreciate
the flexibility.', 'AP0004', 'PI0004');

INSERT INTO FEEDBACK(feedbackID, feedbackDetails, appointmentID,
patientID)
VALUES('FB0005', 'HVHDF hemodialysis has improved my overall
well-being. Thank you to the team for their expertise and
support.', 'AP0005', 'PI0005');
```

```
INSERT INTO FEEDBACK(feedbackID, feedbackDetails, appointmentID,
patientID)
VALUES('FB0006', 'The routine hemodialysis sessions have become
a consistent and essential part of managing my kidney health.
The staff is excellent.', 'AP0006', 'PI0006');

INSERT INTO FEEDBACK(feedbackID, feedbackDetails, appointmentID,
patientID)
VALUES('FB0007', 'Home peritoneal dialysis provides the
independence I need. Grateful for the simplicity and
effectiveness of the process.', 'AP0007', 'PI0007');

INSERT INTO FEEDBACK(feedbackID, feedbackDetails, appointmentID,
patientID)
VALUES('FB0008', 'CCPD has made nightly sessions convenient.
Effective waste removal without disrupting my daily routine.',
'AP0008', 'PI0008');

INSERT INTO FEEDBACK(feedbackID, feedbackDetails, appointmentID,
patientID)
VALUES('FB0009', 'APD has made peritoneal dialysis more
accessible. The automated process is a game-changer for my
lifestyle.', 'AP0009', 'PI0009');

INSERT INTO FEEDBACK(feedbackID, feedbackDetails, appointmentID,
patientID)
VALUES('FB0010', 'Intermittent peritoneal dialysis gives me the
flexibility to adapt my treatment to my daily life. It''s been a
positive experience.', 'AP0010', 'PI0010');
```

```
--PAYMENT

Remarks: Acronym for DB is Debit, CH is Cash and Sp is
Sponsorship

INSERT INTO payment(paymentID, paymentType, amount, paymentDate,
recordID)
VALUES('PM0001', 'CH', 240, TO_DATE('2024-01-17', 'YYYY-MM-DD'),
'PR0001');

INSERT INTO payment(paymentID, paymentType, amount, paymentDate,
recordID)
VALUES('PM0002', 'DB', 400, TO_DATE('2024-03-08', 'YYYY-MM-DD'),
'PR0002');

INSERT INTO payment(paymentID, paymentType, amount, paymentDate,
recordID)
VALUES('PM0003', 'SP', 240, TO_DATE('2024-06-26', 'YYYY-MM-DD'),
'PR0003');

INSERT INTO payment(paymentID, paymentType, amount, paymentDate,
recordID)
VALUES('PM0004', 'DB', 400, TO_DATE('2024-08-14', 'YYYY-MM-DD'),
'PR0004');

INSERT INTO payment(paymentID, paymentType, amount, paymentDate,
recordID)
VALUES('PM0005', 'DB', 400, TO_DATE('2024-10-30', 'YYYY-MM-DD'),
'PR0005');

INSERT INTO payment(paymentID, paymentType, amount, paymentDate,
recordID)
```

```
VALUES('PM0006', 'DB', 240, TO_DATE('2024-05-17', 'YYYY-MM-DD'),
'PR0006');

INSERT INTO payment(paymentID, paymentType, amount, paymentDate,
recordID)
VALUES('PM0007', 'DB', 400, TO_DATE('2024-07-10', 'YYYY-MM-DD'),
'PR0007');

INSERT INTO payment(paymentID, paymentType, amount, paymentDate,
recordID)
VALUES('PM0008', 'SP', 240, TO_DATE('2023-12-28', 'YYYY-MM-DD'),
'PR0008');

INSERT INTO payment(paymentID, paymentType, amount, paymentDate,
recordID)
VALUES('PM0009', 'CH', 240, TO_DATE('2024-02-22', 'YYYY-MM-DD'),
'PR0009');

INSERT INTO payment(paymentID, paymentType, amount, paymentDate,
recordID)
VALUES('PM0010', 'DB', 400, TO_DATE('2024-09-04', 'YYYY-MM-DD'),
'PR0010');
```

```
--DML: SELECT

SELECT * FROM APPOINTMENT
WHERE doctorID = 'DR0001';
```

APPOINTMENTID	APPOINTMENTTIME	PATIENTID	DOCTORID
AP0001	04-OCT-24 09.00.00.000000 AM	PI0001	DR0001
AP0006	17-JUN-24 09.00.00.000000 AM	PI0006	DR0001

2 rows returned in 0.01 seconds Download

Figure 4: Table of Appointment where DoctorID = “DR0001”.

```
SELECT FEEDBACKDETAILS FROM FEEDBACK
WHERE feedbackID = 'DR0001';
```

FEEDBACKDETAILS
Nocturnal hemodialysis allows me to sleep through the sessions, making my nights more restful. Appreciate the flexibility.

Figure 5: Table of FeedbackDetails from Feedback Table where feedbackID = “DR0001”.

```
SELECT *
FROM DOCTOR D
JOIN "USERNAME" U ON D.userID = U.userID
WHERE U.userID = 'US0010';
```

DOCTORID	QUALIFICATION	SPECIALIZATION	USERID	USERID	FIRSTNAME	LASTNAME	IDENTITYNUMBER	CONTACTNUMBER	EMAIL	GENDER
DR0008	Doctor of Philosophy (PhD) in Medicine	Renal Medicine	US0010	US0010	Isabella	Martinez	345-67-8901	432-1098-7654	isabellamartinez@gmail.com	MALE

Figure 6: Table of Joint Doctor and Username where UserID = “US001”

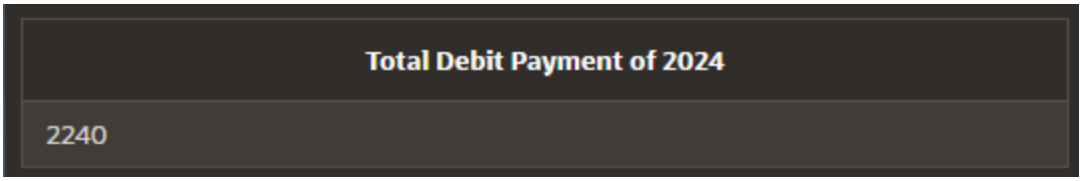
```
SELECT * FROM USERNAME U
    JOIN DOCTOR D
        ON U.userID = D.userID
    JOIN PATIENT_RECORD PR
        ON D.doctorID = PR.doctorID
WHERE U.lastName = 'Mitchell';
```

USERID	PASSWORD	FIRSTNAME	LASTNAME	IDENTITYNUMBER	CONTACTNUMBER	EMAIL	GENDER	DOCTORID	QUALIFICATION	SPECIALIZATION	USERID	RECORDID	DESCRIPTION	TREATMENTTYPE	APPOINTMENTID	PATIENTID	DOCTORID
US0004	800460f08e392aeaa98dd0cddfd908395aa657f3fbfc0129677e7e4a42537d3c	Emma	Mitchell	876-12-3456	321-5678-9012	emmamitchell@gmail.com	FEMALE	DR0002	Master of Medicine (MMed)	Dialysis Medicine	US0004	PR0002	Mia Turner undergoes HVHDF, a specialized hemodialysis technique for advanced toxin clearance, supporting his health goals.	PD	AP0002	PI0002	DR0002
US0004	800460f08e392aeaa98dd0cddfd908395aa657f3fbfc0129677e7e4a42537d3c	Emma	Mitchell	876-12-3456	321-5678-9012	emmamitchell@gmail.com	FEMALE	DR0002	Master of Medicine (MMed)	Dialysis Medicine	US0004	PR0007	Hannah Kamila chooses CCPD, nightly peritoneal dialysis providing effective waste removal with flexibility for her day.	HD	AP0007	PI0007	DR0002

2 rows returned in 0.01 seconds Download

Figure 7: Table of Joint Doctor and Username where UserID = “US001”

```
SELECT SUM(amount) AS "Total Debit Payment of 2024" FROM PAYMENT
WHERE paymentType = 'DB'
  AND EXTRACT(YEAR FROM paymentDate) = 2024;
```

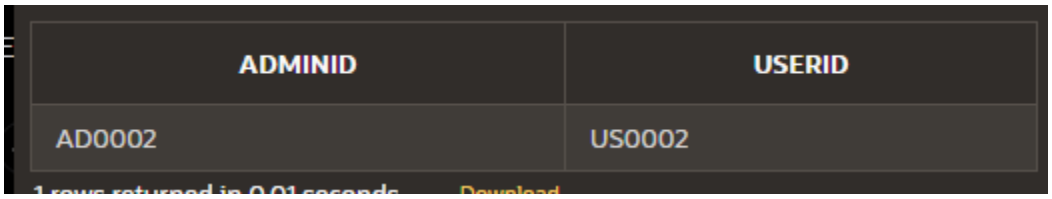


Total Debit Payment of 2024
2240

Figure 8: Table of sum as “Total Debit Payment of 2024” from Payment where paymentType = “DB” and extract from year = “2024”.J

```
--DML: DELETE

DELETE FROM ADMINS
    WHERE adminID = 'AD0001'
SELECT * FROM ADMINS;
```



ADMINID	USERID
AD0002	US0002

Figure 9: Table of Admins where adminID = “AD0002” deleted.

```
--DML: UPDATE

UPDATE APPOINTMENT
SET doctorID = 'DR0006'
WHERE appointmentID = 'AP0005'
```

APPOINTMENTID	APPOINTMENTTIME	PATIENTID	DOCTORID
AP0005	30-NOV-24 09.00.00.000000 AM	PI0005	DR0005

1 rows returned in 0.01 seconds Download

Figure 10: Table Appointment for AppointmentID = “AP0005” BEFORE update.

APPOINTMENTID	APPOINTMENTTIME	PATIENTID	DOCTORID
AP0005	30-NOV-24 09.00.00.000000 AM	PI0005	DR0006

1 rows returned in 0.01 seconds Download

Figure 11: Table Appointment for AppointmentID = “AP0005” AFTER update.

7.0 User Interface

7.1 General View

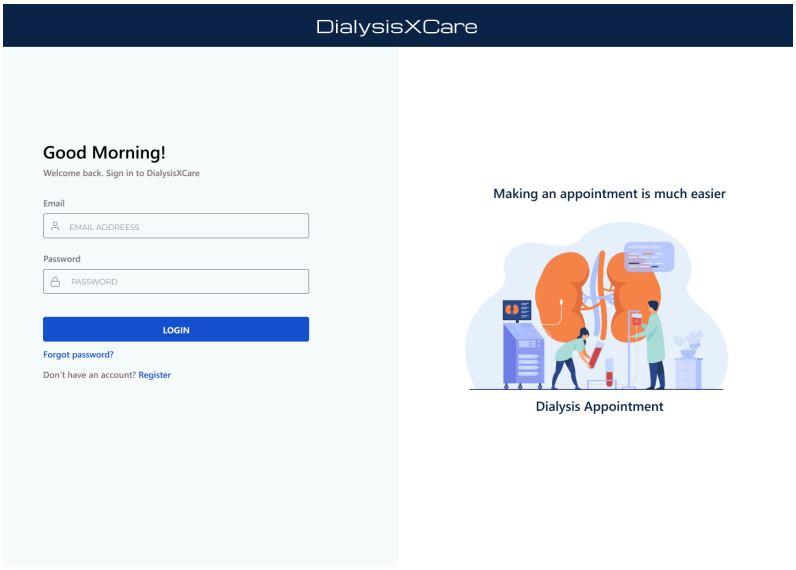


Figure 12: Login Page

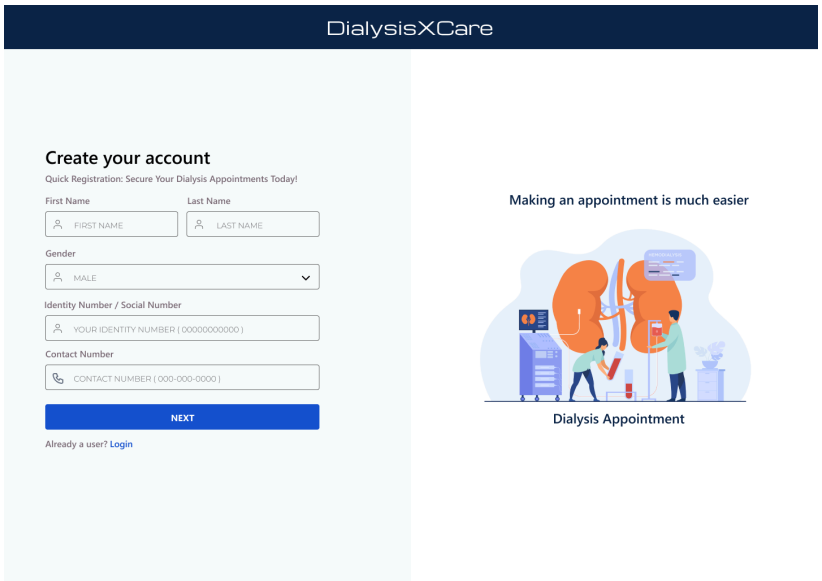


Figure 13: Create Account Page - 1

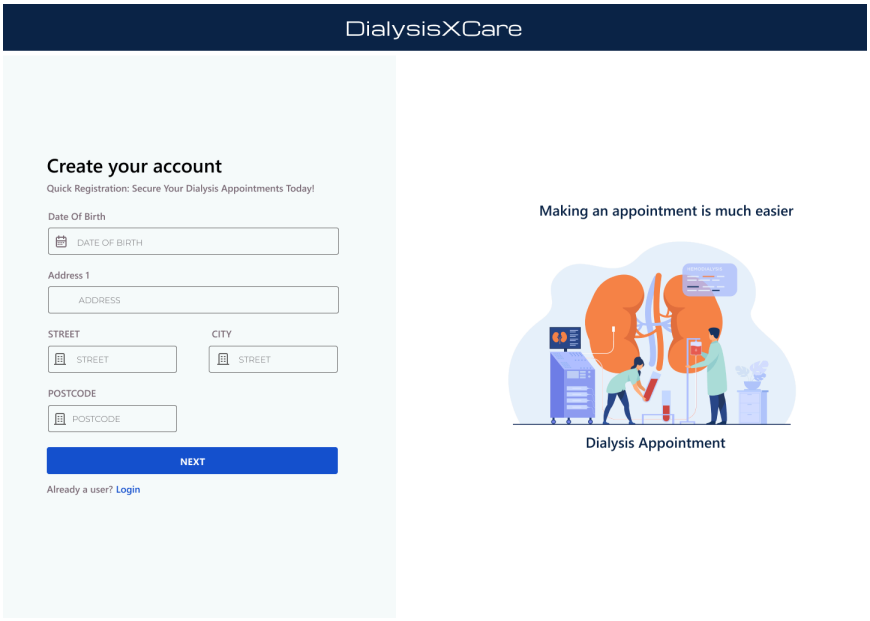


Figure 14: Create Account Page - 2

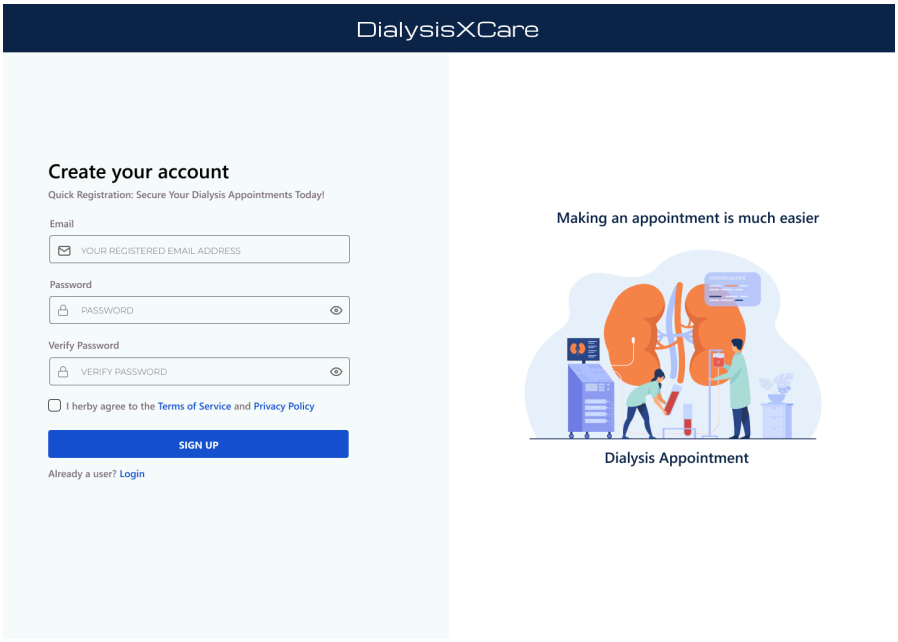


Figure 15: Create Account Page - 3

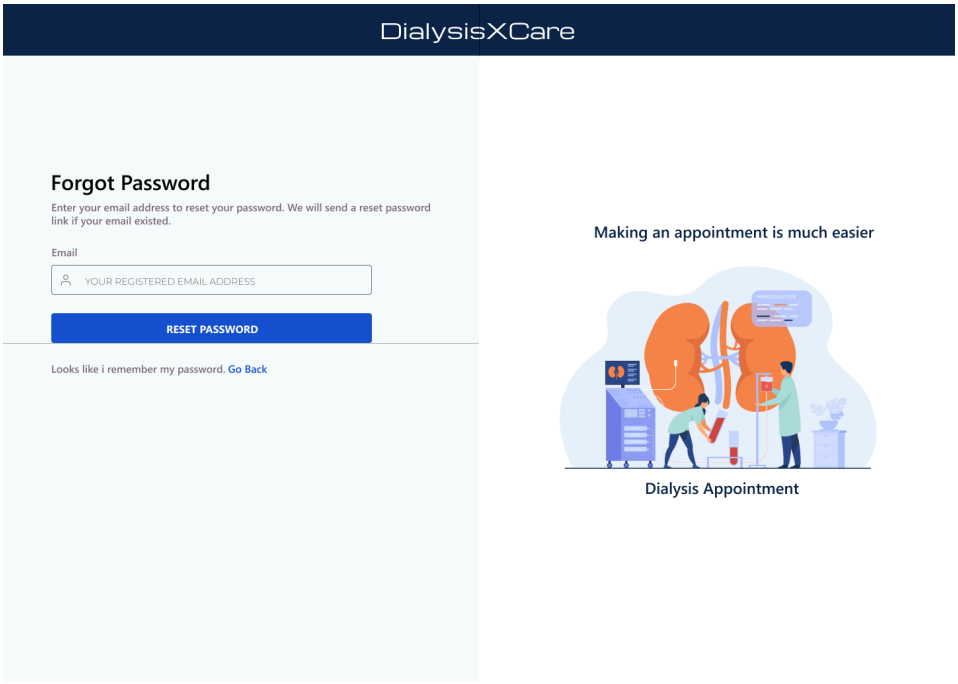


Figure 16: Forgot Password Page

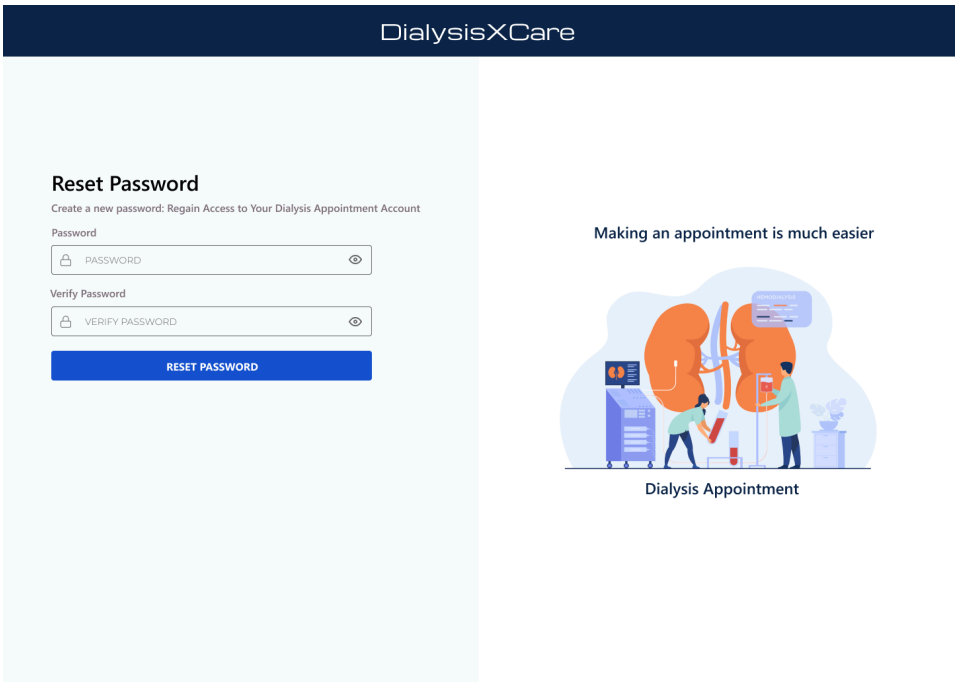


Figure 17: Reset Password Page

7.2 Admin View

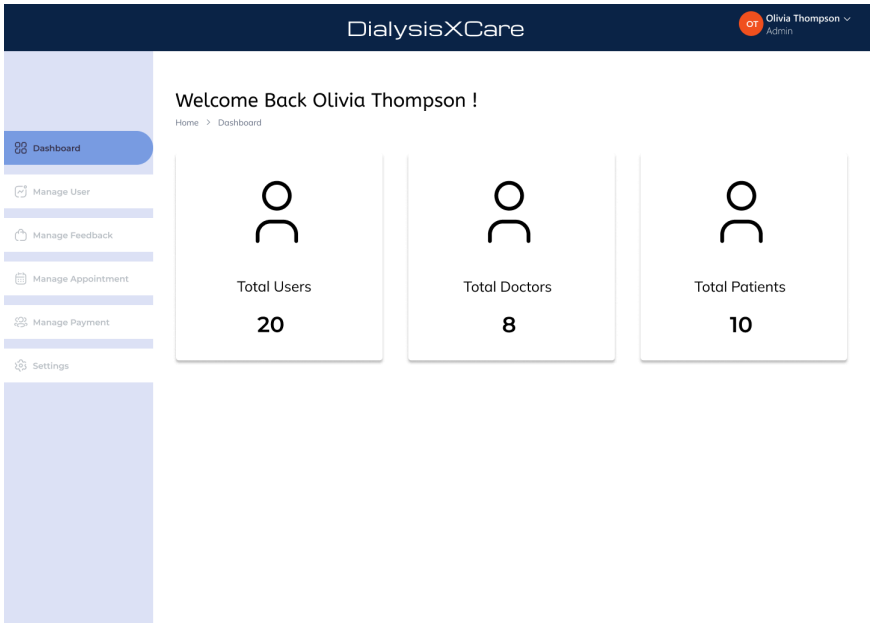


Figure 18: Dashboard Page

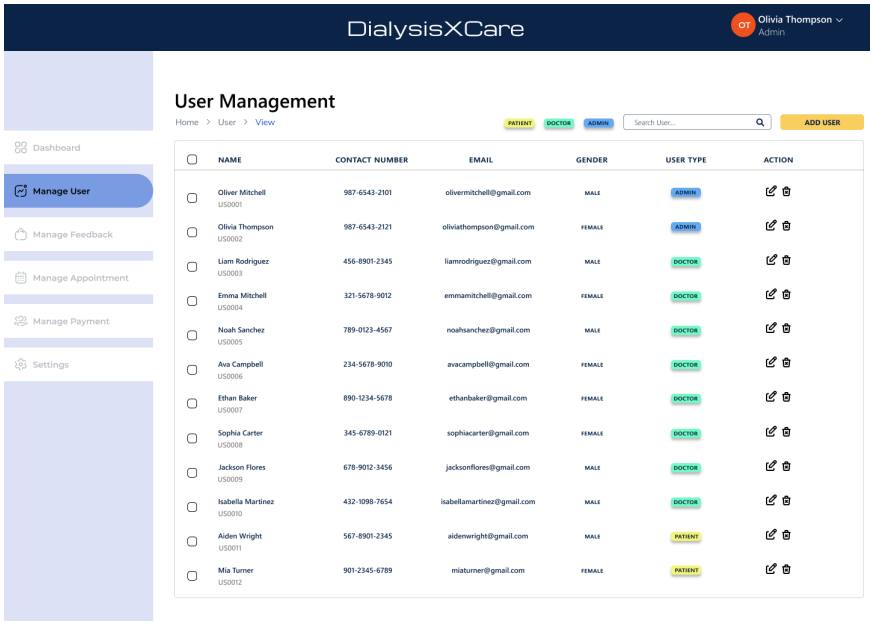


Figure 19: View User Page

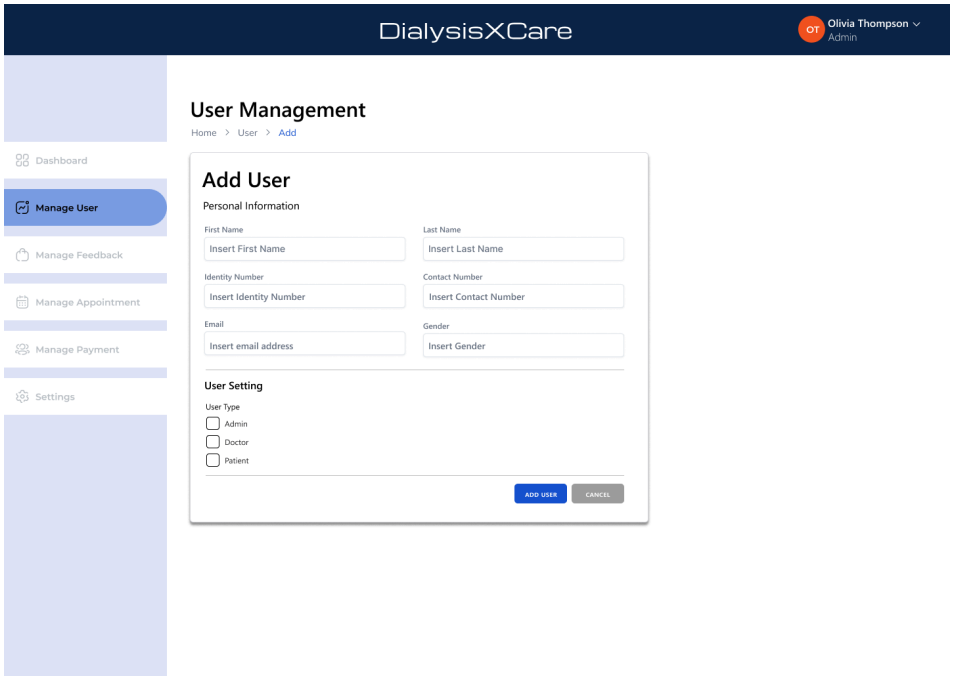


Figure 20: Add User Page

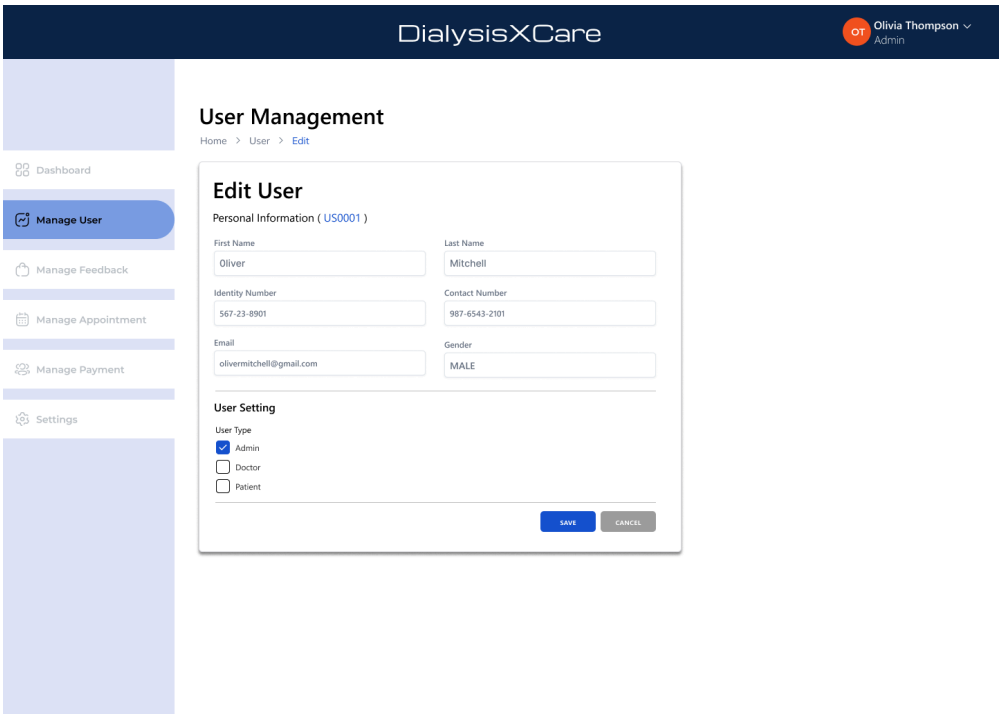


Figure 21: Edit User Page

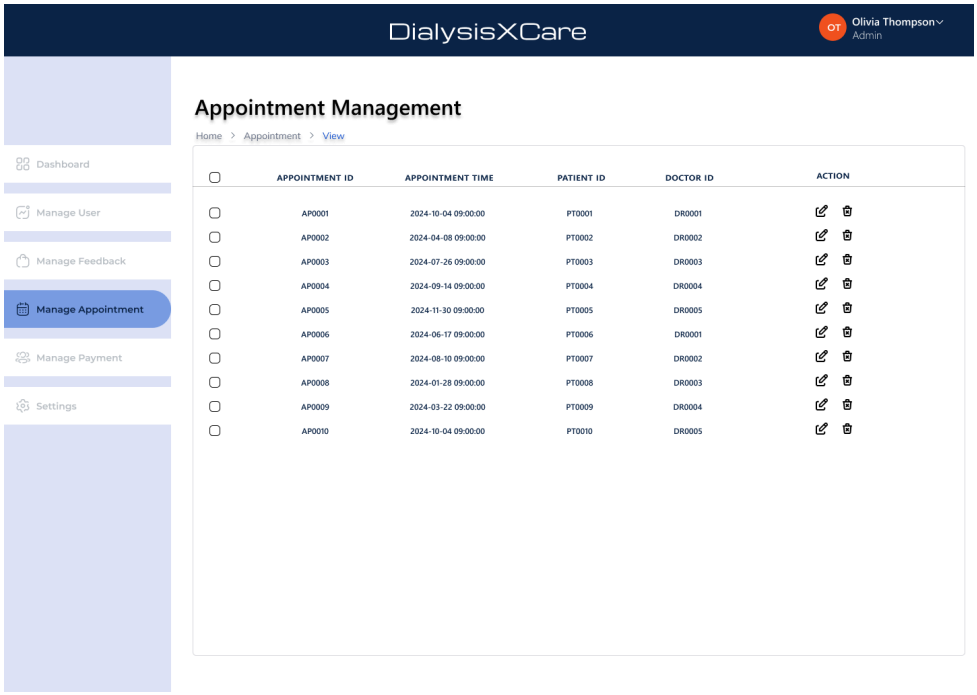


Figure 22: Appointment Page

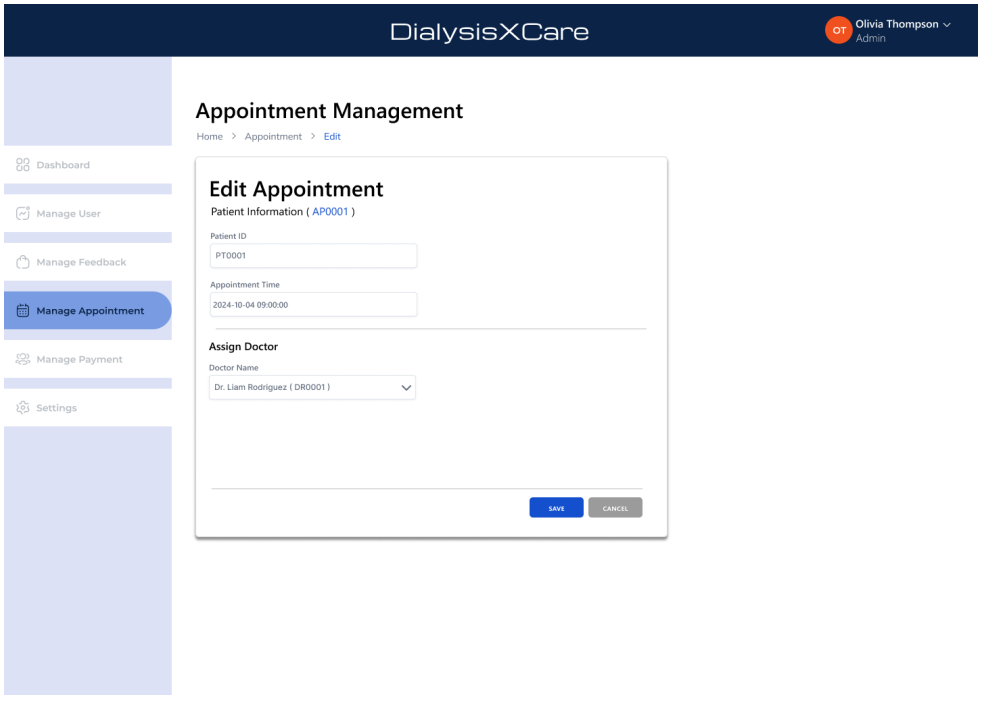


Figure 23: Assign Doctor Page

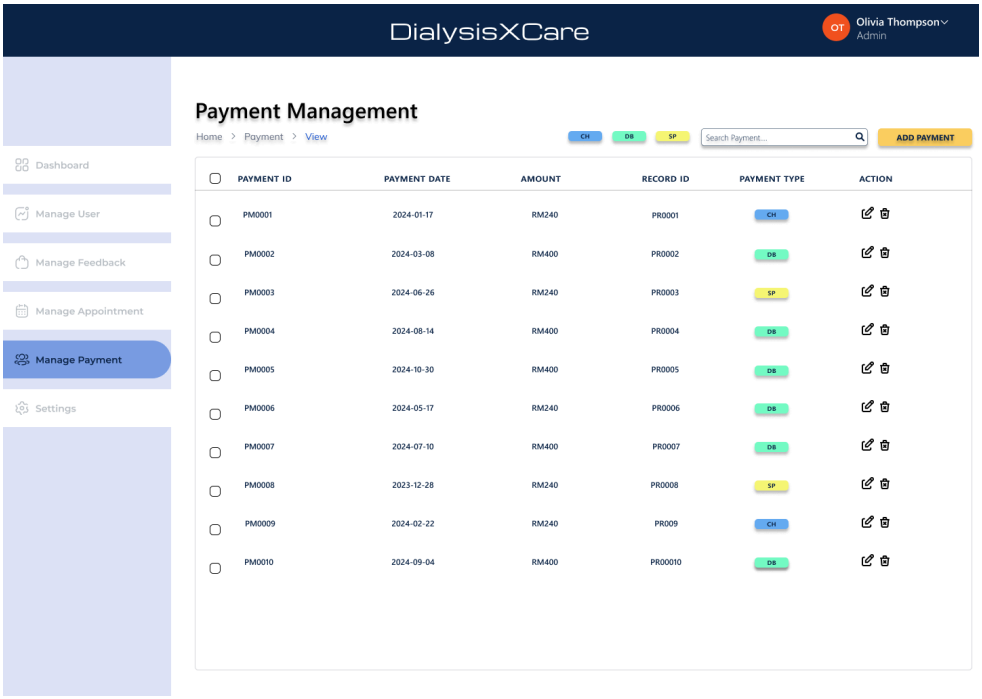


Figure 24: Payment Page

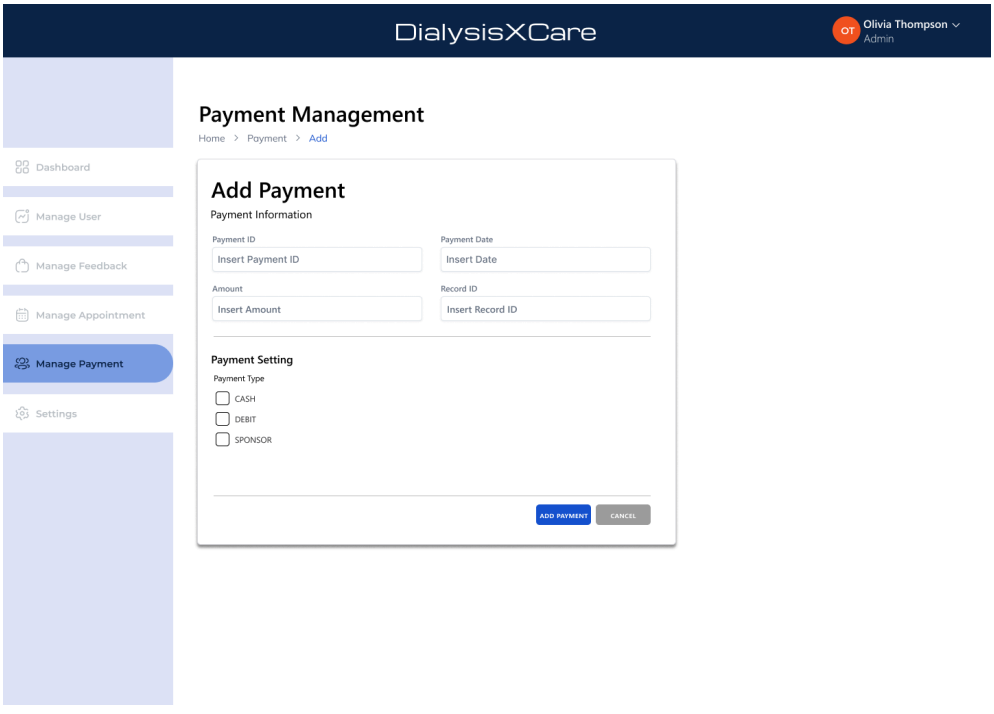


Figure 25: Add Payment Page

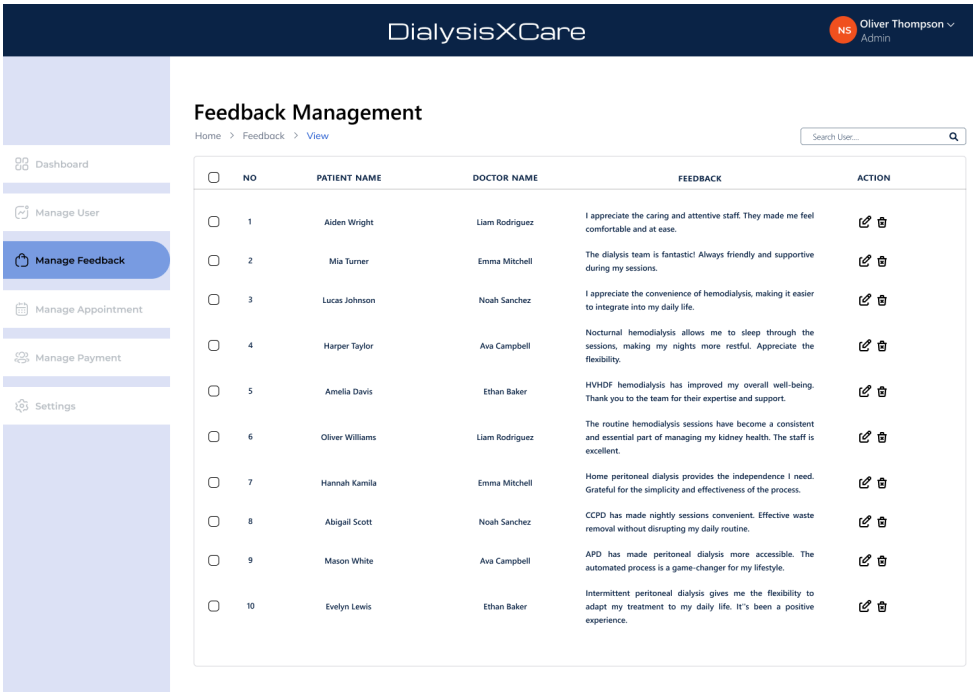


Figure 26: Manage Feedback Page

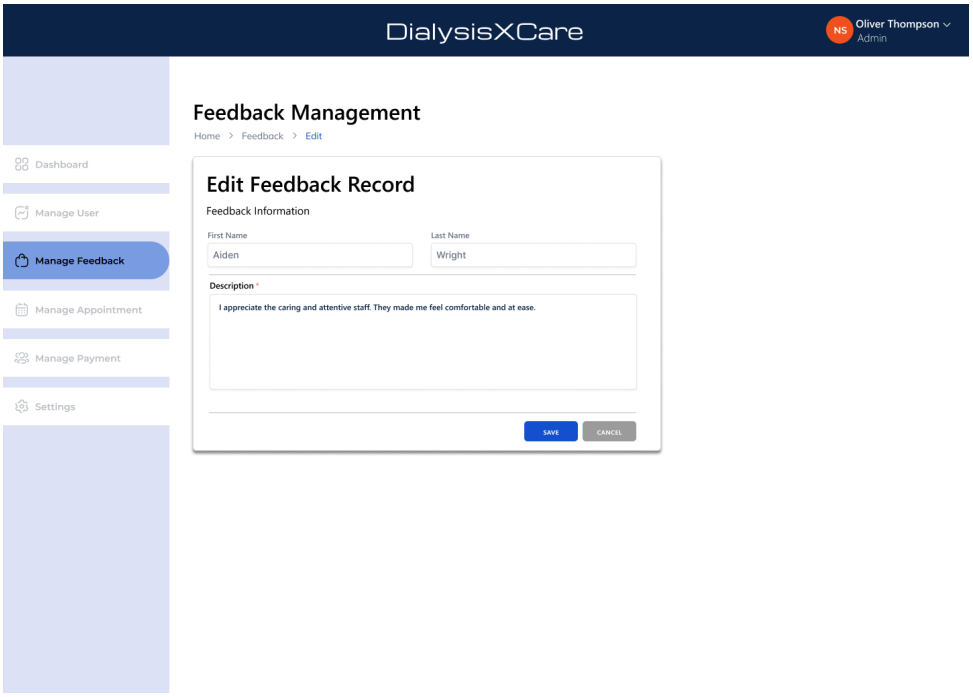


Figure 27: Edit Feedback Page

7.3 Doctor View

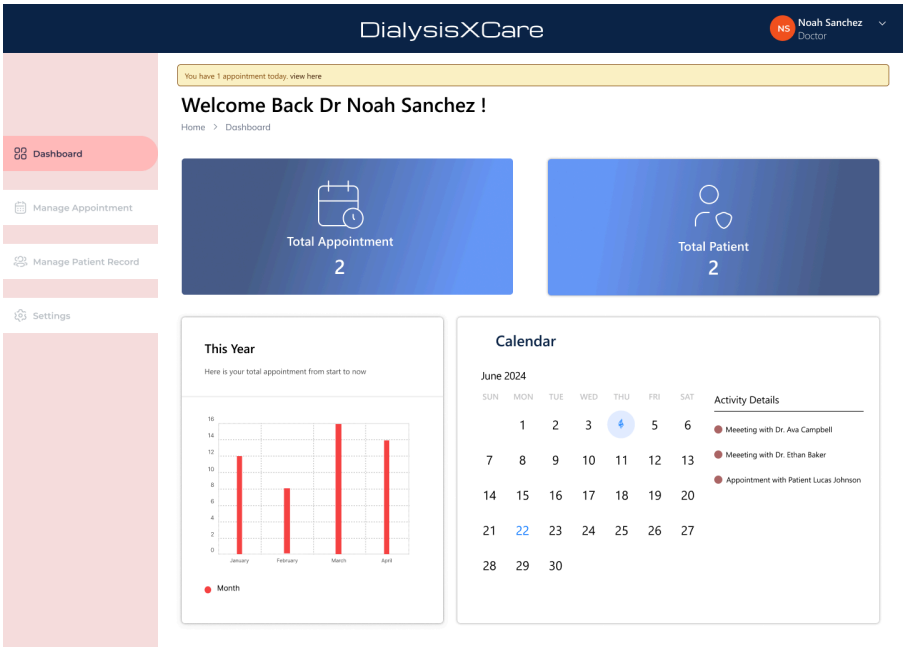


Figure 28: Dashboard Page

Figure 29: Patient Record Page

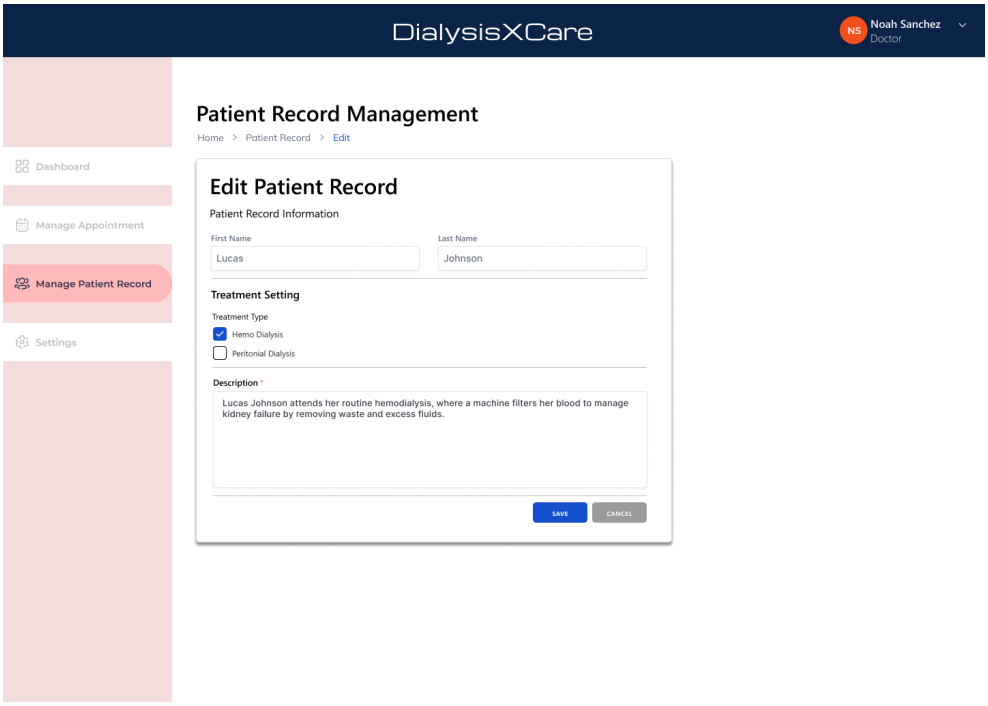


Figure 30: Edit Patient Record Page

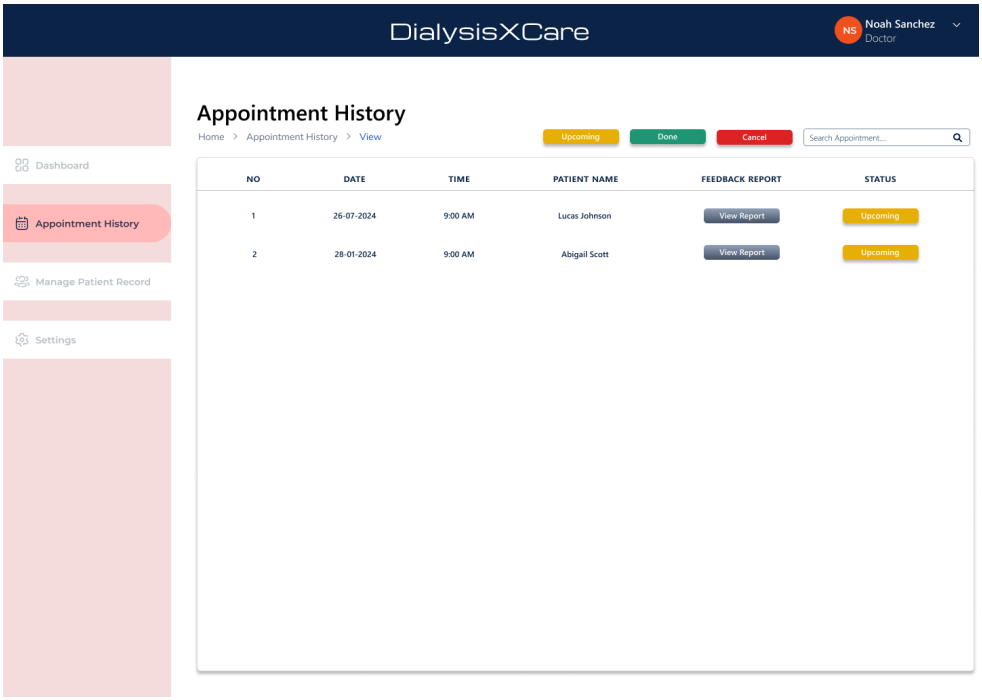


Figure 31: Appointment History Page

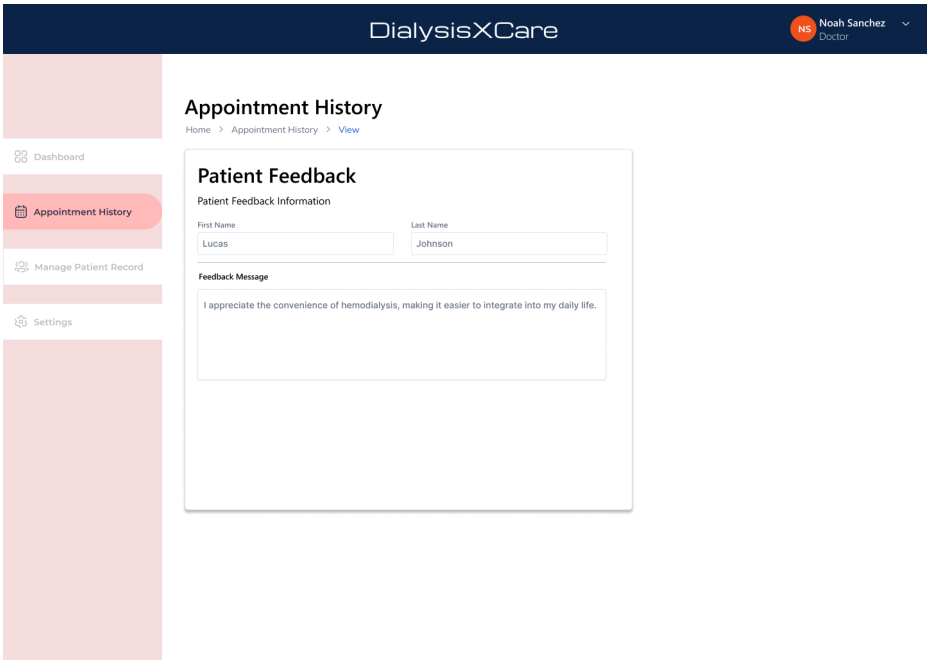


Figure 32: View Feedback Page

7.4 User View

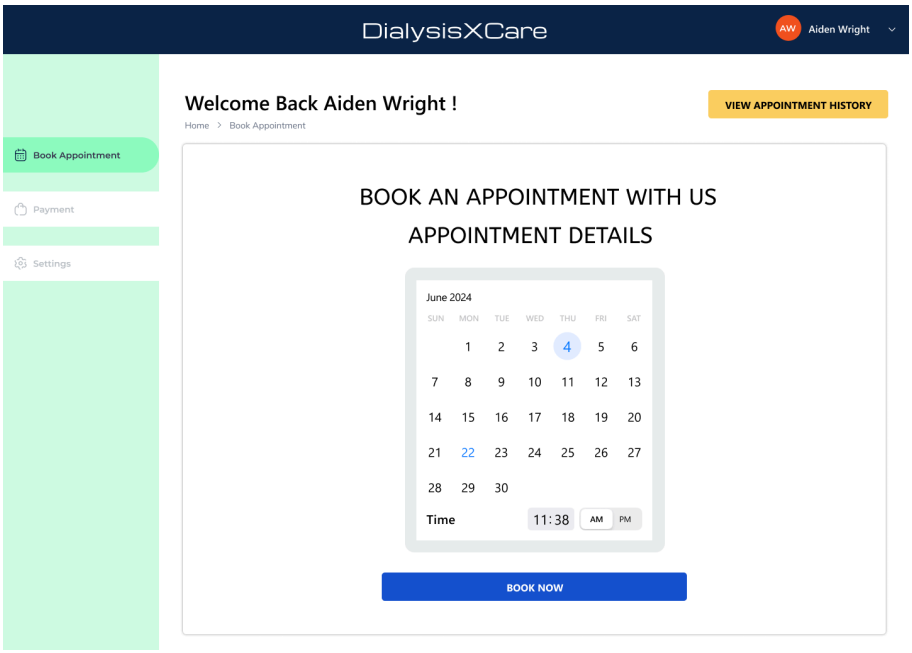


Figure 33: Appointment Page

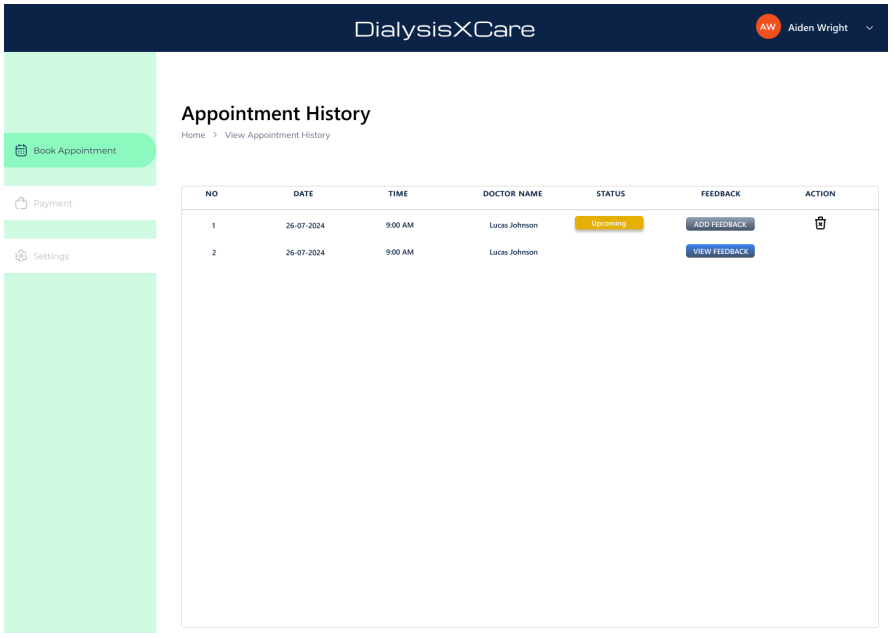


Figure 34: View Appointment History Page

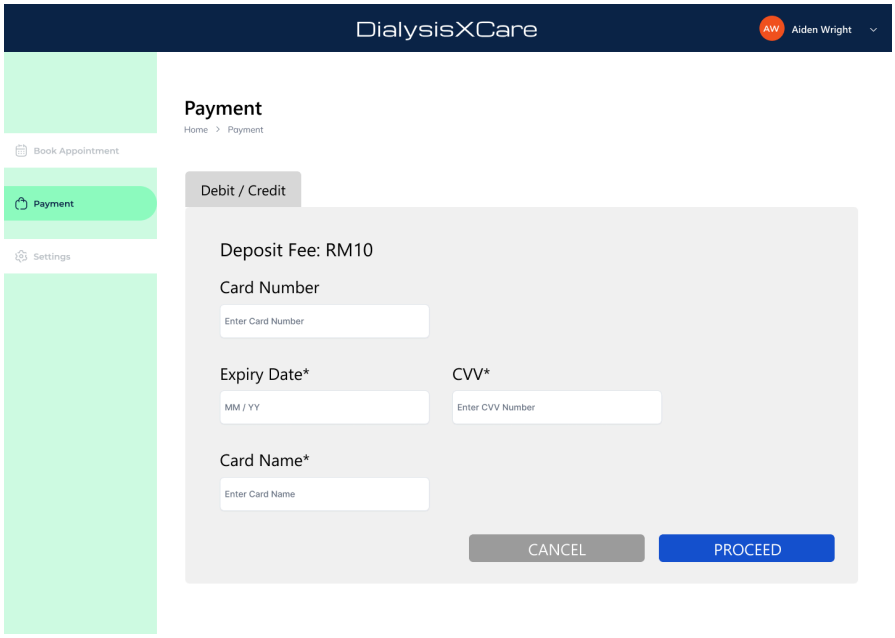


Figure 35: Payment Page

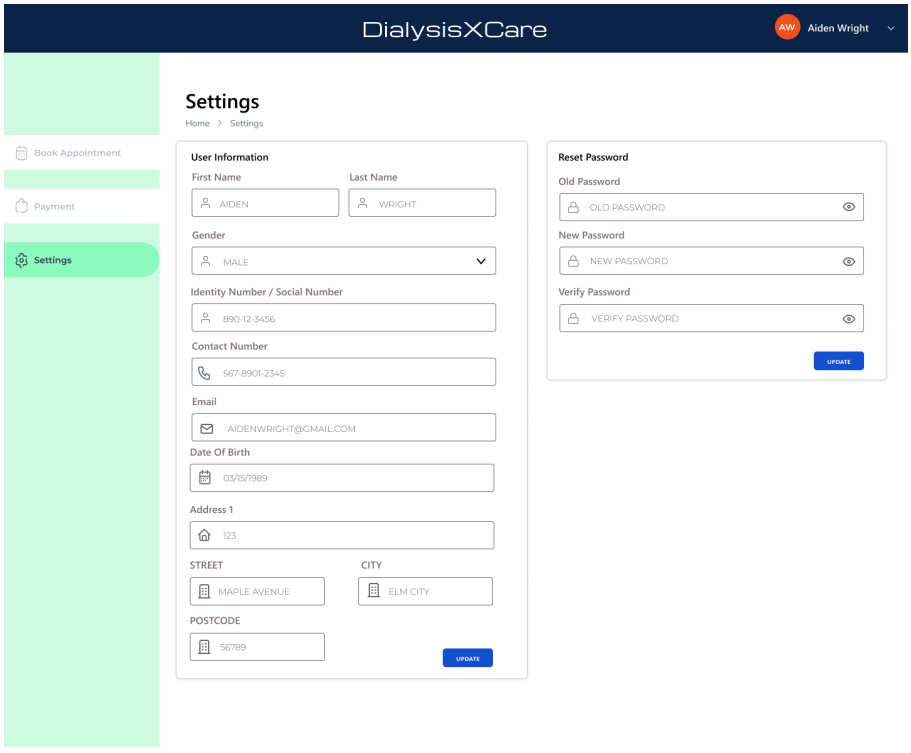


Figure 36: Patient Setting Page

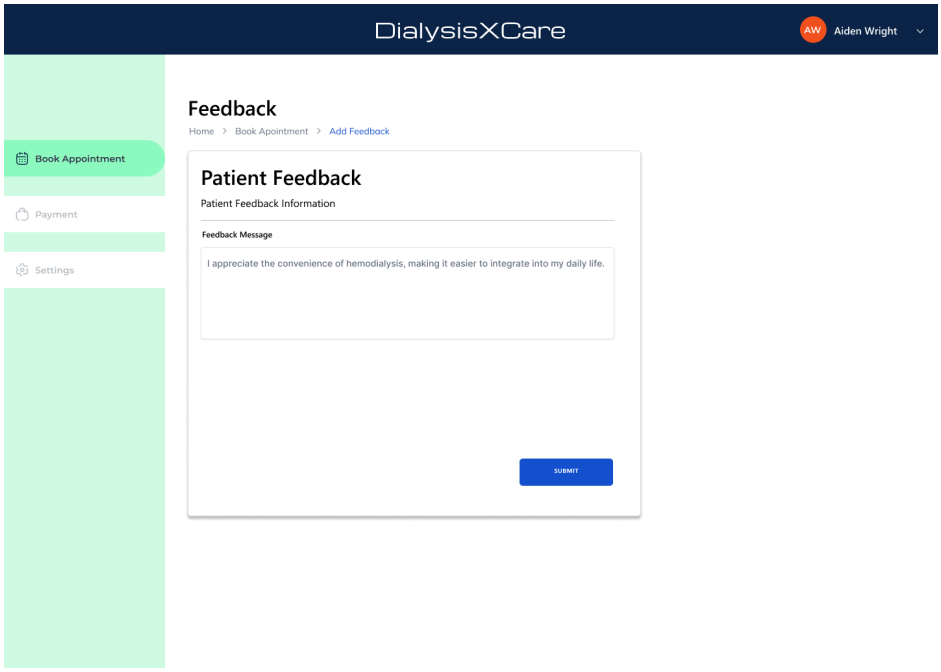


Figure 37: Add Feedback Page

Figure 38: Payment Succeed Page

8.0 Summary

Due to the use of several outdated manual processes, Pusat Dialisis FN has some operational inefficiencies. Among these are the challenges patients have while trying to arrange an appointment, which they face whether they go to the clinic in person, complete paperwork, or provide phone calls. Furthermore, the dialysis center personnel feels that it is difficult to manually organize, manage, and retrieve the current appointment data. The proposed project aims to improve and update Pusat Dialisis FN, a medical appointment and booking system, DialysisXCare by including a digital platform. Our preliminary studies exposed the widespread problems with manual procedures and underscored the importance of having a suitable system for making appointments and providing medical treatment. These problems, which include insufficient and poorly structured patient data as well as appointment scheduling, will be highlighted in particular by the study. The recommended solution involves replacing an outdated manual process with a modern automated one. It addresses the challenges of scheduling appointments with an online system that provides real-time information and automated reminders. The adoption of a more digitized appointment booking system will assist patients and staff by making it easier for them to plan and manage medical visits.

The DialysisXCare system heralds a groundbreaking era in the management and delivery of renal healthcare services. This report provides a comprehensive overview of the system's database conceptual design, showcasing the evolution from business rule refinement to the presentation of a preliminary conceptual ERD. Moving into the database's logical design, the report unfolds the Logical ERD, illustrating refined relationships and attributes based on logical considerations. The accompanying data dictionary provides detailed descriptions of entities, relationships, and attributes, contributing to a comprehensive understanding of the system's architecture. The normalization process further refines the database structure, ensuring optimal organization and efficiency. The report then introduces the Relational Database Schemas for DialysisXCare, demonstrating a structured and optimized organization of data. Additionally, SQL statements are presented, divided into Data Definition Language (DDL) and Data Manipulation Language (DML), outlining the implementation of the database.

In essence, the DialysisXCare system marks a transformative leap forward in renal healthcare delivery. The report concludes by summarizing the key benefits and outcomes associated with the system's implementation, emphasizing its potential to redefine the dialysis experience for healthcare providers and patients alike. The integration of advanced database design principles ensures not only efficiency but also sets the stage for an elevated standard of care in the realm of health.