گزارش مباحث فنی در حوزه‌ی "دسته‌بندی موجودیت‌ها"

Version 1.5

|  |  |  |  |
| --- | --- | --- | --- |
| تاريخ | نسخه | توضيحات | نویسنده |
| 2/6/1388 | Version 1.0 |  | محمد صلواتی |
| 4/6/1388 | Version1.5 | اعمال تغییرات مورد توافق در جلسه‌ی با حضور آقایان نیکنام، جعفری و صلواتی | ناصر صفری‌نیا |

# مقدمه

در این مستند نتیجه‌گیری های بدست آمده در مورد نحوه‌ی دسته‌بندی موجودیت ها و همچنین طراحی جداول مربوط به آن تشریح گردیده است.

# طراحی دیتابیس

در پی سه جلسه بحث و تبادل نظر بر روی ساختار دیتابیس توسط مهندس جعفری,مهندس نیکنام و مهندس صفری در نهایت بر روی موارد زیر توافق گردید:

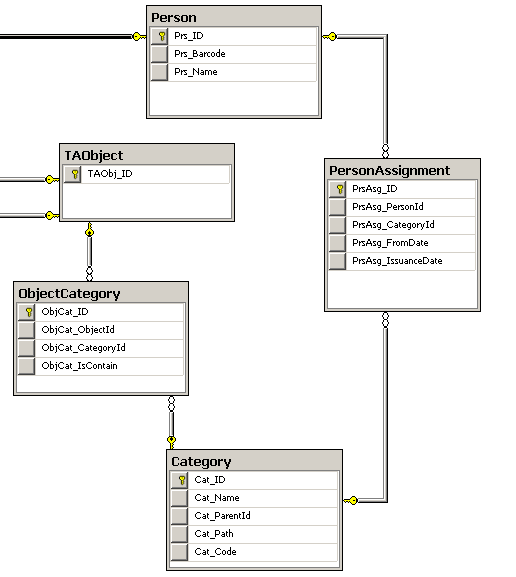
* ساختار سلسله مراتبی تعریف شده توسط کاربر بمنظور دسته بندی اشخاص بصورت جداگانه از ساختار سلسه مراتبی تعریف دیگر موجودیت‌ها(قانون، مفهوم و ...) نگهداری شود
* از دیدگاه دیتابیس , تعریف قوانین و تقویم کاری برای هر شخص جداگانه تعریف می‌گردد.
* همیشه گروهی از قوانین به شخص خاصی انتساب داده می‌شود و مفاهیم هر شخص از طریق مفاهیمی که در قوانین او استفاده شده است واکشی می‌گردد.
* ارتباط هر شخص با دسته‌بندی قوانین بوسیله جدول PersonAssignment (حکم) برقرار میگردد
* از دیدگاه UI این قابلیت برای کاربر محفوظ است که بتواند گروهی از قوانین را به گروهی از اشخاص انتساب دهد
* موجودیت‌های قوانین و تقویم با ساختار وراثتی از موجودیت TAObject در دیتابیس ذخیره شده‌اند
* دسته‌بندی قوانین بصورت سلسله مراتبی ذخیره میگردد
* در یک سلسله مراتب , فرزند میتواند شامل قوانین پدر **باشد** یا **نباشد**
* اینکه آیا یک فرزند در ساختار سلسه مراتبی دسته‌بندی قوانین , شامل قانون که توسط پدر تعریف شده است ,هست یا نه , با فیلد IsContain مشخص میگردد

در این قسمت شمای طراحی شده‌ی جداول با مفروضات بالا درج گردیده است:



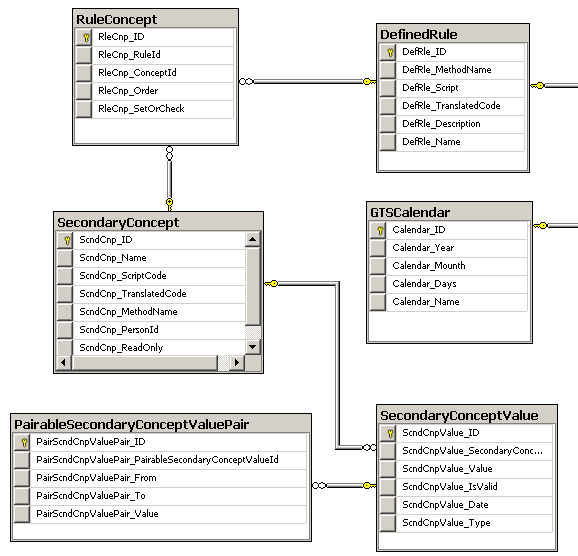
شکل شمای کلی پایگاه‌داده

برای بررسی روشن‌تر این دیاگرام دو قسمت اصلی آن شامل ارتباط **جدول Category با شخص** و ارتباط **جدول Category با اشیاء** به طور جداگانه در دو شکل زیر نمایش داده شده‌اند. لازم به ذکر است جداول یاد شده تنها به منظور ذخیره اطلاعات استفاده می‌گردند و برای واکشی اطلاعات از این جداول(به طور مثال برای یافتن تمام مفاهیم مربوط به یک شخص خاص) از view های جداگانه‌ای که برای هر سه موجودیت قابل انتصاب به شخص(قانون، مفهوم و تقویم) طراحی شده، استفاده می‌شود.



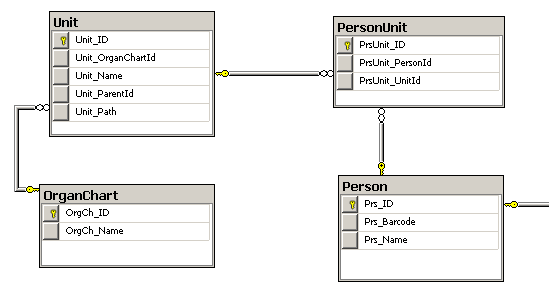
شکل شمای ارتباط جدول Category با شخص

در شکل 3 علاوه بر ارتباط اشیاء(Calendar و DefinedRule) با Category، نحوه‌ی وابستگی SecondaryConcept به DefinedRule (از طریق جدول RuleConcept) نیز لحاظ گردیده است.



شکل شمای ارتباط جدول Category با اشیاء

در شکل 4 نیز جداول مرتبط با نحوه‌ی نگهداری واحد، چارت سازمانی و ... نمایش داده شده است.



شکل شمای نگهداری پرسنل در چارت سازمانی

# طراحی واسط کاربر (UI)

در حال حاضر تنها قسمت مربوط به نمایش و ویرایش قوانین پیاده‌سازی شده است.البته ساختار اصلی واسط کاربر که قصد داریم آنرا در سایر بخش‌ها نیز بکار گیریم , در این بخش رعایت شده است.

ابتدا به معرفی ساختار اصلی وسپس به معرفی قسمتهای این بخش -یعنی ویرایش قوانین- می‌پردازیم.

## ساختار اصلی

## منوی سمت راست

در واقع قرار است قسمت‌های مختلف سیستم از طریق این منوها قابل دسترسی باشند.این منوها بصورت کرکره‌ای دسته‌بندی میشوند.

## زبانه‌های بالایی (Tabs)

فضای عملیاتی کاربر در این زبانه‌ها تعریف خواهد شد به این صورت که هرگاه منوی کناری انتخاب شود به زبانه‌های موجود یک زبانه اضافه میشود و کاربر قسمت‌های مربوطه را مشاهده میکند.هر تب قابلیت بسته شدن نیز دارد

تصویر محیط کاربری را در زیر مشاهده میکنید:



## ویرایشگر قوانین

## جستجوگر قانون

دراین قسمت کاربر قادر به مشاهده دسته بندی قوانین بصورت سلسله مراتبی و قوانین تعریف شده برای هر دسته‌بندی میباشد.مراحل پیمایش شده از درخت قوانین توسط Bread Crumb بالای لیست قابل بازگشت میباشد.در چشم‌انداز این قسمت وجود SearchBox و انتخابگر موجودیت(قوانین یا مفاهیم) دیده شده است.

## نمایشگر قانون با قابلیت ویرایش

پس از انتخاب یک قانون , آن قانون در قسمت پایینی نمایش داده میشود. به منظور جلوگیری از شلوغی و پیچیدگی واسط کاربر , جزئیاتی که شاید کاربر به آنها نیاز نداشته باشد را به بصورت کرکرهای در صورت تمایل کاربر به او نشان میدهیم.

در زیر نمای ویرایشگر آمده است:

