# ساختار محیط کاربری

Version 2.0

|  |  |  |  |
| --- | --- | --- | --- |
| **تاريخ** | **نسخه** | **توضيحات** | **نويسنده** |
| 03/03/1389 | Version 2.0 | تشریح ساختار واسط کاربر | محمد صلواتی |

# 1.مقدمه

یکی از الگوهای طراحی که بمنظور برقراری ارتباط بین لایه واسط کاربر و لایه‌های پایینتر استفاده میشود MVP میباشد.MVP مخفف Model View Presenter میباشد و هدف آن ایجاد ارتباط بین لایه کاربر و دیگر لایه‌ها میباشد بطوری که حداقل وابستگی بین این لایه‌ها وجود داشته باشد.یعنی منطق و روابط business برنامه در UI نقشی نداشته‌باشد.همچنین عکس این رابطه نیز برقرار است.در ادامه مختصری در مورد MVP بحث میکنیم و سپس به شرح نحوه پیاده‌سازی برنامه میپردازیم.

# 2.ساختار MVP:

MVP بطور کلی از سه قسمت تشکیل شده است

* Model:به کلاسهای دامین و لایهای زیرین اعم از DAL و غیره اطلاق میشود.معمولا وظیفه این قست در مجموع بارگزاری و انجام عملیات بر روی دادهای دیتابیس میباشد
* View:شامل اجزای واسط کاربر مانند فرم‌ها , dialog و ... میباشد
* Presenter: نقش یک پل را بین Model و View ایفا میکند.

معمولا هر view یک یا چند presenter دارد که هر دو میدانند که چه عملیاتی را باید انجام دهند.view ,presenter را کاملا میشناسد ولی presenter,view را از طریق یک interface میبیند.بعبارت ساده‌تر هر view یک interface را impalement میکند که شاملproperty های لازم بمنظور binding و دیگر عملیات فرم‌ها میباشد.پس از فراخوانی سرویسهای presenter توسط view, presenter اطلاعات لازم را از طریق این property ها fبه view میدهد.

# 3.مزایای MVP:

* وجود کانال ارتباطی استاندارد بین UI و دیگر لایه‌ها
* امکان عوض کردن تکنولوژی UI با حداقل هزینه(windows base, mobile base , …)
* تست و پشتیبانی راحتر

# 4.سفارشی کردن MVP:

طراحهای MVP یک ساختار کلی برای MVP ارائه داده‌اند و توسعه دهندگان نرم‌افزار بسته به پیچیدکی و حجم پرژه میتوانند آنرا سفارشی کنند.

در پرژه GTS Clock این ساختار بصورت زیر سفارشی شده است:

Forms

1. Forms Layer
2. Presenter Layer
3. Business Layer

Presenter Layer

1. Model Layers

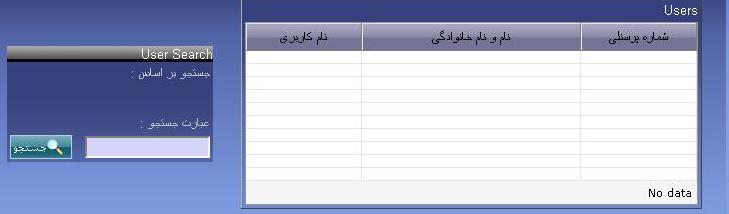
Business Layer

M O D E L L A Y E R

اکنون قصد داریم بشرح GTS Clock بپردازیم:

# 5.سناریو:

فرم زیر را در نظر بگیرید:



کاربر عبارتی را که شامل نام خوانوادگی پرسنل است را جستجو میکند و ما باید نتایج را در جدول نشان دهیم

# 6.مراحل انجام عملیات:

وقتی دکمه جستجو کلیک میشود تابع آن در فرم فراخوانی میشود و سپس این تابع presenter را صدا میزند.پس از انجام یکسری عملیات مانند validation و غیره تابع مورد نظر را در business توسط presenter فراخوانی میشود و business نیز پس از ارتباط با لایه دیتابیس و واکشی اطلاعات و convert اطلاعات در صورت لزوم به کلاسهای proxy آنها را به presenter برمیگرداند. Presenter نیز اطلاعات لازم را یا در خروجی متد و یا به property های form پاس میدهد.فرم هم اطلاعات را bind میکند.