Web application using ‘Flask’

FOR THE COURSE “**DATABASE**”

Submitted By:

Eman Ashour

Arwa Wagdan

Mohamed Hasabalah

Besheer Ibrahim

Belal El-taiby

2022 - 2021

Guided BY:

DR.Mostafa Alzantout

Introduction

Our project is a web application using Flask framework starting with creating database schema for a health insurance company which offers three types of plans (Basic, Premium, and Golden). Each one covers a subset of enrolled hospitals. A hospital could be available under more than one plan.

Data base:

Our schema consists of some tables have relationships with each other.

Tables are ‘customers’, ‘dependents’, ’hospitals’, ‘claims’, ‘plans’,

‘Purchased plans’ and Table ‘enrolled’ which is created by the many_to_many

relationship between tables ’hospitals’ and ‘plans’.

Schema:

Table: customers

Columns:

Customer_Id	int AI PK
Customer_Name	varchar(45)
DateOfBirth	date
City	varchar(45)
Street	varchar(45)
Building_Number	varchar(45)
PhoneNumber	varchar(45)
Beneficiary_Plan	int

Table: dependants

Columns:

Dep_ID	int AI PK
Name	varchar(45)
DateOfBirth	date
RelationShip	varchar(45)
Beneficiary_plan	int
Customer_Id	int

Table: plans

Columns:

Plan_Id	int AI PK
Type	varchar(45)
Price	int

Table: enrolled

Columns:

Hospital_id	int PK
Plan_Id	int PK

Table: hospitals

Columns:

Hospital_id	int AI PK
Name	varchar(45)
City	varchar(45)
Street	varchar(45)
Phone	varchar(45)

Table: purchasd plans

Columns:

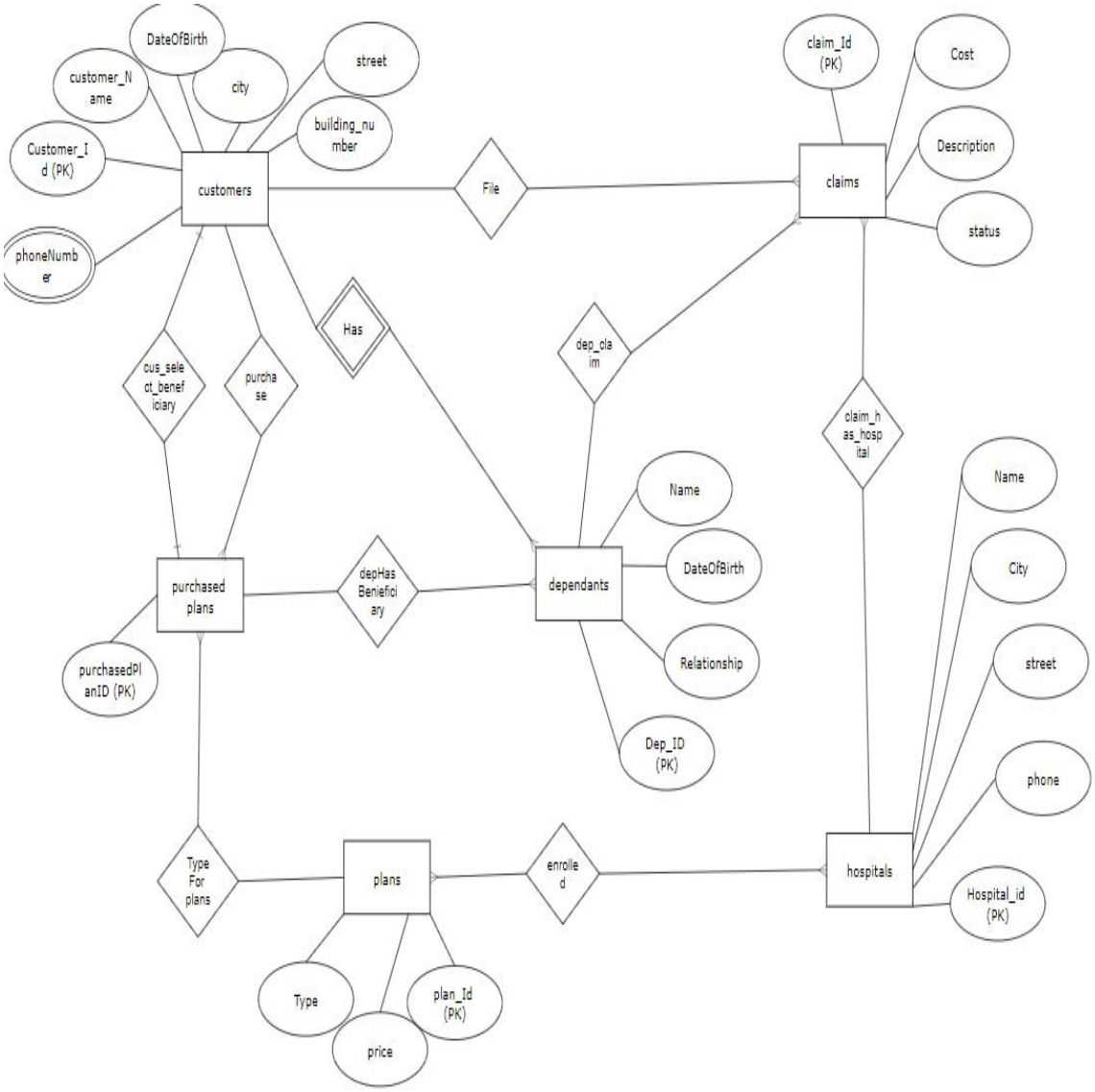
Customer_Id	int
Plan_Id	int
PurchasedPlanID	int AI PK

Table: claims

Columns:

claims_Id	int AI PK
Cost	int
Description	varchar(200)
Hospital_id	int
Customer_Id	int
Dependant_ID	int
Status	tinyint

Er Model:



Development Tools:

1. MySQL Database management system
2. Flask framework

Home:

We have two buttons:

1- The first one for admin
2- The second one for login or sing up as customer

The side bar with:

1- Home
2- Team
3- Login /signup

Admin module:

Admin can view:

- A list of customers.
- A list of claims filed by a customer and pick a claim to view its details.
- Dependent claims which are filed by customer.
- Admin has an option to view either all claims or only unresolved claims.

Admin can add:

- A new hospital details and associating it with a plan type.
- A new plan types.

Customer module:

- Customers can create their own profile.
- Customer can view all hospitals in the health insurance
- Customer can file a claim for themselves or for one of their dependents.
- Customer can add a new dependent associated with a plan.
- A customer can purchase more than one plan for himself and his/her dependents. However, no individual (a customer / dependent) could be a beneficiary of more than one plan.
- Customers can view hospitals available under there purchased plans.

The project consists of some major files:

__INIT__:

Contains all configuration for our database as it is the way to safety bind database handler to flask app to manage connections.

VIEWS:

Mainly returns data that Flask turns into an outgoing response. It contains all http services we need such get and post requests which we use to read and create data in database using our web application as an interface.

Templates: files that contain static data as well as placeholders for dynamic data. It mainly contains customer and admin files to render them with specific data to produce a final document as required. Flask uses the Jinja template library to render templates.

Each step needs the others to complete the project with required constraints.