

بسمه تعالی



دانشگاه صنعتی شریف
دانشکده مهندسی کامپیوتر

سامانه‌ی مطب هوشمند (Smart Office)

توصیف نیازمندی‌های بخش سرپرست (Admin)

گروه ۵

نیما سالم ۹۷۱۰۶۰۰۲

امیرمحمد قاسمی ۹۷۱۰۰۴۹۵

یگانه قره‌داغی ۹۷۱۰۶۲۱۶

حمیدرضا کامکاری ۹۷۱۱۰۱۷۷

سیدعلیرضا هاشمی ۹۷۱۰۲۶۲۲

فهرست

3	۱. مقدمه
3	مروری بر نرم افزار
3	مخاطبان سند
4	۲. معماری
4	کلیت
7	نمودار کلاس
8	نمودار مولفه ها
8	نمودار استقرار
9	۳. نمودار ارتباط موجودیت ها
10	۴. طراحی رابط ها
10	رابط های مربوط به پایگاه داده
11	Notification Management به واسطه های مربوط به NotificationServiceRestController
11	Doctors Management به واسطه های مربوط به DoctorServiceRestController
11	User Management به واسطه های مربوط به AdminServiceRestController
12	AuthenticationService
12	PatientServiceRestController
13	Visit Slot Management به واسطه های مربوط به VisitSlotController
13	Payment Management به واسطه های مربوط به PaymentController
14	۵. طراحی تجربه کاربری و رابط کاربری
15	نرم افزار استفاده شده
15	طراحی های انجام شده
15	ورود
16	صفحات مرتبط با پزشک

18	صفحات مرتبط با کاربران
21	صفحات مرتبط با سرپرستان
22	صفحات مرتبط با پیام
24	۶. ضمیمه
24	۶.۱ ماتریس ردیابی TRM
25	* سناریوهای جدید

۱. مقدمه

هدف از آماده سازی این سند، طراحی یک سامانه ی مطب آنلاین برای تسهیل روند نوبت گیری بیماران، بر اساس نیازمندی های بیان شده در فاز اول پروژه، است. در این سند به طور خاص به طراحی نیازمندی های مطرح شده برای سرپرستان سامانه پرداخته میشود که شامل طراحی معماری، داده ها، واسط ها و واسط نرم افزاری است.

مروری بر نرم افزار

مشخصات نرم افزار	توضیحات
نام	سامانه ی مطب آنلاین
نوع	تحت وب
وضعیت	تحت توسعه
نام پایگاه داده	PostgreSQL

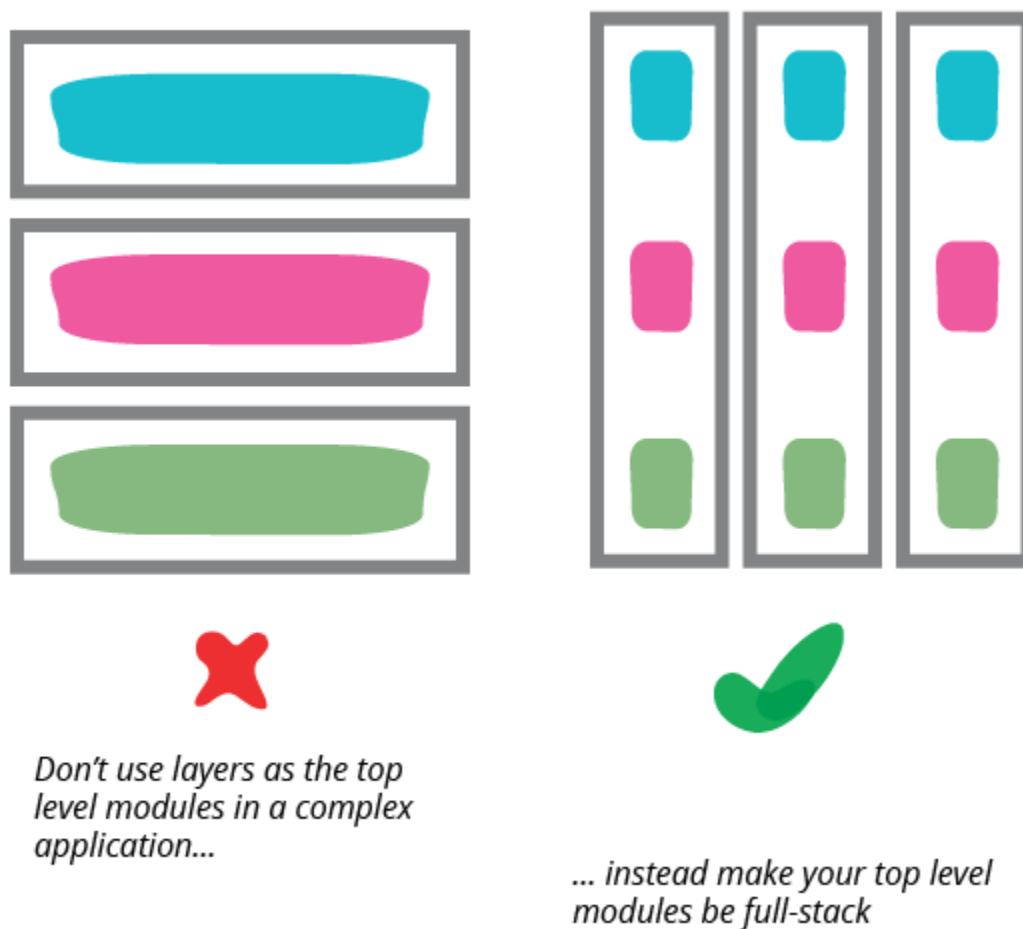
مخاطبان سند

پزشکان و بیماران و تیم توسعه و نگهداری نرم افزار و سرپرستان سامانه، همه از ذینفعان به شمار می آیند.

۲. معماری

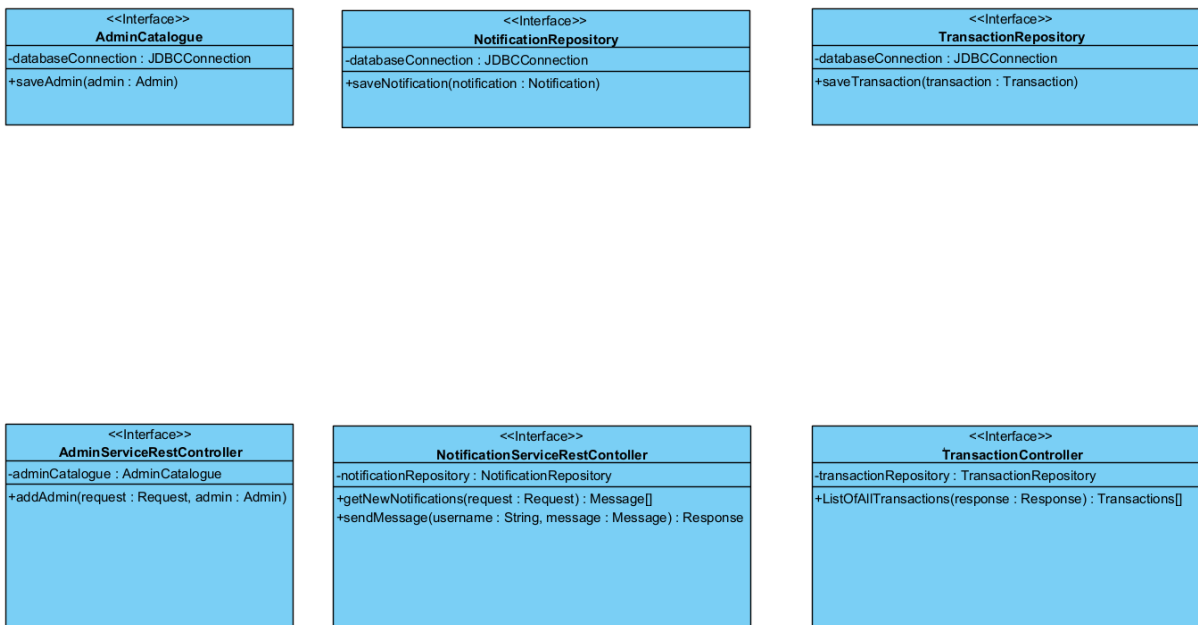
کلیت

سامانه‌های تحت وب معماری‌های مرسوم و جا افتاده‌ای دارند. یکی از این معماری‌ها، معماری چندلایه است (layered). در [این](#) مقاله، مارتین فاولر در مورد این معماری توضیحاتی ارائه میکند. نیازمندی‌های سرپرست سامانه مطب هوشمند، اکثراً از نوع CRUD هستند. در واقع سامانه مطب هوشمند یک سامانه data-intensive است و ذخیره سازی و پایش اطلاعات یکی از مهم‌ترین وظایف این سامانه میباشد.



برای نوع موجودیت‌های اصلی این سامانه، یک پشته مانند پشته عکس بالا در نظر گرفته شده است. به صورت کلی هر نوع موجودیت در سمت سرور، یک controller و یک repository دارد. controller مسئولیت پیاده سازی و مدیریت business logic را دارد. Repository مسئولیت استخراج داده‌ها را از پایگاه‌های داده دارد. لایه نمایش (presentation) نیز در سمت کاربر قرار دارد که با API های سمت سرور ارتباط برقرار میکند.

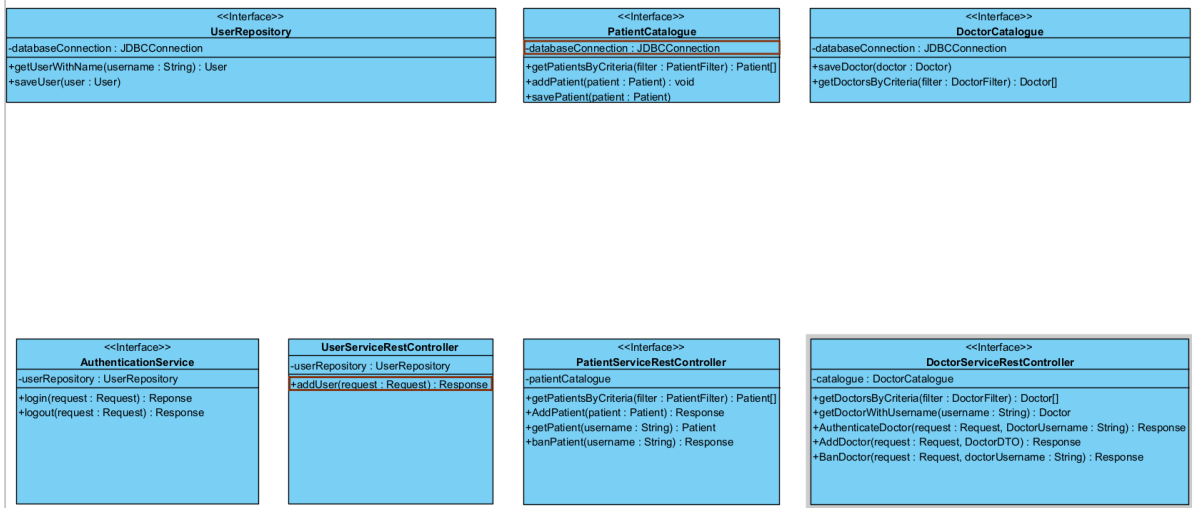
تصویر ۱:



تصویر ۲:

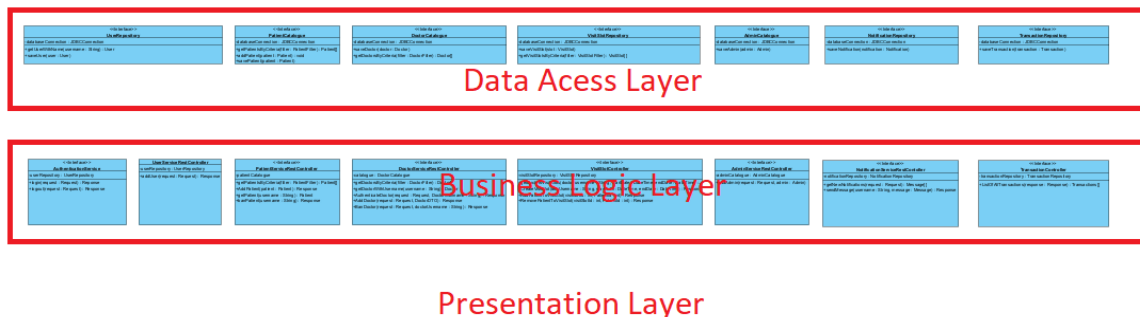


تصویر ۳:



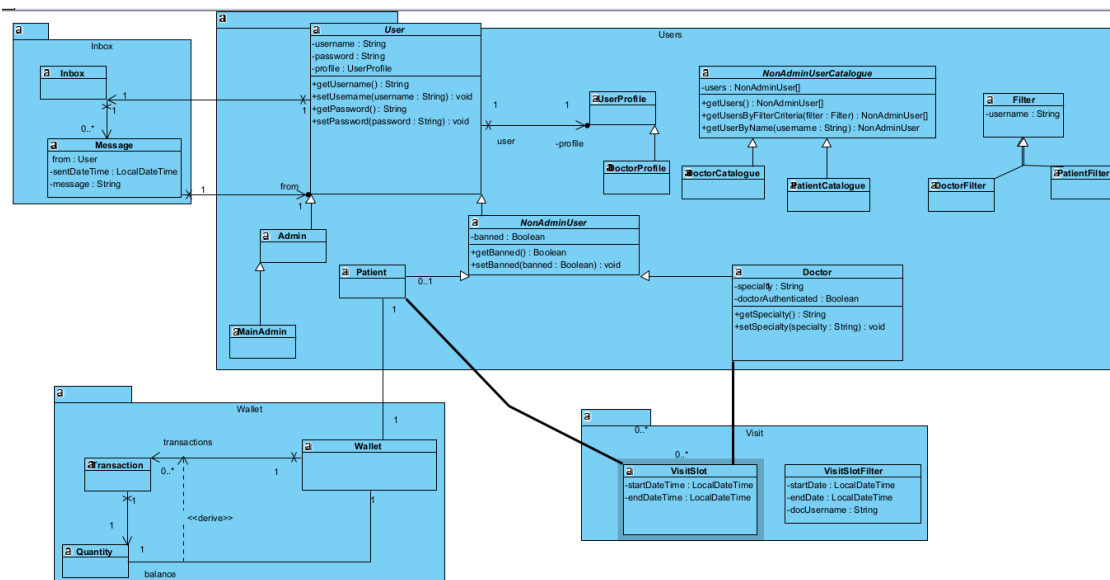
در هر یک از تصاویر بالا قابل مشاهده است که برای یک نوع موجودیت، یک پشته چند لایه‌ای کامل وجود دارد طبق توصیه مارتین فاولر. برای مثال، یک PatientServiceRestController را داریم درخواست‌های Rest را دریافت میکند و با لایه پایین تر خود که PatientCatalogue است ارتباط برقرار میکند برای تغییر داده‌های مربوط به بیماران سامانه. PatientCatalogue در واقع وظیفه ارتباط با پایگاه داده را دارد و واسطه پایگاه داده و لایه business است.

در یک قاب کلی تر، تمام interface های تعریف شده بالا، تصویری مانند شکل زیر دارند که در آن لایه ها بسیار واضح قابل مشاهده اند.



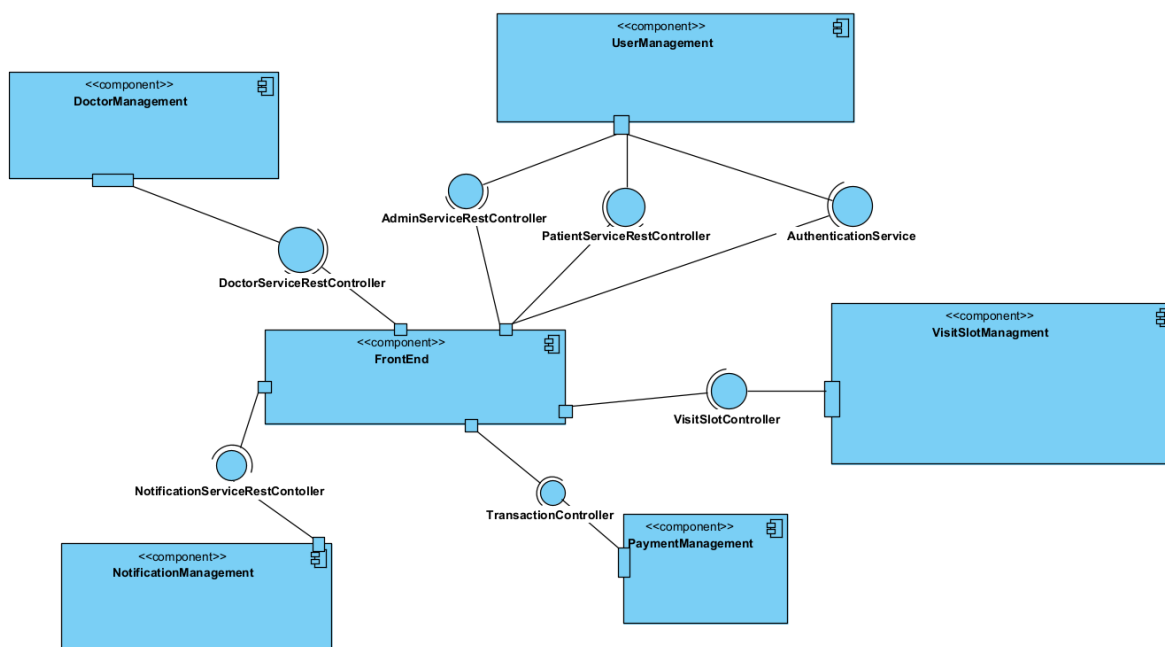
نمودار کلاس

نمودار کلاس‌ها در زیر آمده است. برای بعضی از کلاس‌ها، operation هایی آمده. البته operation ها کامل نیستند اما کلاس‌ها و ارتباط آن‌ها در زیر نمایش داده شده است. در واقع نمودار کلاس زیر، بیشتر شبیه نمودار کلاس تحلیل میباشد.



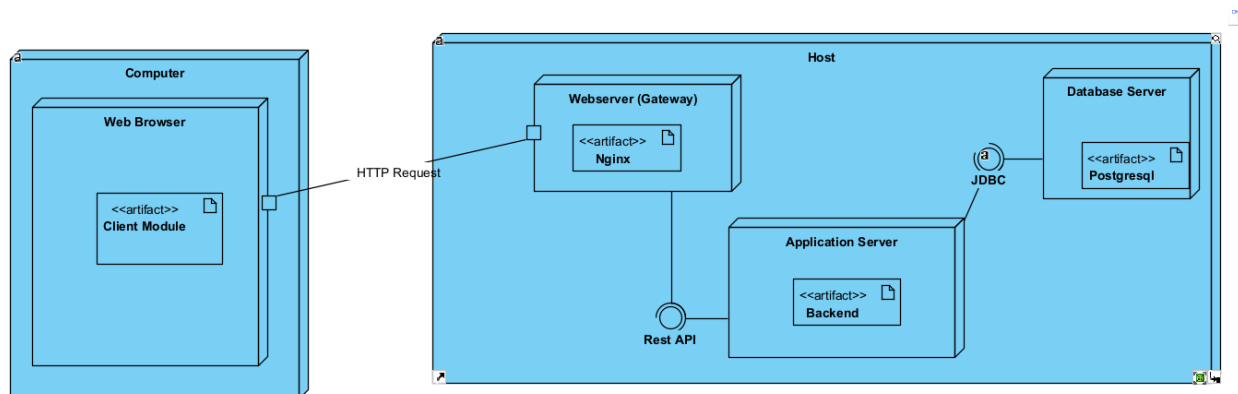
نمودار مولفه‌ها

به طور کلی مولفه‌های سامانه به ۵ بخش مدیریت کاربر، مدیریت پزشک، مدیریت پیام‌ها، مدیریت نوبت‌ها و مدیریت مال تقسیم می‌شود. وظیفه هر کدام از این ما مولفه‌ها با توجه به نامش گویا می‌باشد.



نمودار استقرار

برای استقرار نمونه اولیه سامانه، از نمودار استقرار زیر که بسیار ساده می‌باشد استفاده می‌شود. یک سرور برای استقرار سامانه کافی است. از پایگاه داده PostgreSQL برای ذخیره‌سازی داده‌ها استفاده می‌شود.

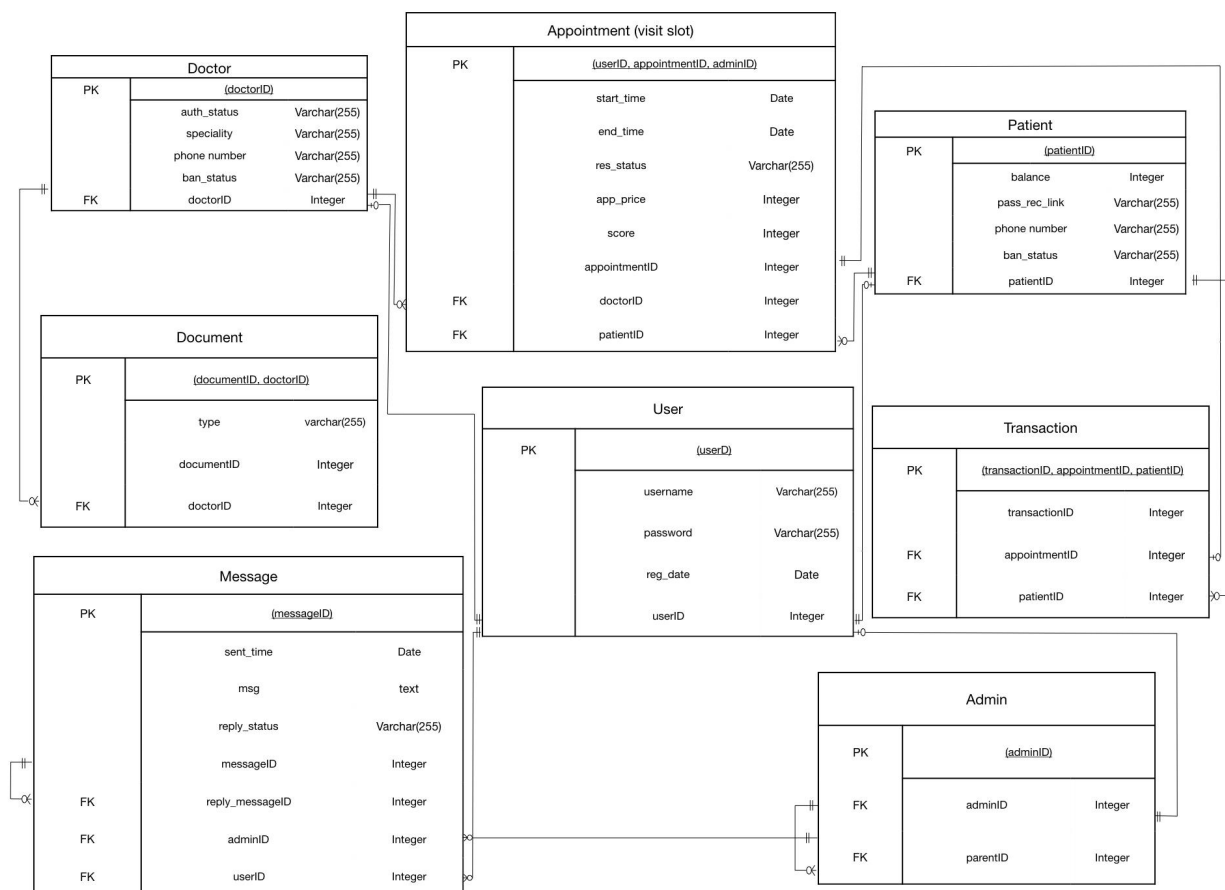


۳. نمودار ارتباط موجودیت‌ها

در شکل زیر می‌توان نمودار ارتباط موجودیت‌های این سامانه را مشاهده کرد. نمودار شامل ۸ موجودیت است که هر کدام با برخی از دیگر موجودیت‌ها رابطه چند یا یک (∞)، یک و تنها یک ($||$) و یک یا صفر ($0+$) دارد. در ادامه به توصیف نیازمندی‌ها توسط این نمودار می‌پردازیم:

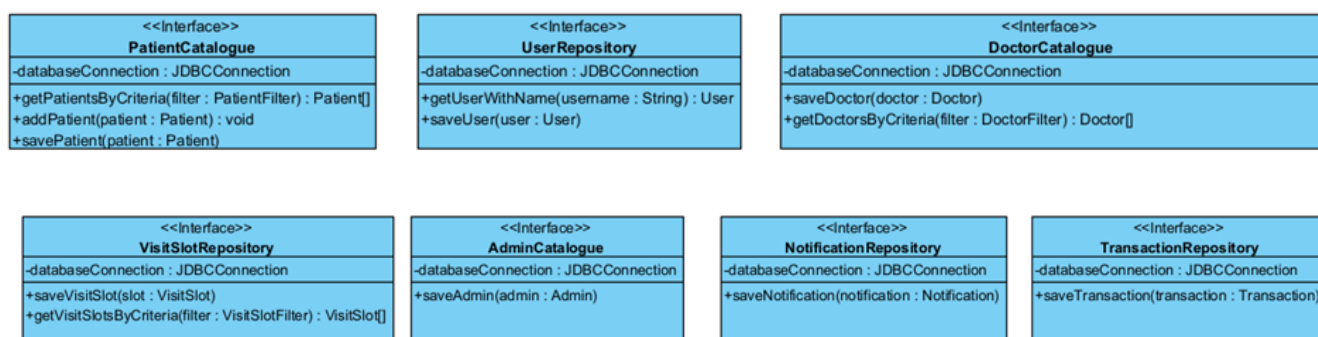
1. هر پزشک تعدادی وقت با قیمت‌های تعیین شده دارد که مراجعین می‌توان آن را انتخاب کنند، هزینه آن را پرداخت کنند و پس از مراجعه امتیازی به آن بدهند.
2. سرپرست می‌تواند فعالیت کاربرها را (اعم از پزشک و مراجعه‌کننده) قدغن کند.
3. سرپرست می‌تواند سرپرست جدیدی را اضافه کند.
4. سرپرست‌ها می‌توانند پیامی به یک کاربر ارسال کنند یا پاسخ او را بدهند (و برعکس).
5. پزشکان قادر به اضافه کردن مدارک خود هستند و همچنین توسط سرپرست تایید می‌شوند.
6. موجودی مراجعه‌کنندگان قابل مشاهده است.

در نهایت با توجه به توصیفات بالا، می‌توان نوع (type) هرکدام از متغیرها را تعیین کرد.

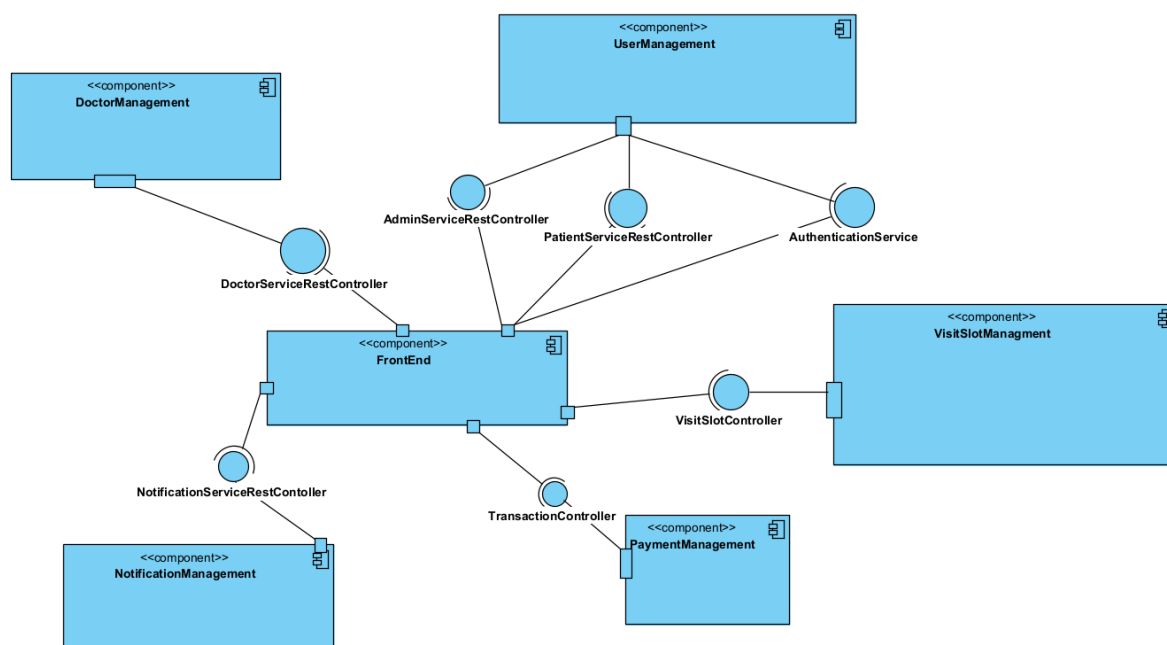


۴. طراحی رابط‌ها

رابط‌های مربوط به پایگاه داده

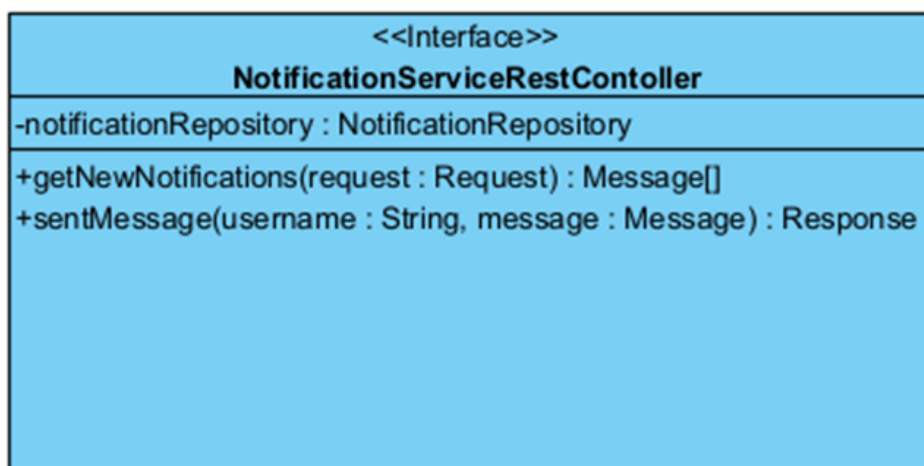


به طور کلی با توجه به component diagram زیر ، نیاز به هفت مجموعه واسط داریم که در ادامه هر یک، به ترتیب توضیح داده شده است.



واسط های مربوط به Notification Management

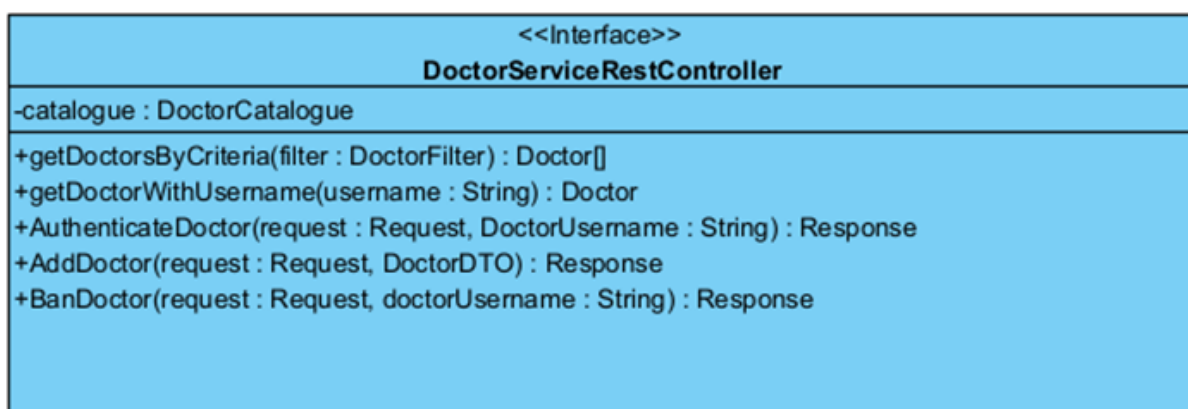
NotificationServiceRestController



در این بخش واسط هایی وجود دارند که برای ارسال پیام به کاربر و همچنین بازبینی پیام های جدید فرستاده شده، استفاده میشوند.

واسط های مربوط به Doctors Management

DoctorServiceRestController



این واسط ها برای برآورده کردن نیازمندی های سرپرستان در قبال پزشک ها طراحی شده اند که شامل موارد زیر میشوند.

- به دست آوردن لیست پزشکان بر اساس معیار های تعیین شده
- گرفتن اطلاعات یک پزشک با کمک نام کاربری
- احراز اصالت یک پزشک
- اضافه کردن یک پزشک به سامانه
- مسدود کردن فعالیت یک پزشک

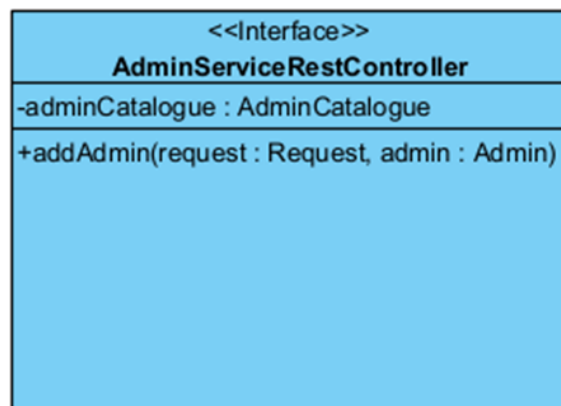
واسط های مربوط به User Management

AdminServiceRestController



این واسط برای اضافه کردن یک فرد به عنوان سرپرست استفاده میشود.

AuthenticationService



این واسط برای احراز هویت و ورود و خروج کاربران از سامانه استفاده میشود.

PatientServiceRestController

<<Interface>>	
PatientServiceRestController	
-patientCatalogue	
+getPatientsByCriteria(filter : PatientFilter) : Patient[]	
+AddPatient(patient : Patient) : Response	
+getPatient(username : String) : Patient	
+banPatient(username : String) : Response	

این واسط ها برای برآورده کردن نیازمندی های سرپرستان در قبال بیماران طراحی شده اند که شما موارد زیر است.

- به دست آوردن بیمارانی با یک مشخصه ی خاص
- به دست آوردن اطلاعات یک بیمار با کمک نام کاربری
- مسدود کردن یک بیمار
- اضافه کردن یک بیمار به سامانه

واسط های مربوط به Visit Slot Management

VisitSlotController

<<Interface>>	
VisitSlotController	
-visitSlotRepository : VisitSlotRepository	
+getDoctorsVisitSlots(doctor username : String, startDate : DateTime, endDate : DateTime) : VisitSlot[]	
+CreateVisitSlot(docUsername : String, startDate : DateTime, endDate : DateTime) : Response	
+AddPatientToVisitSlot(visitSlotId : int, PatientId : int) : Response	
+RemovePatientToVisitSlot(visitSlotId : int, PatientId : int) : Response	

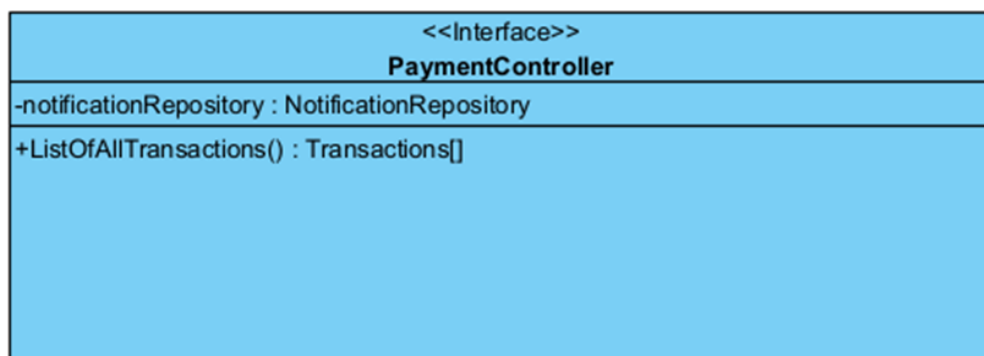
این واسط ها برای مدیریت نوبت های رزرو شده استفاده می شود که شامل موارد زیر است.

- به دست آوردن نوبت های یک پزشک با کمک نام کاربری پزشک و زمان آن
- ساختن یک نوبت برای یک پزشک
- رزرو کردن یک بیمار برای یک نوبت

- کنسل کردن نوبت یک بیمار

واسط های مربوط به Payment Management

PaymentController



این واسط برای مشاهده ی لیست تراکنش های انجام شده استفاده میشود.

۵. طراحی تجربه کاربری و رابط کاربری

نرم افزار استفاده شده

برای طراحی رابط کاربری از نرم افزار Figma استفاده کردیم. می توانید تمام طراحی های انجام شده برای بخش ادمین این پروژه را در [این لینک](#) زیر مشاهده کنید.

طراحی های انجام شده

ورود



تصویر ۵.۱ - صفحه ی ورود

صفحات مرتبط با پزشک

سامانه مطب آنلاین

پزشکان

سرپرست سرپرست زاده

لیست پزشکان

نام آخرین بازدید وضعیت امتیاز

حمیدرضا کامکاری متخصص بی‌هوشی	۱۷ مرداد ۱۴۰۱ ۲۱۳۵	تایید نشده	★★★★★
یگانه قره‌دانی متخصص بی‌هوشی	۱۷ مرداد ۱۴۰۱ ۲۱۳۵	تایید شده	★★★★★

پزشکان
کاربران
سرپرستان
پیامها

تصویر ۵.۲ - لیست پزشکان سامانه

سامانه مطب آنلاین

مشخصات

سرپرست سرپرست زاده

مشخصات پزشک

نام خانوادگی نام تخصص وضعیت تایید تایید نشده نام کاربری

بارگذاری عکس

ذخیره

پزشکان
کاربران
سرپرستان
پیامها

تصویر ۵.۳ - افزودن پزشک به سامانه

سرپرست

سرپرست زاده

مشخصات

سامانه مطب آنلاین

نوبت‌ها

مشخصات

مشخصات پزشک

نام خانوادگی

نام

کامکاری

حمیدرضا

کد نظام پزشکی

تخصص

123456789

بی‌هوشی

شهر

وضعیت ناپید

قزوین

ناپید نشده

نام کاربری

HMKamkari

بارگذاری عکس

ذخیره

مسدود کردن

پزشکان

کاربران

سرپرستان

پیام‌ها

تصویر ۵.۴ - مشاهده‌ی مشخصات یک پزشک

سرپرست

سرپرست زاده

مشخصات

سامانه مطب آنلاین

نوبت‌ها

مشخصات

نوبت‌ها

جدا سازی

مرتب‌سازی

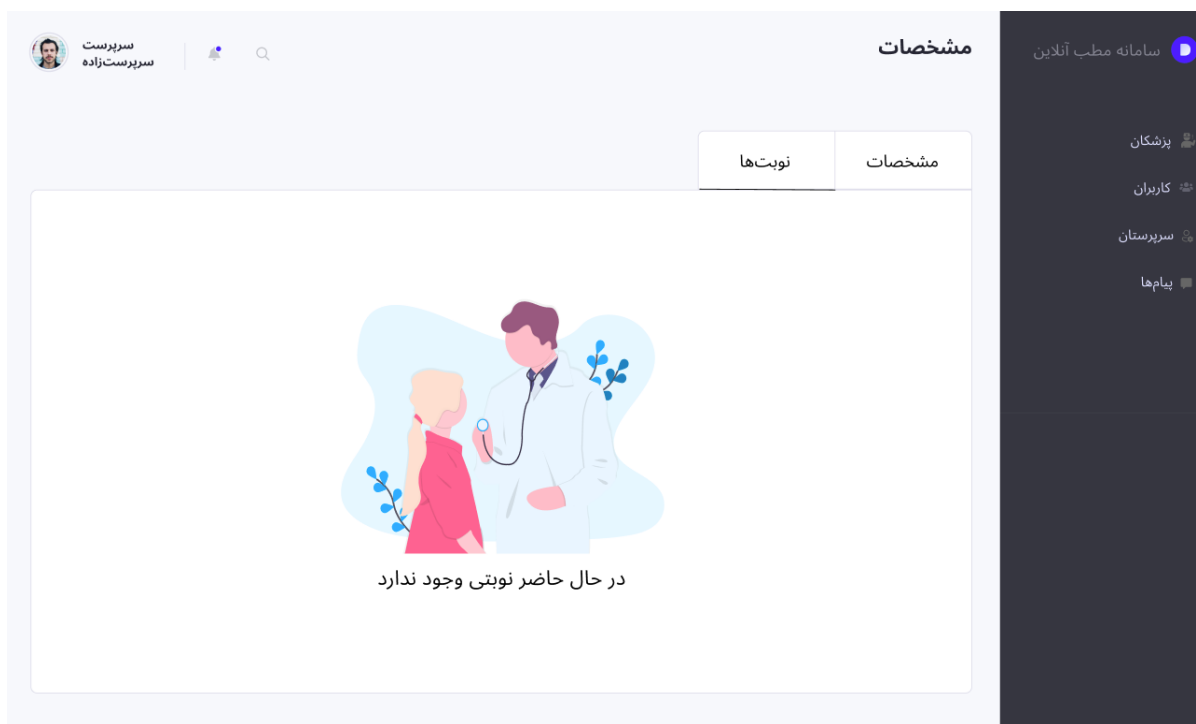
نام بیمار

سن

زمان

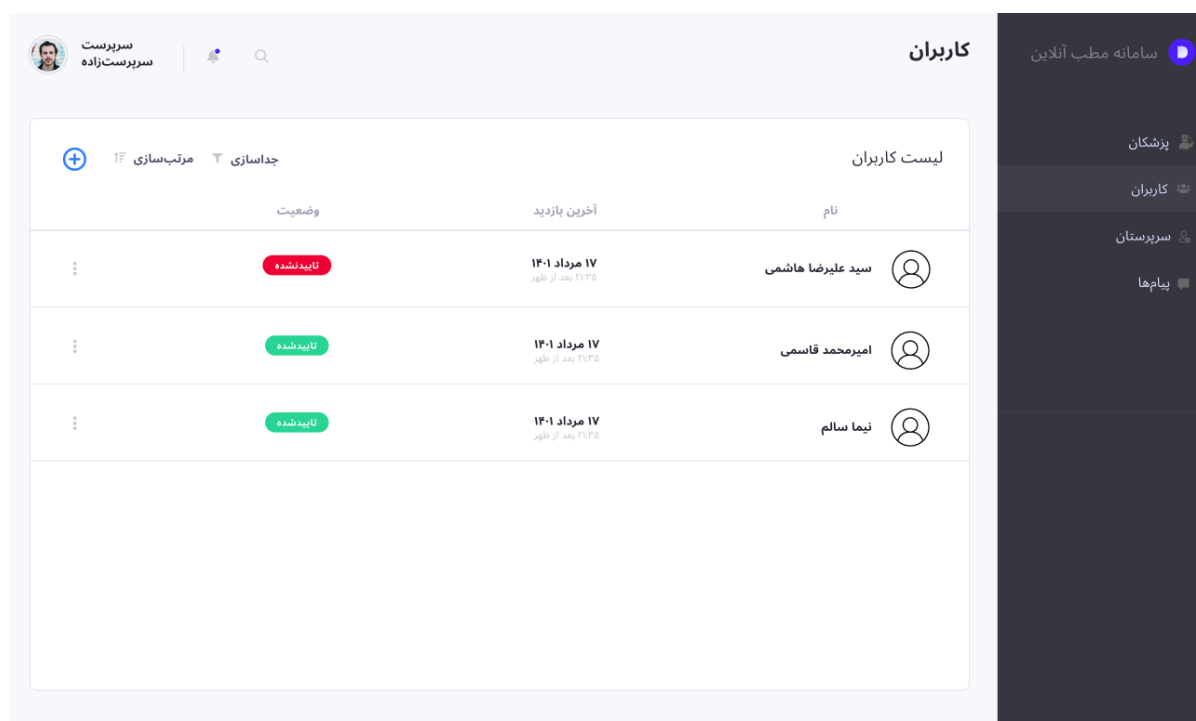
سید علیرضا هاشمی	۲۳	۱۷ مرداد ۱۴۰۱
------------------	----	---------------

تصویر ۵.۵ - مشاهده‌ی نوبت‌های یک پزشک




تصویر ۵.۶ - صفحه‌ی نوبت‌های یک پزشک هنگامی که نوبتی نداشته باشد

صفحات مرتبط با کاربران



تصویر ۵.۷ - مشاهده‌ی لیست کاربران سامانه



سرپرست
سرپرست زاده


🔍

مشخصات

سامانه مطب آنلاین



		نوبت‌ها	تراکنش‌ها	مشخصات
<div style="display: flex; justify-content: space-between;"> رتبه سازی 📊 جداسازی ⚡ </div>				
	وضعیت	زمان	موجودی	نوع عملیات
⋮	موفق	۱۴ مرداد ۱۴۰۱ <small>نیرو</small>	۱۲۴.۵۰۰	-۳۰.۰۰۰ پرداخت حق ویزیت
⋮	موفق	۱۷ مرداد ۱۴۰۱ <small>نیرو</small>	۱۵۴.۵۰۰	+۵۰.۰۰۰ شارژ کیف پول

تصویر ۵.۱۰ - مشاهده‌ی تراکنش‌های یک کاربر



سرپرست

سرپرست زاده

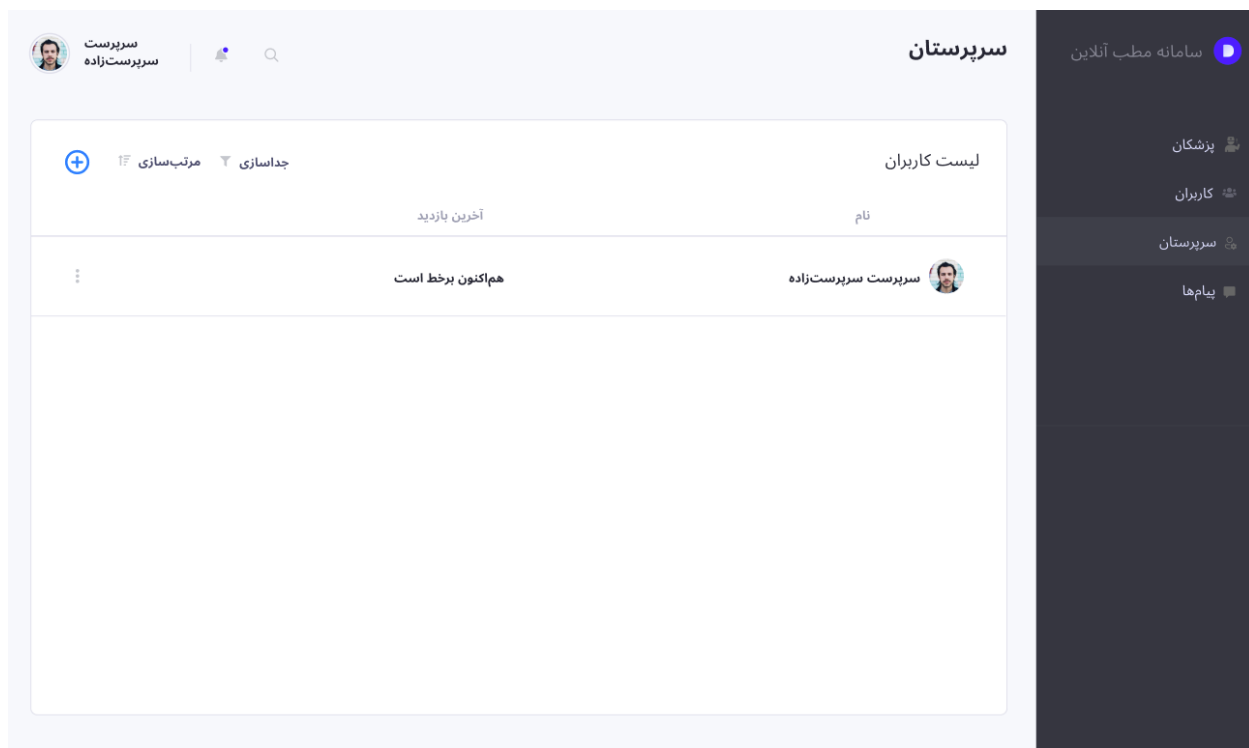
مشخصات

سامانه مطب آنلاین

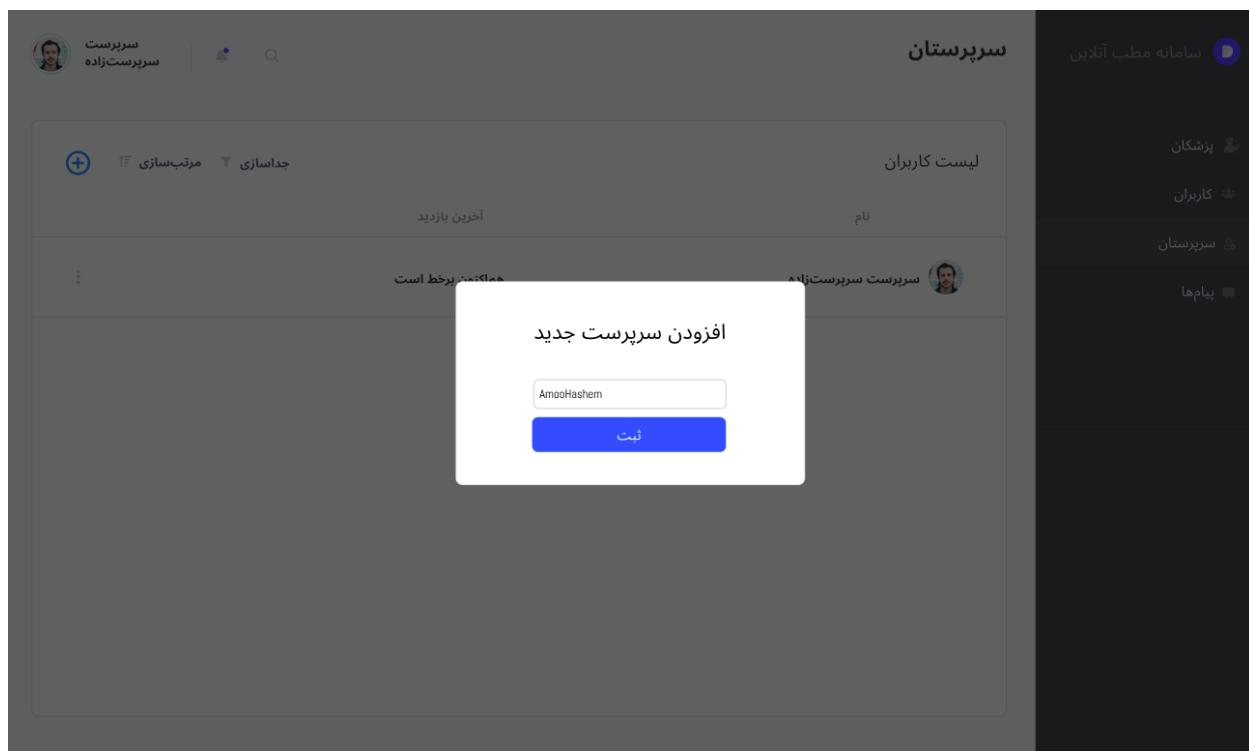
نوبت‌ها	تراکنش‌ها	مشخصات
نوبت‌ها جدا سازی ▼ مرتب سازی ⚙️		
زمان	تخصص	نام دکتر
۱۷ مرداد ۱۴۰۱ <small>TUE</small>	بی هوشی	حمیدرضا کامکاری

تصویر ۵.۱۱ - مشاهده‌ی نوبت‌های یک کاربر

صفحات مرتبط با سرپرستان



تصویر ۵.۱۲ - مشاهده لیست سرپرستان



تصویر ۵.۱۳ - افزودن سرپرست جدید به سامانه

صفحات مرتبط با پیام

پیام

ارسال پیام مستقیم

پیام خود را اینجا بنویسید...

انتخاب گیرنده

انصراف ارسال

سامانه مطب آنلاین

پزشکان

کاربران

سرپرستان

پیامها

تصویر ۵.۱۴ - ارسال پیام مستقیم به یک کاربر یا دکتر

پیامها

لیست پیامهای کاربران

وضعیت

زمان ارسال

فرستنده

نیما سالم

۱۷ مرداد ۱۴۰۱

در دست بررسی

سید علیرضا هاشمی

۱۷ مرداد ۱۴۰۱

پاسخ داده شده

سامانه مطب آنلاین

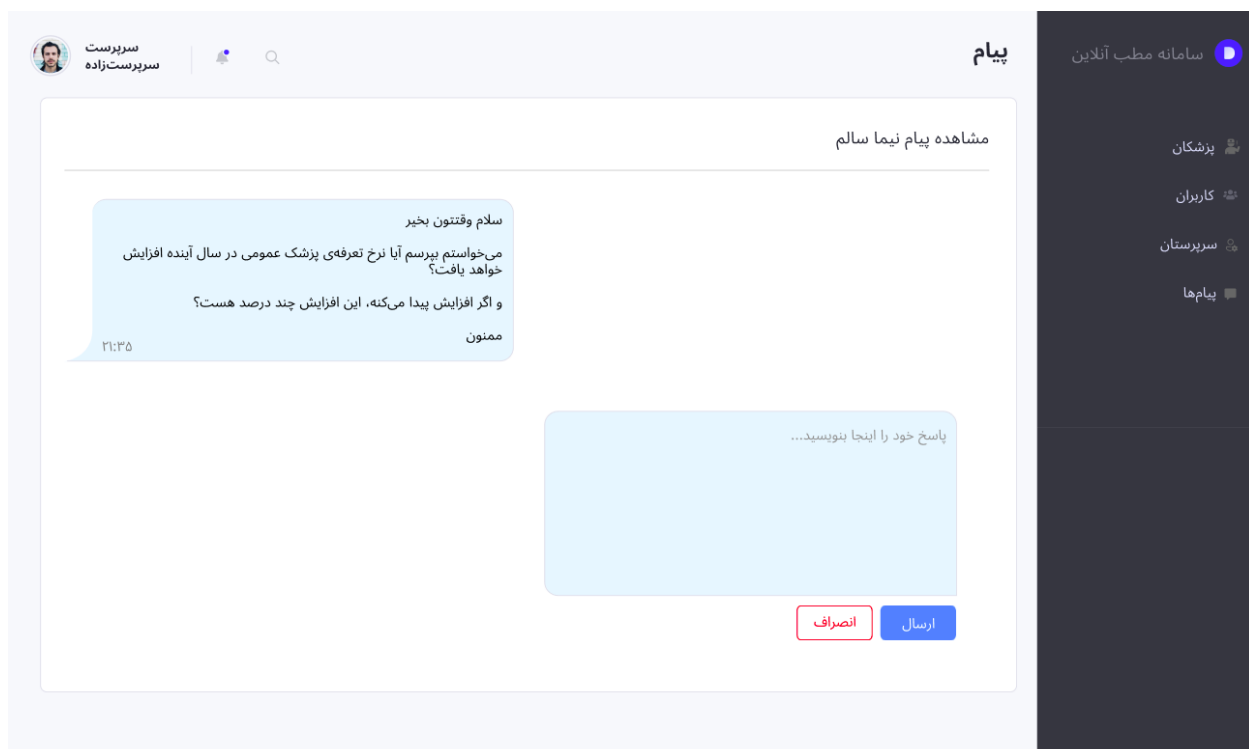
پزشکان

کاربران

سرپرستان

پیامها

تصویر ۵.۱۵ - مشاهده لیست پیامهای ارسال شده به سرپرست



تصویر ۵.۱۶ - پاسخ به یک پیام ارسال شده

۶. ضمیمه

۶.۱ ماتریس ردیابی TRM

ماتریس ردیابی تناظری بین نیازمندها و سناریوهای سیستم و مدل‌های ارائه شده در این سند را برقرار می‌کنند. در این ماتریس شماره سناریوها در ستون اول قرار گرفته‌است، نوع کلاس‌ها و اشیا مورد استفاده هر سناریو در ستون دوم قرار گرفته‌است و نوع رابط‌های مورد استفاده در ستون سوم قرار گرفته. هر نیازمندی با یک شناسه مشخص شده‌است و به علت ایجاد نیازمندی‌های جدید تعدادی شناسه جدید در انتهای این جدول اضافه کردیم که با علامت ستاره در کنارشان مشخص شده‌اند. برای اطلاعات بیشتر به سند SRS ارائه شده توصیف نیازمندی‌های مطب هوشمند نسخه ۱.۰ مراجعه نمایید.

شماره سناریو	کلاس‌ها و اشیا مورد استفاده	رابط‌های مورد استفاده	طراحی رابط کاربری
(۱) ورود و خروج به سامانه توسط سرپرست	Class: User Class: Admin	AuthenticationService.login AuthenticationService.logout UserRepository.getUserWithName	تصویر ۵.۱
(۲) بازیابی لیستی از پزشک‌ها	Class: User Class: NonAdminUser Class: Doctor	DoctorCatalog.getDoctorbyCriteria	تصویر ۵.۲
(۳) بازیابی اطلاعات مربوط به یک پزشک	Class: Doctor.getters() Class: User Class: NonAdminUser Class: Document.getters()	DoctorServiceRestController.getDoctorByUser name	تصویر ۵.۴ و ۵.۵ و ۵.۶
(۴) تایید یا تغییر پروفایل پزشک	Class: Doctor.setters() Class: User Class: NonAdminUser Class: Document.setters()	DoctorServiceRestController.getDoctorByUser name DoctorCatalog.saveDoctor	تصویر ۵.۴
(۵) ثبت نام پزشک جدید	Class: Doctor Class: User Class: NonAdminUser	DoctorServiceRestController.addDoctor DoctorCatalog.saveDoctor	تصویر ۵.۳
(۶) قذف کردن پزشک	Class: User Class: NonAdminUser.banned	DoctorServiceRestController.getDoctorByUser name DoctorCatalog.saveDoctor	تصویر ۵.۴
(۷) اضافه کردن مراجعه‌کننده جدید	Class: User Class: NonAdminUser Class: Patient	PatientServiceRestControoller.addPatient PatientCatalog.savePatient	تصویر ۵.۸

تصویر ۵.۷	PatientServiceRestController.getpatientByCriteria	Class: Patient Class: User Class: NonAdminUser	(۸) بازیابی گروهی از کاربران بر اساس یک سری فیلترهای اعمال شده در فرآیند جست‌وجو
تصویر ۵.۹ و ۵.۱۰ و ۵.۱۱	PatientServiceRestController.getpatient	Class: Doctor.getters() Class: User Class: NonAdminUser	(۹) بازیابی تمامی اطلاعات مراجعه‌کننده
تصویر ۵.۹	PatientServiceRestController.getPatientByUsername PatientCatalog.savePatient	Class: User Class: NonAdminUser.banned	(۱۰) قدغن کردن مراجعه‌کننده
رابط کاربری این نیازمندی مشابه طرحی است که گروه پزشک باید بزند (ما گروه ادمین بودیم)	VisitSlotController.getDoctorsVisitSlots	Class: VisitSlot Class: VisitSlotFilter	(۱۱) بازیابی برنامه پزشک در یک بازه
تصویر ۵.۵	PatientServiceRestController.addPatientToVisitSlot PatientServiceRestController.removePatientToVisitSlot	Class: VisitSlot	(۱۲) رزرو کردن یک قرار ملاقات یا کنسل کردن
تصویر ۵.۱۳	AdminServiceRestController.addAdmin AdminCatalog.saveAdmin	Class: User Class: Admin	(۱۳) اضافه کردن سرپرست جدید
تصویر ۵.۱۵ و ۵.۱۶	NotificationServiceRestController.getNewNotifications	Class: Message	(۱۴) نمایش پیام‌های دریافتی
تصویر ۵.۱۴ و ۵.۱۶	NotificationServiceRestController.sendMessage NotificationRepository.saveNotification	Class: Message	(۱۵) ارسال پیام
رابط کاربری این نیازمندی مشابه طرحی است که گروه کاربر باید بزند (ما گروه ادمین بودیم)	TransactionRepository.saveTransaction	Class: Transaction Class: Wallet Class: Quantity	(۱۶) انجام تراکنش
تصویر ۵.۱۰	TransactionController.listOfAllTransactions	Class: Transaction Class: Quantity	(۱۷) بازیابی تراکنش

* سناریوهای جدید