Online Appointment System for Outpatient Department of Public Hospitals

Project Guide: Ciaran Hayden

December 2018

Vishwajeet Khadilkar(x17169984)

Shikhar Srivastava(x18106960)

Lalit Pathak(x18110088)

Ugwuanyi Arinze J.(x18139442)

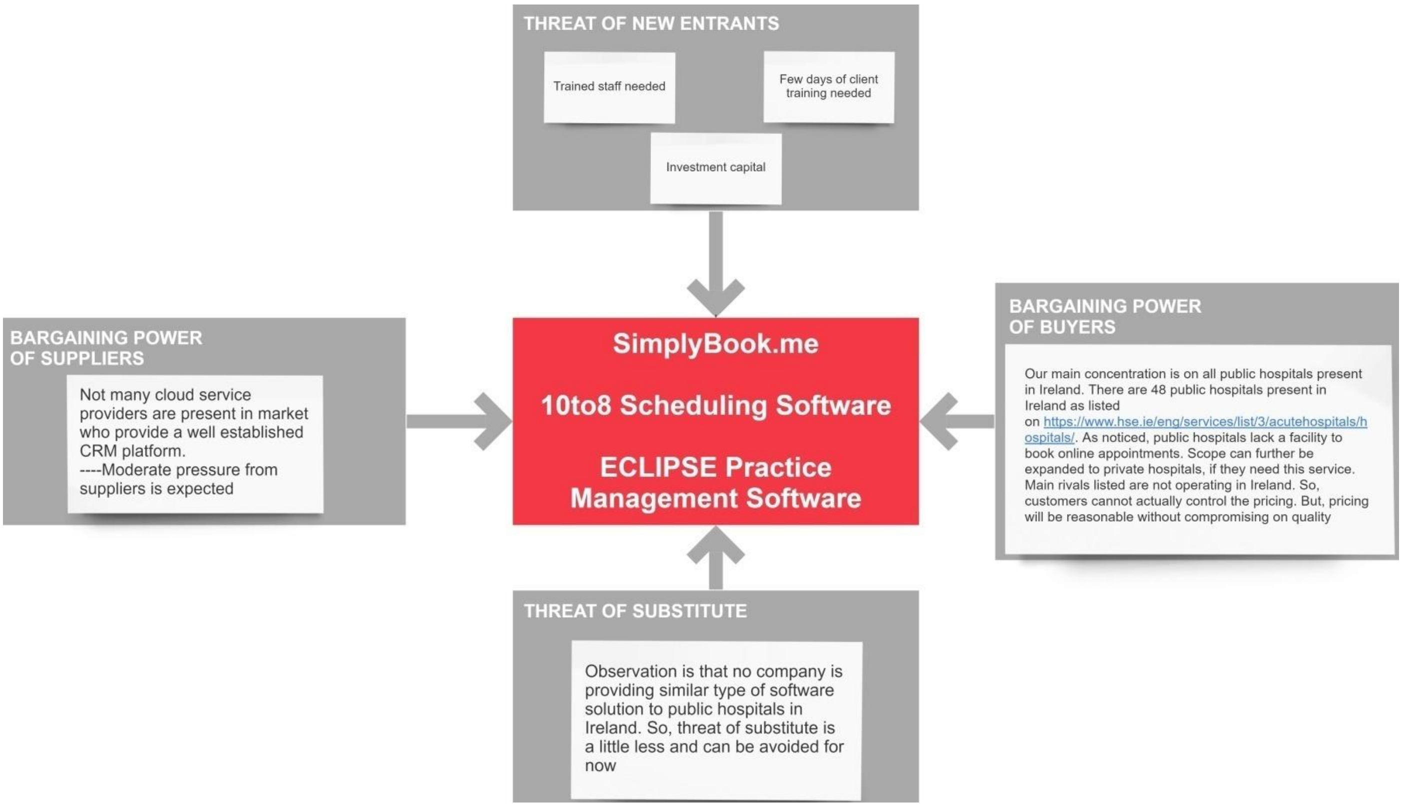
Table of Contents

Team Member	Contribution
Vishwajeet Khadilkar(x17169984)	27%
Shikhar Srivastava(x18106960)	27%
Lalit Pathak(x18110088)	27%
Ugwuanyi Arinze J.(x18139442)	19%

Organisation

We are Considering Beaumont Hospital (Beaumont Hospital) which is one of the biggest academic teaching Hospital in Ireland. This Hospital uses Tradition Booking system which includes Mail, Email and Phone call. All the above method required human interaction to analyse the reference letter. This process takes days to reach out to the patient about appointment status through the different communication Medium. The process is a very time-consuming. To overcome this problem we are implementing a cloud-based system, through which it is possible to book an appointment using a Smartphone, web browser with the help of the internet.

Porter's 5 Forces (Real Time Board)



THREAT OF NEW ENTRANTS

Trained staff needed

Few days of client training needed

Investment capital

BARGAINING POWER OF SUPPLIERS

Not many cloud service providers are present in market who provide a well established CRM platform.
----Moderate pressure from suppliers is expected

SimplyBook.me

10to8 Scheduling Software

ECLIPSE Practice Management Software

BARGAINING POWER OF BUYERS

Our main concentration is on all public hospitals present in Ireland. There are 48 public hospitals present in Ireland as listed on https://www.hse.ie/eng/services/list/3/acutehospitals/hospitals/. As noticed, public hospitals lack a facility to book online appointments. Scope can further be expanded to private hospitals, if they need this service. Main rivals listed are not operating in Ireland. So, customers cannot actually control the pricing. But, pricing will be reasonable without compromising on quality

THREAT OF SUBSTITUTE

Observation is that no company is providing similar type of software solution to public hospitals in Ireland. So, threat of substitute is a little less and can be avoided for now

Target user-group:

- Public Hospitals and hospitals which have traditional Appointment Booking system – Front-Desk, Doctors and Lab technicians
- General Practitioners
- Patients

Marketplace in which organisation participate

Healthcare organizations are one of the most important Organisations of the Country and these organizations are directly related to human health. Lots of healthcare system need IT support and management strategies to efficiently provide quality of services and consultation to the Customers.

Scope and objectives

Nowadays Patients' involvement increasing in terms of searching best hospital, facilities provided by hospitals and for selecting the Consultant. One of the main reasons for increasing the patients' involvement is due to the use of a medium like internet. There are Different ways for the patient to book the appointment for Consultation. First is to visit the hospital and Book the appointment, sending Electronic Mail or phone call to the hospital. These processes of booking have some problems. It is not possible for the patient who is living outside the city to visit the hospital and book an appointment. Many times it is difficult to identify the mail due to overload of documents in the mailbox. Telephone appointment booking required patient details and the person on the phone call who can help you quickly. There is huge chance of loss of document delivery by hand method of appoint booking .All these traditional methods include human interaction and sometimes because of human error, it may possible to increase the waiting period for the patient to book an appointment so appointment booking does not only depend on the availability of the slot but also depends on schedulers and telephone lines. All these problems may cause frustration and satisfaction to the Patient.

Rationale for this Implementing Specific Architecture

Infrastructure

Selection of cloud

- The Cloud-Based Online Appointment Booking System beneficial to the management to track and maintain a history of automatically scheduled appointment.
- At the same time it is useful for the patient to select the facilities provided by Hospital.
- Stakeholders can have a holistic view of the processes that take place in the backend.
- It is possible to track the time spent on each appointment request for the management.

Infrastructure

- Computers, laptop, mobile phones, tablet with internet connectivity
- The organisation Needs to Invest on Salesforce licence.

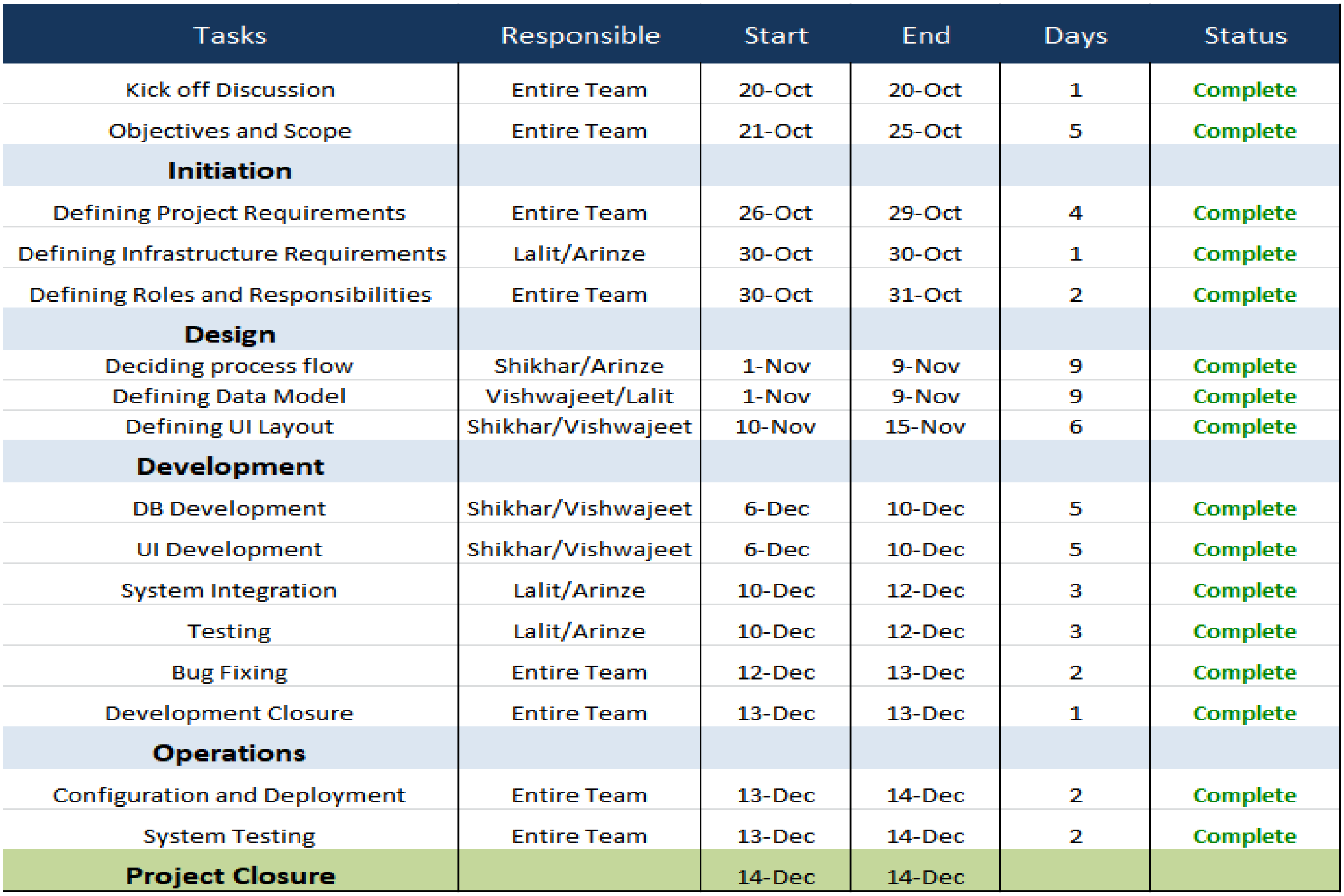
Product platform selection

There are different reasons for selection of cloud based Solution for Customer relationship management and also there is option of on-premise software but the problem with these software is they need maintenance and have high cost. Implementation of on-premise software takes time and require training session of the application.

- Flexibility that is, on demand services to scale up and scale down capacity as per requirement.
- Convenient recovery of data which saves lots of investment and time of the organisation.
- All data stored on servers so there is no need to maintain the system as cloud providers automatically roll out software update.
- Its best platform as it requires minimum hardware.
- Centralised documentation as well as control on document.
- Decreased waiting time
- Effective use of the appointment Slots.

Implementation Approach

Project has been divided in different phases. An image is attached below showing the project plan (Smartsheet.com):



Tasks	Responsible	Start	End	Days	Status
Kick off Discussion	Entire Team	20-Oct	20-Oct	1	Complete
Objectives and Scope	Entire Team	21-Oct	25-Oct	5	Complete
Initiation					
Defining Project Requirements	Entire Team	26-Oct	29-Oct	4	Complete
Defining Infrastructure Requirements	Lalit/Arinze	30-Oct	30-Oct	1	Complete
Defining Roles and Responsibilities	Entire Team	30-Oct	31-Oct	2	Complete
Design					
Deciding process flow	Shikhar/Arinze	1-Nov	9-Nov	9	Complete
Defining Data Model	Vishwajeet/Lalit	1-Nov	9-Nov	9	Complete
Defining UI Layout	Shikhar/Vishwajeet	10-Nov	15-Nov	6	Complete
Development					
DB Development	Shikhar/Vishwajeet	6-Dec	10-Dec	5	Complete
UI Development	Shikhar/Vishwajeet	6-Dec	10-Dec	5	Complete
System Integration	Lalit/Arinze	10-Dec	12-Dec	3	Complete
Testing	Lalit/Arinze	10-Dec	12-Dec	3	Complete
Bug Fixing	Entire Team	12-Dec	13-Dec	2	Complete
Development Closure	Entire Team	13-Dec	13-Dec	1	Complete
Operations					
Configuration and Deployment	Entire Team	13-Dec	14-Dec	2	Complete
System Testing	Entire Team	13-Dec	14-Dec	2	Complete
Project Closure		14-Dec	14-Dec		

An excel spreadsheet has also been attached below showing a Gantt chart representation of our project plan [3]:

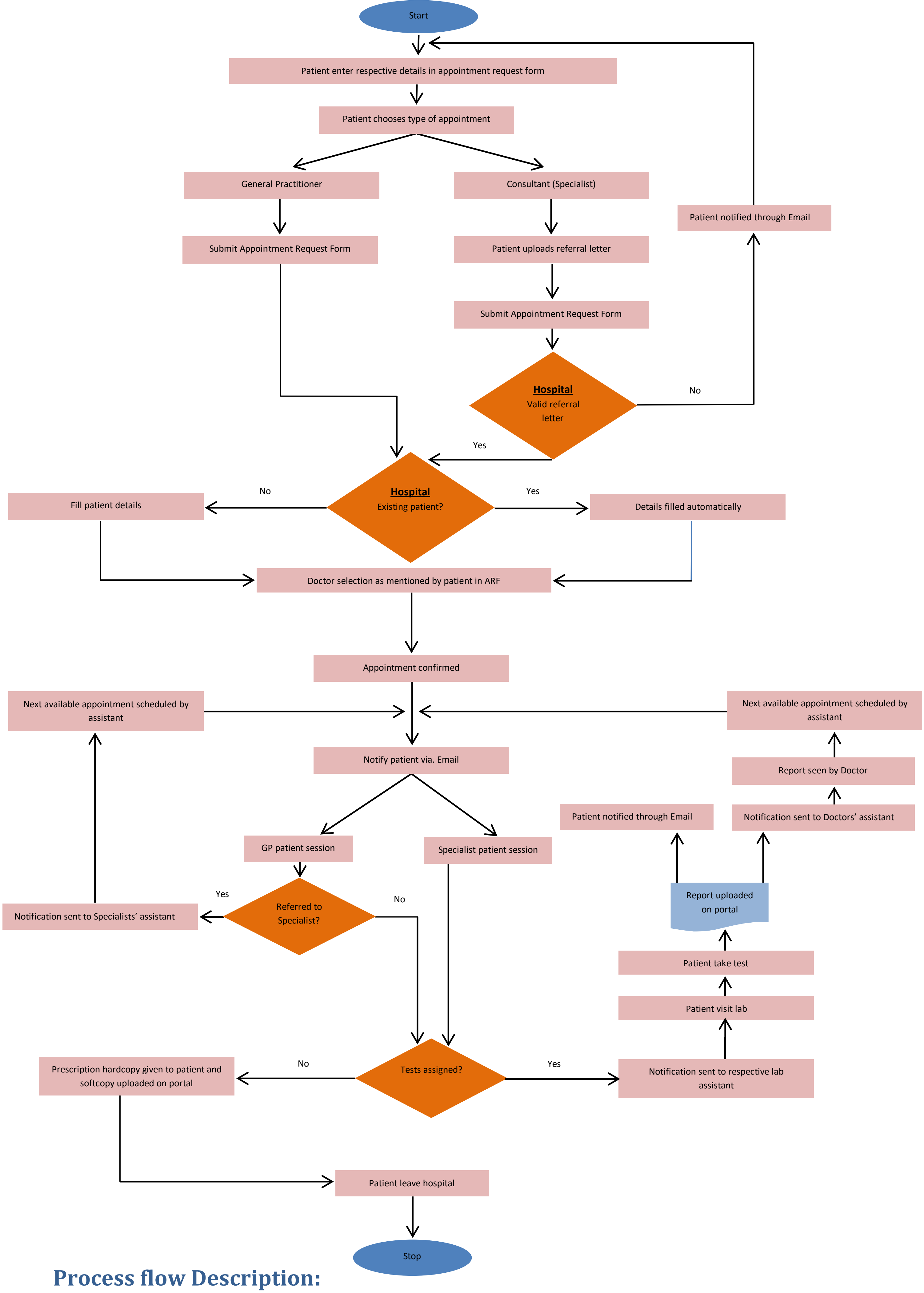
Project Plan.xlsx

Logical and Physical Design

Overall Requirement

- Online Booking System for health care organisation
- Maintain historical data of the appointments .
- Reduce the time required for appointment booking
- Centralise Documentation and ease of access to the user
- Security of the Patient data.

Process Flow Diagram


Start
Patient enter respective details in appointment request form
Patient chooses type of appointment
General Practitioner
Consultant (Specialist)
Submit Appointment Request Form
Patient uploads referral letter
Submit Appointment Request Form
Patient notified through Email
Hospital
Valid referral letter
No
Yes
Hospital
Existing patient?
No
Yes
Fill patient details
Details filled automatically
Doctor selection as mentioned by patient in ARF
Appointment confirmed
Next available appointment scheduled by assistant
Next available appointment scheduled by assistant
Notify patient via. Email
Report seen by Doctor
Patient notified through Email
Notification sent to Doctors' assistant
GP patient session
Specialist patient session
Yes
No
Referred to Specialist?
Notification sent to Specialists' assistant
Report uploaded on portal
Patient take test
Patient visit lab
Prescription hardcopy given to patient and softcopy uploaded on portal
No
Tests assigned?
Yes
Notification sent to respective lab assistant
Patient leave hospital
Stop

Process flow Description:

Step 1. Appointment Request Form

- Patient fills his/her personal details like Name, Age, Address, Contact and Email.
- Choice of “Type of Appointment”, whether patient wants to have a session with a General Practitioner or Consultant. List of GPs and Consultant will be made available for patient to choose from, if he/she is having any preference.
  i. If Consultant appointment is chosen then patient selects respective Consultant name from the list and uploads a referral letter and submits the form, which is then sent to hospital in the form of Email with referral letter as an attachment.
  ii. If General Practitioner appointment option is chosen then patient selects respective GP name from the list and submits the form, which is then sent to hospital in the form of Email. In this case, patient does not need to upload a referral letter.

Step 2. Appointment Booking by Hospital

- After receiving the completed appointment request form though Email, hospital validates the details entered by the patient and checks whether the patient is an existing patient or a new patient.
- If patient is an existing patient then his/her details for booking the appointment will be automatically filled in otherwise the hospital will manually fill the details for the appointment.
- Respective GP/Consultant selection is made as mentioned in the application request form by the patient.
- Next available appointment for respective GP/Consultant is booked and appointment confirmation details are the sent to the patient via. Email.

Step 3. GP/Consultant Patient Session

- GP Session: After the session, General Practitioner can either refer the patient to a consultant or if needed, can mark some tests which are required to be taken by the patient. If GP refers the patient to a consultant then in this case, a notification is sent to consultants’ assistant who then schedules the next available appointment of the patient with respective Consultant. If patient is neither referred to a consultant nor has been assigned any tests, then patient leaves hospital with the hardcopy of the prescription and the same uploaded on portal.
- Consultant Session: After the session, a Consultant can either mark some tests which are required to be taken by the patient or the patient leaves hospital with the hardcopy of the prescription and the same uploaded on portal.

Step 4. If Tests are Assigned

- If any test has been assigned to the patient, then a notification is sent to respective lab assistant listing the tests assigned by GP/Consultant.
- Next, the patient goes to lab and takes respective tests.
- Test report is then uploaded to the portal either immediately or at a later date.
- Notification is then sent to the patient and Consultants’ assistant stating that the report is now ready.
- Consultants’ assistant then books the patients’ next available appointment with Consultant.

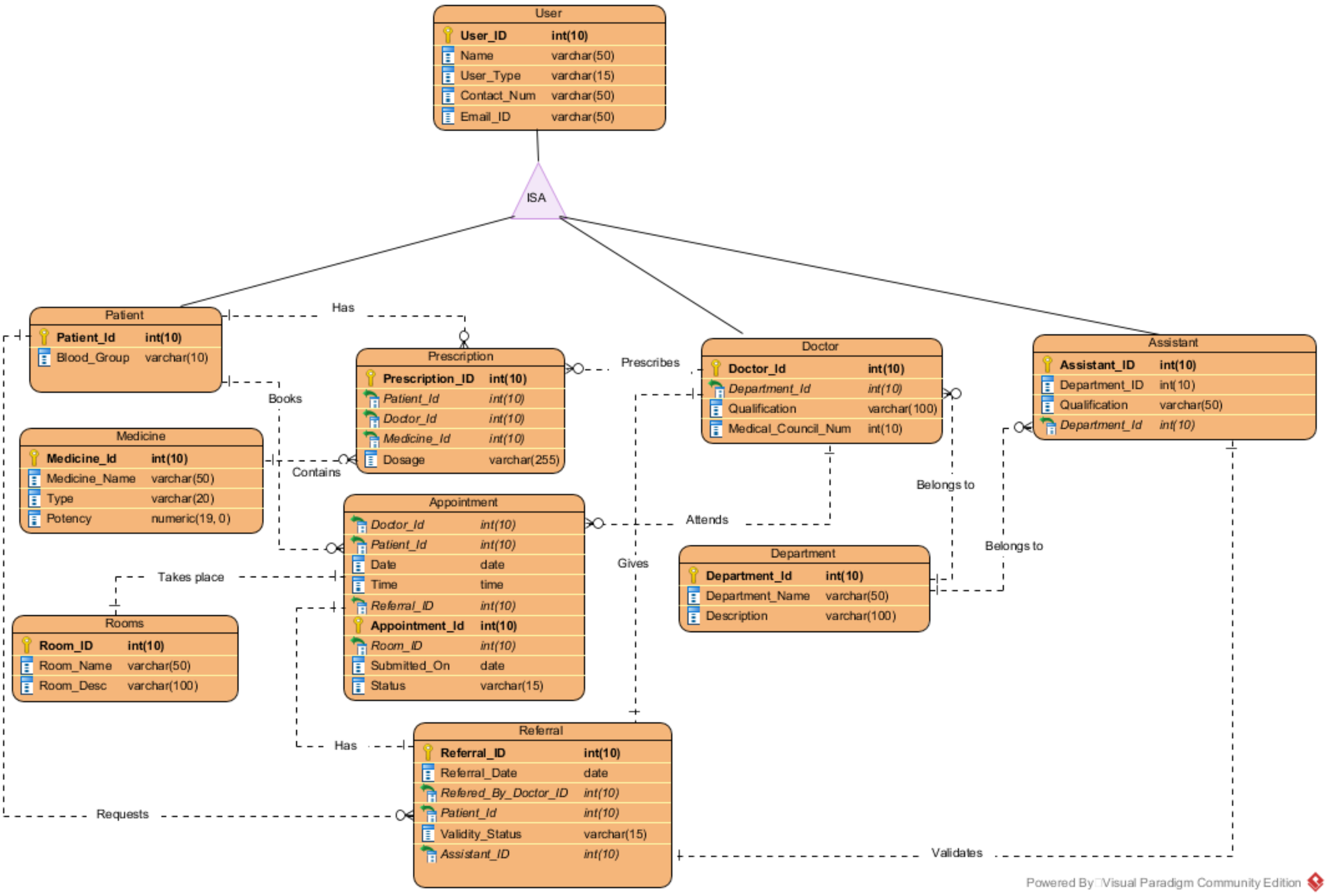
Data Capture Point

1) Patient Details from the form submitted by Patient.

2) Appointment Details.

3) Reports of the patient.

Entity Relationship Diagram:



Data Dictionary:

Table Name: User

Description: This table contains common details for all the subtypes of system users

Field Name	Datatype	Length	Description
User_ID	Integer	10	Primary Key
Name	Varchar	50	
Gender	Varchar	1	
Email_ID	Varchar	50	
DOB	Date		
Contact_Num	Varchar	25	
Type	Varchar	15	This column defines type of the user: Patient, Doctor or Assistant

Table Name: Patient

Description: This table is a derived table of user table for user type – patient and contains patient specific information which is not applicable to other users

Field Name	Datatype	Length	Description
Patient_ID	Integer	10	Foreign key and references to user_ID of the patient from user table. Acts as a primary key of this table.
Blood_Group	Integer	10	

Allergies	Varchar	100	

Table Name: Doctor

Description: This table is a derived table of user table for user type – Doctor and contains Doctor specific information which is not applicable to other users

Field Name	Datatype	Length	Description
Doctor_ID	Integer	10	Foreign key and references to user_ID of the doctor from user table. Acts as a primary key of this table.
Department_ID	Integer	10	Foreign key from department table
Qualification	Varchar	50	
Medical_Council_Num	int	10	Registration number of GP/consultant with medical council

Table Name: Assistant

Description: This table is a derived table of user table for user type – Assistant and contains Assistant specific information which is not applicable to other users

Field Name	Datatype	Length	Description
Assistant_ID	Integer	10	Foreign key and references to user_ID of the assistant from user table. Acts as a primary key of this table.
Department_ID	Integer	10	Foreign key from department table
Qualification	Varchar	50	

Table Name: Department

Description: This table contains information abot the departments of the hospital's outpatient section

Field Name	Datatype	Length	Description
Department_ID	Integer	10	Primary key
Department_Name	Varchar	50	
Description	Varchar	100	

Table Name: Room

Description: This table contains information about the rooms available for appointments, medical tests etc and their location and description

Field Name	Datatype	Length	Description
Room_ID	Integer	10	Primary key
Room_Name	Varchar	50	
Location	Varchar	100	
Description	Varchar	100	

Table Name: Medicine

Description: This table contains information about the medicines. These medicines will be visible to doctors while prescribing them to patients.

Field Name	Datatype	Length	Description
Medicine_ID	Integer	10	Primary key
Medicine_Name	Varchar	50	

Potency	Varchar	100	
Type	Varchar	100	Type of medicine: eg. tablet, syrup etc

Table Name: Referral

Description: This table captures referral details for patient's appointment and its validity

Field Name	Datatype	Length	Description
Referral_ID	Integer	10	Primary key
Referral_Date	date		
Refered_By_Doctor_ID	int	10	Foreign key and references to user_ID of the doctor from user table.
Patient_Id	int	10	Foreign key and references to user_ID of the patient from user table.
Validity_Status	varchar	15	After verification of the referral, assistant will mark this as accept or reject
Assistant_ID	int	10	Foreign key and references to user_ID of the assistant from user table.

Table Name: Appointment

Description: This table stores information about the appointment of patient with doctor

Field Name	Datatype	Length	Description
Doctor_Id	Int	10	Foreign key and references to user_ID of the doctor from user table.
Patient_Id	Int	10	Foreign key and references to user_ID of the patient from user table.
Date	Date		
Time	Time	7	
Referral_ID	Int	10	Foreign key and references to Referral_ID of the from Referral table.
Appointment_Id	Int	10	Primary key
Room_ID	Int	10	Foreign key and references to Room_ID of the from Room table.
Submitted_On	Date		
Status	Varchar	15	Status : Approved/Rejected/OnHOld

Table Name: Prescription

Description: This table contains medicines prescribed to the patient by doctor

Field Name	Datatype	Length	Description
Prescription_ID	Int	10	Primary key
Patient_Id	Int	10	Foreign key and references to user_ID of the patient from user table.
Doctor_Id	Int	10	Foreign key and references to user_ID of the doctor from user table.
Medicine_Id	Int	10	Foreign key and references to Medicine_ID of the from Medicine table.
Dosage	Varchar	255	Frequency and quantity of medicine to be consumed
Date	Varchar	50	Stores Date when Prescription was uploaded

Business & Data Validation rules - Description and Implementation

BR1

Name:	A GP and Consultant must have a Medical council number

Identifier:	BR1
Description:	All the general practitioners and consultants (Specialists) must have a valid registration number provided by medical council.
Implementation	A field named: Medical_Council_Num is included in table #Doctor. This field cannot be left null.

BR2

Name:	A medical test can be performed only when a general practitioner or consultant recommends it
Identifier:	BR2
Description:	Medical tests can only be performed if a GP or consultant suggest them to the patient
Implementation	A field named: Doctor_ID is included in table #Report. This field cannot be left null.

BR3

Name:	An appointment with consultant(specialist) will only be confirmed upon submission of valid referral letter from General Practitioner
Identifier:	BR3
Description:	A patient can request appointment with consultant only if it is accompanied by a valid referral letter from a GP
Implementation	A table named #Referral is maintained to store referral related details. Primary key from this table named: Referral_ID is included in table #Appointment as a foreign key. This field cannot be left null.

BR4

Name:	An appointment can have only predefined status types
Identifier:	BR4
Description:	The possible values of an appointment's status are Approved, Rejected, On Hold, To Be Processed, Processed
Implementation	A field named: Status is included in table #Apointment. Value for this field can only be selected from predefined set of values using a drop-down list

BR5

Name:	The referral letter be in certain file type and size
Identifier:	BR5
Description:	The referral letter must be uploaded in pdf or doc/docx format with size less than 1MB only
Implementation	A validation script will be used to perform this check on patient's appointment submission form

BR6

Name:	An appointment request must be processed within SLA
Identifier:	BR6
Description:	An appointment request must be processed within one business day.
Implementation	A field named: Submitted_On is included in table #Appointment. A reminder will be triggered if a request is not processed within SLA. The value of column status must then be changed from to be processed to either approved/rejected/kept on hold or processed.

BR7

Name:	Patient's health history must be documented and made available to GP/Consultants
Identifier:	BR7
Description:	All the current and historic reports and prescriptions of the patient must

	be made available to GP/Consultants for ease of understanding of patient's health history
Implementation	A table named #Report is maintained for this purpose. For each test taken and prescription prescribed, an entry will be made in this table for future references.

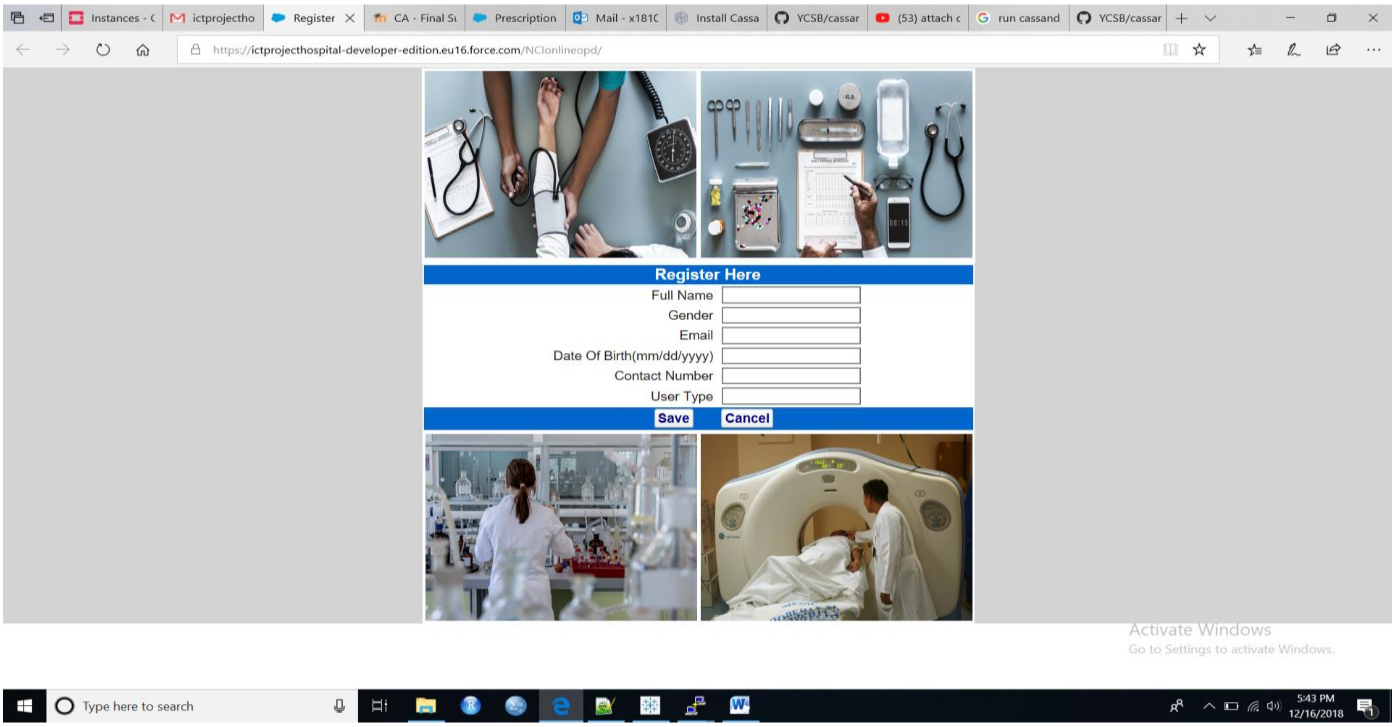
BR8

Name:	A report must contain details of the technician that performs and documents the medical test
Identifier:	BR8
Description:	Medical tests can only be performed if a GP or consultant suggests them to the patient. It is lab technician's responsibility to verify and perform the necessary tests and upload the results in the system for further use.
Implementation	A field named: Assistant_ID is included in table #Report. This field cannot be left null.

Integration

Registration Page

For patients to access this online appointment booking service, we have built a registration page where new patients can go and register themselves. After registering, patient can request for an appointment through an online appointment request form. We have also made our registration page public for which the link is -- https://ictprojecthospital-developer-edition.eu16.force.com/NCIonlineopd/. Image of registration web page is shown as below:

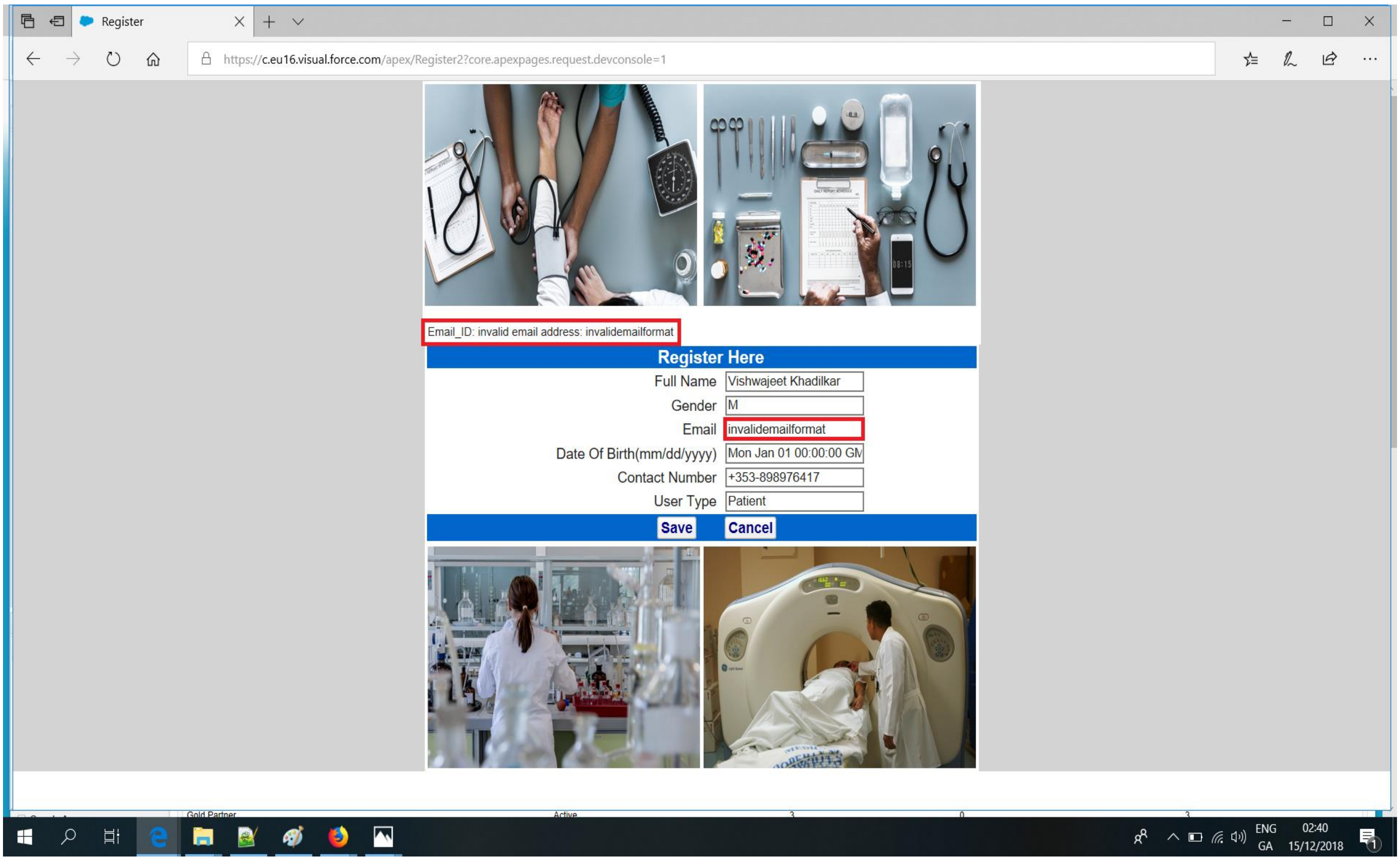


Register Here

Full Name	
Gender	
Email	
Date Of Birth(mm/dd/yyyy)	
Contact Number	
User Type	

Save Cancel

Patient will not be able to create a new record if he/she has entered an invalid value in any of the fields on the registration web page. Please refer below image where we have checked for one of such validation on email ID:



Register
https://c.eu16.visual.force.com/apex/Register2?core.apexpages.request.devconsole=1

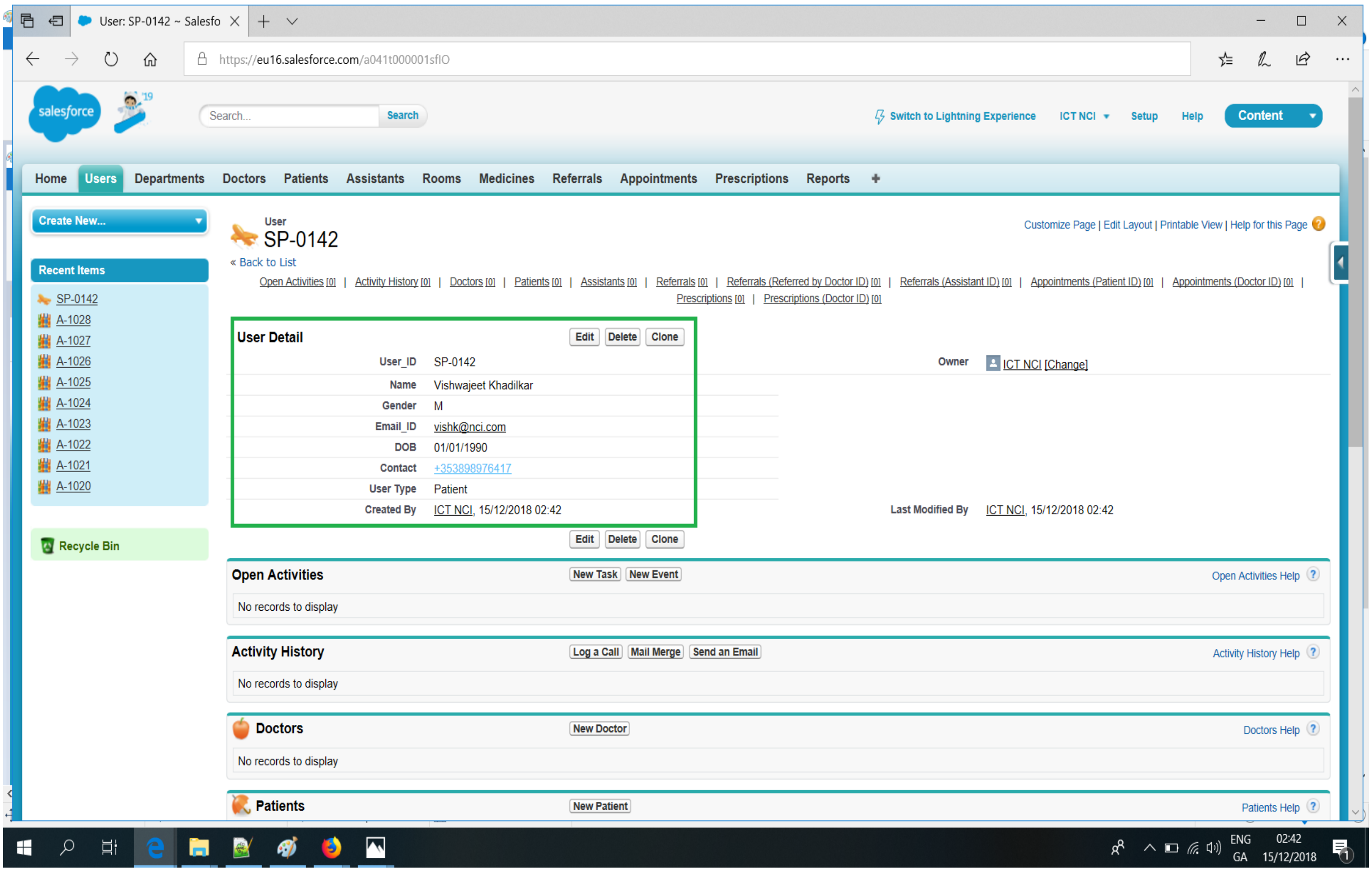
Email_ID: invalid email address: invalidemailformat

Register Here

Full Name	Vishwajeet Khadilkar
Gender	M
Email	invalidemailformat
Date Of Birth(mm/dd/yyyy)	Mon Jan 01 00:00:00 GM
Contact Number	+353-898976417
User Type	Patient

Save
Cancel
ENG
GA
02:40
15/12/2018

We have also checked whether a record is inserted if patient has entered everything in a correctly. Image attached below:



User
SP-0142

User Detail

User_ID	SP-0142
Name	Vishwajeet Khadilkar
Gender	M
Email_ID	vishk@nci.com
DOB	01/01/1990
Contact	+353898976417
User Type	Patient
Created By	ICT NCI, 15/12/2018 02:42
Owner	ICT NCI [Change]
Last Modified By	ICT NCI, 15/12/2018 02:42

Open Activities

No records to display

Activity History

No records to display

Doctors

No records to display

Patients

Registration Page Integration

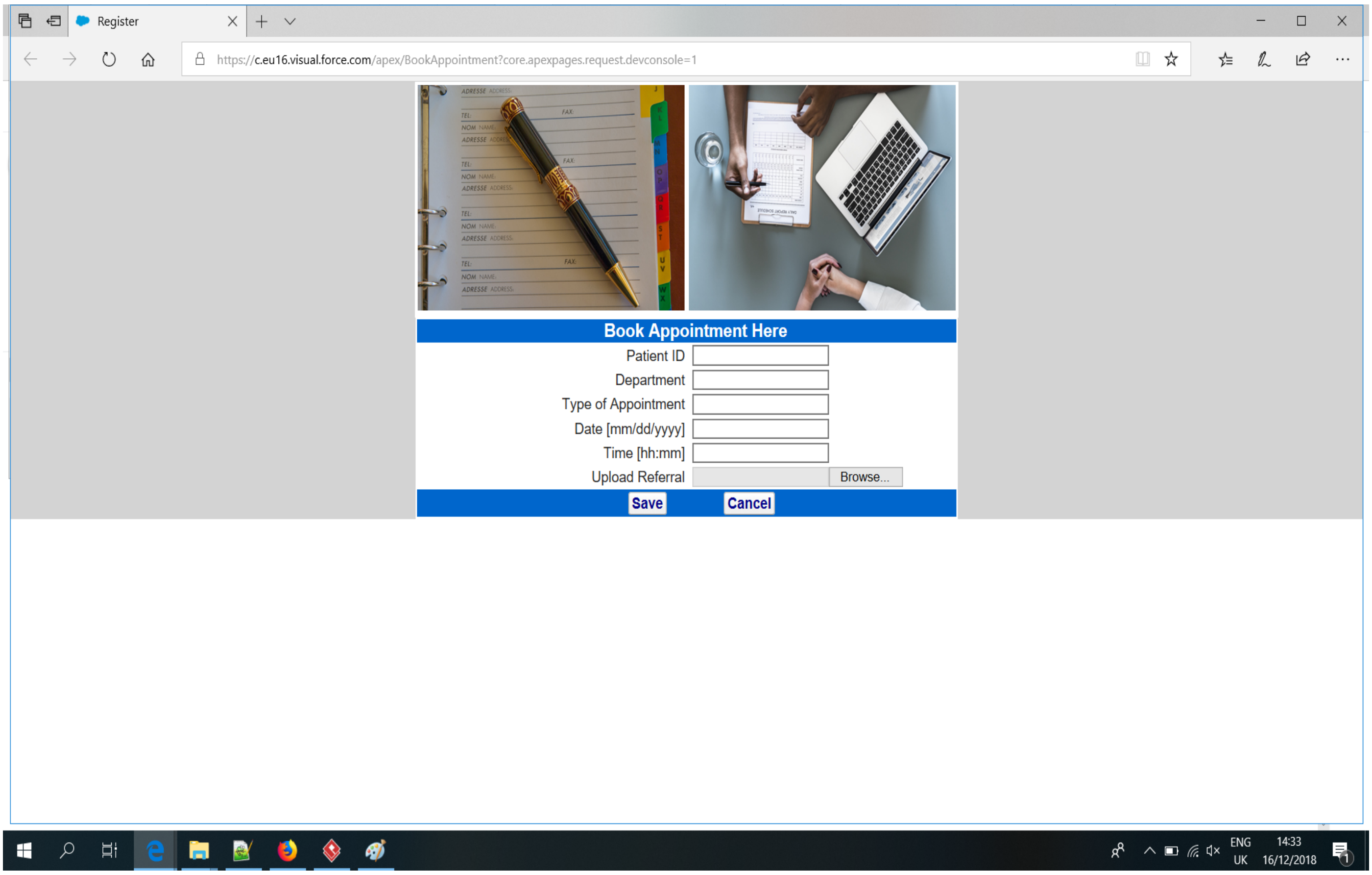
Fields present on the registration web page are mapped to the fields in User custom object as follows:

Registration Web Page	User Custom Object
Full name	User__c.Name__c
Gender	User__c.Gender__c
Email	User__c.Email_ID__c
Date Of Birth(mm/dd/yyyy)	User__c.DOB__c
Contact Number	User__c.Contact__c
User Type	User__c.User_Type__c

Code for the registration web page is present in the appendix of this document.

Appointment Page

This page will be used to book an appointment if a patient wishes to. Appointment booking page will look like:


Register
https://c.eu16.visual.force.com/apex/BookAppointment?core.apexpages.request.devconsole=1
ENG
UK
14:33
16/12/2018

Book Appointment Here

Patient ID

Department

Type of Appointment

Date [mm/dd/yyyy]

Time [hh:mm]

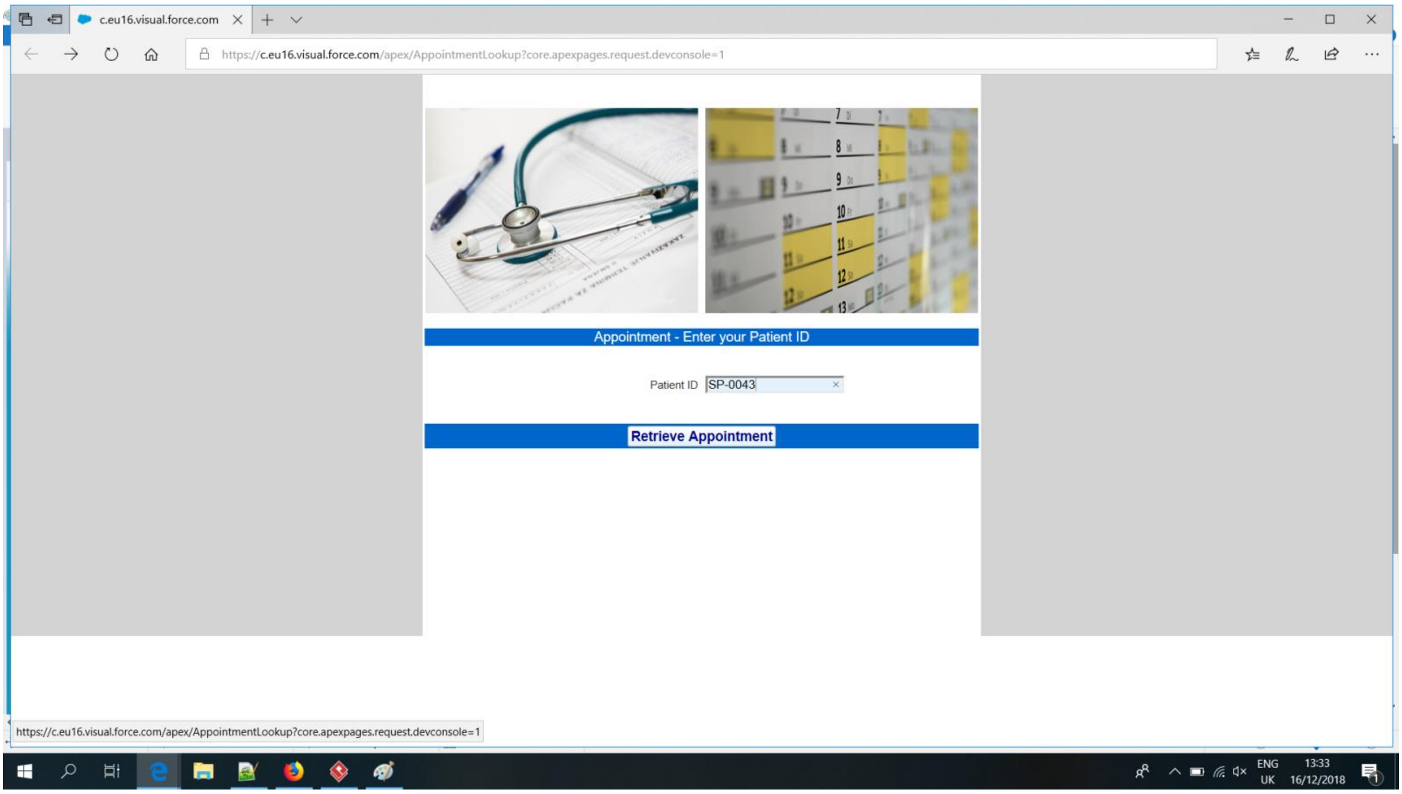
Upload Referral Browse...

Save Cancel

Code written to build this web page is present in the appendix.

History of Appointment Retrieval

Our project also comes with a web page through which a patient will be able to retrieve his/her upcoming appointment details and will also be able to see his/her entire appointment history with the hospital. This is how our appointment retrieval page looks like:


c.eu16.visual.force.com
https://c.eu16.visual.force.com/apex/AppointmentLookup?core.apexpages.request.devconsole=1
Retrieve Appointment
https://c.eu16.visual.force.com/apex/AppointmentLookup?core.apexpages.request.devconsole=1
ENG
UK
13:33
16/12/2018

Appointment - Enter your Patient ID

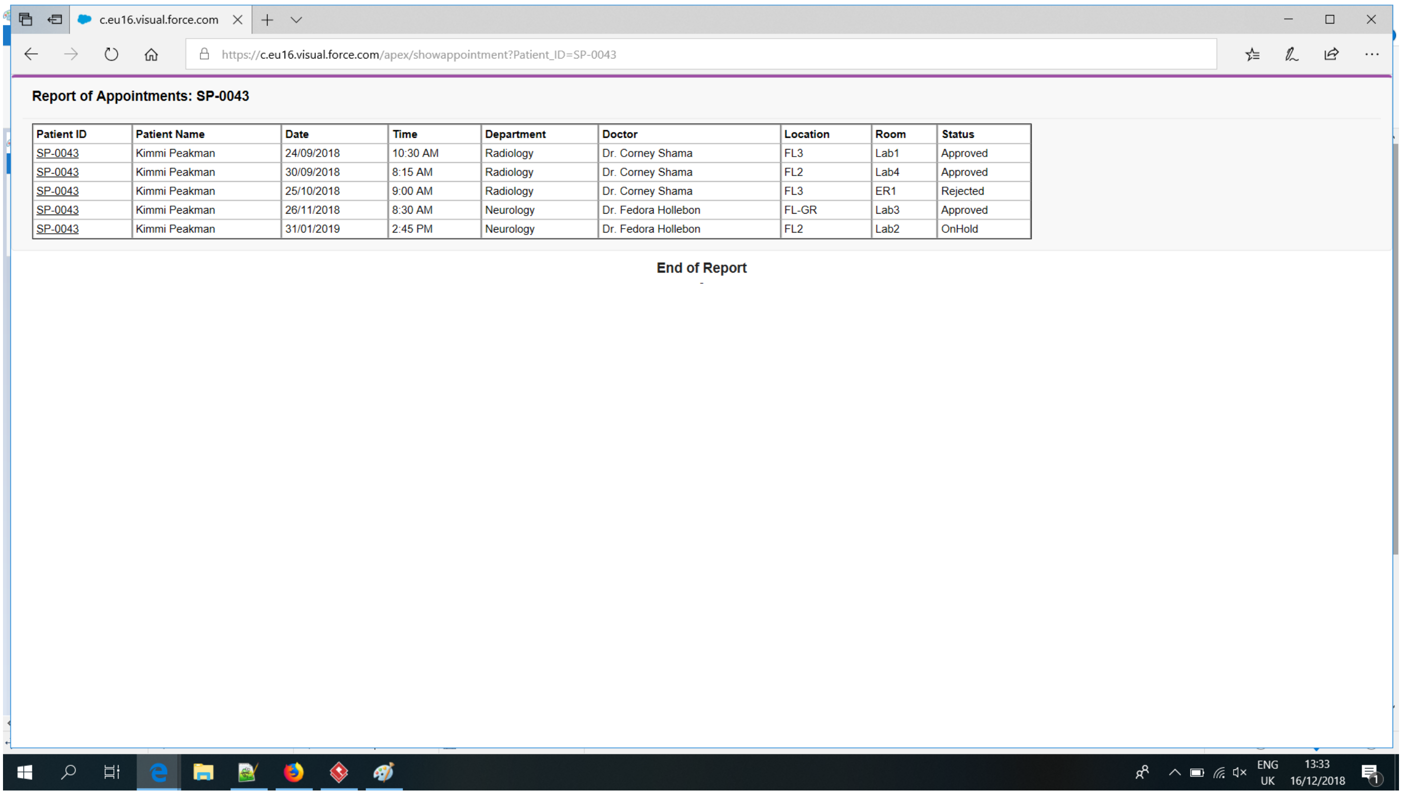
Patient ID SP-0043

Integration of Appointment Retrieval Page

When a patient ID is entered and “Retrieve Appointment” button is clicked, “AppointmentLookup” class is called which further calls the web page “showAppointment” which shows appointment details for the patient ID entered. Details of appointment are shown on screen because “showAppointment” page invokes another class “RetrieveAppointment”. Fields which are shown on screen to user are pulled from “Appointment”, “User”, “Department” and “Room” custom objects. The relationship between these tables is established using SOQL syntax which is equivalent to joins in SQL. All codes for both classes and web pages are present in the appendix. List of fields pulled are:

1. SLx.Patient_ID__c
2. SLx.Patient_ID__r.Name__c
3. SLx.Date__c
4. SLx.Time__c
5. SLx.Department_ID__r.Department_Name__c
6. SLx.Doctor_ID__r.Name__c
7. SLx.Room_ID__r.Location__c
8. SLx.Room_ID__r.Room_Name__c
9. SLx.Status__c

This is how the appointment details retrieval page looks like:


c.eu16.visual.force.com
https://c.eu16.visual.force.com/apex/showappointment?Patient_ID=SP-0043
ENG
UK
13:33
16/12/2018

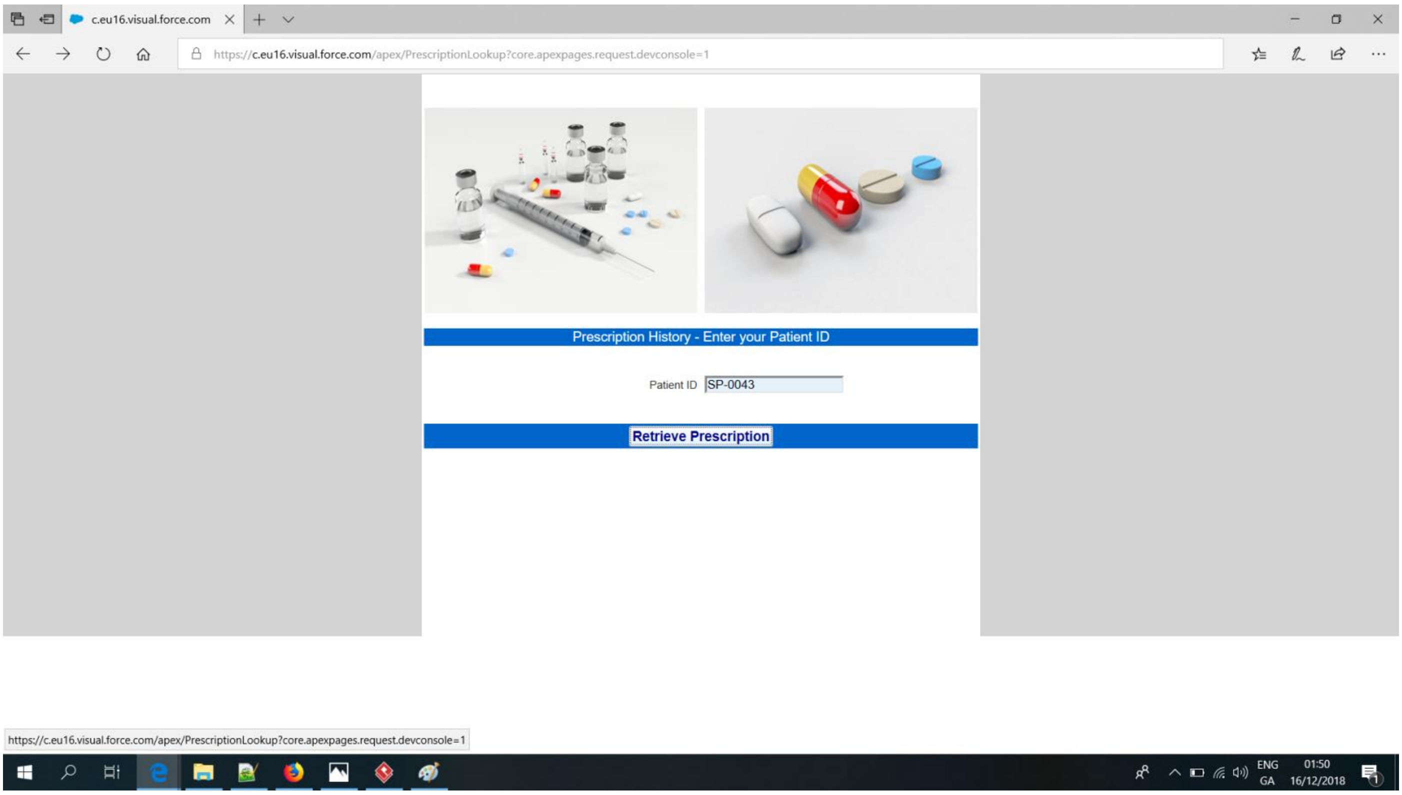
Report of Appointments: SP-0043

Patient ID	Patient Name	Date	Time	Department	Doctor	Location	Room	Status
SP-0043	Kimmi Peakman	24/09/2018	10:30 AM	Radiology	Dr. Corney Shama	FL3	Lab1	Approved
SP-0043	Kimmi Peakman	30/09/2018	8:15 AM	Radiology	Dr. Corney Shama	FL2	Lab4	Approved
SP-0043	Kimmi Peakman	25/10/2018	9:00 AM	Radiology	Dr. Corney Shama	FL3	ER1	Rejected
SP-0043	Kimmi Peakman	26/11/2018	8:30 AM	Neurology	Dr. Fedora Hollebon	FL-GR	Lab3	Approved
SP-0043	Kimmi Peakman	31/01/2019	2:45 PM	Neurology	Dr. Fedora Hollebon	FL2	Lab2	OnHold

End of Report

Prescription Retrieval Page

A patient can see his/her prescription details from this page. History of all prescriptions of a particular patient will be visible on this page. Image of web page:


c.eu16.visual.force.com
https://c.eu16.visual.force.com/apex/PrescriptionLookup?core.apexpages.request.devconsole=1
Prescription History - Enter your Patient ID
Retrieve Prescription
https://c.eu16.visual.force.com/apex/PrescriptionLookup?core.apexpages.request.devconsole=1
ENG
GA
01:50
16/12/2018

Patient ID SP-0043

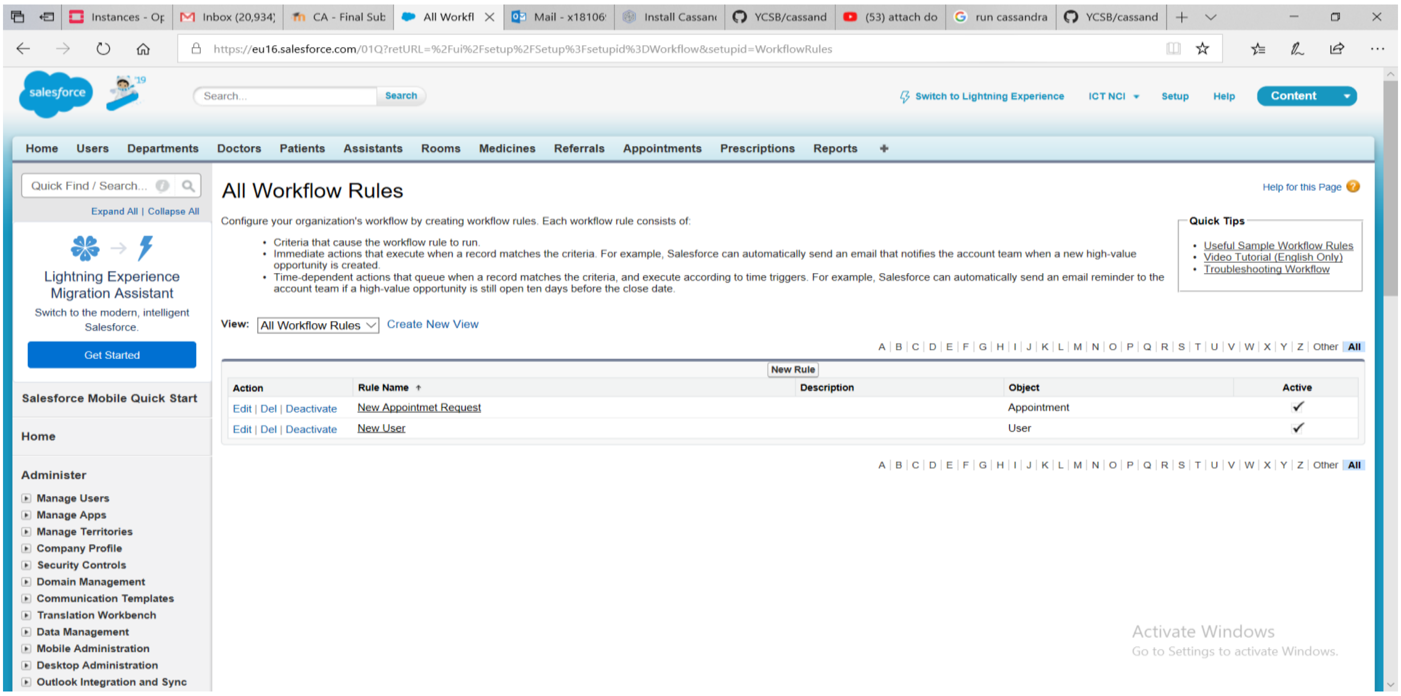
Integration of Prescription Retrieval Page

When a patient ID is entered and "Retrieve Prescription" button is clicked, "PrescriptionLookup" class is called which further calls the web page "showPrescription" which shows prescription details for the patient ID entered. Details of prescription are shown on screen because “showPrescription” page invokes another class “RetrievePrescription”. Fields which are shown on screen to user are pulled from “Prescription”, “User” and “Medicine” custom objects. The relationship between these tables is established using SOQL syntax which is equivalent to joins in SQL. All codes for both classes and web pages are present in the appendix. List of fields pulled are:

1. SLx.Patient_ID__c
2. SLx.Patient_ID__r.Name__c
3. SLx.Doctor_ID__r.Name__c
4. SLx.Medicine_ID__r.Medicine_Name__c
5. SLx.Medicine_ID__r.Potency__c
6. SLx.Medicine_ID__r.Type__c
7. SLx.Dosage__c
8. SLx.Date__c

Workflows and Email Alerts

Project is also capable of sending email alerts if any patient is registering for the first time or if an appointment has been confirmed and booked for a particular patient. An email template was created for sending these automated emails to patients. Images of workflows and email templates created are as follows:



All Workflow Rules

Configure your organization's workflow by creating workflow rules. Each workflow rule consists of:

- Criteria that cause the workflow rule to run.
- Immediate actions that execute when a record matches the criteria. For example, Salesforce can automatically send an email that notifies the account team when a new high-value opportunity is created.
- Time-dependent actions that queue when a record matches the criteria, and execute according to time triggers. For example, Salesforce can automatically send an email reminder to the account team if a high-value opportunity is still open ten days before the close date.

Quick Tips

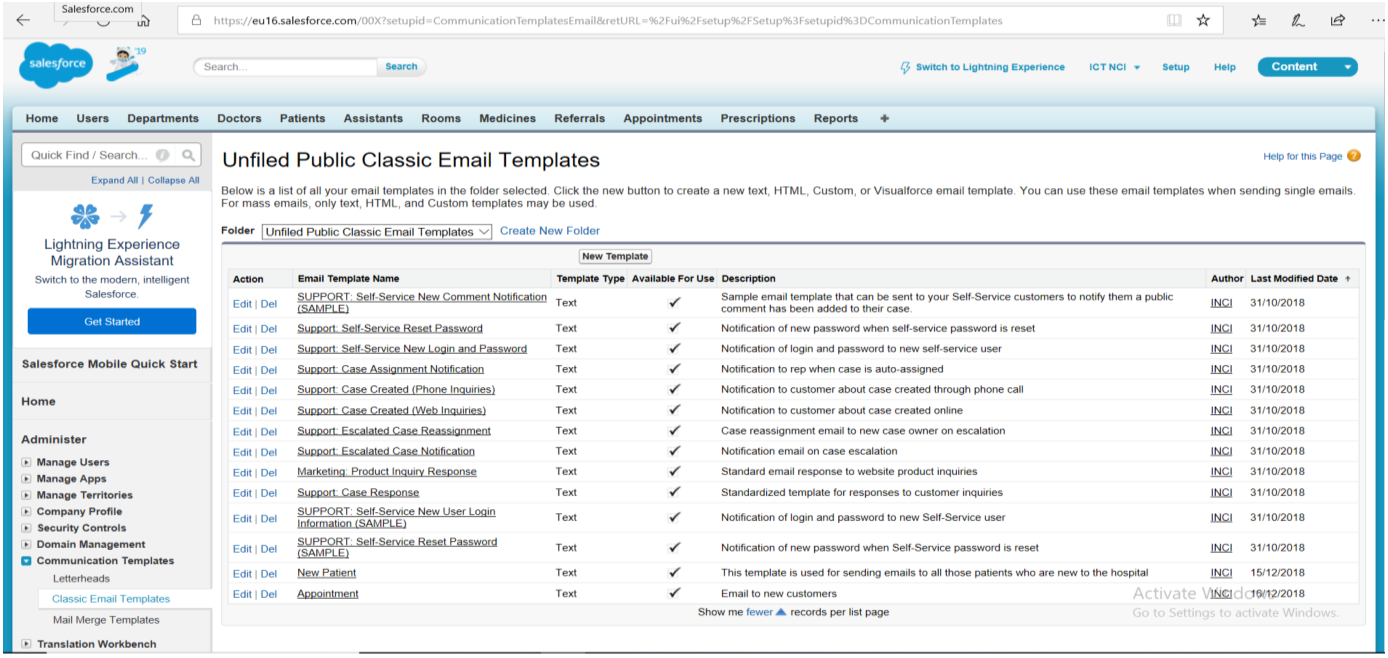
- Useful Sample Workflow Rules
- Video Tutorial (English Only)
- Troubleshooting Workflow

View: All Workflow Rules Create New View

New Rule

Action	Rule Name	Description	Object	Active
Edit | Del | Deactivate	New Appointmet Request		Appointment	✓
Edit | Del | Deactivate	New User		User	✓

Last 2 email templates present in the image below are the templates which we created for automated emails to patients



Unfiled Public Classic Email Templates

Below is a list of all your email templates in the folder selected. Click the new button to create a new text, HTML, Custom, or Visualforce email template. You can use these email templates when sending single emails. For mass emails, only text, HTML, and Custom templates may be used.

Folder: Unfiled Public Classic Email Templates | Create New Folder

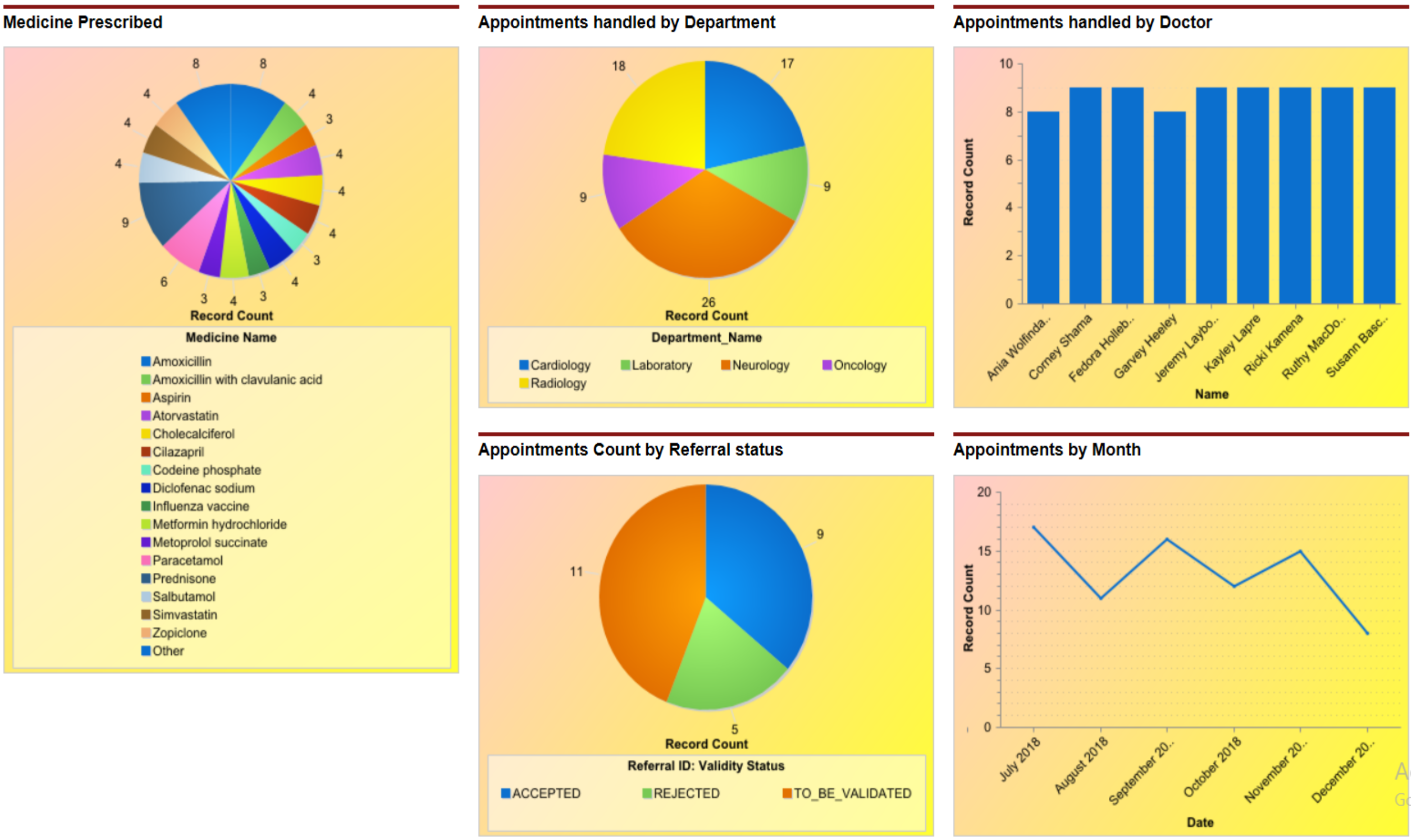
Action	Email Template Name	Template Type	Available For Use	Description	Author	Last Modified Date
Edit | Del	SUPPORT: Self-Service New Comment Notification (SAMPLE)	Text	✓	Sample email template that can be sent to your Self-Service customers to notify them a public comment has been added to their case.	INCI	31/10/2018
Edit | Del	Support: Self-Service Reset Password	Text	✓	Notification of new password when self-service password is reset	INCI	31/10/2018
Edit | Del	Support: Self-Service New Login and Password	Text	✓	Notification of login and password to new self-service user	INCI	31/10/2018
Edit | Del	Support: Case Assignment Notification	Text	✓	Notification to rep when case is auto-assigned	INCI	31/10/2018
Edit | Del	Support: Case Created (Phone Inquiries)	Text	✓	Notification to customer about case created through phone call	INCI	31/10/2018
Edit | Del	Support: Case Created (Web Inquiries)	Text	✓	Notification to customer about case created online	INCI	31/10/2018
Edit | Del	Support: Escalated Case Reassignment	Text	✓	Case reassignment email to new case owner on escalation	INCI	31/10/2018
Edit | Del	Support: Escalated Case Notification	Text	✓	Notification email on case escalation	INCI	31/10/2018
Edit | Del	Marketing: Product Inquiry Response	Text	✓	Standard email response to website product inquiries	INCI	31/10/2018
Edit | Del	Support: Case Response	Text	✓	Standardized template for responses to customer inquiries	INCI	31/10/2018
Edit | Del	SUPPORT: Self-Service New User Login Information (SAMPLE)	Text	✓	Notification of login and password to new Self-Service user	INCI	31/10/2018
Edit | Del	SUPPORT: Self-Service Reset Password (SAMPLE)	Text	✓	Notification of new password when Self-Service password is reset	INCI	31/10/2018
Edit | Del	New Patient	Text	✓	This template is used for sending emails to all those patients who are new to the hospital	INCI	15/12/2018
Edit | Del	Appointment	Text	✓	Email to new customers	INCI	16/12/2018

Show me fewer ▲ records per list page

Activate Windows
Go to Settings to activate Windows.

Reporting and Analytics

We have built a dashboard which is having four reports in it. Image of the dashboard is as attached as below:



Following reports are created to deliver insights of our project:

Appointment Handled by Doctor

This report shows the number of appointments a doctor has handled till date. By this report, hospital management will be able to monitor which doctor is handling a large number of reports and is overloaded.

Appointment Handled by Department

Number of appointments is shown in this graph which is grouped by department. Insight, like which department is getting many appointment requests can be seen using this chart. Hospital management can act accordingly to optimise things in department with high appointment requests which can further streamline processes.

Medicine Prescribed

By looking at the number of times a medicine has been prescribed, hospital can get insights like in a particular season, which medicines were prescribed a lot which can further be shared by the hospital with pharmacies and medical representatives to confirm availability in future.

Appointment Count by Referral Status

Number of appointment count by referral status will help the hospital to see appointment count which required a referral letter and which did not, categorized by four categories – 'REJECTED', 'TO_BE_VALIDATED', 'ACCEPTED', 'REFERRAL_NOT_REQUIRED'.

Appointments by Month

This report will help the hospital to monitor number of appointments it has handled on a monthly basis and will help to track months or periods during which there is a rush and many people look for an appointment at the hospital.

Approach Followed to Build Dataset

Mock data is prepared for our project from Mockaroo which is based on 10 custom objects built in salesforce. Fields of all 10 tables are explained in detail under entity relationship heading of this document. In tables defined in our entity relationship diagram, we have created auto generated unique IDs in salesforce. Data present in other fields like Allergy (Wikipedia), Medicine Names (Pharmac.govt.nz), Department (Medicalcouncil.ie), Doctor Qualification (Medicalcouncil.ie) and Medical Council Number (Medicalcouncil.ie) was searched over the internet and real values were inserted in this system. Fields for which Real like values were inserted in tables are as follows:

1. Allergens – List of common allergens are picked up from Wikipedia and then inserted in this field which is present in 'Patient' table.
2. Medicine Name – Different commonly used medicines are taken from pharmac.govt.nz and are used in 'Medicine' table under medicine name field.
3. Doctor Qualification – Few values are taken from medicalcouncil.ie which are used in field qualification in table 'Doctor'
4. Medical Council Number – Medical council number is also taken from the same web site form where doctor qualification is taken. Here, we noticed that for every doctor (if you search doctor by any forename) the medical council number was of six digits. We have built our data for this field accordingly.

Reports were kept in mind while creating the tables and inserting data.

Steps Taken to Build System

Steps followed to build this system are as follows:

Step 1. All custom objects required for our system were built and represented as tabs. Default objects present in salesforce were hidden. Relationship between objects was defined while creating these custom objects.

Step 2. Data for these custom objects was prepared on mockaroo. Sample of real life data was used for few fields like medicine name, allergy etc.

Step 3. Third step was to build the reports and put them in a single dashboard. Reports like 'Appointment Count by Doctor', 'Appointment Count by Department', 'Number of Times a Medicine has been Prescribed' and 'Number of Appointments by Referral Status'

Step 4. Registration web page was built for patients to register themselves if they are booking an appointment for the first time. ID generated from this page will further be used to apply for an appointment through appointment request for web page.

Step 5. Next, the appointment request form web page was created. This web page will be used by patients to apply for an appointment.

Step 6. Registration Web Page was made public by assigning it a domain. Workflows for email alerts were also created.

Business Benefits

Beaumont hospital currently lacks the capability of booking an appointment online and patient has to visit the hospital in person to book it. If this system of booking an appointment online is used by the hospital, process of booking an appointment can be streamlined with no one standing in queues. It will help the hospital management to keep a record of all the appointments and patients treated by the hospital which can be tracked and monitored by the management through reports which we have built. Moreover, the process of booking an offline appointment and maintaining piles of forms involves a lot of administrative tasks to be performed by the hospital which in turn involves extra expenditure for these tasks. By implementing this salesforce solution, Beaumont hospital can get rid of most of the administrative tasks and can save a lot of money.

Appendix

1. Registration Web page

```
<apex:page title="Register" Standardcontroller="User__c"

        showHeader="false" standardStylesheets="true">

  <apex:define name="body">




      <apex:form style="background-color:lightgrey;">

        <!-- Section to display the error mesages -->

        <table align="center" border="0"  bgcolor="#FFFFFF">

          <tr>

            <td><apex:image id="theImage1" value="{!$Resource.bpcheck}" alt="bpcheck" width="300"
height="225"/></td>
```

```
                <td><apex:image id="theImage2" value="{!$Resource.doctorchecklist}" alt="doctorchecklist" width="300" height="225"/></td>

            </tr>

            <tr><td colspan="2" width="605"> <apex:messages id="error" styleClass="errorMsg"

                    layout="table" style="margin-top:1em;"/></td></tr>

        </table>



        <table align="center" border="0" width="610" class="main"

              style="font-size:15px;color:Blue;"  bgcolor="#FFFFFF">

            <tr Align="Center"  bgcolor="#0066CC"><td colspan="3" style="font-size:18px;color:White;">&nbsp;&nbsp;&nbsp;&nbsp;<b>Register Here</b></td></tr>

            <tr><td align="right">Full Name</td><td></td><td><apex:inputText id="NameInput" value="{!User__c.Name__c}"/></td></tr>

            <tr><td align="right">Gender</td><td></td><td><apex:inputText id="GenderInput" value="{!User__c.Gender__c}"/></td></tr>

            <tr><td align="right">Email</td><td></td><td><apex:inputText id="EmailInput" value="{!User__c.Email_ID__c}"/></td></tr>

            <tr><td align="right">Date Of Birth(mm/dd/yyyy)</td><td></td><td><apex:inputText id="DOBInput" value="{!User__c.DOB__c}"/></td></tr>

            <tr><td align="right">Contact Number</td><td></td><td><apex:inputText id="ContactInput" value="{!User__c.Contact__c}"/></td></tr>

            <tr><td align="right">User Type</td><td></td><td><apex:inputText id="UserTypeInput" value="{!User__c.User_Type__c}" /></td></tr>

            <tr  bgcolor="#0066CC"><td align="center" colspan="3">&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;<apex:commandButton value="Save" action="{!save}" style="font-weight:bold;font-size:15px;color:darkblue;"/>&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;<apex:commandButton value="Cancel" onclick="top.history.go(-1);return false;" style="font-weight:bold;font-size:15px;color:darkblue;"/></td></tr>

        </table>




        <table align="center" border="0" bgcolor="#FFFFFF">

            <tr>

                <td><apex:image id="theImage3" value="{!$Resource.laboratory}" alt="laboratory" width="300" height="225"/></td>
```

```
          <td><apex:image id="theImage4" value="{!$Resource.mri}" alt="mri" width="300" height="225"/></td>
        </tr>
      </table>

    </apex:form>
  </apex:define>
</apex:page>
```

2. Appointment Web page

```
<apex:page title="Register" Standardcontroller="Appointment__c"
      showHeader="false" standardStylesheets="true">
  <apex:define name="body">


    <apex:form style="background-color:lightgrey;">
      <!-- Section to display the error mesages -->
      <table align="center" border="0"  bgcolor="#FFFFFF">
        <tr>
          <td><apex:image id="bap1" value="{!$Resource.bap1}" alt="bap1" width="300" height="225"/></td>
          <td><apex:image id="bap2" value="{!$Resource.bap2}" alt="bap2" width="300" height="225"/></td>
        </tr>
        <tr><td colspan="2" width="605"> <apex:messages id="error" styleClass="errorMsg"
                                layout="table" style="margin-top:1em;"/></td></tr>
      </table>

      <table align="center" border="0" width="610" class="main"
          style="font-size:15px;color:Blue;"  bgcolor="#FFFFFF">
        <tr Align="Center"  bgcolor="#0066CC"><td colspan="3" style="font-size:18px;color:White;">&nbsp;&nbsp;&nbsp;&nbsp;<b>Book Appointment Here</b></td></tr>
```

```
            <tr>
               <td align="right">Patient ID</td><td></td>
               <td><apex:inputtext value="{!Appointment__c.Patient_ID__c}" id="PID"/></td>
            </tr>
            <tr>
               <td align="right">Department</td><td></td>
               <td><apex:inputtext value="{!Appointment__c.Department_ID__c}" id="DID"/></td>
            </tr>
            <tr>
               <td align="right">Type of Appointment</td><td></td>
               <td><apex:inputtext value="{!Appointment__c.Appointment_Type__c}" id="TOA"/></td>
            </tr>
            <tr>
               <td align="right">Date [mm/dd/yyyy]</td><td></td>
               <td><apex:inputtext value="{!Appointment__c.Date__c}" id="DT"/></td>
            </tr>
            <tr>
               <td align="right" width="300">Time [hh:mm]</td><td></td>
               <td><apex:inputtext value="{!Appointment__c.Time__c}" id="TM"/></td>
            </tr>
            <tr>
               <td align="right">Upload Referral</td><td></td>
               <td>
                  <apex:inputFile value="File" fileName="F1" styleclass="form-control"/>
               </td>
            </tr>
            <tr  bgcolor="#0066CC"><td align="center" colspan="3">&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;<apex:commandButton value="Save" action="{!save}" style="font-weight:bold;font-size:15px;color:darkblue;"/>&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;<apex:commandButton value="Cancel" onclick="top.history.go(-1);return false;" style="font-weight:bold;font-size:15px;color:darkblue;"/></td></tr>
```

```
        </table>

      </apex:form>

   </apex:define>

</apex:page>
```

3. Appointment Retrieval Web Page

Web Page:

```
<apex:page showheader="false"  standardController="Appointment__c" extensions="AppointmentLookup"
standardStylesheets="true">

   <div align="Center"></div>

   <apex:form style="background-color:lightgrey;">

      <table align="center"  border="0"  bgcolor="#FFFFFF"  width="610">

         <tr align="center">

            <td colspan="3" height="30"> </td>

         </tr>

         <tr>

            <td><apex:image id="aptmt1" value="{!$Resource.aptmt1}" alt="aptmt1" width="300"
height="225"/></td>

            <td></td>

            <td><apex:image id="aptmt2" value="{!$Resource.aptmt2}" alt="aptmt2" width="300"
height="225"/></td>

         </tr>

         <tr align="center">

            <td colspan="3" height="10"> </td>

         </tr>

         <tr bgcolor="#0066CC" align="center">

            <td colspan="3" style="font-size:15px;color:White;"> Appointment - Enter your Patient ID </td>

         </tr>

         <tr>

            <td align="right" width="280"><apex:outputlabel value="Patient ID" for="Ref"/></td>

            <td></td>

            <td height="80">
```

```
          <apex:inputtext value="{!Patient_ID}" Rendered="True" label="Enter Patient ID" id="Ref"/>&nbsp;&nbsp;&nbsp;&nbsp;
          <BR/>
        </td>
      </tr>
      <tr>
        <div align="center">
          <td colspan="3" align="center" bgcolor="#0066CC" >
            <apex:commandbutton value="Retrieve Appointment" style="font-weight:bold;font-size:15px;color:darkblue;" action="{!getappointment}"/>
          </td>
        </div>
      </tr>

      <tr align="center">
        <td colspan="3" height="200"> </td>
      </tr>
    </table>

  </apex:form>

</apex:page>
```

Class:

```
public with sharing class AppointmentLookup {

  public string Patient_ID {get; set;}

  private ApexPages.StandardController stdCtrl;

  public AppointmentLookup(ApexPages.StandardController std)
  {
      stdCtrl=std;

  }


  public PageReference getappointment() {

      // controller.save();

      PageReference nextpage;

      nextpage = Page.showAppointment;

      nextpage.getParameters().put('Patient_ID',Patient_ID);

      return nextpage.setRedirect(True);

  }


}
```

4. Show Appointment Web Page

Web Page:

```
<apex:page Standardcontroller="Appointment__c" extensions="RetrieveAppointment" recordSetVar="notes"
showheader="False"   >

    <div align="Center">

        <table border="0" cellspacing="0" cellpadding="0">

        </table>

    </div>



    <apex:pageBlock >

        <apex:PageBlockSection >

            <apex:outputText value="Report of Appointments: {!Patient_ID}"  style="font-weight:bold;font-
size:15px;font-style:underline"/>

            <BR/><BR/>

            <b></b>

            <apex:datatable value="{!SLContExt}"  var="SLx" cellPadding="3" border="1" align="center" width="1100"
bgcolor="FFFFFF">

                <apex:column headerValue="Patient ID" value="{!SLx.Patient_ID__c}"/>

                <apex:column headerValue="Patient Name" value="{!SLx.Patient_ID__r.Name__c}"/>

                <apex:column headerValue="Date" value="{!SLx.Date__c}"/>
        <apex:column headerValue="Time" value="{!SLx.Time__c}"/>

        <apex:column headerValue="Department" value="{!SLx.Department_ID__r.Department_Name__c}"/>

        <apex:column headerValue="Doctor" value="Dr. {!SLx.Doctor_ID__r.Name__c}"/>

        <apex:column headerValue="Location" value="{!SLx.Room_ID__r.Location__c}"/>

        <apex:column headerValue="Room" value="{!SLx.Room_ID__r.Room_Name__c}"/>

        <apex:column headerValue="Status" value="{!SLx.Status__c}"/>

      </apex:datatable>

      <BR/>

    </apex:PageBlockSection>


  </apex:pageBlock>

  <div align="center">

    <apex:outputText value="End of Report" style="font-weight:bold;font-size:15px;font-style:underline"/>

    <BR/>-

  </div>


</apex:page>
```

Class:

```
public with sharing class RetrieveAppointment {

  public string Patient_ID { get; set; }


  private ApexPages.StandardController controller {get; set;}


  public RetrieveAppointment(ApexPages.StandardSetController controller)

  {

    Patient_ID = ApexPages.currentPage().getParameters().get('Patient_ID');

  }

  public ApexPages.StandardSetController setCon {


    get {
      setCon = new ApexPages.StandardSetController(Database.getQueryLocator(

        [Select
Patient_ID__c,Patient_ID__r.Name__c,Date__c,Time__c,Status__c,Department_ID__r.Department_Name__c,

        Doctor_ID__r.Name__c, Room_ID__r.Room_Name__c,Room_ID__r.Location__c, Appointment_Type__c
from Appointment__c ORDER BY Patient_ID__c,Date__c]));

      return setCon;

    }


    set;

  }


  public List<Appointment__c> getSLContExt() {


    return  (List<Appointment__c>) setCon.getRecords();;

  }


}
```

5. Prescription Retrieval Web Page

Web Page:

```
<apex:page showheader="false"  standardController="Prescription__c" extensions="PrescriptionLookup"
standardStylesheets="true">

  <div align="Center"></div>

  <apex:form style="background-color:lightgrey;">




      <table align="center"  border="0"  bgcolor="#FFFFFF"  width="610">

                <tr align="center">

          <td colspan="3" height="30"> </td>

        </tr>

        <tr>
```

```
          <td><apex:image id="meds1" value="{!$Resource.meds1}" alt="meds1" width="300"
height="225"/></td>

            <td></td>

          <td><apex:image id="meds2" value="{!$Resource.meds2}" alt="meds2" width="300"
height="225"/></td>

        </tr>

        <tr align="center">

          <td colspan="3" height="10"> </td>

        </tr>

        <tr  bgcolor="#0066CC" align="center">

          <td colspan="3" style="font-size:15px;color:White;"> Prescription History - Enter your Patient ID </td>

        </tr>

      <tr>

        <td align="right" width="280"><apex:outputlabel value="Patient ID" for="Ref"/></td>

        <td></td>

        <td height="80"><apex:inputtext value="{!Patient_ID}" Rendered="True" label="Enter Patient ID"
id="Ref"/>&nbsp;&nbsp;&nbsp;&nbsp;<BR/></td>

        </tr>

      <tr> <div align="center">

        <td colspan="3" align="center" bgcolor="#0066CC" > <apex:commandbutton value="Retrieve Prescription"
style="font-weight:bold;font-size:15px;color:darkblue;" action="{!getprescription}"/></td>

        </div></tr>



                <tr align="center">

          <td colspan="3" height="200"> </td>

        </tr>

      </table>



  </apex:form>

</apex:page>
```

Class:

```
public with sharing class PrescriptionLookup {


    public string Patient_ID {get; set;}


    private ApexPages.StandardController stdCtrl;


    public PrescriptionLookup(ApexPages.StandardController std)

    {

        stdCtrl=std;

    }


    public PageReference getprescription() {

        // controller.save();

        PageReference nextpage;

        nextpage = Page.showPrescription;

        nextpage.getParameters().put('Patient_ID',Patient_ID);

        return nextpage.setRedirect(True);

    }

}
```

6. Show Prescription Web Page

Web Page:

```
<apex:page Standardcontroller="Prescription__c" extensions="RetrievePrescription" recordSetVar="notes"
showheader="False"   >

    <div align="Center">

        <table border="0" cellspacing="0" cellpadding="0">


        </table>

    </div>
```

```
   <apex:pageBlock >
      <apex:PageBlockSection >
         <apex:outputText value="Report of Prescriptions: {!Patient_ID}"  style="font-weight:bold;font-size:15px;font-
style:underline"/>


         <BR/><BR/>


         <b></b>
         <apex:datatable value="{!SLContExt}"  var="SLx" cellPadding="3" border="1" align="center" width="1100"
bgcolor="FFFFFF">


            <apex:column headerValue="Patient ID" value="{!SLx.Patient_ID__c}"/>
            <apex:column headerValue="Patient Name" value="{!SLx.Patient_ID__r.Name__c}"/>
             <apex:column headerValue="Doctor Name" value="Dr. {!SLx.Doctor_ID__r.Name__c}"/>
                        <apex:column headerValue="Medicine Name"
value="{!SLx.Medicine_ID__r.Medicine_Name__c}"/>
            <apex:column headerValue="Medicine Potency" value="{!SLx.Medicine_ID__r.Potency__c}"/>
            <apex:column headerValue="Medicine Type" value="{!SLx.Medicine_ID__r.Type__c}"/>
            <apex:column headerValue="Dosage" value="{!SLx.Dosage__c}"/>
            <apex:column headerValue="Date" value="{!SLx.Date__c}"/>
         </apex:datatable>


         <BR/>





      </apex:PageBlockSection>


   </apex:pageBlock>
   <div align="center">
      <apex:outputText value="End of Report" style="font-weight:bold;font-size:15px;font-style:underline"/>
      <BR/>-

   </div>


</apex:page>
```

Class:

```
public with sharing class RetrievePrescription {

   public string Patient_ID { get; set; }


   private ApexPages.StandardController controller {get; set;}


   public RetrievePrescription(ApexPages.StandardSetController controller)

   {

      Patient_ID = ApexPages.currentPage().getParameters().get('Patient_ID');

   }

   public ApexPages.StandardSetController setCon {


      get {


         setCon = new ApexPages.StandardSetController(Database.getQueryLocator(


            [Select
Patient_ID__c,Patient_ID__r.Name__c,Doctor_ID__r.Name__c,Medicine_ID__r.Medicine_Name__c,Medicine_ID__r
.Potency__c,

             Medicine_ID__r.Type__c,Dosage__c,Date__c from Prescription__c WHERE Patient_ID__c = : Patient_ID
ORDER BY Patient_ID__c,Date__c]));

         return setCon;

      }


      set;

   }
```

```
    public List<Prescription__c> getSLContExt() {


        return (List<Prescription__c>) setCon.getRecords();
    }


}
```

Bibliography

Beaumont Hospital. (n.d.). *Beaumont Hospital*. Retrieved from http://www.beaumont.ie/index.jsp?p=101&n=378

Medicalcouncil.ie. (n.d.). *Medical Council Ireland*. Retrieved from https://www.medicalcouncil.ie/Public-Information/Check-the-Register/

Pharmac.govt.nz. (n.d.). *Pharmac.govt*. Retrieved from https://www.pharmac.govt.nz/about/2015/top-20/

Real Time Board. (n.d.). *Real Time Board*. Retrieved from https://realtimeboard.com/app/board/o9J_kyu8pBw=/

Smartsheet.com. (n.d.). *Smartsheet.com*. Retrieved from https://www.smartsheet.com/top-excel-project-plan-templates

Wikipedia. (n.d.). *Wikipedia*. Retrieved from https://en.wikipedia.org/wiki/List_of_allergens