

بسمه تعالی



## 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

2	فهرست
3	مقدمه
3	هدف
3	محدوده
3	ذینفعان
4	منابع
5	توضیحات سامانه
6	ویژگی‌های سیستم
9	نیازمندی‌های واسط خارجی
9	واسط‌های کاربری
9	واسط‌های نرم افزاری
10	نیازمندی‌های غیرکارکردی
10	performance & scalability معیار
10	portability & compatibility معیار
10	availability معیار
10	Usability معیار
11	نیازمندی‌های دیگر
12	روشن‌شناسی توسعه
14	ضمیمه
14	واژه‌نامه
15	مدل نیازمندی‌ها
31	اولویت‌بندی نیازمندی‌ها

## مقدمه

سندی که در زیر می‌آید، سند توصیف نیازمندی‌های نرم‌افزار سامانه **مطب آنلاین** است. سند توصیف نیازمندی‌های نرم‌افزار به تشریح و تفصیل نیازمندی‌های یک سامانه نرم‌افزاری از دید ذینفعان متفاوت سامانه می‌پردازد. این سند به ذکر نیازمندی‌های یک سامانه، تحت وب نوبت دهی و ملاقات پزشکی می‌پردازد.

## هدف

سامانه **مطب آنلاین** به منظور تسهیل امر ملاقات پزشک و نوبت دهی پزشکی ایجاد میشود. در روند دیجیتال شدن سرویس‌های مختلف، نیاز به یک سرویس آنلاین ملاقات پزشکی حس میشود. به همین منظور، ما قصد داریم تا سامانه‌ای ایجاد کنیم که بیماران بتوانند بدون نیاز به مراجعه به مطب یک پزشک، از خدمات او بهره ببرند. به لطف پیشرفت تکنولوژی و نفوذ اینترنت در زندگی امروزی ما، این امکان میسر شده است تا خدمات پزشکی به مرحله‌ی بعد بروند.

با خدمات الکترونیک تامین اجتماعی، در حال حاضر امکان نوبت‌گیری پزشکی برای بیمارستان‌های کشور به صورت آنلاین وجود دارد. با مراجعه به سایت [nobat.tamin.ir](http://nobat.tamin.ir) میتوانید نوبت پزشکی رزرو کنید. همچنین خدمات بخش خصوصی در این زمینه به تازگی به وجود آمده. برای مثال اسنپ، خدمات پزشکی خود را با نام اسنپ دکتر ارائه میکند که شامل ارتباط و مشاوره با پزشک متخصص به صورت آنلاین است.

## محدوده

بخشی که گروه ما به ایجاد و توسعه آن می‌پردازد، بخش سرپرست (admin) است. در نظر ما این است که که سرپرستان سامانه، باید بتوانند از طرف پزشکان و بیماران در سامانه تغییر ایجاد کنند و در صورت بروز مشکل برای پزشکان یا دیگر کاربران سامانه، مشکلات آن‌ها را برطرف کنند. در واقع پشتیبانی از سامانه از وظایف اصلی آنان است. پس با این اوصاف سرپرستان سامانه باید بتوانند از طرف بیماران، برای آن‌ها وقت بگیرند، پزشکان را در سامانه ثبت‌نام یا تایید کنند، کاربران را غیر فعال کنند، از طرف کاربران برای آن‌ها وقت بگیرند، برای پزشکان نوبت زمانی ایجاد کنند و ... .

## ذینفعان

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

## منابع

سایت رزرو نوبت تامین اجتماعی: <https://nobat.tamin.ir>

سایت اسنپ دکتر: <https://snapp.doctor>

## توضیحات سامانه

در سامانه مطب آنلاین سه گونه کاربر پزشک، بیمار و سرپرست را داریم. پزشکان و بیماران به صورت جداگانه می‌توانند در سامانه نام نویسی کنند یا توسط سرپرستان به سامانه اضافه شوند. سرپرستان تقریباً تمام دسترسی‌های موجود در سامانه را دارند و می‌توانند از طرف بیماران یا پزشک‌ها، تغییراتی در سامانه ایجاد کنند. پزشکان در این سامانه زمان‌های کاری‌شان را به صورت بازه‌هایی با طول ثابت در سامانه مشخص می‌کنند. بیماران از سوی دیگر می‌توانند برای رزرو نوبت، ابتدا در بین پزشک‌های یک تخصص خاص، یکی را انتخاب کنند و از بازه‌های خالی او، یک نوبت را رزرو کنند و هزینه‌ی ویزیت را با کیف پول خود در سامانه پرداخت کنند.

# ویژگی‌های سیستم

## 1. ورود به سامانه و خروج از آن

سرپرستان سامانه باید بتوانند به وارد کردن نام کاربری و رمز عبور خود در سامانه بتوانند به آن وارد شوند. همچنین امکان خروج از سامانه نیز وجود دارد.

## 2. نمایش لیست پزشکان

پزشکان سامانه باید در قالب یک لیست قابل نمایش باشد. سامانه باید از قابلیت‌های مورد نیاز برای جستجو در این لیست پشتیبانی کند، مانند جستجو بر اساس تخصص، وضعیت تایید شده یا نشده، امتیاز و ... .

## 3. مشاهده مشخصات و مدارک یک پزشک

سرپرستان باید بتوانند تا با انتخاب یک پزشک از لیست ذکر شده در مورد قبل، به صفحه مشخصات یک پزشک بروند. در این صفحه سرپرستان می‌توانند اطلاعات یک پزشک را ببینند. همچنین امکان مشاهده مدارک وارد شده توسط پزشک در هنگام ثبت نام، وجود دارد.

## 4. تایید پزشک یا ویرایش اطلاعات پزشک

پزشکان پس از ثبت نام در سامانه باید منتظر احراز هویت توسط سرپرستان سامانه باشند. پس سرپرستان سامانه، باید بتوانند لیست پزشکان احراز هویت نشده را ببینند. سرپرستان در صفحه مورد قبل باید دسترسی تایید پزشک یا ویرایش بعضی از اطلاعات او را داشته باشند. در صورت تایید پزشک، پیامکی از طرف سامانه به پزشک ارسال می‌شود.

## 5. افزودن پزشک

سرپرستان سامانه باید بتوانند با در دست داشتن مدارک و اطلاعات یک پزشک، او را در سامانه ثبت کند. در این حالت، پزشک به صورت خودکار تایید میشود و پیامک تایید به او ارسال می‌شود.

## 6. بستن حساب پزشکان

یک سرپرست باید بتواند در صفحه پزشک که در مورد ۳ آمد، توانایی بستن حساب پزشک را داشته باشد.

#### 7. افزودن کاربر

سرپرستان باید بتوانند با در دست داشتن مدارک یک کاربر، او را در سامانه ثبت نام کند. پس از ثبت نام کاربر، لینک بازیابی رمز عبور کاربر، به او ارسال می شود.

#### 8. مشاهده لیست کاربران

بایستی بخشی در سامانه باشد که سرپرستان بتوانند لیست آن ها را مشاهده کنند یا در میان آن ها جستجو کنند.

#### 9. مشاهده مشخصات کاربر

سرپرست با انتخاب هر مورد از لیست مورد قبل، باید به صفحه مشخصات یک کاربر را ببیند. این صفحه شامل مشخصات کاربر، لیست تراکنش های آن، لیست نوبت های اخیرا گرفته شده او و موجودی کیف پولش است.

#### 10. بستن حساب کاربر

چندی از کاربران سامانه ممکن است بخواهند از سامانه سوء استفاده کنند. سرپرستان باید بتوانند حساب یک کاربر را در صورت نیاز ببندند تا از خدمات سامانه محروم شود.

#### 11. مشاهده وضعیت نوبت های یک پزشک

هر پزشک در این سامانه، میتواند بازه های کاری اش را به صورت یک جدول زمانی در سامانه تعیین کند. از سوی دیگر، بیماران با گزینش بازه های خالی یک پزشک، نوبت های او را رزرو میکنند. سرپرستان در صفحه ای باید نوبت های پزشک را بتوانند مشاهده کنند. با انتخاب هر بازه از نوبت های پزشک، سرپرستان باید نام بیمار رزرو کننده و مشخصات آن را بتوانند مشاهده کنند.

#### 12. رزرو یا حذف نوبت از طرف بیمار

سرپرستان باید امکان رزرو یک بازه ای زمانی یک پزشک را از طرف هر بیماری داشته باشند. همچنین امکان حذف نوبت نیز تا قبل از یک روز از بازه ای زمان باید وجود داشته باشد. این امکان در صفحه توصیف شده در مورد قبل وجود دارد. هنگام رزرو نوبت، هزینه ی ویزیت از کیف پول کاربر باید کم شود. در صورت حذف نوبت، مبلغ ویزیت به کاربر پس گردانده می شود.

#### 13. اضافه کردن سرپرست جدید

در ابتدا سامانه یک سرپرست اصلی دارد. سرپرست اصلی میتواند در یک صفحه خاص که در سامانه وجود دارد، سرپرست های دیگری به سامانه اضافه کند به این صورت که یک نام کاربری و رمز عبور را

مشخص میکند و از این به بعد، این نام کاربری یک سرپرست جدید خواهد بود.

#### **14. مشاهده لیست پیام‌های دریافت شده**

پزشکان و بیماران سامانه در صورت بروز مشکل نیاز با ارتباط با سرپرستان سامانه را دارند تا در صورت بروز هرگونه مشکلی، بتوانند آن را بررسی یا رفع کنند. به همین منظور، بخشی در سامانه باید وجود داشته باشد که لیست پیام‌های دریافت شده سرپرستان را نمایش دهد. این لیست باید قابلیت فیلتر بر اساس تاریخ یا پیام‌های پاسخ داده شده/ نشده را دارا باشد.

#### **15. ارسال پاسخ پیام پشتیبانی یا ارسال نوتیفیکیشن به کاربر یا پزشک**

سرپرست باید بتواند به پیام‌های بی پاسخ پشتیبانی پاسخ بدهد. همچنین پزشک می‌تواند پیامی را برای کاربری خاص یا پزشکی خاص ارسال کند که این پیام در لیستی از نوتیفیکیشن‌های کاربر باید ظاهر شود.



# نیازمندی های واسط خارجی

## واسط های کاربری

- در صورت وجود پرداخت های اینترنتی، باید گواهی اطمینان پرداخت الکترونیک نمایش داده شود.
- وجود بخشی برای نشان دادن نظرات و پیشنهادات پیشین برای ایجاد اطمینان از عملکرد سیستم.
- وجود بخشی برای آموزش نحوه ی کار سیستم به سرپرستان.
- وجود بخشی برای دریافت خلاصه ی فعالیت های پزشکان شامل تعداد نوبت ها و میزان درآمد ماهیانه به صورت اکسل
- توانایی مشاهده ی وضعیت تراکنش های روزانه
- صفحه ی نوبت های رزرو شده تا کنون
- وجود بخشی برای تعیین اطلاعات بانکی یک پزشک برای ایجاد درگاه های پرداخت اینترنتی

## واسط های نرم افزاری

- وجود پایگاه داده برای ذخیره سازی اطلاعات
- واسط برای اتصال به درگاه بانکی
- واسط برای ارسال پیامک
- واسط برای ارسال پیام صوتی ( در صورت نیاز)

# نیازمندی‌های غیرکارکردی

## معیار performance & scalability

- زمان پاسخگویی سیستم در بهترین حالت کمتر از 0.1 ثانیه و در بدترین حالت 5 ثانیه روی مرورگر کروم..
- توانایی مدیریت و پاسخگویی سیستم در حال حاضر 5 هزار نفر است. تا سال آینده این میزان باید به 10 هزار نفر برسد.

## معیار portability & compatibility

- سایت روی نسخه های 6.5 و بالاتر مرورگر کروم سازگار باشد.
- نسخه ی pwa روی گوشی های هوشمند با اندروید های با ورژن 5 و بالاتر سازگار باشد.

## معیار availability

- به دلیل حساس بودن سایت و وابستگی سلامت مردم به در دسترس بودن سیستم، درصد دسترس پذیری در روز های کاری برای بخش مدیریت و سرپرستان بالای 95 درصد در ماه باشد.

## معیار Usability

- نرخ خطای به وجود آمده در رزرو نوبت توسط کاربران باید کمتر از 5 درصد باشد.

## نیازمندی‌های دیگر

کارکردهای سرپرستان این سامانه بسیار تابع کارکردهای مورد نیاز پزشکان و بیماران سامانه است. اصلا وجود سرپرستان، پشتیبانی سامانه و برطرف کردن نیازها یا مشکلاتی است در دیگر کاربران در کار با سامانه دارند. در آینده از سوی پزشکان و بیماران نیازمندی‌هایی زیادی مطرح می‌شوند.

- در حال حاضر پزشکان بازه‌هایی با طول ثابت را در طول روز هفته مشخص میکنند و بیماران میتوانند در این زمان‌ها را به عنوان زمان ملاقات انتخاب کنند. با دیدی واقع بینانه تر، ملاقات یک پزشک با زمان ثابتی ندارد. به همین سبب، تغییراتی جدی در فرآیند و مکانیزم نوبت‌دهی بیماران باید به وجود آید. این مشکل می‌تواند با ایجاد ویژگی اتاق انتظار حل شود. در اتاق انتظار، بیماران منتظر آزاد شدن پزشک برای ملاقات می‌شوند. پزشکان باید بتوانند نفر بعدی را در اتاق انتظار فراخوانی کنند. در این فرایند، به بیمار زمان حدودی ملاقات با پزشک داده میشود و زمان دقیقی مانند حال حاضر انتخاب نمی‌شود.
- امکان ملاقات بین مریض در مطب‌های حضوری بسیار مورد نیاز است. نیاز است که این ویژگی در سامانه تعبیه شود.
- تمرکز ویژگی‌های سامانه فعلی، بر روی ملاقات پزشک و بیمار است. تاریخچه تشخیصات پزشک و تاریخچه نوبت‌های اخذ شده بیمار از امکانات بسیار جالبی است که میتواند به سامانه اضافه شود.
- صدور نسخه الکترونیکی، از نیازمندی‌های اساسی برای تحول دیجیتال است. نیاز است تا سامانه با سامانه متمرکز تامین اجتماعی یکپارچه شود. سامانه متمرکز تامین اجتماعی، وظیفه مدیریت نسخه الکترونیکی در سطح کشور را دارد و لازم است که سامانه ما با این سامانه ارتباط برقرار کند.
- سامانه فعلی بر محوریت ملاقات پزشک با بیمار توسعه داده شده است. گاهی ارتباط‌های کوچکتري با پزشک برای بیمار کافی‌است. به همین منظور باید نیازمندی‌های مربوط به پیام‌های مستقیم پزشک به بیمار در نظر گرفته شوند. البته که این نیازمندی‌ها بسیار فراتر از صرفا ارسال یک پیام است. باید توسط سامانه، توسط یک پروتکل مشخص، ارتباط بیمار با پزشک تدوین شود. یک راه حل ممکن پرداخت پول توسط بیمار به ازای هر پیام یا درخواست از پزشک است. پلتفرم اسنپ دکتر از چنین قابلیت‌های پشتیبانی می‌کند.

## روش‌شناسی توسعه

برای انتخاب روش توسعه‌ی پروژه‌ی smart office، ابتدا باید با ویژگی‌های کلی این پروژه بیشتر آشنا شویم. در زیر به بررسی موردی خصوصیات این پروژه می‌پردازیم:

- مدت زمانی که پروژه باید در آن به انجام برسد، محدود و تا بازه‌ی آخر ترم (حدود ۳ ماه دیگر) است. از این رو در انتخاب ددلاین‌های زمانی باید دقت بیشتری به خرج دهیم و دستانمان در انتخاب زمان ددلاین‌ها بسته‌تر است.
- با توجه به کمبود وقت جهت توسعه‌ی پروژه، زمان کمتری جهت مستندسازی جنبه‌های مختلف پروژه خواهیم داشت.
- انعطاف نیازمندی‌های این پروژه بسیار زیاد است؛ چرا که:
  - نیازمندی‌های این پروژه به صورت دقیق و از پیش مشخص نشده است.
  - در دوران کرونا و با توسعه‌ی فعالیت‌های برخط، روزبه‌روز ایده‌های جدیدتر و درآمدزاتری جهت پیاده‌سازی در فضای وب مطرح می‌شود که پروژه‌ی smart office هم از این ایده‌ها در امان نیست و در هر زمان ممکن است بخشی از پروژه با توجه به ایده‌ای جدید تغییر کند.
  - مخاطبین این پروژه، طیف وسیعی از افراد جامعه را تشکیل می‌دهند و عملاً هر شخصی که به درمان نیاز پیدا کند، مخاطب بالقوه‌ی این پروژه محسوب می‌شود. با توجه به این طیف گسترده از مخاطبین، به مرور زمان و با استفاده از این سامانه، بازخوردهایی را جهت بهبود محصول از مشتریان دریافت می‌کنیم که برای حفظ مزیت رقابتی smart office، مجبور به اعمال آن بازخوردها در محصول هستیم.
- تیم توسعه‌دهندگان این پروژه، ساکن شهرهای مختلف ایران و به صورت دورکار هستند و امکان جمع‌شدن آن‌ها به صورت حضوری در یک مکان وجود ندارد. از این رو بحث ارتباط‌گرفتن مستمر اعضای تیم از هم و درجریان‌کاربودن آن‌ها، سخت‌تر از حالت حضوری است.
- اندازه‌ی تیم توسعه‌ی محصول حدود ۲۰ نفر و نسبتاً کم است؛ در نتیجه از بار فعالیت‌های مدیریت و هدایت تیم در این پروژه کاسته می‌شود.

با توجه به ویژگی‌های ذکرشده در مورد پروژه‌ی smart office و با توجه به مزایا و معایب مدل‌های توسعه‌ی نرم‌افزار در سه دسته‌ی آبشاری، تکرارشونده و چابک، و همچنین با توجه به روی بورس‌بودن استفاده از مدل‌های چابک در پروژه‌های امروزی، تصمیم بر این گرفتیم که از مدل توسعه‌ی چابک برای توسعه‌ی پروژه‌ی smart office استفاده کنیم.

از بین روش‌های توسعه‌ی چابک نیز، با توجه به این که عمده‌ی تمرکز ما در این پروژه صرف رسیدن محصول است و اکثر اعضای گروه از درس تحلیل-طراحی سیستم‌ها که پیش از این گذرانده‌اند، با روش scrum آشنایی دارند، تصمیم به استفاده از این روش در توسعه‌ی smart office گرفتیم.

خطری که با انتخاب این روش ما را تهدید می‌کند، دوربودن فیزیکی اعضای تیم توسعه از هم است؛ چرا که یکی از پایه‌های اصلی scrum، ارتباطات مستمر اعضای تیم توسعه با یکدیگر است. در حالت دورکاری، ارتباط‌گیری و در جریان کار دیگر اعضا بودن، سخت‌تر از حالت حضوری می‌شود، اما به لطف توسعه‌ی وسایل ارتباط جمعی و وجود انواع مختلف اتاق‌های گفت‌وگوی باکیفیت مانند google meet یا jitsi و هم‌چنین، ابزارهای کار اشتراکی تحت وب مانند trello و mirro، این اشکال تا حد خوبی برطرف می‌شود.

## ضمیمه

### واژه‌نامه

admin = سرپرست

wallet = کیف پول

functionality = کارکرد

method = روش

methodology = روش‌شناسی

flexibility = انعطاف

online = برخط

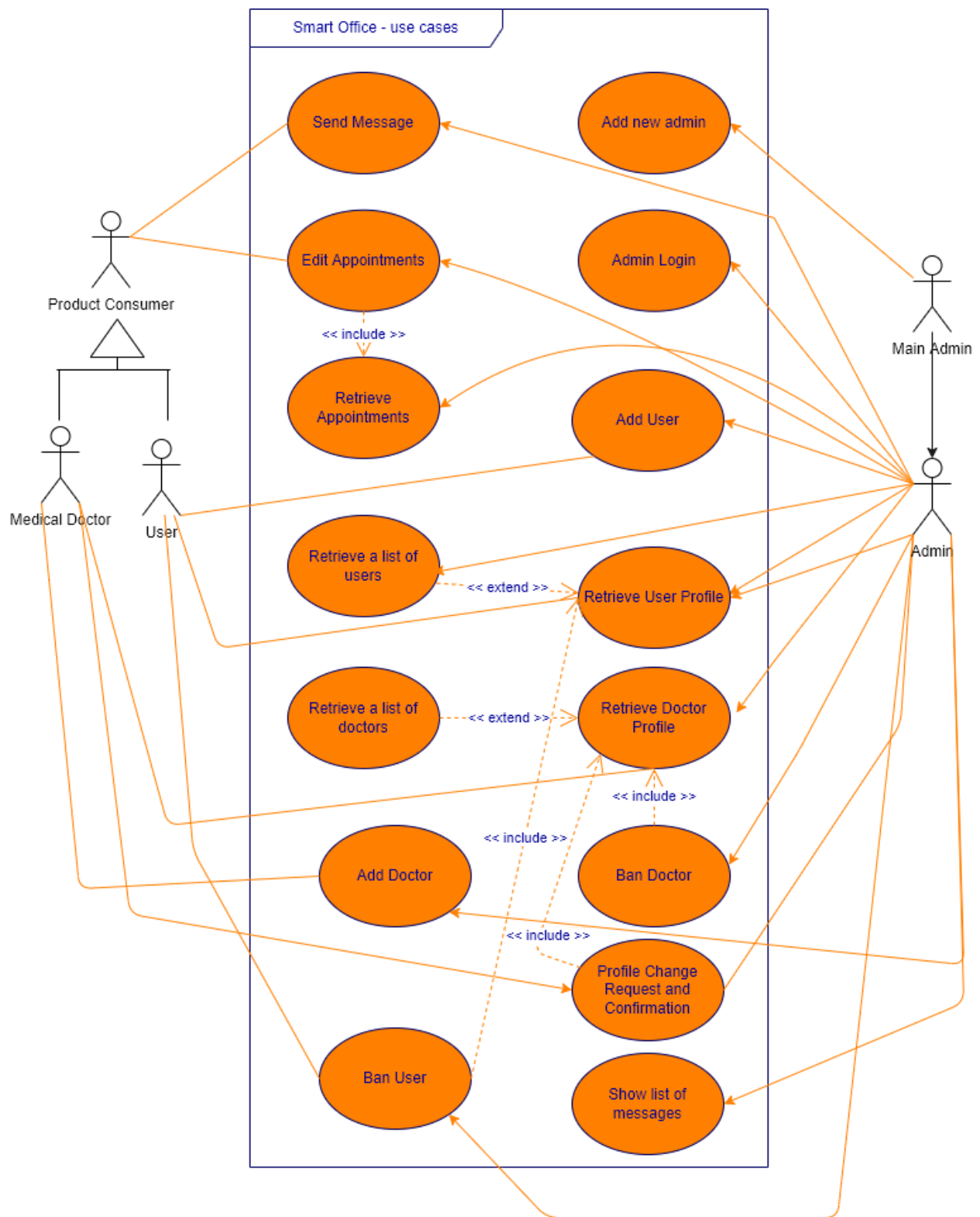
remote = دورکاری

waterfall = آبشاری

iterative = تکرارشونده

agile = چابک

## مدل نیازمندی‌ها



Admin Login
ID: 1
Brief Description: Admins are able to login using their username and password after registering on the website.
Primary Actors: Admin
Secondary Actors: None
Preconditions: <ul style="list-style-type: none"><li>- The user should have been registered in the system before or its credentials are hardcoded in the system before</li></ul>
Main Flow: <ul style="list-style-type: none"><li>- The admin enters their username and password in the website</li><li>- The username and password is authenticated in the system database</li><li>- If the authentication fails, the admin is prompted to the same page again.</li><li>- Otherwise, the user is now redirected to the main page</li></ul>
Postconditions: <ul style="list-style-type: none"><li>- The admin is now online and can control the system</li></ul>
Alternative flows: None



Retrieve a list of doctors
ID: 2
<p>Brief Description:</p> <p>Admins should be able to retrieve a list of doctors after applying a set of filters such as Profession, Location, Overall Score, etc. This can be done for statistical analysis of the system by the admins.</p>
<p>Primary Actor:</p> <p>Admin</p>
<p>Secondary Actor:</p> <p>None</p>
<p>Precondition:</p> <p>Admin should be logged into the system</p>
<p>Main flow:</p> <ul style="list-style-type: none"><li>- Enter the search bar</li><li>- Apply a bunch of filters such as overall score, location, profession, etc.</li><li>- Send search request to database</li><li>- Retrieve a list containing doctors that comply with the search condition</li></ul>
<p>Postconditions:</p> <p>None</p>
<p>Alternative Flows:</p> <p>None</p>

Retrieve Information Regarding a Specific Doctor
ID: 3
Brief Description: Retrieve the profile of a doctor alongside their record and formal documents
Primary Actor: Admin
Secondary Actor: Medical Doctor
Precondition: Admin should be logged into the system and the doctor should be available in the database
Main Flow: <ul style="list-style-type: none"><li>- Enter the doctor id or enter a doctor's profile by clicking on the name from the search bar</li><li>- See information such as name, age, degree, profession, date of registration in the website, education, formal documents regarding their medical certificate, total score</li></ul>
Postcondition: None
Alternative Flows: None

Confirm Doctor Profile Change or Creation
ID: 4
<p><b>Brief Description:</b></p> <p>Admins should be able to have access to a list of pending requests from doctors, review their documents or edits, and then send a confirmation message to the doctors.</p>
<p><b>Primary Actor:</b></p> <p>Medical Doctor</p>
<p><b>Secondary Actor:</b></p> <p>Admin</p>
<p><b>Precondition:</b></p> <p>The doctor should enter their number and their credentials in their request to be added to the system. The admin should be logged in, and after review, they should be able to establish an end to end communication while the process is being done.</p>
<p><b>Main Flow:</b></p> <ul style="list-style-type: none"> <li>- The doctor enters a profile or edits his/her profile and sends it for review to the admins.</li> <li>- The admin opens a page containing all the requests from multiple doctors.</li> <li>- The admin opens a request and checks all the credentials for a doctor</li> <li>- The admin may change some of the information.</li> <li>- After checking, in case of confirmation, the profile will change and a notification will be sent out to the doctor..</li> <li>- In case the new information is not confirmed, a side note next to the request is appended and sent back to the doctor.</li> <li>- Finally, when the admin and doctor reach a negotiation, an id is assigned to the doctor and they are added to the database with default settings.</li> </ul>
<p><b>Postcondition:</b></p> <p>The doctor is added to the database and their profile is updated.</p>
<p><b>Alternate flows:</b></p> <ul style="list-style-type: none"> <li>- If a message stays pending on the doctor's side for more than 4 days, it is removed from the message list.</li> <li>- A doctor cannot send multiple requests to register in the system. In the case of no response on admin's behalf, the doctor can renew the request using a button.</li> </ul>

Registering a new doctor
ID: 5
<p>Brief description:</p> <p>An admin can add a doctor by having a signed request from them alongside their medical credentials and phone number.</p>
<p>Main Actor:</p> <p>Admin</p>
<p>Secondary Actor:</p> <p>Medical doctor</p>
<p>Preconditions:</p> <p>The doctor should send a signed request for the admins alongside their medical credentials and phone number.</p>
<p>Main Flow:</p> <ul style="list-style-type: none"><li>- The admin creates a profile using the information provided</li><li>- The doctor is assigned a unique id in the database and their profile is added</li><li>- A confirmation message is sent to the doctor alongside a link to their profile</li><li>- After the doctor confirms the profile, the profile will now be public.</li></ul>
<p>Postcondition:</p> <p>The doctor is added to the database</p>
<p>Alternate flows:</p> <ul style="list-style-type: none"><li>- If the information is incomplete, the admin will send a message to the doctor stating that.</li><li>- If the profile is not confirmed after 30 days, everything will be removed from the database.</li></ul>

Ban Doctor
ID: 6
Brief Description: The admins should be able to block doctors due to violation of a certain law of conduct.
Main Actor: Admin
Secondary Actor: Medical Doctor
Preconditions: Doctor profile is added and is violating a law of conduct.
Main flow: <ul style="list-style-type: none"><li>- The admin reveals a doctor's profile</li><li>- They select the "Block" or Ban option alongside a valid description.</li><li>- A notification is sent to the doctor regarding the blocking</li><li>- Any ongoing appointment will proceed as before; however, no new appointments can proceed.</li></ul>
Postcondition: The doctor is now banned!
Alternate flows: None

Add user
ID: 7
<p>Brief Description:</p> <p>Admins can add users manually to the system. This functionality is good for users that refer to the doctor without using the smart functionalities.</p>
<p>Main Actor:</p> <p>Admin</p>
<p>Secondary Actor:</p> <p>User</p>
<p>Precondition:</p> <p>The credentials for registration should be complete.</p>
<p>Main flow:</p> <ul style="list-style-type: none"> <li>- The admin has access to a list of user registration requests. These requests might have been initiated by the doctors for patients that referred to them without having a profile and the doctor wants to create their profile.</li> <li>- A profile is created for the user with a default password and a text message is sent to their number containing a link for them to change their password.</li> <li>- The user credentials are then added to the database.</li> </ul>
<p>Postconditions:</p> <p>User registration complete.</p>
<p>Alternate flows:</p> <p>None</p>

Retrieving a list of users with certain search filters
ID: 8
Brief description: The admins can search through the database for users according to their names or ids.
Main actor: Admin
Secondary Actors: None
Precondition: None
Main flow: <ul style="list-style-type: none"><li>- The admin can enter a panel and enter a text showing user id, name, or last name and a list of retrieved users in the database will be shown.</li><li>- By clicking on each item the user profile will be shown.</li></ul>
Postconditions: None
Alternate flows: None

Check user information and profile
ID: 9
<p>Brief description:</p> <p>The admins can click on users from their search bar and checkout user credentials, history of appointments and their account balance.</p>
<p>Main Actor:</p> <p>Admin</p>
<p>Secondary Actor:</p> <p>User</p>
<p>Precondition:</p> <p>The user is available and valid in the database</p>
<p>Main flow:</p> <ul style="list-style-type: none"><li>- The admin can click on a link that redirects to the user profile.</li><li>- The profile contains basic information such as username, first and second name, age, medical history, and also a history of recent appointments as well as their account balance</li></ul>
<p>Postconditions:</p> <p>None</p>
<p>Alternate flows:</p> <p>None</p>



Ban user
ID: 10
Brief description: The admins can ban users if they have proof that they somehow conflicted with the product code of conduct.
Main Actor: Admin
Secondary Actor: User
Precondition: The user has misused the service or has engaged in straight conflict with the codes of conduct.
Main flow: <ul style="list-style-type: none"><li>- In the user profile, admins have a button to ban user</li><li>- After pressing the ban button, they should enter a description</li><li>- The description is sent as a notification to the user that is now banned and the user is added to a blacklist and cannot access services any more.</li></ul>
Postcondition: The user cannot access any of the services.
Alternate flow: None

Retrieving the appointments of a doctor
ID: 11
<p>Brief description:</p> <p>Using this option, the admin can enter the schedule of appointments that a doctor has in a certain timeline. By hovering over each of the appointments, the admin is able to see the Patient id.</p>
<p>Main Actor:</p> <p>Admin</p>
<p>Secondary Actor:</p> <p>None</p>
<p>Precondition:</p> <p>None</p>
<p>Main flow:</p> <ul style="list-style-type: none"> <li>- By choosing an option in the profile of the doctor, the admin is prompted to a page containing all the active appointments in the time table.</li> <li>- The admin can scroll over appointments up to 4 months into the future and by hovering over each appointment, the patient id and name should appear.</li> </ul>
<p>Postcondition:</p> <p>None</p>
<p>Alternate flows:</p> <p>None</p>

Reserve appointment or delete one
ID: 12
<p>Brief Description:</p> <p>Admins should be able to manually remove an appointment or create a new appointment for a certain user in the page described in the last use case.</p>
<p>Main Actor:</p> <p>Admin</p>
<p>Secondary Actors:</p> <p>Corresponding medical doctor and user</p>
<p>Precondition:</p> <p>There is a valid reason for manual addition or deletion of the appointment and there is mutual consent. Additionally, there should be at least 3 hours left before the appointment.</p>
<p>Main flow:</p> <ul style="list-style-type: none"> <li>- The admin enters the appointment page of a doctor</li> <li>- The admin selects a range for the appointment and enters the patient id.</li> <li>- A notification is sent for both the user and the doctor.</li> <li>- After accepting the appointment from both sides, the appointment will be added to the page, before that, the appointment is pending.</li> </ul>
<p>Postcondition:</p> <p>An appointment is added to the database.</p>
<p>Alternate flows:</p> <ul style="list-style-type: none"> <li>- There should not exist any overlapping appointments and a user cannot have multiple appointments intersecting each other.</li> <li>- If the appointment is not confirmed until a limit amount, then the appointment is erased and will not stay in pending; thus, others can reserve that certain time.</li> </ul>

Add new admin
ID: 13
<p>Brief description:</p> <p>The main admin can add a new admin by entering a certain page and entering a new username and password.</p>
<p>Main actor:</p> <p>Main admin</p>
<p>Secondary actor:</p> <p>Admin</p>
<p>Precondition:</p> <p>The logged in admin is the main admin and the entered username and password does not match any of the information from before.</p>
<p>Main flow:</p> <ul style="list-style-type: none"> <li>- The main admin enters a page.</li> <li>- Enters a username and password</li> <li>- If the username and password does not match any of the usernames and passwords from before then the account is created and added to the database.</li> <li>- Otherwise, the main admin is asked to re-enter username and password.</li> </ul>
<p>Postcondition:</p> <p>A new admin is added.</p>
<p>Alternate flows:</p> <p>None</p>

Show received messages
ID: 14
<p>Brief Description:</p> <p>Admins receive multiple messages including profile change messages, Q &amp; A messages, etc. These messages should be listed in a bar for them to scroll through.</p>
<p>Main Actor:</p> <p>Admin</p>
<p>Secondary Actor:</p> <p>None</p>
<p>Precondition:</p> <p>None</p>
<p>Main flow:</p> <ul style="list-style-type: none"><li>- By clicking on a notification pane, the admin should be able to look through all the messages that he/she has received.</li><li>- By clicking on each message, the whole text of the message is shown and an answer text field is also shown to enter an answer.</li></ul>
<p>Postcondition:</p> <p>None</p>
<p>Alternate flows:</p> <p>None</p>

Send message
ID: 15
<p>Brief Description:</p> <p>The admins should be able to communicate with the users or doctors either by answering their questions in the support panel or by sending them a notification.</p>
<p>Primary actor:</p> <p>Admin</p>
<p>Secondary actors:</p> <p>Doctor or User</p>
<p>Precondition:</p> <p>The admin should have selected a user or a doctor to send them a message.</p>
<p>Main flow:</p> <ul style="list-style-type: none"><li>- After selecting a doctor or a user in the support page, the admin can enter a new message in the text field to them in correspondence to their thread of messages.</li></ul>
<p>Postcondition:</p> <p>Send message</p>
<p>Alternate flows:</p> <ul style="list-style-type: none"><li>- The admin can also select a person via their id and directly send them a message even though no kind of communication has been established with them before.</li></ul>

## اولویت‌بندی نیازمندی‌ها

یک روش اولویت‌بندی نیازمندی‌ها که بسیار ساده و سراسر است، روش MoSCoW است. در این روش، نیازمندی‌ها به چهار دسته تقسیم می‌شوند:

- دسته اول، دسته **Must Have** است. نیازمندی‌های این دسته، حتما و الزاما باید پیاده‌سازی شوند و از اساسی‌ترین نیازها می‌باشند. عدم پیاده‌سازی آن‌ها به شکست پروژه می‌انجامد.
- دسته دوم، دسته **Should Have** است. نیازمندی‌های این دسته نیز بسیار مهم هستند. اما انجام نشدن آن‌ها مستقیما باعث شکست نمیشوند بلکه مسیر آن را تغییر میدهد. به بیان دیگر از نظر اهمیت در درجه دوم قرار دارند.
- دسته سوم، دسته **Could Have** است. نیازمندی‌های این دسته برای اساسی نیستند اما بهتر است آن‌ها را پیاده‌سازی کنیم.
- دسته چهارم، دسته **Won't Have** است. نیازمندی‌های این دسته پیاده‌سازی نخواهند شد. یا به دلیل محدودیت زمان یا محدودیت بودجه یا محدودیت‌های دیگر. البته این امر ممکن است با پیشرفت پروژه تغییر کند.

ما از این روش استفاده خواهیم کرد.

اولویت	نام نیازمندی
Must Have	ورود به سامانه و خروج از آن
Must Have	نمایش لیست پزشکان
Must Have	مشاهده مشخصات و مدارک یک پزشک
Must Have	تایید پزشک یا ویرایش اطلاعات پزشک
Must Have	افزودن پزشک
Could Have	بستن حساب پزشکان
Could Have	افزودن کاربر

Must Have	مشاهده لیست کاربران
Must Have	مشاهده مشخصات کاربر
Could Have	بستن حساب کاربر
Must Have	مشاهده وضعیت نوبت‌های یک پزشک
Should Have	رزرو یا حذف نوبت از طرف بیمار
Won't Have	اضافه کردن سرپرست جدید
Should Have	مشاهده لیست پیام‌های دریافت شده
Won't Have	ارسال پاسخ پیام پشتیبانی یا ارسال نوتیفیکیشن به کاربر یا پزشک