



- patientId: String - doctorld: String - requestedDate: LocalDate - requestedTimeSlot: LocalTime

- status: String

AppointmentRequestService - scheduleService: ScheduleService - appointmentService: AppointmentService - appointmentRequestFile: String + AppointmentRequestService(scheduleService: ScheduleService, appointmentService: AppointmentService) + acceptRequest(request: AppointmentRequest): void + getPendingRequests(): List<AppointmentRequest> + processPendingRequests(doctorID: String): void + declineRequest(request: AppointmentRequest): void + saveAppointmentRequest(request: AppointmentRequest): void + save(request: AppointmentRequest): void + getAllRequests(): List<AppointmentRequest> <<interface>> | IAppointementRequestService + acceptRequest(request: AppointmentRequest): void + getPendingRequests(): List<AppointmentRequest> + processPendingRequests(doctorID: String): void + declineRequest(request: AppointmentRequest): void + saveAppointmentRequest(request: AppointmentRequest): void + getAllRequests(): List<AppointmentRequest> UserService - users: Map<String, User> + UserService() + login(hospitalID: String, password: String): boolean + changePassword(hospitalID: String, oldPassword: String, newPassword: String): boolean + getUserRole(hospitalID: String): UserRole + getUserById(hospitalID: String): User + updateUser(user: User): boolean + addUser(user: User): void + removeUser(hospitalID: String): boolean + saveToCSV(): void - loadUsersFromCSV(filePath: String): void saveUsersToCSV(): void - writeUsersToCSV(): void **IUserService** + login(hospitalID: String, password: String): boolean + changePassword(hospitalID: String, oldPassword: String, newPassword: String): boolean + getUserRole(hospitalID: String): UserRole **IProjectAdmService** + addOrUpdateStaff(staffMember: Staff): void + removeStaff(staffId: String): boolean + getAllStaff(): List<Staff> + updateStockLevel(medicineName: String, quantity: int): boolean + updateLowStockAlert(medicineName: String, lowStockLevel: int): + removeMedication(medicineName: String): boolean + getAllInventory(): List<Inventory> + approveReplenishmentRequest(medicineName: String): boolean ProjectAdminService administrator: Administrator - staffDataStore: StaffDataStore inventorvService: IInventorvService - userService: UserService - replenishmentService: ReplenishmentService - STAFF\_CSV\_PATH: static final String - DOCTOR\_CSV\_PATH: static final String - PHARMACIST\_CSV\_PATH: static final String + ProjectAdminService(administrator: Administrator, inventoryService: IInventoryService, userService: UserService) + addOrUpdateStaff(staffMember: Staff): void - addOrUpdateDoctorCSV(staffMember: Staff): void - addOrUpdatePharmacistCSV(staffMember: Staff): void - updateOrAppendToCSV(csvPath: String, staffMember: Staff): void - updateCSV(csvPath: String, staffMember: Staff): void + removeStaff(staffId: String): boolean - removeFromCSV(csvPath: String, staffld: String): boolean + getAllStaff(): List<Staff> + updateStockLevel(medicineName: String, quantity: int): boolean + updateLowStockAlert(medicineName: String, lowStockLevel: int): boolean + addMedication(newMedication: Inventory): void + removeMedication(medicineName: String): boolean + getAllInventory(): List<Inventory> + approveReplenishmentRequest(medicineName: String): boolean + viewPendingReplenishmentRequests(): List<ReplenishmentRequest> + declineReplenishmentRequest(requestId: String): boolean

<<interface>> **InventoryService** + getInventoryList(): List<Inventory> + updateInventory(updatedInventory: Inventory): void + addMedication(newMedication: Inventory): void + removeMedication(medicineName: String): boolean + updateStockLevel(medicineName: String, quantity: int): boolean + updateLowStockAlert(medicineName: String, lowStockLevel: int): boolean + submitReplenishmentRequest(): void + approveReplenishmentRequest(medicineName: String): boolean InventoryService - inventoryDataStore: InventoryDataStore - CSV FILE PATH: String - DELIMITER: String
- appointmentService: AppointmentService + InventoryService(store: InventoryDataStore) + saveDataToCSV(): void + loadDataFromCSV(): void + submitReplenishmentRequest(): void + updateStock(medicineName: String, quantity: int, appointmentID: String): void + viewMedicationInventory(): void + getInventoryDisplay(): List<InventoryDisplay> + getLowStockInventory(): List<InventoryDisplay: + addMedication(newMedication: Inventory): void + removeMedication(medicineName: String): boolean + updateStockLevel(medicineName: String, quantity: int): boolean + updateLowStockAlert(medicineName: String, lowStockLevel: int): boolean + approveReplenishmentRequest(medicineName: String): boolean + getInventoryList(): List<Inventory> - calculateReplenishmentAmount(currentStock: int, lowLevelAlert: int): int - determineInventoryStatus(inventory: Inventory): InventoryStatus - updateInventory(updatedInventory: Inventory); void <<interface>> **IPharmacistService** + loadPharmacistsFromCSV(filePath: String): List<Pharmacist> + savePharmacistsToCSV(pharmacists: List<Pharmacist>, filePath: String): void **PharmacistService** - pharmacists: Map<String, Pharmacist> - userService: UserService - appointmentService: AppointmentService - inventoryService: InventoryService - CSV\_FILE\_PATH: String - DELIMITER: String + PharmacistService(userService: UserService, appointmentService: AppointmentService, inventoryService: InventoryService) + addPharmacist(pharmacist: Pharmacist): void + getPharmacistById(hospitalID: String): Pharmacist + updatePharmacistContact(hospitalID: String, newContactInformation: String): boolean + listAllPharmacists(): void + loadPharmacistsFromCSV(): void + savePharmacistsToCSV(): void + getAppointmentsByPatientId(patientId: String): List<Appointment> + updatePrescriptionStatus(appointmentID: String): boolean

AppointmentService

+ rescheduleAppointment(appointmentld: String, newAppointment: Appointment): boolean

List<Medication>, prescribedQuantities: List<Integer>, consultationNotes: String): void

+ getAvailableSlots(doctorId: String, date: LocalDate): List<LocalDateTime>

+ recordAppointmentOutcome(appointmentId: String, serviceProvided: String, prescribedMedications:

- APPOINTMENT\_FILE: String = "data/appointment.csv"

+ scheduleAppointment(appointment: Appointment): boolean

+ cancelAppointment(appointmentId: String): boolean

+ viewScheduledAppointments(): List<Appointment>

+ getAppointment(appointmentId: String): Appointment

+ updateMedicationStatus(appointmentId: String): void

+ loadAppointmentsFromCSV(): void

+ saveAppointmentsToCSV(): void

- createCSVIfNotExists(): void

+ getAppointmentByld(appointmentId: String): Appointment

+ updateAppointment(updatedAppointment: Appointment): void

+ AppointmentService()

+ findAppointment(patientId: String, doctorId: String, date: LocalDate, timeSlot: LocalTime): Appointment

ReplenishmentService - csvFile: String + ReplenishmentService() + createReplenishmentRequest(medicines: List<String>): List<ReplenishmentRequest>
+ getAllRequests(): List<ReplenishmentRequest> + getRequestsByStatus(status: StatusEnum): List<ReplenishmentRequest> + approveRequest(requestId: String): String + rejectRequest(requestId: String): boolean + updateRequestStatus(requestId: String, newStatus: StatusEnum): boolean - writeToCsv(request: ReplenishmentRequest): void - parseCsvRow(csvRow: String): ReplenishmentRequest - isMedicineAlreadyRequested(medicineName: String): boolean - generateCsvlfNotExists(): void saveRequestsToCSV(requests: List<ReplenishmentRequest>): void

> BillingService - CONSULTATION\_CHARGE: double PARACETAMOL\_CHARGE: double - IBUPROFEN CHARGE: double - AMOXICILLIN\_CHARGE: double + calculateBill(appointment: Appointment): double

> > **IDoctorService**

+ getDoctorById(hospitalID: String): Doctor + updateDoctorContact(hospitalID: String, newContactInformation: String): boolean + listAllDoctors(): void + acceptanceOfRequest(requestId: String): void + declineRequest(requestId: String): void + setAvailablity(doctor: Doctor, dateStr: String, timeSlotStr: String): boolean + setUnavailability(doctor: Doctor, dateStr: String, timeSlotStr: String): boolean + updatePatientDiagnosis(patientID: String, newDiagnosis: String): void + updatePatientPrescription(patientID: String, newPrescription: String): void + recordAppointmentOutcome(appointmentId: String, serviceProvided: String, medicationsInput: String, quantitiesInput: String, consultationNotes: String): void **DoctorService** - doctors: Map<String, Doctor> - userService: UserService - appointmentRequestService: AppointmentRequestService - scheduleService: ScheduleService - medicalRecordService: MedicalRecordService - appointmentService: AppointmentService CSV\_FILE\_PATH: String - DELIMITER: String
- DATE\_FORMAT: DateTimeFormatter TIME\_FORMAT: DateTimeFormatter + DoctorService(userService: UserService, scheduleService: ScheduleService, medicalRecordService: MedicalRecordService, appointmentService: AppointmentService) + addDoctor(doctor: Doctor): void + getDoctorByld(hospitalID: String): Doctor + updateDoctorContact(hospitalID: String, newContactInformation: String): boolean + listAllDoctors(): void + loadDoctorFromCSV(): void + saveDoctorsToCSV(): void + acceptanceOfRequest(requestId: String): void + declineRequest(requestId: String): void + setAvailablity(doctor: Doctor, dateStr: String, timeSlotStr: String): boolean + setUnavailability(doctor: Doctor, dateStr: String, timeSlotStr: String): boolean + updatePatientDiagnosis(patientID: String, newDiagnosis: String): void + updatePatientPrescription(patientID: String, newPrescription: String): void + recordAppointmentOutcome(appointmentId: String, serviceProvided: String, medicationsInput: String, quantitiesInput: String, consultationNotes: String): void - writeBillingEntry(appointment: Appointment): void

This property and selection of the control of the	+ getId(): String	+ getRo	ole(): UserRole			
The Control of Part		+ setRo	ole(): boolean		7	
Part		+ getPa	issword(); string issword(); string); boolean			
Table of the control			isonoi a(pasonoi ai ouiiig). Boolean			
Transmit	setRole(role: String): void		$\wedge$			
Part	getGender(): String		<b>分</b>			
Training						
Tennic processed by the control of t						
Parce   Parc					Pat	tient
Manufacture						
Maintaine   Main						
The content of the	InventoryDisplay		7			
The control of the	String	Pharmacist	Postor			
Part			Doctor		- isRegistered: Boolean	
Part	s: InventoryStatus	- contactInformation: String	- name : String			
**************************************	sishmentStatus: ReplenishmentStatus		- contactInformation : String			ring, bloodType: String, contactInformation: String)
The second continuous			+ Dester/user : Hear name : String cente	actinformation : String)	+ getName(): String	
**************************************	vantoryDienlay/name: String, quantity; int. status: InventoryStatus, replanishmentStatus: ReplanishmentStatus)	+ Pharmacist(user: User, name: String, contactInformation: String)		cumormation . Suring)		
	ame(): String	+ getName(): String	+ getContactInformation() : String		+ getBloodType(): String	
Part		+ getContactInformation(): String + setContactInformation(contactInformation: String): void		on : String) : void	+ getContactInformation(): String	
Registrational distribution is an extensive in the foliage of the controlled of the			+ tostilig() . Stillig		+ getRegistrationStatus(): Boolean + setName(name: String): void	
Rightiminentification  Provided to the provide	theplenishmentstatus(). Replenishmentstatus thiventoryStatus(currentStock; int. lowLevelAlert; int); InventoryStatus	30, 3			+ setName(name: String): Void + setDateOfBirth(dateOfBirth: LocalDate): void	
Paper   Pape	,				+ setGender(gender: String): void	
Part					+ setBloodType(bloodType: String): void	
Internation of the control of the co				_		
Designation of the control straig product (Control Term)  - Control Term Straig - Contro	ReplenishmentRequest	Inve	entory		+ getAppointments(): List <appointment></appointment>	
Appointment  Appoi	· · · · · · · · · · · · · · · · · · ·	modicineName: String				
Approximate in distance in a concentration of SURE in a concentration of SU						
support procession of the pr		- lowLevelAlert: int				
Investment (Control of Control					Appointment	
International content of the properties of the processing and content to the properties of the processing and content to the		- inventorystatus: inventorystatus			· ·	
	and a nick man at Dagues at Managar Link (Ottions)			- appointmentId: String	ng	
Part	premishmentRequest(id: String, medicines: List <string>) eplenishmentRequest(id: String, medicines: List<string>. status: StatusEnum)</string></string>		and Alast int and the last of the same and t	- doctorld: String		Bi
**************************************	etId(): String		everAlert: Int, replenishmentStatus: ReplenishmentStatus)	- appointmentDateTir		
# SpiChamas County of the Carbon and Add by 1						
**** plate of counterful from the counterful	tStatus(): StatusEnum tStatus(status: StatusEnum): void	+ getCurrentStock(): int				
# exclusion/facility conduction from the control of	Id(id: String): void			- serviceProvided: Str	tring	- appointmentId: String
# Special process of the process of	CsvFormat(): String			- appointmentOutcom	meDate: LocalDateTime	- totalAmount: double
* self-control (String) count of String) void control (String) count of String) void cou		+ getReplenishmentStatus(): ReplenishmentStatus		+Appointment/appoi	ntmontid: String nationald: String doctorid: String appointmentDataTime:	- status: String
MedicineConsultation  WedicineConsultation  Introd  Inter  SchalaClineTrod  - patientID: Sirrig - patientI		+ setReplenishmentStatus(replenishmentStatus: Replenish	hmentStatus): void	LocalDateTime)	numeritid. String, patientid. String, doctorid. String, appointmentDate rime.	
Middle Consultation  String  Middle Middle Consultation  Middle Consulta		+ settinventoryStatus(inventoryStatus), vo	ia	+getAppointmentId()	: String	+ Billing()
Subpaction   Sub				+setAppointmentId(a	appointmentId: String): void	
MedicalRecord	Madicine Consultation			+setPatientId(): String	ld: Strina): void	+ getInvoiceId(): String
MedicaRecord  patentials (LocalDella Time patentials (LocalDella Time) pat				+getDoctorId(): String	g	+ setInvoiceId(invoiceId: String): void
Jacks Statisfier String  - setter 100 - String - setter 100 - Stri		MedicalR	Record			
patient (C. Sitting) -				+getAppointmentDate	eTime(): LocalDateTime eTime(appointmentDateTime: LocalDateTime): void	+ getDoctorId(): String
- name Saring - clicineConsultation() - name Saring - clicineConsultation() - name Saring - clicineConsultation() - consultation() - consultati				+getStatus(): Appoint	tmentStatus	+ setDoctorId(doctorId: String): void
- gender: String - gend	district Constitution ()			+setStatus(status: Ap	ppointmentStatus): void	
g, consultation Feer Friahflour couble)  ### Section Annual Process String  ### Section Annual Process String ### Section Annual Process String ### Section Annual Process String ### Section Annual Process String ### Section Annual Process String ### Section Annual Process String ### Section Annual Process String ### Section Annual Process String ### Section Annual Process String ### Section Annual		- gender: String				+ getTotalAmount(): double
### deficiency: String #### deficiency: String, passer String, pas	ng, consultationFeePerHalfHour: double)	- phoneNumber: String		+getConsultationNote	res(): String	+ setTotalAmount(totalAmount: double): vo
### String() String pastPostroff(ref)() double ### String() String pastPostroff(ref)() double ### MedicalRecord(patientID: String, poneNumber: String, newPrescription: String newPrescription: String ### String() String ### Hosting() String	etMedicine(): String			+setConsultationNote	es(consultationNotes: String): void	+ getStatus(): String + setStatus(status: String): void
# from/CSV(line: String)		- pastDiagnoses: String				+ toString(): String
- newPrescription: String   - newIcineName: Strin	setMedicinePriceForTen(medicinePriceForTen: double): void			+getAppointmentOut	tcomeDate(): LocalDateTime	+ fromCSV(line: String): Billing
# MedicalRecord(patient) String, name: String, ob: String, pender, String, phoneNumber: String, pender, String	getDoctorId(): String	- newPrescription: String		+setAppointmentOut	comeDate(appointmentOutcomeDate: LocalDateTime): void	
+ MedicalRecord(patientID: String, name: String, emailAddress: String, string, string, emailAddress: String, string, emailAddress: S						
bloodType: String, pastDiagnoses: String, pastDiagnoses: String, newDiagnosis: String, newPrescription: String) segPalamtDi(): String segPloom ():	setConsultationFee(consultationFeePerHalfHour: double): void	+ Modical Population IID: China name: China de La China	or String phonoNumber String and 14 dd-			
# egPatientID(); String # egPatientID(); String # egPatient(); Str		bloodType: String, pastDiagnoses: String, pastPrescriptions: String.	newDiagnosis: String, newPrescription: String)			
+ getName(y; String	adConsultationsFromCSV(filePath: String): void	+ getPatientID(): String				
+ getPoneNumber(f): String + getPoneNumber(f): String + getBloodType(): String + getBloodType(): String + getBloodType(): String + getPastPrescriptions(): String + getPastPrescriptions(): String + getNewPrescription(): String + getNewPrescription(): String + setName(name: String): void + setPoneNumber(phoneNumber: String): void + setPoneNumber(phoneNumber: String): void + setBloodType(bloodType: String): void	,g/,				SubmitReplenishmentRequest	
+ getPhoneNumber(): String + getEmailAddress(): String + getBloafdype(): String + getPhoseNip(): String + getPhoseNip(): String + getNewDiagnosis(): String + getNewDiagnosis(): String + getNewDiagnosis(): String + getNewPrescription(): String + getNewPrescription(): String + getNewPrescription(): String + setNam(name: String): void + setDob(dob: String): void + setDohneNumber(phoneNumber: String): void + setPhoneNumber(phoneNumber: String): void + setBloodType(bloodType: String) + setBloodType(bloodType: String): void		+ getDob(): String + getGender(): String		- medicineName	: String	- PAYMENT_LOC
+ getBloodType(): String + getPastDiagnoses(): String + getPastPrescriptions(): String + getPastPrescriptions(): String + getNewDiagnosis(): String + getNewPrescription(): String) + setName(name: String): void + setDob(dob: String): void + setPhoneNumber(phoneNumber: String): void + setPhoneNumber(phoneNumber: String): void + setBoodType(bloodType: String): void + setBloodType(bloodType: String): void		+ getPhoneNumber(): String		- currentStock: in	nt	
+ getPastDiagnoses(): String + getPastPrescriptions(): String + getNewDiagnosis(): String + getNewDiagnosis(): String + getNewPrescription(): String + getNewPrescription(): String + getNewPrescription(): String + setName(name: String): void + setDoh(dob: String): void + setPhoneNumber(phoneNumber: String): void + setPhoneNumber(phoneNumber: String): void + setEmailAddress; String): void + setBallAddress(emailAddress: String): void + setBloodType(bloodType: String): void + addNewDiagnosis: (rewDiagnosis: String): void + addNewDiagnosis: String): void + setString + getLogged  + ReplenishmentRequest(medicines: List <string>) + ReplenishmentRequest(ind: String) or status: St</string>		+ getEmailAddress(): String		- requestedAmou	unt: int	+ logPayment(ap
+ getPastPrescriptions(): String + getNewDiagnosis(): String + getNewPrescription(): String + setNewPrescription(): String + setName(name: String): void + setDob(dob: String): void + setPhoneNumber(phoneNumber: String): void + setPhoneNumber(phoneNumber: String): void + setEmailAddress(emailAddress: String): void + setBloodType(bloodType: String): void + setBloodType(bloodType: String): void + addNewDiagnosis(newDiagnosis: String): void + setCoveFrenct(): String						+ getLoggedPayr
+ getNewDiagnosis(): String + getNewPrescription(): String + setName(name: String): void + setDob(dob: String): void + setPhoneNumber(phoneNumber: String): void + setEmailAddress; String): void + setBoodType(bloodType: String): void + setBloodType(bloodType: String): void + addNewDiagnosis: String): void + setdNewDiagnosis: String): void + setBloodType(bloodType: String): void + setBloodType(bloodType: String): void + setStatus(string): void + setStatus(string): void + setStatus(string): void + setStatus(string): void		+ getPastPrescriptions(): String				
+ getNewPrescription(): String + setName(name: String): void + setDob(dob: String): void + setPhoneNumber(phoneNumber: String): void + setEmailAddress: String): void + setBloodType(bloodType; String): void + setBloodType(bloodType; String): void + addNewDiagnosis: (newDiagnosis: String): void + setGloodType(bloodType; String): void		+ getNewDiagnosis(): String				
+ setDob(dob: String): void + setPhoneNumber(phoneNumber: String): void + setEmailAddress(emailAddress: String): void + setBloodType(bloodType: String): void + setBloodType(bloodType: String): void + addNewDiagnosis(newDiagnosis: String): void + addNewDiagnosis: String): void + setStatus(status: StatusEnum): void + setStatus(status: StatusEnum): void + setStatus(status: String): void   + setStatus(status: String): void		+ getNewPrescription(): String + setName(name: String): void		+ Replenishment	tRequest(id: String, medicines: List <string>, status: StatusEnum)</string>	
+ setPhoneNumber(phoneNumber: String): void + setEmailAddress: (emailAddress: String): void + setBloodType(bloodType: String): void + setBloodType(bloodType: String): void + addNewDiagnosis: (newDiagnosis: String): void + cCov(Extract): String + setNeductines(): EstCotting> + getNeductines(): EstCotting> + ge		+ setDob(dob: String): void	+ setDob(dob: String): void			
+ setEmailAddress: String): void + setBloodType(bloodType: String): void + setBloodType(bloodType: String): void + setId(id: String): void   + addNewDiagnosis: String): void		+ setPhoneNumber(phoneNumber: String): void	+ setPhoneNumber(phoneNumber: String): void			
+ addNewDiagnosis: String): void				+ setStatus(statu	us: StatusEnum): void	
+ addNewPrescription(newPrescription: String); void		+ addNewDiagnosis(newDiagnosis: String): void		+ setId(id: String)	I): VOIG	
		+ addNewPrescription(newPrescription: String): void		r toosvronnat().		

```
+ AppointmentRequest(String, String, LocalDate, LocalTime, String)
+ getRequestId(): String
+ setRequestId(): String)
+ setPatientId(): String
+ setPatientId(patientId: String): void
+ getDoctorId(): String)
+ setDoctorId(): String): void
+ getDoctorId(): String): void
+ getRequestedDate(): LocalDate
+ setRequestedDate(): LocalDate
+ setRequestedDate(requestedDate: LocalDate): void
+ getRequestedTimeSlot(: LocalTime)
+ setRequestedTimeSlot(: LocalTime)
+ setRequestedTimeSlot(requestedTimeSlot: LocalTime): void
+ getStatus(): String
+ setStatus(status: String): void
+ toString(): String): AppointmentRequest
```

- doctorID: String
- date: LocalDate
- timeSlot: LocalTime
- status: String

+ Schedule(doctorID: String, date: LocalDate, timeSlot: LocalTime, status: String)
+ getDoctorID(): String
+ getDate(): LocalDate
+ getTimeSlot(): LocalTime
+ getStatus(): String
+ setStatus(status: String): void
+ toString(): String

, appointmentId: String,

- name: String
- quantity: int
- status: MedicationStatus

- name: String
- quantity: int

Medication

- name: String
- quantity: int
- status: MedicationStatus
+ Medication(name: String, quantity: int, status: MedicationStatus)
+ getName(): String
+ setName(name: String): void
+ getQuantity(): int
+ setQuantity(quantity: int): void
+ getStatus(): MedicationStatus
+ setStatus(status: MedicationStatus): void
+ toString(): String
+ setMedicationStatus(status: MedicationStatus): void

PaymentLogger

ring, amountPaid: double, serviceProvided: String): void ring>