دانشگاه بوعلی سینا

دانشکده مهندسی — گروه مهندسی کامپیوتر

گزارش پروژه پایانی دوره کارشناسی

طراحی و پیاده‌سازی ماژول بیمه تکمیلی

سامانه تحت وب با موتور قانون داده‌محور، گردش کار چندمرحله‌ای و ردیابی کامل

نگارنده: **سعید مظاهری**

شماره دانشجویی: **۴۰۲۴۲۹۹۹۰۰۴**

رشته و گرایش: **مهندسی کامپیوتر — نرم‌افزار**

استاد راهنما: **جناب آقای دکتر مهدی سخایی‌نیا**

مدت پروژه: **نیم‌سال ۴۰۴۲ و ۴۰۴۳**

چکیده

در بسیاری از سازمان‌ها، فرایند بیمه تکمیلی کارکنان به‌صورت دستی و با تکیه بر صفحه‌های گسترده اداره می‌شود. این روش سه اشکال جدی دارد: نخست، محاسبه دستی سهم قابل پرداخت — که همزمان به درصد پوشش، سقف هر دفعه و مانده سقف سالانه وابسته است — مستعد خطاست؛ دوم، تصمیم‌ها قابل ردیابی نیستند و در صورت اعتراض کارمند، مبنای تصمیم قابل بازسازی نیست؛ سوم، هر تغییر در سیاست‌های رفاهی سازمان مستلزم تغییر در منطق محاسبه و در نتیجه نیازمند چرخه توسعه نرم‌افزار است.

در این پروژه یک سامانه تحت وب برای مدیریت بیمه تکمیلی طراحی و پیاده‌سازی کرده‌ام که این سه مسئله را هدف گرفته است. دستاورد اصلی، «موتور قانون پوشش و محاسبه سقف تعهدات» است که به‌جای کدنویسی ثابت، بر پایه پیکربندی داده‌محور کار می‌کند: درصد پوشش، سقف هر دفعه، سقف سالانه، دوره انتظار و نسبت‌های مجاز، همه به‌صورت رکوردهای نسخه‌دار در پایگاه داده نگه‌داری می‌شوند. در نتیجه تغییر سیاست رفاهی صرفاً از طریق پیکربندی و بدون تغییر کد و استقرار مجدد اعمال می‌شود.

در کنار آن، گردش کار چندمرحله‌ای درخواست با مسیرهای تأیید، رد و بازگشت برای تکمیل مدارک پیاده شده است و تمام رویدادهای مؤثر — همراه با کاربر، زمان و مقدار قبل و بعد — در یک لاگ ممیزی ثبت می‌شوند؛ به‌گونه‌ای که نوشتن لاگ و تغییر داده در یک تراکنش انجام می‌شود و از نظر ساختاری امکان تغییر بدون ردپا وجود ندارد. کنترل دسترسی مبتنی بر نقش در چهار سطح، گزارش‌گیری مدیریتی، و رابط کاربری کامل فارسی و راست‌به‌چپ نیز پیاده‌سازی شده‌اند.

سامانه با زبان Go در سمت سرور، پایگاه داده PostgreSQL و کتابخانه React با TypeScript در سمت کاربر توسعه یافته و درستی آن با آزمون‌های واحد، آزمون‌های یکپارچگی روی پایگاه داده واقعی و یک مجموعه آزمون سرتاسری روی مرورگر راستی‌آزمایی شده است.

کلمات کلیدی: بیمه تکمیلی، سامانه تحت وب، موتور قانون، گردش کار چندمرحله‌ای، ردیابی و ممیزی، کنترل دسترسی مبتنی بر نقش، معماری لایه‌ای، Go، PostgreSQL

فهرست مطالب

این فهرست به‌صورت خودکار ساخته می‌شود. اگر خالی نمایش داده شد، روی آن کلیک کنید و کلید F9 را بزنید (یا در Word: References → Update Table).

برای نمایش فهرست، روی آن کلیک کنید و کلید F9 را بزنید.

۱. مقدمه و بیان مسئله

بیمه تکمیلی درمان یکی از رایج‌ترین خدمات رفاهی سازمان‌ها است. در الگوی متداول، کارمند فاکتور هزینه درمانی خود را به واحد رفاه تحویل می‌دهد، کارشناس آن را بررسی و سهم قابل پرداخت را محاسبه می‌کند و نتیجه برای پرداخت به واحد مالی اعلام می‌شود. هرچند این فرایند در ظاهر ساده است، در عمل چند دشواری مشخص دارد که انگیزه اصلی این پروژه بوده‌اند.

۱.۱. دشواری محاسبه

مبلغ قابل پرداخت تابعی از چند پارامتر همزمان است: درصد پوشش نوع خدمت، سقف مجاز برای هر دفعه، و مهم‌تر از همه مانده سقف سالانه که به سابقه مصرف همان فرد در همان سال قراردادی بستگی دارد. محاسبه دستی این ترکیب، به‌ویژه هنگام نزدیک شدن به سقف سالانه، مستعد خطاست و خطای آن مستقیماً بر حقوق مالی کارمند اثر می‌گذارد.

۱.۲. نبود شفافیت و ردیابی

در روش دستی، کارمند از وضعیت درخواست خود و از مانده سهمیه‌اش بی‌اطلاع است و برای هر پرسش باید به واحد رفاه مراجعه کند. مهم‌تر آن‌که اگر ماه‌ها بعد اعتراضی مطرح شود، مستندی وجود ندارد که نشان دهد آن تصمیم را چه کسی، در چه زمانی و بر مبنای کدام قاعده گرفته است.

۱.۳. هزینه تغییر سیاست

سیاست‌های رفاهی سازمان‌ها ثابت نیستند؛ درصد پوشش‌ها، سقف‌ها و شرایط واجدبودن هر سال بازنگری می‌شوند. اگر این قواعد در کد نرم‌افزار نوشته شده باشند، هر تغییر کوچک به یک چرخه کامل توسعه، آزمون و استقرار تبدیل می‌شود. این وابستگی، سازمان را برای اجرای تصمیم‌های خودش به تیم فنی وابسته می‌کند.

۱.۴. رویکرد این پروژه

بر پایه این سه مسئله، سامانه‌ای طراحی کرده‌ام که محاسبه را خودکار و یکسان می‌کند، هر رویداد را قابل ردیابی نگه می‌دارد، و — به‌عنوان تصمیم محوری طراحی — قواعد بیمه را از کد جدا کرده و به داده تبدیل می‌کند. در ادامه این گزارش، ابتدا اهداف و محدوده پروژه را مشخص می‌کنم، سپس تحلیل و طراحی، معماری و پیاده‌سازی را شرح می‌دهم، و در پایان نتایج را در برابر معیارهای پذیرشی که در پیشنهاد پروژه تعریف شده بود ارزیابی می‌کنم.

۲. اهداف پروژه

اهداف پروژه را در پنج بند صورت‌بندی کرده‌ام:

- **محاسبه خودکار و دقیق تعهدات:** تعیین مبلغ قابل پرداخت و مانده سقف برای هر فرد، بدون دخالت محاسبه دستی.
- **پیکربندی‌پذیری قواعد:** امکان تغییر درصد پوشش، سقف‌ها، دوره انتظار و شرایط واجدبودن صرفاً از طریق پیکربندی و بدون تغییر کد.

- **گردش کار شفاف:** پیاده‌سازی فرایند چندمرحله‌ای درخواست با مسیرهای تأیید، رد با ذکر دلیل، و بازگشت برای تکمیل مدارک.
- **ردیابی و پاسخ‌گویی:** ثبت تمام رویدادهای مؤثر همراه با کاربر، زمان و مقدار قبل و بعد، و امکان بازسازی تاریخچه هر درخواست.
- **گزارش‌گیری مدیریتی:** ارائه مصرف و مانده هر فرد، هزینه کل سازمان، و تفکیک بر اساس نوع خدمت و بازه زمانی.

علاوه بر این، از آن‌جا که این ماژول بخشی از «سامانه جامع خدمات رفاهی کارکنان» در نظر گرفته شده است، آن را با مرز مشخص و رابط برنامه‌نویسی تعریف‌شده طراحی کرده‌ام تا در آینده در کنار ماژول‌های دیگر (وام، مهمان‌سرا، خانه سازمانی) قابل استقرار باشد.

۳. محدوده پروژه

۳.۱. داخل محدوده

- تعریف قرارداد بیمه، طرح‌های پوشش و انواع خدمت
- تعریف قواعد پوشش به‌صورت نسخه‌دار و قابل پیکربندی
- ثبت کارکنان و اعضای تحت تکفل و اتصال هر کارمند به یک طرح
- ثبت درخواست هزینه برای شخص بیمه‌شده یا اعضای تحت تکفل
- محاسبه خودکار مبلغ قابل پرداخت و مانده سقف
- گردش کار چندمرحله‌ای با شاخه‌های تأیید، رد و بازگشت
- لاگ ممیزی کامل با امکان جست‌وجو و بازسازی تاریخچه
- کنترل دسترسی مبتنی بر نقش در چهار سطح
- گزارش‌گیری مدیریتی و رابط کاربری کامل فارسی
- رابط برنامه‌نویسی برای اتصال به سامانه مادر

۳.۲. خارج از محدوده

موارد زیر آگاهانه و از ابتدا خارج از محدوده تعریف شده‌اند و در این گزارش نیز به‌عنوان محدودیت شناخته‌شده معرفی می‌شوند، نه نقص:

مورد	توضیح
درگاه پرداخت واقعی	پرداخت به‌صورت شبیه‌سازی ثبت می‌شود و شماره پیگیری تولید می‌گردد؛ اتصال به سامانه بانکی انجام نشده است.
اتصال زنده به سامانه‌های سازمان	به‌جای اتصال زنده به منابع انسانی و مالی، یک درگاه برنامه‌نویسی با احراز هویت کلید ساخته شده است.
سایر ماژول‌های رفاهی	وام، مهمان‌سرا و خانه سازمانی خارج از محدوده این ماژول‌اند.
آزمایشگاه	خارج از محدوده.

۴. تحلیل و طراحی

۴.۱. نقش‌ها و سطوح دسترسی

چهار نقش تعریف شده است و هر نقش تنها به بخشی از سامانه دسترسی دارد که برای انجام وظیفه‌اش لازم است. جدول زیر مهم‌ترین قابلیت‌ها را به تفکیک نقش نشان می‌دهد.

قابلیت	کارمند	کارشناس	مدیر	ممیز
مشاهده درخواست‌های خود	دارد	دارد	دارد	دارد
مشاهده درخواست‌های دیگران	ندارد	دارد	دارد	دارد
ثبت درخواست	برای خود	ندارد	برای همه	ندارد
تأیید / رد / بازگرداندن	ندارد	دارد	دارد	ندارد
ثبت پرداخت و بستن پرونده	ندارد	دارد	دارد	ندارد
مشاهده مانده سقف	خود	همه	همه	ندارد
مدیریت کارکنان و تحت تکفل	ندارد	فقط مشاهده	دارد	ندارد
انتشار نسخه جدید قانون	ندارد	ندارد	**دارد**	ندارد
مدیریت کاربران	ندارد	ندارد	دارد	ندارد
گزارش‌ها و لاگ ممیزی	ندارد	ندارد	دارد	دارد

کنترل دسترسی در دو لایه اعمال می‌شود: در لایه مسیرهای وب، نقش کاربر بررسی می‌شود؛ و در لایه سرویس، مالکیت داده کنترل می‌گردد. به این ترتیب یک کارمند حتی با دانستن شناسه درخواست دیگری، به آن دسترسی پیدا نمی‌کند.

۴.۲. موجودیت‌های اصلی داده

پایگاه داده از دوازده جدول تشکیل شده است. موجودیت‌های اصلی و نقش هرکدام:

موجودیت	نقش در سامانه
قرارداد بیمه	بالاترین سطح؛ دارای تاریخ شروع و پایان
طرح پوشش	سطح خدمات زیر یک قرارداد؛ هر کارمند به یک طرح متصل است
نوع خدمت	دسته هزینه: ویزیت، دارو، دندان‌پزشکی، بستری، عینک
قانون پوشش	**پارامترهای محاسبه برای هر ترکیب طرح و نوع خدمت؛ نسخه‌دار**
کارمند	اطلاعات پرسنلی، تاریخ استخدام و وضعیت اشتغال
عضو تحت تکفل	همسر، فرزند یا والدین وابسته به کارمند
کاربر	حساب دسترسی با نقش؛ در صورت نقش کارمند به پرونده پرسنلی متصل
درخواست	هزینه ثبت‌شده با وضعیت و مبالغ محاسبه‌شده
پرداخت	رکورد پرداخت شبیه‌سازی‌شده با شماره پیگیری
لاگ ممیزی	رویدادها با کاربر، زمان و مقدار قبل و بعد
کلید یکپارچه‌سازی	کلید دسترسی سامانه مادر (به‌صورت هش‌شده)
مدارک پیوست	جدول پیش‌بینی‌شده برای پیوست فاکتور

۴.۳. گردش کار درخواست

گردش کار به‌صورت یک ماشین حالت صریح پیاده شده است. هر گذار مجاز در یک جدول تعریف شده و هر گذاری که در آن جدول نباشد رد می‌شود؛ بنابراین پریدن از مراحل امکان‌پذیر نیست.

مسیر اصلی: **پیش‌نویس ← ثبت‌شده ← در حال بررسی ← تأییدشده ← پرداخت‌شده ← بسته‌شده**

مسیر رد: **در حال بررسی ← رد شده ← بسته‌شده**

مسیر تکمیل مدارک: **در حال بررسی ← بازگشت برای تکمیل مدارک ← ثبت‌شده (ارسال مجدد)**

وضعیت مبدأ	اقدام	وضعیت مقصد	مجاز برای
پیش‌نویس	ثبت	ثبت‌شده	مالک یا مدیر
ثبت‌شده	شروع بررسی	در حال بررسی	کارشناس یا مدیر
در حال بررسی	تأیید	تأییدشده (با محاسبه مبلغ)	کارشناس یا مدیر
در حال بررسی	رد (با ذکر دلیل)	رد شده	کارشناس یا مدیر
در حال بررسی	بازگرداندن (با ذکر دلیل)	بازگشت برای تکمیل مدارک	کارشناس یا مدیر
بازگشت برای مدارک	ارسال مجدد	ثبت‌شده	مالک یا مدیر
تأییدشده	ثبت پرداخت	پرداخت‌شده	کارشناس یا مدیر
پرداخت‌شده یا رد شده	بستن	بسته‌شده	کارشناس یا مدیر

۴.۴. موتور قانون و محاسبه سقف

این بخش، دستاورد مهندسی اصلی پروژه است. به‌جای آن‌که قواعدی مانند «پوشش دندان‌پزشکی پنجاه درصد است» در کد نوشته شود، هر قانون یک رکورد در پایگاه داده است با این پارامترها: درصد پوشش، سقف هر دفعه، سقف سالانه، دوره انتظار، و فهرست نسبت‌های مجاز. جدول زیر قواعد پیش‌فرض طرح استاندارد را نشان می‌دهد.

نوع خدمت	درصد پوشش	سقف هر دفعه (ریال)	سقف سالانه (ریال)	دوره انتظار	نسبت‌های مجاز
ویزیت	۷۰٪	۵۰۰٬۰۰۰	۵٬۰۰۰٬۰۰۰	۰ روز	خود، همسر، فرزند، والدین
دارو	۸۰٪	۱٬۰۰۰٬۰۰۰	۱۰٬۰۰۰٬۰۰۰	۰ روز	خود، همسر، فرزند، والدین
دندان‌پزشکی	۵۰٪	۳٬۰۰۰٬۰۰۰	۱۵٬۰۰۰٬۰۰۰	۹۰ روز	خود، همسر، فرزند
بستری	۹۰٪	۵۰٬۰۰۰٬۰۰۰	۱۰۰٬۰۰۰٬۰۰۰	۳۰ روز	خود، همسر، فرزند، والدین
عینک	۶۰٪	۲٬۰۰۰٬۰۰۰	۴٬۰۰۰٬۰۰۰	۱۸۰ روز	خود، همسر، فرزند

هنگام تأیید یک درخواست، موتور قانون ابتدا پنج پیش‌شرط را بررسی می‌کند:

۱. کارمند در وضعیت شاغل باشد.

۲. اگر ذی‌نفع عضو تحت تکفل است، واقعاً به همان کارمند تعلق داشته باشد.

۳. برای ترکیب طرح و نوع خدمت، در تاریخ فاکتور قانون فعالی وجود داشته باشد.

۴. نسبت ذی‌نفع در فهرست نسبت‌های مجاز آن قانون باشد.

۵. دوره انتظار از تاریخ استخدام سپری شده باشد.

در صورت گذر از این پیش‌شرط‌ها، محاسبه در سه گام انجام می‌شود:

۱. مبلغ پوشش‌داده‌شده برابر است با مبلغ درخواستی ضرب در درصد پوشش، گرد شده به ریال کامل.

۲. اگر این مبلغ از سقف هر دفعه بیشتر باشد، به سقف هر دفعه محدود می‌شود.

۳. اگر همچنان از مانده سقف سالانه بیشتر باشد، به مانده سالانه محدود می‌شود.

برای روشن‌شدن اثر سقف‌ها، دو نمونه محاسبه برای خدمت ویزیت در طرح استاندارد (هفتاد درصد پوشش، سقف هر دفعه پانصد هزار ریال):

مبلغ درخواستی	درصد پوشش	پیش از اعمال سقف	مبلغ قابل پرداخت	قید مؤثر
۳۵۰٬۰۰۰	۷۰٪	۲۴۵٬۰۰۰	**۲۴۵٬۰۰۰**	زیر همه سقف‌ها
۱٬۰۰۰٬۰۰۰	۷۰٪	۷۰۰٬۰۰۰	**۵۰۰٬۰۰۰**	سقف هر دفعه

۴.۵. نسخه‌دار بودن قواعد

قواعد هرگز در جای خود ویرایش نمی‌شوند. هنگام تغییر سیاست، نسخه فعلی با ثبت «تاریخ اعمال تا» بسته می‌شود و نسخه جدید درج می‌گردد. انتخاب قانون برای هر درخواست نیز بر اساس تاریخ فاکتور آن درخواست انجام می‌شود، نه تاریخ امروز.

> اهمیت این تصمیم در ممیزی روشن می‌شود: اگر قواعد تاریخ‌دار نبودند، هر تغییر سیاست، محاسبه درخواست‌های گذشته را نیز تغییر می‌داد و بازسازی مبنای تصمیم‌های پیشین ناممکن می‌شد. همچنین برای حالت خاصِ انتشار دو نسخه در یک روز، تاریخ بستن نسخه پیشین به تاریخ شروع خودش محدود می‌شود و در انتخاب قانون فعال، نسخه جدیدتر اولویت دارد؛ برای این حالت آزمون اختصاصی نوشته‌ام.

۵. معماری نرم‌افزار

۵.۱. فناوری‌های استفاده‌شده

بخش	فناوری	دلیل انتخاب
سمت سرور	زبان Go با روتر chi	کارایی بالا، تایپ ایستا، استقرار به‌صورت فایل اجرایی واحد
پایگاه داده	PostgreSQL	پشتیبانی قوی از تراکنش و قیدهای یکپارچگی داده
سمت کاربر	React با TypeScript	ساخت رابط تعاملی با ایمنی تایپ
ظاهر رابط	Tailwind CSS و قلم وزیرمتن	پشتیبانی مناسب از چیدمان راست‌به‌چپ فارسی
استقرار	Docker Compose	اجرای یکسان سه سرویس در محیط‌های مختلف

۵.۲. معماری لایه‌ای

سامانه در چهار لایه سازمان یافته است و وابستگی‌ها تنها به سمت درون جریان دارند. این تفکیک باعث می‌شود هر نگرانی یک جایگاه مشخص داشته باشد:

```
transport/http   ->   service/*   ->   storage/postgres
                          domain
```

لایه	مسئولیت
دامنه	موجودیت‌ها و مفاهیم کسب‌وکار به‌صورت خالص؛ بدون وابستگی به وب یا پایگاه داده
سرویس	منطق کسب‌وکار، تراکنش‌ها و تصمیم‌های دسترسی؛ هفت بسته به تفکیک حوزه
ذخیره‌سازی	پیاده‌سازی دسترسی به داده؛ تنها لایه‌ای که با کتابخانه نگاشت داده کار می‌کند
انتقال	مسیرهای وب، اعتبارسنجی ورودی، و نگاشت خطاها به کدهای وضعیت

دو نکته طراحی که به‌طور خاص رعایت شده‌اند: نخست، شِمای پایگاه داده تنها در اختیار فایل‌های مهاجرت است و از قابلیت ساخت خودکار جدول در کتابخانه نگاشت داده استفاده نکرده‌ام تا تغییرات شِما قابل بازبینی و قابل بازگشت باشند. دوم، ساختار داده‌ای که در پایگاه داده ذخیره می‌شود از ساختاری که در رابط برنامه‌نویسی منتشر می‌شود جدا است؛ بنابراین تغییر نام یک ستون، قرارداد بیرونی سامانه را نمی‌شکند.

۵.۳. رابط برنامه‌نویسی و قرارداد آن

رابط برنامه‌نویسی سامانه شامل سی مسیر و سی‌وهشت عملیات است و به‌صورت رسمی در یک سند OpenAPI توصیف شده است. این سند مرجع قرارداد است: تایپ‌های سمت کاربر از آن تولید می‌شوند و دو آزمون خودکار، پاسخ‌های واقعی سرور را با آن تطبیق می‌دهند تا امکان واگرایی مستندات از پیاده‌سازی از بین برود.

درگاه یکپارچه‌سازی سامانه مادر جدا از نشست کاربران و با کلید دسترسی کار می‌کند و دو عملیات دارد: همگام‌سازی گروهی اطلاعات کارکنان بر اساس شماره پرسنلی، و استعلام وضعیت یک درخواست.

۶. پیاده‌سازی

۶.۱. لاگ ممیزی

تمام رویدادهای مؤثر ثبت می‌شوند: ورود به سامانه، ثبت و ارسال مجدد درخواست، شروع بررسی، تأیید، رد، بازگرداندن برای مدارک، ثبت پرداخت، بستن پرونده، و تغییر پیکربندی قواعد. برای هر رویداد، کاربر، زمان و مقدار قبل و بعد نگه‌داری می‌شود.

نکته کلیدی پیاده‌سازی این است که نوشتن رکورد ممیزی و اعمال تغییر بر داده، در یک تراکنش پایگاه داده انجام می‌شوند. در نتیجه اگر هر یک از این دو با خطا مواجه شود، هر دو بازگردانده می‌شوند و از نظر ساختاری وضعیتی که تغییر یافته اما ردپا نداشته باشد قابل ایجاد نیست.

۶.۲. احراز هویت و کنترل دسترسی

- گذرواژه‌ها با الگوریتم bcrypt هش می‌شوند و هرگز به‌صورت متن آشکار ذخیره نمی‌گردند.
- احراز هویت کاربران با توکن امضاشده و اعتبار هشت‌ساعته انجام می‌شود.
- کنترل دسترسی در دو لایه اعمال می‌شود: نقش در سطح مسیر، و مالکیت داده در سطح سرویس.
- کلید درگاه یکپارچه‌سازی نیز تنها به‌صورت هش‌شده ذخیره می‌شود.
- سامانه در حالت تولید در صورتی که کلید امضای توکن مقدار پیش‌فرض توسعه باشد بالا نمی‌آید.

۶.۳. دقت محاسبات مالی

مبالغ در سراسر سامانه به‌صورت عدد صحیح ریال نگه‌داری و پردازش می‌شوند. دلیل این انتخاب آن است که نمایش اعشاری دودویی نمی‌تواند مقادیر ده‌دهی را به‌طور دقیق نگه دارد و این خطا در جمع‌های مالی و مقایسه با سقف‌ها انباشته می‌شود؛ ریال نیز در عمل واحد خردتر ندارد. بر این اساس، تمام محاسبه با عدد صحیح انجام می‌شود و تنها یک نقطه گردکردن در کل سامانه وجود دارد.

> این تغییر را پس از تثبیت رفتار پیشین انجام دادم: نخست نوزده حالت مرزی محاسبه را در یک پرونده مرجع ثبت کردم، سپس تغییر را اعمال و نتیجه را مقایسه کردم. سیزده حالت بدون تغییر ماند و در شش حالت باقی‌مانده بیشترین اختلاف نیم ریال بود؛ یعنی تنها همان کسر زیر یک ریال حذف شد و تغییر معناداری در مبالغ رخ نداد.

۶.۴. زمان کسب‌وکار

دوره انتظار و سال قراردادی مفاهیمی بر پایه «روز تقویمی» هستند. اگر مرز روز با منطقه زمانی سرور محاسبه شود، یک درخواست نزدیک نیمه‌شب می‌تواند بر روی سرورهای مختلف نتیجه متفاوتی بدهد. به همین دلیل تمام تصمیم‌های مربوط به مرز روز در منطقه زمانی تهران ارزیابی می‌شوند و ساعت سامانه به‌صورت تزریق‌شده در اختیار سرویس‌ها قرار می‌گیرد تا رفتار آن قابل آزمون باشد.

۶.۵. رابط کاربری فارسی

رابط کاربری شامل شانزده صفحه و به‌طور کامل فارسی و راست‌به‌چپ است. تاریخ‌ها با تقویم هجری شمسی و اعداد با ارقام فارسی نمایش داده می‌شوند. قلم وزیرمتن همراه برنامه عرضه می‌شود تا نمایش مستقل از قلم‌های نصب‌شده روی دستگاه کاربر باشد. سه حالت نمایش روشن، تیره و خودکار پیش‌بینی شده است.

منوی کاربر بر اساس نقش او ساخته می‌شود و در صفحه جزئیات درخواست تنها دکمه‌هایی نمایش داده می‌شوند که در وضعیت جاری مجاز هستند؛ بنابراین کاربر با گزینه‌های نامعتبر مواجه نمی‌شود.

۷. تضمین کیفیت و آزمون

درستی سامانه در سه سطح راستی‌آزمایی شده است:

سطح آزمون	دامنه	روش
آزمون واحد	ریاضیات موتور قانون، تبدیل‌های مالی، پیکربندی	بدون وابستگی بیرونی؛ اجرای سریع
آزمون یکپارچگی	سرویس‌ها روی پایگاه داده واقعی	هر آزمون در یک تراکنش که در پایان بازگردانده می‌شود
آزمون سرتاسری	کل چرخه از رابط کاربری تا پایگاه داده	هدایت خودکار مرورگر در سیزده گام

آزمون سرتاسری، سناریوی کامل کسب‌وکار را می‌آزماید: ثبت درخواست توسط کارمند، بررسی و تأیید و پرداخت توسط کارشناس، مسیر رد با ذکر دلیل، انتشار نسخه جدید قانون توسط مدیر و راستی‌آزمایی این‌که درخواست بعدی با نرخ جدید محاسبه می‌شود، بررسی محدودیت‌های دسترسی، و صحت لاگ ممیزی و گزارش‌ها.

علاوه بر این، یک خط لوله یکپارچه‌سازی پیوسته تعریف شده است که در هر تغییر، بررسی سبک کد، ساخت پروژه و اجرای آزمون‌ها را به‌صورت خودکار انجام می‌دهد.

> برای اطمینان از این‌که آزمون‌های تطبیق قرارداد واقعاً کار می‌کنند، به‌صورت عمدی یک اختلاف میان پیاده‌سازی و سند ایجاد کردم و تأیید کردم که آزمون آن را تشخیص می‌دهد و شکست می‌خورد.

۸. ارزیابی نتایج در برابر معیارهای پذیرش

در پیشنهاد پروژه، پنج معیار پذیرش برای تعریف اتمام کار مشخص شده بود. وضعیت هر معیار به شرح زیر است:

معیار پذیرش	وضعیت	شاهد
محاسبه درست مبلغ پرداختی و مانده سقف برای حداقل پنج نوع خدمت با درصد و سقف‌های متفاوت	**محقق شد**	آزمون‌های واحد موتور قانون و پرونده مرجع نوزده حالت مرزی
اعمال تغییر قانون پوشش صرفاً از طریق پیکربندی و بدون تغییر کد	**محقق شد**	صفحه قوانین پوشش و گام اختصاصی در آزمون سرتاسری
اجرای کامل گردش کار شامل مسیرهای تأیید، رد و بازگشت	**محقق شد**	آزمون‌های یکپارچگی گردش کار و آزمون سرتاسری
ثبت کامل رویدادها در لاگ ممیزی و امکان بازسازی تاریخچه	**محقق شد**	صفحه تاریخچه اقدامات و آزمون ثبت رویدادهای چرخه کامل
اعمال درست محدودیت دسترسی بر اساس نقش	**محقق شد**	آزمون‌های دسترسی در سطح سرویس و سرتاسری

۸.۱. اندازه پیاده‌سازی

شاخص	مقدار	شاخص	مقدار
کد سمت سرور	حدود ۵٬۱۰۰ خط	مسیرهای رابط برنامه‌نویسی	۳۰ مسیر / ۳۸ عملیات
کد آزمون سمت سرور	حدود ۱٬۱۰۰ خط	جداول پایگاه داده	۱۲
کد سمت کاربر	حدود ۴٬۰۰۰ خط	صفحات رابط کاربری	۱۶
نقش‌های کاربری	۴	وضعیت‌های درخواست	۹
انواع خدمت پیش‌فرض	۵	انواع رویداد ممیزی	۱۰

۹. محدودیت‌ها و کارهای آینده

علاوه بر مواردی که در بخش محدوده به‌عنوان خارج از دامنه پروژه معرفی شدند، موارد زیر برای استفاده عملیاتی در یک سازمان واقعی لازم‌اند و به‌عنوان ادامه کار پیشنهاد می‌شوند:

مورد	توضیح
بارگذاری مدارک	جدول مدارک پیوست در پایگاه داده پیش‌بینی شده است، اما رابط بارگذاری فایل پیاده نشده است.
سامانه اعلان	اطلاع‌رسانی تغییر وضعیت درخواست از طریق رایانامه یا پیام کوتاه.
انتخابگر تاریخ شمسی	نمایش تاریخ‌ها شمسی است، اما برای ورود تاریخ از انتخابگر پیش‌فرض مرورگر استفاده می‌شود.
گزارش ماهانه بر پایه تقویم شمسی	گروه‌بندی ماهانه در گزارش‌ها بر اساس ماه میلادی انجام می‌شود.
آزمون بار	رفتار سامانه در حجم بالای داده و کاربر همزمان سنجیده نشده است.
احراز هویت دوعاملی	با توجه به حساسیت داده‌های درمانی توصیه می‌شود.
گردش کار چندسطحی	افزودن سطح تأیید مدیر در کنار کارشناس، برای سازمان‌هایی که چنین رویه‌ای دارند.
بارگذاری گروهی کارکنان	برای ورود اولیه داده و انتقال از سامانه پیشین.

همچنین از آن‌جا که هسته سامانه — یعنی ترکیب «سهمیه قابل پیکربندی» با «گردش کار چندمرحله‌ای» و «ممیزی کامل» — مستقل از حوزه بیمه است، می‌توان همان هسته را برای سایر خدمات رفاهی مانند وام، هزینه آموزش و هزینه سفر نیز به کار گرفت؛ این همان مسیری است که در پیشنهاد پروژه برای گسترش ماژول پیش‌بینی شده بود.

۱۰. نتیجه‌گیری

در این پروژه سامانه‌ای تحت وب برای مدیریت بیمه تکمیلی کارکنان طراحی و پیاده‌سازی کردم که سه مسئله اصلی روش دستی را هدف گرفته است: خطای محاسبه، نبود ردیابی، و هزینه بالای تغییر سیاست. محاسبه مبلغ قابل پرداخت و مانده سقف به‌صورت خودکار و بر پایه قواعد نسخه‌دار انجام می‌شود؛ گردش کار درخواست با مسیرهای تأیید، رد و بازگشت پیاده شده است؛ و تمام رویدادهای مؤثر به‌صورت تفکیک‌ناپذیر از تغییر داده در لاگ ممیزی ثبت می‌شوند.

دستاوردی که بیش از همه بر آن تأکید دارم، جدا کردن قواعد کسب‌وکار از کد است. با این طراحی، تغییر سیاست رفاهی سازمان به یک عملیات داده‌ای تبدیل می‌شود که مدیر سامانه خودش انجام می‌دهد و همان لحظه اعمال می‌گردد، در حالی که خودِ این تغییر نیز ممیزی شده و قابل بازبینی است.

پنج معیار پذیرشی که در پیشنهاد پروژه تعریف شده بود، محقق شده و با آزمون‌های خودکار در سه سطح راستی‌آزمایی شده‌اند. محدودیت‌های باقی‌مانده نیز در بخش پیشین به‌صورت صریح فهرست شده‌اند تا مسیر تکمیل سامانه برای استفاده عملیاتی روشن باشد.

پیوست الف — راهنمای اجرای سامانه

سامانه از سه سرویس تشکیل شده است: پایگاه داده، سرویس برنامه، و رابط کاربری. برای اجرا با استفاده از Docker:

فرمان	کارکرد
make up	ساخت تصویرها و اجرای هر سه سرویس
make seed	بارگذاری داده نمونه
make down	توقف سرویس‌ها

پس از اجرا، رابط کاربری روی نشانی زیر در دسترس است:

```
http://localhost:5173
```

حساب‌های نمونه برای بررسی نقش‌های مختلف:

نام کاربری	گذرواژه	نقش
admin	!Admin123	مدیر سامانه
reviewer	!Reviewer123	کارشناس بررسی
sara.ahmadi	!Employee123	کارمند — طرح استاندارد
reza.karimi	!Employee123	کارمند — طرح ویژه
auditor	!Auditor123	ممیز

گذرواژه‌های بالا صرفاً برای داده نمونه و محیط آزمایش است و در استقرار واقعی باید تغییر کنند. سرویس برنامه در حالت تولید با کلید امضای پیش‌فرض بالا نمی‌آید.

پیوست ب — ساختار پوشه‌های پروژه

مسیر	محتوا
backend/api/openapi.yaml	سند رسمی رابط برنامه‌نویسی
backend/cmd/api	نقطه ورود سرویس برنامه
backend/cmd/seed	بارگذاری داده نمونه
backend/internal/domain	موجودیت‌ها و مفاهیم کسب‌وکار
backend/internal/service	منطق کسب‌وکار (هفت بسته)
backend/internal/storage	دسترسی به پایگاه داده
backend/internal/transport	مسیرهای وب و نگاشت خطا
backend/migrations	مهاجرت‌های شِمای پایگاه داده
frontend/src/api	لایه ارتباط با سرویس برنامه
frontend/src/pages	صفحات رابط کاربری
frontend/src/components	اجزای مشترک رابط کاربری

مسیر	محتوا
/e2e	آزمون سرتاسری روی مرورگر
/deploy	تنظیمات استقرار
/docs	مستندات پروژه

پیوست ج — فهرست مستندات پروژه

سند	محتوا
ARCHITECTURE.md	معماری سامانه و تصمیم‌های طراحی
ERD.md	نمودار موجودیت–رابطه و شرح جداول
USE-CASES.md	سناریوهای کاربری و جریان‌های جایگزین
API-CONTRACT.md	شرح رابط برنامه‌نویسی
openapi.yaml	توصیف رسمی و ماشین‌خوان رابط برنامه‌نویسی
/adr	مستند تصمیم‌های معماری