**مستندات بکارگیری WWF .NET در پروژه نرم­افزاری سیستم حضور­ و غیاب شرکت غدیر(نسخه 2)**

* **مقدمه :**

پیرو مباحث مطرح شده در نسخه شماره 1 مستندات مربوط به بکارگیری تکنولوژی WWF در معماری نرم­افزار سیستم حضور و غیاب غدیر، این سری از مجموعه مستندات نیز به نحوه بکارگیری جریان کار در این پروژه اختصاص داده شده است. در این نسخه به شرح و تبیین چگونگی شخصی سازی چارچوب کاری WWF در این پروژه پرداخته شده است.

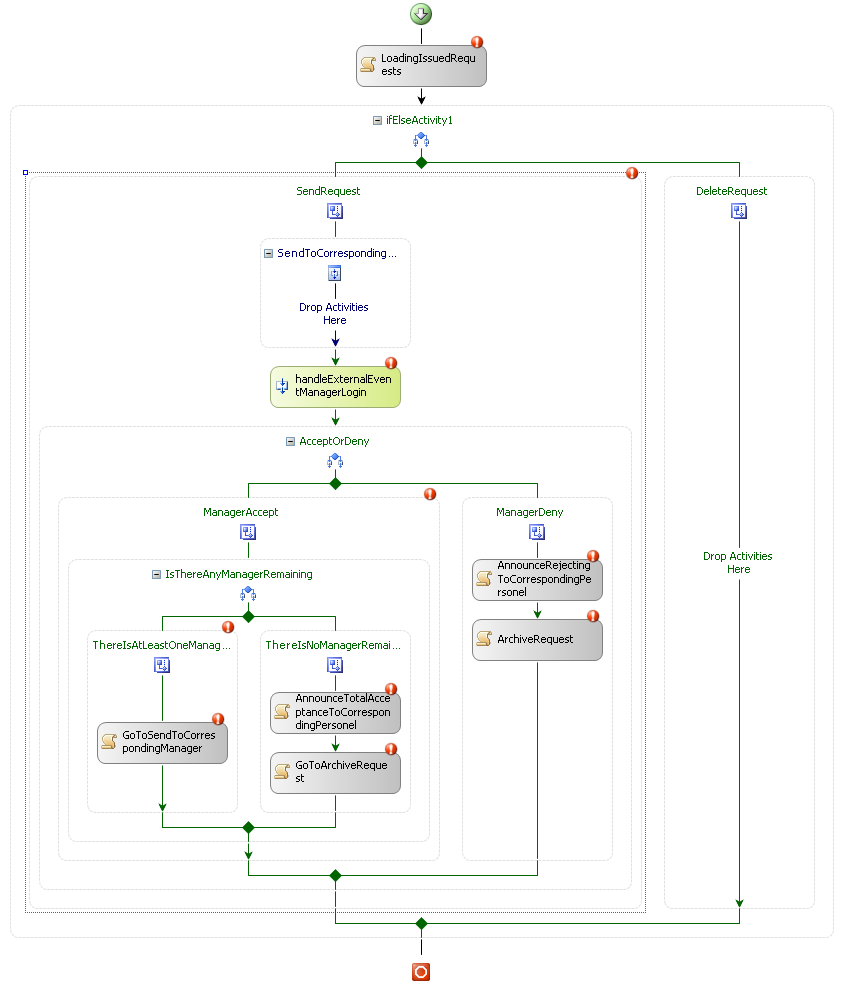
در بخش اول، عمومی­ترین گردش کار سیستم، تعریف شده است. به دلیل اینکه این گردش کار دارای بالاترین سطح انتزاع می­باشد از آن به عنوان گردش کار سطح صفر تعبیر شده است. در بخش دوم، با شرح جزییات بیشتری به تعریف گردش کار سطح یک این سیستم پرداخته شده است. این گردش کار، وظیفه فراهم نمودن مقدمات ارسال درخواست به مدیر را بر عهده دارد. نمودار گردش کار سطح یک، به دلیل گستردگی زیاد در قالب یک مطالعه موردی مورد بررسی قرار گرفته است. نهایتا در بخش سوم، فعالیت­ مهم ارسال درخواست به مدیر که از فعالیت­های موجود در گردش کار سطح یک است، بسط داده شده است.

1. **گردش کار عمومی سیستم (سطح 1) :**

عمومی­ترین گردش کار سیستم، شامل کلی­ترین فعالیت­های تشکیل دهنده جریان کاری خواهد بود. شکل 1 دیاگرام مربوط به این گردش کار را نمایش می­دهد. لازم به ذکر است که در شرایطی که در این نمودار نیاز به بازگشت به یک فعالیت بوده است، از تکرار فعالیت مذکور استفاده شده است. همان طور که در شکل 1 قابل مشاهده است، گردش کار با بارگذاری درخواست­های صادر شده توسط یک کارمند آغاز شده است. منظور از درخواست­های صادر شده، در خواست­هایی است که کارمند پیش­نویس آن­ها را در کارپوشه خود دارد ولی هنوز آن­ها را ارسال ننموده است و به عبارت دیگر هنوز تقاضای تأیید آن­ها را ننموده است. در حالتی که کارمند بخواهد یک یا چند عدد از درخواست­های موجود در کار پوشه خود را برای مدیر مربوطه ارسال نماید، گردش کار وارد فعالیت SendToCorrespondingManager می­گردد. این فعالیت از نوع SequenceActivity است که خود دارای مجموعه­ای از فعالیت­های استاندارد و یا دلخواه (CustomActivity) به عنوان فرزندهای خود می­باشد و در گردش کار سطح 1 بسط داده شده است. فعلا کافی است در مورد این فعالیت بدانیم که این فعالیت اولین مدیر مربوط به در خواست کارمند را یافته و درخواست را به وی ارسال می­نماید. در این مرحله، گردش کار به انتظار log in کردن مدیر مربوطه می­ماند. هنگامی که مدیر مربوطه log in نماید و در خواست کاربر را تأیید و یا رد نماید، گردش کار دوباره به جریان خواهد افتاد. فعالیت handleExternalEventManagerLogin نمایانگر این است که گردش کار منتظر وقوع یک event می­باشد.

در صورتی که درخواست مورد نظر توسط مدیر تأیید شود، بر حسب اینکه آیا مدیر دیگری در سلسله مراتب اداری وجود دارد و یا نه، جریان اجرا وارد دو مسیر جداگانه خواهد شد. در حالتی که مدیر با درخواست موافقت کرده باشد و مدیر دیگری در سلسله مراتب باقی نمانده باشد، توافق مدیر مربوطه با درخواست کارمند، تحت عنوان موافقت نهایی با درخواست، توسط فعالیت AnnounceTotalAcceptanceTOCorrespondingPersonel به کارمند مربوطه اطلاع رسانی خواهد شد و درخواست مورد نظر در سیستم ثبت می­گردد. چنانچه حد اقل یک مدیر در سلسله مراتب اداری باقی مانده باشد، مجددا فعالیت SendToCorrespondingManage اجرا خواهد شد و مراحل مذکور برای مدیر یا مدیران باقی مانده نیز انجام خواهد شد. در صورتی که یکی از مدیران موجود در سلسله مراتب اداری، عدم توافق خود را با درخواست اعلام نماید، به هنگام log in کردن کاربر، رد شدن درخواست، به وی اطلاع رسانی خواهد شد و درخواست در سیستم ثبت و بایگانی خواهد شد.

اگر کارمند یک یا چند تا از درخواست­های صادر شده موجود در کارپوشه خود را حذف نماید، بدون انجام هیچ عملی گردش کار پایان می­یابد، به عبارت دیگر در صورت انتخاب گزینه حذف در UserInterface، انجام هیچ عملی بر عهده گردش کار نخواهد بود. بنابراین به دلیل اینکه عمل delete، صرفا بر عهده UI می­باشد، آن را در گردش کار لحاظ ننمود. ولی با توجه به انتزاعی بودن گردش کار عمومی و همچنین برای شفافیت بیشتر موضوع، فرایند delete نیز در در نظر گرفته شده است.

****

**شکل 1: جریان کار عمومی سیستم**

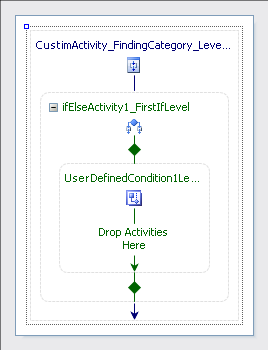
1. **گردش کار مقدمات ارسال درخواست به مدیر(سطح 2):**

در این بخش، به بررسی جزییات فعالیت SendToCorrespondingManager – که به عنوان یک فعالیت انتزاعی در گردش کار عمومی قرار دارد- پرداخته شده است. گردش کار سطح دوم دارای دو وظیفه عمومی خواهد بود:

* اختصاص یک کارمند و یا گروهی از کارمندان جهت شرکت در یک رویه اجرایی خاص
* تصمیم­گیری در مورد روند اجرا با توجه به نوع و مشخصات درخواست

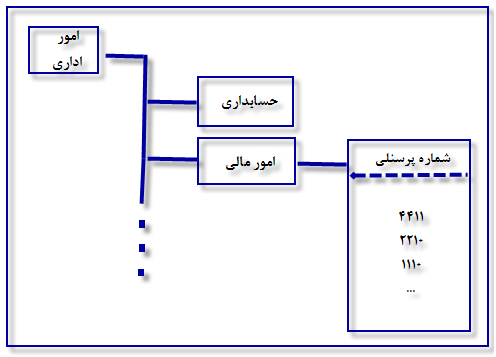
در ادامه به ذکر جزییات بیشتری در مورد این وظایف پرداخته شده است :

جهت پوشش دادن وظیفه اول، طراحی یک CustomActivity کارگشا خواهد بود. شکل 2 ساختار نسبتا ساده این CustomActivity را - که CustomActivity\_FindingCategory\_Level2\_ نام دارد - نشان می­دهد. همان طور که در این شکل قابل رویت است، یک IfElseActivity - با توجه به یک شرط از پیش تعریف شده توسط کاربر (مدیر سیستم) - در مورد اینکه روند اجرایی بعد، در مورد کدام گروه از پرسنل موجود در یک سازمان صدق نماید، تصمیم گیری می­نماید.



**شکل 2: طراحی فعالیت CustomActivity جهت بکارگیری در گردش کار سطح 2**

جهت روشن تر شدن ابعاد این فعالیت به بررسی یک سناریوی فرضی خواهیم پرداخت. به عنوان یک مثال موردی فرض کنید که کاربر (مدیر سیستم) از پیش و از طریق UI، یک سری نقش (Category) تعریف نموده است و هریک از پرسنل را بنا بر صلاحدید خود به یک یا چند نقش از نقش­های مذکور اختصاص داده است. لازم به ذکر است که نقش­ها، به دلخواه مدیر سیستم و مبتنی بر چارت سازمانی می­باشند. شکل 3 مثال ساده شده­ای از نقش­های مذکور را نشان می­دهد.



**شکل 3: نمونه­ای از دسته­بندی کارمندان توسط مدیر سیستم**

هر CustomActivity دارای تعدادی property است که مدیر سیستم در پنل property به آن­ها دسترسی دارد. یکی از property های CustomActivity مذکور شرط دلخواهی است که مدیر سیستم آن را تعریف می­نماید. این شرط در قالب یک Declarative Role Condition تعریف می­گردد و در زمان اجرای برنامه به کد زیر تبدیل می­گردد:

If ( Request.PersonelId In(FinancialSection) ) { … }

در مورد مثال فوق می­توان جهت معنی دار تر شدن و افزایش خوانایی برنامه، به جای نام UserDefinedCondition1Level1، نام IsTheRequestOwnerAMemberOfFinancialSection را به شرط مذکور نسبت داد.

همان طور که پیشتر اشاره شد، از دیگر وظایف مهم گردش کار سطح 2، تصمیم گیری در مورد مورد روند اجرا با توجه به نوع و مشخصات درخواست می­باشد. برای روشن شدن مطلب، در ادامه سناریوی قبلی سناریوی تکمیلی زیر را بررسی می­کنیم:

*" طبق صلاحدید مدیر سیستم (بر طبق قوانین سازمانی)، تمامی درخواست­ها به دو گروه درخواست مرخصی ساعتی، درخواست مرخصی روزانه تقسیم بندی می­شوند. درخواست­های مرخصی ساعتی باید توسط مدیر یا مدیران بخش حسابداری و درخواست­های مرخصی روزانه باید توسط مدیر یا مدیران بخش کارگزینی تایید گردند. مرخصی بیش از 12 ساعت، مرخصی روزانه و مرخصی کمتر از 12 ساعت، مرخصی ساعتی تلقی می­گردد. "*

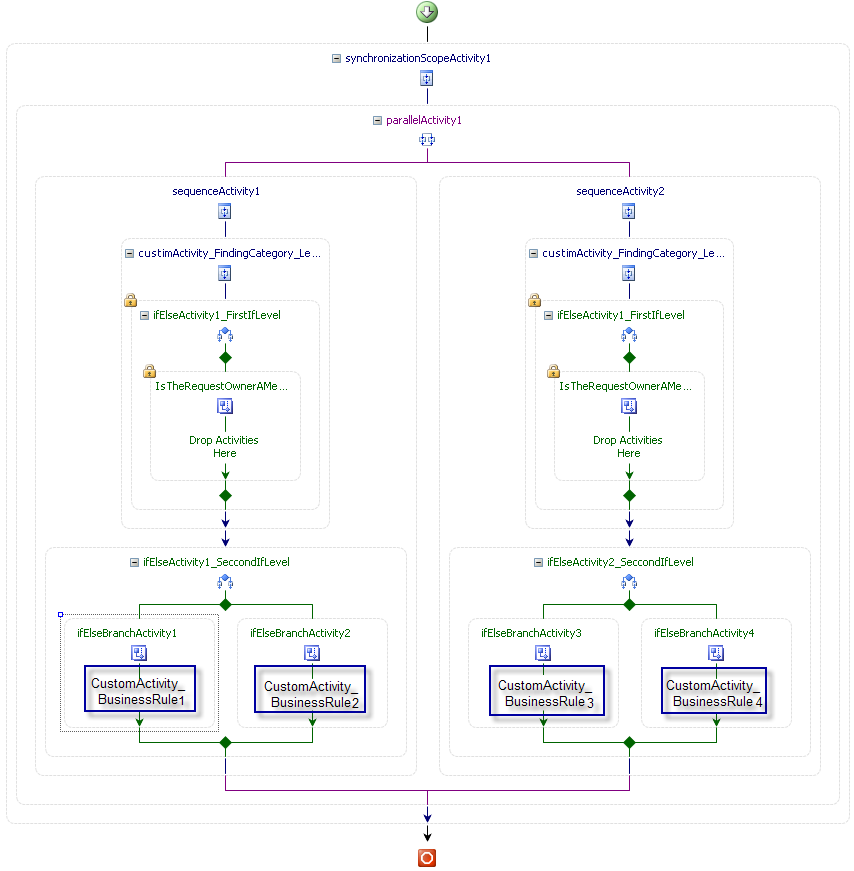
شکل 4، نمای کلی و کامل شده گردش کار سطح 2 و همچنین چگونگی تصمیم گیری در مورد سناریوی فوق را نمایش می­دهد. همان طور که در این شکل دیده می­شود، این تصمیم گیری توسط یک IfElseActivity با نام IfElseActivity1SeccondLevel انجام می­شود. در صورتی که درخواست از نوع مرخصی و ساعتی باشد، IfElseBranchActivity1 ودر صورت روزانه بودن درخواست مرخصی، IfElseBranchActivity2 اجرا خواهد شد. در این مثال، جهت افزایش خوانایی برنامه، شاخه اول IfElseActivity را IsHourlyVecationRequest و شاخه دوم آن را IsDailyVecationRequest می­نامیم. شرط ورود به شاخه اول به صورت زیر خواهد بود:

If ( (Request.Type == Vecation) && (Request.Hours <= 12) ) { … }

شرط ورود به شاخه دوم نیز به صورت زیر خواهد بود:

If ( (Request.Type == Vecation)&& (Request.Hours > 12) ) { … }

در صورتی که نوع درخواست کارمند از نوع مرخصی ساعتی باشد، CustomActivity سمت چپ و در صورت روزانه بودن در خواست CustomActivity سمت چپ اجرا خواهد می­شود که در این مثال، به ترتیب CustomActivty\_BusinessRule1 و CustomActivity\_BusinessRule2 نامیده شده­اند. این فعالیت­ها در دیاگرام گردش کار سطح 3 بسط داده شده­اند که در بخش سوم به ذکر جزییات آن پرداخته خواهد شد. همان طور که خواهیم دید این فعالیت وظیفه ارسال درخواست کاربر به متصدیان نقش­های مختلف را بر عهده دارد. منظور از متصدیان نقش­ها، در اینجا مدیران بخش­های مختلف سازمان است.



**شکل4: دیاگرام عام گردش کار مربوط به ارسال درخواست به مدیر**

1. **گردش کار ارسال درخواست به مدیر مربوطه:**

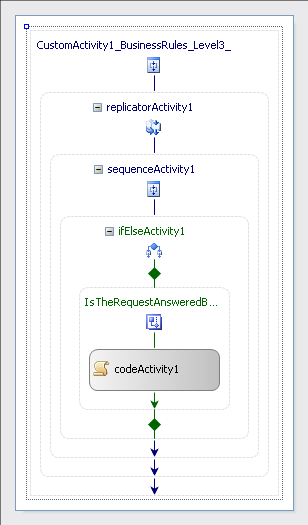
همان طور که پیشتر اشاره شد، در گردش کار 3 به ارسال درخواست پرداخته می­شود. این وظیفه بر عهده یک CustomActivity به نام CustomActivity\_BusinessRule\_ می­باشد. شکل 5 دیاگرام مربوط به این فعالیت را نمایش می­دهد. این فعالیت دارای یک ReplicatorActivity جهت ارسال درخواست به تمام مدیران یا بخش­های موجود در سلسله مراتب اداری است که مدیر سیستم آن­ها را از پیش تعریف می­کند.

با توجه به سناریوی مطرح شده در بخش قبل و اینکه یک درخواست مرخصی روزانه باید به تأیید مدیران کارگزینی برسد، چنانچه فرض کنیم این نوع از درخواست­ها به تایید مدیری با شماره پرسنلی 10011 و مدیری با شماره 20001 نیاز داشته باشند، آنگاه اجزای داخل ReplicatorActivity دو بار ( یک بار برای هر یک از مدیران) اجرا خواهد شد. همان طور که در این شکل دیده می­شود، این اجزا در یک SequenceActivity قرار داده شده­اند.

در این مثال، شرطی که به صورت Declerative Role Condition، در IfElseActivityBranch تعریف شده است، IsTheRequestAnsweredByManager نام دارد. در صورتی که مدیر مربوطه درخواست را پاسخ نداده باشد CodeActivity بعد از IfElseActivity اجرا خواهد شد تا درخواست مربوطه برای وی ارسال گردد. کد این شرط به هنگام اجرای برنامه به صورت زیر خواهد بود:

If ( !Request.Accept.Find(Managers) { … }

لازم به ذکر است که Accept لیستی از تمامی مدیرانی است که درخواست مورد نظر را تایید نموده­اند.



**شکل 5: طراحی CustomActivity ارسال درخواست به مدیر مربوطه**