

PHASE-4

Validation Rules

Object: Patient(Contact)

Name	Purpose	Formula
PrivateInsurancePolicy	If coverage_Type is private Insurance user needs to provide the Provider details and policy number	AND(ISPICKVAL(Coverage_Type__c, "Private Insurance"), OR(ISBLANK(Insurance_Provider__c), ISBLANK(Insurance_Policy_Number__c)))
GovtScheme	If the patient's coverage_type is a government scheme like CGHS, ECHS, etc, provide schemeID	AND(OR(ISPICKVAL(Coverage_Type__c, "CGHS"), ISPICKVAL(Coverage_Type__c, "ECHS"), ISPICKVAL(Coverage_Type__c, "ESIC"), ISPICKVAL(Coverage_Type__c, "State Health Scheme")), ISBLANK(Scheme_ID__c))
InvalidDOB	Date of Birth can't be in the future	Date_of_Birth >TODAY()
EmergencyContact	If the patient is a minor Emergency Contact is a must	(TODAY() - Date_of_Birth__c) / 365 < 18 && ISBLANK(EmergencyContact__c)

SETUP > OBJECT MANAGER

Patient

Details

Data Cloud Related List

Data Cloud Copy Field

Fields & Relationships

Page Layouts

Lightning Record Pages

Buttons, Links, and Actions

Compact Layouts

Field Sets

Object Limits

Validation Rules

4 Items, Sorted by Rule Name

RULE NAME

▲

ERROR LOCATION

ERROR MESSAGE

ACTIVE

MODIFIED BY

EmergencyContact

EmergencyContact

An emergency contact is required for patients under 18

✓

anushka bhatt, 9/23/2025, 8:38 PM

▼

GovtScheme

Scheme_ID

Please enter the Scheme ID for government health coverage patients.

✓

anushka bhatt, 9/19/2025, 11:21 AM

▼

InvalidDOB

Date_of_Birth

Date of Birth cannot be in the future.

✓

anushka bhatt, 9/21/2025, 1:23 PM

▼

PrivateInsurancePolicy

Insurance_Provider

"Please enter Insurance Provider and Policy Number when Coverage Type is Private Insurance.

✓

anushka bhatt, 9/19/2025, 11:19 AM

▼

New

Object: Doctor__c

Name	Purpose	Formula
EmailValidation	Email should be valid	NOT(REGEX(Email__c , "^[A-Za-z0-9._%+-]+@[A-Za-z0-9.-]+\.[A-Za-z]{2,}\$"))
ValidPhoneNo	Phone must contain 10 digits	NOT(REGEX(Phone__c , "^0-9]{10}\$"))

Setup

Home

Object Manager ▼

SETUP > OBJECT MANAGER

Doctor

Details

Fields & Relationships

Page Layouts

Lightning Record Pages

Buttons, Links, and Actions

Validation Rules

2 Items, Sorted by Rule Name

RULE NAME

▲

ERROR LOCATION

ERROR MESSAGE

ACTIVE

MODIFIED BY

EmailValidation

Email

Enter valid email

✓

anushka bhatt, 9/23/2025, 8:41 PM

▼

ValidPhoneNo

Availability_Status

Enter valid 10 digit number

✓

anushka bhatt, 9/23/2025, 9:04 PM

▼

New

Object: Appointment__c

Name	Purpose	Formula
AppointmentDate	It can't be of past	Appointment_DateTime__c < NOW()
DoctorOnLeave	An appointment can't be made for a date on which the Doctor is on leave	ISPICKVAL(Doctor__r.Availability_Status__c, "On Leave")

SETUP > OBJECT MANAGER

Appointment

Details

Fields & Relationships

Page Layouts

Lightning Record Pages

Buttons, Links, and Actions

Validation Rules

2 Items, Sorted by Rule Name

RULE NAME

AppointmentDate

ERROR LOCATION

Appointment_DateTime

ERROR MESSAGE

Appointment time cannot be in the past

ACTIVE

✓

MODIFIED BY

anushka bhatt, 9/23/2025, 8:42 PM

▼

DoctorOnLeave

Appointment_DateTime

Doctor is not available for this appointment

✓

anushka bhatt, 9/23/2025, 8:43 PM

▼

New

Approval Process:

System contains custom object Treatment__c , approval process if based on it .

Entry Criteria

If EstimatedCost__c for treatment > 0

Conditions:

If estimatedCost is less than 50,000 → Automatically approve

Else: need Hospital Manager approval

Step2:

If estimated cost is greater than 100000 → need Director Approval



SETUP

Approval Processes

Process Name	TreatmentCost	Active	✓
Unique Name	TreatmentCost	Next Automated Approver Determined By	Manager of Record Submitter
Description			
Entry Criteria	Treatment: EstimatedCost GREATER THAN 0		
Record Editability	Administrator ONLY	Allow Submitters to Recall Approval Requests	✓
Approval Assignment Email Template	<u>CostApproval</u>		
Initial Submitters	Appointment Owner, <u>User: Henry William</u> , <u>User: Kevin Joshep</u> , <u>User: Sarah Roger</u> , <u>User: Tony Stark</u>		
Created By	<u>anushka bhatt</u> , 9/24/2025, 9:42 AM	Modified By	<u>anushka bhatt</u> , 9/24/2025, 9:56 AM

Initial Submission Actions

Add Existing

Add New ▼

Action	Type	Description
	Record Lock	Lock the record from being edited

Approval Steps

Action	Step Number	Name	Description	Criteria	Assigned Approver	Reject Behavior
Show Actions Edit	1	ManagerApproval		Treatment: EstimatedCost GREATER THAN 50000 , else Approve	<u>User:Henry William</u>	Final Rejection
Show Actions Edit	2	DirectorApproval		Treatment: EstimatedCost GREATER THAN 100000	Approval based on first response <u>User: anushka bhatt</u> , <u>Cassie Thompson</u>	Go Back 1 Step

Care Team can submit the approval request

Treatment
TRT-0001

Related Details

Treatment Name
TRT-0001

Visit Date
25/09/2025, 12:00 pm

Diagnosis
need surgery

Prescription

Followup_Required
☐

Appointment
[APT-0002](#)

Doctor
[DOC-0003](#)

EstimatedCost
\$9,000.00

Created By
 Sarah Roger, 24/09/2025, 11:14 am

Last Modified By
 Sarah Roger, 24/09/2025, 11:14 am

Treatment
TRT-0001

Related Details

Notes & Attachments (0) [Upload Files](#)

[Upload Files](#)
Or drop files

Approval History (2) [Recall](#)

Step Name	Date	Status	Assigned To
ManagerApproval	24/09/2025, 11:18 am	Pending	Henry William
Approval Request S...	24/09/2025, 11:18 am	Submitted	Sarah Roger

[View All](#)

If the manager agree to approve he /she approves

If Cost is > 100000 goes to the director for approval

The screenshot displays a web application interface for a medical treatment record. At the top, a header bar shows a green icon with a white 'E' and the text 'Treatment TRT-0003'. Below this, a tabbed interface has 'Related' and 'Details' tabs, with 'Details' being the active tab. The 'Details' section contains several rows of data, each with a label on the left and a value on the right, followed by a small edit icon (a pencil). The rows are: 'Treatment Name' with value 'TRT-0003'; 'Visit_Date' with value '17/09/2025, 12:00 pm'; 'Diagnosis' with an empty field; 'Prescription' with an empty field; 'Followup_Required' with a checkbox and an empty field; 'Appointment' with a blue link 'APT-0003'; 'Doctor' with a blue link 'DOC-0004'; 'EstimatedCost' with value '1,55,000.00'; and 'Created By' with a user icon and text 'Sarah Roger, 24/09/2025, 12:11 pm'.

Related	Details
Treatment Name	TRT-0003
Visit_Date	17/09/2025, 12:00 pm
Diagnosis	
Prescription	
Followup_Required	<input type="checkbox"/>
Appointment	APT-0003
Doctor	DOC-0004
EstimatedCost	1,55,000.00
Created By	Sarah Roger, 24/09/2025, 12:11 pm

Agent using agentforce

The goal of this Agent is to streamline patient registration and doctor appointment booking in the **CuraForce App**.

- Patients or staff can interact with the chatbot to **create a new patient record**.
- The chatbot will ask for basic patient details (name, age, gender, health issue).
- Based on the **health issue description**, the bot uses an **Apex-powered NLP-like keyword matcher** to **suggest a suitable doctor specialty** (e.g., Cardiologist for chest pain, Dermatologist for skin issues, General Physician for other conditions).
- The chatbot then shows a list of **available doctors** in that specialty.
- If the patient confirms, the chatbot books an **appointment** (DoctorPatient record) with the selected doctor.

This automation reduces manual data entry for staff and provides patients with quick doctor recommendations.

Current Implementation (Prototype Stage)

Apex Class for Doctor Suggestion

Implemented **HealthIssueDoctorService** class which:

- Reads patient health issue text.
- Maps keywords → suggested specialty.
Queries doctors with **Availability__c = 'Available'**.
- Returns a formatted list of doctor names & IDs.

```

public with sharing class HealthIssueDoctorService {
    // Invocable Request (input)
    public class Request {
        @InvocableVariable(required=true)
        public String healthIssue;
        @InvocableVariable
        public Integer maxResults; // number of doctors return
    }

    public class Response {
        @InvocableVariable
        public String suggestedSpecialty;
        // doctorListCsv is "Id|Doctor Name;Id|Doctor Name;..."
        @InvocableVariable
        public String doctorListCsv;
    }

    @InvocableMethod(label='Suggest Specialty & Doctors' description='Return suggested specialty and small list of doctors based on health issue text')
    public static List<Response> suggest(List<Request> requests) {
        List<Response> outputs = new List<Response>();
        for(Request req : requests) {
            String issue = (req.healthIssue == null) ? '' : req.healthIssue.toLowerCase();
            Integer maxR = (req.maxResults == null || req.maxResults <= 0) ? 5 : req.maxResults;
            String spec = 'General Physician';

            // keyword rules
            if(issue.contains('chest') || issue.contains('heart') || issue.contains('angina') || issue.contains('palpiti')) {
                spec = 'Cardiologist';
            } else if(issue.contains('skin') || issue.contains('rash') || issue.contains('itch') || issue.contains('eczema') || issue.contains('psoriasis')) {
                spec = 'Dermatologist';
            } else {
                spec = 'General Physician';
            }

            // Querying doctors for that specialty
            List<Doctor__c> doctors = new List<Doctor__c>();
            try {
                doctors = [
                    SELECT Id, Name, Specialty__c, Availability_Status__c
                    FROM Doctor__c
                    WHERE Specialty__c = :spec
                    AND Availability_Status__c = 'Available'
                    LIMIT :maxR
                ];
            } catch (Exception e) {
                // if the Doctor__c object/fields not matched, fallback to empty list
                doctors = new List<Doctor__c>();
            }

            String csv = '';
            for(Doctor__c d : doctors) {
                if(csv != '') csv += ';';
                csv += String.valueOf(d.Id) + '|' + (d.Name == null ? '' : d.Name);
            }

            Response out = new Response();
            out.suggestedSpecialty = spec;
            out.doctorListCsv = csv;
            outputs.add(out);
        }
        return outputs;
    }
}

```

Tested apex class using executing this in anonymous window

```

List<HealthIssueDoctorService.Request> reqs = new List<HealthIssueDoctorService.Request>();
HealthIssueDoctorService.Request r = new HealthIssueDoctorService.Request();
r.healthIssue = 'I have chest pain and difficulty breathing';
r.maxResults = 3;
reqs.add(r);

List<HealthIssueDoctorService.Response> res = HealthIssueDoctorService.suggest(reqs);
System.debug(JSON.serializePretty(res));

```

Result return Doc 004 and 007 are cardiologist that are available.

Execution Log

Timestamp	Event	Details
13:19:04.041	USER_DEBUG	[{"DEBUGID": {
13:19:04.000	USER_DEBUG	"suggestSpecialty": "Cardiologist",
13:19:04.000	USER_DEBUG	"selectLastCase": "a07g.000008KQmQAP(DOC-0001.a07g.000008KQmQAP(DOC-0004.a07g.000008KQmQAP(DOC-0007"
13:19:04.000	USER_DEBUG	}]

CuraForce

ClinicSchedulePagePatientsTasksDoctorsAppointmentsTreatmentsFeedbacksMore

Doctors

All

Q Search this list...

13 Items • Sorted by Doctor ID • Updated a few seconds ago

	Doctor ID ↑	Full Name	Specialty	Availability Status	
1	DOC-0001	Joey Macow	Cardiologist	Available	
2	DOC-0002	Ragini Mishra	Dermatologist	Available	
3	DOC-0003	R K Ghosh	General Physician	Available	
4	DOC-0004	J P Narayan	Cardiologist	Available	
5	DOC-0005	Dr.Rishi Parmar	General Physician	On Leave	
6	DOC-0006	Dr.Shweta Singh	Dermatologist	Booked	
7	DOC-0007	Dr Alia Sharma	Cardiologist	Available	
8	DOC-0009	Rodson	Cardiologist		
9	DOC-0010	Yin Su	Cardiologist		
10	DOC-0011	Chole	Dermatologist	On Leave	
11	DOC-0012	Jin Su	Dermatologist		
12	DOC-0014	Kim Su	Dermatologist	Available	
13	DOC-0015	Alia Cruise	Cardiologist		

Steps to create Agent

I already have all my agent permissions active so I directly head to creating new agent.

Setup

Home

Object Manager

Q agent

Feature Settings

Commerce

Commerce Agentforce Settings

Einstein

Agentforce Data Library

Einstein Generative AI

Agentforce Studio

Agent Analytics (Beta)

Agentforce Agents

Agentforce Assets

Testing Center

Development

Agentforce for Developers

Didn't find what you're looking for?

Try using Global Search.

SETUP > AGENTFORCE STUDIO

Agentforce Agents

Agentforce

On

+ New Agent

Get to Know Agentforce Agents

Hit the Trail with Agents

Follow along with an admin as they explore what makes agents tick and launch an agent for their business.

Go to Trailhead

Customize Agents with Actions

Actions are how an agent gets things done. Learn how to create agent actions tailored to your business needs in the Salesforce Admins Blog.

Go to Blog

Succeed with Documentation

Salesforce Help has all the info you need to customize, test, and activate an agent for your use cases.

Go to Help

Enable the Agentforce (Default) Agent

Help your employees accomplish key business tasks in Salesforce and Slack with the default AI assistant for Salesforce CRM.

On

Select the Model Provider for Agents

SETUP > AGENT DETAILS

Healthcare Service Agent

Version 1

Open in Builder

API Name

Type

Active

Healthcare_Service_Agent

Service Agent

Details

Topics

System Messages

Language Settings

Connections

Name

Healthcare Service Agent

Created On

September 25, 2025 at 11:51 PM

API Name

Healthcare_Service_Agent

Created By

anushka bhatt

Description

You are an AI Agent whose job is to help patients with registration, health issue analysis, doctor recommendations, and appointment scheduling for CuraForce.

Last Modified By

anushka bhatt

Role

Assists patients with registration, health issue analysis, doctor recommendations, and appointment scheduling.

Company

CuraForce

Agent User

EinsteinServiceAgent User (healthcare_service_agent@)

Keep a record of conversations with enhanced event logs to review agent behavior

Set the topics for agent so that it give relevant answers

The screenshot shows the Agentforce Builder interface for a "Healthcare Service Agent". The "Topics" panel on the left lists four topics: "Appointment Scheduling Support", "Doctor Recommendation System", "Health Records Retrieval", and "Patient Registration Assistance". The "Conversation Preview" panel on the right shows a chat interface with a cartoon robot character and the text "Let's chat! Hi, I'm an AI assistant. How can I help you?".

This detailed screenshot shows the "Reasoning" process of the agent. The "Topics" panel on the left is the same as in the previous image. The central panel shows the "Reasoning" process for a user prompt "no allergies". The "Reasoning" process includes "Reasoning: Topic Selection" (0.36 sec) and "Reasoning: Action Selection" (1.44 sec). The "Topic Selected" is "Patient Registration Assistance". The "Reasoning" section shows a "GROUNDING" summary: "The response accurately summarizes the information provided by the user in the conversation history and aligns with the context of assisting with patient registration." The "Agent Response" section shows the final output: "Thanks for confirming! Here's the next step for your registration process." The "Conversation Preview" panel on the right shows the chat history, including the user's input "no allergies" and the agent's response "I'm sorry to hear that you're feeling unwell. I can help recommend a doctor for you. Could you let me know your location and any preferences you have for the doctor (e.g., gender, language, or specialty)?".

The screenshot displays a healthcare agent interface with three main panels:

- Topics Panel:** Contains a search bar and a list of topics: Appointment Scheduling Support, Doctor Recommendation System, Health Records Retrieval, and Patient Registration Assistance.
- Reasoning Panel:** Shows the internal logic of the agent. It starts with a "User Prompt" (yes), followed by "Reasoning: Action Selection" (0.21 sec), then "Action Launched: Finalize Reservation (FinalizeRe..." (0.27 sec). The input is a JSON object with a dateTime, and the output is a JSON object with an outcomeMessage stating "No reservation was made because no customer account was found. Try...". This is followed by another "Reasoning: Action Selection" (0.79 sec) and a "Reasoning" block with a grounded response.
- Conversation Preview Panel:** Shows the chat history. The user asks for registration. The agent provides contact information, medical concerns, and address. The user confirms no allergies. The agent thanks the user and provides final registration information, including full name, date of birth, contact information, medical concerns, address, and allergies. The user asks to find a doctor, and the agent suggests visiting a nearby hospital or clinic. The user asks to register now, and the agent responds that it can only guide through the registration process.

The agent is asking me as a patient to give information for required fields so that can create my patient account or check if i am already there. But I am working on functionality to book appointment and suggesting doctor. Right now the agent is under development.

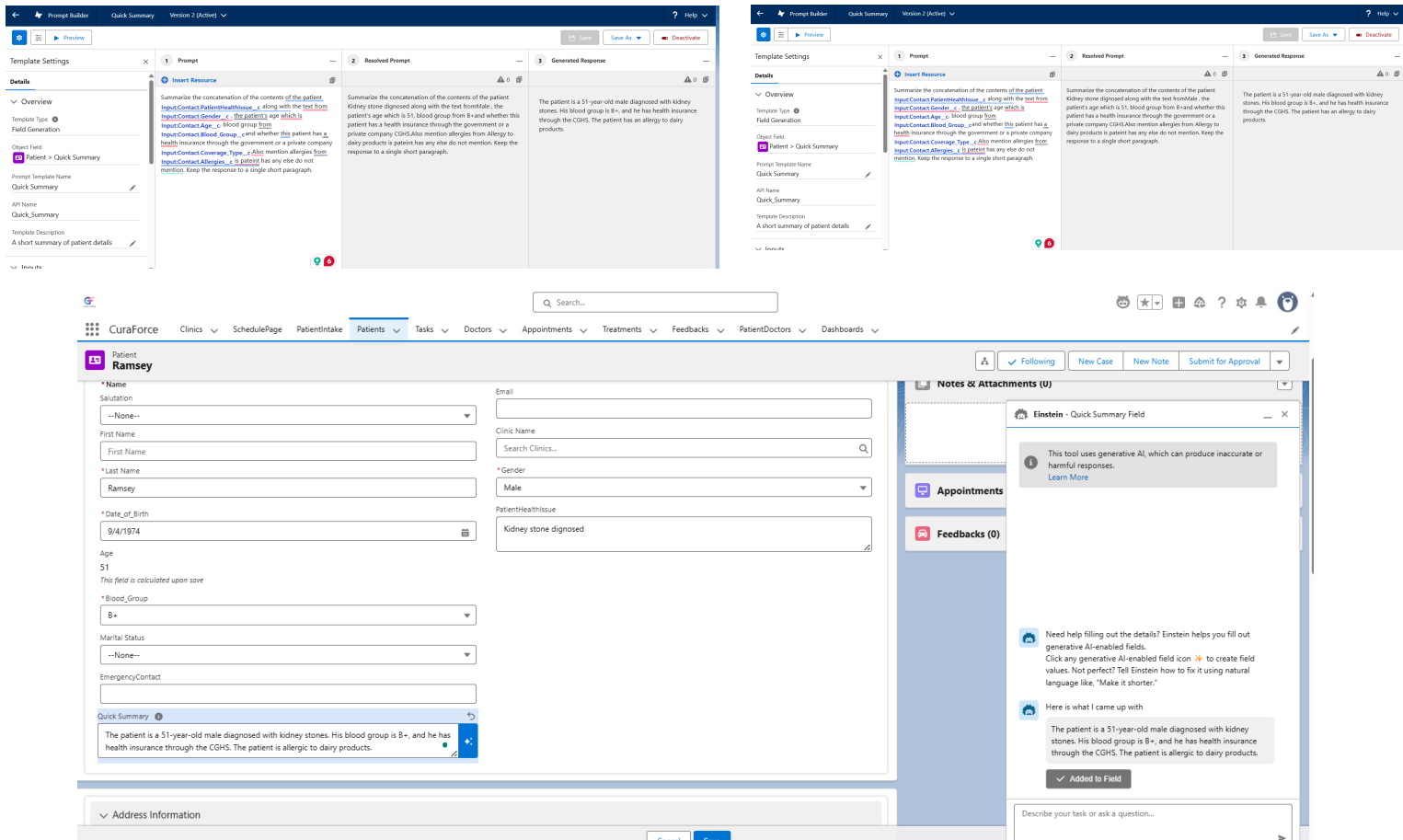
Prompt Template

I used the "Field Generation" prompt template to create a quick summary for patient using AI.

Steps to create prompt template are

- Setup → in the quick find box head to prompt builder → New Prompt Template → select type, Object, field of that object that you want to generate with AI, give suitable name and description.
- In the new window type the prompt / instruction you want what type of summary and all.
- Preview various models and generated responses for each of them.
- And Select the one that suits you click save and active.

Model used OpenAI GPT 4 Omni Mini, 0 toxicity detected.



Screen Flow

Instead of agent, for the time being I implement **screen flow** for the same cause. I have utilized the apex class as apex action in the screen flow.

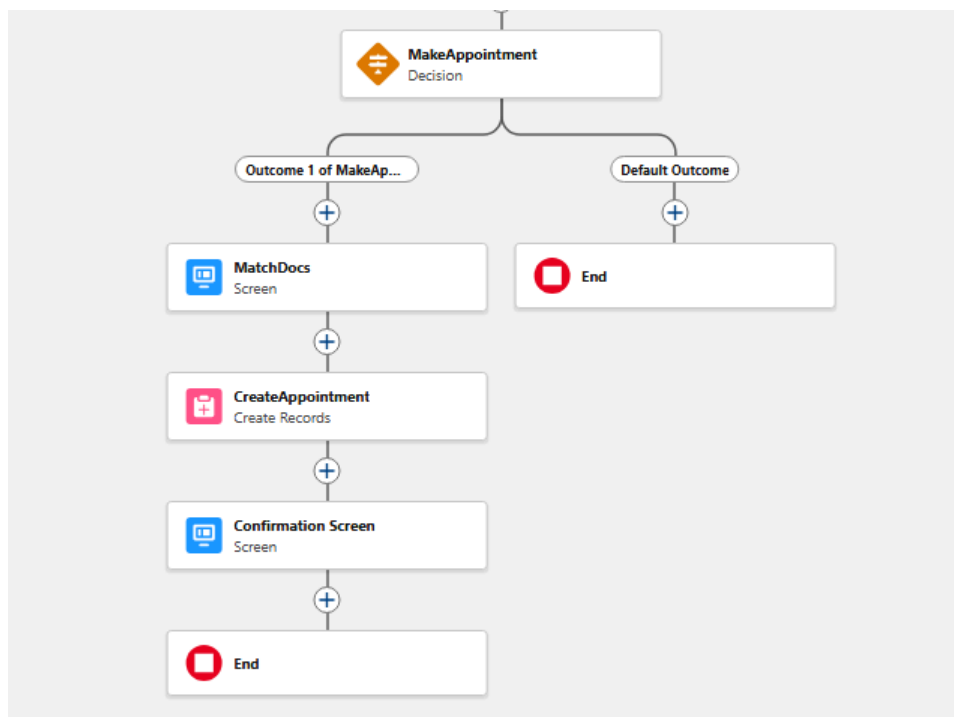
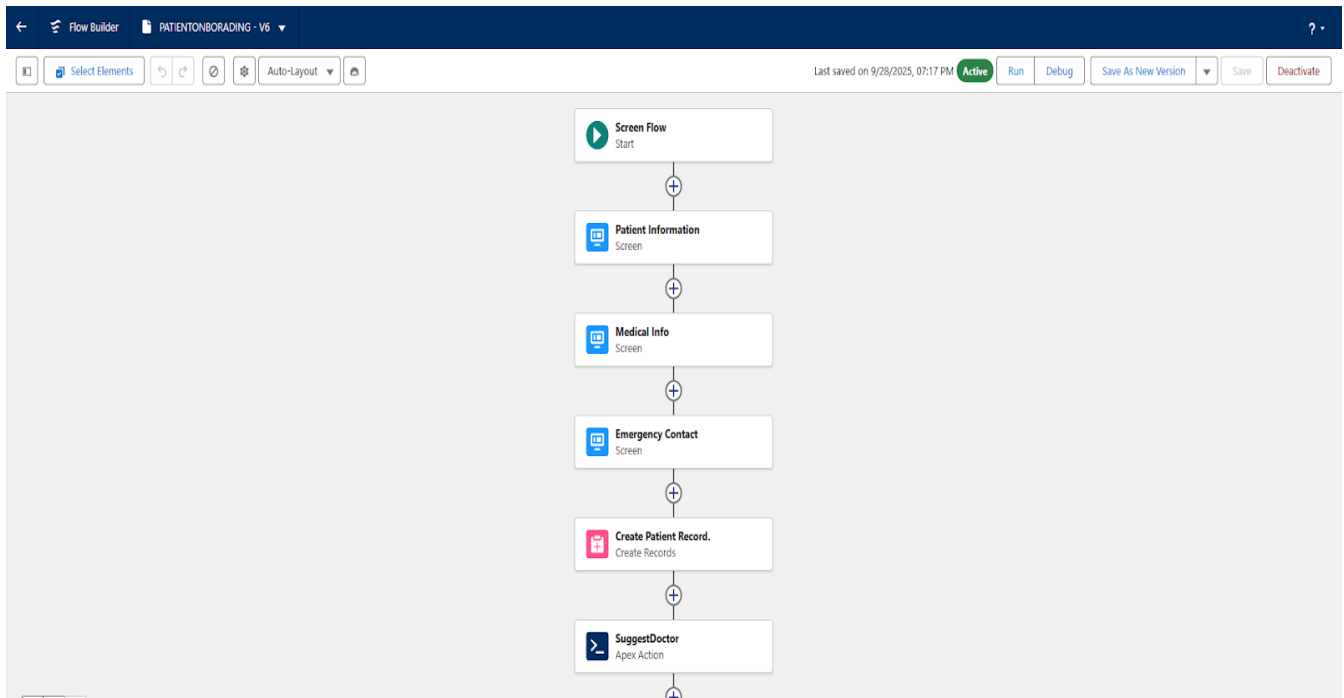
This flow starts by **collecting the required information of the patient** like name, gender, blood group, any allergies, and most importantly, current health issues, based on which later the patient gets suggested a specialty and a list of available doctors with that specialty.

The **create patient record component** simply maps the flow variables to the actual patient record **field** and creates a patient record.

Using the same **HealthIssueDoctorService** apex class calling it using apex action component, **patient gets the suggested doctor and their speciality list**. Then, asked does he/she wants to book the appointment.

If it says yes in the next screen, it gets to choose one of the suggested doctors and preferred date and time, if that date and time obeys other validations we previously created, like preventing Double

booking, and if the doctor is available on that day. The booking gets confirmed, and the database gets updated.



PATIENTONBORADING

Fill this patient form

Name

First Name

Shrey

Last Name

Sharma

* DOB

Jul 15, 1999

* Gender

☒ Male

☐ Female

☐ Other

* Marital Status

Single

Address

Country

India

Street

PATIENTONBORADING

Any Past medical history

Allergies

allergie due to soya products

* BloodGroup

O+

PatientHealthIssue

Chest pain

Previous

Next

PATIENTONBORADING

Based on the patient health issue, we suggest consulting:
a07gl.000000EPQMmQAIPDOC-0001;a07gl.000000Epc9QA8IDOC-0004;a07gl.000000EgTPIQA3IDOC-0007 with specialised in Cardiologist

Make Appointment

☐ Yes

☐ No

Previous

Next

PATIENTONBORADING

Suggested Doctors

--None--

Preffred Date and Time

Previous

Next

PATIENTONBORADING

Yay! Your appointment is confirmed with doctor J P Narayan.
Time and date of your appointment is 9/30/2025, 3:00 PM.

Previous

Finish

This flow is embedded in an app page named “**PatientIntake**” in the curaForce app.

CuraForce Clinics SchedulePage PatientIntake Patients Tasks Doctors Appointments Treatments Feedbacks PatientDoctors Dashboards

PatientIntake

PATIENTONBORADING

Fill this patient form

Name

First Name

First Name

Last Name

Last Name

* DOB

* Gender

☐ Male

☐ Female

☐ Other

* Marital Status

--None--

Address

Country

--None--