UTM UNIVERSITI TEKNOLOGI MALAYSIA

FACULTY OF COMPUTING
UTM Johor Bahru

Project : Phase 2

<PKU PRO MAX>

SECD2523 - DATABASE

SEMESTER I, SESSION

2023/2024

Lecturer: Dr. Nur Hidayah binti Zakaria

Group Name: 404 Found

Name	Matric No.
Khoo Teong lee	A22EC0174
Lim Bo Yuan	A22EC0181
Lim Yu An	A22EC0183
Loh Chee Huan	A22EC0186

Section: 8

Table of Contents

1.0 Introduction

Phase 2 of this project will mainly focus on developing the logical and technical designs to optimize the operations at Pusat Kesihatan Universiti(PKU) at UTM. This phase will translate the proposed solutions in Phase 1 into more concrete system specifications and data structures through the following features.

First of all, we will map out essential data flows and processes in the new integrated PKU management system using data flow diagrams(DFD). The DFD will visualize the to-be process on how the entities interact with the enhanced system. Mapping out the DFD will help us to identify core data and transactional requirements easily.

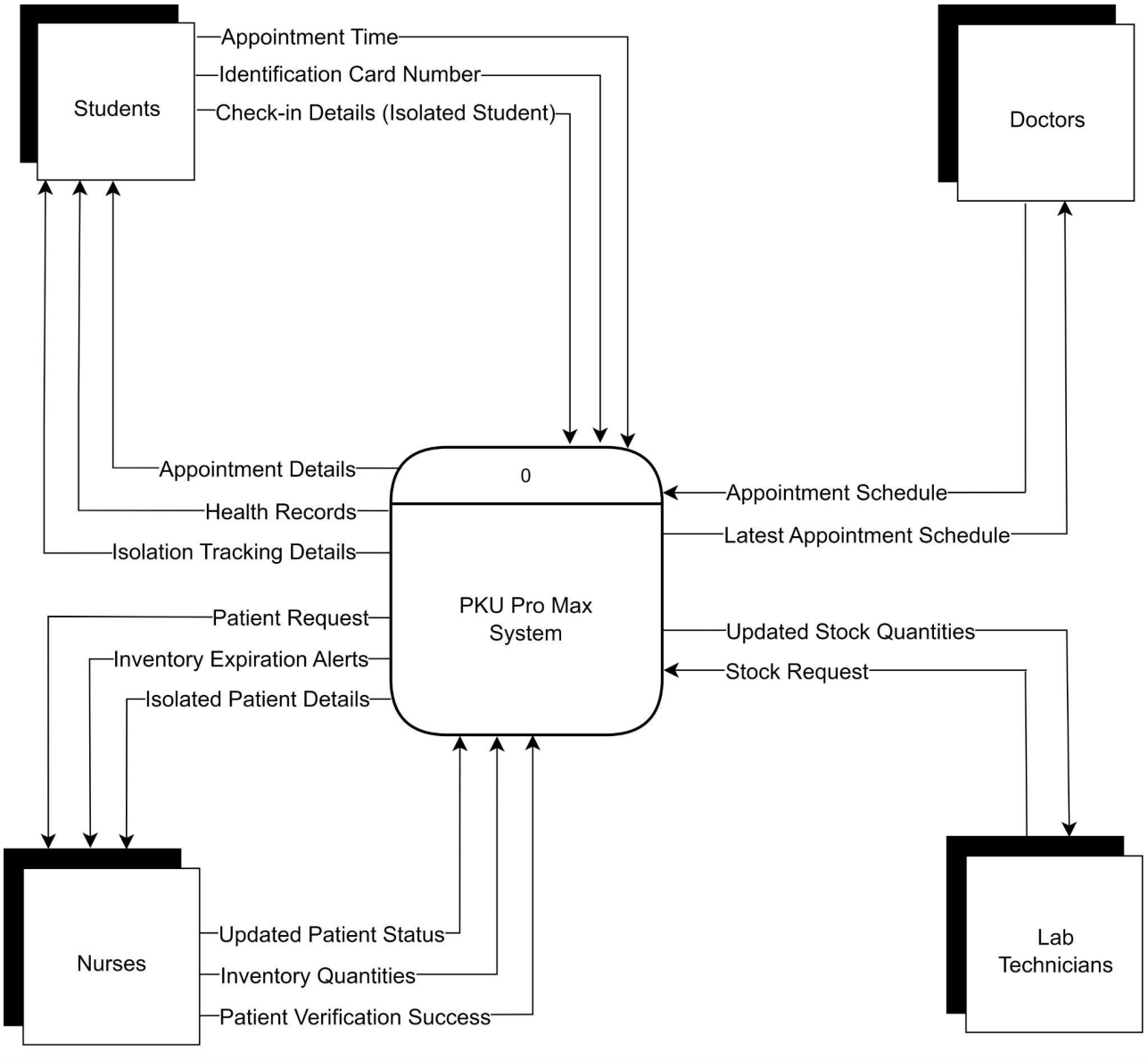
Next, we will define proposed business rules that set policies and constraints within the system. For instance, business rules could enforce permissions around medical record access. The rules aim to standardize the operations as well as data integrity.

After that, we will create a conceptual entity-relationship-diagram(ERD) followed by enhanced ERD to show how the data entities have a relationship with each other. This conceptual database design is crucial to provide a clear overview of the relationship between entities.

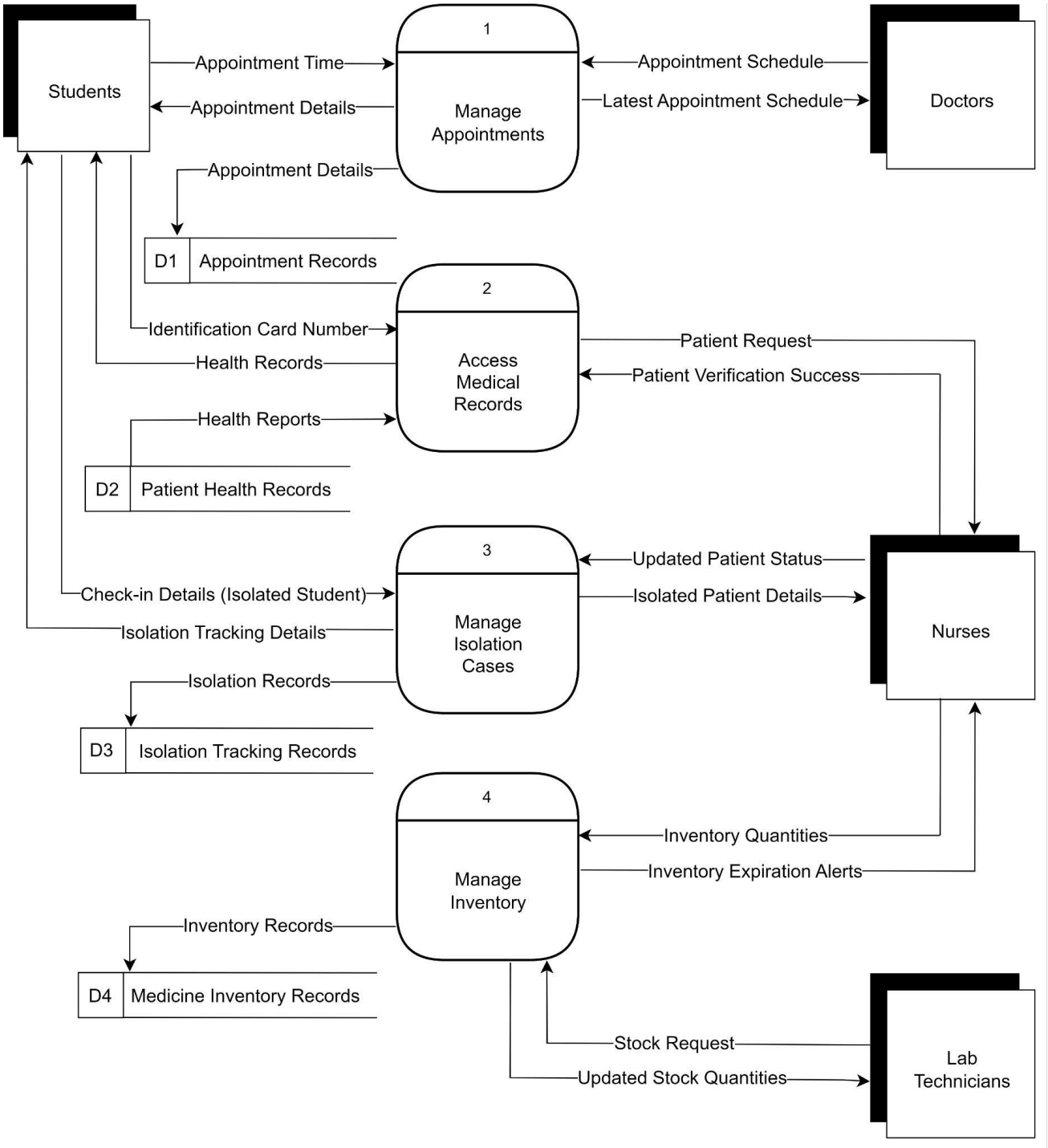
Finally, we will compile a data dictionary documenting all the data elements in our system. The data dictionary will give technical specifications and descriptive metadata for each entity, relationship and attribute. This will play an important role in developing the physical database and interfaces.

2.0 DFD (To-Be)

2.1 Context Diagram

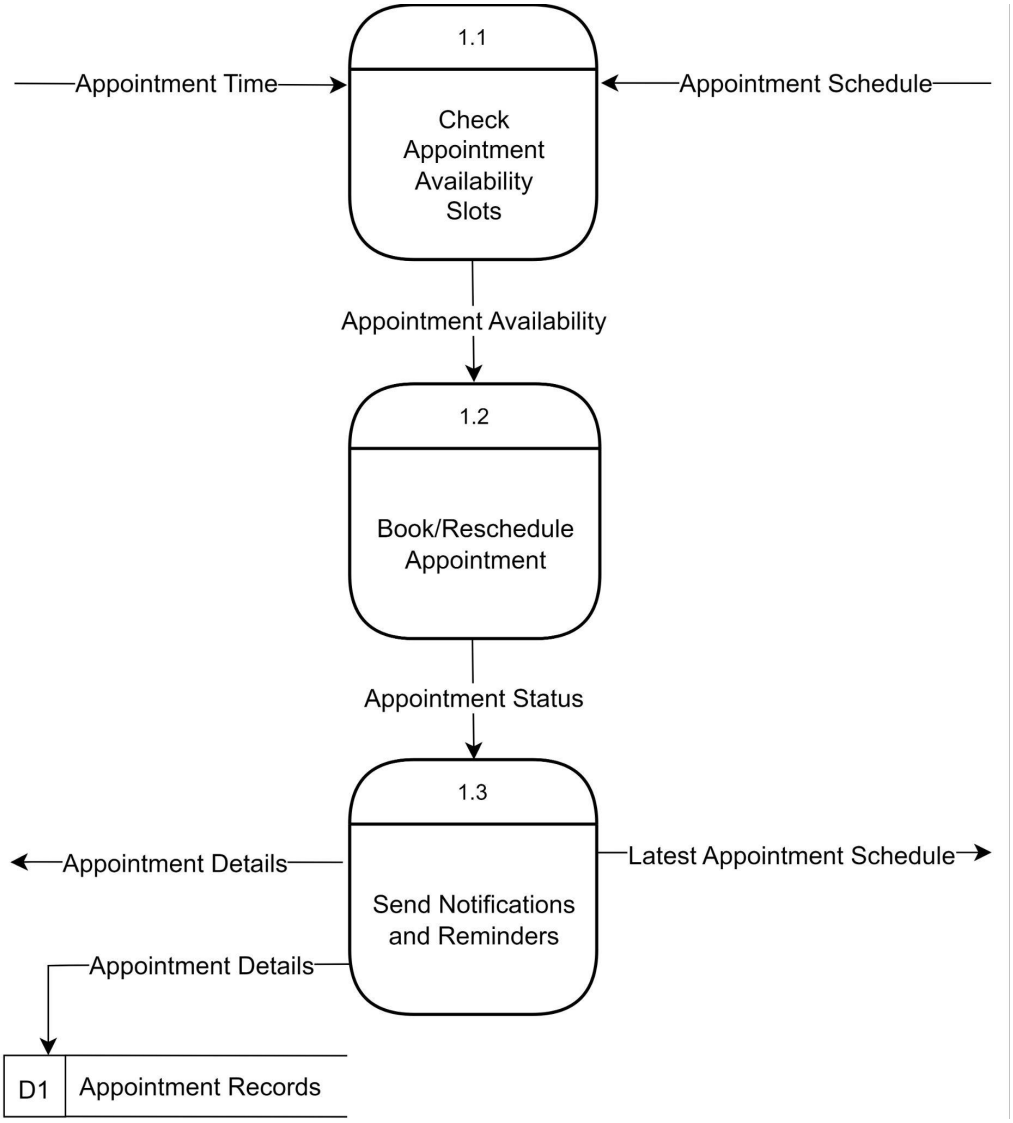


2.2 Level-0 Diagram

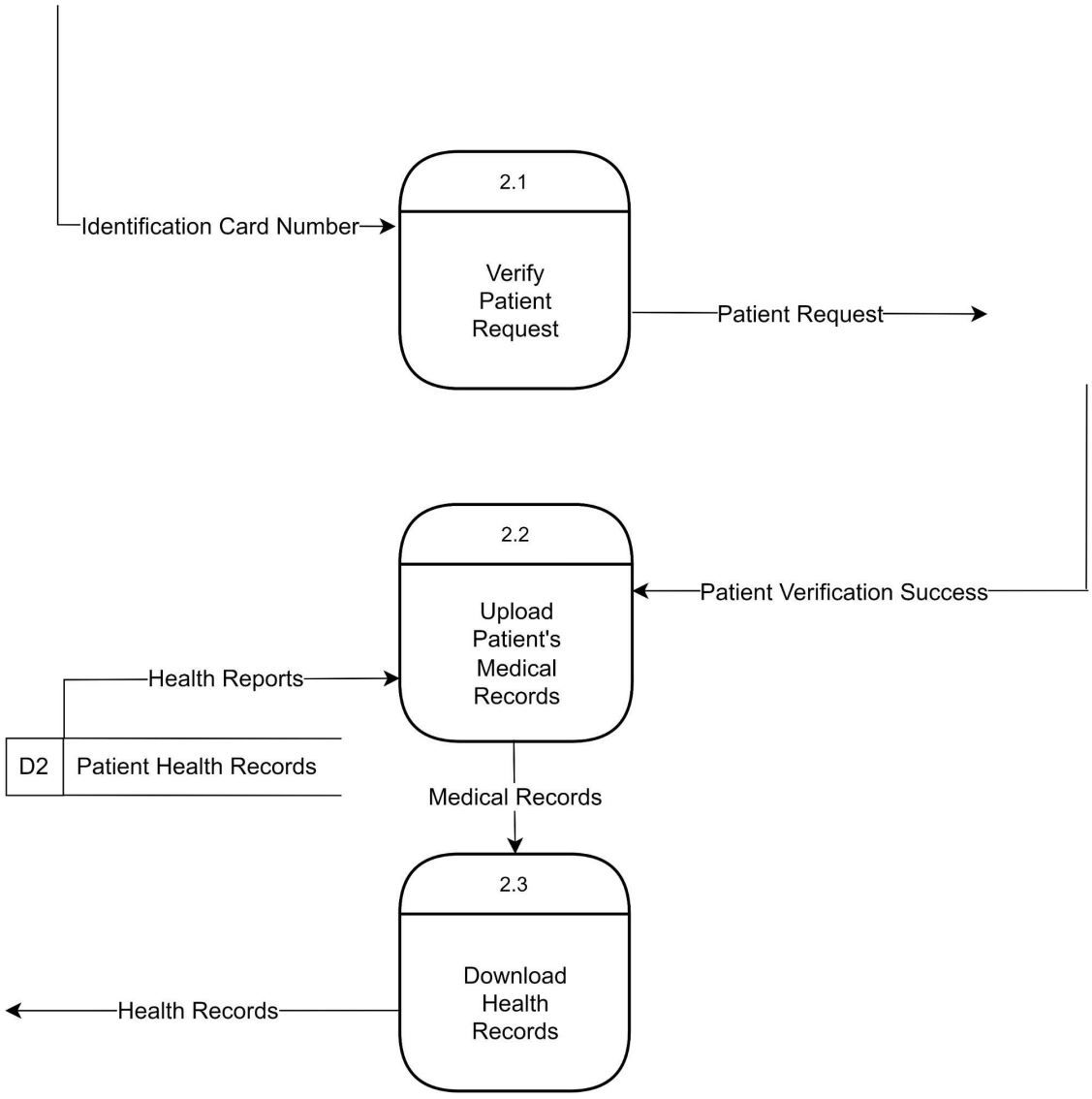


2.3 Level-1 Diagram

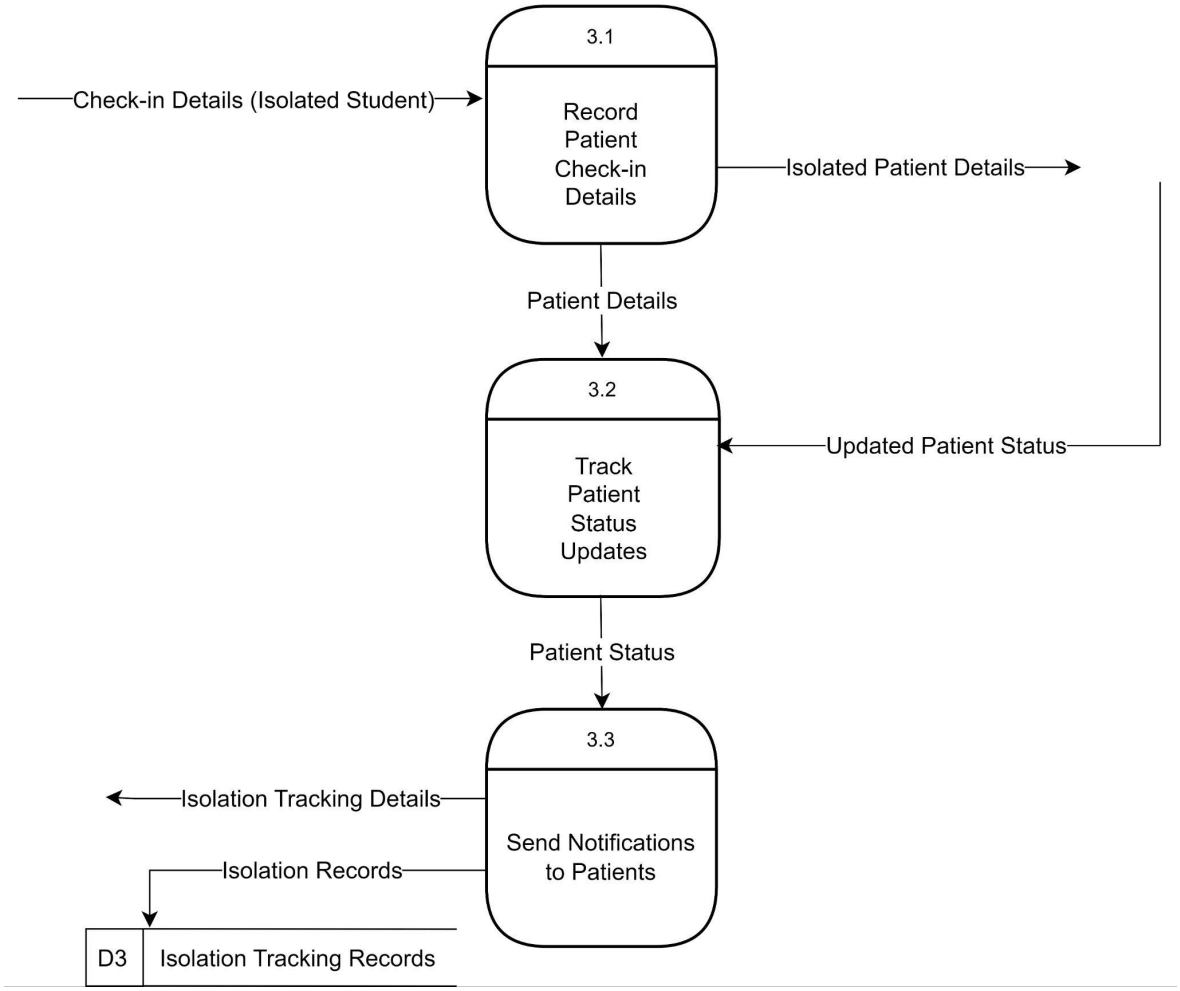
2.3.1 Process 1: Manage Appointments



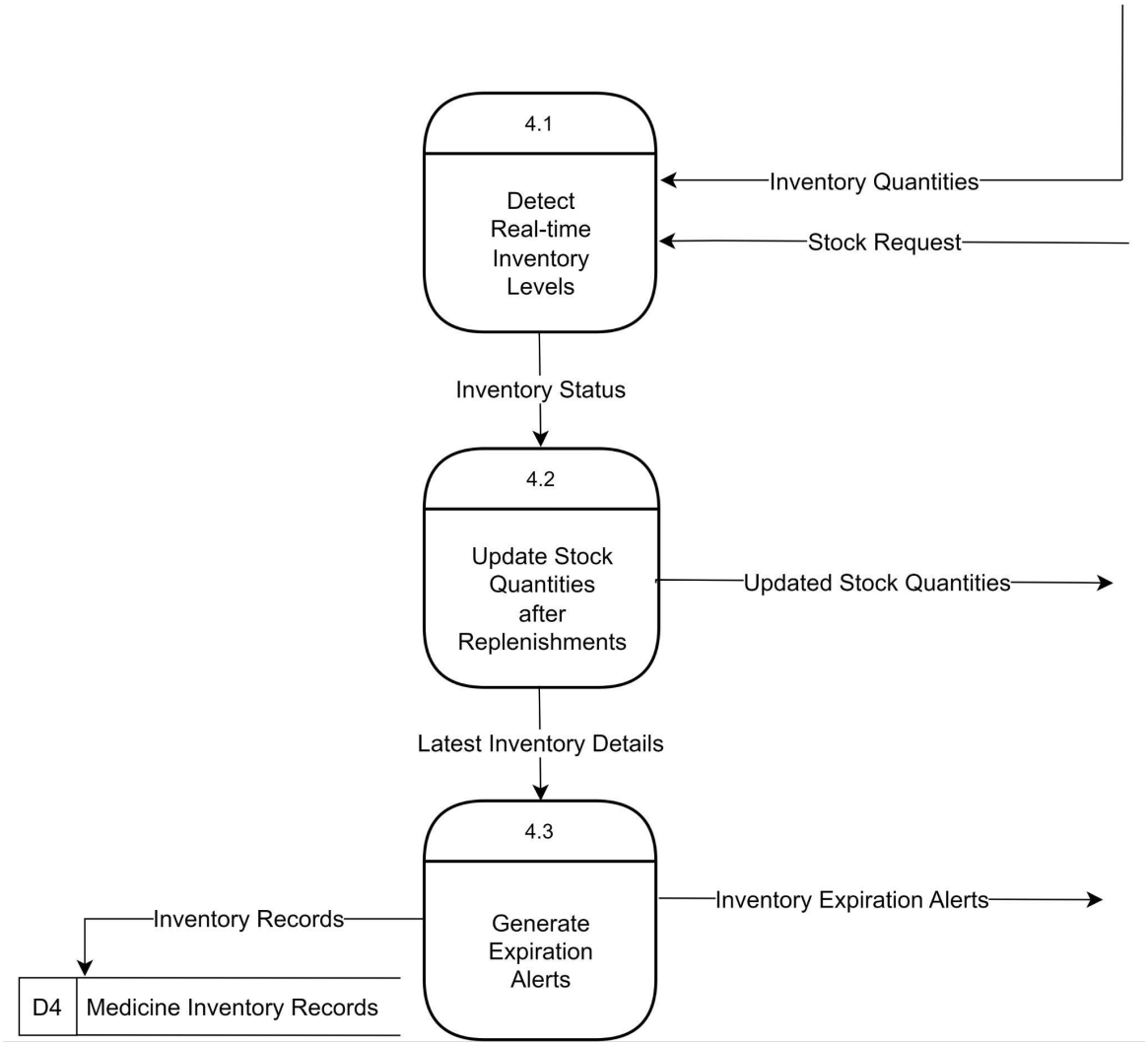
2.3.2 Process 2: Access Medical Records



2.3.3 Process 3: Manage Isolation Cases



2.3.4 Process 4: Manage Inventory



3.0 Data & Transaction Requirements

3.1 Proposed Business Rules

1. The system is working 20 hours for 7 days per week.
2. For appointment :
    a. Patients schedule their appointments online based on the availability slots.
    b. Patients fill in their details (name, date of birth, email, etc..) before booking an appointment.
    c. Patients select their data, doctor and available time slot while booking appointments.
    d. System sends confirmation notification to patients through Email after booking.
3. Only hospital staff (doctor, nurse and clerk) can access patients' medical reports.
4. Patient details like date, symptoms and diagnosis notes are recorded in the medical report of each patient.
5. The medical record can be updated once there are any changes made.
6. The medical details are filled in by staff if there is any issuance or replenishment on the medicine stock.
7. The system will automatically track the medicine stock level once the medicine details have been filled in to the system.
8. For payment, the fees calculation is automated based on the medical fees and medicine costs.
9. The system will accept payment through cash, credit and debit card, Duit Now and online banking.
10. A recipects with relevant medical details is provided once the payment was successful.

3.2 Proposed data & transactional

Proposed Transaction Requirement

Data entry :

- Enter patient booking details.
- Enter appointment booking request.
- Enter medical report information after diagnosis.
- Enter medicine prescribed by the doctor.
- Enter student check-in and check-out data for isolation.
- Enter inventory receipts for medicine.

Data update /delete :

- Update/delete patient information.
- Update/delete appointment booking status.
- Update/delete patient medical report.
- Update/delete medicine quantity after issuance or replenishment.
- Update/delete isolation status based on student check-in.

Data queries :

- List details of patients by name, ID etc.
- Identify appointment availability and slots.
- View medical history for a patient.
- Identify medicine stock availability.
- Display isolation detail and history for a student.
- Generate reports on appointments, patients and medicine inventory.

Proposed Data Requirement

Patient:

The data stored on Patient includes the patient ID, patient name, date of birth, address and email. The patient ID uniquely identifies the instance of the patient, hence it is the primary key. The patient name is a composite attribute, which consists of first name and last name.

Appointment:

The data required on Appointment includes appointment ID, patient ID, doctor ID, date, time, type and status. Since each appointment refers to its own patient and doctor, the patient ID and doctor ID are also a foreign keys, which references the Patient and Doctor entity. Appointment ID is the unique primary key for this entity. In our proposed system, each appointment can only be booked by one patient, and only assigned by one related doctor.

Medical Service:

The data stored on the Medical Service includes the service ID, service name, category and cost. The service ID is the primary key that uniquely identifies each instance of the medical service.

Medical Record:

The data stored on the Medical Record includes record ID, patient ID, recordDate, symptoms and diagnosis. The record ID serves as a primary key. Since the medical record is involved by every patient, thus the patient ID referenced from Patient is required to serve as a foreign key.

Doctor:

The data stored in Doctor includes doctor ID, doctor name, its contact numbers, specialization, and department. The doctor ID is the unique primary key. The doctor name is a composite attribute, consisting of first name and last name.

Medicine:

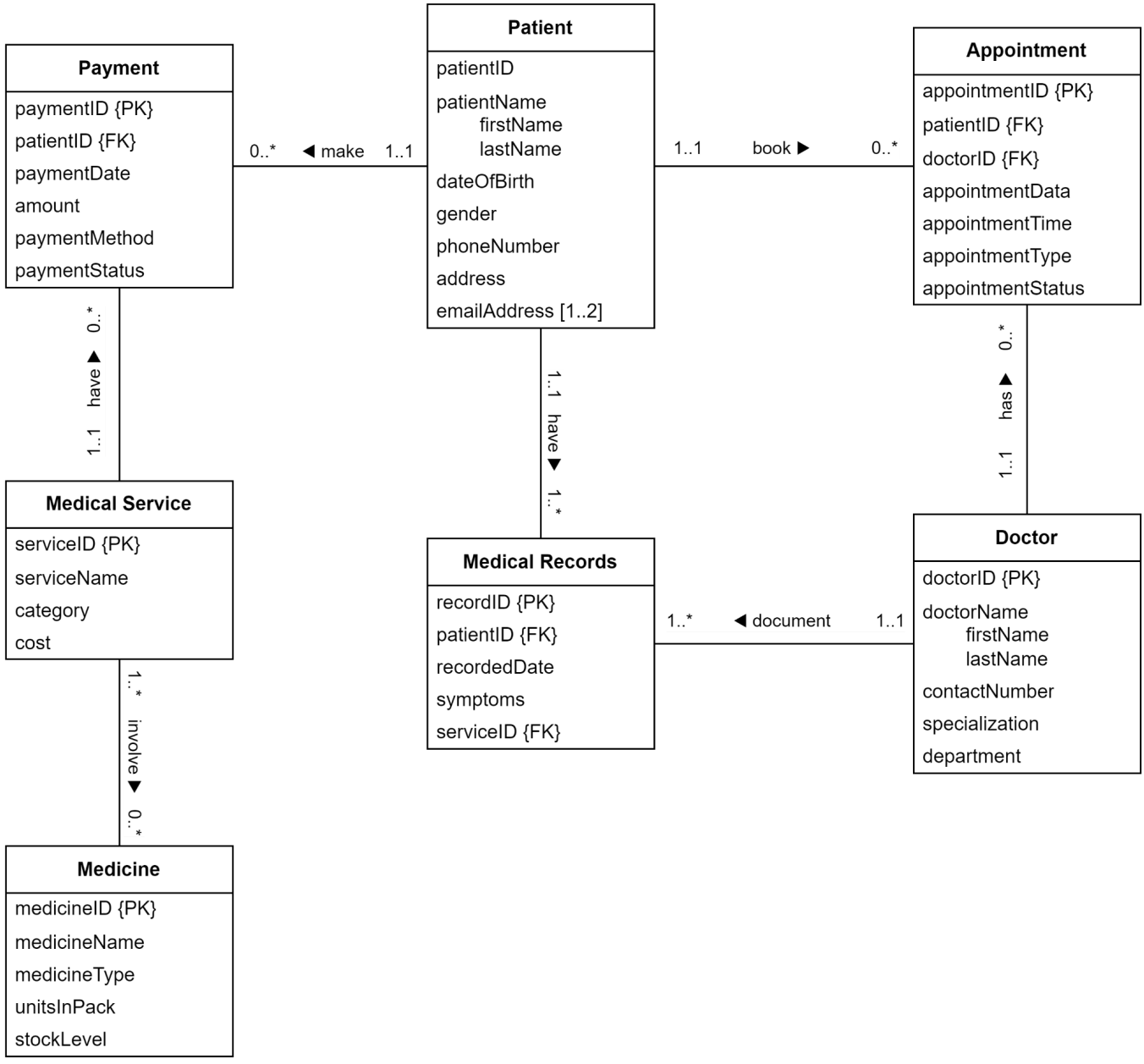
The data store in Medicine includes medicine ID, medicine name, medicine type, units in pack and stock level. There are only three types of medicine, which are capsule, syrup and tablet. Medicine ID serves as a primary to identify each unique medicine in the hospital.

Payment:

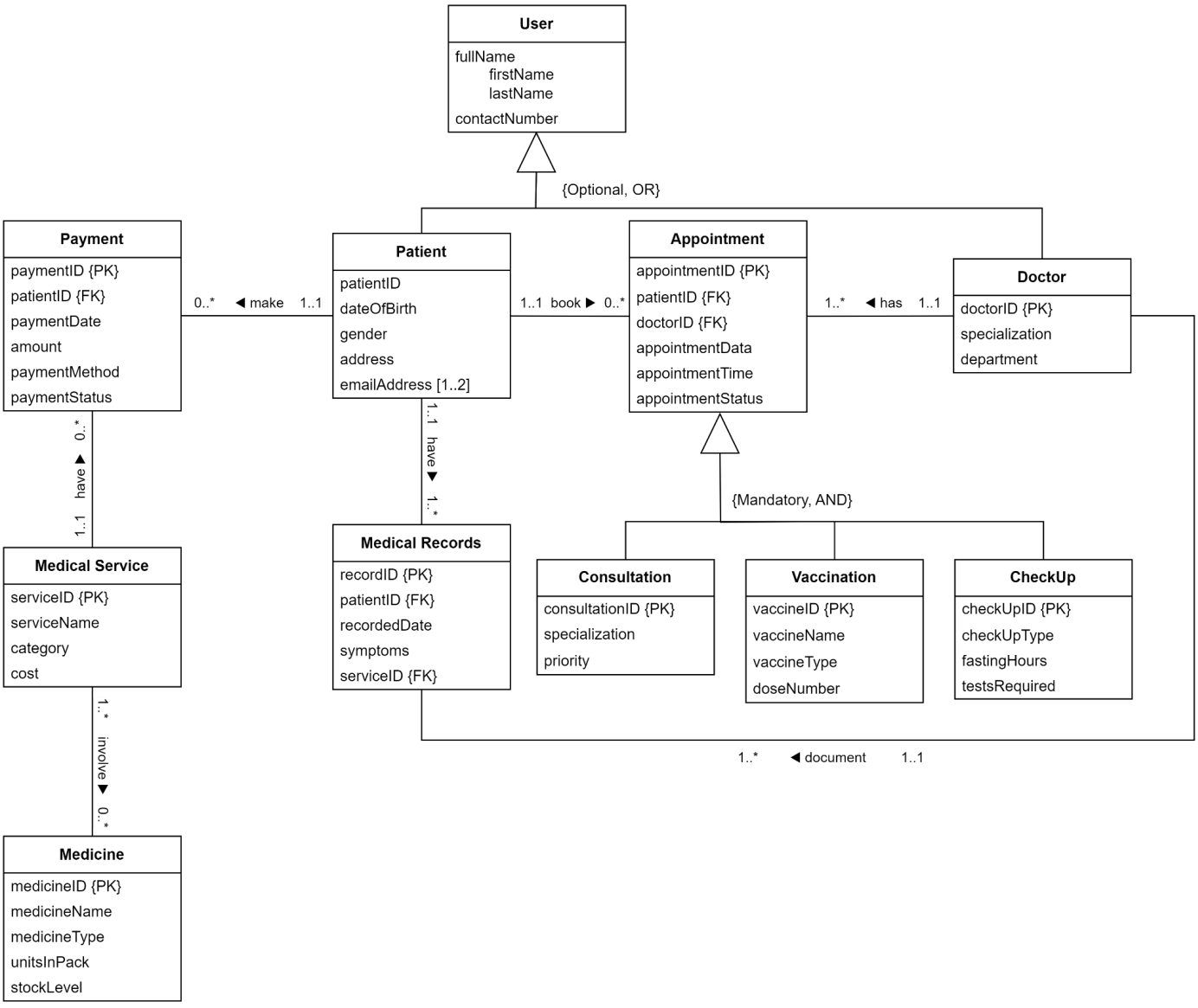
The data stored in Payment includes payment ID, date, amount, payment method, payment status and patient ID. Since each payment refers to a patient who visits the hospital, the patient ID from Patient entity is required to identify the patient who makes payment. The payment ID serves as a primary key, to identify every particular payment made by patients.

4.0 Database Conceptual Design

4.1 Conceptual ERD



4.2 Enhanced ERD (EERD)



5.0 Data Dictionary

Relation : User

Attribute	Data type	Data length	Constraint	Description
userID	NUMBER	10	PRIMARY KEY	User id, auto generated
firstName	VARCHAR2	20	NOT NULL	First name of the user
lastName	VARCHAR2	20	NOT NULL	Last name of the user
contactNumber	NUMBER	11	NOT NULL	Contact number of the user

Relation : Patient

Attribute	Data type	Data length	Constraint	Description
patientID	NUMBER	10	PRIMARY KEY	Patient id, auto generated
dateOfBirth	DATE	10	NOT NULL	Birth date of the patient
gender	VARCHAR2	10	NOT NULL	Gender of the patient
address	VARCHAR2	60	NOT NULL	Address of the patient
emailAddress	VARCHAR2	50	NOT NUKK	Email address of the patient

Relation : Doctor

Attribute	Data type	Data length	Constraint	Description
doctorID	INT	10	PRIMARY KEY	Doctor id, auto generated
specialization	VARCHAR2	50	NOT NULL	Specialization of the doctor such as critical care medicine specialists, emergency medicine specialists, family physicians
department	VARCHAR2	30	NOT NULL	Department that the doctor belongs to

Relation : Appointment

Attribute	Data type	Data length	Constraint	Description
appointmentID	NUMBER	10	PRIMARY KEY	Appointment id, auto generated
patientID	NUMBER	10	FOREIGN KEY	Patient id, auto generated
doctorID	NUMBER	10	FOREIGN KEY	Doctor id, auto generated
appointmentDate	DATE	10	NOT NULL	Appointment date
appointmentTime	TIMESTAMP	10	NOT NULL	Appointment time slot
appointmentStatus	VARCHAR2	20	NOT NULL	Status of appointment booking such as successful or fail

Relation : Consultation

Attribute	Data type	Data length	Constraint	Description
consultationID	NUMBER	10	PRIMARY KEY	Consultation id, auto generated
specialization	VARCHAR2	50	NOT NULL	Specialization of the consultation such as general medicine, cardiology, dermatology or others
priority	VARCHAR2	25	NOT NULL	Priority of the consultation such as normal or urgent

Relation : Vaccination

Attribute	Data type	Data length	Constraint	Description
vaccineID	NUMBER	10	PRIMARY KEY	Vaccine id, auto generated
vaccineName	VARCHAR2	50	NOT NULL	Vaccine name
vaccineType	VARCHAR2	25	NOT NULL	Type of vaccine for such as Influenza, Hepatitis B and others
doseNumber	VARCHAR2	10	NOT NULL	Number of dose taken

Relation : CheckUp

Attribute	Data type	Data length	Constraint	Description
checkUpID	NUMBER	10	PRIMARY KEY	Medical check up id, auto generated
checkUpType	VARCHAR2	50	NOT NULL	Type of check up such as full body, diabetes or others
fastingHours	VARCHAR2	15	NOT NULL	Recommended fasting time before check up
testRequired	VARCHAR2	10	NOT NULL	Tests included for the check up such as blood test, X-ray

Relation : Medicine

Attribute	Data type	Data length	Constraint	Description
medicineID	NUMBER	10	PRIMARY KEY	Medicine id, auto generated
medicineName	VARCHAR2	50	NOT NULL	Name of the medicine
medicineType	VARCHAR2	15	NOT NULL	Type of the medicine such as liquid, tablet or capsules
unitsInPack	NUMBER	5	NOT NULL	Number of medicine packed in the each dispensing bags
stockLevel	NUMBER	10	NOT NULL	The remaining quantity of the medicine

Relation : Medical Service

Attribute	Data type	Data length	Constraint	Description
serviceID	NUMBER	10	PRIMARY KEY	Medical service id, auto generated
serviceName	VARCHAR2	50	NOT NULL	Medical service given to the patient
category	VARCHAR2	15	NOT NULL	Category of the medical service such as diagnosis or treatment
cost	VARCHAR2	10	NOT NULL	Cost for the medical service

Relation : Medical Record

Attribute	Data type	Data length	Constraint	Description
recordID	NUMBER	10	PRIMARY KEY	Record id, auto generated
patientID	NUMBER	10	FOREIGN KEY	Patient id, auto generated
recordedDate	DATE	10	NOT NULL	Date for that specific medical service
symptoms	VARCHAR2	80	NOT NULL	Symptoms for the patient
serviceID	NUMBER	10	FOREIGN KEY	Medical service id, auto generated

Relation : Payment

Attribute	Data type	Data length	Constraint	Description
paymentID	NUMBER	10	PRIMARY KEY	Record id, auto generated
patientID	NUMBER	10	FOREIGN KEY	Patient id, auto generated
paymentDate	DATE	10	NOT NULL	Date of making payment
amount	VARCHAR2	15	NOT NULL	Amount of payment need to paid
paymentMethod	VARCHAR2	20	NOT NULL	Method used to make payment
paymentStatus	VARCHAR2	20	NOT NULL	Status of the payment such as successful or fail

6.0 Summary

Our project's expectation for the improvement of the current PKU system can be achieved through step-by-step procedures that we implement in Phase 2. First of all, mapping data flows and processes through data flow diagrams(DFD) will help us to understand more on how various data entities relate to one another. When drawing the DFD, we have to identify the main entities that might exist in the system so we will not miss out any of them. DFD are separated into three different types of diagrams which are Context Diagram, Level 0 Diagram and Level 1 Diagram. Each of the diagrams will provide a clear overview of our system.

Moreover, we list out the proposed business rules to help us to identify the limitations of the system so the system can work properly when it is launched. We also list out the proposed transaction requirement so we can know what operations are used to perform when interacting with the system. Finally, we write down all the proposed data requirement so we can identify the necessary attributes for each of the entities in our system.

Furthermore, we had developed ERD and EERD to show a clear relationship between the entities such as the multiplicity constraint and the name of the relationship. This can help us in developing a physical database easily later. Finally, we had construct a data dictionary to catalog the content of the data, providing meaningful descriptions for individually named attributes.