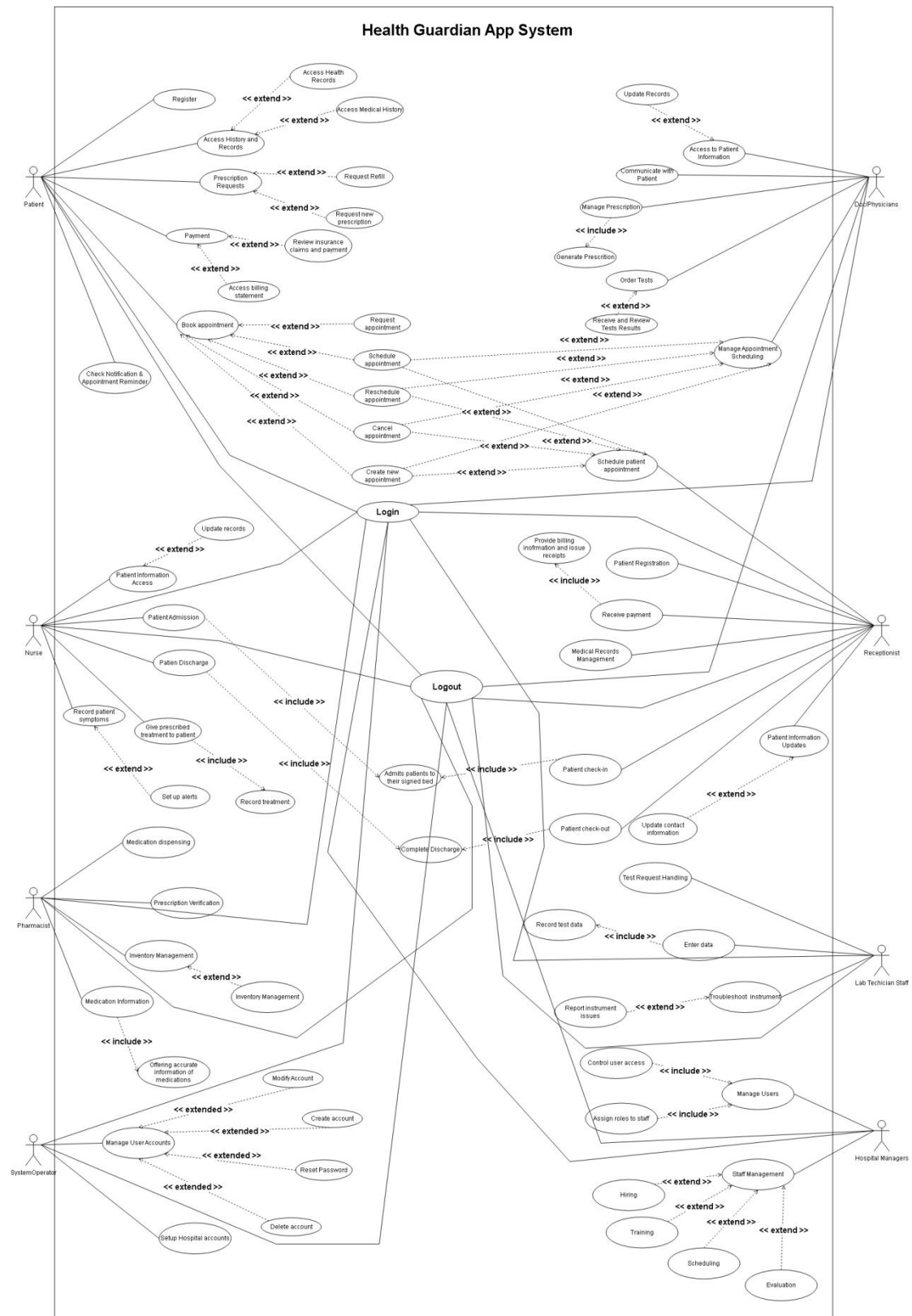
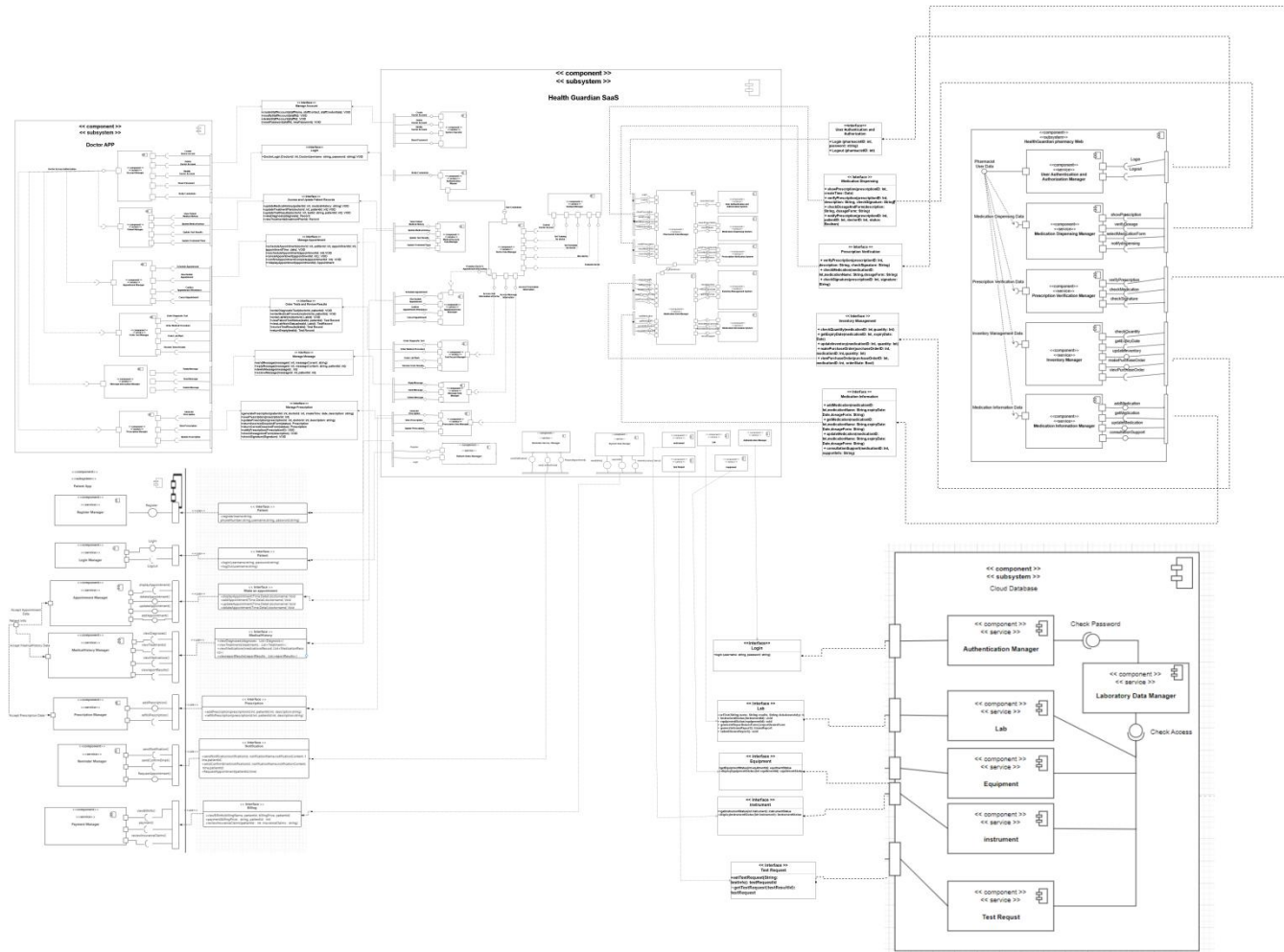


## **Task 1: Software Modelling and Specification**

### **(a) Use Case Diagram**



**(a) Component Diagram**



## Interfaces and component table explanation

### 2.1 Doctor

#### 2.1.1 Component Table

Component	System Operator
Description	This is a micro service system running on the cloud system and
Type	Microservice
Provided Interface	Create Doctor Account Delete Doctor Account Modify Doctor Account Reset Staff Account Password
Required Interface	N/A

Component	Authentication Manager
Description	This is a manager system on health guardian cloud system , which is used for authenticate the identity of the staff member , in this case, doctor.
Type	Microservice
Provided Interface	Verify Credentials
Required Interface	Give Credentials

Component	Patient Data Manager
Description	This is a data management system running on Health Guardian System within the cloud database for patient medical history information storage.
Type	Microservice
Provided Interface	View Patient Medical History Update Medical History Update Test Results Update Treatment Plans
Required Interface	N/A

Component	Doctor Data Manager
Description	It is a microservice system operating on cloud system of health guardian specified for doctor data management which provides the essential information and data, for instance, providing the appointment information when the doctor app is required.
Type	Microservice
Provided Interface	Give Credentials, Provides Appointment Information, Access Test Information of Doctor. Access Message Information, Access Prescription generated by doctor
Required Interface	Control Doctor Access, Set Training for Doctor, Hire Doctor, Evaluate Doctor

Component	Hospital Manager
Description	The is the managing system operating on the cloud service of Health Guardian to manage the specific.
Type	Microservice
Provided Interface	Control Doctor Access, Set Training for Doctor, Hire Doctor, Evaluate Doctor
Required Interface	N/A

Component	Appointment Data Manager
Description	It is a data managing directly operating on cloud server of Health Guardian system for managing specific information of appointments related to doctor.
Type	Microservice
Provided Interface	Schedule Appointment Reschedule Appointment Confirm Appointment Attendance Cancel Appointment
Required Interface	Provides Doctor's Appointment Information

Component	Test Record Manager
Description	It is a data record system for recording and storing the information of the test doctor had requested either to lab or patient. And also contains the returned information of test, for instance, report, issues etc
Type	Microservice
Provided Interface	Order Diagnostic Test Order Medical Procedure Order Lab Work Receive Tests Results
Required Interface	Access Test Information of Doctor

Component	Message Data Manager
Description	It is a data manage system that operates on the server of Health Guardian System, which is used for storing messaging data of between doctors and patients. It contains detail information of communication records including consultants, communications etc.
Type	Microservice
Provided Interface	Reply Message Send Message Delete Message Receive Message
Required Interface	Access Message Information

Component	Prescription Data Manager
Description	This is a system operates on the Health Guardian Cloud server that stores the information of the prescription which includes the specifications generated by doctors and details involved within the prescription, for instance, the dosage, multiple drugs, drug procedure.
Type	Microservice
Provided Interface	Generate Prescription View Prescription Update Prescription

Required Interface	Access Prescription Information

Component	Account Manager
Description	Account Manager is a Doctor APP subsystem operates on cloud system to manage the account of doctor. It includes the account information of doctors, modification operation.
Type	Microservice
Provided Interface	Doctor Access Authorization
Required Interface	Create Doctor Account Delete Doctor Account Modify Doctor Account Verify Credentials Reset Password

Component	Patient Manager
Description	This is the doctor manage system of doctor app, operating on the cloud system. For managing the accessing the patient information such as patient medical history only.
Type	Microservice
Provided Interface	N/A
Required Interface	Doctor Account Authorization View Patient Medical History Update Medical History Update Test Results Update Treatment Plans

Component	Appointment Manager
Description	Appointment Manage System is a subsystem of doctor APP which operates on cloud system that manages the access of appointment information of doctor.
Type	Microservice
Provided Interface	N/A
Required Interface	Schedule Appointment Reschedule Appointment Confirm Appointment Attendance Cancel Appointment

Component	Order Test Manager
Description	Test Manager is a service of Doctor APP, and it manages the test that doctor had requested, which includes information such as test type, test content, returned reports.
Type	Microservice
Provided Interface	N/A
Required Interface	Order Diagnostic Test Order Medical Procedure Order Lab Work Receive Tests Results Doctor Access Authorization

Component	Message Information Manager
Description	The is the manage system of Doctor APP that operates on the cloud server contain the message between doctors and patients. It controls the access of message information.
Type	Microservice
Provided Interface	N/A
Required Interface	Reply Message Send Message Delete Message Receive Message

Component	Prescription Manager
Description	Prescription Manager system is the service within the Doctor APP that operates on cloud server which manages the prescription related information
Type	Microservice
Provided Interface	N/A
Required Interface	Generate Prescription View Prescription Update Prescription



### 2.1.2 Interface Table

Interface Name	Manage Account
Interface Provider	Health Guardian SaaS
	Operation: createStaffAccount
	<p>Request Parameters: staffName, staffContact, staffCredentials</p> <p>Response Parameters: StaffID, Success Status, Error Message</p> <p>Function: Create the account for staff with basic information including the doctors basic information and a unique staff ID given by system</p>
	Operation: modifyStaffAccount
	<p>Request Parameters: StaffId</p> <p>Response Parameters: StaffId, If Success, If Failed</p> <p>Function: Modify the account for, such as the basic information, contact information etc.</p>
	Operation: deleteStaffAccount
	<p>Request Parameters: StaffId</p> <p>Response Parameters: StaffId, If deleted Successfully, delete failed, incorrect StaffId</p> <p>Function: Delete the account of the staff once the staff had resigned or left the official job, or the staff had violated regulations of the hospital</p>
	Operation: resetPassword
	<p>Request Parameters: StaffId, new password</p> <p>Response Parameters: Success Message, Failed Message</p> <p>Function: Reset the password of the doctor if the doctor had issue with their accounts</p>

Interface Name	Login
Interface Provider	Health Guardian SaaS
	Operation: Login
	<p>Request Parameters: DoctorId, DoctorUserName, password</p> <p>Response Parameters: Doctor Id, Success Message, failed message, Error In Credentials</p> <p>Function: Doctor is able to login their accounts after the authentication manager on the cloud system verified the credentials</p>

Interface Name	Access and Update Patient Records
Interface Provider	Health Guardian SaaS
	Operation: updateMedicalHistory
	<p>Request Parameters: patientId, medicalHistory</p> <p>Response Parameters: Successfully displayed, Error in Displaying the Message, Empty Message,</p> <p>Function: Updates the patient medical history by first accessing the patient basic information then modify the medical history, such as diagnose history, medical procedure history, allergy history</p>
	Operation: updateTreatmentPlan
	<p>Request Parameters:</p> <p>Response Parameters: Patient Information Management</p> <p>Function: Update the treatment plan for the patient on patients' medical records</p>
	Operation: updateTestResults
	<p>Request Parameters: doctorId, patientId, testId</p> <p>Response Parameters: Patient Information Management</p> <p>Function: Update the test results of patient, such as the results after procedure, diagnosis</p>
	Operation: viewDiagnosis

	<p>Request Parameters: diagnosis</p> <p>Response Parameters: If success, If Failed, error message for displaying the results</p> <p>Function: View the diagnosis details of patient which the test was requested by doctors</p>
	Operation: viewTreatment
	<p>Request Parameters: treatmentId</p> <p>Response Parameters: Success Displayed Message, Error Displaying Message, Treatment Content Empty</p> <p>Function: View the treatment content that doctor had assigned or updated to the corresponding patient</p>

Interface Name	Manage Appointment
Interface Provider	Health Guardian SaaS
	Operation: scheduleAppointment
	<p>Request Parameters: doctorId, patientId, appointmentId appointmentTime</p> <p>Response Parameters: If Successful, If Failed, Error Appointment</p> <p>Function: Allows doctor to schedule an appointment with patient</p>
	Operation: rescheduleAppointment
	<p>Request Parameters: appointmentId, appointmentTime</p> <p>Response Parameters: If Successful Rescheduled, If Failed Reschedule, Error Appointment Rescheduling</p> <p>Function: Allows doctor to reschedule the appointment which had been set earlier according to the their time table</p>
	Operation: cancelAppointment
	Request Parameters: appointmentId

	<p>Response Parameters: If Successful, If Failed, Error Appointment</p> <p>Function: Allows doctor to cancel the appointment set earlier</p>
	Operation: confirmAppointment
	<p>Request Parameters: appointmentId</p> <p>Response Parameters: If Successful, If Failed, Error Appointment</p> <p>Function: Allows the doctor to confirm the attendance of the patient and therefore complete the current appointment</p>
	Operation: displayAppointment
	<p>Request Parameters: appointmentId</p> <p>Response Parameters: If Successful, If Failed, Error appointment, Empty Appointment</p> <p>Function: Allows the doctor to check the appointment that they had made</p>

Interface Name	Order Tests and Review Results
Interface Provider	Health Guardian SaaS
	Operation: orderDiagnosticTest
	<p>Request Parameters: doctorId, patientId</p> <p>Response Parameters: If Successful, If Failed, Error Test Message</p> <p>Function: Doctors order the test contents and type to specific patients</p>
	Operation: orderMedicalProcedure
	<p>Request Parameters: doctorId, patientId</p> <p>Response Parameters: If Successful, If Failed, Error Test Message</p> <p>Function: Doctors order the corresponding medical procedure for patient to take and undergo</p>

	Operation: orderLabWork
	<p>Request Parameters: doctorId, LabId</p> <p>Response Parameters: If Successful, If Failed, Error Test or Work Message, Lab Not Available</p> <p>Function: Doctor is able to order specific laboratory works for the lab</p>
	Operation: viewPatient TestStatus
	<p>Request Parameters: testId, patientId</p> <p>Response Parameters: If Successful, If Failed, Error Test Message, Test Expired Message</p> <p>Function: Allows doctors to check whether the test they gave are still available and active</p>
	Operation: viewLabWork
	<p>Request Parameters: testId, LabId</p> <p>Response Parameters: If Successful, If Failed, Error Work Message</p> <p>Function: Doctors could check the status and details of the work request they sent, which keep track of the test or work.</p>
	Operation: receiveTestResults
	<p>Request Parameters: testId</p> <p>Response Parameters: If Successful, If empty, Error Message of the results</p> <p>Function: Doctors are able to decide whether they are going to receive the message into their database.</p>

Interface Name	Manage Message
Interface Provider	Health Guardian SaaS
	Operation: sendMessage
	Request Parameters: messageId messageContent

	<p>Response Parameters: If Successful, If Failed, Error Message</p> <p>Function: Doctors can send the message to the patient to communicate and make consultant</p>
	Operation: replyMessage
	<p>Request Parameters: messageId, messageContent, patientId</p> <p>Response Parameters: If Successful, If Failed, Error Message</p> <p>Function: Doctors can reply to the patient when patient sends them message</p>
	Operation: deleteMessage
	<p>Request Parameters: messageId</p> <p>Response Parameters: If Successful, If Failed, Message Non-exist</p> <p>Function: Doctors can delete the message withing the server when the online conversation is officially ended.</p>
	Operation: receiveMessage
	<p>Request Parameters: patientId, messageId</p> <p>Response Parameters: If Successful, If Failed, Error Message</p> <p>Function: Doctors can receive the patients' messages once patients' sent the message</p>

Interface Name	Manage Prescription
Interface Provider	Health Guardian SaaS
	Operation: generatePrescription
	Request Parameters: patientId, doctorId, createTime, description

	<p>Response Parameters: If Successful, If Failed, Error Prescription Message, Signature Missing</p> <p>Function: Doctors can generate electronic prescription for patients with signature and other contents within</p>
	Operation: viewPrescription
	<p>Request Parameters: prescriptionId</p> <p>Response Parameters: If Successful, If Failed, Error Prescription Message, Prescription Verification Failed</p> <p>Function: Doctors can check the prescription they generated and track the status</p>
	Operation: updatePrescription
	<p>Request Parameters: doctorId, prescriptionId, description</p> <p>Response Parameters: If Successful, If Failed, Error Message when updating</p> <p>Function: Doctors can update the prescription of the earlier generated prescription either to correct information or keep the information up to date</p>
	Operation: checkSignature
	<p>Request Parameters: Signature</p> <p>Response Parameters: Missing Signature, Error in Signature, Mismatch in Signature</p>

## 2.2 Pharmacist

### 2.2.1 Component Table

Component	User Authentication and Authorization Manager
Description	Manages user login sessions and permissions for accessing different components of the system.
Type	HealthGuardian Pharmacy Web
Required Interfaces:	Login Logout
Provided Interfaces:	Pharmacist User Data(dataAccess)

Component	Medication Dispensing Manager
Description	Handles the dispensing of medication, including verification, selection, and notification processes.
Type	HealthGuardian Pharmacy Web
Required Interfaces:	showPrescription verifyDosage selectMedicationForm notifyDispensing Medication Dispensing Data (dataAccess)
Provided Interfaces:	None

Component	Prescription Verification Manager
Description	Verifies the validity of prescriptions by checking medication details and signatures.
Type	HealthGuardian Pharmacy Web
Required Interfaces:	verifyPrescription checkMedication checkSignature Prescription Verification Data (dataAccess)
Provided Interfaces:	None

Component	Inventory Manager
Description	Manages stock levels, checks for expired items, and processes purchase orders.



Type	HealthGuardian Pharmacy Web
Required Interfaces:	checkQuantity getExpiryDate viewPurchaseOrder Inventory Management Data (dataAccess)
Provided Interfaces:	updateInventory makePurchaseOrder

Component	Medication Information Manager
Description	Maintains and updates medication information and provides consultation support.
Type	HealthGuardian Pharmacy Web
Required Interfaces:	getMedication Medication Information Data (dataAccess)
Provided Interfaces:	addMedication updateMedication consultationSupport

Component	Pharmacist Data Manager
Description	Manages pharmacist user data for the system, including account creation and search.
Type	HealthGuardian Pharmacy SaaS
Required Interfaces:	Login Logout showPrescription verifyDosage selectMedicationForm notifydispensing verifyPrescription checkMedication checkSignature searchAccount searchprescription showMedicationInfo showSignature
Provided Interfaces:	checkAccess checkPatient verifyPrescriptionInfo

Component	User Authentication and Authorization System
Description	A system for managing user credentials and granting authorization to various services.

Type	HealthGuardian Pharmacy SaaS
Required Interfaces:	None
Provided Interfaces:	searchAccount

Component	Medication Dispensing System
Description	System dedicated to the process of dispensing medication, ensuring correct dosages and forms are provided.
Type	HealthGuardian Pharmacy SaaS
Required Interfaces:	searchPrescription
Provided Interfaces:	checkPatient

Component	Prescription Verification System
Description	System responsible for the verification of prescription details and authenticity.
Type	HealthGuardian Pharmacy SaaS
Required Interfaces:	verifyPrescriptionInfo
Provided Interfaces:	showMedicationInfo showSignature

Component	Medication Data Manager
Description	Manages the medication data including maintaining records, updating details, and facilitating the retrieval of medication information.
Type	HealthGuardian Pharmacy SaaS
Required Interfaces:	updateInventory makePurchaseOrder viewPurchaseOrder addMedication updateMedication consultationSupport
Provided Interfaces:	checkQuantity getExpiryDate viewPurchaseOrder getMedication

Component	Inventory Management System
Description	A system that manages inventory, including checking levels,

	expiry dates, and managing orders.
Type	HealthGuardian Pharmacy SaaS
Required Interfaces:	generateOrder
Provided Interfaces:	checkInventory checkExpiryDate

Component	Medication Information System
Description	Provides comprehensive information about medications and supports consultation services.
Type	HealthGuardian Pharmacy SaaS
Required Interfaces:	None
Provided Interfaces:	retrieveMedicationInfo retrieveMedicationDetails recordSupport

### 2.2.2 Interface Table

Interface Name	User Authentication and Authorization	
Provider	HealthGuardian Pharmacy SaaS	
	Operation:	Login
	Request Parameters: pharmacistID, password Response Parameters: accessCode (if successful); errorMessage (if failure) Function: Validates pharmacist's credentials. Returns an access code upon successful login, or an error message if credentials are invalid.	
	Operation:	Logout
	Request Parameters: pharmacistID Response Parameters: confirmationMessage Function: Logs out the pharmacist from the system and invalidates the current session.	

Interface Name	Medication Dispensing
Provider	HealthGuardian Pharmacy SaaS

	Operation:	showPrescription
	Request Parameters: prescriptionID, createTime Response Parameters: prescriptionDetails Function: Retrieves and displays comprehensive details of a specific prescription, including patient information, prescribed medication, dosages, and instructions, based on its ID and timestamp.	
	Operation:	verifyPrescription
	Request Parameters: prescriptionID, description, checkSignature Response Parameters: verificationStatus Function: Conducts a thorough verification of a prescription's authenticity by checking the prescription ID, detailed description, and doctor's signature against hospital records.	
	Operation:	notifyPrescription
	Request Parameters: prescriptionID, patientID, doctorID, status Response Parameters: notificationStatus Function: Notifies relevant parties (patients, doctors, etc.) about the status of a prescription, including approvals, rejections, or need for further information.	
	Operation:	checkDosageAndForm
	Request Parameters: description, dosageForm Response Parameters: dosageConfirmation Function: Examines the prescribed medication's dosage and form (e.g., tablet, liquid) to ensure it aligns with standard medical guidelines and is appropriate for the patient's condition.	

Interface Name	Prescription Verification	
Provider	HealthGuardian Pharmacy SaaS	
	Operation:	verifyPrescription
	Request Parameters: prescriptionID, description, checkSignature Response Parameters: verificationStatus Function: Performs a detailed validation of the prescription by cross-referencing it with medical records, ensuring the medication, dosage, and prescribing doctor's signature are accurate and legitimate.	
	Operation:	checkMedication

	Request Parameters: medicationID, medicationName, dosageForm Response Parameters: medicationStatus Function: Validates medication details against the hospital's pharmaceutical database, confirming the correct medication is being dispensed according to the prescription.	
	Operation:	checkSignature
	Request Parameters: prescriptionID, signature Response Parameters: signatureValidity Function: Authenticates the doctor's signature on the prescription, cross-checking it with a database of registered medical professionals to prevent fraudulent prescriptions.	

Interface Name	Inventory Management	
Provider	HealthGuardian Pharmacy SaaS	
	Operation:	checkQuantity
	Request Parameters: medicationID, quantity Response Parameters: quantityStatus Function: Monitors and reports the current stock levels of medications in the pharmacy, ensuring adequate supply and flagging items for reorder as needed.	
	Operation:	getExpiryDate
	Request Parameters: medicationID, expiryDate Response Parameters: expiryInformation Function: Retrieves and displays the expiry dates of medications, crucial for maintaining the safety and effectiveness of the pharmaceutical inventory.	
	Operation:	updateInventory
	Request Parameters: medicationID, quantity Response Parameters: updateStatus Function: Adjusts inventory records, either adding or subtracting stock as medications are received or dispensed, ensuring accurate and up-to-date inventory tracking.	
	Operation:	makePurchaseOrder

	Request Parameters: purchaseOrderID, medicationID, quantity Response Parameters: orderStatus Function: Facilitates the creation of purchase orders for medications, specifying quantities and details, to maintain adequate stock levels.		
	<table> <tr> <td>Operation:</td><td>viewPurchaseOrder</td></tr> </table> Request Parameters: purchaseOrderID, medicationID, orderState Response Parameters: orderDetails Function: Provides detailed views of purchase orders, including medication details, quantities, and current processing status, aiding in inventory management.	Operation:	viewPurchaseOrder
Operation:	viewPurchaseOrder		

Interface Name	Medication Information		
Provider	HealthGuardian Pharmacy SaaS		
	<table> <tr> <td>Operation:</td><td>addMedication</td></tr> </table> Request Parameters: medicationID, medicationName, expiryDate, dosageForm Response Parameters: additionStatus Function: Introduces new medications into the system, recording essential details like ID, name, expiry date, and dosage form for future reference and dispensing.	Operation:	addMedication
Operation:	addMedication		
	<table> <tr> <td>Operation:</td><td>getMedication</td></tr> </table> Request Parameters: medicationID, medicationName, expiryDate, dosageForm Response Parameters: medicationDetails Function: Retrieves detailed information on specific medications, including their name, expiry date, and form, aiding pharmacists in dispensing and consultation.	Operation:	getMedication
Operation:	getMedication		
	<table> <tr> <td>Operation:</td><td>updateMedication</td></tr> </table> Request Parameters: medicationID, medicationName, expiryDate, dosageForm Response Parameters: updateStatus Function: Updates existing medication records in the system, ensuring that all information remains current and accurate for safe dispensing practices.	Operation:	updateMedication
Operation:	updateMedication		
	<table> <tr> <td>Operation:</td><td>consultationSupport</td></tr> </table>	Operation:	consultationSupport
Operation:	consultationSupport		

	Request Parameters: medicationID, supportInfo Response Parameters: consultationDetails Function: Provides pharmacists with additional support and information regarding medications, aiding in patient consultations and ensuring effective medication management.
--	--

## 2.3 Patient

### 2.3.1 Component Table

Component	Register Manager
Description	Manages user registration, including capturing and storing user details.
Type	Patient App
Required Interface	Patient
Provided Interface	Register

Component	Login Manager
Description	Manages user logins, including authentication and session
Type	Patient App
Required Interface	Patient
Provided Interface	Login, Logout
Component	Appointment Manager
Description	Handles appointment scheduling, modifications, and cancellations.
Type	Patient App
Required Interface	displayAppointment()
Provided Interface	deleteAppointment(),updateAppointment(),addAppointment()

Component	MedicalHistory Manager
Description	Manages the storage and retrieval of patients' medical history records.

Type	Patient App
Required Interface	viewDiagnoses(),viewTreatments(),viewMedications(),viewreportResults()
Provided Interface	None

Component	Prescription Manager
Description	Manages the processing and refilling of patient prescriptions.
Type	Patient App
Required Interface	refillsPrescription()
Provided Interface	addPrescription()

Component	Reminder Manager
Description	Sends notifications and reminders to patients for appointments and medications.
Type	Patient App
Required Interface	sendNotification(),sendConfirmEmail()
Provided Interface	RequestAppointment()

Component	Payment Manager
Description	This component is responsible for handling the billing process, including issuing invoices, processing payments, and managing payment records for healthcare services.
Type	Patient App
Required Interface	viewBillInfo(),payment(),reviewInsuranceClaims()
Provided Interface	None

Component	Authetication Manager
Description	This is a microservice that runs on the hospital's cloud system and is used to store, read and authenticate each patient's personal information.



Type	Health Guardian SaaS
Required Interface	Resigter, Login, CheckAcccount
Provided Interface	Logout, update Patient Info,Create new patient

Component	Appointment Data Manager
Description	This is a database for storing, modifying, updating, and deleting patient appointment information.
Type	Health Guardian SaaS
Required Interface	addAppointment(),updateAppointment(),delateAppointment()
Provided Interface	displayAppointment()

Component	Patient Data Manager
Description	It's a database for storing all the patient's personal information
Type	Health Guardian SaaS
Required Interface	Authetication,Appointment Data, MedicalHistory Data
Provided Interface	Patient Data

Component	Prescription Data Manager
Description	Maintained patient prescription database.
Type	Health Guardian SaaS
Required Interface	addPrescription(),refillsPrescription()
Provided Interface	addPrescription(),refillsPrescription()

Component	MedicalHistory Data Manager
Description	Maintains the database of medical histories, diagnoses, and treatments.
Type	Health Guardian SaaS
Required Interface	None
Provided Interface	viewDiagnoses(),viewTreatments(),viewMedications(),viewreportResults()

Component	Reminder Service Manager
Description	Oversees the reminder services, ensuring timely notifications for appointments and medications.
Type	Health Guardian SaaS
Required Interface	RequestAppointment()
Provided Interface	sendNotification(),sendConfirmEmail()

Component	Payment Data Manager
Description	Responsible for the secure storage and retrieval of payment data. This component interacts with the data layer to manage payment records and billing information.
Type	Health Guardian SaaS
Required Interface	UpdateBillInfo()
Provided Interface	UpdateBillInfo(),viewBillInfo(),payment(),reviewInsuranceClaims()

### 2.3.2 Interface Table

Interface Name	Patient
Provider	Patient App
	Operation: register
	Request Parameters: username (string), phoneNumber (string), password (string)  Response Parameters: successMessage (string), if registration is successful; errorMessage (string), if failure.  Function: Registers a new patient by collecting their username, phone number, and password, creating a new patient record in the system.
	Operation: login
	Request Parameters: username (string), password (string)  Response Parameters: sessionToken (string), if login is successful; errorMessage (string), if failure.

	Function: Authenticates a patient by verifying their username and password against stored records and generates a session token for successful logins.
	Operation:   logout
	Request Parameters: sessionToken (string)
	Response Parameters: successMessage (string), confirming successful logout.
	Function: Terminates a patient's session in the system, ensuring their authentication token is invalidated.

Interface Name	Make an Appointment
Provider	Appointment Manager
	Operation:   displayAppointment
	Request Parameters: patientId (string), date (date)
	Response Parameters: appointmentDetails (AppointmentDetail[]), list of appointments.
	Function: Retrieves and displays all scheduled appointments for a given patient on a specified date.
	Operation:   addAppointment
	Request Parameters: patientId (string), appointmentDetail (AppointmentDetail)
	Response Parameters: appointmentId (string), confirming the creation of the appointment.
	Function: Schedules a new appointment for the patient and stores the details in the system.
	Operation:   updateAppointment
	Request Parameters: appointmentId (string), updatedDetail (AppointmentDetail)
	Response Parameters: successMessage (string), confirming the update.
	Function: Updates the details of an existing appointment.

	Operation:	deleteAppointment
	Request Parameters: appointmentId (string)	
	Response Parameters: successMessage (string), confirming the deletion.	
	Function: Removes an existing appointment from the system.	

Interface Name	MedicalHistory	
Provider	MedicalHistory Manager	
	Operation:	viewDiagnoses
	Request Parameters: patientId (string)	
	Response Parameters: diagnosesList (Diagnosis[])	
	Function: Retrieves a list of past diagnoses for the patient.	
	Operation:	viewTreatments
	Request Parameters: patientId (string)	
	Response Parameters: medicationsList (Medication[])	
	Function: Retrieves a list of medications that have been prescribed to the patient in the past	
	Operation:	viewLabResults
	Request Parameters: patientId (string)	
	Response Parameters: labResultsList (LabResult[])	
	Function: Retrieves a list of lab results for the patient.	
	Operation:	deleteAppointment
	Request Parameters: appointmentId (string)	
	Response Parameters: successMessage (string), confirming the deletion.	
	Function: Removes an existing appointment from the system.	

Interface Name	Prescription	
Provider	Prescription Manager	
	Operation:	addPrescription
	Request Parameters: patientId (string), prescriptionDetail (PrescriptionDetail)	
	Response Parameters: prescriptionId (string), confirming the creation of the prescription.	
	Function: Adds a new prescription record for the patient.	
	Operation:	refillPrescription
	Request Parameters: prescriptionId (string), quantity (integer)	
	Response Parameters: successMessage (string), confirming the refill.	
	Function: Processes a request to refill an existing prescription for the patient.	

Interface Name	Notification	
Provider	Reminder Manager	
	Operation:	sendNotification
	Request Parameters: patientId (string), notificationDetail (NotificationDetail)	
	Response Parameters: successMessage (string), confirming the sending of the notification.	
	Function: Sends a notification to the patient for appointments or medication reminders.	
	Operation:	sendConfirmationEmail
	Request Parameters: patientId (string), emailDetail (EmailDetail)	
	Response Parameters: successMessage (string), confirming	

	the sending of the email.	
	Function: Sends an email confirmation for appointments or other interactions with the patient.	
	Operation:	updateAppointment
	Request Parameters: appointmentId (string), updatedDetail (AppointmentDetail)	
	Response Parameters: successMessage (string), confirming the update.	
	Function: Updates the details of an existing appointment.	
	Operation:	deleteAppointment
	Request Parameters: appointmentId (string)	
	Response Parameters: successMessage (string), confirming the deletion.	
	Function: Removes an existing appointment from the system.	

Interface Name	Billing	
Provider	Payment Data Manager	
	Operation:	viewBillInfo
	Request Parameters: billingName: string, patientId: int, billingPrice: decimal, patientId: int	
	Response Parameters: billDetails	
	Function: Retrieves a comprehensive view of the bill for a patient, including the breakdown of charges.	
	Operation:	payment
	Request Parameters: billingPrice: string, patientId: int	
	Response Parameters: paymentConfirmation: boolean, transactionId: string	
	Function: Processes the payment for the patient's bill based on the amount due and updates the billing status.	
	Operation:	reviewInsuranceClaims

	<p>Request Parameters: appointmentId (string), updatedDetail (AppointmentDetail)</p> <p>Response Parameters: claimsStatus</p> <p>Function: Reviews and processes insurance claims associated with the patient's account, including coordination with insurance providers.</p>
--	---

## 2.4 Laboratory

### 2.4.1 Component

Component	Login manager
Description	This microservice running on the could to manage laboratory login.
Type	Laboratory App
Required Interface	Login
Provided interface	Access Code

Component	Data Entry manager
Description	This microservice running on the could to manage data entry.
Type	Laboratory App
Required Interface	Lab, Access Code, Get request
Provided interface	/

Component	Instrument Troubleshooting manager
Description	This microservice running on the could to manage instrument Troubleshooting.
Type	Laboratory App
Required Interface	Equipment, instrument, Access Code
Provided interface	/

Component	Hand Test Request manager
Description	This microservice running on the could to handle the test request

	from the doctor.
Type	Laboratory App
Required Interface	Test request, Access code
Provided interface	Get Request

Component	Authentication manager
Description	This microservice running on the could to handle the login request from the user.
Type	Microservice
Required Interface	Check password
Provided interface	Login

Component	Laboratory Data manager
Description	This microservice running on the could to manage Laboratory Data
Type	Microservice
Required Interface	/
Provided interface	Check password, Check Access

Component	Lab
Description	This microservice running on the could to manage Lab Data
Type	Microservice
Required Interface	Check Access
Provided interface	Lab

Component	Equipment
Description	This microservice running on the could to manage equipment Data
Type	Microservice
Required Interface	Check Access
Provided interface	Equipment

Component	Instrument
-----------	------------



Description	This microservice running on the could to manage equipment Data
Type	Microservice
Required Interface	Check Access
Provided interface	Instrument

Component	Test Result
Description	This microservice running on the could to manage test result
Type	Microservice
Required Interface	Check Access
Provided interface	Test Result

## 2.4.2 Interface Table

Interface name	Login
Provider	Authentication Manager
	Operation: login
	Request Parameters: username, password Response parameter: True if success, False if fail Function: Check if the user's password matches the stored password. If so, return Ture; otherwise, return False.

Interface name	Test Request
Provider	Test Request
	Operation: setTestRequest
	Request Parameters: testInfo Response parameter: testRequestId if success, False if fail Function: Store the test request into the database. Return testInfo, if success; otherwise, return False.
	Operation: getTestRequest
	Request Parameters: testResultId Response parameter: testRequest if success, False if fail Function: Get the test request store in the database use the test request id. Return testRequest, if success; otherwise, return False.

Interface name	Equipment
Provider	Equipment
	Operation: getEquipmentStatus
	Request Parameters: equipmentId Response parameter: equipmentId if success, False if fail Function: Get the equipmentStatus store in the database use the equipment id. Return equipmentStatus, if success; otherwise, return False.
	Operation: displayEquipmentStatus
	Request Parameters: equipmentId Response parameter: equipmentStatus if success, False if fail Function: Get the equipmentStatus store in the database use the equipment id. Return equipmentStatus, if success; otherwise, return False.

Interface name	instrument
Provider	instrument
	Operation: getInstrumentStatus
	Request Parameters: InstrumentId Response parameter: InstrumentId if success, False if fail Function: Get the InstrumentStatus store in the database use the Instrument id. Return InstrumentStatus, if success; otherwise, return False.
	Operation: displayInstrumentStatus
	Request Parameters: InstrumentId Response parameter: InstrumentStatus if success, False if fail Function: Get the InstrumentStatus store in the database use the Instrument id. Return InstrumentStatus, if success; otherwise, return False.

Interface name	Lab
Provider	Lab
	Operation: setTest
	Request Parameters: name, results, dataAccurately Response parameter: True if success, False if fail Function: Record the test the database. Return true, if success; otherwise, return False.
	Operation: instrumentStatus
	Request Parameters: instrumentId

	Response parameter: True if success, False if fail Function: Send the get instrument statue request to instrument database. Return true if sucess; otherwise, return False.
	Operation: equipmentStatus Request Parameters: equipmentId Response parameter: True if success, False if fail Function: Send the get equipment statue request to equipment database. Return true if sucess; otherwise, return False.
	Operation: generateReportIssuesForm Request Parameters: / Response parameter: reportIssuesForm if success, False if fail Function: Generate report Issues Form for user to fill in information. Return reportIssuesForm, if success; otherwise, return False.
	Operation: generateIssuesReport Request Parameters: / Response parameter: IssuesReport if success, False if fail Function: Generate Issues report for user to see what they update. Return issuesReport, if success; otherwise, return False.
	Operation: submitIssuesReport Request Parameters: / Response parameter: True if success, False if fail Function: Set the issues that user input into the database. Return true, if success; otherwise, return False.

#### Database

Table: Authentication Manager	Fields: uername; password
Table: Lab	Fields: laboratoryId; laboratoryIdName; username; password; telephone; age
Table: Equipment	Fields: equitmentId; equitmentStatue
Table: Instrument	Fields: instrumentId; instrumentStatue
Table: Test Requist	Fields: testRequestId; testInfo