Patient Management System (PMS) Application for Heath Care Industries

In this Case Study we are going to develop a customized "**Patient Management System (PMS)**" application using Force.com platform.

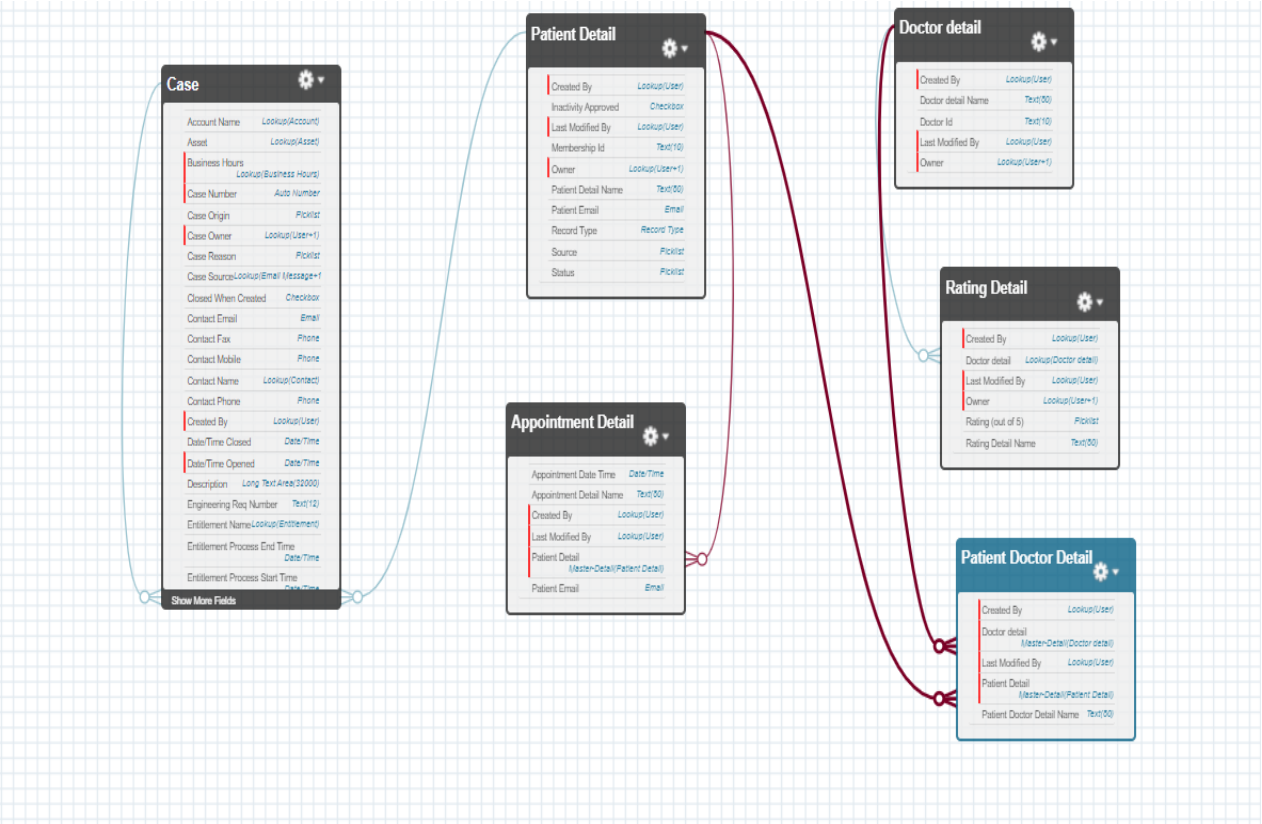
The purpose of the application is to automate the Patient life cycle from taking an appointment till doctor treatment's feedback in a very systematic way in a hospital.

- **Different Entities would be**: Patient, Appointment, Doctor, Cases (Standard Object) and Ratings
- **Object Wise Fields are:** -
    - **Patient:** Patient Name (Text), Source (Picklist values based on Record Type), Status (Active/Inactive), Membership Id (Auto Number), Patient Email, Inactivity Approved (Checkbox)
    - **Appointment:** Appointment Name, Appointment Date & Time (Date & Time), Patient Email (Email), Patient (MD on patient)
    - **Doctor:** Doctor Id (Auto-Number), Doctor Name (Text)
    - **Cases (Standard):** Source, Status, Priority, Patient (LKP on patient)
    - **Ratings:** Rating Name (Text), Rating (out of 5) -Picklist, Feedback (Text area), Doctor Name (LKP on doctor)
    - **Patient Doctor:** Patient Doctor Name (Text), Patient (MD on patient), Doctor (MD on doctor), It is a junction object between Patient and Doctor

Patient Management System (PMS) Application for Heath Care Industries

Relationships among Objects as below:

ERD:



Assumptions/Details

- Patient Possible values for **source** are**: -**
    - **Marketing (Record Type):** Social media, Newspaper
    - **Non-Marketing (Record Type):** Direct, Referral.
- **OWD would be Private by default unless stated**
- Possible values for **Status** are: Active and Inactive

Patient Management System (PMS) Application for Heath Care Industries

- Users using our application have Profile similar to 'Cloned System Admin Profile i.e., Helpdesk Profile' for Helpdesk User with below permissions: -
    - **Patient**: Read, Create

- **Relationship between different combinations to be identified: -**
    - Patient & Doctor (Many to Many)
    - Patient & Appointment (1:M using MD)
    - Patient & Case (1:M using LKP)
    - Rating & Doctor (1:M using LKP)

- **Following Fields should be visible for below Record Type:**
    - **Marketing RT:** Name, Membership Id, Source, Status, email, Inactivity Approved

    - **Non-Marketing RT:** Name, Membership Id, Source, Status, email, Inactivity Approved

Business Rules/Scenarios: -

Validation Rules: -

1. User should not be able to enter new patient record without Email Id. – Using Validation

2. Status for **Patient** cannot be inactive while creating new record.

3. **Appointment** time can't be in past.

4. System should not allow creation of case for inactive **patient**.

Workflow Rules: -

5. If no status entered while **Patient** creation, Assign status as active for patient. – Using Field Update

6. Send email to 'Patient email' on Appointment creation (auto email population) -> we have to perform 2 actions.

    a. Patient Email field update on Appointment from Patient Email Field

Patient Management System (PMS) Application for Heath Care Industries

b. Email Alert

Process Builder:

7. If **Patient** become inactive than update all child cases status to closed

8. Post on creator a chatter message whenever a “High Priority Case” created.

9. Create a Dummy appointment whenever a new patient is created.

Flows: -

10. Create a mechanism where user can submit feedback on scale 0-5 for doctor, using Screen flow.

a. Let’s create one screen flow, where you have to populate 2 fields 1st Doctor Id (Text field) and 2nd Rating out of 5(Picklist 0-5).

b. Based of Get-Record Element, we will fetch doctor record with matching doctor Id e.g. (DR101)

c. There will be the confirmation screen with populated details of fetched doctor.

d. Then Create Record element we will be creating a Rating record which will be child of Doctor record.

11. If Status of patient is Active and record type is non-marketing and source is blank then set source as direct as default (using record triggered)

Sharing Rules/Manual Sharing

13. Share patient record with Mr. Sunil<<**Helpdesk User**>> if source is newspaper, using criteria-based sharing

14. Share All patients with Mr. Suni<<**Helpdesk User**>>, which are created by admin using owner-based sharing

Field History Tracking

15. All Changes on patient should be tracked.

Patient Management System (PMS) Application for Heath Care Industries

Assignment Rules

16. Case assignment should be as follows: -

- If Priority is High-> **High Priority Queue**
- IF Priority is low-> **Low Priority Queue**

Escalation Rules

17. Cases should be dealt as per below: -

- High Priority Cases should be closed in 2 Hours otherwise assigned to Admin
- Low priority cases should be closed in 4 Hours otherwise assigned to Admin

Approval process (Done with manual submission and auto submission using process builder)

18. All the status change to inactive for a patient should be approved by Admin, and as soon as Admin approved, 'Inactivity approved' checkbox should be checked.

Note: this checkbox is read only on page layout.

Reports and dashboards

19: **Information Required by Higher Management (create reports accordingly)**: -

- Create a Dashboard which would give quick view about patients as per different sources. – Summary Report
- Create a dashboard which would give quick view about doctor's ratings. – Summary Report

Permission Set

20: One patient Mr. Sunil<<**Helpdesk user**>> having same profile needs edit access on patient object

Data Loader

21: Perform all operations on patient object: -

a. Export

Patient Management System (PMS) Application for Heath Care Industries

b. Insert
c. Update
d. Delete
e. Hard delete
f. Upsert

Apex Batch & Schedulable:

22: Create a daily job which will deactivate the patient if they don't have appointment for last 180 days.

Web Services:

23: Create a Webservices where you have to expose all the Active patients' details (Membership Id, Name, email, Inactivity Approved) to third party application using Workbench

LWC (Lightning Web Component)

24**:** Create a LWC Component where you have to search Patient details either by Name or Email and then populate searched result.

a. **Searched result must include following fields:**
    a. Membership Id, Name, email, Inactivity Approved