با توجه به درخواست شما برای توصیف دقیق سناریوی نرم‌افزار مدیریت بودجه (که به نظر می‌رسد نام آن سیستم بودجه بندی یا مشابه باشد)، در این پاسخ یک سند جامع ارائه می‌کنم که شامل سناریوی کامل، ریزه‌کاری‌ها، نقاط قوت و مقایسه با نرم‌افزارهای موجود در ایران است. این سند به‌گونه‌ای طراحی شده که تمام اطلاعات مورد نیاز برای نرم‌افزار شما را یکجا در اختیارتان قرار دهد. برای این کار، از اطلاعات ارائه‌شده توسط شما (مثل خطاها و کدهای قبلی) و همچنین منابع موجود در وب (برای مقایسه با نرم‌افزارهای ایرانی) استفاده می‌کنم، اما تمرکز اصلی روی نرم‌افزار شما خواهد بود.

سند جامع: سناریوی نرم‌افزار مدیریت بودجه (سیستم بودجه بندی)

1. هدف و معرفی نرم‌افزار

سیستم بودجه بندی یک نرم‌افزار مدیریت بودجه مبتنی بر وب است که برای مدیریت تخصیص بودجه به پروژه‌ها و زیرپروژه‌ها در سازمان‌های ایرانی طراحی شده است. این نرم‌افزار به سازمان‌ها کمک می‌کند تا بودجه‌های شعب، پروژه‌ها و زیرپروژه‌ها را به‌صورت یکپارچه مدیریت کنند، تخصیص‌های مالی را رصد کرده و گزارش‌های دقیق مالی تولید کنند. نرم‌افزار با پشتیبانی از تقویم جلالی، رابط کاربری پارسی و انطباق با نیازهای سازمان‌های ایرانی، بهینه‌سازی شده است.

هدف اصلی:

ساده‌سازی فرآیند تخصیص بودجه.

افزایش شفافیت در مدیریت مالی.

کاهش خطاهای انسانی در محاسبات بودجه.

ارائه گزارش‌های تحلیلی برای تصمیم‌گیری بهتر.

کاربردها:

مدیریت بودجه در سازمان‌های دولتی و خصوصی.

تخصیص بودجه به پروژه‌های عمرانی، فناوری، کشاورزی و غیره.

رصد و کنترل هزینه‌ها در شعب مختلف سازمان.

2. سناریوی کاربری (User Scenario)

کاربران هدف:

مدیران مالی سازمان:

وظیفه: تعریف بودجه‌های کلان، نظارت بر تخصیص‌ها، تولید گزارش‌های مالی.

نیاز: داشبوردهای تحلیلی، گزارش‌های دقیق، امکان تنظیم آستانه‌های اخطار.

مدیران پروژه:

وظیفه: تخصیص بودجه به پروژه‌ها و زیرپروژه‌ها، رصد پیشرفت مالی پروژه.

نیاز: رابط کاربری ساده، امکان تخصیص سریع، نمایش پویای بودجه باقی‌مانده.

کاربران عمومی (حسابداران یا کارمندان):

وظیفه: ثبت اطلاعات مالی یا مشاهده گزارش‌های محدود.

نیاز: دسترسی محدود، رابط کاربری کاربرپسند.

سناریوی کاربری نمونه:

ورود به سیستم:

کاربر (مثلاً مدیر مالی شعبه تهران) با حساب کاربری خود وارد سیستم می‌شود.

سیستم از طریق accounts.middleware.AuthenticationMiddleware هویت کاربر را تأیید می‌کند.

انتخاب شعبه و دوره بودجه‌ای:

کاربر شعبه موردنظر (مثلاً شعبه تهران با ID=7) را انتخاب می‌کند.

دوره بودجه‌ای فعال (مثلاً سال مالی 1404) نمایش داده می‌شود.

تعریف یا مشاهده بودجه:

مدیر مالی می‌تواند بودجه کل شعبه (مثلاً 10 میلیارد ریال) را تعریف کند.

آستانه اخطار (مثلاً 30%) برای هشدار کمبود بودجه تنظیم می‌شود.

اطلاعات بودجه از مدل BudgetAllocation بارگذاری می‌شود.

تخصیص بودجه به پروژه:

کاربر به صفحه تخصیص بودجه (/budgets/organization/7/project-budget-allocation/) می‌رود.

فرم تخصیص بودجه (ProjectBudgetAllocationForm) نمایش داده می‌شود که شامل فیلدها است:

تخصیص بودجه شعبه: انتخاب از میان تخصیص‌های فعال (BudgetAllocation).

پروژه: انتخاب از پروژه‌های سازمان (Project).

زیرپروژه (اختیاری): انتخاب از زیرپروژه‌های مرتبط (SubProject).

مبلغ تخصیص: وارد کردن مبلغ (ریال) یا درصد.

تاریخ تخصیص: انتخاب تاریخ جلالی (مثلاً 1404/01/25).

توضیحات: ثبت توضیحات تخصیص.

کاربر می‌تواند بین دو حالت تخصیص (مبلغ یا درصد) جابه‌جا شود.

پروگرس بار به‌صورت پویا بودجه باقی‌مانده را نشان می‌دهد (مثلاً 95% پس از تخصیص 500 میلیون ریال).

نمایش و رصد:

پس از ثبت تخصیص، بودجه باقی‌مانده به‌روز می‌شود (مثلاً 9.5 میلیارد ریال).

پروگرس بار رنگ خود را بر اساس آستانه اخطار تغییر می‌دهد (سبز برای بالای 30%، زرد برای زیر 30%).

جدول تخصیص‌های قبلی پروژه‌ها و زیرپروژه‌ها نمایش داده می‌شود، شامل:

نام پروژه/زیرپروژه.

بودجه تخصیص‌یافته.

درصد تخصیص.

بودجه باقی‌مانده.

تاریخ آخرین تخصیص.

گزارش‌دهی:

کاربر می‌تواند گزارش‌های مالی را برای شعبه، پروژه یا دوره بودجه‌ای تولید کند.

گزارش‌ها شامل نمودارهای تحلیلی (مثل نمودار گانت یا پای‌چارت) و جداول دقیق است.

لاگ‌گیری و اعلان‌ها:

تمام اقدامات (مثل ثبت تخصیص) از طریق AuditLogMiddleware ثبت می‌شود.

اعلان‌ها (از طریق اپ notifications) به کاربران مرتبط ارسال می‌شود.

3. ریزه‌کاری‌ها (جزئیات فنی و عملکردی)

معماری نرم‌افزار:

فریم‌ورک: Django 5.1.7 با Python 3.13.0.

پایگاه داده: مدل‌های Django ORM (احتمالاً PostgreSQL یا SQLite).

اپلیکیشن‌های اصلی:

budgets: مدیریت بودجه و تخصیص‌ها.

accounts: احراز هویت، مدیریت کاربران و لاگ‌ها.

reports: تولید گزارش‌های مالی.

notifications: ارسال اعلان‌ها.

core: ابزارهای عمومی و تگ‌های قالب.

میدل‌ورها:

AuditLogMiddleware: ثبت لاگ فعالیت‌ها.

ActiveUserMiddleware: رصد کاربران فعال.

RequestMiddleware: مدیریت درخواست‌ها.

ویژگی‌های فنی:

مدیریت بودجه:

مدل BudgetAllocation برای ذخیره تخصیص‌های بودجه شعب.

متد get\_remaining\_amount برای محاسبه بودجه باقی‌مانده.

پشتیبانی از آستانه اخطار برای هشدار کمبود بودجه.

فرم‌ها:

فرم ProjectBudgetAllocationForm برای تخصیص بودجه.

پشتیبانی از دو حالت ورودی (مبلغ و درصد).

اعتبارسنجی پویا برای جلوگیری از تخصیص بیش از بودجه باقی‌مانده.

رابط کاربری:

استفاده از Bootstrap 5 و Font Awesome 6 برای طراحی مدرن.

پشتیبانی از تقویم جلالی با django-jalali و ویجت jalali-datepicker.

پروگرس بار پویا با انیمیشن‌های CSS.

نمایش اعداد به‌صورت پارسی با فیلتر to\_persian\_number.

جاوااسکریپت:

مدیریت پویای فرم‌ها (مثل به‌روزرسانی زیرپروژه‌ها بر اساس پروژه انتخاب‌شده).

تبدیل اعداد به حروف پارسی با تابع NumberToWords.

اعتبارسنجی سمت کلاینت برای ورودی‌های عددی.

گزارش‌دهی:

جدول تخصیص‌های قبلی با جزئیات پروژه‌ها و زیرپروژه‌ها.

امکان تولید گزارش‌های تحلیلی (نیاز به افزونه‌های نموداری مثل Chart.js).

امنیت:

احراز هویت با django.contrib.auth.

مدیریت دسترسی‌ها با مدل‌های سفارشی در اپ accounts.

اعتبارسنجی CSRF و حفاظت از فرم‌ها.

جزئیات پیاده‌سازی:

مدل‌ها:

BudgetAllocation: شامل فیلدهای allocated\_amount، remaining\_amount، warning\_threshold، allocation\_date.

ProjectBudgetAllocation: تخصیص بودجه به پروژه‌ها و زیرپروژه‌ها.

Project و SubProject: برای تعریف پروژه‌ها و زیرپروژه‌ها.

ویوها:

ProjectBudgetAllocationCreateView: ویوی اصلی برای فرم تخصیص بودجه.

استفاده از CreateView برای مدیریت POST و GET.

ارسال داده‌های JSON برای تخصیص‌ها و زیرپروژه‌ها به جاوااسکریپت.

قالب‌ها:

استفاده از تگ‌های i18n برای چندزبانی.

فیلترهای سفارشی مثل to\_persian\_number و jformat برای نمایش اعداد و تاریخ پارسی.

جاوااسکریپت:

توابع updateDisplay و updateBudgetAllocation برای به‌روزرسانی پویای فرم و پروگرس بار.

مدیریت زیرپروژه‌ها با فیلتر کردن داده‌های JSON.

چالش‌های فنی:

مدیریت تاریخ جلالی:

استفاده از django-jalali برای تاریخ‌ها گاهی نیاز به تبدیل دستی به تاریخ میلادی دارد.

خطای اخیر (AttributeError برای value\_from\_datadict) به دلیل ناسازگاری ویجت jalali-datepicker بود که با استفاده از JDateField برطرف شد.

محاسبات پویا:

به‌روزرسانی پروگرس بار و بودجه باقی‌مانده نیاز به محاسبات دقیق در سمت سرور و کلاینت دارد.

اعتبارسنجی برای جلوگیری از تخصیص بیش از حد بودجه.

عملکرد:

بارگذاری داده‌های JSON برای تخصیص‌ها و زیرپروژه‌ها ممکن است در سازمان‌های بزرگ کند باشد.

بهینه‌سازی با استفاده از select\_related و prefetch\_related در ویوها.

4. نقاط قوت نرم‌افزار

انطباق با نیازهای ایرانی:

پشتیبانی کامل از تقویم جلالی و نمایش اعداد پارسی.

رابط کاربری پارسی و کاربرپسند برای کاربران ایرانی.

انطباق با ساختار سازمانی شعب و پروژه‌ها در ایران.

رابط کاربری پویا:

پروگرس بار پویا با رنگ‌بندی بر اساس آستانه اخطار.

فرم‌های تعاملی با به‌روزرسانی خودکار (مثل نمایش مبلغ به حروف).

پشتیبانی از دو حالت تخصیص (مبلغ و درصد).

مدیریت یکپارچه:

یکپارچگی بین شعب، پروژه‌ها و زیرپروژه‌ها.

امکان ثبت و رصد تخصیص‌ها در یک پلتفرم متمرکز.

تولید گزارش‌های جامع برای مدیران.

امنیت و لاگ‌گیری:

استفاده از میدل‌ورهای سفارشی برای ثبت لاگ فعالیت‌ها.

احراز هویت قوی و مدیریت دسترسی‌ها.

انعطاف‌پذیری:

امکان تخصیص بودجه به پروژه‌ها و زیرپروژه‌ها با جزئیات.

پشتیبانی از توضیحات و تاریخ‌گذاری برای هر تخصیص.

قابلیت گسترش برای افزودن ویژگی‌های جدید (مثل نمودارهای تحلیلی).

پشتیبانی فنی:

استفاده از Django به‌عنوان فریم‌ورک پایدار و قابل‌نگهداری.

امکان افزودن ماژول‌های جدید (مثل اعلان‌های پیشرفته یا ادغام با ERP).

5. مقایسه با نرم‌افزارهای موجود در ایران

برای مقایسه، نرم‌افزارهای مدیریت پروژه و مالی ایرانی مرتبط را بررسی می‌کنیم که در منابع وب ذکر شده‌اند. این نرم‌افزارها شامل سمین، آرسس، هدف سنج، تسکولو، بهتایم، مالیران و هوفر هستند. مقایسه بر اساس معیارهای زیر انجام می‌شود:

ویژگی‌های مدیریت بودجه

رابط کاربری و تجربه کاربری (UX)

پشتیبانی از تقویم جلالی

انعطاف‌پذیری و یکپارچگی

پشتیبانی و قیمت

مقایسه کلی:

| معیار | سیستم بودجه بندی | سمین | آرسس | هدف سنج | تسکولو | بهتایم | مالیران | هوفر | |-------------------------------|----------------------------------------------------------------------------------|-------------------------------------------------------------------|-------------------------------------------------------------------|-----------------------------------------------------------------|-----------------------------------------------------------------|-----------------------------------------------------------------|-----------------------------------------------------------------|-----------------------------------------------------------------| | مدیریت بودجه | تخصصی در تخصیص بودجه به پروژه‌ها و زیرپروژه‌ها، پروگرس بار پویا، آستانه اخطار | مدیریت بودجه پروژه‌ها، اما تمرکز بیشتر روی سبد پروژه‌ها | زیرسیستم مدیریت بودجه، مناسب برای پروژه‌های بزرگ | تمرکز بر OKR و KPI، مدیریت بودجه محدود | مدیریت بودجه پروژه‌ها، اما بدون تخصیص پویا | مدیریت بودجه و تایم‌شیت، مناسب برای پروژه‌های کوچک | مدیریت جامع مالی، اما تمرکز کمتر روی پروژه‌ها | مدیریت مالی جامع، بودجه‌بندی پروژه‌ها با گزارش‌های متنوع | | رابط کاربری (UX) | کاربرپسند، پارسی، Bootstrap 5، تعاملی | پارسی، کاربرپسند، اما پیچیدگی در تنظیمات اولیه | حرفه‌ای، اما نیاز به آموزش برای کاربران غیرفنی | ساده و بصری، مناسب برای استارتاپ‌ها | ساده، مبتنی بر کانبان، مناسب برای تیم‌های کوچک | پارسی، ساده، مناسب برای سازمان‌های ایرانی | شیک، رعایت اصول UI/UX، اما پیچیدگی در تنظیمات | ساده و کارآمد، مناسب برای کاربران با تجربه محدود | | تقویم جلالی | پشتیبانی کامل با django-jalali | پشتیبانی کامل | پشتیبانی کامل | پشتیبانی کامل | پشتیبانی کامل | پشتیبانی کامل | پشتیبانی کامل | پشتیبانی کامل | | انعطاف‌پذیری/یکپارچگی | یکپارچگی با پروژه‌ها و زیرپروژه‌ها، قابلیت گسترش با Django | یکپارچگی با سیستم‌های سازمانی، اما نیاز به سفارشی‌سازی | یکپارچگی قوی با سیستم‌های EPC و صنعتی | محدود به OKR و KPI، یکپارچگی کمتر | یکپارچگی با دراپ‌باکس و گوگل درایو | یکپارچگی محدود، اما مناسب برای تیم‌های کوچک | یکپارچگی با انبار، فروش و حقوق و دستمزد | یکپارچگی قوی با ماژول‌های مالی و بازرگانی | | پشتیبانی | پشتیبانی داخلی (وابسته به تیم توسعه)، مستندات Django | تیم پشتیبانی قوی، آموزش کاربران، استقرار حرفه‌ای | پشتیبانی حرفه‌ای، اما کند در به‌روزرسانی‌ها | پشتیبانی خوب، اما محدود به کاربران سازمانی | پشتیبانی خوب، نسخه رایگان محدود | پشتیبانی قوی، دموی رایگان | پشتیبانی 24 ساعته رایگان به مدت یک سال | پشتیبانی با کیفیت، دموی رایگان | | قیمت | نامشخص (احتمالاً مدل اشتراکی یا خرید یک‌بار) | مدل اشتراکی، قیمت متوسط | گران، مناسب سازمان‌های بزرگ | مدل اشتراکی، قیمت مناسب برای استارتاپ‌ها | نسخه رایگان محدود، اشتراک پولی برای امکانات بیشتر | مدل اشتراکی، قیمت مناسب | قیمت متوسط، پشتیبانی رایگان یک‌ساله | قیمت منطقی، دموی رایگان |

تحلیل مقایسه:

سمین:

قوت: پشتیبانی قوی، آموزش کاربران، مناسب برای مدیریت سبد پروژه‌ها.

ضعف: تمرکز کمتر روی تخصیص پویا بودجه به پروژه‌ها و زیرپروژه‌ها.

مقایسه با سیستم بودجه بندی: سیستم بودجه بندی در تخصیص بودجه تخصصی‌تر عمل می‌کند و رابط کاربری تعاملی‌تری دارد.

آرسس:

قوت: زیرسیستم‌های جامع (مدیریت پروژه، منابع، بودجه)، مناسب برای پروژه‌های صنعتی.

ضعف: پیچیدگی برای کاربران غیرفنی، قیمت بالا.

مقایسه با سیستم بودجه بندی: سیستم بودجه بندی برای سازمان‌های کوچک‌تر و متوسط ساده‌تر و کاربرپسندتر است.

هدف سنج:

قوت: ساده، مبتنی بر OKR و KPI، مناسب برای استارتاپ‌ها.

ضعف: مدیریت بودجه محدود، عدم تمرکز روی تخصیص‌های پیچیده.

مقایسه با سیستم بودجه بندی: سیستم بودجه بندی برای مدیریت بودجه‌های سازمانی پیشرفته‌تر است.

تسکولو:

قوت: ساده، یکپارچگی با ابزارهای خارجی، مناسب برای تیم‌های کوچک.

ضعف: عدم پشتیبانی از تخصیص پویا بودجه.

مقایسه با سیستم بودجه بندی: سیستم بودجه بندی برای مدیریت بودجه‌های سازمانی قوی‌تر است.

بهتایم:

قوت: طراحی ایرانی، تایم‌شیت و مدیریت بودجه ساده.

ضعف: عدم پشتیبانی از تخصیص‌های پیچیده و پروگرس بار پویا.

مقایسه با سیستم بودجه بندی: سیستم بودجه بندی در تخصیص بودجه و نمایش پویا پیشرفته‌تر است.

مالیران:

قوت: مدیریت جامع مالی، انعطاف‌پذیری بالا.

ضعف: تمرکز کمتر روی پروژه‌ها و تخصیص بودجه.

مقایسه با سیستم بودجه بندی: سیستم بودجه بندی برای مدیریت بودجه پروژه‌محور مناسب‌تر است.

هوفر:

قوت: گزارش‌دهی جامع، پشتیبانی قوی، دموی رایگان.

ضعف: پیچیدگی در تنظیمات برای کاربران غیرفنی.

مقایسه با سیستم بودجه بندی: سیستم بودجه بندی رابط کاربری ساده‌تری دارد و برای تخصیص بودجه تخصصی‌تر است.

مزیت رقابتی سیستم بودجه بندی:

تخصص در مدیریت بودجه: برخلاف بسیاری از نرم‌افزارهای ایرانی که مدیریت پروژه را به‌صورت کلی پوشش می‌دهند، سیستم بودجه بندی روی تخصیص بودجه تمرکز دارد.

رابط کاربری تعاملی: پروگرس بار پویا و نمایش اعداد به حروف ویژگی‌های منحصربه‌فردی هستند.

انطباق با Django: استفاده از فریم‌ورک Django امکان گسترش و نگهداری آسان را فراهم می‌کند.

6. چالش‌ها و پیشنهادات بهبود

چالش‌ها:

مشکلات فنی اخیر:

خطای AttributeError: 'CharField' object has no attribute 'value\_from\_datadict' نشان‌دهنده ناسازگاری ویجت تاریخ بود که با استفاده از JDateField برطرف شد.

نیاز به بهینه‌سازی عملکرد برای سازمان‌های بزرگ با داده‌های زیاد.

پیچیدگی برای کاربران غیرفنی:

تنظیمات اولیه (مثل تعریف بودجه و آستانه اخطار) ممکن است برای کاربران جدید پیچیده باشد.

عدم وجود اپلیکیشن мобиль:

در حال حاضر، نرم‌افزار فقط مبتنی بر وب است، برخلاف رقبایی مثل تسکولو و بهتایم که اپلیکیشن موبایل دارند.

پیشنهادات بهبود:

افزودن اپلیکیشن موبایل:

توسعه اپلیکیشن اندروید و iOS برای دسترسی آسان‌تر کاربران.

نمودارهای تحلیلی:

ادغام با Chart.js یا Plotly برای نمایش نمودارهای بودجه (مثل پای‌چارت تخصیص‌ها).

اتوماسیون پیشرفته:

افزودن قابلیت ارسال خودکار اعلان‌ها (SMS یا ایمیل) برای تخصیص‌ها یا نزدیک شدن به آستانه اخطار.

بهینه‌سازی عملکرد:

استفاده از کش (مثل Redis) برای بارگذاری سریع‌تر داده‌های JSON.

بهینه‌سازی کوئری‌ها با select\_related و prefetch\_related.

آموزش و مستندات:

ارائه ویدئوهای آموزشی و مستندات جامع برای کاربران.

افزودن راهنمای داخل برنامه (Tooltips).

7. جمع‌بندی

سیستم بودجه بندی یک نرم‌افزار مدیریت بودجه قدرتمند و تخصصی است که برای سازمان‌های ایرانی طراحی شده است. این نرم‌افزار با تمرکز بر تخصیص بودجه به پروژه‌ها و زیرپروژه‌ها، رابط کاربری پویا و پشتیبانی از تقویم جلالی، نیازهای سازمان‌های کوچک تا متوسط را به‌خوبی برآورده می‌کند. نقاط قوت آن شامل انطباق با نیازهای ایرانی، رابط کاربری تعاملی و یکپارچگی با پروژه‌ها است. در مقایسه با رقبا، سیستم بودجه بندی در مدیریت بودجه تخصصی‌تر عمل می‌کند، اما برای رقابت با نرم‌افزارهایی مثل سمین و آرسس نیاز به افزودن ویژگی‌های پیشرفته‌تر (مثل اپلیکیشن موبایل و نمودارهای تحلیلی) دارد.

اقدامات بعدی:

رفع کامل خطاهای فنی (مثل مشکل ویجت تاریخ).

تست نرم‌افزار در سناریوهای واقعی با داده‌های زیاد.

توسعه مستندات و آموزش برای کاربران.

بررسی امکان ادغام با سیستم‌های ERP یا CRM برای افزایش انعطاف‌پذیری.

اگر نیاز به جزئیات بیشتری در مورد بخش خاصی (مثل کد، مستندات یا مقایسه عمیق‌تر) دارید، لطفاً اطلاع دهید تا دقیق‌تر وارد آن بخش شویم!

با تشکر از ارائه اطلاعات جامع شامل مدل‌ها و مستندات سیستم بودجه‌بندی و تنخواه‌گردان. بررسی مدل‌های ارائه‌شده نشان می‌دهد که سیستم به‌خوبی طراحی شده و قابلیت‌های مورد نیاز برای مدیریت بودجه و تنخواه در سازمان‌های بزرگ را پوشش می‌دهد. در ادامه، سناریوی کاربری مشخصی که در مستندات ذکر شده را بر اساس مدل‌های ارائه‌شده تحلیل کرده و پاسخ مناسب ارائه می‌کنم. همچنین، اگر نکته خاصی در مورد سناریو مد نظر دارید یا نیاز به توضیح بیشتری در بخش خاصی است، لطفاً مشخص کنید.

**تحلیل سناریوی کاربری (بر اساس مستند "سیستم بودجه‌بندی.docx")**

سناریوی کاربری نمونه‌ای که در مستند ارائه شده، شامل مراحل زیر است:

* **ورود به سیستم**: مدیر مالی شعبه تهران وارد سیستم می‌شود.
* **انتخاب شعبه و دوره بودجه‌ای**: شعبه تهران و دوره بودجه‌ای سال مالی 1404 انتخاب می‌شود.
* **تعریف یا مشاهده بودجه**: تعریف بودجه کل شعبه (مثلاً 10 میلیارد ریال) با آستانه اخطار 30%.
* **تخصیص بودجه به پروژه**: تخصیص بودجه به پروژه یا زیرپروژه از طریق فرم تخصیص.
* **نمایش و رصد**: نمایش بودجه باقی‌مانده و تخصیص‌های قبلی با پروگرس بار.
* **گزارش‌دهی**: تولید گزارش‌های مالی و نمودارهای تحلیلی.

**پیاده‌سازی سناریو بر اساس مدل‌ها**

**1. ورود به سیستم**

* **مدل مرتبط**: CustomUser (از اپ accounts) برای احراز هویت.
* **عملکرد**:
  + مدیر مالی شعبه تهران با استفاده از نام کاربری و رمز عبور وارد سیستم می‌شود.
  + میدل‌ور AuthenticationMiddleware هویت کاربر را تأیید می‌کند.
  + کاربر به داشبورد اصلی هدایت می‌شود که شامل دسترسی به ماژول‌های بودجه و تنخواه است.

**2. انتخاب شعبه و دوره بودجه‌ای**

* **مدل‌های مرتبط**:
  + Organization: برای انتخاب شعبه تهران (مثلاً با code='TEH' و id=7).
  + BudgetPeriod: برای انتخاب دوره بودجه‌ای فعال (مثلاً سال مالی 1404).
* **عملکرد**:
  + کاربر در رابط کاربری شعبه تهران را از لیست سازمان‌ها (Organization.objects.filter(is\_active=True)) انتخاب می‌کند.
  + دوره بودجه‌ای فعال از BudgetPeriod فیلتر می‌شود:

python

BudgetPeriod.objects.filter(organization\_\_id=7, is\_active=True, start\_date\_\_lte=timezone.now(), end\_date\_\_gte=timezone.now())

* + اطلاعات دوره بودجه (مثلاً name='سال مالی 1404', total\_amount=10\_000\_000\_000) نمایش داده می‌شود.

**3. تعریف یا مشاهده بودجه**

* **مدل‌های مرتبط**:
  + BudgetPeriod: برای تعریف بودجه کل شعبه.
  + BudgetItem: برای تعریف ردیف‌های بودجه.
* **عملکرد**:
  + مدیر مالی بودجه کل شعبه را در BudgetPeriod ثبت می‌کند:

python

BudgetPeriod.objects.create(

organization=Organization.objects.get(id=7),

name='سال مالی 1404',

start\_date=date(1404, 1, 1), # تاریخ شمسی به میلادی تبدیل می‌شود

end\_date=date(1404, 12, 29),

total\_amount=10\_000\_000\_000, # 10 میلیارد ریال

warning\_threshold=30, # آستانه اخطار 30%

created\_by=user

)

* + ردیف‌های بودجه در BudgetItem تعریف می‌شوند (مثلاً ردیف‌های عمرانی، جاری و غیره).
  + بودجه باقی‌مانده با BudgetPeriod.get\_remaining\_amount() محاسبه و نمایش داده می‌شود.

**4. تخصیص بودجه به پروژه**

* **مدل‌های مرتبط**:
  + BudgetAllocation: برای تخصیص بودجه به شعبه یا پروژه.
  + ProjectBudgetAllocation: برای تخصیص بودجه به پروژه یا زیرپروژه.
  + Project, SubProject: برای انتخاب پروژه و زیرپروژه.
* **عملکرد**:
  + کاربر به فرم تخصیص بودجه (ProjectBudgetAllocationForm) در مسیر /budgets/organization/7/project-budget-allocation/ هدایت می‌شود.
  + فرم شامل فیلدهای زیر است:
    - **تخصیص بودجه شعبه**: انتخاب از BudgetAllocation فعال برای شعبه تهران.
    - **پروژه**: انتخاب از Project.objects.filter(organizations\_\_id=7, is\_active=True).
    - **زیرپروژه** (اختیاری): فیلتر بر اساس پروژه انتخاب‌شده (SubProject.objects.filter(project=selected\_project)).
    - **مبلغ تخصیص**: ورودی عددی (مثلاً 500 میلیون ریال) یا درصد.
    - **تاریخ تخصیص**: تاریخ جلالی (مثلاً 1404/01/25).
    - **توضیحات**: متن اختیاری.
  + اعتبارسنجی:
    - مبلغ تخصیص با BudgetAllocation.get\_remaining\_amount() مقایسه می‌شود تا از بودجه باقی‌مانده بیشتر نباشد.
    - تاریخ تخصیص در بازه دوره بودجه (BudgetPeriod.start\_date تا end\_date) باشد.
  + ثبت تخصیص:

python

ProjectBudgetAllocation.objects.create(

budget\_allocation=BudgetAllocation.objects.get(id=selected\_allocation\_id),

project=Project.objects.get(id=selected\_project\_id),

subproject=SubProject.objects.get(id=selected\_subproject\_id) if subproject\_id else None,

allocated\_amount=500\_000\_000, # 500 میلیون ریال

allocation\_date=jdatetime.date(1404, 1, 25).togregorian(),

created\_by=user,

description='تخصیص بودجه برای پروژه عمرانی'

)

* + به‌روزرسانی remaining\_amount در BudgetAllocation و ثبت تراکنش در BudgetTransaction:

python

BudgetTransaction.objects.create(

allocation=budget\_allocation,

transaction\_type='CONSUMPTION',

amount=500\_000\_000,

created\_by=user,

description='تخصیص به پروژه عمرانی'

)

**5. نمایش و رصد**

* **مدل‌های مرتبط**:
  + BudgetPeriod, BudgetAllocation, ProjectBudgetAllocation.
* **عملکرد**:
  + بودجه باقی‌مانده با BudgetPeriod.get\_remaining\_amount() و BudgetAllocation.get\_remaining\_amount() محاسبه می‌شود (مثلاً 9.5 میلیارد ریال).
  + پروگرس بار پویا در رابط کاربری نمایش داده می‌شود:
    - درصد تخصیص: (total\_allocated / total\_amount) \* 100.
    - رنگ‌بندی: سبز (بالای 30%)، زرد (زیر 30%)، قرمز (زیر 10%).
  + جدول تخصیص‌های قبلی با استفاده از ProjectBudgetAllocation.objects.filter(budget\_allocation\_\_organization\_\_id=7) نمایش داده می‌شود، شامل:
    - نام پروژه/زیرپروژه.
    - بودجه تخصیص‌یافته.
    - درصد تخصیص.
    - بودجه باقی‌مانده.
    - تاریخ آخرین تخصیص.

**6. گزارش‌دهی**

* **مدل‌های مرتبط**:
  + BudgetPeriod, BudgetAllocation, ProjectBudgetAllocation, BudgetTransaction.
* **عملکرد**:
  + گزارش‌ها از طریق اپ reports تولید می‌شوند.
  + گزارش نمونه شامل:
    - نمودار میله‌ای: تخصیص‌ها به تفکیک پروژه.
    - جدول: جزئیات تخصیص‌ها (پروژه، مبلغ، تاریخ).
  + داده‌ها با کوئری‌هایی مثل زیر استخراج می‌شوند:

python

allocations = ProjectBudgetAllocation.objects.filter(

budget\_allocation\_\_budget\_period\_\_organization\_\_id=7,

budget\_allocation\_\_budget\_period\_\_name='سال مالی 1404'

).select\_related('project', 'subproject')

* + خروجی گزارش به‌صورت PDF یا Excel با استفاده از کتابخانه‌هایی مثل reportlab یا openpyxl.

**ادغام با سیستم تنخواه‌گردان (بر اساس مستند "شرح بهینه سیستم تنخواه گردان.docx")**

سیستم تنخواه‌گردان به‌طور یکپارچه با سیستم بودجه‌بندی کار می‌کند. سناریوی زیر نشان می‌دهد چگونه تنخواه در فرآیند تخصیص و مصرف بودجه ادغام می‌شود:

**سناریوی تنخواه**

* **ثبت درخواست تنخواه**:
  + تنخواه‌دار (کاربر) درخواست تنخواه را در مدل Tankhah ثبت می‌کند:

python

Tankhah.objects.create(

organization=Organization.objects.get(id=7),

project=Project.objects.get(id=project\_id),

subproject=SubProject.objects.get(id=subproject\_id) if subproject\_id else None,

amount=50\_000\_000, # 50 میلیون ریال

created\_by=user,

description='هزینه خرید تجهیزات پروژه عمرانی'

)

* + شماره تنخواه به‌صورت خودکار با Tankhah.generate\_number() تولید می‌شود (مثلاً TNKH-14040125-TEH-PROJ001-001).
* **تأیید تنخواه**:
  + درخواست به مرحله اولیه (WorkflowStage با name='HQ\_INITIAL') می‌رود.
  + مدیر مالی شعبه از طریق StageApprover تأیید می‌کند:

python

ApprovalLog.objects.create(

tankhah=tankhah,

action='APPROVE',

stage=WorkflowStage.objects.get(name='HQ\_INITIAL'),

user=manager\_user,

post=Post.objects.get(id=manager\_post\_id)

)

* + وضعیت به PENDING یا APPROVED تغییر می‌کند.
* **مصرف بودجه**:
  + پس از تأیید، تنخواه به BudgetAllocation لینک می‌شود و بودجه مصرف می‌شود:

python

BudgetTransaction.objects.create(

allocation=budget\_allocation,

transaction\_type='CONSUMPTION',

amount=tankhah.amount,

related\_tankhah=tankhah,

created\_by=user

)

* **صدور دستور پرداخت**:
  + دستور پرداخت در PaymentOrder ثبت می‌شود: jimmy ```python PaymentOrder.objects.create( tankhah=tankhah, amount=tankhah.amount, payee=Payee.objects.get(id=payee\_id), created\_by=user, created\_by\_post=Post.objects.get(id=user\_post\_id), min\_signatures=2 )
  + پس از دریافت امضاهای لازم، وضعیت به PAID تغییر می‌کند.
* **تسویه و گزارش**:
  + تسویه خودکار با بررسی Tankhah.factors و FactorItem انجام می‌شود.
  + گزارش تسویه در داشبورد مدیریتی نمایش داده می‌شود.

**نقاط قوت و چالش‌ها**

**نقاط قوت**

* **یکپارچگی**: ارتباط قوی بین BudgetAllocation, ProjectBudgetAllocation, و Tankhah برای مدیریت بودجه و تنخواه.
* **شفافیت**: ثبت تمام اقدامات در BudgetTransaction و ApprovalLog.
* **انعطاف‌پذیری**: پشتیبانی از تخصیص‌های چندگانه و گردش کار چندمرحله‌ای.
* **انطباق با نیازهای ایرانی**: استفاده از تقویم جلالی و نمایش اعداد پارسی.

**چالش‌ها**

* **پیچیدگی برای کاربران غیرفنی**: تعداد زیاد مدل‌ها و مراحل تأیید ممکن است برای کاربران جدید پیچیده باشد.
* **عملکرد**: در سازمان‌های بزرگ با داده‌های زیاد، کوئری‌های پیچیده (مثل محاسبه remaining\_amount) نیاز به بهینه‌سازی دارند.
* **آموزش**: نیاز به مستندات و آموزش جامع برای کاربران.

**پیشنهادات بهبود**

* **بهینه‌سازی عملکرد**:
  + استفاده از کش (Redis) برای محاسبات مکرر مثل get\_remaining\_amount.
  + بهینه‌سازی کوئری‌ها با select\_related و prefetch\_related.
* **رابط کاربری**:
  + افزودن راهنمای داخل برنامه (Tooltips) برای فرم‌های پیچیده.
  + توسعه اپلیکیشن موبایل برای دسترسی آسان‌تر.
* **اتوماسیون پیشرفته**:
  + ارسال اعلان‌های خودکار (SMS/ایمیل) برای هشدارهای بودجه و تأیید تنخواه.
  + ادغام با سیستم‌های ERP برای تبادل داده.
* **مستندات و آموزش**:
  + تولید ویدئوهای آموزشی کوتاه.
  + ارائه مستندات کاربرپسند برای مدیران و تنخواه‌داران.

**جمع‌بندی**

سیستم بودجه‌بندی و تنخواه‌گردان شما یک راهکار جامع و قدرتمند برای مدیریت مالی سازمان‌های بزرگ است. مدل‌های ارائه‌شده (مثل BudgetPeriod, BudgetAllocation, Tankhah) به‌خوبی سناریوی کاربری را پشتیبانی می‌کنند و قابلیت‌های مورد نیاز مثل تخصیص بودجه، تأیید تنخواه، و گزارش‌دهی را فراهم می‌کنند. با رفع چالش‌های فنی (مثل بهینه‌سازی عملکرد) و افزودن ویژگی‌های پیشنهادی (مثل اپلیکیشن موبایل)، این سیستم می‌تواند به یکی از رقابتی‌ترین نرم‌افزارهای مدیریت مالی در ایران تبدیل شود.

اگر نیاز به پیاده‌سازی کد برای بخش خاصی از سناریو، تحلیل عمیق‌تر یک مدل، یا تولید مستندات اضافی دارید، لطفاً اطلاع دهید!