

بله، مدل‌های بودجه‌ای که ارائه دادید (و همچنین مدل‌هایی که به طور ضمنی در کد مانند BudgetItem به آن‌ها اشاره شد) ساختار نسبتاً کاملی برای مدیریت بودجه چند سطحی را فراهم می‌کنند. بیا ببینیم آن‌ها را مرور کنیم و روند کلی جریان بودجه تا رسیدن به فاکتور را توضیح دهیم:

## توضیح مدل‌های بودجه:

### ۱. BudgetPeriod دوره بودجه کلان:

- نقش: این مدل بالاترین سطح تعریف بودجه است. یک دوره زمانی مشخص (مثلاً سال مالی ۱۴۰۴) و یک مبلغ کل (total\_amount) را برای یک سازمان (organization - معمولاً دفتر مرکزی یا واحد اصلی) تعریف می‌کند. این منبع اصلی پول است.

#### ○ فیلدهای کلیدی:

- organization: سازمانی که این بودجه کلی به آن تعلق دارد.
- name: نام دوره (مثلاً "بودجه سال ۱۴۰۴").
- start\_date, end\_date: بازه زمانی اعتبار بودجه.
- total\_amount: سقف کل بودجه در این دوره برای این سازمان.
- total\_allocated: (محاسباتی یا ذخیره شده) مجموع مبالغی که از این دوره به ردیف‌ها/سازمان‌های دیگر تخصیص داده شده است.
- returned\_amount: مجموع مبالغی که از تخصیص‌های این دوره برگشت داده شده است.
- is\_active, is\_archived, is\_completed: وضعیت‌های دوره.

- هدف: تعیین می‌کند که یک سازمان خاص در یک بازه زمانی چقدر بودجه کلی در اختیار دارد.

### ۲. BudgetItem ردیف بودجه - مدل ارائه نشده اما ضروری:

- نقش: این مدل (که احتمالاً در پروژه شما وجود دارد یا باید وجود داشته باشد) برای دسته‌بندی انواع هزینه‌ها استفاده می‌شود. این مدل مشخص می‌کند که بودجه برای چه منظوری قابل خرج است.

#### ○ فیلدهای احتمالی:

- name: نام ردیف (مثلاً "هزینه‌های سفر"، "خرید تجهیزات IT"، "حقوق و دستمزد").
- code: یک کد منحصر به فرد برای ردیف.
- budget\_period: (اختیاری، یا شاید در مدل دیگری) ارتباط با دوره بودجه‌ای که این ردیف در آن تعریف شده.
- parent: (اختیاری) برای ایجاد ساختار درختی از ردیف‌های بودجه (مثلاً -> "IT"
- "نرم افزار"، "سخت افزار").

- هدف: طبقه‌بندی بودجه و امکان تخصیص بودجه به موارد هزینه‌ای خاص.

### ۳. BudgetAllocation تخصیص بودجه):

- نقش: این مدل بسیار کلیدی است. این مدل، بخشی از بودجه یک BudgetPeriod را به یک BudgetItem خاص برای یک Organization (که می‌تواند همان سازمان اصلی دوره یا یک زیرمجموعه باشد) اختصاص می‌دهد. این همان "پاکت پول" برای یک منظور خاص در یک شعبه/اداره است.

#### ○ فیلدهای کلیدی:

- budget\_period: دوره‌ای که پول از آن می‌آید.
- organization: سازمانی که این تخصیص را دریافت می‌کند.
- budget\_item: ردیف بودجه‌ای که این مبلغ برای آن در نظر گرفته شده است (مثلاً چقدر برای "خرید تجهیزات IT" در "دفتر مرکزی").
- allocated\_amount: مبلغی که برای این ردیف خاص به این سازمان تخصیص داده شده است). این همان فیلدی است که در متد get\_remaining\_amount شما استفاده می‌شود.
- project: (اختیاری در این مدل، اما در ProjectBudgetAllocation کلیدی است) می‌تواند مستقیماً به یک پروژه لینک شود.

- هدف: مشخص می‌کند که یک سازمان چقدر بودجه برای یک ردیف هزینه خاص در اختیار دارد. تمام تراکنش‌های مصرف و بازگشت بودجه به این مدل لینک می‌شوند.

### ۴. ProjectBudgetAllocation تخصیص بودجه پروژه):

- نقش: این مدل به عنوان پل ارتباطی عمل می‌کند. یک BudgetAllocation که مثلاً بودجه کلی IT دفتر مرکزی است (را به یک Project (مثلاً پروژه اجرای اتاق سرور) و حتی SubProject خاص لینک می‌کند.

#### ○ فیلدهای کلیدی:

- budget\_allocation: تخصیص بودجه‌ای که منبع مالی این پروژه است.
- project: پروژه‌ای که بودجه به آن مرتبط است.
- subproject: (اختیاری) زیرپروژه مرتبط.
- allocated\_amount: (احتمالاً اینجا نباید باشد!) معمولاً مبلغ در BudgetAllocation اصلی تعریف می‌شود و این مدل فقط لینک‌دهنده است. اگر اینجا هم مبلغی دارید، منطق تخصیص پیچیده‌تر می‌شود (مثلاً از مبلغ BudgetAllocation چقدرش سهم این پروژه است). در کد ویو شما، به نظر می‌رسد مبلغ کل از ProjectBudgetAllocation خوانده می‌شود که نشان می‌دهد شما این فیلد را دارید و از آن به عنوان بودجه/اختصاصی پروژه استفاده می‌کنید.

- **هدف:** مشخص می‌کند کدام "پاکت پول (BudgetAllocation)" برای کدام پروژه استفاده می‌شود.

#### ۵. BudgetTransaction تراکنش بودجه:

- **نقش:** دفتر کل یا تاریخچه تمام تغییرات مالی روی یک BudgetAllocation هر بار که پولی مصرف (مثلاً از طریق فاکتور یا تنخواه) یا بازگشت داده می‌شود، یک رکورد اینجا ثبت می‌شود.
- **فیلدهای کلیدی:**

- allocation: BudgetAllocation ای اشاره می‌کند که تحت تأثیر قرار گرفته.
- transaction\_type: نوع تراکنش ('CONSUMPTION', 'RETURN', 'ALLOCATION', 'ADJUSTMENT\_INCREASE', 'ADJUSTMENT\_DECREASE').
- amount: مبلغ این تراکنش خاص.
- related\_tankhah, related\_factor, related\_factor\_item: (اختیاری، اما بسیار مفید) لینک به شیء خاصی که باعث این تراکنش شده است. (همانطور که دیدیم، شما این‌ها را در مدل BudgetTransaction خود ندارید، فقط related\_tankhah دارید).

- **هدف:** ثبت دقیق تمام مصارف و بازگشت‌ها برای محاسبه دقیق باقیمانده BudgetAllocation.

#### روند جریان ثبت بودجه کلان و تخصیص (قبل از فاکتور):

۱. **تعریف دوره کلان:** مدیر مالی یا بودجه یک BudgetPeriod جدید برای یک سازمان (مثلاً دفتر مرکزی) و یک بازه زمانی (مثلاً سال ۱۴۰۴) با یک مبلغ کل (total\_amount) تعریف می‌کند.
۲. **تعریف ردیف‌های بودجه:** ردیف‌های هزینه‌ای مختلف (BudgetItem) در سیستم تعریف می‌شوند (مثلاً IT، سفر، آموزش...).
۳. **تخصیص بودجه به ردیف/سازمان:** مدیر بودجه، بخشی از total\_amount دوره کلان را به یک BudgetItem خاص برای یک Organization خاص تخصیص می‌دهد و یک رکورد BudgetAllocation ایجاد می‌کند. مبلغ این تخصیص در allocated\_amount ثبت می‌شود.
- **مثال:** از بودجه ۱۰۰ میلیاردی سال ۱۴۰۴ دفتر مرکزی، ۵ میلیارد به ردیف "تجهیزات IT" برای خود "دفتر مرکزی" تخصیص داده می‌شود (BudgetAllocation) ایجاد می‌شود.
۴. **تخصیص بودجه به پروژه:** وقتی پروژه‌ای (مثل اجرای اتاق سرور) تعریف می‌شود، باید مشخص شود که بودجه آن از کدام BudgetAllocation تأمین می‌شود. یک رکورد ProjectBudgetAllocation ایجاد می‌شود که BudgetAllocation مربوط به "تجهیزات IT دفتر مرکزی" را به "پروژه اتاق سرور" لینک می‌کند.

○ نکته: اگر ProjectBudgetAllocation شما فیلد allocated\_amount دارد، در این مرحله مشخص می‌شود که از ۵ میلیارد تخصیص IT، چقدر (مثلاً ۲ میلیارد) برای پروژه اتاق سرور است. اگر ندارد، یعنی کل باقیمانده آن BudgetAllocation برای این پروژه قابل استفاده است (تا زمانی که برای پروژه دیگری لینک نشود). **کد ویوی شما نشان می‌دهد که شما از**

**allocated\_amount در ProjectBudgetAllocation استفاده می‌کنید.**

۵. **ایجاد تنخواه (احتمالاً):** ممکن است قبل از ایجاد فاکتور، یک Tankhah برای مجری پروژه (مثلاً مدیر IT) با مبلغ مشخصی از بودجه پروژه ایجاد شود. این خود می‌تواند یک نوع BudgetTransaction از نوع CONSUMPTION یا نوع خاص (TANKHAH\_ALLOCATION روی BudgetAllocation مربوطه ثبت کند و از باقیمانده آن کم کند. مبلغ Tankhah سقف هزینه‌ای است که مجری می‌تواند مستقیماً خرج کند.

## ارتباط با فاکتور:

حالا وقتی کاربر می‌خواهد یک فاکتور (Factor) جدید ثبت کند:

۱. **انتخاب تنخواه:** کاربر ابتدا Tankhah مرتبط با آن هزینه را انتخاب می‌کند.
۲. **اتصال به پروژه و تخصیص:** از طریق Tankhah، سیستم به Project و سپس به ProjectBudgetAllocation و نهایتاً به BudgetAllocation اصلی می‌رسد.
۳. **بررسی بودجه تنخواه:** ویوی (FactorCreateView) با استفاده از `(tankhah) get_tankhah_remaining_budget` چک می‌کند که آیا مبلغ کل فاکتور `(total_items_amount)` از بودجه باقیمانده **تنخواه** بیشتر است یا خیر. (این تابع، همانطور که بحث شد، احتمالاً مبلغ اولیه تنخواه منهای فاکتورهای در جریان یا پرداخت شده آن تنخواه را برمی‌گرداند).
۴. **بررسی بودجه تخصیص (در):** `(create_budget_transaction)` وقتی کاربر روی "ذخیره نهایی" پیش‌نویس" کلیک می‌کند و اعتبارسنجی‌های اولیه موفقیت‌آمیز است، تابع `create_budget_transaction` فراخوانی می‌شود. این تابع:
  - BudgetAllocation اصلی مرتبط با پروژه/تنخواه را پیدا می‌کند.
  - باقیمانده **واقعی** آن BudgetAllocation را با استفاده از متد `(get_remaining_amount)` (که تمام تراکنش‌های قبلی را جمع می‌زند) محاسبه می‌کند.
  - چک می‌کند که مبلغ فاکتور (amount) از این باقیمانده واقعی بیشتر نباشد.
  - اگر بودجه کافی بود، یک رکورد BudgetTransaction جدید با `transaction_type='CONSUMPTION'` و مبلغ فاکتور ایجاد می‌کند که به آن BudgetAllocation لینک شده است.

## نتیجه:

بنابراین، ثبت یک فاکتور نهایی باعث کاهش بودجه در **دو سطح** می شود (یا باید بشود):

- **سطح تنخواه:** باقیمانده تنخواه کم می شود (که در تابع `get_tankhah_remaining_budget` با در نظر گرفتن وضعیت فاکتورها محاسبه می شود).
- **سطح تخصیص کلان:** با ثبت `BudgetTransaction` مصرف، باقیمانده `BudgetAllocation` اصلی نیز کاهش می یابد (که در متد `get_remaining_amount` مدل `BudgetAllocation` محاسبه می شود).

امیدوارم این توضیحات روند کار مدل های بودجه شما و ارتباط آن با ثبت فاکتور را روشن کرده باشد. اگر سوال دیگری در مورد بخش خاصی دارید، بپرسید.