

فصل ۱

چرخه حیات

۱-۱ مقدمه

یک سوال عمده در ارتباط با متدولوژیها و روشهای تولید نرم افزار تعیین مراحل تولید نرم افزار و مستندات و مدلهایی است که در طی هر مرحله باید ایجاد شود. متاسفانه این مراحل حتی در فرایند آر.یو.پی نیز به طور دقیق مشخص نشده است. برای هر مرحله در نسخه های متفاوت رشنال رز گردش کاری متفاوت از سایرین مشخص شده است. برای نمونه نسخه های رز ۲۰۰۰، ۲۰۰۲ و رشنال ایکس.دی.ای هر کدام نسبتاً فرایند متفاوتی برای آر.یو.پی در نظر گرفته اند. لذا، در این فصل سعی شده تا مراحل به طور منطقی و مبتنی بر واقعیت متدولوژیها مشخص و تجزیه و تحلیل شود. نکته قابل توجه در زمانبندیو میزان تلاش برای مراحل مختلف است که پیاده سازی را هدف و بخش عمده کار میداند.

۲-۱ چرخه حیات تکرار شونده

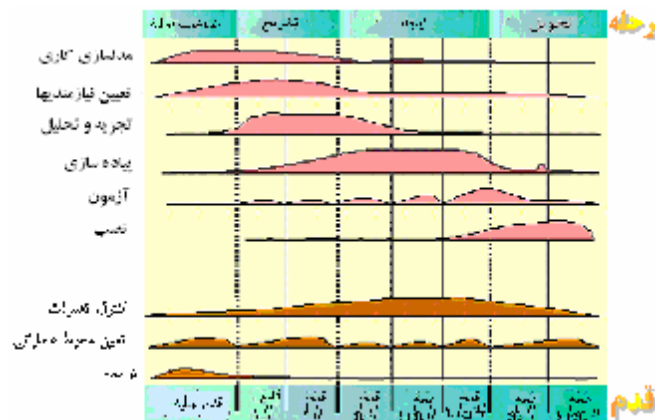
در چرخه حیات آبشاری مراحل ایجاد نرم افزار یکی پس از دیگری به انجام می رسند. در واقع یک توالی بدون برگشت در بین مراحل فرایند تولید نرم افزار وجود داشت.



شکل ۱- چرخه حیات آبشاری

روش فوق به علت جمع شدن اطلاعات و طولانی شدن زمان مشاهده نتایج، منسوخ است. امروزه سعی می کنند در روشهای تکراری کلیه مراحل فوق تکرار شود. و در هر تکرار تاکید بر یکی از مراحل چرخه حیات آبشاری است. در شکل ۲ فرایند تولید نرم افزار در آر.یو.پی مشخص شده است. همانگونه که مشاهده می کنید فرایند تولید یا به عبارت دیگر چرخه حیات نرم افزار از چهار مرحله

اصلی شناخت، تشریح، ایجاد و تحویل تشکیل شده است. در مرحله شناخت به طور عمده بر روی شناخت نیازها تاکید شده است و در مرحله ایجاد، بطور عمده تاکید بر روی پیاده سازی و ایجاد کد برنامه ها است.



شکل ۲- چرخه حیات

همانگونه که در شکل فوق مشخص است، هر مرحله از چرخه حیات با تعیین نیاز کاری آغاز و با پیاده سازی، آزمون و نهایتاً نصب نرم افزار خاتمه می یابد. هر مرحله از تعدادی تکرار یا قدم تشکیل شده است. تعداد قدمهای یک تکرار وابسته به پیچیدگی پروژه متغیر است. نسبت تلاش و زمان لازم برای انجام هر مرحله در شکل ۳ مشخص شده است. همانگونه که در شکل ۳ مشخص شده، عمده وقت و تلاش، صرف ایجاد برنامه ها می شود.

	شناخت اولیه	تشریح	ایجاد	تحویل
% تلاش	۵	۲۰	۶۵	۱۰
% زمانبندی	۱۰	۳۰	۵۰	۱۰

شکل ۳- درصد تلاش و زمان لازم برای انجام مراحل آ.ر.یو.پی

بر طبق جدول فوق تاکید اصلی در چرخه حیات آ.ر.یو.پی بر مرحله پیاده سازی است. هدف ایجاد برنامه است. لذا، از ابتدا باید این هدف را مد نظر داشت. پیاده سازی در قالب نمونه های سریع در دو مرحله شناخت اولیه و تشریح به ترتیب جهت درک بیشتر نیازها و ارائه نمونه ای ظاهری از نرم افزار مطرح است.

۱-۲-۱ مرحله شناخت اولیه

در این بخش ابتدا هدف از مرحله شناخت بیان شده است. سپس، مشخص شده که چه مستنداتی در این مرحله باید تهیه شود و بالاخره چگونگی تهیه مستندات مشخص شده است. هدف اصلی در مرحله شناخت تعیین نیازمندیها و اهداف پروژه است. در این راستا باید حوزه عملکرد پروژه و سازماندهی و عملکرد سیستم جاری را مشخص نمود. بر اساس نیازها و اهداف است که می توان قابلیتها و دامنه عملکرد محصول نرم افزاری را مشخص کرد. در این راستا مستندات تحت عنوان چشم انداز^۱ پروژه تهیه می شود. بر اساس چشم انداز پروژه هزینه و زمانبندی در قالب برنامه ریزی پروژه مشخص می شود. برای ادامه مراحل پروژه باید اولویتها برای طرح و پیاده سازی موردهای استفاده از سیستم مشخص و منابع مورد نیاز تعیین شوند. به طور خلاصه خروجیهای این مرحله عبارتند از:

مستنداتی تحت عنوان چشم انداز یا دیدگاه های پروژه که بطور کلی شامل توصیف نیازها، نکات کلیدی و محدودیتهای پروژه است.

لیستی از موردهای استفاده که می توان در این مراحل اولیه مورد استفاده قرار داد

دیکشنری اصطلاحات

یک قالب کاری شامل

زمینه کاری

معیارهای موفقیت از قبیل درآمد، وضعیت در بازار

پیش بینی مالی

تخمین اولیه از میزان ریسک

برنامه ریزی پروژه شامل برنامه ریزی مراحل و تکرارها

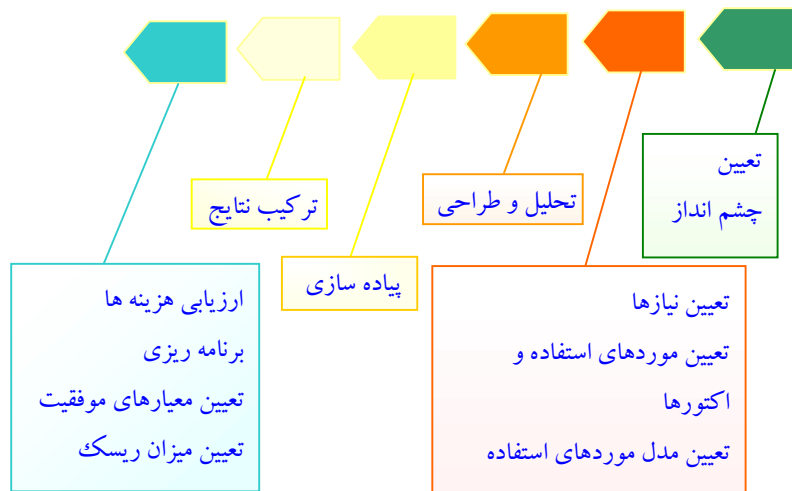
موارد فوق در ادامه مورد بررسی قرار خواهد گرفت. برای ارزیابی نتایج حاصل از این مرحله می بایست مشخص نمود که اولاً، همه کاربرها در مورد هزینه ها و زمانبندی پروژه موافقت دارند. همچنین باید مشخص نمود که آیا موردهای استفاده در برگیرنده نیازهای تعیین شده هستند. باید اطمینان حاصل نمود که تخمین هزینه ها، زمانبندیها و میزان ریسک، قابل قبول است.

۱. مستندات چشم انداز

مستندات چشم انداز بیانگر نیازها و قابلیت‌های سیستم مکانیزه در پاسخ به نیازها است. نکته اصلی چگونگی تعیین نیازها است. در ابتدای مرحله شناخت مستندات چشم انداز تهیه می شود. بر مبنای چشم انداز پروژه مستندات قالب کاری پروژه تهیه می شود. مستند چشم انداز پروژه در طول چرخه حیات تصحیح و تکمیل می شود.

در طی مرحله شناخت اولیه در واقع قالب کاری و اهداف پروژه مشخص می شود. قالب کاری ابزاری برای سنجش میزان سوددهی پروژه است. در واقع بدینوسیله اقتصادی بودن و ادامه کار پروژه ارزیابی می شود. در ارتباط با قراردادها از طریق مزایده، تقاضای پروژه، پیشنهاد ارائه شده و بالاخره قرارداد منعقد شده در مجموع قالب پروژه را مشخص می کنند. یک قالب پروژه شامل موارد ذیل است:

- مقدمه: خلاصه ای از کل مستند
- هدف: هدف از این مستند
- دامنه :
- توصیف محصول :
- زمینه کاری :
- اهداف محصول :
- پیش بینی مالی : میزان سوددهی و بازدهی پروژه در ارتباط با نسبت درآمد حاصل از فروش محصول به میزان سرمایه مصرفی برای انجام پروژه مشخص می کند. همچنین برای نمونه میزان بازدهی را نسبت به مدت زمان انجام پروژه تعیین می کنند. برای نمونه اگر پروژه یک ساله انجام شود، میزان بازدهی ۵ و دو ساله ۲ و برای سه سال بازدهی منفی خواهد بود. این تقریبها در پایان هرمرحله به روز می شود.
- محدودیتها :
- گزینه ها : شامل بررسی ایجاد امکانات مختلف و هزینه های مربوطه است. به طور خلاصه می توان گفت عملیات در فاز شناخت اولیه به شرح زیر می باشند:



شکل ۱- عملیات مراحل شناخت اولیه

۳-۱ تعیین نیازها

برای تعیین نیازها ابتدا باید مدل کاری برای سیستم جاری ایجاد شود. در واقع این مدل کاری ابزاری جهت بازکردن سیستم و شناخت نیازمندیها است. بر مبنای این مدل کاری نیازمندیها در قالب مستندی تحت عنوان ویزن یا چشم انداز پروژه مشخص می شود.

برنامه ریزی قدمها

همانگونه که در بالا نیز توضیح داده شد، هر مرحله از چرخه حیات آر.یو.پی تکراری از مراحل چرخه حیات آبخاری است. بنا براین هر تکرار در واقع شامل کلیه مراحل چرخه حیات آبخاری است. البته برای مراحل متفاوت در فرایند آر.یو.پی، طول مدت زمان هر مرحله از چرخه حیات آبخاری متفاوت است. برای نمونه در تکرارهای مربوط به مرحله تشریح تاکید بیشتری بر روی مرحله تحلیل از چرخه حیات آبخاری است. در ادامه برنامه ریزی تکرار اول از مرحله شناخت اولیه مشخص شده است:

مدیریت پروژه

بررسی پروژه جدید

تعیین دامنه و ریسک انجام پروژه

برنامه ریزی
برنامه ریزی مابقی تکرار اول
مدیریت تکرار
سرپرستی و کنترل پروژه
ارزیابی مجدد پروژه
برنامه ریزی برای تکرار بعدی
اصلاح برنامه ریزی پروژه
ارزیابی وضعیت کاری
تعیین فرایندهای کاری
اصلاح فرایندهای کاری
طرح تحقق فرایندهای کاری
اصلاح مسئولیتها و نقشها
بررسی برای مکانیزه کردن فرایند کاری

نیازها

تحلیل مساله
درک نیازها
تعریف سیستم
تعیین دامنه سیستم
اصلاح تعریف سیستم
مدیریت تغییر نیازها
آنلیز و طراحی
تعیین ساختار
آزمون

فصل ۲

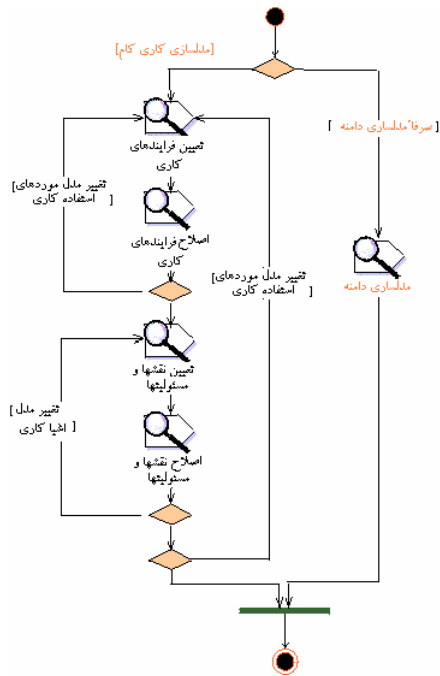
مراحل انجام مدل‌سازی سیستم کاری

۲-۱ مقدمه

مرحله شناخت نیازمندیها با شناخت سیستم جاری آغاز می‌شود. یکی از اهداف این مرحله از شناخت، تعیین نیازها با مطالعه عملکرد سیستم کاری یا سیستم جاری است. لذا، حاصل شناخت می‌بایست در مراحل بعدی مورد تجزیه و تحلیل و نهایتاً به پیاده‌سازی خاتمه یابد. مساله، ارتباط بین مدل‌های حاصل از شناخت و مدل‌های حاصل از مرحله آنالیز و طراحی است.

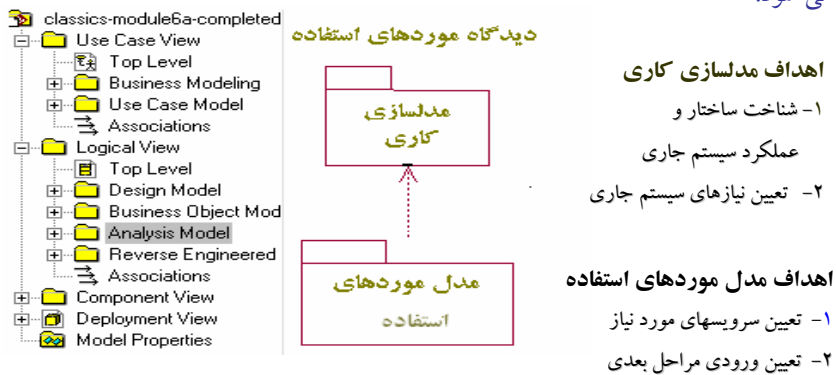
۲-۲ مدل‌سازی سیستم کاری

مدل‌سازی سیستم موجود یا در اصطلاح سیستم کاری ابزاری برای شناخت نیازمندیها است. روش مبتنی بر تجزیه عملکرد سیستم جاری است. لذا، می‌بایست در ابتدا نمودار عملیاتی سیستم جاری را مشخص نمود. برای این منظور ابتدا چارت سازمانی و با ارجاع به شرح وظایف چارت عملیاتی مشخص می‌شود. در واقع با تجزیه به اجزا کار شناخت تسهیل می‌گردد. برای نمونه هدف از مرحله مدل‌سازی کاری تجزیه سیستم جاری جهت شناخت و تحلیل نیازمندیها است. با ارائه مدلی از سیستم فعلی و تایید صحت مدل، کارفرما و تولیدکنندگان سیستم درک یکسانی از سیستم موجود خواهند داشت. شکل ۲-۱ روند کلی این مرحله را نشان می‌دهد. در طی این مرحله فعالیتهای مختلفی انجام می‌گردد که مجموعه این فعالیتها در شکل ۲ نشان داده شده است.



شکل ۱- مراحل کلی روند مدلسازی کاری

با توجه به مراحل کلی ارائه شده برای مرحله مدلسازی کاری، برای هر سیستمی در بالاترین سطح بسته های زیر در بخش دیدگاه های مورد استفاده و دیدگاه منطقی در نظر گرفته می شود.
 ۱- در دیدگاه موردهای استفاده، گزینه ای برای موردهای استفاده کاری مطابق شکل زیر قرار داده می شود:



شکل ۲- اولین دیاگرام در دیدگاه موردهای استفاده

۲- در دیدگاه منطقی مدل اشیاء کاری در جهت انجام و تحقق موردهای استفاده کاری مشخص می شود.

ssics-module6a-completed
 Use Case View
 Top Level
 Business Modeling
 Use Case Model
 Associations
 Logical View
 Top Level
 Design Model
 Business Object Mod
 Analysis Model
 Reverse Engineered
 Associations
 Component View
 Deployment View
 Model Properties

مدل اشیاء کاری

مدل اشیاء کاری نمایانگر چگونگی تحقق (Realization) موردهای استفاده کاری است. در واقع خلاصه ای از عملکرد کارکنها بر روی مدارک و مستندات جهت انجام امور است.

مدل آنالیز

مدل آنالیز شامل کلاسها و روابط بین آنها در جهت تحقق موردهای استفاده و خلاصه ای از مدل طراحی است.

مدل طراحی

شکل ۳- اولین دیاگرام در دیدگاه منطقی

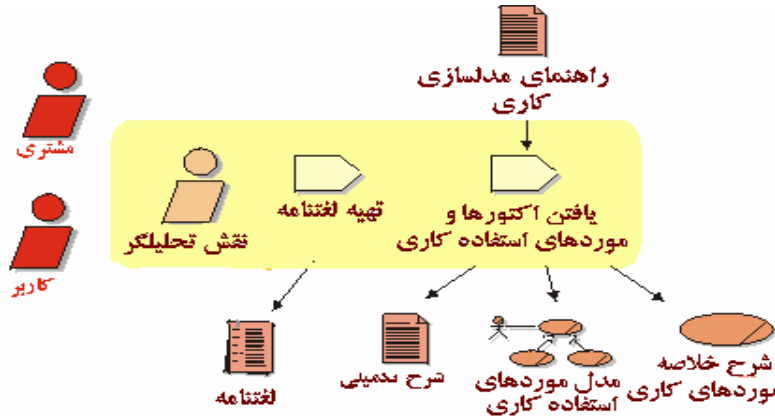
مراحل انجام مدلسازی کاری شامل دو مرحله ایجاد مدل‌های استفاده و مدل اشیاء است. مرحله اول با تهیه لغت نامه برای توصیف انواع اصطلاحهای مورد استفاده در سیستم کنونی آغاز می‌شود. پس از آن اکتورها و موردهای استفاده آنها از سیستم مشخص می‌شود و نهایتاً توسط تحلیلگر سیستم مدل موردهای استفاده کاری ایجاد می‌شود. در مرحله دوم بر مبنای توصیف موردهای استفاده کاری، کارکنها و موجودیتهای کاری مشخص می‌شوند. منظور از موجودیت کاری کلیه مستنداتی است که جهت انجام امور، کارکنها به آنها ارجاع و یا آنها را ایجاد می‌کنند. مراحل انجام کار در شکل ۲ مشخص شده است.



شکل ۳- فعالیتهای مدلسازی کاری در دو مرحله تعیین موردهای استفاده و مدل اشیاء کاری در ادامه به ذکر فعالیتهای مشخص شده در شکل فوق پرداخته خواهد شد.

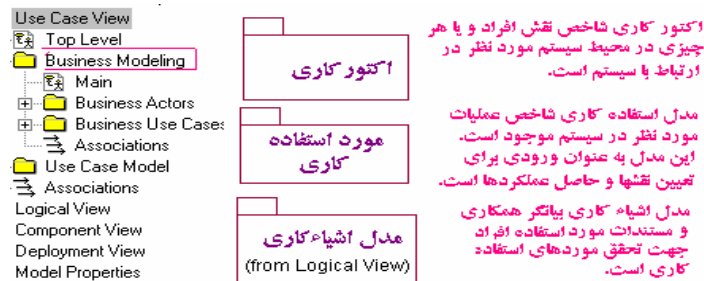
۲-۳ فرایند کاری

جهت شناخت عملکرد سیستم جاری می بایست استفاد کننده ها و مورد های استفاده آنها از سیستم جاری مشخص شود. در این راستا تیم مهندسی نرم افزار در صورت نیاز از افرادی که در امور جاری سیستم مورد نظر صاحب نظر هستند استفاده می کند تا به کمک آنها اطلاعات صحیح را در مورد سیستم جاری گردآوری نماید.



شکل ۴- عملیات مرحله تعیین مورد های استفاده کاری

با در نظر گرفتن فرایند عملیاتی مشخص شده در شکل ۴، پس از تعیین لغتنامه می بایست بازیگرها یا اکتورهای کاری را مشخص نمود و سپس مورد های استفاده آنها از سیستم را مشخص نمود. برای مدلساز این مرحله می بایست بسته مدلسازی کاری را که در شکل ۲ مشخص شده به صورت زیر توسعه داد.



شکل ۵- توسعه بسته مدلسازی کاری

۲-۳-۱ تعیین لغتنامه

لغتنامه در برگیرنده کلیه واژه ها و اصطلاحات مورد استفاده در سیستم است. برای هر سیستم تنها یک لغتنامه ایجاد می شود. لغتنامه در واقع مرجعی برای تیم مهندسی نرم افزار در جهت درک

مستندات تهیه شده و یا موجود برای سیستم مورد بررسی است. در ادامه یک لغتنامه برای سیستم مکاتبات اداری ارائه شده است:

واژه نامه

۱. **مقدمه:** هدف از این سند جمع آوری، تعریف و نمایش واژه های تجاری سطح بالای سیستم نامه رسانی اداری است که بر روی تعریف اجزاء و موجودیتهای تجاری موجود در سیستم، کاربران و افراد درگیر با سیستم تکیه دارد.
- ۲- **هدف:** جمع آوری، تعریف و نمایش واژه های مورد استفاده در سیستم نامه رسانی اداری.
- ۳- **حوزه:** این سند واژه نامه درگیر با یک پروژه نرم افزار Rational RequistPro جهت جمع آوری، تعریف و مدیریت نیازمندیها و خصوصیات سیستم و یک فایل مدل نرم افزار Rational Rose جهت مستندسازی، مشخص سازی و بصری سازی سیستم نامه رسانی اداری و کلا مدل سازی رفتار سیستم فعلی، تعریف نیازمندیها و تهیه مدل تحلیل و طراحی مدل نهایی جهت پیاده سازی و استفاده از نرم افزار Rational SoDA for WORD جهت تیه گزارشات نهایی است.
- ۴- **مراجع:** بجز استفاده از منابع موجود در سایت www.rational.com آنهم جهت شناخت فرآیند RUP و استفاده از آن در شناخت سیستم موجود و تهیه سیستم کامپیوتری جدید، مرجع یا منبع دیگری جهت ذکر وجود ندارد.
- ۵- **مور:** در ادامه این سند به معرفی تعاریف جمع آوری شده در مرحله مدلسازی تجاری میپردازیم.

تعاریف

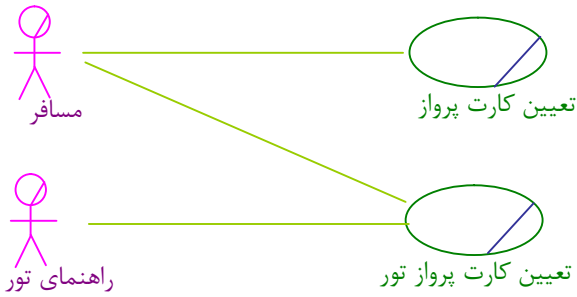
- نامه داخلی:** نامه ای است که بین کارکنان داخل دانشگاه (اداره) شامل کارمندان، اعضاء هیئت علمی، مدیران و یا ریاست از طریق قفسه مبادله می شود.
- نامه خارجی:** نامه ای است که گیرنده یا فرستنده آن در خارج از دانشگاه (اداره) قرار دارد.
- دبیرخانه:** فرستنده: هر کاربری که اقدام به ارسال نامه (داخلی یا خارجی) می کند.
- گیرنده:** هر کاربری که مقصد یک نامه ارسالی است و اقدام به دریافت نامه (داخلی یا خارجی) می کند.
- نامه رسان دبیرخانه:** شخص حمل کننده نامه ها از دانشگاه (اداره) به دبیرخانه و بالعکس.
- ریاست:** رئیس دانشگاه (اداره) که روال خاصی برای مبادله نامه هایش دارد.
- محل تبادل نامه های خارجی:** است

شکل ۶- نمونه ای از واژه نامه

۲-۳-۲ تعیین موردهای استفاده کاری

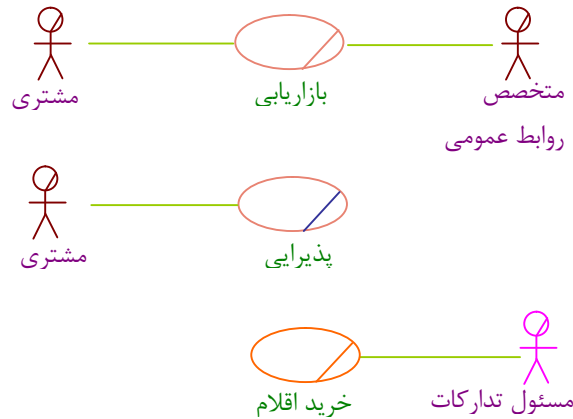
موردهای استفاده کاری شاخص عملکرد یک سازمان یا مجموعه کاری بوده نشان می دهد که چه وظایفی باید انجام شود تا نتایج مطلوب ایجاد شود. استفاده عمده از موردهای استفاده کاری تعیین چگونگی استفاده افراد خارج از سیستم کاری را از آن مشخص می کنند. موردهای استفاده کاری را

فرایندهای کاری نیز می‌نامند. برای نمونه در شکل زیر برای دریافت کارت سوار شدن به هواپیما مسافر از مورد تعیین کارت پرواز استفاده می‌کند.



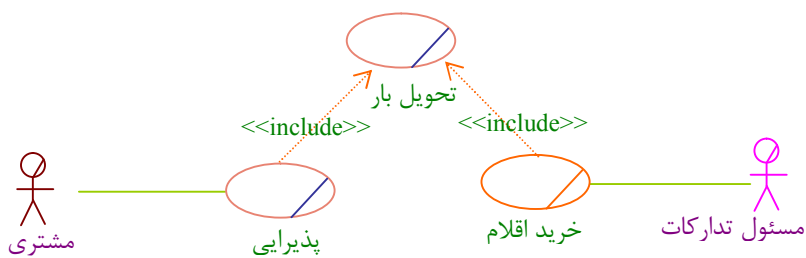
شکل ۷- اکتورها و موردهای استفاده برای بازرسی در ایستگاه مرزی

هنگامی که فعالیتهای یک سیستم کاری مورد بررسی قرار می‌گیرند، باید توجه نمود که یک سری از فعالیتهای داخلی مهم سیستم هستند که از لحاظ سرویس دهی به خارج از سیستم مهم نیستند. برای نمونه عملیات مربوط به نیروی انسانی، خدمات و امنیت، سرویسی به بیرون نمی‌دهند اما ضروری هستند. امور مدیریتی نیز موثر در سایر موردهای استفاده هستند. برای نمونه در مورد یک رستوران موردهای استفاده اصلی در ارتباط با بازار یابی، پذیرایی، و تدارکات است.



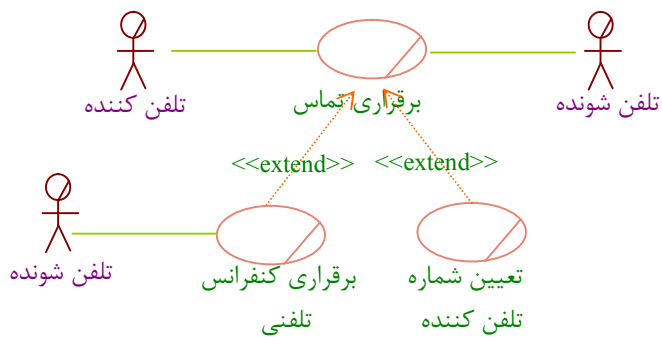
شکل ۸- موردهای استفاده اصلی در سیستم رستوران

رفتار مشترک بین موردهای استفاده در قالب رابطه <<Include>> مشخص می‌شود.



شکل ۹- مورد های استفاده مشترک

چنانچه اجرای بخشی از مورد استفاده اختیاری، وابسته به شرایط و یا وابسته به عملکرد اکتور باشد، آن بخش را در قالب مورد استفاده توسعه دهنده مشخص می کنند. برای نمونه در حالت عادی سیستم تلفن امکان برقراری ارتباط تلفنی را باید ایجاد کند در موارد خاص ممکن است تلفن شونده تقاضای مشخصات تلفن کننده را داشته باشد و یا بخواهد امکان حضور بیش از دو نفر در یک مکالمه تلفنی فراهم شود.



شکل ۱۰- مورد های استفاده مشترک

رفتار احتمالی و یا مشروط مورد های استفاده را می توان در قالب یک مورد توسعه دهنده از آنها مجزا نمود. برای نمونه در هنگام صدور کارت پرواز ممکن است که برای بار خاص مسافر عملیات ویژه انجام شود.

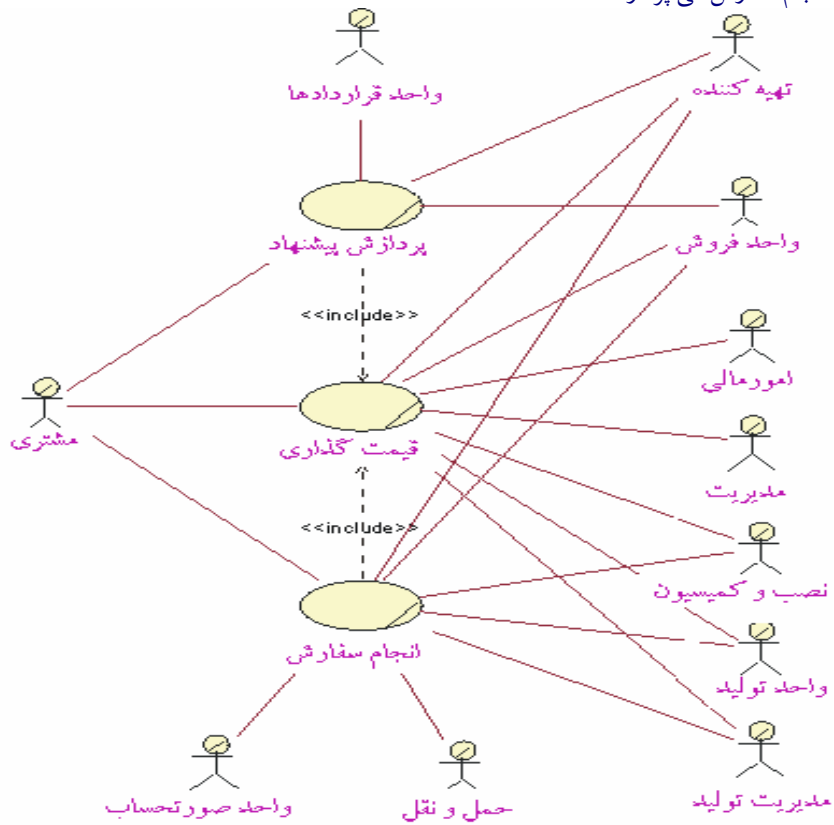


شکل ۱۱- توسعه عملکرد مورد استفاده

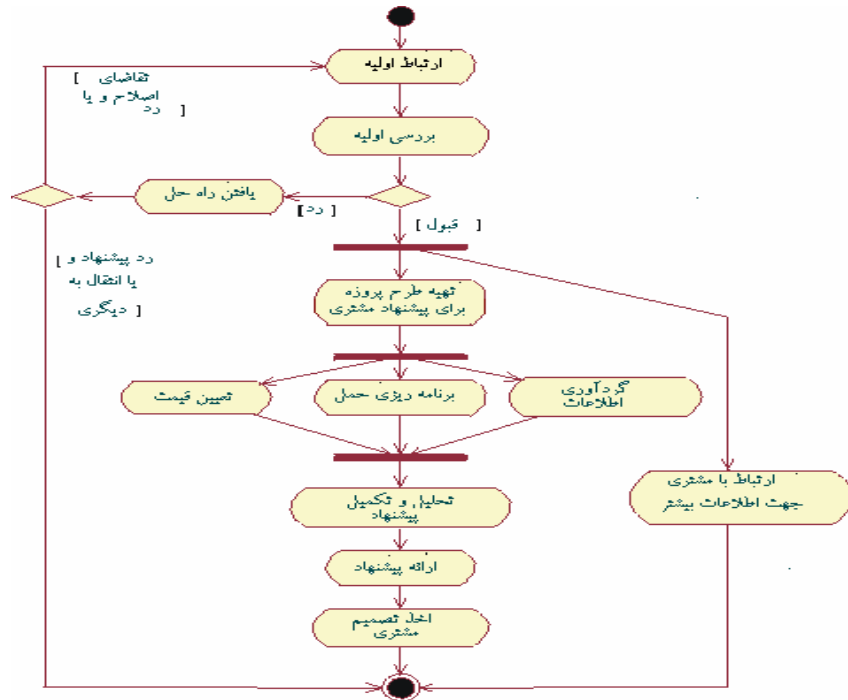
رفتار مشترک موردهای استفاده را می توان در قالب جنس موردهای استفاده مشخص نمود. برای نمونه در شکل ذیل مورد استفاده سفارش تلفنی و مورد استفاده سفارش اینترنتی هر دو از جنس سفارش کالا مشخص شده اند. به این نوع رابطه، رابطه وراثت گفته می شود:



باید توجه داشته باشید که هنگام تعیین موردهای استفاده از یک زیر سیستم، سایر زیر سیستمها به عنوان بازیگر یا اکتور برای آن زیر سیستم مطرح هستند. برای نمونه در شکل ذیل سفارش مشتری در قالب یک پیشنهاد به واحد سفارشات داده می شود و این واحد پس از بررسی پیشنهاد و قیمت گذاری به انجام سفارش می پردازد.



معمولاً پس از تعیین دنباله ای از موردهای استفاده باید آنها را در مجموع مورد بررسی قرار داد و مشخص نمود که توالی زمانی درستی داشته باشند. همچنین مجموعه موردهای استفاده باید شاخص عملکرد سیستم و اهداف آن نیز باشد. برای بیان مفهوم موردهای استفاده می توان به صورت دنباله ای از رویدادها عملکرد مورد استفاده را مشخص نمود. یا اینکه با استفاده از دیاگرام فعالیت، عملکرد را در قالبی مشابه فلوچارت مشخص نمود. دیاگرامهای فعالیت در فصلهای بعدی توضیح داده خواهند شد. پس از تعیین موردهای استفاده کاری باید در صورت پیچیده بودن عملکرد، آنها را بر حسب اولویت با استفاده از دیاگرامهای فعالیت توصیف نمود.



شکل ۱۴- دیاگرام فعالیت برای توصیف مورد سفارش کالا

موردهای استفاده کاری را می توان با توصیف گام به گام مورد استفاده نیز مشخص نمود. برای نمونه در مورد بازرسی مسافر که در شکل ۱ مشخص شد به صورت ذیل مورد استفاده مشخص می شود:

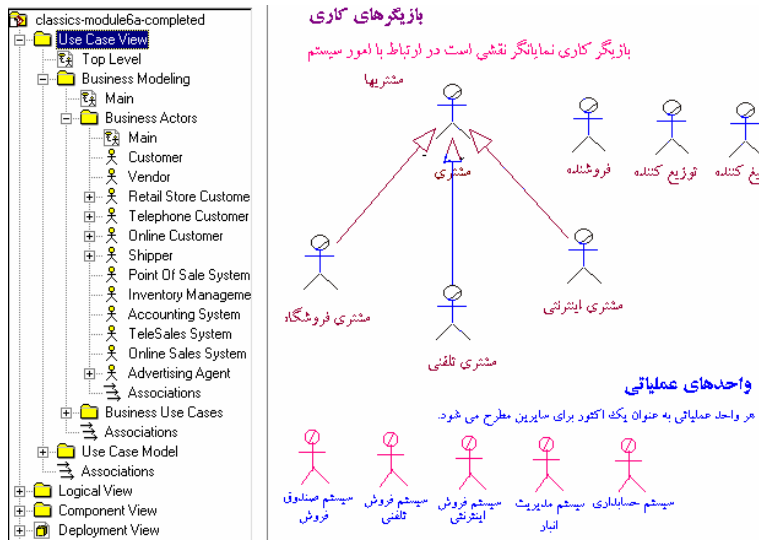
مسافر وارد صف انتظار برای دریافت کارت پرواز می شود
 مسافر بلیط خود را به کارمند شرکت هواپیمایی می دهد
 کارمند بلیط مسافر را بررسی می کند

کارمند بار مسافر را ثبت می کند
 کارت پرواز صادر می شود
 کارمند کارت پرواز را به مسافر می دهد
 مسافر از بانه کارت پرواز دور می شود.

برای هر مورد استفاده می توان نیازهای مرتبط با آن مورد استفاده را در شرح آن گنجانده. در توصیف مورد های استفاده همچنین می توان پیش شرطها و پس شرط برای اجرا مورد را مشخص نمود.

۲-۳-۳ ایجاد مدل بازیگرها و مورد های استفاده آنها

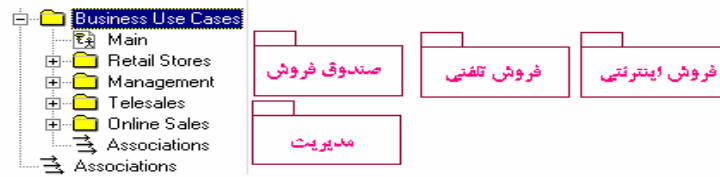
در شکل ۵ مشخص شد که در دیدگاه مورد های استفاده برای مدل های کاری می بایست اکتورها و مورد های استفاده مشخص شوند. در شکل ۱۵، بسته اکتور های کاری مشخص شده است. در این بسته هر واحد سازمانی نیز به عنوان یک بازیگر یا اکتور برای سایر واحدها در نظر گرفته شده است. در واقع ارتباط و تعامل زیر سیستمها با یکدیگر از طریق این بازیگرها مشخص می شود.



شکل ۱۵- مدل سازی بازیگرها

برای بسته مورد های استفاده کاری که در شکل ۵ مشخص شد نیز می بایست دیاگرام های مورد استفاده مشخص شود. باید توجه نمود که هدف از مورد های استفاده کاری تعیین عملیات مورد نظر

سیستم کاری است. هدف عمده از تعیین بازیگره و موردهای استفاده کاری نمایش این واقعیت است که چگونه سیستم کاری توسط مشتریها و سایر واحدها مورد استفاده قرار می گیرد. در ادامه هر واحد سازمانی به عنوان یک بسته مشخص شده است.

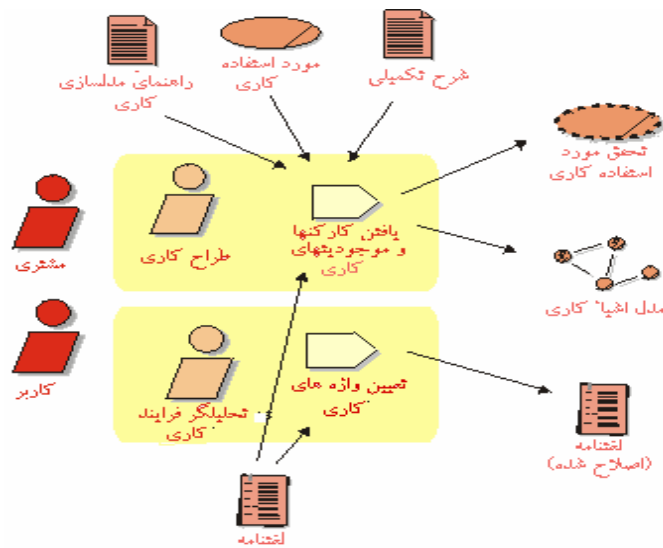


شکل ۱۶- هر واحد سازمانی به عنوان یک بسته مشخص می شود.

باید توجه نمود که هر بسته ممکن است توسط بسته دیگر و یا یک دیاگرام مورد استفاده توسعه داده شود.

۲-۴ مدل اشیاء کاری

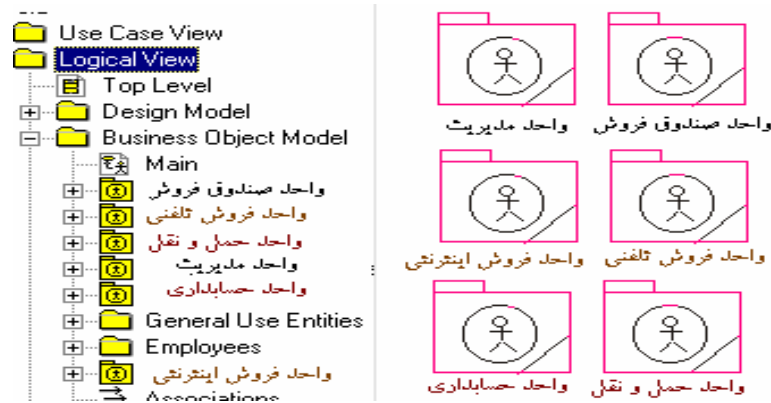
جهت ارائه هر گونه مورد استفاده کاری می بایست افراد با یکدیگر همکاری نمایند. هر فرد در قالب یک کارکن عملیات خود را با ایجاد و ارجاع به مستندات و پرونده های موجود انجام می دهد. هر نوع محل نگهداری و بازیابی اطلاعات را در اصطلاح یک موجودیت کاری می نامند. در شکل زیر فعالیتهای مرحله مدلسازی اشیاء کاری مشخص شده است. همانگونه که در شکل مشخص شده است می بایست ارتباط بین مدل اشیاء کاری و تحقق موردهای استفاده مشخص شود.



شکل ۱۷- فعالیتهای مرحله مدلسازی اشیاء کاری

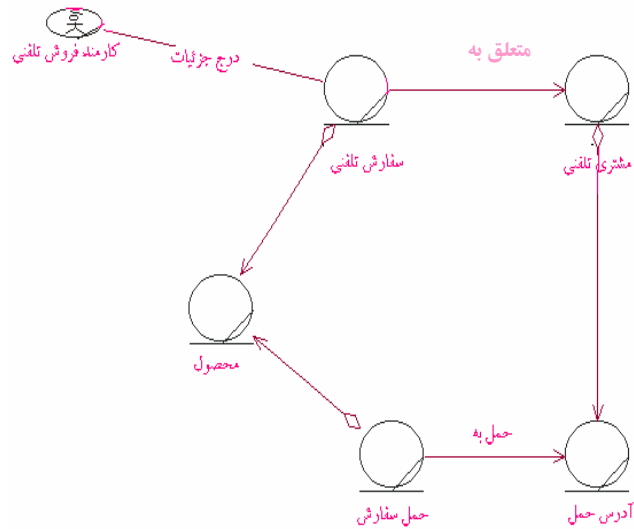
۲-۴-۱ تعیین واحدهای سازمانی

پس از تعیین فرایندهای کاری در مرحله دوم از مدلسازی کاری باید نقشها و مسئولیتها را مشخص نمود. برای این منظور باید ابتدا واحدهای سازمانی را مشخص کرد. مشخص کنید که کدام واحد سازمانی می بایست در عملیات سیستم دخیل باشد. در این راستا چارت عملیاتی می تواند راهگشا باشد. برای مدلسازی واحدهای سازمانی، بسته مدل اشیاء کاری در شکل ۳ به صورت زیر گسترش داده می شود.



شکل ۱۸- واحدهای عملیاتی

واحدهای سازمانی در واقع مسئولیت سرویس دهی را بر عهده دارند. ممکن است بخشی از عملکرد یک مورد استفاده توسط یک واحد سازمانی به انجام رسد. برای این منظور باید افراد در داخل واحدهای سازمانی با یکدیگر همکاری کنند. برای مشخص نمودن امکانات واحدهای سازمانی از مدل‌های اشیاء کاری استفاده می شود. مدل اشیاء کاری بیانگر رابطه بین کارکنها و موجودیتهای کاری است. منظور از موجودیتهای کاری هر نوع مکانی برای ذخیره و بازیابی اطلاعات است که توسط کارکنها مورد استفاده قرار می گیرد. برای نمونه در ادامه عملکرد واحد سازمانی فروش تلفنی که در شکل قبلی مشخص شده بود مشخص شده است.

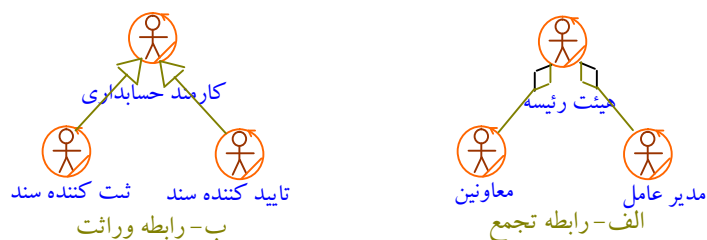


شکل ۱۹ - مدال ارتباطی اشیاء

در شکل فوق، کارمند فروش تلفنی در نقش یک کارکن جزئیات سفارش تلفنی را در موجودیت سفارش تلفنی درج می‌کند. هر سفارش تلفنی متعلق به یک مشتری تلفنی است. مشتری تلفنی دارای یک یا چند آدرس حمل است. علامت لوزی شاخص رابطه تجمع است. این رابطه را به صورت "بخشی است از" نیز می‌توان مشخص نمود. در شکل فوق برای نمونه: "آدرس حمل بخشی از مشتری تلفنی است".

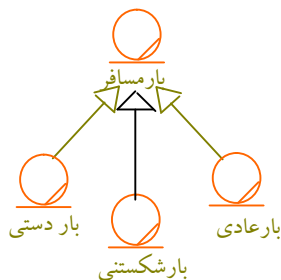
۲-۴-۲ تعیین کارکنها و موجودیتهای کاری

پس از تعیین واحدهای سازمانی می‌بایست کارمندان و افراد را مشخص نمود. نقش کارمندا در سازمان در قالب یک یا چند کلاس مشخص می‌شوند. مشخصات هر نقش در قالب ویژگیها یا صفات و وظایف در قالب متدهای آن کلاس مشخص شود. هر نقش کاری یا به عبارت دیگر هر کارکن کاری در انجام یک یا چند مورد استفاده باید دخیل باشد و همکاریهای وی نیز باید مشخص باشند. ارتباط بین کارکنها باید مشخص شود. چند کارکن کاری ممکن است از یک جنس باشند. برای نمونه ثبت کننده و تایید کننده اسناد مالی هر دو از جنس کارمند حسابداری می‌باشند. در شکل زیر این ارتباط بین کارکنهای کاری مشخص شده است. رابطه تجمع نیز نوعی دیگر از رابطه است که در بین کارکنها ممکن است برقرار شود. برای نمونه در شکل زیر، با استفاده از رابطه تجمع نشان داده شده که هیئت رئیسه برای یک سازمان از تجمع معاونها و مدیرعامل تشکیل شده است.



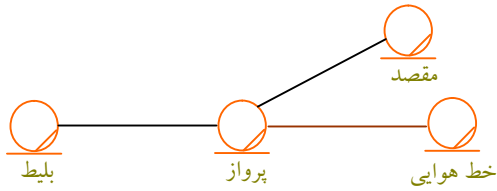
شکل ۱۹- رابطه های وراثت و تجمع

این افراد هستند که با همکاری با یکدیگر و با کار بر روی موجودیتهای کاری موردهای استفاده را فراهم می کنند. نابراین هر کارکن باید در حداقل یک مورد استفاده کاری موثر باشد. یک موجودیت کاری به هر نوع مدرک و مستندی که برای انجام امور مورد استفاده و یا ایجاد شود، اطلاق می شود. اصولاً یک موجودیت به آنچیزی که اطلاعات در آن نگهداری می شود اطلاق می شود. برای هر کارمند باید مشخص نمود که چه مستنداتی را جهت انجام امور محوله استفاده یا ایجاد می کند. هر مستند در واقع یک موجودیت کاری است. بین مستندات نیز ممکن است رابطه برقرار باشد. برای نمونه پرونده یک کارمند، از رابطه تجمع بین مدرک تحصیلی، سابقه کاری و مدرک پست سازمانی و غیره تشکیل می شود. اصولاً رابطه بین هر دو کلاس اشیا را در قالب رابطه تجمع مشخص می کنند. برای نمونه رابطه بین کارمندی که کارت پرواز برای مسافرها صادر می کند و موجودیت دستورالعملهای پرواز یک رابطه تجمع است.



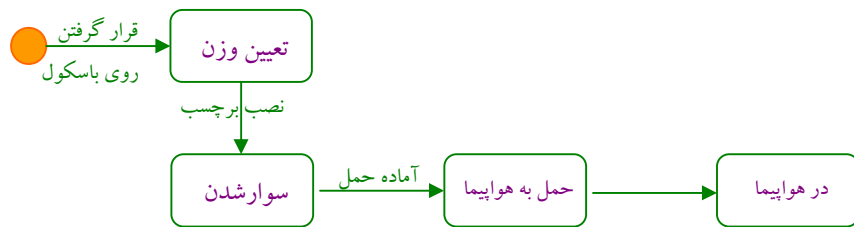
شکل ۲۰- رابطه وراثت در بین موجودیتهای کاری

هر نوع ارتباط بین موجودیتهای کاری را باید مشخص نمود. برای نمونه موجودیت کاری پرواز در ارتباط با موجودیت مقصد پرواز قرار می گیرد. همچنین در ارتباط با بلیط و خط هوایی است. بنابراین مدل ارتباطی به صورت زیر خواهد بود.



شکل ۲۱- رابطه بین موجودیتهای کاری

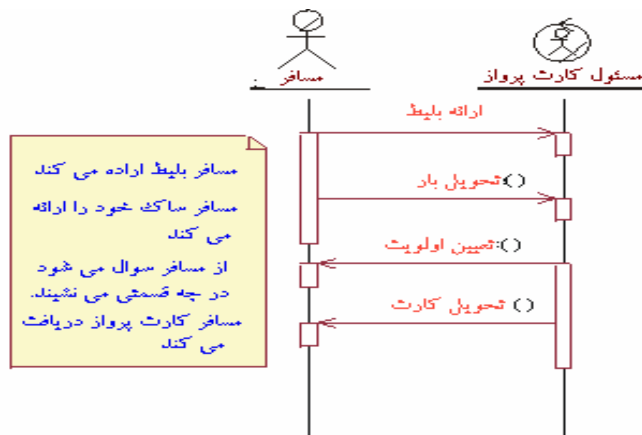
برای توصیف عملکرد موجودیتهای کاری می توان از دیاگرامهای حالت استفاده نمود. اصولاً، ارتباط بین موجودیتهای کاری از طریق فرامین و پیامهایی است که برای یکدیگر ارسال می کنند. در واقع واکنش موجودیت در مقابل رویداد موجب می شود که شیئی به حالتی جدید وارد شود. برای نمونه به مثال ذیل توجه نمایید.



شکل ۲۲- دیاگرام حالت برای بار مسافر

۲-۴-۳ تحقیق موردهای استفاده کاری

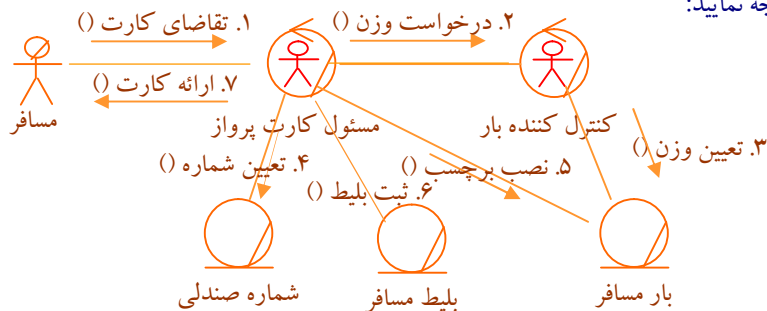
از همکاری و عملکرد کارکنها بر روی موجودیتهای کاری موردهای استفاده کاری تحقیق می یابند. این همکاری را می توان در قالب دیاگرامهای توالی و یا دیاگرامهای همکاری مشخص نمود. باید توجه نمود که دیاگرام توالی به صورت دقیقتر با تعیین توالی زمانی در عملیات، عملکرد یک مورد استفاده کاری را مشخص می کند. برای نمونه در شکل زیر یک دیاگرام توالی برای نمایش مراحل گرفتن کارت پرواز جهت تعیین شماره صندلی و تحویل بار مشخص شده است.



شکل ۲۳- دیاگرام توالی برای دریافت کارت

باید توجه نماییم که تولید دیاگرام توالی برای موردهای استفاده در مرحله آنالیز معمولاً انجام می شود. مدل اشیاء کاری در این مرحله برای هر واحد سازمانی مشخص می شود. یک نمونه در ابتدای این فصل مشخص شد.

ترسیم دیاگرام همکاری معمولاً ساده تر از دیاگرام توالی است. در واقع می توان منطق عملیات یک مورد استفاده کاری را در قالب یک دیاگرام همکاری، بسادگی مشخص نمود. سوال اینجا است که با وجود سناریو و دیاگرام فعالیت که قبلاً برای توصیف عملکرد سرویسها یا در اصطلاح موردهای استفاده از سیستم کاری مشخص شده اند، چرا باید دیاگرام همکاری نیز ترسیم نمود. علت این است که با ایجاد دیاگرام همکاری می توان یک گام به سمت ایجاد برنامه شیئی گرا نزدیکتر شد زیرا، در واقع مشخص می شود که چگونه سرویسهای ارائه شونده و یا به عبارت دیگر موردهای استفاده از سیستم کنونی از همکاری بین کارکنها با ارجاع و ثبت نتایج در مستندات، پرونده ها و مدارک که در قالب موجودیتهای کاری مشخص می شوند به انجام می رسد. برای نمونه به شکل زیر توجه نمایید:



شکل ۷-۲ دیاگرام همکاری تیمی

۴-۳ تبدیل مدل کاری

از مدل‌های کاری برای تعیین نیازها استفاده می‌شود. همچنین در مرحله تحلیل هر مورد استفاده کاری را معمولاً به عنوان یک زیر سیستم در نظر می‌گیرند. برای نمونه اگر فرایندی برای پردازش سفارش در سیستم جاری دارید. این فرایند کاری تبدیل به یک زیر سیستم در مدل تحلیلی شما زیر سیستم سفارش مشخص می‌شود. برای عملیات هر کارکن سیستم جاری مورد های استفاده را مشخص کنید تا مشخص شود که چه بخشهایی از عملیات کارکنها باید توسط سیستم مکانیزه انجام شود. برای هر موجودیت کاری می‌بایست در مرحله تحلیل کلاسها از نوع موجودیت کاری ایجاد شود. عمل تبدیل مدل‌های کاری در مرحله تحلیل با تجزیه و تحلیل کارکنها آغاز می‌شود. یک کارکن ممکن است در قالب کاربر سیستم را مورد استفاده قرار دهد. معمولاً اکتورهای تعین شده در مرحله مدل‌سازی کاری با کاربرهای واقعی سیستم مکانیزه یعنی کارکنها جایگزین می‌شوند. در اینصورت هر گونه تصمیم‌گیری کارکنها در داخل موردهای استفاده کاری باید به عنوان یک ورودی برای مورد استفاده در نظر گرفته شود. باید در صورت امکان عملیات مربوط به کارکن که توسط کامپیوتر انجام می‌شود در قالب یک مورد استفاده از سیستم کامپیوتری مشخص شود. در این صورت بخشی از مسئولیتهای کارکنها به سیستم مکانیزه سپرده می‌شود. در واقع بر اساس یک روش متمرکز بر خواسته‌های کارکنها، فرایند مهندسی نرم افزار می‌توانست مستقیماً با تعیین کارکنها و نیازهای آنها آغاز شود. در اینجا مشاهده می‌کنید که مستندهای اولیه که بیانگر اکتورها یا بازیگرهای سیستم بود دیگر در مورد سیستم مکانیزه هیچگونه مصدافی ندارد و تعیین آنها عملی بیهوده و عبس بود. به هر حال از آنجا که عمده از نیازها در عملکرد سیستم جاری نهفته است مدل‌سازی کاری می‌تواند به قیمت بعضاً بسیار گزاف در شناخت نیازهای متمرکز بر خواسته‌های سیستم موثر باشد.

فصل ۳

تحلیل نیازها

۳-۱ مقدمه

بر اساس مدل‌سازی کاری که در مرحله قبلی توضیح داده شد می‌توان برخی از نیازهای که در ارتباط با عملکرد سیستم جاری است را مشخص نمود. البته یوردون در سال ۱۹۸۹ ایده مدل‌سازی جزئیات سیستم جاری را برای تعیین نیازها منتقی و کاری بهبوده دانست. حتی در روش UP نیز تأکیدی بر مدل‌سازی سیستم کاری نشده است. در روش USDP تأکید شده که مطالعه سیستم جاری می‌تواند در شناخت نیازها موثر باشد زیرا، اغلب نیازها سیستم آتی همان عملیاتی است که در سیستم جاری انجام می‌شود. اما، این روش نیز تأکیدی بر مدل‌سازی سیستم جاری ندارد. در این فصل به ذکر مراحل تعیین نیازها و نهایتاً "تهیه گزارشی به نام چشم انداز سیستم پرداخته خواهد شد.

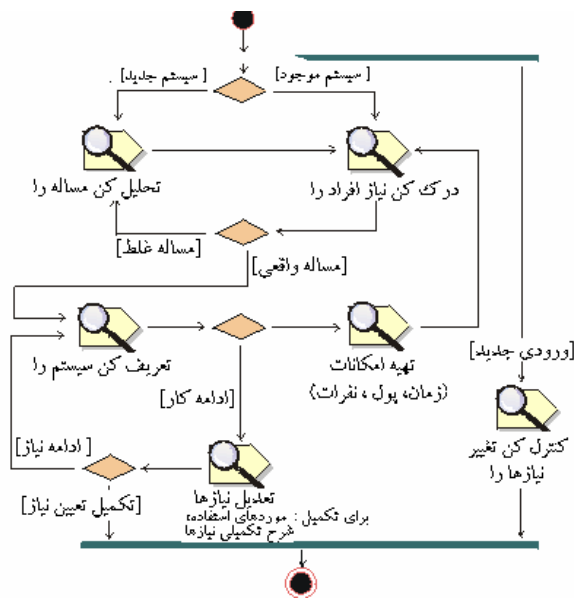
۳-۲ مراحل تعیین نیازها

نیازها اغلب مبتنی بر عملکرد افراد می‌باشد. لذا، با شناخت و مدل‌سازی سیستم جاری تا حد زیادی نیازها مشخص می‌شوند. برای نمونه مسلماً "برای یک سیستم آموزش نیاز است که اطلاعات ثبت نام نگهداری شوند و سیستم باید این قابلیت را داشته باشد که اطلاعات ثبت نام را در اختیار کارمند آموزش قرار دهد. برخی دیگر از نیازها براساس مسائل موجود و تحلیل آنها مشخص می‌شود. برای نمونه تاخیر در رساندن آمار فروش به بخش تولید یک مشکل است که می‌توان با وجود سیستم مکانیزه آخرین آمار فروش را در هر لحظه در اختیار واحد تولید قرار داد. برخی دیگر از نیازها امکانات اضافی است که ممکن است مورد نظر کاربر باشد و یا اینکه به وی پیشنهاد شود.

هدف از مرحله تحلیل نیازها، توافق با کاربر در مورد امکانات و قابلیت‌های مورد انتظار از سیستم مورد نظر است. این قابلیت‌ها در قالب مستندی تحت عنوان چشم انداز پروژه مشخص می‌شود. در واقع چشم انداز پروژه گزارشی است که بر مبنای آن قرار داد با مشتری مشخص می‌شود. در چشم انداز پروژه، نیازها ابتدا در قالب ویژگی‌های کلیدی مشخص می‌شوند. همگام با ایجاد مستندات چشم انداز

باریگرها و موردهای استفاده از سیستم مطلوب نیز مشخص می شوند. مستندات چشم انداز سیستم نمایانگر کلیت نیازها و قابلیت‌های سیستم است. جزئیات نیازها در مستندی به نام شرح تکمیلی نیازها مشخص می شود. در این مرحله جهت تعیین نیازها مدل موردهای استفاده از سیستم مکانیزه ایجاد می شود و با تحلیل نیازها تکمیل می شود.

باید همانگونه که در شکل زیر مشخص شده است، نیازهای افراد را مشخص نمود و تحلیل کرد تا اینکه مسائل و مشکلات مشخص شوند.



شکل ۳-۱ گردش عملیات برای تعیین نیازها

۳-۳ انواع نیازها

نیازها بیانگر قابلیت‌ها و یا شرایطی است که سیستم مورد نظر باید داشته باشد. نیازها ممکن است متمرکز بر خواسته کاربرها و یا متمرکز بر اهداف سیستم باشند. جهت افزایش قابلیت بهره‌بری که یکی از معیارهای عمده در مهندسی نرم افزار برای سنجش نرم افزار است، نیازهای متمرکز بر خواسته‌های کاربر اهمیت بیشتری دارند. نیازها در حالت کلی در سه دسته عملیاتی، کیفی و قابلیت بهره‌بری تقسیم بندی می شوند. البته در مستندات آر.یو.پی نیازها را به پنج دسته عملیاتی، قابلیت بهره‌بری، قابلیت اطمینان، کارایی و قابلیت‌های پشتیبانی تقسیم بندی کرده‌اند.

نیازهای عملیاتی مشخص می کنند که یک سیستم چه کارهایی باید انجام دهد. جهت تعیین عملیات مورد نیاز مسلماً "فرایندهای عملیاتی یا به عبارت دیگر پردازش ها و هر گونه ورودی، خروجی و داده ای که مورد استفاده قرار می دهند باید مشخص شود. نیازهای عملیاتی معمولاً در قالب مجموعه ای از قابلیتها و پیشگیریهای امنیتی مشخص می شود. نیازهای عملیاتی را در قالب کلی؛ سیستم باید عمل X را انجام دهد مشخص می کنند. برای نمونه: سیستم باید معدل دانشجو را کنترل کند.

نیازهای کیفی بیانگر ویژگیهای خاص و یا محدودیتهای سیستم هستند. در واقع کیفیت عملیات مورد نیاز را مشخص می کنند. این دسته از نیازها در ارتباط با کارایی سیستم، امنیت، رابط کاربر، تحمل خطا و بالاخره محیط و شرایط عملیاتی مورد نظر برای سیستم مکانیزه می باشند. برای نمونه: سیستم باید در صورت قطع شبکه اطلاعات ارسالی را در یک صف انتظار قرار دهد.

نیازهای کارایی نوعی از نیازهای کیفی هستند زیرا، بیانگر شرایطی هستند که بر نیازهای عملیاتی اعمال می شود. این نوع نیازها در ارتباط با سرعت، بازدهی، قابلیت دسترسی، صحت، حجم داده های مورد پردازش، زمان پاسخگویی، زمان لازم برای رفع مشکلات و استفاده از منابع می باشند. منظور از قابلیتهای پشتیبانی، نیازهایی در ارتباط با قابلیت تطبیق نرم افزار با محیطهای مختلف، قابلیت توسعه و ترمیم نرم افزار، هماهنگی نرم افزار با نرم افزارهای موجود، قابلیتهای مورد نیاز برای آزمون سریع نرم افزار و بالاخره قابلیت نصب و تطبیق با کاربر است. قابلیت پشتیبانی را نیز می توان یک نوع نیاز کیفی در نظر گرفت. نیازها در ارتباط با طراحی نرم افزار ممکن است مطرح شود. نیازهای پیاده سازی، در ارتباط با سیستم عامل، بانک اطلاعاتی، زبان، نرم افزارهای مورد استفاده و استانداردهای مطلوب جهت پیاده سازی نرم افزار است. هر گونه محدودیت و شرایط تعیین شده برای ارتباط سیستم مکانیزه با کاربر، دستگاه های جانبی و محیط خارج از سیستم، یک نوع نیاز کیفی است.

قابلیتهای بهره بری در واقع مشخص می کنند که تحت چه شرایطی برای کاربر، سیستم مکانیزه قابل استفاده خواهد بود. معمولاً "قابلیت یادگیری که از طریق مستندات و راهنماهایی های موجود در محیط اجرایی نرم افزار فراهم می شود در قابلیت بهره بری موثر است. سرعت انجام عملیات برای کاربرها می تواند بسیار حساس باشد. قابلیت اعمال سریع تغییرات در محیطهایی که دستخوش تغییر و تحول زیاد هستند نیز حائز اهمیت است.

۳-۴ چشم انداز پروژه

مستند چشم انداز پروژه انتظارات متقاضی را از نرم افزار مورد نظر وی مشخص می کند. لذا، این مستند باید قابلیت‌های سیستم کامپیوتری را با کیفیت قابل قبول برای متقاضی مشخص کند. در چشم انداز پروژه باید همچنین ظرفیتهای عملیاتی از قبیل حجم داده ها، سرعت واکنش و میزان دقت در عملیات و میزان منفعت سیستم برای کاربرها مشخص شود.

چشم انداز پروژه بر اساس نیازهای تعیین شده، سیستم مکانیزه ایجاد و ارزیابی می شود. بنابراین نیاز به روشی مشخص برای تعیین و تدوین نیازها و تایید حاصل با کاربر است. برای سازماندهی نیازهای تعیین شده ساختاری در قالب چشم انداز پروژه توسط آر.یو.پی ارائه شده است. برای ایجاد چشم انداز^۲ باید مساله مطرح و تحلیل شود. مراحل تحلیل عبارتند از:

فعالیت: ایجاد چشم انداز

مراحل آنالیز مساله

۱- ایجاد توافق در مساله ای که باید حل شود

۲- تعیین کاربرها

۳- تعیین سرردهای سیستم

۴- تعیین محدودیتهای اعمال شونده بر سیستم

در هنگام تهیه چشم انداز پروژه باید در نظر داشت که مساله مطرح شود و قابلیت‌های سیستم در پاسخگویی به مسائل و نیازهای مطرح شده تعیین شود. مستند چشم انداز در ابتدای مرحله شناخت ایجاد می شود و در مرحله های بعدی تکمیل می گردد. در چشم انداز پروژه بطور خلاصه موارد زیر مطرح است:

چرا؟ برای تعیین علت انجام پروژه موقعیتها مشخص می شوند منظور از موقعیتها، تعیین موقعیت کاری حاصل از بکارگیری محصول نرم افزاری، شرح کلی مساله و مشکلات حاصل از آن و بالاخره موقعیت محصول در بازار است.

چه کسانی؟ برای تعیین اینکه چه کسانی در ارتباط با پروژه و محصول نرم افزاری حاصل از آن قرار می گیرند افراد مرتبط با پروژه، بازار محصول نرم افزاری شامل آمار تعداد متقاضی های احتمالی، کاربرها و نیازهای کلیدی آنها و پاسخ سیستم به نیازهای آنها مشخص می شود. در این ارتباط در بخش بعد، محصول و ویژگیهای آن در پاسخ به نیازها مشخص می شود.

چه محصولی؟ مسلماً پس از ذکر مشکلات و نیازها باید محصول نرم افزاری و قابلیت‌های آن مشخص گردد. برای این منظور محیط عملیاتی، خلاصه‌ای از قابلیت‌ها و مزایای آنها برای مشتری، عوامل موثر در قابلیت‌ها و هزینه‌های لازم برای دستیابی به قابلیت‌ها مشخص می‌شود. در ادامه شرح خلاصه‌ای از هر یک از قابلیت‌های سیستم ارائه می‌شود.

چگونه؟ در این راستا محدودیت‌های سیستم، کیفیت محصول و معیارهای ارزیابی کیفیت و بالاخره کارایی محصول نرم افزاری مطرح می‌شود.

در ادامه شرح کامل چشم انداز پروژه همراه با مثالی از یک سیستم ثبت نام ارائه شده است.

۱. مقدمه

۱-۱ هدف: در ابتدا هدف از گزارش چشم انداز پروژه یا در اصطلاح ویرژن باید مشخص شود. برای نمونه، هدف از این گزارش تعیین نیازهای کاربرهای سیستم ثبت نام دانشگاه است.

۲-۱ دامنه: باید مشخص شود که چه کسانی یا سازمانهایی در ارتباط با سیستم قرار دارند. برای نمونه در ارتباط با سیستم آموزش دانشگاه دامنه به این صورت مشخص می‌شود که سیستم ثبت نام در ارتباط با بانک اطلاعاتی موجود که شامل جزئیات دوره‌های آموزشی در دانشگاه است قرار می‌گیرد. دانشجویان می‌توانند مستقیماً با این سیستم در ارتباط قرار گرفته، ثبت نام کنند. همچنین اساتید می‌توانند به صورت برخط درس مورد ارائه خود را مشخص کنند و یا نمره دانشجویان را وارد کنند.

۳-۱ تعاریف: در این بخش تعریفی از کلیه علائم اختصاری و اصطلاحاتی که در پروژه بکار برده می‌شود ارائه می‌شود. این نوع اطلاعات معمولاً در واژه نامه پروژه نیز مشخص می‌شود. لذا؛ معمولاً در این قسمت، خواننده را به دیکشنری لغات ارجاع می‌دهند.

۴-۱ مراجع: در این بخش سایر مستندات که از درون این مستند به آنها ارجاع شده است باید مشخص شوند. مشخصات شامل شماره گزارش، تاریخ و محل دستیابی به گزارش است. برای نمونه در مورد سیستم ثبت نام مراجع عبارتند از:

گزارش شماره د.و.۱۲۲ شرح عملیات سیستم ثبت نام، ۱۹۹۸، انتشارات دانشگاه.

گزارش شماره ص.و.۱۱۰ شرح صورتحسابهای دوره، ۱۹۹۵، انتشارات دانشگاه

بروشور شماره ب.د.ت.۹۹ شرح دوره‌های تحصیلی، ۱۹۹۴، وزارت علوم

گزارش شماره ن.ی. ۵۰ شرح نیازهای کاربرهای سیستم، ۱۹۹۸
لغتنامه شماره ل.ن. ۵۶۰ شرح اصطلاحات سیستم ثبت نام، ۱۹۹۸

۱-۵ خلاصه : در این بخش از مقدمه، خلاصه ای از سایر قسمتهای گزارش چشم انداز ارائه می شود.

۲. موقعیت

۲-۱ موقعیت کاری : در این بخش بطور خلاصه موقعیت کاری که توسط این پروژه ایجاد می شود تعیین می گردد. برای نمونه پروژه ثبت نام، بطور کامل ارتباط با دانشجویها و اساتید را از طریق ایستگاه های سرویس گیرنده امکان پذیر می کند. سیستم جاری از سال ۱۹۸۵ فعال بوده است و قادر به پاسخگویی به تعداد زیاد دروس و دانشجویان در سال ۲۰۰۳ نیست. سیستم مکانیزه جذابیت خاصی برای کالج فراهم نموده که موجب می شود دانشجویان به سادگی با امکانات کالج آشنا شوند و ثبت نام نمایند.

۲-۲ مشکل : در این بخش مشکلات و مسائلی که سیستم جاری با آن مواجه است مشخص می شود. برای هر مشکل باید ابتدا تعریفی ارائه شود. سپس، باید مشخص شود که این مشکل بر چه مواردی تاثیر می گذارد. و تاثیر آن چیست. نهایتاً باید مشخص شود که سیستم مکانیزه چه مزایایی در جهت رفع مشکلات ذکر شده، خواهد داشت.

برای نمونه مشکل، روش قدیمی و دستی برای ثبت نام در کالج است که دانشجو، استاد و کارمندیهای آموزش را تحت تاثیر خود قرار داده است. کندی مراحل ثبت نام موجب عدم رضایت این افراد شده است. وجود سیستم مکانیزه تصویر بهتری از کالج فراهم نموده، موجب رضایت کارمند، استاد و دانشجو و بالتیجه موجب جذب تعداد دانشجوی بیشتری برای کالج خواهد شد.

۲-۳ موقعیت محصول : مسلماً جایگاه یک محصول نرم افزاری در بازار برای تولید کننده های آن بسیار مطرح است. باید در ابتدا مشخص نمود که چه ویژگیهایی از محصول نرم افزاری موجب موفقیت آن در بازار خواهد شد. برای این منظور باید مشخص نمود که برای کدام دسته از مشتریان چه ویژگی و مزایای خاصی این محصول نرم افزاری دارد و در مقایسه با محصولات موجود چه برتری ای دارد. برای نمونه در یک سیستم اتوماسیون اداری قابلیت برقراری ارتباط با سیستم از طریق وب

برای مدیران بسیار قابل توجه است. موقعیت محصول برای کالچ مورد نظر را می توان به صورت زیر مشخص نمود.

برای دانشجویها، اساتید و واحد ثبت نام که به ترتیب درخواست کننده، آموزش دهنده و ثبت نام کننده هستند، سیستم مکانیزه ثبت نام ابزاری است که دسترسی برخط و سریع به عملیات ثبت نام، لیست دروس، وضعیت و کارنامه دانشجویها را فراهم می کند. برخلاف سیستم کنونی که دستی و خارج از رده است، محصول نرم افزاری ما اطلاعات به روز در مورد کلیه دروس، ثبت نامها، مدرسین و کارنامه ها را از طریق هر کامپیوتر شخصی فراهم می نماید.

۳. توصیف افراد و کاربرها

قابلیتهای محصول نرم افزاری بر اساس نیازها و خواسته های متقاضی مشخص می شوند. اصولاً "نیازها و خواسته ها از دیدگاه افراد و سیستم باید مشخص شود. هر فردی در ارتباط با سیستم مکانیزه نیازهایی را مطرح می کند. اما سازمان یا سیستم مربوطه نیز نیازهایی را دارد که در قالب شرح وظایف برای افراد مشخص شده است. بنابراین خواسته ها و نیازها بر دو منبع افراد و سیستم متمرکز هستند. برای نمونه در شرح وظایف مسئول آموزش تهیه کارنامه برای دانشجویان درج شده است. این یک نیاز سیستمی است که از شرح وظایف می توان آنرا استخراج نمود و همچنین ممکن است خود کارمند نیز آنرا مطرح کند.

برای تعیین نیازها باید چارت سازمانی، چارت عملیاتی و شرح وظایف افراد را مشخص نمود. اکنون برای هر واحد عملیاتی و هر پست سازمانی می توان نیازها را مشخص کرد و قابلیتهای محصول نرم افزاری که در پاسخ با این نیازها می باشند را تعیین نمود. در اینجا تاکید بر عملکرد افراد است.

۳-۱ آمار بازار: در این بخش به طور خلاصه آمار گیری هایی که موجب تصمیم گیری در مورد محصول نرم افزاری می شوند مشخص می گردد. برای این منظور اندازه و پیشرفت بازار محصول با تعیین تعداد افرادی که از این محصول استفاده می کنند، مشخص می شود. با تعیین مبلغی که بطور متوسط افراد حاضر به پرداخت برای رفع نیازهایشان توسط محصول نرم افزاری می شوند نیز می توان بازار محصول را مشخص کرد. در واقع باید مشخص نمود که تا چه حدی سیستم نرم افزاری موجب پیشرفت سرویس دهی و بازار کار متقاضی می شود.

استفاده کننده ها از سیستم ثبت نام افراد دانشگاهی هستند که مجموعه نسبتاً بزرگی را تشکیل می دهند. بعضاً دارای کامپیوترهای شخصی هستند و یا اینکه می توانند با استفاده از امکانات دانشگاه به کامپیوتر دسترسی داشته باشند. قابلیت ثبت نام برای کلیه دانشجویان و کارمندان بخش آموزش از طریق سیستم کامپیوتری بوجود خواهد آمد. نرم افزار در ابتدا برای کالج مورد نظر و سپس برای کلیه دانشگاه ها ایجاد خواهد شد.

۳-۲ خلاصه افراد مرتبط : در این قسمت لیستی از نام، پست سازمانی و مسئولیتهای کلیه افرادی که می توانند در ارتباط با محصول نرم افزاری باشند، مشخص می شود. ممکن است افراد زیادی وجود داشته باشند که متقاضی محصول نرم افزاری باشند اما، مستقیماً با سیستم کامپیوتری درگیر نشوند. جدول زیر شامل شرح خلاصه ای از افراد درگیر با سیستم ثبت نام می باشد

نام	نماینده	نقش
مدیر فن آوری اطلاعات	- واحد فن آوری کالج	- مسئولیت تصویب پروژه - کنترل کننده پیشرفت پروژه
آموزش	- واحد ثبت نام - کارمندا - وارد کننده های جزییات	- حصول اطمینان از برآورده شدن خواسته های آموزش که ثبت نام را انجام می دهد.
دانشجو	- دانشجوها	- تایید خواسته های دانشجوها
استاد	- هیات علمی	- ارائه خواسته های دانشکده

۳-۳ خلاصه کاربر : در این قسمت ابتدا نوع کاربر مشخص می شود. برای نمونه کارمند یک نوع از کاربر است. سپس شرح خلاصه ای از مسئولیتهای هر نوع کاربر که در ارتباط با محصول مورد نظر است مشخص می شود. و بالاخره مشخص می شود که نوع کاربر مشخص شده نماینده و شاخص کدام دسته از افراد است.

نام	شرح	افراد مرتبط
آموزش	کنترل و ثبت اطلاعات دانشجویان و اساتید، اعلان شروع و خاتمه ثبت نام	نماینده خود
دانشجو	ثبت نام دروس، درخواست نمره و جزییات دروس	نماینده خود
استاد	انتخاب دروس برای تدریس. وارد کردن نمره دانشجوها.	نماینده خود

۳-۴ محیط کاربر : در این قسمت محیط کاری کاربر مشخص می شود. برای نمونه تعداد افراد، مدت زمان، تعداد برنامه های کامپیوتری و محدودیتهایی که برای انجام یک وظیفه وجود دارد را می توان مشخص نمود. در صورت وجود برنامه های کامپیوتری باید مشخص نمود که ارتباط سیستم جدید کامپیوتری با آن برنامه ها به چه صورت خواهد بود. باید مشخص کرد که چه سیستم عامل و شبکه کامپیوتری در حال حاضر مورد استفاده قرار می گیرد و در آینده به چه صورتی تغییر می کند.

برای نمونه استفاده کننده ها از کامپیوتر در سیستم ثبت نام، جامعه نسبتاً بزرگی را تشکیل می دهند که انتظار انعطاف و پاسخگویی سریع از یک سیستم برخط ثبت نام دارند. استفاده کننده ها افراد تحصیلکرده و اغلب صاحب کامپیوترهای شخصی هستند. برنامه می بایست برای ثبت نام در دانشگاه های مختلف تهیه شود.

۳-۵ وضعیت افراد مرتبط : در این قسمت برای هر کدام از کاربرها پس از توصیف نوع کاربر و اینکه شاخص چه مجموعه ای از کاربرها می باشد، مسئولیتها و معیارهای وی در ارتباط با موفقیت پروژه مشخص می شود. همچنین نوع ارتباط هر کاربر با پروژه تعیین می شود.

نام کاربر	
نماینده	کاربر نماینده کدام دسته از افراد است ؟
شرح	شرح خلاصه ای از این دسته از کاربرها در این قسمت ارائه شود.
نوع	تجربه فرد، سابقه تکنیکی، نوع کاربر : ناشی، مجرب، دائمی و یا گهگاه.
مسئولیتها	مسئولیتهای کلیدی کاربر در ارتباط با سیستم مکانیزه جدید
معیار موفقیت	از دیدگاه کاربر در چه صورتی نرم افزار موفق خواهد بود.
نوع ارتباط	کاربر به چه صورتی در پروژه موثر خواهد بود؟ برای نمونه ممکن است در تهیه لیست نیازمندیها، بررسی مستندات و غیره از وی استفاده شود.
محصول	چه محصولی کاربر برای چه واحدی تولید می کند.
نکات	مشکلاتی که مانع موفقیت پروژه می شوند. این نکته در ارتباط با نرم افزارهایی است که وجود آنها کار کاربر را سختتر نموده است.

برای نمونه در سیستم ثبت نام دانشگاه می توان سابقه افراد مرتبط با پروژه را برای مدیر واحد فن آوری، نماینده واحد ثبت نام، استاد و دانشجو مشخص نمود. در ادامه برای مدیر واحد فن آوری و دانشجو وضعیت کاری در ارتباط با پروژه ثبت نام مشخص شده است.

مدیر فن آوری	
نماینده	آقای ایکس، مدیر واحد فن آوری
شرح	حق تایید پروژه را دارد
نوع	از وضعیت اقتصادی کالج و نقطه نظرهای ریاست کالج اطلاع دارد.
مسئولیتها	پیشرفت پروژه را پیگیری می کند و تصویب بودجه به عهده وی است. همچنین اطمینان حاصل می کند که عملکرد محصول نرم افزاری در جهت اهداف برنامه ریزی های بلند مدت و کوتاه مدت

کالج است.	
از دیدگاه مدیریت موفقیت پروژه در تکمیل شدن آن با بودجه و زمان تعیین شده و کاهش بار کاری اداره آموزش است. ریاست دانشگاه تکیه بر این نکته دارد که پروژه پاسخگوی خواسته های کاربر و بسادگی قابل توسعه باشد.	معیار موفقیت
مرور پروژه، بررسی افزایش کارایی حاصل از محصول نرم افزاری و تصویب بودجه.	نوع ارتباط
هیچ	نکات

	دانشجو
خانم ایگرگ استفاده کننده	نماینده شرح
نماینده شورای صنفی دانشجویی	نوع
اطمینان حاصل می کند که سیستم برای دانشجویان قابل استفاده است. بخصوص معیارهای سهولت در استفاده، قابلیت اطمینان و کارایی بسیار حائز اهمیت است. حدود ۱۰٪ از دانشجویان برای اولین بار از سیستم استفاده خواهند کرد اما، بقیه با سیستمهای کامپیوتری آشنایی دارند. در مجموع ۲۰۰۰ دانشجو از این برنامه استفاده خواهند کرد.	مسئولیتها
موفقیت هنگامی بدست می آید که در هنگام ثبت نام دانشجویها بتوانند به سادگی سیستم را مورد استفاده قرار دهند و همگی اذعان داشته باشند که سیستم به سادگی قابل استفاده است.	معیار موفقیت
بررسی کننده پروژه بخصوص در زمینه هایی که مربوط به دانشجویها می شود.	نوع ارتباط
هیچ	نکات

۳-۶ وضعیت کاربر : در این قسمت با استفاده از جدولی مشابه با جدول استفاده شده برای توصیف افراد مرتبط با پروژه، انواع کاربرها مشخص می شوند. برای نمونه در سیستم ثبت نام، دانشجو یک کاربر است که برای وی وضعیت در قسمت قبلی مشخص شده است.

۳-۷ نیازهای کلیدی کاربر : در این قسمت نیازها، مشکلات و راه حلهای پیشنهادی ارائه می شود. نیازها دسته بندی و طبقه بندی می شوند. برای تعیین نیازها می توان از امکانات رشنال ریکوایزیت پرو که ابزاری برای پیگیری نیازها است استفاده نمود. در صورت عدم استفاده از ریکوایزیت پرو می توان از جدولی که در انتهای این قسمت ارائه شده استفاده کرد. برای نمونه در سیستم ثبت نام نماینده ای از دانشجویها، اساتید و واحد ثبت نام برای بررسی نیازها و مسائل کاربر در ارتباط با سیستم جاری ثبت نام به صورت زیر مشخص شده است.

نیاز	اولویت	دلیل	راه حل کنونی	راه حل پیشنهادی
ثبت نام دروس	بالا	کند بودن ثبت نام	تکمیل فرم-انتظار برای حداقل دوهفته-تغییر برنامه در این مدت-نیاز صورت بر خط از وضعیت به تکرار مراحل بواسطه تغییرات-عدم موفقیت دانشجو در ثبت نام برای کلیه دروس مورد نظر وی. نام کنند.	دانشجوها می خواهند به صورت برخط به اطلاع پیدا کرده در هر درس مستقلاً ثبت نام کنند.
دسترسی سریع به نمره ها	متوسط	تاخیر زیاد در گرفتن نمره	استاد درس را به واحد ثبت نام اعلان می کند و دانشجو ملزم به ارجاع به واحد آموزش برای مشاهده نمرات خود است.	دسترسی برخط به نمرات از طریق سیستم کامپیوتری
تقلیل پرداخت حقوق	متوسط	پرداختهای نسبتاً بالا	۲ تا ۳ کارمند آموزش و دو تا سه نفر افراد کمکی برای مدت ۴۰۰ الی ۵۰۰ ساعت برای عملیات ثبت نام و همچنین از بین بردن همزمانی در ارائه دروس با متقاضی زیاد	این تلاش با وجود سیستم کامپیوتری به صفر تقلیل می یابد.

۳-۸ راه حلهای مختلف : در این قسمت لیستی از نرم افزارهای موجود می توان مشخص نمود. برای هر نرم افزار نقاط قوت و ضعف مشخص می شود و تعیین می شود که آیا مقرون به صرفه است که نرم افزار جدید ایجاد شود. البته در مورد سیستم ثبت نام عجله کاربر برای سیستم جدید و عدم دانش وی از سیستمهای مشابه و عدم تعهد اخلاقی تولید کننده نرم افزار موجب شد که در این قسمت هیچگونه محصولی ذکر نشود.

۴. شرح کلی محصول

مسلماً پس از ذکر نیازها و مشکلات کاربر می بایست کارایی محصول نرم افزاری مشخص شود. در این بخش، چشم اندازی سطحی از قابلیت‌های محصول نرم افزاری ارائه می شود. این بخش معمولاً از سه قسمت به شرح زیر تشکیل می شود:

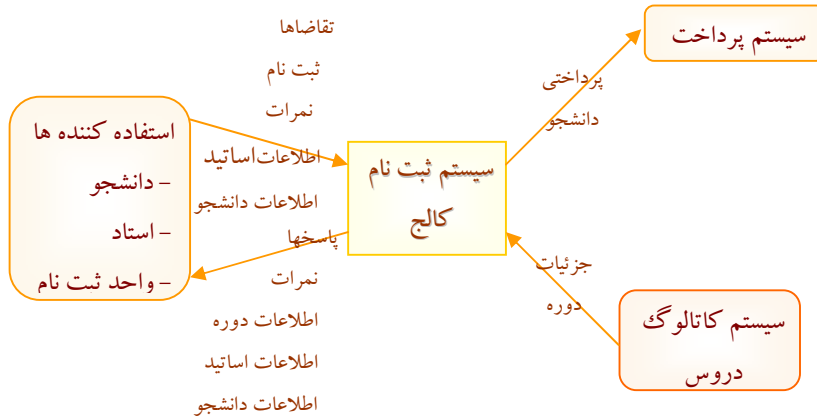
- ابعاد محصول
- عملکرد محصول
- پیش فرضها و وابستگیها

برای سیستم ثبت نام باید متذکر شد که قابلیت‌ها در این بخش به صورت کلی و در ارتباط با سیستم موجود پرداختهای مالی و بانک اطلاعاتی دروس مشخص شده است.

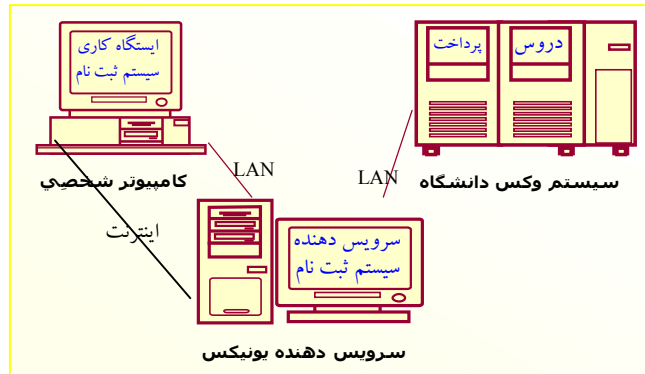
۴-۱ ابعاد محصول :

در این قسمت ابعاد محصول در مقایسه با سایر محصولات و محیط کاربر مشخص می شود. باید تعیین نمود که آیا محصول نرم افزاری به صورت مستقل عمل خواهد کرد و یا اینکه بخشی از نرم افزار موجود خواهد بود. چگونه با نرم افزارهای موجود ارتباط برقرار خواهد نمود و رابطهای ارتباطی به چه صورت خواهند بود. یک روش برای نمایش ارتباطات بین نرم افزارها، استفاده از بلاک دیاگرامها است. به قسمیکه، هر نرم افزار در یک کادر مستطیلی قرار گرفته و توسط خطوط، ارتباط آن با سایر نرم افزارهای موجود مشخص می شود.

برای نمونه سیستم در ارتباط با سیستم ثبت نام می توان گفت که این سیستم جایگزین سیستم موجود ثبت نام که بر روی کامپیوترهای بزرگ قرار داد، خواهد شد. این سیستم در ارتباط با سیستم پرداختهای واحد مالی و بانک اطلاعاتی دروس در اداره آموزش خواهد بود. این ارتباطها در دیاگرام متن زیر مشخص شده است.



نرم افزار سیستم ثبت نام از دو بخش تشکیل شده یک قسمت سرویس گیرنده یا کلاینت برای نصب بر روی ایستگاه های کاری و یک قسمت نرم افزار سرویس دهنده است که بر روی کامپیوتر سرویس دهنده کالج با سیستم عامل یونیکس نصب می شود. این سرویس دهنده در ارتباط با بانک اطلاعاتی سیستمهای پرداخت و کاتالوگ دروس در کامپیوتر بزرگ و کس دانشگاه قرار خواهد گرفت.



۴-۲ خلاصه قابلیت‌های محصول : خلاصه ای از مزایا و قابلیت‌های اصلی سیستم در این بخش مشخص می شود. در این قسمت لیست عملیات سیستم به صورتی قابل درک برای مشتری تهیه می شود. در ادامه جدولی برای تعیین مزایای عمده سیستم و قابلیت‌های آن مشخص شده است. در این جدول به عنوان نمونه در این قسمت قابلیت‌ها و مزایای کلی سیستم ثبت نام مشخص می شود. قابلیت‌ها به صورت دقیقتر در بخش ۷ مشخص شده اند.

قابلیت‌ها	مزایا برای مشتری
<ul style="list-style-type: none"> - سیستم به بانک اطلاعاتی دروس دسترسی دارد تا آخرین تغییرات دروس را در اختیار قرار دهد. - برای هر درس استاد و دانشجو پیش نیازها، مباحث درسی، محل برگزاری و نام استاد را می توانند مشاهده کنند. 	اطلاعات بروز دروس
<ul style="list-style-type: none"> - ثبت نامها برای هر درس در بانک اطلاعاتی ذخیره می شود. تا بتوان در مورد برگزاری یا حذف درس تصمیم گیری نمود. 	اطلاعات به روز ثبت نام
<ul style="list-style-type: none"> - اساتید نمره دانشجویها را حتی در خانه در سیستم ثبت نام وارد می کنند. - دانشجویها می توانند در خانه و یا در دانشگاه وارد سیستم ثبت نام شده نمره خود را مشاهده کنند. 	دسترسی سهل و سریع به نمره

۳-۴ پیش فرضها و وابستگیها : تمامی عوامل موثر بر کارایی و قابلیت‌های محصول نرم افزاری را در این بخش مشخص نمایید. برای نمونه ممکن است که فرض بر آن باشد که سیستم عامل خاصی مورد استفاده قرار می گیرد در غیر این صورت قابلیت‌های سیستم ممکن است فرق کند. برای نمونه در ارتباط با سیستم ثبت نام پیش فرضها به شرح زیر هستند:

- فرض بر این است که سیستم کنونی کاتالوگ دروس و پرداخت‌های مالی که بر روی کامپیوتر و کس قرار دارد پشتیبانی شود و حداقل تا سال ۲۰۰۵ برقرار باشد.
- ساختار بانک اطلاعاتی دروس و پرداخت‌های مالی هیچگونه تغییری نکند.
- تحویل بموقع وابسته به پرداخت بموقع بر طبق قرار داد منعقد خواهد بود.

۴-۴ هزینه ها و قیمت‌ها : محدودیت‌های مالی تاثیر مستقیم بر قابلیت‌ها و کارایی محصول نهایی نرم افزاری دارند. در این قسمت باید این محدودیتها مشخص شود. برای نمونه محدودیت‌های مالی برای بخش محصول نرم افزاری را می توان در این قسمت مشخص کرد. در مورد سیستم ثبت نام می توان مشخص کرد که بخاطر محدودیت‌های مالی، هزینه توسعه سیستم نباید بیش از یکس تومان شود. پیش بینی می شود که کامپیوتر کنونی کالج مورد استفاده قرار گیرد و سخت افزار جدیدی مورد نیاز نباشد.

۴-۵ مجوز و نصب

مجوز نصب و نکات نصب می تواند تاثیر مستقیم بر کد برنامه ها داشته باشد. برای نمونه ایجاد کلمه عبور و یا مجوزهای ارتباطی در شبکه بار برنامه نویسی را افزایش می دهد. برای سیستم ثبت نام هیچگونه مجوزی مشخص نشد است.

۵- قابلیت‌های محصول

در این قسمت لیستی از قابلیت‌های محصول مشخص می شود. باید توجه نمود که منظور از قابلیت، توانایی‌های سطح بالای سیستم است که برای کاربر سودمند خواهد بود. سطح بالا بودن قابلیت‌ها به این مفهوم است که وارد جزئیات طراحی و پیاده سازی نباید شد و به صورت ساده و قابل درک برای افراد غیر کامپیوتری بیان شود. باید تاکید بر توانایی‌های مورد نیاز نمود و مشخص کرد که این قابلیت پاسخگوی آن خواهد بود.

برای نمونه در این قسمت برای سیستم ثبت نام مشخص شده است که این بخش شامل توصیف قابلیت‌های سیستم ثبت نام است. با توصیف قابلیت‌ها در واقع نمایانگر کلیت توانایی‌های سیستم ثبت نام هستند. در ادامه لیستی از قابلیت‌های سیستم ثبت نام به تفکیک ارائه می‌شود.

۱-۵ ورود : سیستم قابلیت تعیین کلمه عبور و نام کاربر را خواهد داشت. به این ترتیب هر کدام از دانشجویان و استادها می‌توانند جهت ورود به سیستم ثبت نام از کلمه عبور تخصیصی به خود استفاده کنند. برای کارمندان آموزش نیز کلمه عبور به صورت موقتی برای طول مدت ثبت نام ایجاد می‌شود. افراد قادر به تعویض کلمه عبور خود خواهند بود.

۲-۵ ثبت نام دروس : سیستم قابلیت نمایش لیست دروس بر اساس دانشکده، نام درس و کد درس را خواهد داشت. پس از اینکه دانشجو برای ثبت نام، درسی را مشخص نماید سیستم قادر به کنترل پیش نیاز و ظرفیت کلاس خواهد بود. البته تا قبل از خاتمه مهلت ثبت نام، سیستم باید امکان اصلاح و تغییر جزئیات ثبت نام را برای دانشجویان فراهم کند.

۳-۵ حذف دروس : سیستم امکان حذف دروس را برای کارمندان واحد آموزش فراهم خواهد نمود. واحد ثبت نام در پایان هر دوره ثبت نام به بررسی آمار ثبت نام در کلیه دروس پرداخته و در صورتی که در یک درس به اندازه کافی ثبت نام نشده باشد آن درس را از لیست دروس حذف می‌کند. واحد ثبت نام از طریق پست الکترونیک قادر به اعلان حذف دروس به کلیه ثبت نام کنندگان خواهد بود.

۴-۵ صورتحساب دانشجویان : سیستم در خاتمه دوره ثبت نام برای واحد حسابداری لیستی از دروس که دانشجو در آنها ثبت نام کرده همراه با مشخصات و آدرس دانشجو ارسال می‌کند تا این واحد مبلغ قابل پرداخت را به دانشجو اعلان نماید.

۵-۵ مشاهده مشخصات استادها : سیستم قادر به نمایش مشخصات استادها برای کارمندان آموزش و استادها خواهد بود.

۶-۵ مشاهده کارنامه : سیستم قادر به نمایش کارنامه برای دانشجو و استاد درس خواهد بود. هر دانشجو صرفاً قادر به مشاهده کارنامه خود خواهد بود.

۵-۷ انتخاب درس برای تدریس : سیستم قابلیت انتخاب درس جهت تدریس را برای استادها فراهم خواهد نمود.

۵-۸ درج جزئیات دانشجویها : واحد ثبت نام قادر به درج مشخصات دانشجو شامل مشخصات سجلی، آدرس و دانشکده وی خواهد بود. استاد راهنما قادر به مشاهده مشخصات دانشجویها خواهد بود. هر دانشجو تنها قادر به مشخصات خودش خواهد بود.

۵-۹ ثبت نمره دانشجویها : سیستم قادر به ثبت نمره دانشجویها از طریق استاد درس خواهد بود.

۵-۱۰ مشاهده لیست دروس : سیستم امکان نمایش لیست دروس و جزئیات هر درس را بر اساس نام دانشکده، نام درس، کد درس و حتی نام استاد درس فراهم خواهد کرد.

۵-۱۱ مشاهده برنامه هفتگی دروس : سیستم برای هر دانشجو بر اساس دروسی که در آنها ثبت نام کرده است برنامه هفتگی و تاریخ امتحانات را اعلان خواهد کرد.

۶. محدودیتها

در این قسمت محدودیتهای طراحی، محدودیتهای داخلی سیستم، محدودیتهایی که از محیط به سیستم اعمال می شود و وابستگیهای سیستم مشخص می شوند. برای نمونه در مورد سیستم ثبت نام محدودیتها به این صورت هستند که سیستم نباید به سخت افزار اضافه نیاز داشته باشد و با سخت افزار موجود قابل بهره وری باشد. اطلاعات دروس از بانک اطلاعاتی موجود باید اخذ شود و نیاز به یک بانک اطلاعاتی محزا نباشد.

۷. دامنه های کیفیت

در این قسمت دامنه های کیفی برای کیفیت، یکپارچگی، تحمل خطا، قابلیت استفاده و ویژگیهای مشابه دیگر، درسیستم مشخص می شود. رای نمونه در سیستم ثبت نام کیفیتها به صورت زیر مشخص شده است.

دسترسی : سیستم در تمام ساعتهای شبانه روز در دسترس خواهد بود.

بهره وری : سیستم برای افراد غیر کامپیوتری به سادگی قابل استفاده خواهد بود.

بهره وری : سیستم راهنمای برخط خواهد داشت به قسمیکه، هیچگونه نیازی به منوال استفاده نباشد.
قابلیت ترمیم : سیستم طوری طراحی خواهد شد که به سادگی قابل ترمیم باشد. همچنین، کلیه اطلاعات مربوط به کالج در جداول نگهداری خواهند شد تا به سادگی قابل مشاهده و تصحیح باشند.

۸. اولویتها

در این بخش برای هر یک از قابلیت‌های سیستم اولویت مشخص می شود. برای نمونه در مستند چشم انداز برای سیستم ثبت نام ابتدا مشخص شده است که در این بخش در ارتباط با اهمیت نسبی قابلیت‌های پیشنهاد شده برای سیستم مطالبی ارائه می شود. قابلیت‌های تعیین شده در مستند چشم انداز برای سیستم ثبت نام در نسخه ۱ و نسخه ۲ نرم افزار قابل دسترسی خواهند بود. همگام با پیشرفت پروژه برای هر یک از قابلیت‌های تعیین شده، کیفیت بررسی می شود. این بررسی بر اساس معیارهای قابلیت دسترسی، قابلیت بهره وری و قابلیت ترمیم که در بخش ۷ در بالا مطرح شد، انجام می شود. در نتیجه این بررسی برای هر یک از قابلیت‌های سیستم درجه اهمیت نسبی و اولویت مشخص خواهد شد. قابلیت‌های نسخه ۱ و ۲ نرم افزار به تفکیک در زیر مشخص شده است. این قابلیت‌ها در بخش ۱ توضیح داده شدند.

نسخه ۱	نسخه ۲
مجوز ورود به سیستم	• ثبت نمره دانشجویان
ثبت نام دروس	• مشاهده کارنامه دانشجویان
مشاهده جزئیات دروس	• انتخاب درس برای تدریس
درج جزئیات دانشجویان	
مشاهده مشخصات استادها	

۹. سایر انتظارات از محصول

در این قسمت به صورت کلی استانداردهای مورد استفاده، نیازهای سخت افزاری و ایستگاه‌های کاری، نیازهای کارایی و نیازهای محیطی مشخص می شوند. در ادامه این دسته از نیازها برای سیستم ثبت نام کالج مشخص شده است.

۱-۹ استانداردها : در این قسمت کلیه استانداردهای مورد استفاده مشخص می شود. این استانداردها برای نمونه می توانند در ارتباط با پروتکل‌های شبکه، سیستم‌های عامل، ایمنی و کیفیت باشد. برای نمونه در ارتباط با سیستم ثبت نام مشخص شده که رابط کاربر باید قابل نمایش بر روی ویندوز ۲۰۰۰ باشد.

۲-۹ نیازهای سیستمی :

در این قسمت هر گونه امکانات سخت افزاری، ارتباطی، شبکه، دستگاه های جانبی و نرم افزاری مورد نیاز سیستم را می توان مشخص نمود. برای نمونه در ارتباط با سیستم ثبت نام موارد زیر مشخص شده است:

سیستم باید با بانک اطلاعاتی موجود برای جزئیات دروس در ارتباط قرار گیرد. فرمت و قالب مشخص شده برای رکوردهای این بانک در مرجع شماره ۳ مشخص شده است.

سیستم باید با سیستم موجود حسابداری در ارتباط قرار گیرد. اطلاعات لازم در مورد سیستم حسابداری مرجع شماره ۲ مشخص شده است.

کامپیوتر سرویس دهنده موجود در دانشگاه با سیستم عامل یونیکس به عنوان سرویس دهنده سیستم ثبت نام باید استفاده شود.

کامپیوترهای سرویس گیرنده یا در اصطلاح کلاینت، هر یک از کامپیوترهای شخصی موجود در دانشگاه ممکن است باشد.

کامپیوترهای سرویس گیرنده نبتاید به بیش از ۳۲ مگابایت حافظه اصلی و ۲۰ مگابایت دیسک سخت نیاز داشته باشند.

سیستم عامل مورد استفاده برای سرویس گیرنده ها ویندوز ۲۰۰۰ خواهد بود.

۳-۹ نیازهای کارایی :

نیازهای کارایی دربرگیرنده قابلیت اطمینان، بار کاری سیستم بواسطه تعداد کاربرها، ظرفیت ارتباطی، سرعت انتقال داده ها، پهنای باند برای خطوط ارتباطی و میزان صحت عملکرد نرم افزار است. برای نمونه در سیستم ثبت نام موارد زیر مشخص شده است:

سیستم باید امکان دسترسی همزمان تا ۲۰۰۰ کاربر را به طور همزمان به بانک اطلاعاتی فراهم نماید.

همچنین سیستم باید قابلیت دسترسی همزمان ۵۰۰ نفر به ایستگاه سرویس دهنده را فراهم کند.

سیستم باید در حداکثر ۱۰ ثانیه به بانک اطلاعاتی جزئیات دروس دسترسی پیدا کند.

سیستم باید در حداکثر ۲ دقیقه ۸۰٪ از اطلاعات مورد نیاز را از بانک اطلاعاتی جزئیات دروس انتقال دهد.

۴-۹ نیازهای محیطی :

نیازهای محیطی در واقع مبین تاثیر محیط بر محصول نرم افزاری است. چنانچه نرم افزار درون سخت افزاری گنجانده ود، آنگاه عواملی مثل درجه حرارت، امواج الکترومغناطیسی موجود در

محیط، تشعشع و شک می تواند بر عملکرد آن تاثیر بگذارد. برای سایر نرم افزارها فاکتورهای محیطی شامل شرایط لازم برای استفاده از سیستم، در دسترس بودن منابع مورد نیاز در محیط کاربر، موارد ترمیم و تصحیح و بهبود از خطا، مطرح می باشد.

۱۰. مستندات

در این قسمت انواع مستنداتی که برای سیستم باید ایجاد و مشخص می گردد.

۱-۱۰ منوال کاربر : در این قسمت ساختار متوال یا در واقع دفترچه راهنمای کاربر مشخص می شود. باید عمق جزئیات مشخص شونده درون دفترچه یا منوال راهنمای کاری مشخص گردد. همچنین اندازه و تعداد صفحه های منوال مشخص گردد. به عنوان نمونه برای سیستم ثبت نام منوال کاربر استاده از سیستم را برای کارمند ثبت نام، دانشجو و استاد مشخص کرده است. این منوال باید شامل جزییات زیر باشد:

- ۱- حداقل امکانات سخت افزاری و نرم افزاری مورد نیاز
- ۲- طریق نصب نرم افزار بر روی ایستگاه کاری کاربر
- ۳- چگونگی ورود و خروج از سیستم
- ۴- قابلیت های سیستم
- ۵- اطلاعاتی برای پشتیبانی مشتری

مனால் باید بر طبق استاندارد مورد استفاده در کالج ایجاد شود. تعداد صفحه ها باید بین ۵۰ تا ۱۰۰ عدد باشد. ابعاد صفحه ها باید ۱۷،۵ در ۲۲،۵ سانتیمتر باشد. منوال کاربر باید به صورت فایل کامپیوتری تهیه شود و از طریق راهنمای برخطی که درون برنامه ایجاد شده قابل دسترسی باشد.

۲-۱۰ راهنمای برخط : در این قسمت معمولاً هدف از راهنمای برخط و ساختار محتویات آن مشخص می شود. برای نمونه در سیستم ثبت نام برای هر عملکرد سیستم باید راهنمای برخط ایجاد شود و تمام سرفصل های منوال کاربر را پوشاند.

۳-۱۰ راهنمای نصب و فایل های راهنما : در این قسمت ساختار منوال راهنمای نصب و یک فایل با نام مرابخوانید برای نصب نرم افزار آماده می شود. برای نمونه راهنمای نصب برای ایستگاه سرویس دهنده در سیستم ثبت نام باید شامل موارد زیر باشد:

- حداقل سخت افزار و نرم افزار مورد نیاز

- دستورالعملهای نصب
- پارامترهای نصب
- ایجاد بانک اطلاعاتی اولیه
- روش بازیابی بانک اطلاعاتی موجود در سیستم
- اطلاعات پشتیبانی مشتری
- روش سفارش برای توسعه نرم افزار

فایل مرا بخوانید برای سیستم ثبت نام می بایست شامل اطلاعاتی در مورد خطاهای موجود در نرم افزار و روش احراز از آنها و قابلیت‌های خاص نسخه موجود باشد.

۴-۱۰ برچسب گذاری و بسته بندی : در این بخش آرم شرکت کامپیوتری و یا آرم مشتری که در نرم افزار مورد استفاده قرار می گیرد، انحصاری بودن محصول و امتیاز استفاده از آن و کلیه استانداردهای گرافیکی خاص فرمهای ورودی / خروجی مشخص می شود. برای نمونه در سیستم ثبت نام آرم کالج باید در کلیه گزارشها و فرمهای ورودی / خروجی ظاهر شود.

الف. ویژگیهای قابلیت‌ها

فصل ۳

مراحل تحلیل و طراحی سیستم

در آر.یو.پی

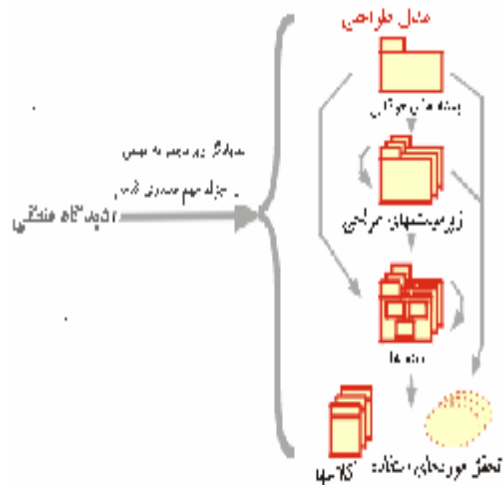
۳-۱ مقدمه

هدف از مرحله تحلیل تصویر اولیه نیازها به مدل‌های سیستم جدید مکانیزه است. این مدل‌ها در مرحله طراحی به فرم‌های قابل پیاده سازی تبدیل می‌شوند. معمولاً مدل‌های حاصل از آنالیز زمانی متفاوت با مدل‌های مرحله طراحی هستند که سیستم برای محیط‌های متفاوت با معماری‌های مختلف طراحی شود. در واقع مدل‌های حاصل از مرحله تحلیل بیانگر کلیت و چکیده ای از مدل‌های مرحله طراحی هستند. معمولاً هنگامی که پیاده سازی‌های متفاوت از یک سیستم مورد نیاز است کلیت طرح در قالب مدل آنالیز مشخص می‌شود. در غیر این صورت نیازی به مرحله جداگانه آنالیز نیست.

۳-۲ دیدگاه‌های معماری

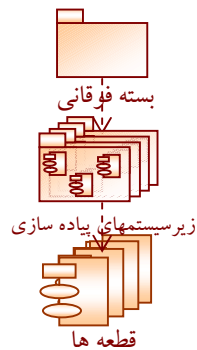
معماری نرم افزار در آر.یو.پی از طریق دیدگاه‌های تعیین شده در رشنال رز مشخص می‌شود. هر دیدگاه ویژگی‌های خاصی از نرم افزار را مشخص می‌کند. دیدگاه موردهای استفاده که شاخص موردهای استفاده و سناریو آنها است، در قالب مدل‌های موردهای استفاده، رفتار و عملکرد سیستم را به عنوان توصیفی برای معماری نرم افزار مشخص می‌کند. در دیدگاه منطقی کلاس‌های حاصل از مرحله طراحی در قالب بسته‌ها گروه بندی می‌شوند. بسته‌ها شاخص دسته بندی عملیات و زیر سیستم‌ها هستند. ممکن است هر بسته و یا مجموعه ای از بسته‌ها بر اساس نوع عملیات لایه بندی شوند. برای نمونه شکل زیر مدل منطقی سیستم را به صورت یک بسته فوقانی که به چند زیر سیستم تقسیم شده، مشخص نموده است. عملیات زیر سیستم به نوبه خود هر کدام به چند بسته تقسیم شده‌اند. هر کدام از این بسته‌ها توسط یک مدل ارتباطی کلاس‌ها مشخص شده‌اند و یا اینکه برای اینکه بتوان برای هر مورد استفاده چند طرح متفاوت ایجاد نمود، برای هر مورد استفاده چند تحقق مورد استفاده مشخص شده است. هر تحقق مورد استفاده در ارتباط از نوع تحقق یا در اصطلاح

<<realization>> با یک مورد استفاده قرار می گیرد. برای یک مورد استفاده ممکن است چند تحقق وجود داشته باشد. تحقق مورد استفاده با بیضی خط چین شده، مشخص می شود.



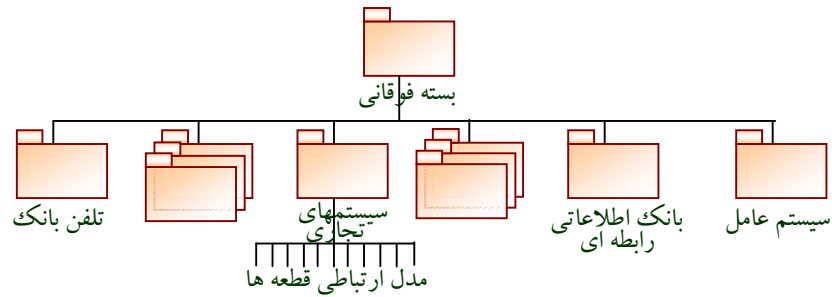
شکل ۱-۳ مدل طراحی بر اساس اجزاء مهم معماری

دیدگاه پیاده سازی، معماری کد برنامه را در قالب پیمانه ها مشخص می کند. در واقع ساختار برنامه ها در این دیدگاه در قالب قطعه های^۳ کد اولیه و قطعه های کد اجرایی و رابطه بین آنها مشخص می شود. مدل ارتباطی قطعات در دیدگاه قطعه ها^۴ در محیط رشنال رز مشخص می شود. همانگونه که در شکل ۲-۳ مشخص شده است، در بالاترین سطح در از مدل پیاده سازی برای معماری یک بسته قرار گرفته است. هر زیر سیستم مجموعه ای از قطعه ها و زیر سیستمهای دیگر است. هر قطعه ای تنها در یک زیر سیستم قرار می گیرد.



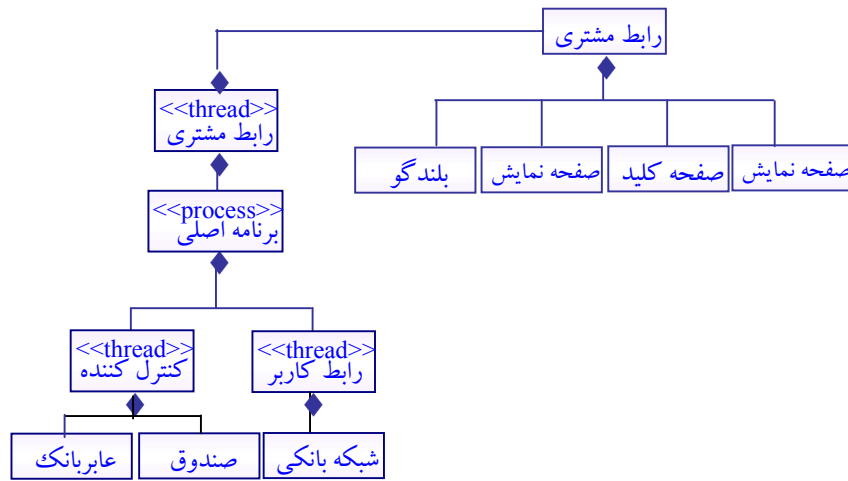
شکل ۲-۳ طرح کلی مدل معماری برای برنامه ها

برای نمونه در شکل زیر ساختار معماری برای برنامه یک سیستم بانکی در اولین عمق مشخص شده است.



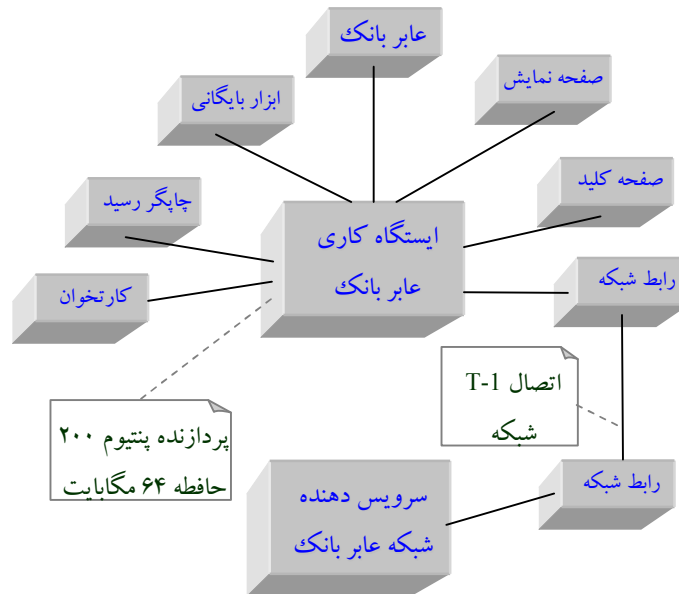
شکل ۳-۳ نمونه ای از مدل معماری پیاده سازی

جهت تعیین همروندی در عملیات سیستم دیدگاه فرایند مورد استفاده قرار می گیرد. دیدگاه فرایند شاخص توصیف وظایف ریسمانها و پردازش های همروند و در مجموع هر گونه همروندی در اجرای برنامه است. برای نمونه در شکل زیر با ایجاد یک مدل، فرایند همروندی در عملیات ماشین عابر بانک مشخص شده است.



شکل ۳-۴ تصویر کلاسها به مدل فرایند برای ماشین عابر بانک

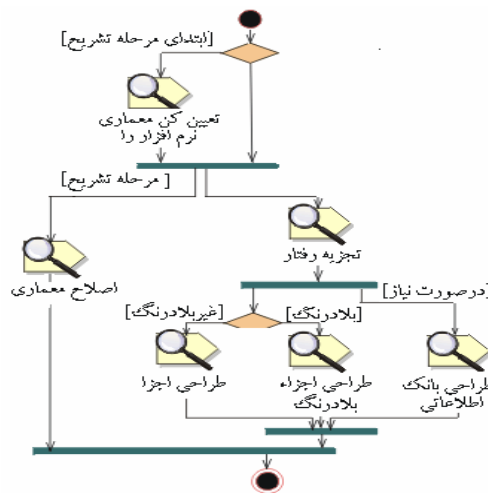
دیدگاه نصب^۵، نمایانگر ساختار فیزیکی معماری سیستم است. دستگاه های جانبی، ایستگاه های کاری و توزیع برنامه ها در سطح شبکه از طریق این دیدگاه انجام می شود. برای نمونه به شکل زیر توجه نمایید. در این شکل ساختار معماری فیزیکی شبکه عابر بانک ارائه شده است.



شکل ۳-۵ دیدگاه نصب نمایانگر توزیع فیزیکی فرایندها در سیستم

۲-۳ مراحل

مراحل تحلیل و طراحی در شکل ۳-۱ مشخص شده است. در اولین گامها از مرحله تشریح، معماری نرم افزار را باید مشخص نمود.



شکل ۳-۶ مراحل طراحی نرم افزار

اجزاء معماری را در برنامه های شیئی گرا کلاسها تشکیل می دهند. ارتباط و همکاری بین کلاسها بر

مبنای دیاگرامهای همکاری مشخص می شود. دیاگرامهای همکاری نیز بر اساس توصیف موردی استفاده مشخص می شوند. در این مرحله عملیات تحلیل معماری و تحلیل موردی استفاده مکرراً انجام می شود.

با تحلیل موردی استفاده اجزاء تشکیل دهنده معماری، کلاسها مشخص می شوند. بعضاً نیاز می باشد که جهت تعیین کلاسها، سناریو موردی استفاده اصلاح شود تا کلیه اشیاء مشخص شوند. با تعیین دیاگرام همکاری برای موردی استفاده می توان کلاسها و مدل ارتباطی آنها را مشخص نمود. برای اینکه بتوان برای یک مورد استفاده طراحی های متفاوت انجام داد از نوعی خاص موردی استفاده استفاده به نام تحقق موردی استفاده، بکار برده می شوند. به این ترتیب برای هر مورد استفاده ممکن است چند تحقق موردی استفاده ایجاد شود. برای نمونه به شکل زیر توجه کنید. در این شکل یک مورد استفاده و تحقق آن در حالت کلی مشخص شده است.



شکل ۳-۷ ارتباط مدل موردی استفاده با مدل طراحی

در مرحله طراحی برای به واقعیت پیوستن و یا به عبارت دیگر تحقق هر مورد استفاده دیاگرامهای توالی، همکاری و مدل ارتباطی کلاسها ممکن است ایجاد شود.

پس از تعیین معماری نرم افزار باید اطمینان حاصل نمود که عناصر تکراری با وظایف یکسان وجود ندارند. برای این منظور به بررسی مجدد و اصلاح معماری باید پرداخت. با پیشرفت طراحی عملیات توزیع شونده در سطح شبکه و همروندی در عملیات به مرور مشخص می شوند. در شکل زیر فعالیتهای لازم برای تحلیل و طراحی نرم افزار مشخص شده است:



شکل ۳-۸ فعالیتهای لازم برای تحلیل و طراحی

با در نظر گرفتن دیاگرام گردش کار برای مرحله تحلیل و طراحی که در بالا ارائه شد برای مراحل به شرح زیر می باشد. البته باید توجه نمایید که بر طبق دیاگرام گردش کار این فعالیتها ممکن است تکرار شوند و یا اینکه بطور همزمان اجرا شوند. این مراحل در ادامه توضیح داده خواهند شد.

۱- تعیین معماری کاندید

۲- اصلاح معماری

۳- تحلیل رفتار

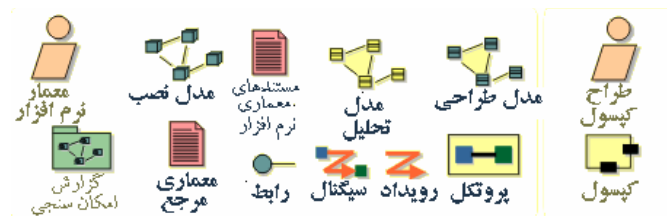
۴- طراحی قطعه ها یا پیمانانه های برنامه

۵- ترکیب معماری

۳-۲-۱ تعیین معماری کاندید

۳-۲-۱ طرح معماری نرم افزار

بر اساس دیاگرام گردش عملیات تحلیل و طراحی و دیاگرام فعالیتها، ابتدا باید به تحلیل و طراحی معماری سیستم نرم افزاری پرداخت. هدف از این فعالیت تعیین ساختار کلی نرم افزار است. برای هر یک از دیدگاه های منطقی، پیاده سازی، نصب و فرایند، دیاگرامهای جداگانه ایجاد می شود. دیدگاه منطقی از لحاظ طراحی منطقی، دیدگاه پیاده سازی از لحاظ ساختار کلی قطعه های کد ایجاد شونده، دیدگاه نصب از ساختار معماری فیزیکی و دیدگاه فرایند در صورت نیاز ساختار فرایندهای موازی و همروند را مشخص می کند. در شکل زیر خروجیهای مرحله تحلیل و طرح معماری مشخص شده است.



شکل ۳-۹ خروجیهای حاصل از فعالیتهای معمار نرم افزار

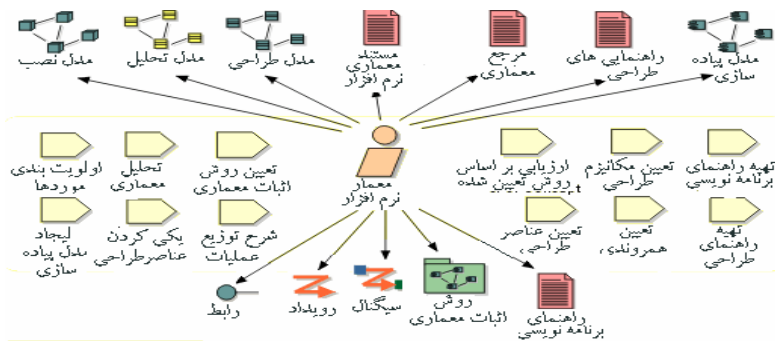
در شکل فوق گزارش امکان سنجی، در ارتباط با تعیین میزان ریسک و هزینه های تخمینی برای تبدیل نیازها در قالب معماری نرم افزار مشخص می شود. مدل نصب، مدل فیزیکی معماری نرم افزار

است که در دیدگاه نصب مشخص می شود. منظور از معماری مرجع، الگوها و یا معماریهای قبلی است که مجدداً مورد استفاده قرار می گیرند.

مدل تحلیل، مدل ارتباطی کلاسهایی است که از مطالعه موردهای استفاده و نیازهای مرتبط با هر مورد استفاده و جهت تحقق موردهای استفاده ایجاد می شود. این مدل انتخابی و موقتی است و در قالب مدل طراحی تکمیل می شود. مدل طراحی پیاده سازی مدل موردهای استفاده را مشخص می کند. مدل طراحی تبدیل به مدل پیاده سازی می شود. برای این منظور چگونگی تویر مدل به زیر سیستمها و قطعه های پیاده سازی باید مشخص شود.

سیگنال برای هماهنگ کردن فرایندهای همروند مورد استفاده قرار می گیرد. در دیاگرام حالت یک سیگنال موجب گذر از یک حالت به حالت دیگر می شود. با استفاده از نوع <<signal>> می توان کلاسها را از نوع سیگنال مشخص کرد. رویدادها معمولاً برای تعیین حادته های خارج از سیستم که موثر بر سیستم هستند را مشخص می کند. در ادامه هر یک از فعالیتهای مشخص شده در دیاگرام فعالیتهای، توضیح داده خواهد شد.

در شکل زیر لیست فعالیتهای مرحله طراحی معماری مشخص شده است. این فعالیتهای در ادامه توضیح داده خواهند شد.



شکل ۳-۱۰ فعالیتهای تحلیل و طراحی معماری

- اولویت بندی موردهای استفاده

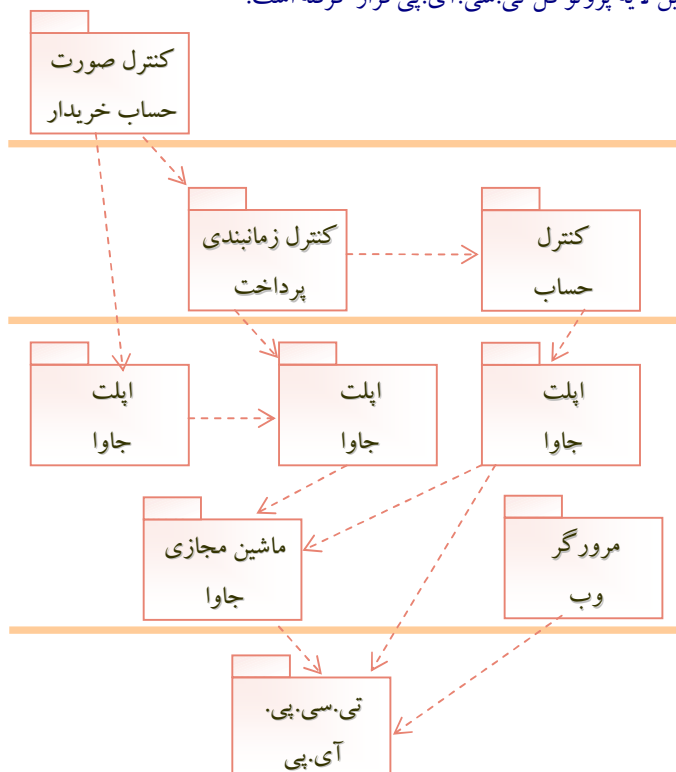
برای تحلیل و طراحی در هر تکرار از مراحل چرخه حیات، می بایست تعدادی از موردهای استفاده انتخاب شوند. انتخاب موردهای استفاده بر اساس عواملی مثل میزان اهمیت از دیدگاه کاربر، تاثیر مورد استفاده بر معماری نرم افزار، تقلیل ریسک با انتخاب صحیح موردهای استفاده و بالاخره حصول اطمینان از اینکه با انتخاب مناسب و تدریجی موردهای استفاده در خاتمه مرحله تشریح از

چرخه حیات طرح نرم افزار آماده برای مرحله پیاده سازی خواهد بود، انجام می شود. مدل موردهای استفاده در بخشی از مستند معماری نرم افزار تشریح می شود.

- تحلیل معماری

پس از تعیین اولویت برای موردهای استفاده می توان به تحلیل معماری پرداخت. هدف از تحلیل معماری ارائه ساختار اولیه برای نرم افزار است. برای این منظور ابتدا در بالاترین سطح باید زیر سیستمها را مشخص نمود. طرح معماری معمولاً لایه ای است. در هنگام تحلیل، امکان استفاده مجدد از کد و طرح برنامه را باید مد نظر داشت. در صورت وجود الگوی طراحی مناسب می توان طرح معماری را بر مبنای یک الگوی شناخته شده ایجاد کرد. باید برای تحلیل معماری، نیازهای کاربر و محدودیتها و امکانات سخت افزاری و نرم افزاری موجود را مشخص کرد. معمولاً در مرحله شناخت جهت تکمیل چشم انداز پروژه شمایی از ساختار و معماری کلی در سطح منطقی، کد برنامه ها و طرح فیزیکی ارائه می شود. این طرحهای اولیه مبنایی برای تحلیل و طرح معماری نرم افزار می شوند.

با در نظر گرفتن امکانات، سازماندهی زیر سیستمها در سطوح بالا مشخص می شود. البته، معمولاً معماری لایه ای برای زیر سیستمها در نظر گرفته می شود. لایه بندی نرم افزار در واقع ابزاری برای تقلیل تاثیر تغییرات در برنامه ها می باشد. با افزایش میزان پیچیدگی سیستمها، تعداد لایه ها افزایش می یابد. هر لایه را معمولاً برای یک تیم مجزا مشخص می کند. برای نمونه به شکل زیر توجه نمایید. در این شکل یک لایه ویژه برنامه کاربردی خاص در نظر گرفته شده است. لایه زیرین آن مستقل از لایه فوقانی ایجاد شده و شامل دو بسته عمومی است که لایه فوقانی از آن استفاده می کند. لایه میان افزار، شامل اپلتهای جاوا و امکاناتی فراخوانی از راه دور در جاوا تحت عنوان آ.ام.آی است. در پایینترین لایه پروتوکول تی.سی.پی.آی قرار گرفته است.



شکل ۳-۱۰ نمونه ای از طرح لایه ای برای برنامه های جاوا تحت وب

پس از تعیین زیر سیستمها و در واقع ساختار کلی طرح معماری، می بایست مدل ارتباطی کلاسها را مشخص نمود. مدل ارتباطی کلاسها یک خلاصه و چکیده از ساختار سیستم است. همانگونه که در بالا مشخص شده است، طرح معماری نرم افزار شاخص یک خلاصه سازی یا در واقع تجریدی از نرم افزار است. در این راستا الگوهای طراحی که نمایانگر معماریهای شناخته شده و مدل ارتباطی کلاسها هستند را می توان به عنوان ساختار چکیده ای برای معماری، طراحی و پیاده سازی نرم افزار، مورد استفاده قرار داد. گروه او.ام.جی تعداد زیادی رابطهای کلی و پروتکلهای ارتباطی را برای ایجاد نرم افزار مشخص کرده است [۲].

در این مرحله مدل ارتباطی کلاسها به عنوان عناصر تشکیل دهنده اجزاء معماری مشخص می شود. در رشنال رز طرح معماری نرم افزار در دیدگاه منطقی^۶ مشخص می شود. البته، کلاسهای حاصل از تحلیل نرم افزار، ممکن است که در مرحله پیاده سازی تغییر کنند. مدل ارتباطی کلاسها در واقع نمایانگر چگونگی تحقق موردهای استفاده است. از آنجا که برای تحقق یک مورد استفاده ممکن است چند روش مختلف انتخاب شود، با استفاده از نماد تحقق مورد استفاده، یک مورد را به چند طریق محقق می کنند. باید توجه نمایید که کلاسهای حاصل از مرحله تحلیل ممکن است در طی مراحل طراحی دستخوش تغییرات زیادی شوند. لذا، نیازی به ذکر جزئیات کلاسها در این مرحله نیست.

پس از چکیده سازی سیستم در قالب ساختار سلسله مراتبی زیر سیستمها و مدل ارتباطی کلاسها، می توان ساختار فیزیکی سیستم را در قالب مدل نصب مشخص نمود. ساختار فیزیکی با توجه به مکان فیزیکی کاربرها و موردهای استفاده آنها از سیستم مشخص شود. مسلماً برای استفاده از سیستم

در ساختار فیزیکی باید قطعات مربوطه کد برای دسترسی کاربر گنجانده شود. این اولین مدل نصبی است که در پروژه ایجاد می شود و بعداً در طول مراحل طراحی و پیاده سازی تصحیح می شود.

- تعیین روش اثبات صحت معماری

معماری نرم افزار بر اساس نیازهای کاربر مشخص می شود. لذا، باید انواع معماریهای نرم افزار را شناخت تا بتوان بر مبنای آن معماری مناسب را انتخاب کرد. برای نمونه ممکن است کاربر بخواهد که برای هر واحد برنامه ای جداگانه ایجاد شود. ممکن است برای کلیه واحدها تنها یک برنامه ایجاد شود و در واحدهای مختلف کپی های آن نصب شود. ارتباط بین واحدها ممکن است از طریق یک بانک اطلاعاتی متمرکز و یا از طریق فراخوانی از راه دور انجام گیرد. بنابراین باید بر اساس نیاز، معماری تعیین شود. باید نیازهای موثر در معماری را مشخص و امکان رسیدن به آنرا مورد سنجش قرار داد. ممکن است برای نمونه از لحاظ تکنیکی نتوان نیازهای مطرح شده را برآورده نمود. برای این منظور باید معیارهایی جهت امکان سنجی مشخص نمود.

پس از تحلیل معماری می توان برای حصول اطمینان از امکان ایجاد معماری نرم افزار مبتنی بر نیازهای مطرح شده با تکنیکهایی از قبیل نمونه سازی سریع، شبیه سازی، مدل مفهومی، تبدیل اتوماتیک طرح به کد و یا استفاده از نوع خاص توصیفهای معماری که قابل اجرا است، به بررسی بیشتر در مورد امکانپذیری نرم افزار پرداخت.

ریسک لیست یکی از خروجیهای این مرحله از تحلیل است. در این لیست ریسکها و عوامل عدم موفقیت پروژه در چارچوب تعیین شده مشخص می شود. برای نمونه به مثال زیر توجه کنید. شرح کاملتر در مبحث ریسک ارائه خواهد شد. در این مثال درجه نسبی اهمیت ریسک هشت می باشد. و برای مقابله با ریسک عدم کارایی سیستم ثبت نام، نمونه برداری سریع برای ارزیابی کارایی پیشنهاد شده است.

اهمیت	توصیف و تاثیر ریسک	برنامه ریزی برای تقلیل یا مقابله
۸	با ازدیاد تعداد دانشجویهای ثبت نام کننده ای که بطور همزمان به سیستم می خواهند دسترسی پیدا کنند، کارایی ممکن است بسیار کم شود.	با استفاده از تکنیکهای نمونه برداری سریع باید کارایی ارزیابی شود.

محصول دیگر مرحله بررسی و امکان سنجی نیازها، مستندی تحت عنوان مورد کاری^۷ است. این مستند، گزارش سود و زیان حاصل از پروژه است. در این گزارش در ضمن مشخص نمودن بازار محصول، استفاده کننده ها مشخص می شوند. اگر مستند به عنوان متن قرار داد مورد استفاده می

شود، شرایط قرار داد نیز باید در آن ذکر شود. در ضمن در این مستند معمولاً هدف، گزینه های مختلف برای محصول و محدودیتها و قیودات برای ایجاد آن مشخص می شود. مستند چشم انداز نیز حاصل این مرحله است.

- تعیین مکانیزم طراحی

مکانیزم طراحی، فرم اصلاح شده مکانیزم تحلیل است. مکانیزم تحلیل به الگویی اطلاق می شود که راه حلی برای یک مساله مشترک ارائه می دهد. این الگو ممکن است ساختاری، رفتاری و یا ترکیبی از این دو باشد. مسلماً با تعیین مسائل مشترک از پیچیدگی تحلیل کاسته می شود. معمولاً، مکانیزمهای تحلیل مسائل عمومی و در حالت کلی مفاهیم علوم نرم افزار هستند لذا، در لایه های میانی و تحتانی قرار می گیرند. مکانیزمهای تحلیل اغلب در ارتباط مستقیم با اهداف پروژه نیستند. بلکه در ارتباط با رفتار خاص و یا پیاده سازی همکای بین کلاسها و یا قطعات نرم افزار هستند. برای نمونه شرایط ارتباط بین فرایندها در سطح شبکه را می توان در قالب سرعت ارتباط، همروندی، اندازه پیامها، پروتکل و بافر کردن اطلاعات، مشخص نمود. پس از تعیین مکانیزمهای تحلیل باید مشخص کرد که چه کلاسها، قطعه ها و زیر سیستمهایی با آنها مرتبط می شوند. برای تعیین عملکرد مکانیزمهای تحلیل از دیاگرامهای همکاری استفاده می شود.

مکانیزم طراحی جزئیات بیشتری به مکانیزم تحلیل اضافه می کند. مکانیزمهای پیاده سازی روش خاصی برای پیاده سازی مکانیزم طراحی مثل استفاده از زبان برنامه سازی و یا محصول نرم افزاری خاصی را مشخص می کند. برای نمونه ممکن است مکانیزم پیاده سازی در قالب الگوی پیاده سازی شناخته شده ای مطرح شود. برای نمونه چنانچه مکانیزم تحلیل در مورد پایداری اشیاء در حافظه به شرح زیر مشخص شده باشد:

§ ممکن است نیاز باشد که در حدود ۲۰۰۰ شیئی کوچک که هر کدام در حدود ۲۰۰ بایت هستند برای چند ثانیه فقط نگهداری شوند.

§ ممکن است که نیاز باشد تا چند شیئی بزرگ برای چند ماه بر روی حافظه دیسک نگهداری شوند بدون اینکه این اشیاء هیچگونه تغییری بکنند. البته سرعت دسترسی به این اشیاء باید بسیار سریع باشد.

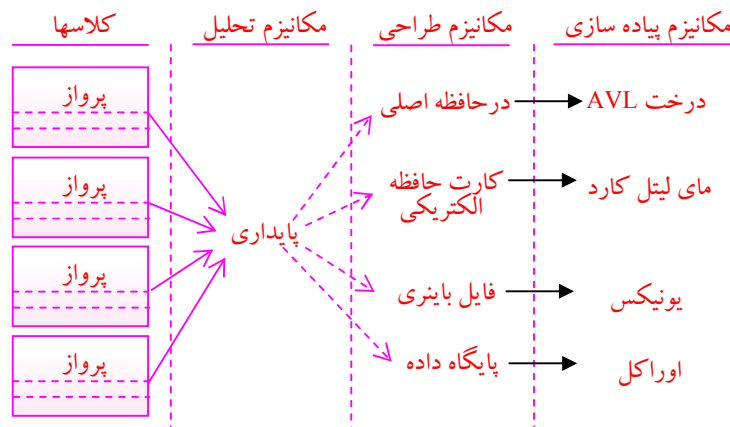
در مرحله طراحی جزئیات زیر به مکانیزمهای فوق برای پایداری اشیاء، افزوده می شود:

§ نگهداری در حافظه اصلی سرعت دسترسی را بسیار سریع می کند، اما فضای تخصیصی از حافظه اصلی محدود به یک مگابایت برای برنامه است.

§ کارت حافظه الکتریکی تا هشت مگابایت، با سرعت کم برای خواندن متوسط و نوشتن و اصلاح

§ فایل‌های باینری از ۱۰۰ کیلو بایت تا ۲۰۰ مگا بایت با سرعت دسترسی کم

§ مدیریت بانک اطلاعاتی برای نگهداری صد کیلو بایت با امکان بروز رسانی و سرعت دسترسی کم. باید توجه نمایید که سرعت دسترسی با توجه به سرعت دسترسی به حافظه اصلی کم است. البته با ایجاد تکنیکهای استفاده از حافظه سریع کش می توان تا حدی سرعت دسترسی را افزایش داد. اکنون می توان مکانیزم پایداری را برای کلاسها مطابق شکل زیر مشخص نمود.



شکل ۳-۱۱ ارتباط کلاسها با مکانیزم پایداری

در شکل فوق جزئیات یک پرواز ممکن است در حافظه اصلی برای دسترسی سریع و در یک پایگاه داده برای مدت طولانی نگهداری شود. به این ترتیب نیاز به مکانیزمی برای هماهنگی بین داده های ذخیره شده بر روی دیسک با داده های درون حافظه اصلی وجود داد. در حالت کلی برخی مکانیزمها ممکن است وابسته به مکانیزمهای دیگر باشند. برای نمونه مکانیزم گزارش خطا ممکن است وابسته به مکانیزم پایداری باشد و یا اینکه مکانیزم هشداردهی ممکن است وابسته به مکانیزم ارتباطی بین فرایندها باشد.

روش و مکانیزم طراحی در واقع بیانگر چگونگی تصویر نتایج حاصل از مرحله تحلیل است. این مکانیزمها به صورت بالا به پایین و یا پایین به بالا مشخص می شوند. حاصل تحلیل مدلهایی است که مشخص می کنند چه باید انجام شود. چگونگی انجام امور در مرحله طراحی مشخص می شود. اما، محدودیتهای پیاده سازی موجب می شود که تغییراتی در حاصل تحلیل ایجاد شود. باید امکانهایی موجود برای پیاده سازی نرم افزار را بررسی کرد تا بر مبنای آن طرح واقع گرایانه شود. برای نمونه ممکن است بر اساس پارامترهای تعیین شده امکانات کوربا در مرحله طراحی مناسب دیده شود. که این انتخاب اصلاحاتی را در نتایج حاصل از تحلیل ایجاد می کند. مکانیزمهای پیاده سازی در ارتباط با میان افزارها مثل ماشین مجازی جاوا، سیستمهای عامل مثل لینوکس، قطعه های مورد استفاده مثل اکتیویاکس، کتابخانه کلاسها و بسته های نرم افزاری مثل بسته های گرافیکی و بسته های بانک

اطلاعاتی، همگی محدودیتهایی بر طرح منطقی حاصل از تحلیل قرار داده موجب اصلاح و تغییراتی در آن می شوند. باید هزینه ها را در هنگام انتخاب این مکانیزمها نیز مد نظر داشت.

- تهیه راهنمای برنامه نویسی

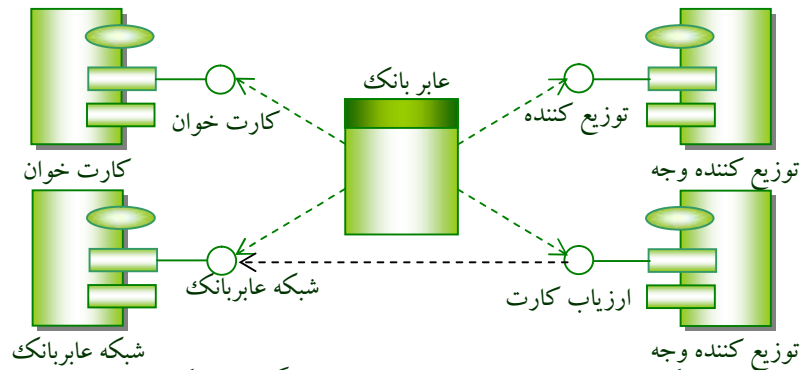
باید روشی کلی برای ایجاد کد برنامه ها مشخص شود تا کلیه برنامه ها شکل یکسان داشته باشند. برای نمونه پیشنهاد می شو که اسامی متغیرها گویا باشد. برای نمونه firstName و corporateCustomer دو اسم گویا هستند. از اسامی مختصر مثل ii و jz باید خود داری شود. از اسامی و اصطلاحاتی که کاربرها استفاده می کنند، استفاده شود. برای نمونه اگر لفظ تعرفه برای فرم ثبت نام بکار برده می شود، در داخل برنامه هم از نام تعرفه استفاده شود. اسامی متغیرها را با حرف کوچک و اسامی کلاسها و رابطها را با حرف بزرگ مشخص کنید. برای خلاصه کردن اسامی استاندارد و قانونی مشخص کنید تا در مواردی که اسامی بیش از ۱۵ حرف می شوند بتوانید آنها را خلاصه کنید. از علامت Underscore در ابتدا و انتهای اسامی خودداری نمایید.

هر بخش برنامه ابتدا با جمله های تفسیری کامنت آغاز گردد و سپس با کد برنامه پر شود. کد باید گویا و خواندنی باشد. قرار دادن کامنتها درون چهار چوب مربعی کار درستی نیست و گویایی کد را کم می کند. بنابراین کامنتها باید بسیار ساده ظاهر شوند. برای کد خود ابتدا مستندات ایجاد کنید. اگر کدی ارزش مستند کردن نداشته باشد، مطمئناً ارزش اجرا کردن هم نخواهد داشت. قبل از هر تابع، یا هر رابط و فیلدی اگر قرار است کامنتی قرار داده شود. یک کامنت چند سطری باشد و کامنتهای یک سطری به صورت متوالی قرار داده نشوند.

- ایجاد ساختار اولیه پیاده سازی

برای انتقال از مرحله طراحی به مرحله پیاده سازی می ایست ساختار مدلهای طراحی را به پیاده سازی تصویر نمود. بسته های طراحی می بایست زیر سیستمهای مشابه پیاده سازی را داشته باشند. اجزا معماری در مرحله پیاده سازی را قطعه ها یا در اصطلاح کامپوننتها تشکیل می دهند. باید توجه داشته باشید که در واقع قطعه ها پیمانها را مشخص می کنند. نکته قابل توجه در اینجا است که در یک زبان مثل ویژوال بیسیک یک کلاس درون قطعه ای مجزا از نوع فرم، پیمانها کلاس یا کلاس ماجول یا کنترل قرار می گیرد. در زبان جاوا و C++ یک کلاس طراحی مشابه یک یا چند کلاس پیاده سازی است و چند کلاس ممکن است در یک پیمانها قرار گیرند. در برخی از زبان مثل جاوا و سی امکان مجزا کردن توصیف کلاس از پیاده سازی آن وجود دارد. برای نمایش مدل پیاده سازی

از دیاگرام قطعه‌ها استفاده می‌شود. برای نمونه در شکل زیر دیاگرام قطعه‌ها برای یک دستگاه عابر بانک مشخص شده است.



شکل ۳-۱۲ مدل معماری پیاده سازی برای دستگاه عابر بانک

- یکی کردن عناصر طراحی

هدف از یکی کردن عناصر طراحی، تحلیل همکاری کلاسها با یکدیگر جهت یافتن رابطهای کاربر، کلاسهای طراحی و زیر سیستمهای طراحی است. به این ترتیب طرح معماری نرم افزار در دیدگاه منطقی توسط مدل ارتباطی کلاسها، بسته‌ها، و زیر سیستمها و در مجموع مدل طراحی مشخص می‌شود.

ابتدا باید با تعیین تشابه رابطها برای زیر سیستمهای حاصل از مرحله تحلیل، در صورت امکان رابطهای مشابه را یکسان نمود. به این ترتیب می‌توان زیر سیستمها با رابطهای مشابه را درون قطعه‌هایی که برای ایجاد معماری کد برنامه‌ها استفاده می‌شود، قرار داد. برای هر قطعه می‌توان در دیدگاه قطعه‌ها یک رابط مشخص نمود.

مقوله استفاده مجدد در سطوح طراحی نیز مطرح است. در صورت امکان، طرح معماری موجود را می‌توان برای استفاد مجدد مورد استفاده قرار داد. با استفاده از ابزار مهندسی معکوس می‌توان طرح نرم افزارهای موجود را استخراج و در صورت امکان بخشهایی از آنرا مورد استفاده مجدد قرار داد.

با افزایش عناصر جدید به معماری سیستم در مرحله طراحی، لازم می‌شود که دوباره اجزاء در بسته‌های جدید تر قرار داده شوند. باید کلاسهای درون هر بسته حداکثر میزان چسبندگی را داشته باشند و میزان ارتباط بین کلاسهای داخل هر بسته حداکثر ممکن باشد. با افزایش میزان وابستگی نیلز به لایه بندی طرح معماری نیز افزایش می‌یابد. معمولاً، در هنگام لایه بندی، لایه فوقانی خاص برنامه، لایه زیرین آن کلی تر و لایه تحتانی در ارتباط با محیط اجرایی برنامه است. برای نمونه در طرح چهار لایه به ترتیب:

- ۱- بالاترین سطح، تحت عنوان لایه کاربرد، خاص برنامه مورد نظر
 - ۲- لایه دوم، تحت عنوان لایه کاری، شامل قطعات مورد استفاده برای چند کاربرد
 - ۳- لایه سوم، تحت عنوان میان افزار، شامل قطعات گرافیکی، رابط پایگاه داده، سرویسهای سیستم عاملی مستقل از ایستگاه کاری، قطعه های OLE و کلا" هر گونه محیطی ک" مستقیما" در ارتباط با سیستم عامل است می باشد.
 - ۴- لایه چهارم، تحت عنوان لایه سیستم، شامل قطعاتی مثل سیستم عامل، پایگاه داده و رابطهای سخت افزاری است.
- اصولا"، هر لایه مستقل از لایه فوقانی ایجاد می شود. لایه فوقانی مبتنی بر لایه زیرین و عناصر درون لایه خود ایجاد می شوند. تعداد لایه ها بطور معمول و برای سیستمهای کوچک ۳ عدد می باشد. برای سیستمهای بزرگ تعداد لایه های بین ۵ تا ۷ عدد می باشد. تعداد بیش از ۱۰ لایه ماملا" غیر قابل قبول است. جدول شکل زیر شاخص تعداد لایه ها بر اساس تعداد کلاسها است:

تعداد کلاسها	تعداد لایه ها
۱۰-۰	بدون لایه
۵۰-۱۰	۲ لایه
۱۵۰-۲۵	۳ لایه
۱۰۰۰-۱۰۰	۴ لایه

شکل ۳-۱۱ رابطه بین کلاسها و تعداد لایه ها

لایه بندی باید موجب شود که ارتباط وابستگی بین زیرسیستمها به حداقل ممکن تقلیل یابد. در لایه های فوقانی ممکن است افزاز های دیگری هم ایجاد شود. این افزازها ممکن است بر اساس معیارهای زیر ایجاد شود.

- ۱- خطوط عملیاتی سازمانی و واحدهای سازمانی می توانند معیاری برای افزاز لایه های فوقانی معماری نرم افزاز باشند. البته معمولا"، ساختار سازمانی در حال تحول و تغییرات است و بهتر است که مستقل از آن عمل کرد.
- ۲- معمولا"، واحدهای تخصصی در یک سازمان به عنوان واحدهای سرویس دهنده توسط سایر واحدها مورد استفاده قرار می گیرند. لذا، بر مبنای این واحدها می توان لایه های میانی را افزاز نمود.
- ۳- بر اساس توزیع عملیات می توان هر لایه را نیز به چند افزاز تقسیم بندی نمود. به این ترتیب می توان طرح شبکه را نیز در داخل معماری نرم افزاز مشاهده نمود.

۴- ممکن است بر اساس امنیت و کنترل دسترسی لایه ها افزای شوند و یا اینکه زیر سیستمهای مستقل با رابطهای خاص جهت کنترل دسترسی و امنیت سیستمها ایجاد شود. معماری طراحی شده نرم افزار در دیدگاه منطقی در محیط رشنال رز گنجانده می شود.

- تعیین عناصر طراحی

تحلیل موردها استفاده موجب آشکار شدن کلاسهای مرحله طراحی می شود. در مرحله طراحی کلاسهای جاصل از مرحله تحلیل تبدیل به عناصر مرحله طراحی شامل:

- کلاسها که نمایانگر مجموعه ای از مسئولیتها هستند

- زیر سیستمها که مجموعه های بزرگتری از مسئولیتها را مشخص می کنند.

- کلاسهای فعال، برای نمایش ریسمانهای کنترل کننده سیستمهای همروند

- رابطها بری نمایش چکیده مسئولیتهای کلاسها یا زیرسیستمها

علاوه بر این باید عناصر زیر نیز در این مرحله مشخص شوند:

- رویدادها که بیانگر اتفاقاتی هستند که بر سیستم تاثیر گذاشته و برای آنها واکنشی وجود دارد.

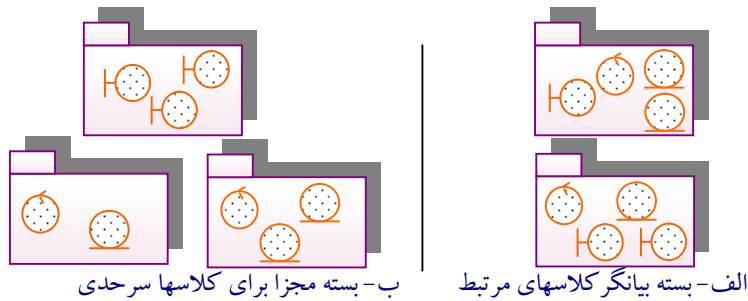
- سیگنالها که مکانیزمهای ناهماهنگی را که با یکدیگر درون سیستم با ایجاد رویدادها ارتباط برقرار می کنند را مشخص می کند

رویدادها را می توان از درون دیاگرامهای حالت و دیاگرامهای توالی استخراج نمود. سیگنال یک موجودیتی است با نام مشخص که برای ارتباطات ناهماهنگ در بین اشیاء ایجاد می شود. هر سیگنالی لیست مشخصی از پارامترها دارد.

هنگامی که کلاسهای مرحله طراحی مشخص شدند، باید آنها را بر اساس میزان ارتباط دسته بندی کرد. معمولاً بسته مستقل از یکدیگر می باشند. زیر سیستمهای طراحی نیز به صورت بسته مشخص می شوند. اینها بر خلاف بسته ها لزوماً دارای یک یا چند رابط هستند. نمایانگر یک واحد رفتاری می باشند. به عبارت دیگر ارتباط چند کلاس جهت هدف خاصی را مشخص می کنند. معمولاً زیر سیستمها در ارتباط یک به یک با قطعه های اجرایی کد که در مدل پیاده سازی مشخص می شوند دارند.

کلاسهایی که در داخل یک بسته مشخص می شوند عمومی یا در اصطلاح پابلیک و یا خصوصی یا در اصطلاح پرایویت هستند. کلاسهای عمومی در واقع رابط بسته با خارج از آن هستند. در صورتیکه امکان تغییر رابطهای سیستم در آینده زیاد باشد، بهتر است که کلاسهای سرحدی درون یک بسته مجزا قرار داده شوند. در غیر این صورت کلاسهای سرحدی همراه با کلاسهای کنترلی و

کلاسهای موجودیتی که در مجموع یک وظیفه خاصی را انجام می دهند در یک بسته قرار می گیرند. برای نمونه به شکل زیر توجه نمایید:

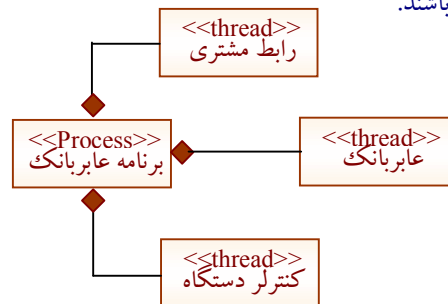


شکل ۳-۱۲ روشهای دسته بندی کلاسها

باید میزان چسبندگی بین کلاسهای درون هر بسته ماکزیمم و میزان ارتباط بین کلاسها به حداقل ممکن تقلیل داده شود.

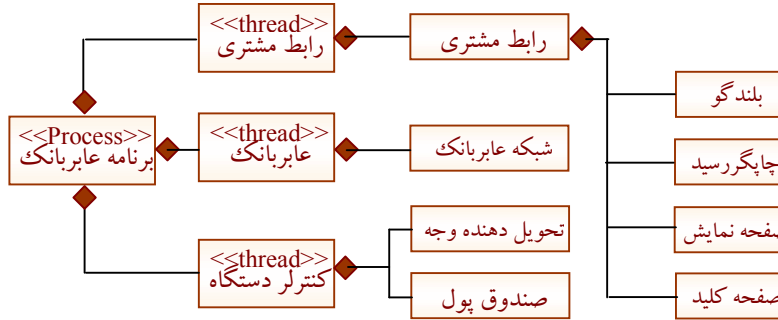
- توصیف مدل زمان اجرا

برای تعیین معماری اجرایی لازم است که همروندی در عملیات، فرایندها، شبکه ارتباط بین فرایندها، منابع مورد نیاز و توزیع عملیات در بین فرایندها مشخص شود. همروندی در عملیات بر معمولا بر مبنای میزان توزیع شدن عملیات، میزان پیچیدگی و زمان اجرای عملیات، قدرت پردازش موازی سیستم مورد استفاده و میزان نیاز به فرایندهای پشتیبانی جهت افزایش تحمل خطا مپخص می شود. در واقع استفاده از تکنیکهای چند ریسمانی^۸ و موازی سازی جهت تقلیل زمان اجرایی و یا دقت در عملیات انجام می شود. در آ.یو.پی با ایجاد مدلهای فرایند در دیدگاه فرایند می توان همروندی در فرایندها و عملیات را مشخص نمود. برای نمونه در شکل زیر فرایند اصلی ماشین عابر بانک در قالب سه ریسمان مشخص شده است. این سه ریسمان که با نوع <<thread>> مشخص شده اند، به طور همزمان می توانند فعال باشند.



شکل ۳-۱۲ فرایندها و ریسمانهای سیستم عابر بانک

پس از تعیین مدل فرایند می بایست برای نزدیک شدن به مدل پیاده سازی، کلاسها در بین فرایندها و ریسمانها، توزیع می شوند. برای نمونه به شکل زیر توجه نمایید. در این شکل رابط مشتری یک ریسمان یا در اصطلاح thread است که شامل کلاس رابط مشتری است. علامت لوزی نمایانگر شمول می باشد. و رابط مشتری نیز خود شامل چند کلاس دیگر است.



شکل ۳-۱۳ فرایندها و ریسمانهای سیستم عابر بانک

با تحلیل موردهای استفاده کلاسها مشخص می شوند. مدل ارتباطی کلاسها ساختار ایستای سیستم را مشخص می کند. برای تعیین مدل ارتباطی کلاسها ابتدا مسئولیتها مشخص می شوند. مسئولیتها به کلاسها تفویض می شوند. کلاسها بر اساس وابستگی و همکاری جهت انجام مسئولیتهای بزرگتر در قالب زیر سیستمها دسته بندی می شوند. کلاسها ممکن است در قالب ریسمانها دسته بندی شوند. رابطها نیز در این مرحله در قالب کلاسهای مجزا مشخص می شوند.

باید مشخص نمود که فرایندها و عملیات سیستم چگونه در سطح شبکه توزیع می شود. برای این منظور باید ساختار شبکه مشخص شود. برای ارائه طرح فیزیکی شبکه جزئیات زیر لازم است.

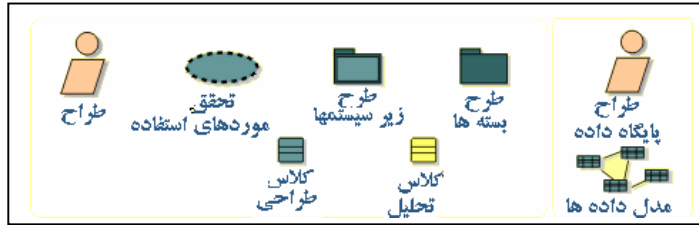
- ۱- مدل فیزیکی شبکه
- ۲- مشخصات فیزیکی سخت افزارها و نرم افزارهای نصب شده بر روی هر گره در شبکه
- ۳- پهنای باند
- ۴- هدف اصلی از هر گره در شبکه
- ۵- نقش اینترنت
- ۶- روش برقراری ارتباط فیزیکی در شبکه

حاصل در قالب طرح فیزیکی سیستم در قالب مدل نصب و در دیدگاه نصب در رشنال رز مشخص می شود. با در نظر گرفتن معماری نرم افزار در این مرحله، معماری فیزیکی نرم افزار تحت عنوان

مدل نصب نرم افزار نیز مشخص می شود. این مدل باید قادر به سرویس دهی به کاربرها بر اساس موردهای استفاده تعیین شده برای آنها و در محل فیزیکی مورد نظرشان باشد. طرح فیزیکی معماری در دیدگاه دیگری از محیط رز نشال رز تحت عنوان دیدگاه نصب⁹ مشخص می شود.

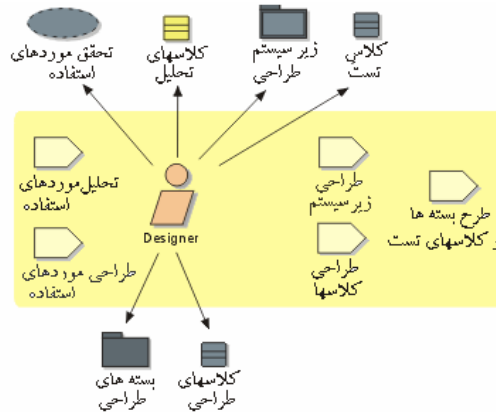
۲-۲-۳ طراحی نرم افزار

تحلیل شیئی گرا روشی خاص از تحلیل است که نیازها را بر اساس کلاسها و اشیاء مشخص شده در دامنه مساله، مورد بررسی قرار می دهد. طراحی شیئی گرا روشی است که بر اساس اشیای مورد دسترسی هر زیرسیستم، معماری و طرح نرم افزار را مشخص می کند. خروجیهای مرحله طراحی در شکل زیر مشخص شده است.



شکل ۳-۱۴ خروجیهای مرحله طراحی

وظیفه طراح تعیین ویژگیها، عملیات، رفتار و ارتباط بین کلاسها است. باید طراح پس از تعیین کلاسها مشخص کند که چگونه برای هماهنگی با محیط پیاده سازی هر کلاس اصلاح می شود.



شکل ۳-۱۵ فعالیتهای طراح سیستم

- تحلیل موردهای استفاده

توصیف موردهای استفاده ممکن است برای یافتن کلاسها و ارتباط بین آنها کامل نباشد. بنابراین لازم است که توصیف موردهای استفاده تکمیل شود تا کلاسها و جزئیات آنها را با ارجاع به سناریوی موردهای استفاده بتوان مشخص نمود. برای نمونه چنانچه در شرح سناریوی مورد استفاده از ماشین عابربانک جمله زیر درج شده باشد:

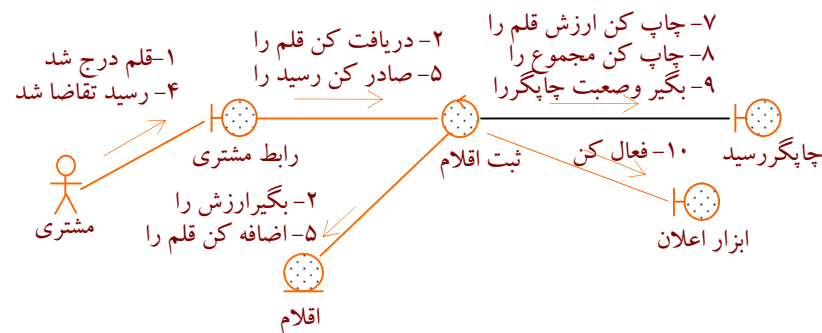
"ماشین عابربانک شماره کارت مشتری بانک را بررسی می کند"

جمله مبهم است زیرا، جزئیات بررسی مشخص نیست و نمی توان اسامی کلاسها و مسئولیت آنها را از متن این جمله مشخص نمود. برای این منظور جمله را به صورت زیر تکمیل می شود:

"ماشین عار بانک شماره حساب مشتری و شماره پین را برای شبکه عابر بانک ارسال می کند تا مورد بررسی قرار گیرد. شبکه عابر بانک مجوز استفاده را برگشت می دهد و مشتری تراکنش مورد نظر خود را وارد کند. در صورتی که شماره حساب یا شماره پین نامعتبر باشد مجوز داده نمی شود."

برای اینکه تصویری کامل از عملیات داخلی سیستم ایجاد شود، از توضیح عملیات در صورت امکان، اشیاء، ارتباط بین اشیاء و فایلها را بتوان مشخص نمود. عملیات خواندن، نوشتن، حذف و بروزرسانی اطلاعات فایلها، در شرح موردهای استفاده مشخص باشد. اطلاعات رد و بدل شونده و اسامی توابع را بتوان از شرح موردهای استفاده تعیین کرد. بطور خلاصه در شرح مورد استفاده مشخص شود که چه لازم است انجام شود.

البته، با استفاده از دیاگرامهای همکاری می توان عملکرد موردهای استفاده را مشخص نمود و به عنوان تکمیل کننده شرح مورد استفاده مشخص کرد. بنابراین پیشنهاد می شود که دیاگرامهای همکاری با توجه به سناریو برای موردهای استفاده ایجاد شود. برای نمونه در شکل زیر یک دیاگرام همکاری ارائه شده است. این دیاگرام بر اساس سناریو برای دریافت سفارش خرید از یک فروشگاه مشخص شده است.



شکل ۳-۱۵ دیاگرام همکاری برای سفارش خرید

پس از استخراج کلاسها از متن موردهای استفاده، مسئولیت کلاسها مشخص می شود. مسئولیتها شامل عملیات کلاس و دانش کلاس است. مسئولیتها از پیامهای مبادله شونده بین کلاسها در دیاگرام همکاری مشخص می شوند. مسئولیتهای دیگر را می توان از شرح نیازهای غیر عملیاتی مشخص کرد. بر مبنای مسئولیتها فیلهها، توابع و متدهای کلاسها را می توان مشخص نمود. دیاگرام همکاری در واقع شاخص مدل ارتباطی کلاسها نیز می باشد. ارتباط بین کلاسها معمولاً به این ترتیب ایجاد می شود که از طریق یک کلاس متدی از کلاس دیگر فراخوانی می شود.

فصل ۴

مدیریت پروژه در آ.یو.پی

۴-۱ مقدمه

۴-۲ مراحل مدیریت پروژه

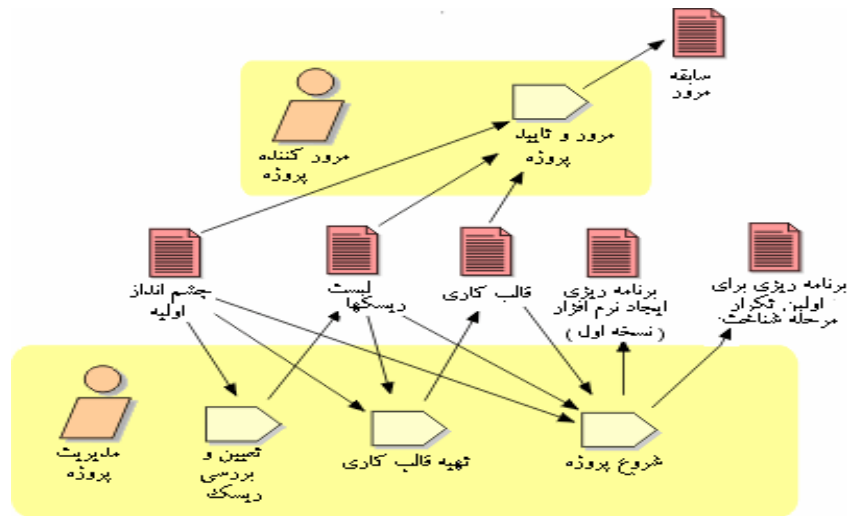
مدیریت پروژه های نرم افزاری در ارتباط با موازنه اهداف، کنترل ریسک و غلبه بر محدودیتها در تهیه محصولی است که پاسخگوی نیازهای مشتریان است. هدف در این فصل تعیین چارچوبی برای مدیریت پروژه ها، راهنمایی جهت تهیه برنامه ریزی، سازماندهی، اجرا و کنترل پروژه های نرم افزاری و بالاخره تعیین چارچوبی برای مدیریت ریسک است. در این قسمت تأکیدی بر روی قسمتهایی از امور مدیریت پروژه که در ارتباط با تعیین و استخدام افراد، تعیین و تخصیص بودجه و انعقاد قرارداد تأکیدی نشده است. تأکید بیشتر بر مدیریت ریسک، برنامه ریزی و کنترل پیشرفت پروژه ها است. مراحل مدیریت پروژه به شرح زیر است:

مراحل مدیریت پروژه ها

- § بررسی پروژه جدید
- § تعیین دامنه و ریسک انجام پروژه
- § برنامه ریزی
- § هدایت و کنترل پروژه
- § برنامه ریزی برای تکرار بعدی
- § مدیریت تکرار
- § مرحله پایانی
- § خاتمه پروژه

۴-۲-۱ بررسی پروژه جدید

هدف از این مرحله بررسی ایده اولیه پروژه تا رسیدن به مرحله ای است که بتوان درمورد انجام و یا عدم انجام پروژه تصمیم گیری نمود. برای این منظور با مطالعه اولیه پروژه چشم انداز اولیه برای انجام پروژه تهیه می شود. بر مبنای این چشم انداز قالب کاری مشخص می شود. در صورت تایید، پروژه به انجام می رسد. برای انجام پروژه، مدیر پروژه می بایست در قالب کاری پروژه حداقل دو روش را برای حصول به قابلیت‌های تعیین شده در چشم انداز پروژه تعیین نماید. در این راستا میزان ریسک در موفقیت پروژه برای هر روش را باید مشخص کند. در ضمن مرور و تایید پروژه، یکی از روشهای مشخص شده انتخاب می شود.



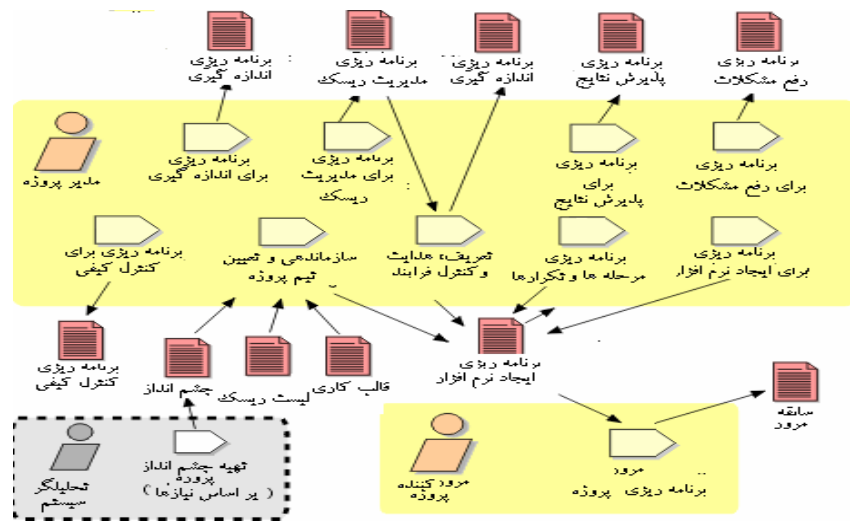
شکل ۱- گردش کار برای بررسی پروژه جدید

۴-۲-۲ برنامه ریزی پروژه

پس از تایید پروژه کار برنامه ریزی پروژه ها آغاز می شود. برنامه ریزی پروژه معمولاً با کاربر مورد تایید قرار می گیرد. برای برنامه ریزی پروژه می بایست مراحل انجام پروژه مشخص و زمانبندی شود. برای زمانبندی پروژه میزان اندازه پروژه را تعیین می کنند. البته ممکن است اندازه گیری در جهت تعیین میزان کیفیت، ریسک، تلاش، تغییرات، کامل شدن، قابلیت ردیابی و زمان لازم برای انجام پروژه و به طور کلی اندازه گیری دقیق اهداف باشد. برای نمونه ممکن است یک هدف، رضایت مشتری باشد. در این صورت باید ابتدا معیارهای رضایت مشتری تعیین شود و سپس بر اساس این معیارها میزان رضایت وی اندازه گیری شود.

کیفیت بر اساس تعداد نقصهای مشاهده شده، تلاش بر اساس نفر در ماه و اندازه گیری پروژه معمولاً بر اساس معیارهای تعداد خطوط، درجه عملکرد، درجه شیئی، درجه وب و امثال آن تعیین می شود. پس از تعیین اندازه پروژه، آریویی پیشنهاد می کند که از مدل‌های کوکومو برای تعیین میزان تلاش، مدت انجام پروژه و تعداد نفرات لازم برای انجام پروژه، استفاده می شود. به این ترتیب با تعیین تعداد نفرات و زمان انجام پروژه و سایر هزینه‌ها میزان بودجه‌ای که باید به پروژه تخصیص شود مشخص می شود.

همانطور که در شکل ۲ مشخص شده است با استفاده از مستندهای چشم انداز و قالب کاری و لیست ریسک، برنامه ریزی آغاز می شود. کنترل کیفیت نیز یکی از مباحث مهم در مهندسی نرم افزار است که می بایست برای آن برنامه ریزی شود.



شکل ۲- مراحل برنامه ریزی پروژه

پذیرش نهایی نرم افزار حاصل از انجام پروژه می تواند مشکلساز باشد. برای این منظور معمولاً همراه با مشتری باید برای چگونگی تحویل پروژه و روشهای ارزیابی آن بر اساس نیازهای تعیین شده برنامه ریزی شود. معیارهای پذیرش نرم افزار معمولاً در مرحله شناخت تعیین و در مرحله تشریح تکمیل می شوند. این معیارها با همکاری و تایید مشتری مشخص می شوند. در این توافقنامه می توان لیست افراد برای پذیرش و دریافت پروژه، محل تست، شرط موفقیت در ارزیابی، شرط موفقیت در آموزش مشتری، شرط موفقیت در نصب و بالاخره اندازه گیری میزان حصول اهداف و قابلیت‌های محصول نرم افزاری مشخص می شود. باید برای هر محصول نرم افزاری که تحویل می شود روش ارزیابی و منابع مورد نیاز تعیین شوند. بر مبنای روشهای ارزیابی معمولاً چک لیست تهیه می شود.

برای پذیرش پروژه، باید زمانبندی کرد و مشخص نمود که توسط چه افرادی و در چه زمانی پذیرش صورت می گیرد. همچنین باید مشخص نمود که در صورت عدم توافق هزینه های اضافه برای رفع مشکل چگونه تخصیص خواهد شد و ضرر و زیان به چه صورت است. برای نمونه برنامه ریزی کلی پروژه برای سیستم ثبت نام مشخص شده است:

برنامه ریزی توسعه نرم افزار

۱- مقدمه

۱-۱ هدف: هدف از برنامه ریزی، تعیین فعالیتهای لازم در قالب مراحل و تکرارهای مشخص شده بر طبق متدولوژی برای پیاده سازی یک سیستم کامپیوتری است. برای نمونه برنامه ریزی برای سیستم ثبت نام دانشجویان در این برنامه ریزی کلی مشخص شده است.

۱-۲ دامنه: این برنامه ریزی توسط واحد فن آوری اطلاعات در کالج ایکس جهت تولید برنامه نرم افزار برای استفاده در واحد ثبت نام، از اداره آموزش کالج است. این برنامه ریزی کلی است. برای هر تکرار به صورت مجزا جزئیات مشخص خواهد شد.

۱-۳ تعاریف، اصطلاحات، افتخارها: به لغتنامه ارجاع شود.

۱-۴ مراجع: مستندات چشم انداز، قالب پروژه، مستندات تقاضاهای افراد مرتبط، لغتنامه پروژه، گزارش مدل هزینه سیستم و بالاخره لیست ریسک برای سیستم ثبت نام کالج ایکس.

۱-۵ خلاصه: برنامه ریزی پروژه شامل جزئیات زیر است:

- خلاصه پروژه: شامل شرحی از هدف پروژه، دامنه و اهداف پروژه است. همچنین شامل محصولاتی است که پروژه می بایست در اختیار افراد قرار دهد.
- سازماندهی پروژه: ساختار سازمانی تیم پروژه را مشخص می کند.
- فرایند مدیریت: میزان تخمینی هزینه و مراحل اصلی پروژه را مشخص می کند و تعیین می کند که پروژه چگونه هدایت خواهد شد.
- برنامه ریزی فرایندهای تکنیکی: شامل خلاصه ای از فرایند تولید نرم افزار شامل متدها، ابزارها و تکنیکهایی است که باید پیروی شوند.

- برنامه ریزیهای پشتیبانی کننده : این شامل مدیریت تغییرات است.

۲- خلاصه پروژه

۲-۱ هدف و دامنه : برنامه ریزی پروژه شامل توصیف کلی برنامه ای است که باید سیستم ثبت نام کالج باید ایجاد شود. جزئیات هر یک از تکرارها در برنامه ریزی تکرارها مشخص شده است.

۲-۲ پیش فرضها و محدودیتها : سیستم قرار است که در مهرماه سال ۲۰۰۳ مورد استفاده دانشجویها برای عملیات ثبت نام قرار گیرد. سیستم باید تا آن زمان کاملاً آماده استفاده باشد.

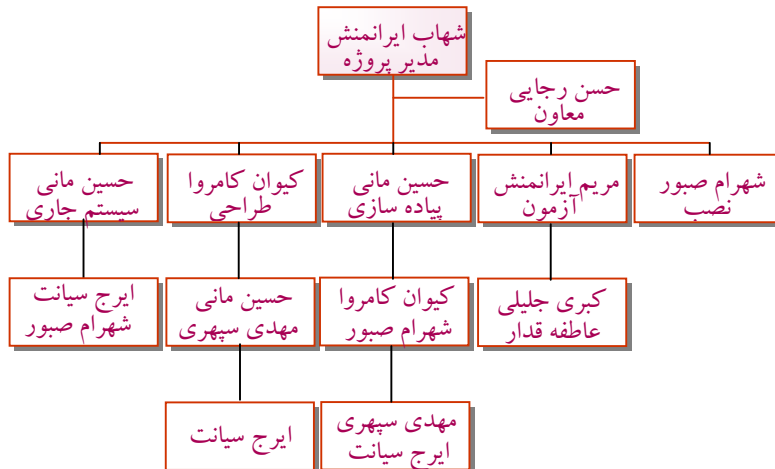
۲-۳ محصولات پروژه : محصولات زیر در ضمن انجام پروژه تهیه خواهند شد:

- برنامه ریزی پروژه
- چشم انداز
- موردهای استفاده
- لغتنامه
- شرح تکمیلی
- مستند معماری نرم افزار
- مدل طراحی
- زیر سیستم مورد پیاده سازی
- پیاده سازی
- آزمونهای نرم افزار
- تقاضای تغییرات
- خلاصه آزمون
- راهنمای استفاده از نسخه های نرم افزار

۲-۴ سیر تکاملی برنامه ریزی پروژه : برنامه ریزی پروژه برای هر تکرار قبل از آن تکرار ایجاد می شود.

۳- سازماندهی پروژه

۳-۱ ساختار سازمانی



۴- فرایند مدیریت

۴-۱ تخمین پروژه: تخمینهای پروژه بر اساس گزارش هزینه ها که در گزارش شماره ۷ مشخص شده می باشد. میزان سختی و پیچیدگی سیستم ثبت نام مشابه سیستم موجود کتابخانه دانشگاه است. پایگاه داده سیستم ثبت نام حدوداً ۲۹٪ و با در نظر گرفتن موردهای استفاده مجموع ۲۰٪ از سیستم موجود کتابخانه پیچیده تر است. بنابر این بر مبنای زمان و میزان تلاش بکار رفته در گزارش سیستم کتابخانه می توان بودجه را برای پروژه سیستم ثبت نام مشخص کرد.

۴-۲ برنامه ریزی پروژه

۴-۲-۱ برنامه ریزی مراحل: برنامه ریزی پروژه در بالاترین سطح با تعیین مدت زمان لازم و تعداد تکرارها برای هر مرحله به صورت زیر مشخص می شود:

مرحله	تعداد تکرارها	شروع	خاتمه
مرحله شناخت	۱	هفته ۱	هفته ۸
مرحله تشریح	۱	هفته ۸	هفته ۱۵
مرحله ایجاد	۳	هفته ۱۵	هفته ۳۱
مرحله انتقال	۳	هفته ۲۵	هفته ۳۷

در جدول زیر برای هر مرحله شرح و مقاطع عمده مشخص شده است:

مرحله	شرح	مقاطع مهم
مرحله شناخت	در مرحله شناخت نیازها مشخص می شوند و هزینه ها و درآمد حاصل از پروژه در قالب مستند قالب کاری تعیین می شود. در این مرحله موردهای استفاده اصلی توسعه داده شده، برنامه ریزی پروژه انجام می شود.	بررسی مورد کاری، تصمیم گیری برای ادامه یا عدم ادامه پروژه
مرحله تشریح	در مرحله تشریح نیازها تحلیل شده، نمونه سریع برای معماری نرم افزار ایجاد می شود. در انتهای مرحله تشریح کلیه موردهای استفاده برای نسخه یک نرم افزار طراحی خواهند شد. در ضمن موردهای استفاده با ریسک بالا برای نسخه ۲ طراحی خواهد شد.	آمادگی نمونه سریع معماری نمایانگر خاتمه مرحله تشریح است.
مرحله ایجاد	در مرحله ایجاد، مابقی موردهای استفاده تحلیل و طراحی می شوند. نسخه بتا آماده شده و برای ارزیابی توزیع می شود. فعالیتهای پیاده سازی و آزمون برای ایجاد نسخه های ۱ و ۲ خواهد بود.	قابلیتهای نسخه ۲ نرم افزار سیستم ثبت نام پیاده سازی می شوند. نسخه ۲ آماده بسته بندی خواهد بود.
مرحله انتقال	در مرحله انتقال نسخه ۱ و نسخه ۲ نرم افزار برای توزیع آماده خواهند بود. اطمینان حاصل می شود که کاربر آموزش داده شده و نرم افزار نصب خواهد شد.	ارائه نسخه ۲ و حصول اطمینان از پیاده سازی قابلیتهای تعیین شده در مستند چشم انداز پروژه.

۲-۲-۴ اهداف هر تکرار: هر مرحله شامل تعدادی تکرار چرخه حیات آبخاری است. در واقع هر تکرار یک بخش کامل از پروژه است. هدف از انجام پروژه در چند تکرار بجای یک تکرار که در روشهای قدیمی رایج بود به شرح زیر خلاصه می شود:

- کاهش ریسک تکنیکی
 - تهیه نسخه اولیه از یک سیستم عملیاتی
 - ایجاد ماکزیمم انعطاف در برنامه ریزی قابلیتها برای هر نسخه
 - ایجاد امکان تغییرات در دامنه برای هر تکرار
- جدول زیر تکرارها و مقاطع اصلی پروژه را مشخص می کند.

مرحله	تکرار	شرح	مقاطع اصلی	ریسک
مرحله شناخت	تکرار اصلی	تشریح مدل کاری، نیازهای محصول، برنامه ریزی و مورد کاری	مرور مورد کاری (شرح سود و زیان) - برنامه ریزی توسعه - تصمیم گیری در مورد اقتصادی بودن پروژه	- تعیین عجولانه نیاز
مرحله تشریح سریع	تکرار ۱ تهیه نمونه سریع معماری	تکمیل تحلیل و طراحی برای موردهای استفاده نسخه ۱. تهیه نمونه سریع از معماری. تکمیل تحلیل و طراحی برای موردهای استفاده با ریسک بالا برای نسخه ۲	نمونه سریع معماری	نکات مربوط به معماری دسته بندی می شود. ریسکهای تکنیکی کاهش داده می شود. نمونه سریع برای بررسی کاربر ایجاد می شود.
مرحله ایجاد	تکرار ۱- ایجاد نسخه بتا	پیاده سازی و آزمون موردهای استفاده نسخه ۱ برای ایجاد نسخه ۱	نسخه ۱ بتا	تمام ویژگیها از دیدگاه کاربر و معماری در این نسخه وجود دارد. گرفتن بازتاب کاربر قبل از ارائه نسخه نهایی
	تکرار ۲- ایجاد نسخه ۱	پیاده سازی مابقی موردهای استفاده و رفع نقایص نسخه بتا	نسخه ۱ نرم افزار	نسخه ۱ کاملاً توسط کاربر بررسی شده، کیفیت باید بالا باشد و نقصها حداقل باشد.
	تکرار ۳- ایجاد نسخه ۲	طرح، پیاده سازی و آزمون موردهای استفاده نسخه ۲، کلیه عملکردهای کلیدی در نسخه ۲ قرار می گیرد.	نرم افزار نسخه ۲	ارائه سریع نسخه ۲ موجب جلب رضایت مشتری می شود. تمامی ویژگیهای کلیدی سیستم در نسخه ۲
مرحله انتقال	تکرار ۱- ارائه نسخه ۱	بسته بندی مستندات و نصب نسخه ۱.	نسخه ۱	ارائه نرم افزار در دو مرحله موجب حداقل شدن نقصها

و عیبهای آن است				
ارائه نرم افزار در دومرحله موجب حداقل شدن نقصها و عیبهای آن است	نسخه ۲	بسته بندی مستندات و نصب نسخه ۲.	تکرار ۲- ارائه نسخه ۲	

۴-۲-۴ نسخه های نرم افزار: برنامه ریزی پروژه برای ایجاد دو نسخه اول از نرم افزار سیستم ثبت نام است. قابلیت های اصلی در مستند چشم انداز پروژه مشخص شده است [۱]. کلیه قابلیت هایی که برای دانشجویان مهم است در نسخه اول گنجانده شده است.

متن برنامه ریزی شده ممکن است بواسطه عوامل درونی یا محیطی با پیشرفت پروژه تغییر کند. برای کنترل تغییرات و ردیابی نیازها ابزار رشنال رز تحت عنوان ریکوایزیت پرو مورد استفاده قرار خواهد گرفت.

سیستم مکانیزه ثبت نام برای استفاده عموم در سطح دانشگاه باید مورد استفاده قرار گیرد. نسخه شماره ۱ می بایست موارد زیر را تحت پوشش قرار دهد:

- لاگ آن یا ورود به سیستم
- ثبت نام دروس
- ارتباط با پایگاه داده موجود برای دروس
- نگهداری اطلاعات دانشجویان
- نگهداری اطلاعات استادها
- نسخه ۲ از نرم افزار شامل موارد زیر است:
- درج نمره دروس دانشجویان
- مشاهده نمره ها
- گزینش درس برای تدریس

۴-۲-۴ زمانبندی پروژه: زمانبندی کلی پروژه شامل زمانبندی مراحل، تکرارها و مراحل پایانی به شرح زیر است.

نام وظیفه	شروع	خاتمه
مراحل پایانی	سه ۹۸/۱۲/۱	پنج ۹۹/۲۴/۶
شروع	سه ۹۸/۱۲/۱	سه ۹۸/۱۲/۱

سه ۹۹/۱/۱۹	سه ۹۹/۱/۱۹	مرور مورد کاری (خاتمه مرحله شناخت اولیه)
سه ۹۹/۳/۲	سه ۹۹/۳/۲	تهیه نمونه سریع معماری (خاتمه مرحله تشریح)
جمعه ۹۹/۴/۲	جمعه ۹۹/۴/۲	نسخه بتا آماده نصب
دو ۹۹/۵/۱۰	دو ۹۹/۵/۱۰	قابلیتهای عملیاتی اولیه نسخه شماره ۱، آماده
چهار ۹۹/۵/۱۹	چهار ۹۹/۵/۱۹	ارائه نسخه شماره یک
سه ۹۹/۶/۱۵	سه ۹۹/۶/۱۵	قابلیتهای عملیاتی اولیه نسخه شماره ۲، آماده
پنج ۹۹/۶/۲۴	پنج ۹۹/۶/۲۴	ارائه نسخه شماره دو
		مرحله شناخت اولیه
سه ۹۹/۱/۱۹	سه ۹۸/۱۲/۱	تکرار اول
سه ۹۹/۱/۱۹	سه ۹۸/۱۲/۱	مدلسازی سیستم جاری
دو ۹۹/۱۲/۲۱	سه ۹۸/۱۲/۱	شرح نیازها
سه ۹۹/۱/۱۹	سه ۹۸/۱۲/۱	مدیریت تغییرات
دو ۹۹/۱/۱۱	سه ۹۸/۱۲/۱	مدیریت
سه ۹۹/۱/۱۹	سه ۹۸/۱۲/۱	
		مرحله تشریح
سه ۹۹/۳/۲	چهار ۹۹/۱/۲۰	تکرار اول - توسعه نمونه سریع معماری
سه ۹۹/۳/۲	چهار ۹۹/۱/۲۰	مدلسازی سیستم جاری
جمعه ۹۹/۱/۲۲	چهار ۹۹/۱/۲۰	شرح نیازها
جمعه ۹۹/۱/۲۹	دو ۹۹/۱/۲۵	تحلیل و طراحی (معماری و ریسکهای اصلی)
جمعه ۹۹/۲/۱۲	دو ۹۹/۲/۱	پیاده سازی (معماری و ریسکهای اصلی)
جمعه ۹۹/۲/۱۹	دو ۹۹/۲/۱۵	آزمون (معماری و ریسکهای اصلی)
سه ۹۹/۳/۲	دو ۹۹/۲/۲۲	مدیریت
سه ۹۹/۳/۲	چهار ۹۹/۱/۲۰	
		مرحله ایجاد
سه ۹۹/۶/۱۵	چهار ۹۹/۳/۳	تکرار ۱ - توسعه نسخه ۱ بتا
جمعه ۹۹/۴/۲	چهار ۹۹/۳/۳	پیاده سازی نسخه بتا
چهار ۹۹/۳/۲۴	چهار ۹۹/۳/۳	آزمون (رابط کاربر و عملیات پیچیده)
جمعه ۹۹/۴/۲	پنج ۹۹/۳/۲۵	

مدیریت	چهار ۹۹/۳/۳	جمعه ۹۹/۴/۲
تکرار ۲- توسعه نسخه ۱	دو ۹۹/۴/۵	دو ۹۹/۵/۱۰
تحلیل و طراحی	دو ۹۹/۴/۵	دو ۹۹/۴/۱۲
پیاده سازی	سه ۹۹/۴/۱۳	دو ۹۹/۴/۲۶
آزمون (رابط کاربر و عملیات پیچیده)	سه ۹۹/۴/۲۷	دو ۹۹/۵/۱۰
مدیریت	دو ۹۹/۴/۵	دو ۹۹/۵/۱۰
تکرار ۳- توسعه نسخه ۲	سه ۹۹/۵/۱۱	سه ۹۹/۶/۱۵
تحلیل و طراحی	سه ۹۹/۵/۱۱	سه ۹۹/۵/۱۸
پیاده سازی	چهار ۹۹/۵/۱۹	چهار ۹۹/۶/۲
آزمون (رابط کاربر و عملیات پیچیده)	پنج ۹۹/۶/۳	سه ۹۹/۶/۱۵
مدیریت	سه ۹۹/۵/۱۱	سه ۹۹/۶/۱۵
مرحله انتقال		
تکرار ۱- نسخه ۱	سه ۹۹/۵/۱۱	پنج ۹۹/۶/۲۴
نصب	سه ۹۹/۵/۱۱	چهار ۹۹/۵/۱۹
تکرار ۲- نسخه ۲	پنج ۹۹/۶/۱۶	پنج ۹۹/۶/۲۴
نصب	پنج ۹۹/۶/۱۶	پنج ۹۹/۶/۲۴
محیط	سه ۹۸/۱/۱۲	جمعه ۹۹/۶/۲۵

۴-۲-۵ منابع پروژه
۴-۲-۵-۱ برنامه ریزی کارمندها

۴-۲-۶ بودجه

افراد	فعاليتها	تلاش (نفر در روز)	هزینه \$
	مدلسازی سیستم جاری	۴۵	۳۱۵۰۰
	نیازها	۹۰	۶۳۰۰۰
	تحلیل و طراحی	۱۳۰	۹۱۰۰۰

۱۴۴۲۰۰	۲۰۶	پیاده سازی	
۹۸۰۰۰	۱۴۰	آزمون	
۵۶۰۰۰	۸۰	مدیریت	
۳۵۰۰۰	۵۰	نصب	
۶۳۰۰۰	۹۰	محیط	
	۸۳۱	جمع تلاش	
۵۸۱۷۰۰	جمع افراد		
۰		مسافرت و اقامت	غیر فردی
۰		حمل و نقل	
۶۰۰۰		سرویس	
۳۲۰۰۰		اجناس	
۷۵۰۰		سایر هزینه های مستقیم	
۴۵۵۰۰	جمع غیر فردی		
۶۲۷۲۰۰	جمع بودجه		

۳-۴ برنامه ریزی برای تکرارها: به برنامه ریزی تکرارها در مرجع شماره [۱۱] ارجاع شود.

۴-۴ برنامه ریزی تکرارها

۱-۴-۴ برنامه ریزی برای هدایت و کنترل: نیازهای سیستم در مستند چشم انداز پروژه مشخص شده است.

۲-۴-۴ برنامه ریزی برای کنترل زمانبندی: مدیریت پروژه یک خلاصه زمانبندی تهیه می کند.

۳-۴-۴ برنامه ریزی برای کنترل بودجه:

۴-۴-۴ برنامه ریزی برای کنترل کیفی:

۵-۴-۴ برنامه ریزی برای تهیه گزارش:

۶-۴-۴ برنامه ریزی برای تخمین:

۵-۴ برنامه ریزی برای مدیریت ریسک:

۴-۶ برنامه ریزی برای خاتمه پروژه : برطبق این برنامه ریزی به مرور افراد از پروژه جدا می شوند. نهایتاً یک برنامه نویس برای پشتیبانی پروژه باقی خواهد ماند.

۵ برنامه ریزی فرایندهای تکنیکی

۵-۱ ابزار توسعه نرم افزار : به مرجع ابزار انجام پروژه [۷] ارجاع شود

۵-۲ متدها، ابزار و تکنیک : به مرجع ابزار توسعه سیستم ثبت نام ارجاع شود.

۵-۳ برنامه ریزی زیربنایی :

۵-۴ برنامه ریزی برای پذیرش محصول : برنامه ریزی برای پذیرش پروژه : ندارد

۶- برنامه ریزیهای پشتیبانی فرایند: ...

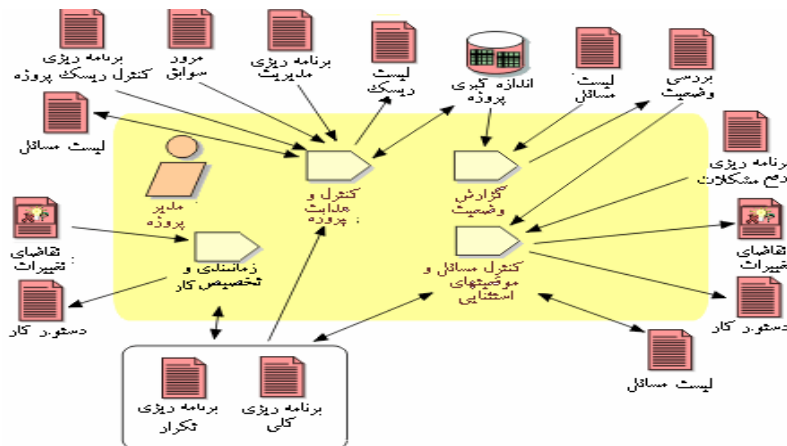
۷- برنامه ریزی های بیشتر

۸- ضمیمه ها

۹- فهرست : ...

۴-۲-۳ کنترل و هدایت پروژه

مدیر پروژه باید بطور پیوسته میزان پیشرفت پروژه را کنترل و ارزیابی کند. باید کیفیتها کنترل شود. برای اطمینان از کیفیت معمولاً نفراتی تخصص می شوند. میزان ریسک در انجام امور پروژه باید همواره مشخص و کنترل شود. باید گزارشهایی به طور منظم در ارتباط با میزان پیشرفت پروژه تهیه شود. مدیر باید قادر به درک مسائل و مشکلات و تدبیر راه حل‌های ممکن باشد. عملیات لازم برای کنترل و هدایت پروژه در شکل ۳ مشخص شده است:



شکل ۳- گردش کار برای مدیریت و کنترل پروژه

همانطور که در شکل ۳ مشخص شده است برای کنترل و هدایت پروژه، مدیر باید کارها را زمانبندی و به افراد تخصیص کند. معمولاً در ابتدای هر تکرار یا قدم از مراحل پروژه باید برنامه ریزی برای انجام آن تکرار انجام شود. در طی انجام یک تکرار ممکن است تغییراتی در برنامه ریزی نیاز گردد. این تغییرات می بایست مستند و تحت کنترل مدیر اعمال گردد. تغییرات ممکن است توسط تحلیلگر برای افزایش قابلیتها و توسط تیم کنترل کیفیت برای رفع نقصهای مشاهده شده درخواست شود. باید برای هر تغییر میزان تغییرات، میزان پیچیدگی تغییرات، زمان و نیروی لازم برای انجام تغییرات، میزان تاثیر، تست لازم، ارتباط با تغییرات دیگر و نهایتاً هزینه اعمال تغییرات را مشخص نمود.

برای کنترل پیشرفت پروژه، میزان کیفیت و میزان پیشرفت توسط معیارهای کمی نرم افزار مشخص می شوند. معمولاً گزارش پیشرفت پروژه توسط اعضای تیم تهیه می شود. این گزارش شامل، میزان تلاش انجام شده تا کنون و میزان تلاش لازم برای خاتمه وظیفه محول شده، نرم افزارهای ایجاد شده و مشکلات و مسائلی که با کمک مدیریت حل شده می باشد. بر اساس این گزارشها میزان پیشرفت پروژه توسط مدیر پروژه مشخص می شود. میزان پیشرفت بر اساس میزان تکمیل وظایف مشخص شده در برنامه ریزی پروژه تعیین می شود.

برای گزارش وضعیت پروژه معمولاً مدیر پروژه به لیستی از موارد و مشکلات و نکاتی که در ارتباط با پروژه مشاهده شده ارجاع می کند و گزارش نتایج حاصل از کنترل پروژه را برای افراد مشخص می کند. این گزارش شامل موارد زیر است:

میزان پیشرفت تکنیکی پروژه شامل کارهای تکمیل شده تا زمان تهیه گزارش و تغییرات کیفی و کمی اعمال شده.

پیشرفت بودجه شامل میزان بودجه هزینه شده نسبت به بودجه پیش بینی شده

میزان پیشرفت پروژه در مقایسه با زمانبندی پروژه

گزارشی مبتنی بر تجدید نظر در زمانبندی پروژه مبتنی بر میزان کار انجام شده و باقیمانده

گزارشی در ارتباط با وضعیت افراد تیم پروژه

میزان ریسک در انجام پروژه

گزارش مسائل مربوط به انجام پروژه و چگونگی رفع مسائل و مشکلات

لیستی از مسائلی که در گزارش قبلی مطرح شد و میزان پیشرفت در رفع آنها

همانطور که مشخص است در گزارش وضعیت باید مشکلات و مسائل مشخص شوند. این مشکلات ممکن است در ارتباط با برنامه ریزی غلط و یا افراد باشد و یا اینکه بواسطه مشکلاتی تکنیکی از قبیل نقصهای گزارش شده توسط تیم کنترل کیفیت، ابهام در نیازهای مطرح شده و یا

مشکلات در ابزار مورد استفاده باشد. در گزارش وضعیت پروژه همچنین حالات استثنایی از قبیل مثل مشکل در دیافت پاسخ از کاربرها، دسترسی به ابزار و کلا" هر گونه مشکل پیش بینی نشده ای که موجب عدم پیشرفت پروژه می شود است.

برای کنترل موارد استثناء و مسائل گزارش شده، مدیر باید دقیقاً هر مورد را ارزیابی و علت و معلول را مشخص کند و راه حل‌های ممکن را تعیین و مقایسه کند. پس از انتخاب راه حل مدیر باید تغییرات ممکن را مستند نموده، برنامه ریزی پروژه را اصلاح نماید.

۴-۲-۴ برنامه ریزی برای تکرار بعدی

برای هر تکرار به صورت جداگانه برنامه ریزی می شود. هر تکرار یک پروژه کوچک است. در برنامه ریزی که در ابتدا مرحله شناخت انجام میشود کلیات هر تکرار از جمله زمانبندی مشخص می گردد. برای هر تکرار باید مجموعه موردهای استفاده و نیازهای کیفی که در آن تکرار مورد بررسی قرار می گیرند مشخص شود. در خاتمه هر تکرار باید زمانی را به ارزیابی نتایج آن تکرار، تخصیص داد. ساختار تعیین شده برای برنامه ریزی هر تکرار به شرح زیر است:

۱- مقدمه: شامل خلاصه ای از کل مطالب در برنامه ریزی تکرار است.

۱-۱- هدف: بیانگر هدف از برنامه ریزی تکرار مورد نظر است. برای نمونه در اولین تکرار از مرحله شناخت در سیستم ثبت نام می توان هدف را به این صورت مشخص کرد که در این تکرار نیازهای سیستم مشخص خواهد شد و برنامه ریزی کلی برای کل پروژه مشخص خواهد شد. در این تکرار همچنین قالب کاری سیستم مورد بررسی قرار گرفته، مشخص می شود که آیا پروژه باید ادامه یابد یا اینکه مقرون به صرفه نیست.

در تکرار ۱ از مرحله تشریح برای سیستم ثبت نام، نیازها مورد تحلیل بیشتری قرار می گیرند. کلیه موردهای استفاده، تحلیل و طراحی خواهند شد. نمونه اولیه یا در اصطلاح پروتوتایپی از معماری کلی سیستم ثبت نام در این مرحله ایجاد خواهد شد. در تکرار ۱ از مرحله ایجاد نسخه ۱ نرم افزار تولید می شود. در تکرار ۲ نسخه ۱ تکمیل شده نقصها و اشکالات مشاهده شده رفع می گردد. به این ترتیب محصول تکرار ۲ نسخه ۱ از نرم افزار ثبت نام خواهد بود. در تکرار اول از مرحله انتقال سیستم ثبت نام حاصل کار پروژه در قالب یک بسته نرم افزاری آماده و نصب می شود. در تکرار دوم از مرحله انتقال نسخه دوم نرم افزار نصب می شود. این نسخه شامل کلیه قابلیت‌های مشخص شده در مستندات چشم انداز خواهد بود.

۱-۲- دامنه : شرح خلاصه ای از مقصود از برنامه ریزی تکرار مورد نظر در این قسمت مشخص می شود. برای نمونه برای اولین تکرار از مرحله شناخت در سیستم ثبت نام می توان گفت که دامنه پروژه در حد کالج ایکس است.

۱-۳- تعاریف : در این بخش یک دیکشنری از اصطلاحات، اسامی مختصر و مشابه ارائه می شود.

مراجع : لیستی از اسامی مستندات که برای برنامه ریزی تکرار از آنها استفاده شده است. برای نمونه در مورد تکرار اول از مرحله شناخت در سیستم ثبت نام مراجع شامل:

- مستندات چشم انداز پروژه

- مستندات تقاضاهای افراد برای سیستم ثبت نام

- دیکشنری اصطلاحات مورد استفاده در سیستم ثبت نام

- زمانبندی مرحله شناخت

- برنامه ریزی پروژه برای سیستم ثبت نام

در مرحله تشریح مراجع مورد استفاده برای اولین تکرار مشابه پنج مورد فوق هستند، با این تفاوت که زمانبندی برای مرحله شناخت مورد استفاده قرار می گیرد. در مرحله ایجاد مستندات برنامه ریزی و زمانبندی برای این مرحله به عنوان مرجع مورد استفاده قرار می گیرد. مستند دیگری که در این مرحله استفاده می شود، برنامه ریزی برای آزمون نرم افزار است.

۱-۵- خلاصه : در این قسمت خلاصه ای از ادامه گزارش برنامه ریزی ارائه می شود. در تکرار اول از مرحله تشریح بطور خلاصه زمانبندی شامل مراحل اصلی برای تکرار ارائه می شود. همچنین وضعیت مورد انتظار برای موردهای استفاده و معیارهای ارزیابی برای تکمیل تکرار مشخص می شود.

۲- برنامه ریزی : برنامه ریزی برای هر تکرار شامل تهیه دیاگرامهای زمانی، زمان شروع تست برنامه ها، نمونه های اجرایی و غیره است. در تکرار اول از مرحله شناخت در سیستم ثبت نام نیازهای سیستم مشخص می شوند و قالب پروژه تهیه می شود. موردهای استفاده اصلی توسعه داده می شوند و برنامه ریزی کلی پروژه تهیه می شود. در انتهای این مرحله است که برای سرمایه گذاری بر روی این پروژه کالج ایکس تصمیم گیری می نماید.

در مرحله تشریح برنامه اصلی تحلیل و طراحی کلیه موردهای استفاده سیستم است. نمونه اولیه از معماری سیستم تهیه می شود تا اینکه قابلیت و کارایی معماری طراح شده مشخص گردد. قابلیت‌های نسخه اول از نرم افزار ثبت نام به شرح زیر خواهد بود:

- لاگ کردن به داخل سیستم
- بستن ثبت نام
- ثبت نام دروس
- رابطی برای برقراری ارتباط با سیستم کاتالوگ دروس
- رابطی برای برقراری ارتباط با سیستم مالی
- تهیه اطلاعات مورد نیاز دانشجویان
- تهیه اطلاعات مورد نیاز اساتید

در تکرار دوم از مرحله ایجاد، باید بازتاب کاربر در ارتباط با نقصها و مشکلات عملی مشاهده شده از نسخه ۱ نرم افزار بر طرف شود. در این مرحله همچنین بخشهایی از نسخه ۱ نرم افزار که کامل نشده اند می بایست کامل گردد. موردهای استفاده اصلی در این مرحله عبارتند از:

- لاگ کردن به داخل سیستم
- بستن ثبت نام
- ثبت نام دروس
- رابطی برای برقراری ارتباط با سیستم کاتالوگ دروس
- رابطی برای برقراری ارتباط با سیستم مالی
- تهیه اطلاعات مورد نیاز دانشجویان
- تهیه اطلاعات مورد نیاز اساتید

تکرار اول از مرحله پشتیبانی زمانی آغاز می شود که نسخه ۱ نرم افزار آماده شود. نرم افزار بسته بندی شده و برای توزیع و نصب آماده می شود. تیم انتقال مسئولیت تهیه منوالهای کاربر و نصب نرم افزار را بر عهده خواهد داشت. نصب نرم افزار بر روی سرویس دهنده کنونی کالج و ایستگاه های کاری مشخص شده برای کاربرها انجام خواهد شد. تیم انتقال همچنین مسئولیت انجام آزمون پذیرش و گرفتن پذیرش از مشتری به عهده این تیم است. تیم انتقال وظیفه آموزش و پشتیبانی نرم افزار را بر عهده خواهد داشت.

تکرار دوم مرحله پشتیبانی با تکمیل شدن نسخه ۲ نرم افزار ثبت نام آغاز می شود. فعالیتها برای تکرار دوم مشابه تکرار اول است. با این تفاوت که داده های حاصل از نسخه ۱ می بایست به نسخه ۲ انتقال داده شود.

۱-۲- فعالیتهای تکرار : در این قسمت زمانهای مشخص شده برای شروع و خاتمه فعالیتهای پروژه مستند می گردند. برای نمونه فعالیتها در تکرار اول از مرحله شناخت در سیستم ثبت نام به شرح زیر هستند.

فعالیت	تاریخ شروع	تاریخ خاتمه
مدلسازی سیستم جاری	اول مهر ۱۳۸۲	۲۴ مهر ۱۳۸۲
تعیین نیازها	اول مهر ۱۳۸۲	۲۵ آبان ۱۳۸۲
مدیریت ترکیب	اول مهر ۱۳۸۲	۱۵ آبان ۱۳۸۲
مدیریت	اول مهر ۱۳۸۲	۱ آذر ۱۳۸۲

برای مرحله تشریح فعالیتها در تکرار اول به صورت زیر می باشند:

فعالیت	تاریخ شروع	تاریخ خاتمه
مدلسازی سیستم جاری	۳ آبان ۱۳۸۲	۵ آبان ۱۳۸۲
تعیین نیازها (موردهای استفاده س ۱ و س ۲ ریسک بالا)	۳ آبان ۱۳۸۲	۹ آبان ۱۳۸۲
تحلیل و طراحی (موردهای استفاده س ۱ و س ۲ ریسک بالا)	۵ آبان ۱۳۸۲	۱۸ آبان ۱۳۸۲
پیاده سازی (نمونه اولیه معماری س ۱)	۱۵ آبان ۱۳۸۲	۲۴ آبان ۱۳۸۲
آزمون (نمونه اولیه معماری س ۱)	۲۲ آبان ۱۳۸۲	۲ آذر ۱۳۸۲
محیط	۲ آبان ۱۳۸۲	۴ آذر ۱۳۸۲
مدیریت	۲ آبان ۱۳۸۲	۴ آذر ۱۳۸۲

فعالیتهای لازم برای انجام تکرار اول از مرحله ایجاد به شرح زیر است:

فعالیت	تاریخ شروع	تاریخ خاتمه
تحلیل و طراحی	۱۶ آذر ۱۳۸۲	۲۵ آذر ۱۳۸۲
پیاده سازی نسخه ۱	۱۶ آذر ۱۳۸۲	۶ بهمن ۱۳۸۲
آزمون نسخه ۱	۱۶ آذر ۱۳۸۲	۸ بهمن ۱۳۸۲
محیط	۱۵ آذر ۱۳۸۲	۱۲ بهمن ۱۳۸۲

مدیریت	۱۵ آذر ۱۳۸۲	۱۲ بهمن ۱۳۸۲
--------	-------------	--------------

فعاليتها برای تکرارهای ۱ و ۲ از مرحله انتقال به شرح زیر هستند

تاریخ خاتمه	تاریخ شروع	فعالیت
		تکرار ۱ - نسخه ۱
۱ خرداد ۱۳۸۳	۲۷ اردیبهشت ۱۳۸۳	تولید نرم افزار
۷ خرداد ۱۳۸۳	۲ خرداد ۱۳۸۳	بسته بندی و توزیع نسخه ۱
۱۱ خرداد ۱۳۸۳	۱۰ خرداد ۱۳۸۳	آزمون پذیرش
		تکرار ۲ - نسخه ۲
۲ تیر ۱۳۸۳	۳۰ خرداد ۱۳۸۳	تولید نرم افزار
۸ تیر ۱۳۸۳	۵ تیر ۱۳۸۳	بسته بندی و توزیع نسخه ۱
۲۰ تیر ۱۳۸۳	۳۰ خرداد ۱۳۸۳	انتقال داده ها از نسخه ۱ به نسخه ۲
۱۴ تیر ۱۳۸۳	۱۳ تیر ۱۳۸۳	آزمون پذیرش

۲-۲- زمانبندی تکرار: در این قسمت ریز جزئیات زمانبندی پروژه برای یک تکرار مشخص می شود. برای تکرار اول از مرحله شناخت سیستم ثبت نام زمانبندی به شرح زیر است:

نام وظیفه	مدت	شروع	خاتمه	منابع
مدت کل	۴۵ روز	۸۲/۷/۱	۸۲/۹/۱	
- شروع	۰ روز	۸۲/۷/۱	۸۲/۷/۱	
- مرور قالب کاری (خاتمه مرحله شناخت)	۰ روز	۸۲/۹/۱	۸۲/۹/۱	
مرحله شناخت	۴۵ روز	۸۲/۷/۱	۸۲/۹/۱	
اولین تکرار	۴۵ روز	۸۲/۷/۱	۸۲/۹/۱	
• مدلسازی سیستم کاری	۱۸ روز	۸۲/۷/۱	۸۲/۷/۲۴	
- تهیه دیکشنری اصطلاحات و تعاریف	۹ روز	۸۲/۷/۱	۸۲/۷/۱۱	تحلیلگر فرایند کاری
- یافتن بازیگرها و مورد های استفاده کاری	۳ روز	۸۲/۷/۹	۸۲/۷/۱۱	تحلیلگر فرایند کاری
- توصیف مورد های استفاده کاری	۹ روز	۸۲/۷/۱	۸۲/۷/۱۱	
-- توصیف مورد استفاده کاری ۱	۹ روز	۸۲/۷/۱	۸۲/۷/۱۱	طراح سیستم کاری

طراح سیستم کاری	۸۲/۷/۱۱	۸۲/۷/۱	روز ۹	-- توصیف مورد استفاده کاری ۲
تحلیلگر فرایند کاری	۸۲/۷/۱۸	۸۲/۷/۱	روز ۱۴	- ایجاد مدل موردهای استفاده کاری
بررسی کننده مدل فرایند کاری	۸۲/۷/۲۴	۸۲/۷/۲۱	روز ۴	- بررسی مدل موردهای استفاده کاری
	۸۲/۸/۲۵	۸۲/۷/۱	روز ۴۰	• تعیین نیازها
تحلیلگر سیستم	۸۲/۸/۴	۸۲/۷/۱	روز ۲۵	- تهیه مستندات چشم انداز پروژه
تحلیلگر سیستم	۸۲/۸/۸	۸۲/۸/۵	روز ۴	- گردآوری نیازهای افراد
تحلیلگر سیستم	۸۲/۸/۵	۸۲/۷/۱	روز ۲۶	- مدیریت وابستگیها
تحلیلگر سیستم	۸۲/۸/۵	۸۲/۷/۲۳	روز ۱۰	- تکمیل دیکشنری اصطلاحات
تحلیلگر سیستم	۸۲/۸/۵	۸۲/۷/۲۳	روز ۱۰	- تعیین بازیگرها و موردهای استفاده
معمار	۸۲/۸/۲۵	۸۲/۸/۱۲	روز ۱۰	- اولویت بندی موردهای استفاده
تحلیلگر سیستم	۸۲/۸/۱۱	۸۲/۸/۱۱	روز ۱	- تعیین محدودیتهای سیستم
	۸۲/۸/۱۵	۸۲/۷/۱	روز ۳۴	• مدیریت ترکیب بخشهای پروژه
مدیر پروژه	۸۲/۷/۱۸	۸۲/۷/۱	روز ۱۴	- تعیین عملیات ترکیب
مدیر ترکیب	۸۲/۸/۱۵	۸۲/۷/۲۱	روز ۲۰	- ایجاد محیط عملیات ترکیب
	۸۲/۹/۱	۸۲/۷/۱	روز ۴۵	• مدیریت
مدیر پروژه	۸۲/۸/۱۸	۸۲/۷/۱	روز ۳۵	- تعیین ریسک
مدیر پروژه	۸۲/۸/۱	۸۲/۷/۱	روز ۲۴	- برنامه ریزی مراحل
مدیر پروژه	۸۲/۸/۱۸	۸۲/۸/۴	روز ۱۱	- تعیین افراد تیم پروژه
مدیر پروژه	۸۲/۹/۱	۸۲/۸/۱۱	روز ۱۶	- تولید قالب کاری

برای تکرار اول از مرحله تشریح جزئیات زمانبندی به صورت زیر می باشد:

منابع	خاتمه	شروع	نام وظیفه
	۸۲/۱۰/۴	۸۲/۹/۲	مدت کل
	۸۲/۹/۲	۸۲/۹/۲	- شروع
	۸۲/۹/۳	۸۲/۹/۳	- مرحله مهم برای تهیه رابط کاربر
	۸۲/۱۰/۴	۸۲/۱۰/۴	- مرحله مهم برای نمونه اولیه معماری
	۸۲/۱۰/۴	۸۲/۹/۲	مرحله شناخت
	۸۲/۱۰/۴	۸۲/۹/۲	اولین تکرار - تهیه نمونه اولیه معماری

	۸۲/۹/۵	۸۲/۹/۳	• مدلسازی سیستم کاری
- تعیین موجودیها و کارکنهای سیستم کاری	۸۲/۹/۳	۸۲/۹/۳	
- تشریح کارکنهای سیستم کاری			
- یافتن کارکنها و موجودیهای سیستم کاری	۸۲/۹/۵	۸۲/۹/۳	
- توصیف کارکنهای سیستم کاری	۸۲/۹/۴	۸۲/۹/۴	
-- توصیف کارکنهای سیستم کاری ۱	۸۲/۹/۴	۸۲/۹/۴	
-- توصیف کارکنهای سیستم کاری ۲	۸۲/۹/۴	۸۲/۹/۴	
- توصیف موجودیهای سیستم کاری	۸۲/۹/۴	۸۲/۹/۴	
-- توصیف موجودیهای سیستم کاری ۱	۸۲/۹/۴	۸۲/۹/۴	
-- توصیف موجودیهای سیستم کاری ۲	۸۲/۹/۴	۸۲/۹/۴	
- بررسی مدل اشیاء سیستم کاری	۸۲/۹/۴	۸۲/۹/۴	
	۸۲/۹/۹	۸۲/۹/۳	• تعیین نیازها
- کنترل وابستگیها	۸۲/۹/۹	۸۲/۹/۳	
- تهیه جزئیات موردهای استفاده بر طبق اولویت	۸۲/۹/۳	۸۲/۹/۳	
-- تهیه جزئیات موردهای استفاده ۱ الی ۵	۸۲/۹/۳	۸۲/۹/۳	
- تعیین ساختار مدل موردهای استفاده	۸۲/۹/۴	۸۲/۹/۴	
- مدلسازی رابط کاربر	۸۲/۹/۳	۸۲/۹/۳	
- تهیه نمونه اولیه رابط کاربر	۸۲/۹/۳	۸۲/۹/۳	
- مرور نیازها	۸۲/۹/۵	۸۲/۹/۵	
	۸۲/۹/۱۸	۸۲/۹/۵	• تحلیل و طراحی (معماری و ریسک)
- تحلیل معماری	۸۲/۹/۵	۸۲/۹/۵	
- تحلیل موردهای استفاده بر طبق اولویت	۸۲/۹/۸	۸۲/۹/۸	
-- تحلیل موردهای استفاده ۱ تا ۴	۸۲/۹/۸	۸۲/۹/۸	
- طراحی معماری	۸۲/۹/۹	۸۲/۹/۹	
- تشریح همروندی	۸۲/۹/۱۰	۸۲/۹/۱۰	
- تشریح توزیع عملیات	۸۲/۹/۱۱	۸۲/۹/۱۱	
- مرور و بررسی معماری	۸۲/۹/۱۲	۸۲/۹/۱۲	

	۸۲/۹/۱۵	۸۲/۹/۱۵	- طراحی زیر سیستمها بر اساس اولویت
طراح	۸۲/۹/۱۵	۸۲/۹/۱۵	-- طراحی زیر سیستم ۱ الی ۲
	۸۲/۹/۱۵	۸۲/۹/۱۵	- طراحی کلاسها بر اساس اولویت
طراح	۸۲/۹/۱۵	۸۲/۹/۱۵	-- طراحی کلاسهای ۱ الی ۴
	۸۲/۹/۱۶	۸۲/۹/۱۶	- طراحی موردهای استفاده بر اساس اولویت
طراح	۸۲/۹/۱۶	۸۲/۹/۱۶	-- طراحی موردهای استفاده ۱ الی ۲
طراح بانک اطلاعاتی	۸۲/۹/۱۷	۸۲/۹/۱۷	- طراحی بانک اطلاعاتی
بررسی کننده طرح	۸۲/۹/۱۸	۸۲/۹/۱۸	- مرور و بررسی طراحی
	۸۲/۹/۲۴	۸۲/۹/۱۵	• پیاده سازی (معماری و ریسک)
معمار	۸۲/۹/۱۵	۸۲/۹/۱۵	- ایجاد ساختار پیاده سازی
یکپارچه کننده سیستم	۸۲/۹/۱۹	۸۲/۹/۱۹	- برنامه ریزی برای یکپارچه سازی سیستم
پیاده کننده سیستم	۸۲/۹/۲۲	۸۲/۹/۲۲	- برنامه ریزی برای یکپارچه سازی زیرسیستمها
	۸۲/۹/۲۲	۸۲/۹/۲۲	- پیاده سازی قطعات نرم افزاری بر طبق اولویت
پیاده ساز	۸۲/۹/۲۲	۸۲/۹/۲۲	-- پیاده سازی قطعه ۱ الی ۴
	۸۲/۹/۲۳	۸۲/۹/۲۳	- رفع نقصها
پیاده کننده سیستم	۸۲/۹/۲۳	۸۲/۹/۲۳	- یکپارچه سازی زیر سیستمها
یکپارچه کننده سیستم	۸۲/۹/۲۴	۸۲/۹/۲۴	- یکپارچه سازی سیستم
	۸۲/۱۰/۳	۸۲/۹/۲۲	• آزمون (معماری و ریسک)
طراح آزمون	۸۲/۹/۲۲	۸۲/۹/۲۲	- برنامه ریزی آزمون (ترکیب و سیستم)
طراح آزمون	۸۲/۹/۲۳	۸۲/۹/۲۳	- طرح آزمون (ترکیب و سیستم)
طراح آزمون	۸۲/۹/۲۴	۸۲/۹/۲۴	- پیاده سازی آزمون
طراح	۸۲/۹/۲۵	۸۲/۹/۲۵	- طراحی آزمون کلاسها و بسته ها
پیاده ساز	۸۲/۹/۲۵	۸۲/۹/۲۵	- پیاده سازی آزمون کلاسها و بسته ها
مسئول آزمون یکپارچگی	۸۲/۱۰/۱	۸۲/۱۰/۱	- اجرا آزمون ترکیب
مسئول آزمون سیستم	۸۲/۱۰/۲	۸۲/۱۰/۲	- اجرا آزمون سیستم
مسئول آزمون کارایی	۸۲/۱۰/۱	۸۲/۱۰/۱	- اجرا آزمونهای کارایی
طراح آزمون	۸۲/۱۰/۲	۸۲/۱۰/۲	- ارزیابی آزمون
	۸۲/۱۰/۴	۸۲/۹/۲	• مدیریت

مدیر پروژه	۸۲/۹/۲	۸۲/۹/۲	- ایجاد برنامه ریزی برای تکرار
مدیر پروژه	۸۲/۹/۳	۸۲/۹/۳	- تعیین نفرات تیم پروژه
مدیر پروژه	۸۲/۱۰/۲	۸۲/۹/۲	- بررسی مجدد لیست ریسک
مدیر پروژه	۸۲/۱۰/۳	۸۲/۱۰/۳	- ارزیابی تکرار
مدیر پروژه	۸۲/۱۰/۴	۸۲/۱۰/۴	- اصلاح برنامه ریزی پروژه
	۸۲/۱۰/۴	۸۲/۹/۲	• محیط
	۸۲/۹/۱۱	۸۲/۹/۲	- هماهنگی کردن فرایندها
	۸۲/۱۰/۳	۸۲/۹/۱۲	- بهبود فرایندها
	۸۲/۱۰/۳	۸۲/۹/۲	- انتخاب و تهیه ابزار
	۸۲/۱۰/۳	۸۲/۹/۲	- ملاقات برای تهیه ابزار
	۸۲/۱۰/۳	۸۲/۱۰/۳	- تهیه امکانات برای پشتیبانی فرایند تولید
	۸۲/۱۰/۴	۸۲/۱۰/۴	- آموزش

در ادامه زمانبندی برای تکرار اول از مرحله ایجاد ارائه شده است. زمانبندی برای مراحل دوم و سوم به صورت مشابه پس از تکرار اول انجام می شود.

منابع	خاتمه	شروع	نام وظیفه
	۸۲/۱۲/۱۷	۸۲/۹/۱۵	مراحل اصلی
	۸۲/۹/۱۵	۸۲/۹/۱۵	- شروع تکرار ۱ از مرحله ایجاد
	۸۲/۱۰/۱۲	۸۲/۱۰/۱۲	- مراحل اصلی در تولید نسخه ۱
	۸۲/۱۰/۱۲	۸۲/۱۰/۱۲	- شروع تکرار ۲ از مرحله ایجاد
	۸۲/۱۱/۱۴	۸۲/۱۱/۱۴	- مرحله اصلی در قابلیت‌های اجرایی نسخه ۱
	۸۲/۱۱/۱۴	۸۲/۱۱/۱۴	- شروع تکرار ۳ از مرحله ایجاد
	۸۲/۱۲/۱۷	۸۲/۱۲/۱۷	- مرحله اصلی در قابلیت‌های اجرایی نسخه ۲
	۸۲/۱۲/۱۷	۸۲/۹/۱۵	مرحله ایجاد
	۸۲/۱۰/۱۲	۸۲/۹/۱۵	اولین تکرار - تولید نسخه ۱
	۸۲/۹/۲۵	۸۲/۹/۱۶	• تحلیل و طراحی
	۸۲/۹/۲۵	۸۲/۹/۱۶	- تعیین موجودی‌ها و کارکنهای سیستم کاری

معمار	۸۲/۹/۲۵	۸۲/۹/۲۲	- تعدیل معماری
	۸۲/۹/۱۷	۸۲/۹/۱۶	- طراحی زیرسیستمها بر اساس اولویت
طراح	۸۲/۹/۱۷	۸۲/۹/۱۶	-- طراحی زیر سیستم زس- ف ک
طراح	۸۲/۹/۱۷	۸۲/۹/۱۶	-- طراحی زیر سیستم زس- س ر
	۸۲/۹/۲۵	۸۲/۹/۱۶	- طراحی کلاسها بر اساس اولویت
طراح	۸۲/۹/۱۷	۸۲/۹/۱۶	-- طراحی کلاسهای ک-رب خ
طراح	۸۲/۹/۱۷	۸۲/۹/۱۶	-- طراحی کلاسهای ک-چ ک خ
طراح	۸۲/۹/۱۷	۸۲/۹/۱۶	-- طراحی کلاسهای ک-م و د
طراح	۸۲/۹/۱۷	۸۲/۹/۱۶	-- طراحی کلاسهای ک-ف و د
	۸۲/۹/۱۹	۸۲/۹/۱۸	- طراحی موردهای استفاده بر اساس اولویت
طراح	۸۲/۹/۱۹	۸۲/۹/۱۸	-- طراحی موردهای م-ه ی و
طراح	۸۲/۹/۱۹	۸۲/۹/۱۸	-- طراحی موردهای م-م ر ک
طراح بانک	۸۲/۹/۲۳	۸۲/۹/۲۲	- طراحی بانک اطلاعاتی
بررسی کننده بانک	۸۲/۹/۲۳	۸۲/۹/۲۲	- بررسی طراحی
	۸۲/۱۰/۶	۸۲/۹/۱۶	• پیاده سازی
یکپارچه کننده سیستم	۸۲/۹/۱۷	۸۲/۹/۱۶	- برنامه ریزی برای یکپارچه سازی سیستم
پیاده کننده	۸۲/۹/۱۹	۸۲/۹/۱۸	- برنامه ریزی برای یکپارچه سازی زیرسیستمها
	۸۲/۹/۲۳	۸۲/۹/۲۲	- پیاده سازی قطعه های کد بر اساس اولویت
پیاده کننده	۸۲/۹/۲۳	۸۲/۹/۲۲	-- پیاده سازی قطعه ق-و ن س
پیاده کننده	۸۲/۹/۲۳	۸۲/۹/۲۲	-- پیاده سازی قطعه ق-و م س
پیاده کننده	۸۲/۹/۲۳	۸۲/۹/۲۲	-- پیاده سازی قطعه ق-و ن ر
پیاده کننده	۸۲/۹/۲۳	۸۲/۹/۲۲	-- پیاده سازی قطعه ق-و م ز
پیاده کننده	۸۲/۹/۲۳	۸۲/۹/۲۲	-- پیاده سازی قطعه ق-و ن ج
پیاده کننده	۸۲/۹/۲۳	۸۲/۹/۲۲	-- پیاده سازی قطعه ق-و م چ
پیاده کننده	۸۲/۹/۲۴	۸۲/۹/۲۲	- رفع نقصها
بررسی کننده کد	۸۲/۹/۲۶	۸۲/۹/۲۵	- بررسی کد

انجام آزمون واحدها یا پیمانها	۸۲/۹/۲۵	۸۲/۹/۲۹	پیاده کننده
- یکپارچه سازی زیرسیستمها	۸۲/۹/۳۰	۸۲/۱۰/۱	پیاده کننده
- یکپارچه سازی سیستمها	۸۲/۱۰/۲	۸۲/۱۰/۶	یکپارچه کننده سیستم
• آزمون (رابطها و توابع یکپارچه سازی)	۸۲/۹/۱۶	۸۲/۱۰/۸	
- برنامه ریزی آزمون (ترکیب و سیستم)	۸۲/۹/۱۶	۸۲/۹/۱۶	طراح آزمون
- طرح آزمون (ترکیب و سیستم)	۸۲/۹/۱۷	۸۲/۹/۱۷	طراح آزمون
- پیاده سازی آزمون	۸۲/۹/۱۸	۸۲/۹/۱۸	طراح آزمون
- طراحی آزمون کلاسها و بسته ها	۸۲/۹/۱۹	۸۲/۹/۱۹	طراح
- پیاده سازی آزمون کلاسها و بسته ها	۸۲/۹/۲۲	۸۲/۹/۲۲	پیاده ساز
- اجرا آزمون یکپارچه سازی	۸۲/۱۰/۲	۸۲/۱۰/۲	مسئول آزمون یکپارچگی
- اجرا آزمون سیستم	۸۲/۱۰/۷	۸۲/۱۰/۷	مسئول آزمون سیستم
- اجرا آزمونهای کارایی	۸۲/۹/۲۳	۸۲/۹/۲۳	مسئول آزمون کارایی
- ارزیابی آزمون	۸۲/۱۰/۸	۸۲/۱۰/۸	طراح آزمون
• مدیریت	۸۲/۹/۱۵	۸۲/۱۰/۱۲	
- ایجاد برنامه ریزی برای تکرار	۸۲/۹/۱۵	۸۲/۹/۱۵	مدیر پروژه
- تعیین نفرات تیم پروژه	۸۲/۹/۱۶	۸۲/۹/۱۶	مدیر پروژه
- بررسی مجدد لیست ریسک	۸۲/۹/۱۵	۸۲/۹/۲۹	مدیر پروژه
- ارزیابی تکرار	۸۲/۱۰/۳	۸۲/۱۰/۳	مدیر پروژه
- اصلاح برنامه ریزی پروژه	۸۲/۱۰/۱۲	۸۲/۱۰/۱۲	مدیر پروژه
تکرار دوم و سوم مشابه تکرار اول هستند			
• محیط	۸۲/۹/۳	۸۲/۱۲/۱۷	
- پشتیبانی پیاده سازی	۸۲/۹/۱۵	۸۲/۱۲/۱۷	

تکرار اول و دوم از مرحله انتقال به شرح ذیل زمانبندی شده اند:

نام وظیفه	شروع	خاتمه
بخشهای اصلی	۸۳/۲/۲۷	۸۳/۴/۱۴
- شروع تکرار ۱ از مرحله انتقال	۸۳/۲/۲۷	۸۳/۲/۲۷
- پذیرش نسخه ۱	۸۳/۳/۱۱	۸۳/۳/۱۱

۸۳/۳/۳۰	۸۳/۳/۳۰	- شروع تکرار ۲ از مرحله انتقال
۸۳/۴/۱۴	۸۳/۴/۱۴	- پذیرش نسخه ۲
۸۳/۴/۲۰	۸۳/۲/۲۷	مرحله انتقال
۸۳/۳/۱۶	۸۳/۲/۲۷	• تکرار ۱- نسخه ۱
۸۳/۲/۲۸	۸۳/۲/۲۷	- نصب نرم افزار
۸۳/۳/۱	۸۳/۲/۲۷	-- تولید نرم افزار
۸۳/۲/۲۸	۸۳/۲/۲۷	--- دستورالعملهای نصب
۸۳/۳/۱	۸۳/۲/۲۷	--- مستندهای نصب
۸۳/۲/۲۸	۸۳/۲/۲۷	--- داده های ترکیب
۸۳/۳/۳	۸۳/۳/۲	-- بسته بندی نسخه ۱
۸۳/۳/۷	۸۳/۳/۴	-- توزیع نسخه ۱
۸۳/۳/۹	۸۳/۳/۸	-- نصب نرم افزار
۸۳/۳/۱۶	۸۳/۳/۱۰	-- تهیه راهنما و کمک به کاربر
۸۳/۳/۱۱	۸۳/۳/۱۰	-- آزمون پذیرش
۸۳/۴/۲۰	۸۳/۳/۳۰	• تکرار ۲- نسخه ۲
۸۳/۴/۲۰	۸۳/۳/۳۰	- نصب نرم افزار
۸۳/۴/۲	۸۳/۳/۳۰	-- تولید نرم افزار
۸۳/۳/۳۰	۸۳/۳/۳۰	--- دستورالعملهای نصب
۸۳/۴/۲	۸۳/۳/۳۰	--- مستندهای نصب
۸۳/۳/۳۰	۸۳/۳/۳۰	--- داده های ترکیب
۸۳/۴/۱	۸۳/۳/۳۰	--- برنامه های جانبی برای تبدیل داده ها
۸۳/۴/۶	۸۳/۴/۵	-- بسته بندی نسخه ۲
۸۳/۴/۸	۸۳/۴/۷	-- توزیع نسخه ۲
۸۳/۴/۱۲	۸۳/۴/۹	-- نصب نرم افزار
۸۳/۴/۲۰	۸۳/۴/۱۵	-- تبدیل و انتقال داده ها
۸۳/۴/۱۹	۸۳/۴/۱۳	-- تهیه راهنما و کمک به کاربر
۸۳/۴/۱۴	۸۳/۴/۱۳	-- آزمون پذیرش

۲-۳- محصولات یک تکرار : در این قسمت محصولات پروژه مشخص می شود. برای نمونه محصولات حاصل از تکرار اول از مرحله شناخت به شرح زیر است.

مستول	محصول	مجموعه محصولات
ع. کارگرنژاد	دیکشنری	مجموعه مدل‌های کاری
ح. مفیدی	مدل موردهای استفاده کاری	
ع. کارگرنژاد	شرح تکمیلی امور	
ح. مفیدی	تحقق موردهای استفاده کاری	
س. بینشگر	مستند چشم انداز پروژه	مجموعه نیازمندیها
ک. کریمی	مستند تقاضاهای افراد مرتبط	
س. بینشگر	شرح موردهای استفاده	
س. بینشگر	شرح تکمیلی	
س. بینشگر	شرح ویژگیهای نیازمندیها	
س. بینشگر	مدل موردهای استفاده	
غ. مدیری	برنامه ریزی اولین تکرار	مجموعه مدیریتی
غ. مدیری	برنامه ریزی پروژه	
غ. مدیری	زمانبندی پروژه	
غ. مدیری	لیست ریسک پروژه	
غ. مدیری	برنامه ریزی برای تعیین اندازه پروژه	
غ. مدیری و ک. کاوه	بررسی وضعیت	
غ. مدیری و ک. کاوه	بررسی اولین تکرار	
م. مقدم	برنامه ریزی برای عملیات ترکیب	
ک. کاوه	راهنمایی برای تعیین ویژگیهای نیازها	استانداردها و راهنمایها
ک. کاوه	محیط عملیات ترکیب	

برای تکرار اول از مرحله تشریح محصولات و خروجیهای تکرار به صورت زیر مشخص شده است:

مستول	محصول	مجموعه محصولات
ع. کارگرنژاد	دیکشنری	مجموعه مدل‌های کاری
ح. مفیدی	مدل موردهای استفاده کاری	

ع. کارگرنژاد	شرح تکمیلی امور	
ح. مفیدی	تحقق موردهای استفاده کاری	
س. بینشگر	مستند چشم انداز پروژه	مجموعه نیازمندیها
ک. کریمی	مستند تقاضاهای افراد مرتبط	
س. بینشگر	شرح موردهای استفاده	
س. بینشگر	شرح تکمیلی	
س. بینشگر	شرح ویژگیهای نیازمندیها	
س. بینشگر	مدل موردهای استفاده	
م. تفضلی	مدل تحلیل (آنالیز)	مجموعه طراحی
ن. قادری	شرح موردهای استفاده	
م. تفضلی	شرح تکمیلی	
ن. قادری	بررسی مدل موردهای استفاده	
م. تفضلی	تهیه نمونه اولیه از رابط کاربر	
ن. قادری	برنامه ریزی ایجاد بکپارچه	مجموعه پیاده سازی
م. تفضلی	ایجاد نمونه اولیه معماری	
ک. ملکوتی	آزمون برنامه ها	
ر. سلماسی	گزارش ارزیابی آزمون	
غ. مدیری	برنامه ریزی تکرار	مجموعه مدیریتی
غ. مدیری	اصلاح برنامه ریزی پروژه	
غ. مدیری	اصلاح زمانبندی پروژه	
غ. مدیری	اصلاح لیست ریسک پروژه	
غ. مدیری	برنامه ریزی برای تعیین اندازه پروژه	
غ. مدیری	بررسی وضعیت	
غ. مدیری	ارزیابی تکرار	
م. مقدم	برنامه ریزی برای آزمون برنامه ها	
س. پازوکی	راهنمای طراحی	استانداردها و راهنماها
س. پازوکی	راهنمای برنامه نویسی	
ر. رهبری	راهنمای آزمون	

م. چیتگر	راهنمای رابط کاربر	
----------	--------------------	--

برای تکرار دوم از مرحله ایجاد محصولات و خروجیهای تکرار به صورت زیر مشخص شده است:

مسئول	محصول	فرایند
س. پارسا م. کهن ن. نادران	مدل تحلیل (اصلاح) مدل طراحی (اصلاح) مستندات معماری نرم افزار (اصلاح)	مجموعه طراحی
ح. معمار د. پزشکچور س. دهکردی ف. سادات الحسنی	برنامه ریزی یکپارچه سازی بخش اصلی نسخه ۱ راه اندازهای آزمون گزارش ارزیابی آزمون	مجموعه پیاده سازی
ف. مدیری ف. مدیری ف. مدیری ف. مدیری ف. مدیری ف. مدیری ف. مدیری	برنامه ریزی تکرار ۲ برنامه ریزی پروژه (اصلاح) زمانبندی پروژه (اصلاح) لیست ریسک (اصلاح) ارزیابی وضعیت ارزیابی تکرار برنامه ریزی آزمون نرم افزار	مجموعه مدیریت

برای تکرار اول و دوم از مرحله انتقال محصولات و خروجیهای تکرار به صورت زیر مشخص شده است:

مسئول	محصول	مجموعه محصولات
ع. کارگرنژاد ح. مفیدی ع. کارگرنژاد ح. مفیدی ح. مفیدی ک. کریمی	دستورالعملهای نصب منوال کاربر داده های ترکیب رسانه مورد استفاده برای نصب قابلیتهای نسخه ۱ منوال آموزش کاربر	تکرار اول

ک. ملکوتی	آزمون پذیرش	
ع. کارگرنژاد	دستورالعملهای نصب	تکرار دوم
ح. مفیدی	منوال کاربر	
ع. کارگرنژاد	داده های ترکیب	
ر. رهنورد	برنامه تبدیل و انتقال داده ها	
ح. مفیدی	رسانه مورد استفاده برای نصب	
ح. مفیدی	قابلیتهای نسخه ۲	
ک. کریمی	منوال آموزش کاربر	
ک. ملکوتی	آزمون پذیرش	

۳-منابع: شامل لیست منابع مورد نیاز برای تکرار است. در برنامه ریزی اولین تکرار از سیستم ثبت نام مشخص شده است که نیازها، قالب کاری، موردهای استفاده عمده و برنامه ریزی کلی پروژه مشخص خواهد شد. در انتهای این تکرار کالج ایکس می بایست درمورد ادامه و یا عدم ادامه پروژه تصمیم گیری کند.

منابع انسانی « افراد مورد نیاز برای تکرار اول از مرحله شناخت عبارتند از:

- مدیریت پروژه مدیر پروژه : آقای ریسی
- گروه مدلسازی سیستم کاری مدیر: رییس زاده تحلیلگر: فکوری مرورگر: هوشیار
- مهندسی سیستم مهندس سیستم : حسنی تحلیلگر: رضایی معمار: بنایی
- مهندسی نرم افزار مهندس فرایند: همتی لیست بقیه افراد دربرنامه ریزی پروژه
- گروه آزمون مدیر: مسعود مدیری، آزمون کننده سیستم: مریم

سرفراز

در تکرارهای ۱ و ۲ از مرحله انتقال در سیستم ثبت نام منابع انسانی عبارت است از:

- مدیریت پروژه مدیر پروژه : آقای ریسی مدیر نصب: حسن حسن زاده
- مهندسی نرم افزار مهندس فرایند: همتی لیست بقیه افراد دربرنامه ریزی پروژه
- گروه آزمون مدیر: مسعود مدیری
- نصب و راه اندازی مدیر نصب: آرش قصاب باشی

منابع مالی : جدول زیر شاخص بودجه تعیین شده برای اولین تکرار از مرحله تشریح است. برای تکرار اول ۱۶۹۵۰۰۰۰ تومان بودجه تخصیص داده شده است.

پروژه ثبت نام : بودجه تکرار ۱ شناخت

افراد

فعالیتها	تلاش (نفر روز)	هزینه (تومان)
مدلسازی سیستم کاری	۴۰	۲۲۴۰۰۰۰۰
تعیین نیازها	۵۰	۲۸۰۰۰۰۰۰
مدیریت	۳۰	۱۶۸۰۰۰۰۰
بررسی محیط	۲۵	۱۳۶۰۰۰۰۰
مجموع	۱۴۵	۸۱۴۰۰۰۰۰

سایر هزینه ها (جزئیات در فرم ضمیمه)

فعالیتها	تلاش (نفر روز)	هزینه (تومان)
سفر و محل اقامت		۰۰۰۰۰۰
حسن انجام وظایف		۰۰۰۰۰۰
سرویسها		۸۰۰۰۰۰
لوازم و مواد مصرفی		۹۶۰۰۰۰
بررسی محیط		۱۶۰۰۰۰۰
مجموع		۳۳۶۰۰۰۰
بودجه کل		۸۴۵۶۰۰۰۰

برای تکرار اول از مرحله تشریح ۸۰۶۴۰۰۰۰ تومان بودجه به شرح زیر تخصیص داده شده است.

پروژه ثبت نام : بودجه تکرار ۱ تشریح

افراد

فعالیتها	تلاش (نفر روز)	هزینه (تومان)
مدلسازی سیستم کاری	۶	۳۳۶۰۰۰۰
تعیین نیازها	۱۰	۵۶۰۰۰۰۰
تحلیل و طراحی	۲۵	۶۰۰۰۰۰۰
پیاده سازی	۲۵	۶۰۰۰۰۰۰
آزمون	۲۰	۳۲۰۰۰۰۰

مدیریت	۳۰	۱۶۸۰۰۰۰۰
بررسی محیط	۱۰	۵۶۰۰۰۰۰
مجموع	۱۲۶	۶۰۹۶۰۰۰۰
سایر هزینه ها (جزئیات در فرم ضمیمه)		
فعالیتها	تلاش (نفر روز)	هزینه (تومان)
سفر و محل اقامت		۲۴۰۰۰۰۰
حسن انجام وظایف		۰۰۰۰۰۰۰
سرویسها		۳۳۶۰۰۰۰
لوازم و مواد مصرفی		۱۶۰۰۰۰۰
سایر هزینه ها		۲۷۲۰۰۰۰
مجموع		۱۰۰۸۰۰۰۰
بودجه کل		۸۰۶۴۰۰۰۰

برای تکرار دوم از مرحله ایجاد ۹۳۲۸۰۰۰۰ تومان بودجه به شرح زیر تخصیص داده شده است.

پروژه ثبت نام : بودجه تکرار ۱ ایجاد		
افراد		
فعالیتها	تلاش (نفر روز)	هزینه (تومان)
تحلیل و طراحی	۲۳	۱۲۸۸۰۰۰۰
پیاده سازی	۵۷	۳۱۹۲۰۰۰۰
آزمون	۳۰	۱۶۸۰۰۰۰۰
مدیریت	۲۰	۱۱۲۰۰۰۰۰
بررسی محیط	۱۴	۷۸۴۰۰۰۰
مجموع	۱۴۴	۸۰۶۴۰۰۰۰
سایر هزینه ها (جزئیات در فرم ضمیمه)		
فعالیتها	تلاش (نفر روز)	هزینه (تومان)
سفر و محل اقامت		۰۰۰۰۰۰۰
حسن انجام وظایف		۰۰۰۰۰۰۰
سرویسها		۴۰۰۰۰۰۰
لوازم و مواد مصرفی		۳۲۰۰۰۰۰

۵۴۴۰۰۰۰	سایر هزینه ها
۱۲۶۴۰۰۰۰	مجموع
۹۳۲۸۰۰۰۰	بودجه کل

برای تکرار اول از مرحله انتقال ۵۰۵۶۰۰۰۰ تومان بودجه به شرح زیر تخصیص داده شده است.

پروژه ثبت نام : بودجه تکرار ۱ تشریح		
افراد		
فعالیتها	تلاش (نفر روز)	هزینه (تومان)
عملیات نصب	۴۵	۲۵۲۰۰۰۰۰
مدیریت	۱۵	۸۴۰۰۰۰۰
مجموع	۶۰	۳۳۶۰۰۰۰۰
سایر		
فعالیتها	تلاش (نفر روز)	هزینه (تومان)
سفر و محل اقامت		۰۰۰۰۰۰۰
حسن انجام وظایف		۰۰۰۰۰۰۰
سرویسها		۷۲۰۰۰۰۰
لوازم و مواد مصرفی		۴۳۲۰۰۰۰۰
سایر هزینه ها		۵۴۴۰۰۰۰
مجموع		۱۶۹۵۰۰۰۰۰
کل بودجه		۵۰۵۶۰۰۰۰۰

۲-۳ منابع ابزار و تسهیلات : در مرحله شناخت امکانات سخت افزار، نرم افزار و شبکه مورد نیاز به شرح ذیل است:

- ۴ عدد میکرو کامپیوتر با ویندوز ۲۰۰۰، آفیس ۲۰۰۰، رشنال سوییت ۲۰۰۲
- یک دستگاه چاپگر لیزری اچ-پی
- لرتباط با کامپیوتر سرویس دهنده آموزش

در تکرار ۱ از مرحله تشریح امکانات مورد نیاز به شرح ذیل است:

- ۴ عدد میکرو کامپیوتر با ویندوز ۲۰۰۰، آفیس ۲۰۰۰، رشنال سوییت ۲۰۰۲
- سه دستگاه چاپگر لیزری اچ-پی
- لرتباط با کامپیوتر سرویس دهنده آموزش

در تکرار دوم از مرحله ایجاد ابزار کامپیوتری زیر مورد نیاز هستند:

- ۱۸ عدد میکرو کامپیوتر با ویندوز ۲۰۰۰، آفیس ۲۰۰۰، رشنال سوییت ۲۰۰۲

- سه دستگاه چاپگر لیزری اچ-پی

- لرتباط با کامپیوتر سرویس دهنده آموزش

در تکرار های ۱ و ۲ از مرحله انتقال، نیازی به منبع جدید وجود ندارد. از وب سایت موجود برای انتقال برنامه های سیستم ثبت نام استفاده خواهد شد. برای چاپ منوالهای کاربر با چاپخانه قرار داد خواهد شد.

۳-موردهای استفاده : در اولین تکرار از مرحله شناخت در سیستم ثبت نام اغلب موردهای استفاده اصلی و بازیگرها و گردش کار آنها مشخص خواهند شد. طرح و پیاده سازی موردهای استفاده در تکرار بعدی مشخص خواهد شد. موردهای استفاده ای که در تکرار اول از مرحله تشریح سیستم ثبت نام توسعه داده خواهند شد شامل اقلام ذیل می باشند:

۱. ورود به سیستم

۵- تهیه اطلاعات دانشجویها

۲. ثبت نام در درسها

۶- تهیه رابط برای بانک اطلاعاتی کاتالوگ دروس

۳. تهیه اطلاعات مدرسها

۷- تهیه رابط برای سیستم آتی

۴. ثبت نام برای دوره

در تکرار دوم از مرحله ایجاد هدف اصلاح و تکمیل کد ایجاد شده در تکرار ۱ از این مرحله است. در این تکرار موردهای استفاده زیر، مورد بررسی بیشتری قرار گرفته و بطور کاملتری پیاده سازی می شوند.:

۱- بستن ثبت نام

۲- رابط برای بانک اطلاعاتی کاتالوگ دروس

۳- رابط برای سیستم مالی

در مرحله های ۱ و ۲ از مرحله انتقال موردهای استفاده ایجاد نمی شوند. اما، موردهای استفاده

اصلی که برای نسخه ۱ پیش بینی شده و در این مراحل باید آماده باشند عبارتند از:

۱. ورود به سیستم

۵- تهیه اطلاعات دانشجویها

۲. ثبت نام در درسها

۶- رابط برای بانک اطلاعاتی کاتالوگ دروس

۳. ثبت نام برای دوره

۷- رابط برای سیستم مالی

۴. تهیه اطلاعات مدرسها

در نسخه ۲ علاوه بر مورد های استفاده فوق سه مورد دیگر به شرح ذیل به امکانات سیستم ثبت نام اضافه می شود:

- ارسال نمره دانشجویا - مشاهده نمره - انتخاب درس برای تدریس

۴- معیارهای ارزیابی : هدف اصلی از تکرار اول از مرحله شناخت در سیستم ثبت نام، تعریف سیستم در حدی است که بتوان در مورد انجام یا عدم انجام پروژه تصمیم گیری کرد. در تکرار اول در مرحله تشریح، هدف تکمیل تحلیل و طراحی مورد های استفاده انتخاب شده و تهیه یک نمونه اولیه قابل اجرا از معماری نرم افزار است.

هدف اصلی از تکرار ۲ در مرحله ایجاد، تکمیل نسخه ۱ نرم افزار است. علاوه بر این معیارهای زیر می بایست مورد ارزیابی قرار گیرد:

- کلیه نقصهای گزارش شده در نسخه ۱ می بایست بر طرف شود.
- بازده کاربر در مورد نسخه ۱ می بایست بررسی و مشکلات اصلی بر طرف شود.
- کلیه نیازها و مورد های استفاده که برای نسخه ۱ مشخص شده بودند باید پیاده سازی و تست شوند
- کلیه زمانبندیها و نیازهای کیفی مشخص شده در مستند توصیفهای تکمیلی باید رعایت شوند.
- کلیه آزمونها همانگونه که در برنامه ریزی آزمون مشخص شده باید انجام شده باشند.
- نتیجه کلیه آزمونها باید مجدداً بررسی و مرور شود.
- نسخه شماره ۱ باید مورد تایید مدیر پروژه و مدیریت واحد کامپیوتر کالج ایکس قرار گیرد.
- هدف عمده در تکرارهای اول و دوم از مرحله انتقال نصب نسخه های شماره ۱ و ۲ به ترتیب از نرم افزار سیستم ثبت نام است. علاوه بر این قبل از خاتمه این دو مرحله معیارهای زیر می بایست در ارتباط با نسخه های ۱ و ۲ می بایست ارزیابی شود:

- هیچگونه نقص عمده ای نباید در نسخه های ۱ و ۲ وجود داشته باشد.

- نسخه ۲ می بایست یک هفته قبل از شروع ثبت نام دانشجویا در ترم پاییزی، نصب شده باشد.

- نسخه های شماره ۱ و ۲ می بایست رسماً مورد قبول کاربرها و مسئولین کالج ایکس قرار گیرد.

- کاربرها می بایست کاملاً آموزش داده شوند.

- سایتهای وبی می بایست برای پشتیبانی نرم افزاری ایجاد شوند.

- منوالهای کاربر می بایست از طریق کتابخانه کالج توزیع شود.

- می بایست امکانات و منابع لازم جهت پشتیبانی مستمر کارها فراهم شود.

فصل ۴

مراحل ایجاد بانک اطلاعاتی

۴-۱ مقدمه

آنچه که در استانداردهای یو.ام.ال و متدولوژیهای شیئی گرا مبتنی بر آن مورد تاکید قرار نگرفته، مدلسازی بانک اطلاعاتی است. به این مدل در زمان طراحی لفظ مدل داده ها اطلاق می شود. البته این مشکل در ژوئن ۲۰۰۲ با ارائه امکاناتی تحت عنوان مدلساز داده ها^۱ حل شده است. می توان با استفاده از امکاناتی مدلساز داده ها به طور اتوماتیک مدل اشیا را تبدیل به مدل داده ها و بالعکس نمود. سوال اینجا است آیا لازم است برای هر موجودیت یک کلاس در نظر گرفته شود؟ اگر لازم نیست چگونه موجودیتها را در سایر مدلها باید مشخص کرد؟

در این فصل چگونگی تبدیل مدل اشیا به مدل داده ها و بالعکس مورد بررسی قرار می گیرد. همچنین روشهای تولید اتوماتیک کد برای تبدیل مدل داده ها به کد برنامه و بالعکس مطرح خواهد شد.

۴-۲ تبدیل مدل اشیا به مدل داده ها

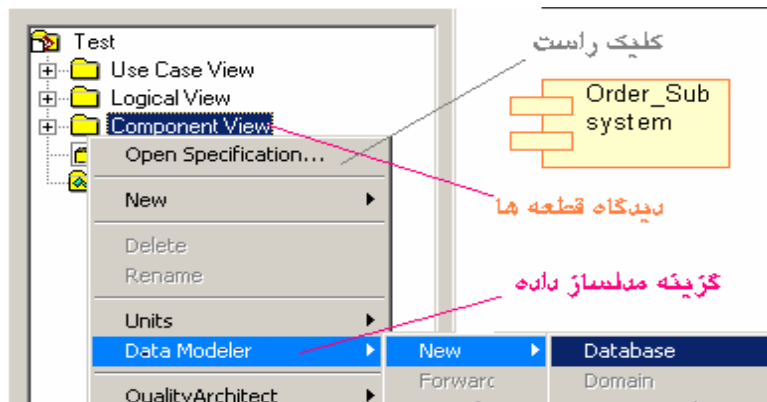
جهت ایجاد بانک اطلاعاتی در مرحله طراحی می بایست مدل داده ها ایجاد شود. مدل داده ها بیانگر چگونگی ارتباط بین موجودیتها است. هر موجودیت در عمل تبدیل به یک جدول یا فایل از سیستم بانک اطلاعاتی می شود. مدل داده ها از مدل ارتباطی کلاسها استخراج می شود. هر کلاس از نوع موجودیت تبدیل به یک جدول می شود. مراحل تبدیل به شرح ذیل است.

۱. **ایجاد بانک اطلاعاتی:** در دیدگاه قطعه ها می بایست قطعه ای از نوع بانک

اطلاعاتی ایجاد نمود. برای این منظور مطابق شکل زیر از گزینه مدلساز داده ها

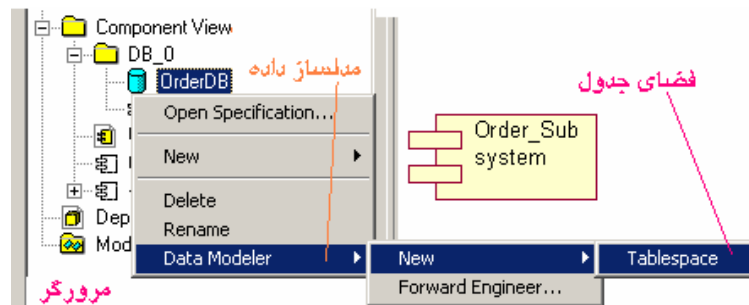
مطابق شکل ذیل استفاده می شود.

۲.



شکل ۱- تعیین قطعه بانک اطلاعاتی

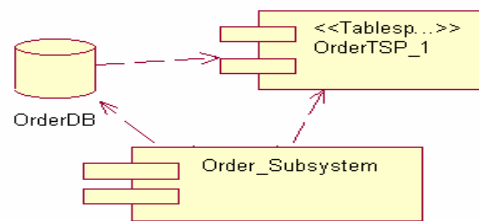
می توان برای قطعه بانک اطلاعاتی فضاهای جدول متفاوت ایجاد نمود. برای این منظور کافی است در داخل مرورگر بر روی قطعه بانک اطلاعاتی گزینه مدلساز داده را انتخاب و هر تعداد که نیاز است فضا برای دسته بندی جدولها مشخص کنید. برای نمونه به شکل زیر توجه نمایید.



شکل ۲- ایجاد فضای جداول

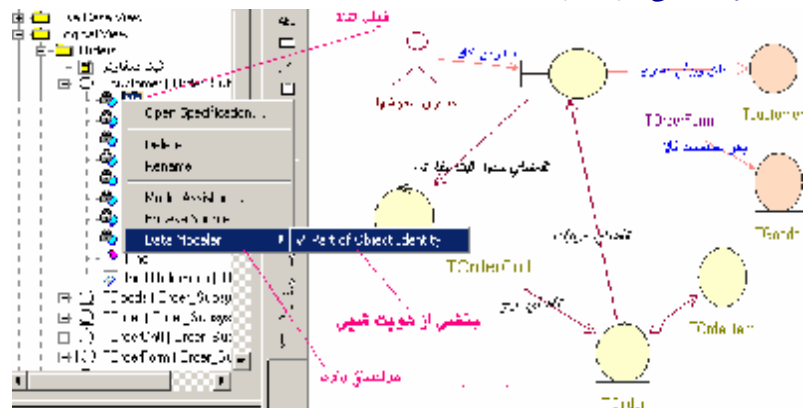
برای مشاهده مدل بانک می توانید از داخل مرورگر، قطعه بانک اطلاعاتی و فضای جدولهای مربوطه را به داخل مدل قطعه ها انتقال دهید.

سیستم ثبت سفارشی



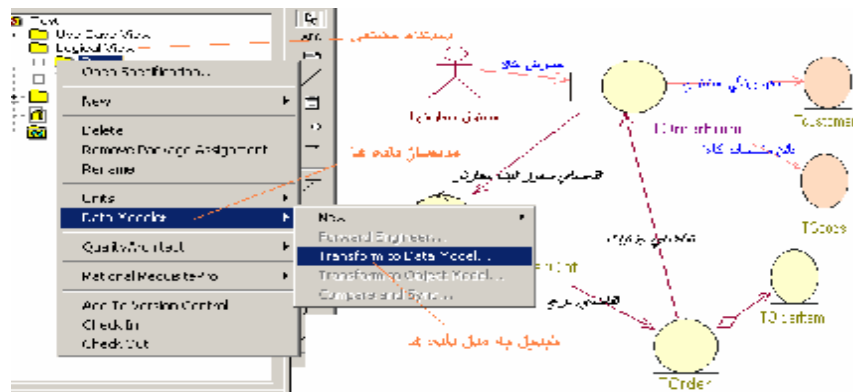
شکل ۳- مدل قطعه ها همراه با قطعه بانک اطلاعاتی

۲- ایجاد اسکیمای: برای تبدیل مدل اشیاء به مدل داده‌ها، اولاً مدل اشیاء باید تحت یک بسته قرار داشته باشد و ثانياً کلاسهای موجودیت باید از نوع مانا یا Persistence تعیین شده باشند. جهت مشخص شدن کلید اصلی برای جداول حاصل از کلاسها باید برای هر کلاس به تعریف کلاس در بخش دیدگاه منطقی از مرورگر ارجاع نموده، فیلد مورد نظر را مشخص نموده و با استفاده از مدل‌ساز داده‌ها آنرا به عنوان کلید اصلی مشخص کرد. برای نمونه در شکل ذیل فیلد no به عنوان کلید اصلی مشخص شده است. چنانچه بیش از یک فیلد به عنوان کلید اصلی تعریف شوند، ترکیب آنها نمایانگر ملید اثلی خواهد بود.



شکل ۴- تعیین هویت برای شی و کلید اصلی برای جدول

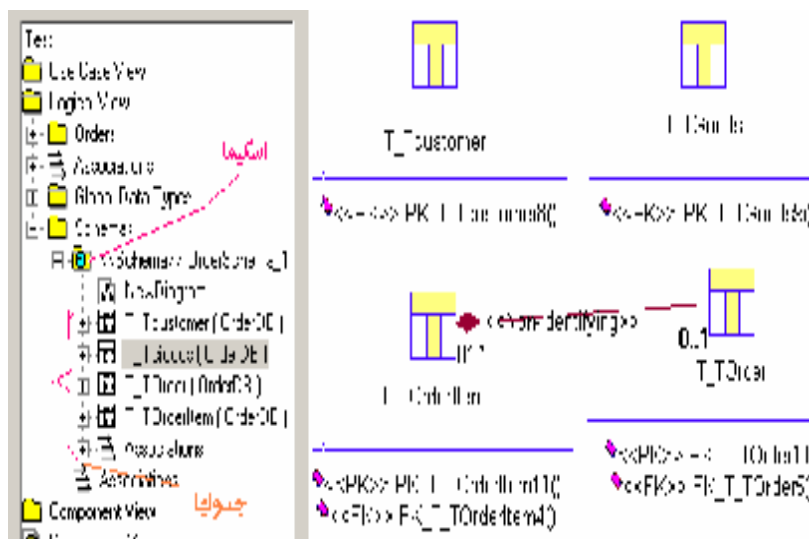
برای ایجاد یک اسکیمای کلی می‌بایست در دیدگاه منطقی ابتدا از طریق مدل‌ساز داده‌ها یک اسکیمای کلی ایجاد شود. برای نمونه به شکل زیر توجه نمایید.



شکل ۵- ایجاد اسکیمای و ایجاد جداول برای مدل داده‌ها

به این ترتیب کلاسها از نوع موجودیت که با ویژگی مانا مشخص شده‌اند، تبدیل به جداولی

تحت اسکیمای ایجاد شده خواهند شد. برای ایجاد مدل داده ها کافیست که با استفاده از گزینه مدلسازی داده ها برای اسکیمای ایجاد شده یک صفحه خالی برای ایجاد مدل داده ایجاد شود و سپس با استفاده از ماس جداول تحت اسکیمای داخل مرورگر بر روی این صفحه خالی انتقال داده شوند. ارتباط بین جداول نیز بر اساس ارتباطهای موجود در مدل ارتباطی اشیاء ایجاد خواهد شد. برای نمونه به شکل زیر توجه نمایید. در این شکل با فزردن دکمه راست ماس بر روی اسکیمای ایجاد شده برای بسته سفارش و انتخاب گزینه مدلساز داده یک صفحه خالی برای دیاگرام مدل داده ها تحت عنوان NewDiagram ایجاد شده است. سپس بدر ضمن پایین نگهداشتن دکمه چپ ماس بر روی جداول هر یک را به تنهایی به روی صفحه خالی حاصل از NewDiagram انتقال داده اند تا اینکه نهایتاً دیاگرام مدل داده ها با چهار جدول به صورت زیر مشخص شده است. برای هر جدول کایدیهای اصلی بر اساس کلیدهای تعیین شده در اشیاء مربوطه، مشخص شده است.



شکل ۶- استخراج مدل داده ها

۳- **تولید کد**: اکنون برای تولید کد کافی است که بسته مورد نظر از دیدگاه منطقی را در مرورگر انتخاب نمایید. اکنون با فشردن دکمه سمت راست ماس بر روی بسته می توان از طریق گزینه مدلساز داده، عمل مهندسی به جلو را آغاز و مدل داده ها را تبدیل به دستورالعملهای SQL نمود. به این ترتیب برای مدل ارائه شده در شکل فوق دستورالعملهای زیر ایجاد می شود.

صاحب جزئیات مشتریها است.
 CREATE TABLE T_Tcustomer (
 شماره مشتری
 No SMALLINT NOT NULL,
 تاریخ پیوستن به لیست مشتریها

Date SMALLINT NOT NULL,
نام مشتری
Name SMALLINT NOT NULL,
Address SMALLINT NOT NULL,
شماره تلفن
tel SMALLINT NOT NULL,
شماره موبایل
Mobile SMALLINT NOT NULL,
شماره فکس
Fax SMALLINT NOT NULL,
جمع کل مبلغ خرید
TotalPurchase SMALLINT NOT NULL,
CONSTRAINT PK_T_Tcustomer8 PRIMARY KEY NONCLUSTERED (No)
)
GO ...

فصل ۱۲

مدلسازی وب

۱-۱۲ مقدمه

سسس

۲-۱۲ ویژگیهای سیستمها تحت وب

وجود صفحه های اچ-تی-ام-ال که در متن برنامه ها معمولاً در قالب انواع کلاس مطرح نیستند، عامل اصلی در بوجود آمدن امکاناتی خاص طراحی سیستمها تحت وب بوده است. صفحه های اچ-تی-ام-ال را ابرمتن یا متن برتر می نامند زیرا، امکان پیمایش از طریق مسیرهای ارتباطی پیوند دهنده بین صفحه ها که تحت بخشهای مشخص شده با رنگ مجزا در متن آنها پنهان است را فراهم می کنند. به عبارت دیگر می توان از طریق یک متن به متن های دیگر مراجعه نمود. در ارتباط با ایجاد ابرمتنها سه مشکل اصلی مطرح است:

۱. ارجاعات مکرر به صورت تو در تو به متن صفحه های اچ-تی-ام-ال ممکن است موجب فراموش شدن متن اولیه که اولین ارجاع از آن آغاز گردیده، شود.
۲. کلی گرایی و عمومیت بخشیدن به متن صفحه های اچ-تی-ام-ال بواسطه تعدد انواع متقاضی ها در عمل اینگونه متنها را بلااستفاده می نماید
۳. گستردگی وب، مسائل کنترل دسترسی و امنیتی داده ها را در قالب پیچیده تری مطرح می کند

بدین ترتیب برای جلوگیری از مشکلات فوق الذکر، می بایست در طرح سیستمهای تحت وب، سه نکته ذیل را مد نظر داشت:

۱. باید بتوان مسیر پیمایش بین صفحات وب را مشخص کرد. تعیین مسیر پیمایش را در اصطلاح طراحی ناوبری می گویند.

۲. باید مطالب بر اساس نوع متقاضی و وابسته به نیازها و خواسته های وی، تنظیم گردد. این مقوله در قالب تخصصی سازی سایتهای وب مطرح است.

۳. سیستمهای تحت وب نسبتاً در معرض خطر بیشتری بوده، احتمال دسترسی غیر مجاز و شکستن سطوح امنیتی در آنها بالاتر است تا حدی که برای محیطهای نظامی و امن پیشنهاد نمی شوند.

تعیین نحوه پردازش صفحات وب نکته قابل توجه دیگری است که می تواند در سمت کاربر یا سرویس دهنده باشد.

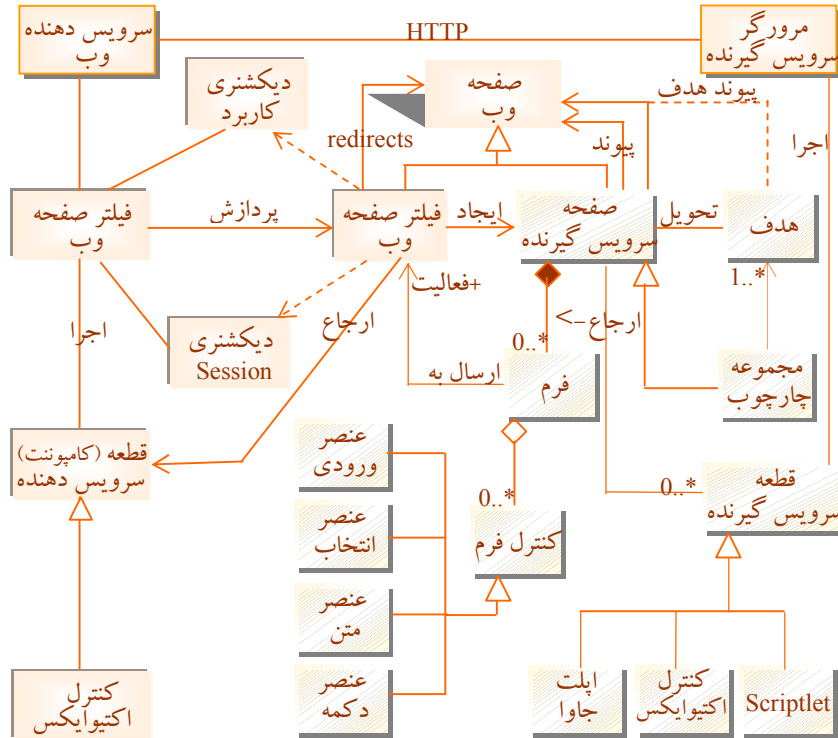
تحلیل طراحی سیستمها در سطح وب اغلب برای سیستمهای گسترده تجارت الکترونیک مطرح است. در حالت کلی زبانهای اسکریپتی برای ایجاد سیستمهای نرم افزاری تحت وب مورد استفاده قرار می گیرند. معمولاً، اینگونه سیستمها بواسطه عدم کفایت زبانهای اسکریپتی در ایجاد برنامه های بزرگ و مطمئن، نسبتاً کوچک هستند و در عمل نیازی به تحلیل و طراحی اصولی و دقیق ندارند.

در یو-ام-ال هر صفحه وب به عنوان یک شیء با کلیدهای صفحه وب مدلسازی می شود. برای نمایش ارتباطات بین صفحات از شیء ای با کلید پیوند استفاده می شود. برای سایر موارد مطرح در صفحات وب نیز هر یک به نوبه خود اشیاء با کلیدهای مربوطه تعریف شده است.

همانگونه که در بالا توضیح داده شد، اطلاعات صفحه های وب باید بر اساس نوع کاربر ایجاد و در اختیار وی قرار داده شود. لذا، در تحلیل سیستمها باید انواع کاربر به عنوان عاملی جهت تعیین صفحه های وب مطرح باشد. در یو-ام-ال کاربرها طبقه بندی نمی شوند. تاکید بر نوع کاربرها و نقش آنها در سیستم در متدولوژی دلبیو-اس-دی-ام بخصوص مطرح است. قبل از بیان چگونگی مدلسازی سیستمها در یو-ام-ال، لازم است آشنایی کلی با ساختار کلی برنامه ها در سطح وب ایجاد شود. در بخش بعد این ساختار بررسی می شود.

۳-۱۲ ساختار کلی

در این بخش ساختار کلی معماری سیستمها تحت وب مورد بررسی قرار خواهد گرفت. اصولاً اینگونه سیستمهای نرم افزاری در قالب سرویس دهنده و سرویس گیرنده مطرح هستند. سرویس دهنده ها معمولاً از ابزاری مثل آی-آی-اس، PWS و یا آپاچی می باشند.



شکل ۱-۱۲ ساختار کلی سیستمها تحت وب

۴-۱۲ صفحه ها

صفحه های وب در قالب صفحه های سرویس دهنده و صفحه های سرویس گیرنده به ترتیب با کلیشه های <<server page>> و <<client page>> مشخص می شوند. از آنجاییکه صفحه های وب دارای عملیات و ویژگیهای خاص خود می باشند، هر صفحه به عنوان یک شیئی از نوع یک کلاس مشخص می شود. صفحه سرویس دهنده معمولاً در ارتباط با کلاسها و کتابخانه ها قرار داشته و عملیات اصلی بر عهده آن است. اسکریپتهای سمت سرویس گیرنده و ساختار رابط کاربر به عنوان

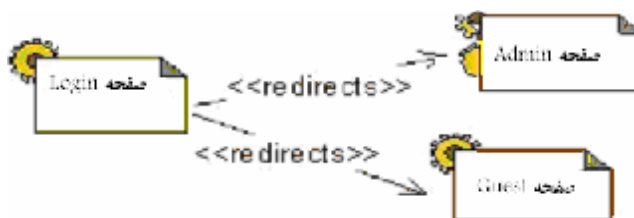
اجزاء کلاس سرویس دهنده در نظر گرفته نمی شوند. صفحه سرویس دهنده می تواند با قطعات روی ایستگاه کاری سرویس دهنده وب در ارتباط باشد. در دیاگرام کلاسها، قطعات و کلاسهایی که روی سرویس دهنده اجرا می شوند در کنار صفحه های سرویس دهنده که از آنها استفاده می کنند نمایش داده می شوند. ویژگی های صفحه سرویس گیرنده متغیرها و توابعی هستند که در سمت سرویس گیرنده اجرا می شوند. صفحات سرویس گیرنده هم در ارتباط با قطعاتی نظیر اپلتهای جاوا و اکتیو ایکس ها هستند که روی سرویس گیرنده اجرا می شوند.

یک نکته اساسی، نمایش چگونگی برقراری ارتباط در بین صفحه های وب است. برای نمونه چنانچه اجرای صفحه سرویس دهنده موجب ایجاد صفحه سرویس گیرنده شود، رابطه بین صفحه ها در قالب کلیشه <<builds>> مشخص می شود. همانگونه که در شکل ۱ مشخص شده است، این کلیشه یک طرفه بوده و از سمت صفحه سرویس دهنده به سرویس گیرنده است. رابطه <<builds>> نشان می دهد که صفحه سرویس دهنده، صفحه سرویس گیرنده را می سازد. ممکن است صفحه سرویس دهنده دو صفحه مجزای سرویس گیرنده را بسازد.



شکل ۱۲-۲ رابطه ساخت بین صفحه سرویس گیرنده و صفحه سرویس دهنده

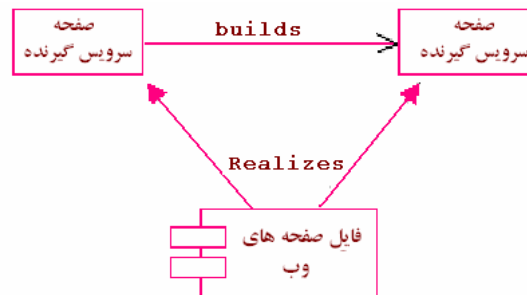
قابلیت دیگر برنامه های وب امکان انتقال عملیات درخواست شده به صفحه سرویس دهنده دیگری است. این ارتباط می تواند در مدل به صورت ارتباطی با قالب <<redirect>> نمایش داده شود. شکل زیر این نوع ارتباط را نشان می دهد.



شکل ۱۲-۳ نحوه ساخت دو صفحه سرویس گیرنده از یک صفحه سرور

در واقع با اجرا شدن redirect ادامه کار به صفحه دیگری منتقل می شود که ممکن است در ضمن این کار پارامترهایی هم منتقل گردد. رابطه اساسی و مهم بین صفحه سرویس گیرنده و صفحه

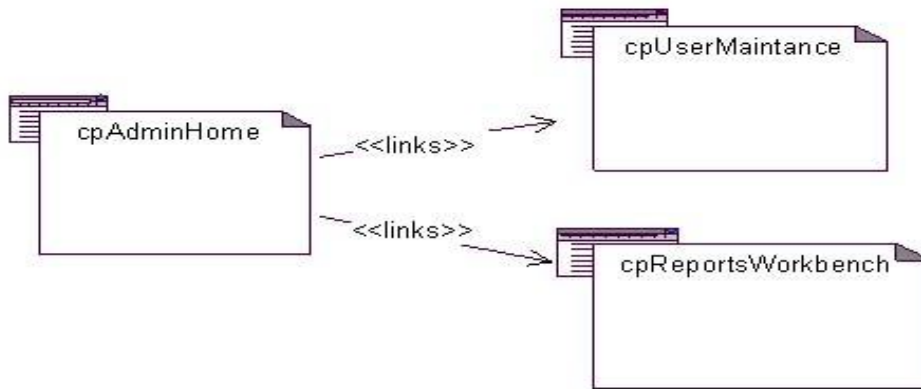
سرویس دهنده در دیاگرام پیاده سازی مشخص می شود.^{۱۲} قطعاتی که در دیاگرام پیاده سازی وجود دارند سیستم را به صورت قطعات مجزا نشان می دهد. (زیر سیستم ها) که در این مورد صفحه وب هستند. در واقع در دیاگرام پیاده سازی یک قطعه، یک فایل واقعی است که می تواند به وسیله سرور وب تقاضا شود و در نهایت منجر به صفحه سرور یا صفحه سرویس گیرنده یا هر دو گردد. شکل زیر این رابطه را نشان می دهد:



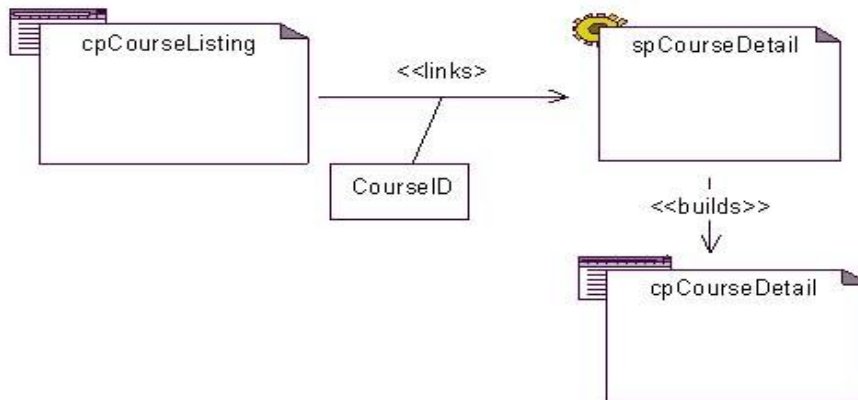
شکل ۱۲-۴ رابطه بین صفحه وب با صفحات سرویس دهنده و سرویس گیرنده

رابطه پیوند مسیر پیمایش بین صفحه های وب را مشخص می کند. صفحه های سرویس گیرنده گاهی اوقات مرتبط با صفحاتی هستند که ممکن است سرویس گیرنده یا سرویس دهنده باشند. البته در نهایت به وسیله مرورگر سرویس گیرنده است که یک صفحه تقاضا می شود. اگر صفحه سرویس دهنده مورد تقاضا باشد، صفحه سرویس دهنده پردازش شده و صفحه سرویس گیرنده را می سازد. در غیر این صورت، سرویس دهنده وب به سادگی فایل صفحه مورد نظر را به مرورگر ارسال می کند. رابطه پیوند همانگونه که در شکل بعدی مشخص شده، توسط کلیکشه <<links>> مشخص می شود.

شاید لازم نباشد که پیوند به صفحه های خارج از سیستم مورد نظر مدل شود. البته همه اینها به نظر طراح بستگی دارد. ارتباط <<link>> ممکن است یک ارتباط دو جهته باشد. ارتباط <<link>> از صفحات سرویس دهنده محسوس نیست و در نظر گرفته نمی شود. (شاید اصلا به کار نرود). اگر پیوند دارای پارامترهایی باشد این پارامترها به عنوان ویژگیهای پیوند در نظر گرفته و مدل می شوند.



پیوند بدون پارامتر (صفحات سرویس گیرنده می توانند شبیه به هم باشند)



پیوند با پارامتر

۲-۱-۵- قطعات^{۱۳}

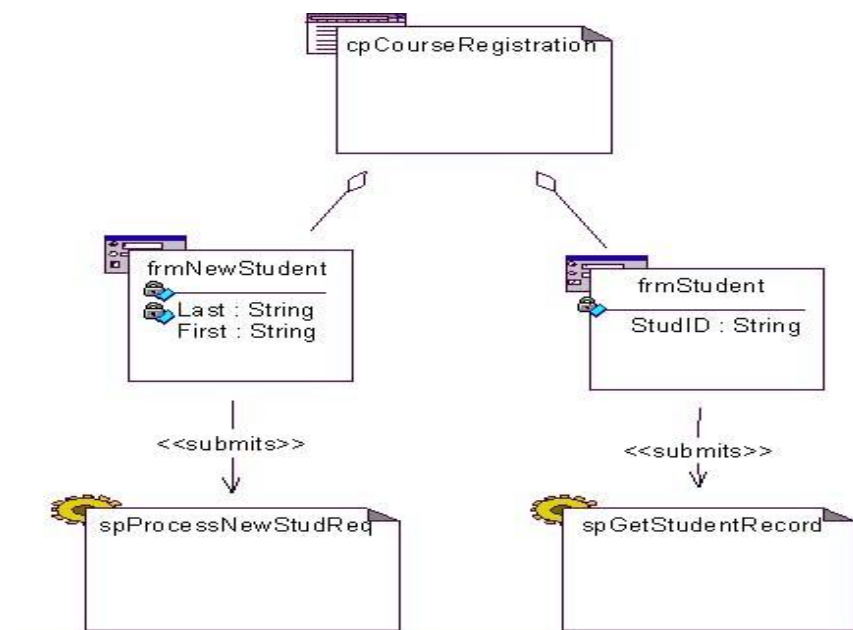
قطعاتی که در برنامه های وب استفاده می شوند، همانگونه که قبلا گفته شد ممکن است در سمت سرور یا سرویس گیرنده اجرا شود. برای این منظور از قالب های <<Server component>> و <<Client component>> استفاده می شود.

۳-۲-۱-۵- فرم ها

قالب دیگری نیز برای جداسازی و مشخص کردن فرم های اچ.تی.ام. ال استفاده می شود. فرم ها به درستی مجزای از صفحات سرویس گیرنده هستند و ویژگی هایی اضافه بر صفحات سرویس گیرنده دارند. علاوه بر این ممکن است چندین فرم در صفحه وجود داشته باشند که هر کدام صفحه فعال خاص خود را خواهد داشت. لذا برای مدل سازی فرم ها از قالب دیگری استفاده می کنیم که به نام <<Form>> شناخته می شود.

فیلد های روی فرم به عنوان ویژگی های کلاس فرم در نظر گرفته می شود. متد هایی که در صفحه سرویس گیرنده قرار دارند به همه ویژگی هایی که روی فرم شامل این صفحه وجود دارند دسترسی خواهند داشت. رابطه مناسب بین صفحه سرویس گیرنده و فرم رابطه شامل بودن یا در بر داشتن است. صفحه سرویس گیرنده شامل یک سری فرم هاست. هر فرم صفحه وبی را (معمولا از نوع صفحه سرور) تعریف می کند که اطلاعات فرم را دریافت و آنها را پردازش می کند.

قالب <<Submit>> ارتباط بین فرم و صفحه ای که آن را پردازش می کند نشان می دهد. شکل زیر:

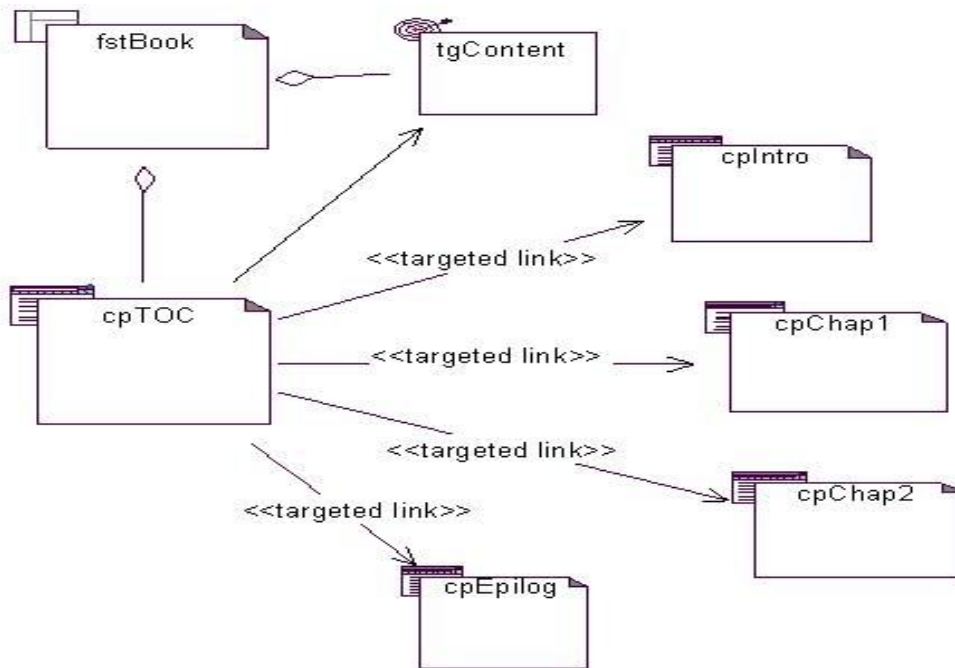


فرم های ارسال شده به صفحات سرور

Frameset-5-1-2-4

همانگونه که قبلا گفته شد یکی دیگر از جنبه های اینترفیس کاربردر برنامه های وب فریم است که قابلیت نمایش چندین صفحه را به طور همزمان فراهم می آورد . فریم ها در اچ.تی.ام. ال با تعریف یک فریم ست¹⁴ پیاده سازی می شوند . فریم ست فریم ها را نامگذاری می کند که در هرکدام صفحه وب می تواند قرار گیرد. برای فریم ست ها هم از قالب جدیدی به نام <<Frameset>> استفاده می شود. درمدل طراحی کلاس با قالب ،<<Frameset>> می تواند شامل همان ارتباطات صفحه سرویس گیرنده باشد .

¹⁴ Frameset



نحوه استفاده از فریم ست ها و Targetها

۵-۲-۱-۵- سایر Stereotype ها

در حال حاضر توسعه یو.ام.ال (یا یو.ام.ال توسعه یافته برای وب) ادامه دارد و قالب های زیر مورد مطالعه و توجه است:

یک کلاس که یک ساختار سلسله مراتبی از داده دارد و می تواند بین سرویس گیرنده و سرور جا به جاشود را با `<<XML>>` نشان می دهد.

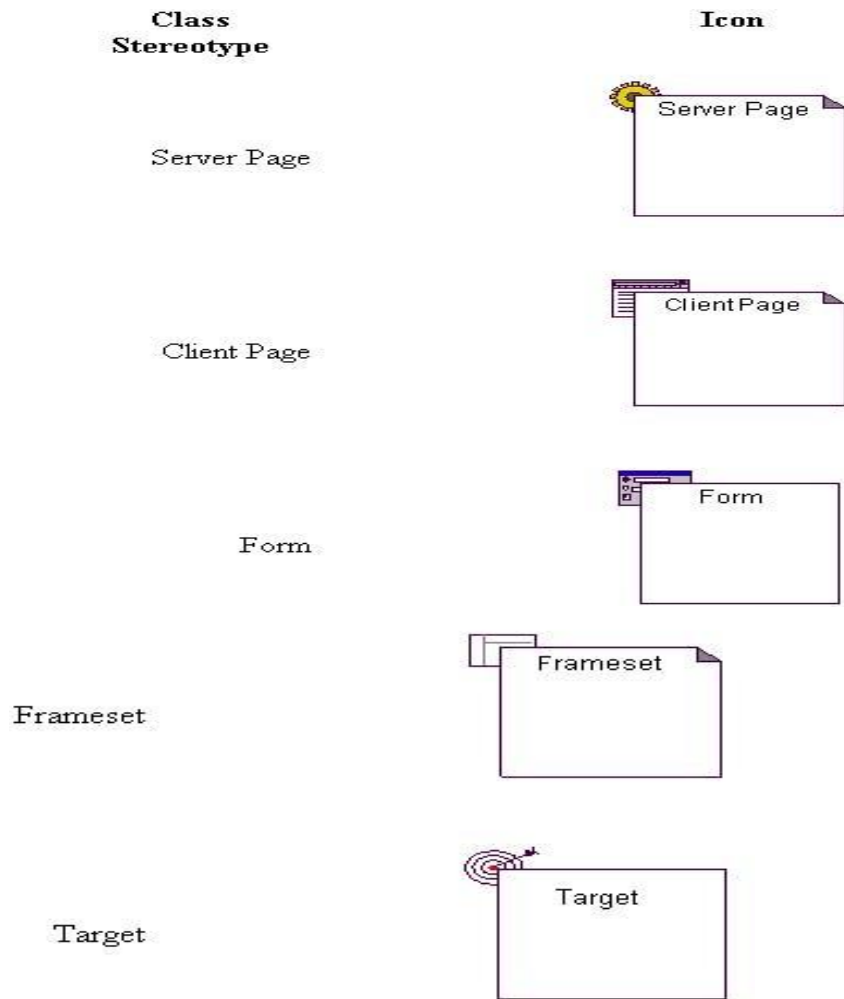
یکی دیگر از ویژگی های اینترفیس کاربر که هنوز مطرح نشده است، توانایی صفحه سرویس گیرنده برای نشان دادن یک کادر مکالمه^{۱۵} است. اسکریپت های سمت سرویس گیرنده این عملیات را انجام می دهد و این با یک ارتباط جهت دار از صفحه سرویس گیرنده به صفحه وب دیگر (سرور یا سرویس گیرنده) مدل می شود.

این ارتباط دارای قالب `<<Dialog>>` است و بر این موضوع دلالت می کند که صفحه سرویس گیرنده که دیالوگ^{۱۶} را فراخوانی کرده است به طور موقت به تعویق می افتد تا زمانی که

¹⁵ DialogBox
¹⁶ dialog

<<Dialog>> صدا زده شده بسته شود. به طور معمول صفحه سرویس گیرنده زمانی صفحه دیالوگ را فراخوانی می کند که مقداری از ورودی های کاربر را که روی صفحه اصلی پر نشده اند ، دریافت کند. آخرین قالب تحت مطالعه <<Event>> است.

۶-۲-۱-۵-آیکونها



۲-۵- متد یو.ام.ال^{۱۷} توسعه یافته

این متد شامل سه گام است که به صورت مکرر در پروسس طراحی انجام می شود. این گامها، طراحی مفهومی^{۱۸}، ناوبری^{۱۹} و نمایش^{۲۰} هستند که مدل های زیر را تولید می کنند:

- مدل مفهومی
 - مدل ساختاری ناوبری و مدل فضای ناوبری^{۲۱}
 - مدل نمایش
- مدل مفهومی با توجه به نیازهای کاری که از مورد استفاده^{۲۲} گرفته می شود، ساخته می شود. تکنیک های شی گرا متداول برای ساختن مدل مفهومی استفاده می شود. مثل پیدا کردن کلاسها، تعریف ساختارهای وراثتی و مشخص کردن محدودیت ها که با دیاگرام کلاسی نمایش داده می شوند. بر اساس مدل مفهومی مدل فضای ناوبری به دست می آید که یک مدل کلاس استاتیکی را نشان می هد. یک مجموعه راهنمایی برای مدل کردن فضای ناوبری پیشنهاد می شود. مدل ساختاری ناوبری ساختار نودها و پیوند های برنامه کاربردی وب^{۲۳} را تعریف می کند که نشان می دهد چگونه ناوبری به وسیله ساختارهای دسترسی پشتیبانی می شود. بر اساس مدل ساختاری ناوبری مدل نمایش را بدست می آوریم. این مدل می تواند به وسیله فریم های اچ.تی.ام. ال پیاده سازی شود.
- طراحی ناوبری یک گام مهم در طراحی برنامه کاربردی وب است. برنامه های کاربردی ساده با یک ساختار درختی کم عمق به زودی با اضافه کردن پیوند های جدید پیچیده خواهد شد. پیوند های اضافی قابلیت جابجایی را از یک طرف بهبود می بخشند و از طرف دیگر جهت یابی های بی ربط را

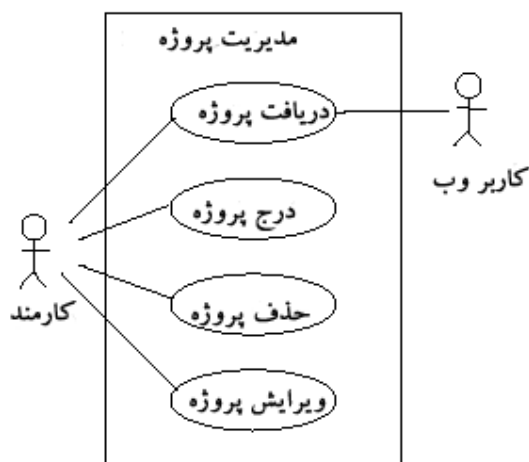
¹⁷ UML

¹⁸ conceptual
¹⁹ navigational
²⁰ presentational
²¹ Navigation space
²² usecase
²³ hypermedia

به وجود می آورند. توانایی این متدولوژی این است که دارای مراحل است که می تواند با یک روش اتوماتیک انجام شوند. بدین ترتیب اساس مکانیزم تولید برای طراحی وب را فراهم می کند.

۱-۲-۵- شروع با مورد استفاده ویک مدل مفهومی

این تکنیک بر کاربر متمرکز است. برای همین باید تعریف کنیم کاربران برنامه کاربردی چه کسانی هستند و یک روش مستقیم برای نمایش کاری که برنامه کاربردی بایستی انجام دهد پیشنهاد کنیم. به عنوان یک مثال از وب سایت یک کمپانی استفاده می کنیم. این وب سایت اطلاعاتی درباره خود کمپانی، کارمندان، رابطه شان با پروژه ها، مشتریان و سازمانها ارائه می دهد. در این مثال خود را به این مفاهیم محدود نکرده، گر چه اطلاعات دیگری مثل اطلاعات درباره محصولات، سند^{۲۴}ها، رویدادها و پیشنهادهای شغلی را می تواند شامل شود. شکل ۱ یک مورد استفاده برای مدیریت پروژه که قسمتی از مدل مورد استفاده، برنامه کاربردی وب^{۲۵} است را نشان می دهد:



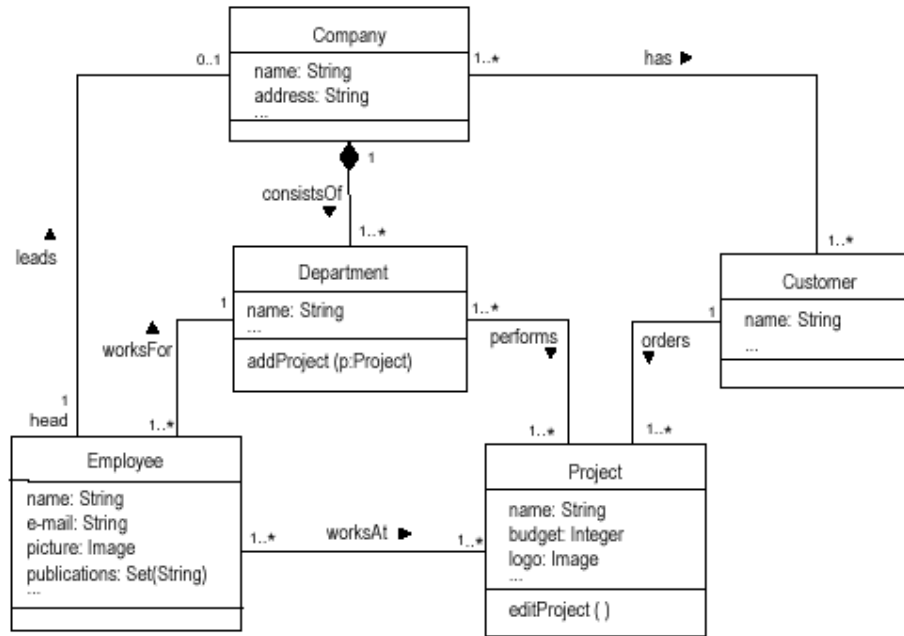
شکل ۱: مورد استفاده برای مدیریت پروژه

مدیریت کمپانیها، سازمانها و کارمندان می توانند به روش مشابه مدل شوند. طراحی مفهومی براساس این مورد استفاده هاست که شامل شیءهایی است که به فعالیت های عادی که کاربر با برنامه انجام می دهد مرتبط هستند. مدل مفهومی یک مدل است که سعی می کند مسیرهای ممکن ناوبری

جنبه های ارتباط متقابل^{۲۶} و نمایش را در نظر بگیرد. فعالیت های شناخته شده در طول مدل مفهومی عبارتند از:

۱. پیدا کردن کلاسها
۲. مشخص کردن بیشتر مشخصات و عملیات های مربوطه
۳. تعیین ارتباط بین کلاسها
۴. تعیین رابطه های وراثتی^{۲۷}
۵. پیدا کردن ارتباطات
۶. مشخص کردن اینترفیس ها
۷. تعریف محدودیت ها

نتایج این فعالیت ها یک مدل کلاسی برای مساله است. کلاسها از طریق ویژگیها و عملیاتها شرح داده می شوند. مدل مفهومی برای وب سایت کمپانی در شکل ۲ نشان داده شده است. کلاسها و ارتباطات، تعریف شده در این مرحله در طراحی ناوبری استفاده می شوند که برای بدست آوردن نودهای ساختار وب است. ارتباطات برای بدست آوردن پیوند ها استفاده می شود.



²⁶ Interaction
²⁷ hierarchies

شکل ۲: مدل مفهومی

۲-۲-۵- از یک مدل مفهومی به یک مدل فضای ناوبری

برای بدست آوردن مدل فضای ناوبری از مدل مفهومی راهنمائی‌هایی ا ذکر می شود. مدل فضای ناوبری شامل شیء‌هایی است که می تواند از طریق ناوبری در کاربرد وب دیده شود. چگونگی نمایش این شیء ها به وسیله مدل ساختاری ناوبری تعریف می شود که در قسمت بعدی بدست می آید. در ساختن مدل فضای ناوبری تصمیمات طراحی که مهم هستند گرفته می شوند. مثل ویو‌هایی روی مدل مفهومی که برای برنامه کاربردی نیاز می شوند یا مسیرهای ناوبری که برای برنامه نیاز است. تصمیمات طراح بر اساس مدل مفهومی، مدل مورد استفاده و نیازهای ناوبری است که برای برنامه بایستی فراهم شوند.

۱-۲-۲-۵- مدل سازی قطعات

برای ساختن مدل فضای ناوبری دو عنصر برای مدل کردن استفاده می شود:
کلاسهای ناوبری و ارتباطات ناوبری، قابلیت جابجایی مستقیم را بیان می کنند که همان نود و پیوند در اصطلاحات فنی وب هستند.
کلاس ناوبری: مدل های کلاس ناوبری درخواست هایی هستند که به وسیله کاربر در طول ناوبری می شوند. کلاسهای ناوبری دارای همان نام کلاسهای مفهومی هستند. برای نمایش آنها از <<Navigational class>> یو.ام.ال استفاده می شود که در شکل ۳ نشان داده شده است:

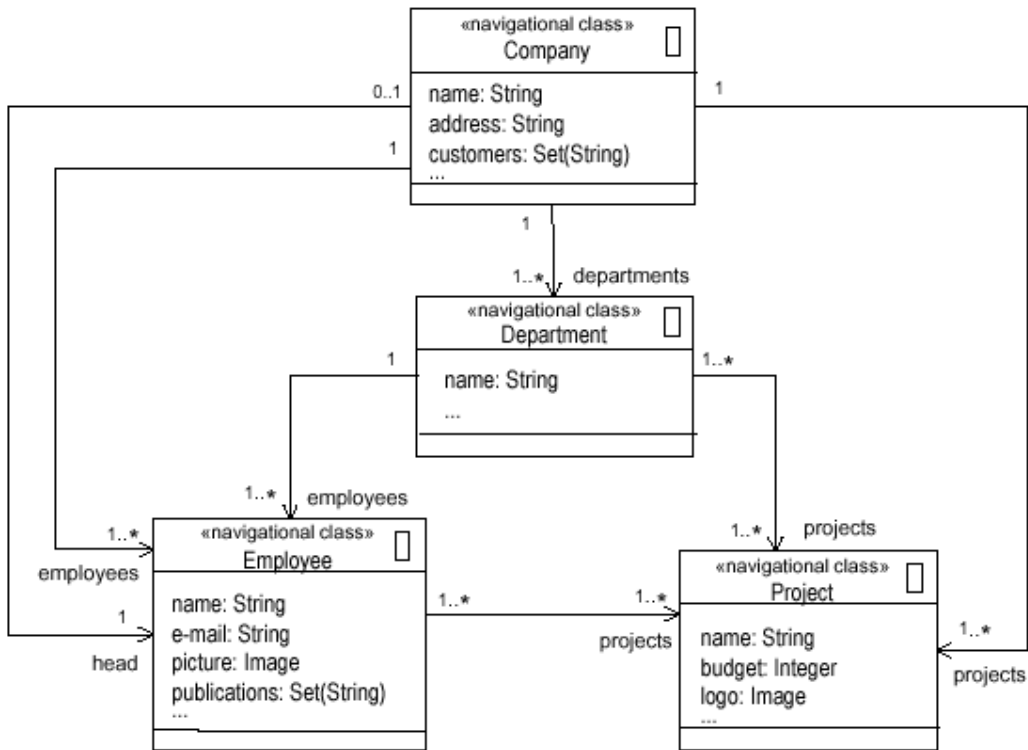
۲-۲-۲-۵- قابلیت ناوبری مستقیم^{۲۸}

ارتباطاتی است که در مدل فضای ناوبری به عنوان قابلیت ناوبری مستقیم از منبع کلاس ناوبری به مقصد کلاس ناوبری بیان می شوند. بنابراین شکل آنها با ارتباطاتی که در مدل مفهومی استفاده می شوند تفاوت دارد. جهت ارتباطات ناوبری در این مدل یک جهت (احتمالاً دو جهت هستند) می باشد که با یک فلش که به یک یا دو مقصد (هدف) وصل می شود نشان داده می شوند. همچنین هر مقصد

²⁸ Direct navigability

یک ارتباط به صورت نام نقش (وظیفه) در نظر گرفته می شود. اگر نام وظیفه واضح نباشد به صورت زیر عمل می شود. اگر ارتباط دارای مقصد است نام کلاس مقصد به عنوان نام وظیفه استفاده می شود. اگر بزرگتر از یکی است شکل جمع کلاس مقصد استفاده می شود. در دیاگرام های زیر همه ارتباطات بدون شرط فرض شده اند (به استثنای آنها که مرکبند) و به وسیله «Direct Stereotype <<Direct Stereotype >> navigability مشخص شده اند.

مدل فضای ناوبری با کلاسهای ناوبری و ارتباطات ناوبری ساخته می شوند که به صورت گرافیکی به وسیله دیاگرام گرافیکی یو.ام.ال نشان داده می شوند:



شکل ۳: مدل فضای ناوبری

اگر چه به طور واضح هیچ روشی به صورت اتوماتیک مدل فضای ناوبری را نمی سازد اما چندین راهنمایی وجود دارند که می تواند به وسیله طراح دنبال شوند :

۱_ کلاسهای مدل مفهومی که برای ناوبری مناسبند به عنوان کلاسهای ناوبری در مدل فضای ناوبری داخل می شوند. (کلاسهای ناوبری از کلاسهای مفهومی به دست می آیند.) اگر کلاس

مفهومی در مدل مورد استفاده، استفاده نشده باشد آن به پروسس ناوبری مربوط نمی شود و بنابراین در مدل فضای ناوبری حذف می شود. (شبهه کلاس Customer در مثال)

۲- اطلاعات لازم از کلاسهای حذف شده می تواند به عنوان ویژگی های سایر کلاسهای مدل فضای ناوبری در نظر گرفته شود. (نظیر متغیر Customers از کلاس ناوبری کمپانی) سایر ویژگی های کلاسهای ناوبری مستقیماً به ویژگی های مربوطه در کلاسهای مفهومی نگاشت می شوند و ویژگی هایی هم که به نظر مهم و لازم نمی رسند تا به کاربر نشان داده شوند حذف می گردند.

۳- اغلب ارتباطات اضافی که برای ناوبری مستقیم اضافه می شوند از مسیرهای ناوبری بزرگتر از یک اجتناب می کنند. مثالی که جدیداً معرفی شده اند ارتباطات ناوبری بین کمپانی و کارمند^{۲۹} و بین کمپانی و پروژه می باشد.

۳-۲-۵- از مدل فضای ناوبری به یک مدل ساختاری ناوبری

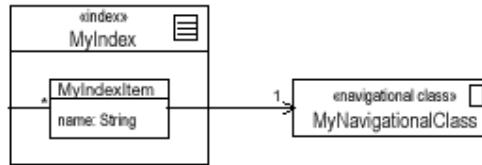
مدل فضای ناوبری می گوید که کدام شیء ها می توانند با ناوبری های مستقیم از طریق شیء ها ی دیگر ملاقات شوند. در این قسمت اینکه چگونه ناوبری می تواند با استفاده از عناصر دسترسی شبهه ایندکس ها ، تورهای راهنما ، پرس وجو^{۳۰} ها و منو انجام شود، شرح داده می شود. به طور فنی مسیرهای ناوبری (ارتباطات) با عناصر دسترسی نشان داده می شود. مدل ساختاری ناوبری می تواند به طور اصولی از مدل فضای ناوبری در دو مرحله به دست آید: ابتدا مدل فضای ناوبری به وسیله ایندکس ها ، تورهای راهنما و پرس وجوها بهبود می یابد. سپس می توان به طور مستقیم منوهای را که امکان انتخاب ها را برای ناوبری ارائه می دهد بدست آورد.

۱-۳-۲-۵- تعریف ایندکس ها ، تورهای راهنما و پرس وجو ها

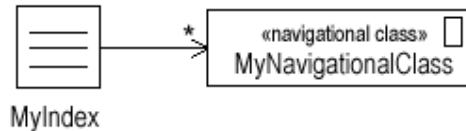
برای شرح ایندکس ها ، تورهای راهنما و پرس وجو ها از عناصر مدل زیر استفاده می شود:

ایندکس: یک ایندکس به وسیله یک شیء مرکب که شامل تعدادی دلخواه از عناصر ایندکس است مدل می شود. هر عنصر ایندکس ، یک شیء را که دارای یک نام و پیوند است به یک مورد از کلاس ناوبری تغییر حالت می دهد. هر ایندکس عضو همان به وسیله <<Index>> با آیگون مربوطه

مشخص می شود. کلاس ایندکس بایستی مطابق با ترکیب ساختار کلاسهای شکل ۴ ساخته شود ولی در عمل ما همیشه نماد خلاصه شکل ۵ را استفاده می کنیم.

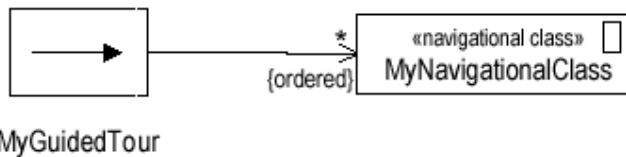


شکل ۴: کلاس ایندکس



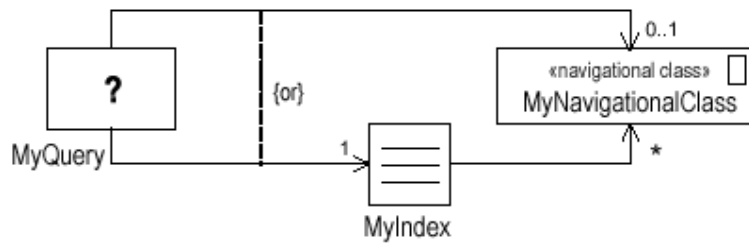
شکل ۵: نماد خلاصه برای کلاس ایندکس

تور راهنما: تور راهنما به وسیله یک شیء مدل می شود که دسترسی ترتیبی به نمونه های کلاس ناوبری را فراهم می کند. برای کلاسهایی که شامل شیء های تور راهنما هستند Stereotype «Guided Tour» را استفاده می کنیم و آیکن مربوطه آن در شکل ۶ نشان داده شده است. هر کلاس تور راهنما بایستی به یک کلاس ناوبری به وسیله یک ارتباط مستقیم که ویژگی {ordered} را دارد متصل شود.



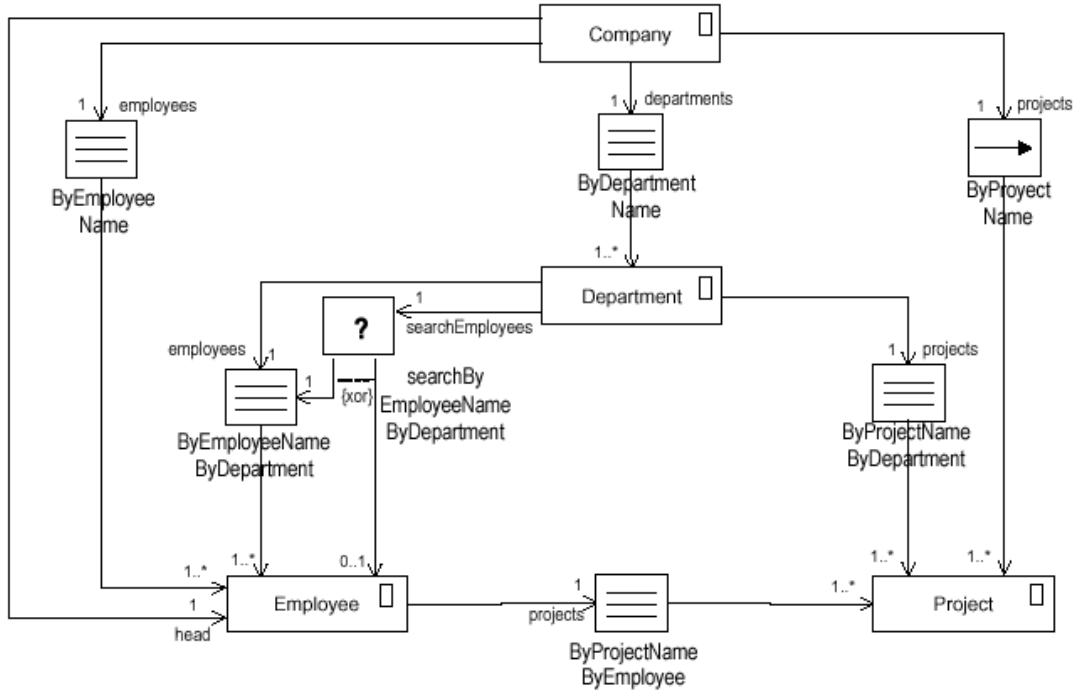
شکل ۶: کلاس تور راهنما

پرس وجو: پرس وجو به وسیله یک شیء که یک رشته را به عنوان ویژگی دارد مدل می شود. برای کلاسهای پرس وجو از Stereotype «Query» استفاده می کنیم و آیکن آن در شکل ۷ نشان داده شده است. همانطور که در شکل ۷ نشان داده شده هر کلاس پرس وجو منبع یک ارتباط دو جهته است که به وسیله محدودیت {or} مرتبط می شوند. در این روش می توان برای مدل یک پرس وجو با چندین شیء نتیجه ، ابتدا به یک ایندکس هدایت شد، سپس اجازه داد یک مورد خاص از یک کلاس ناوبری انتخاب شود:



شکل ۷: کلاس پرس و جو

شکل ۸ نشان می دهد چگونه مدل فضای ناوبری ، وب سایت کمپانی می تواند به وسیله ایندکس ها و تورهای راهنما و پرس و جو ها بهبود یابد. توجه کنید که دو روش ممکن دسترسی به کارمندان یک سازمان وجود دارد: به وسیله ایندکس و به وسیله پرس و جو .



شکل ۸: مدل فضای ناوبری با ایندکس ها، تور راهنما و پرس و جو

بهبود مدل فضای ناوبری به وسیله عناصر دسترسی از نوع ایندکس ، تور راهنما و پرس وجو با قوانین مشخص زیر دنبال می شوند که به صورت زیر خلاصه می شوند:

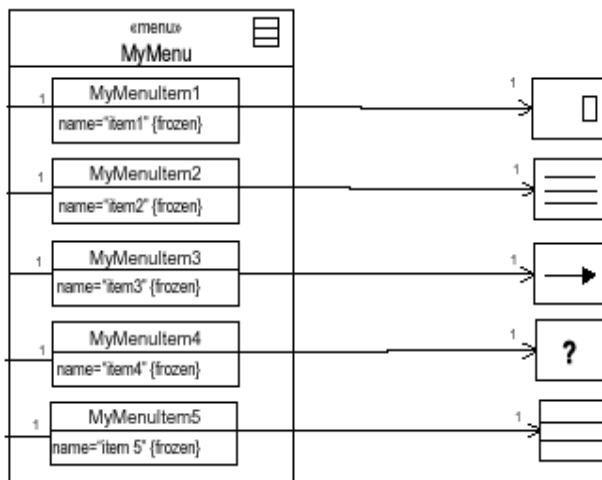
۱. تنها ارتباطاتی از مدل فضای ناوبری را در نظر بگیرید که درجه بزرگتر از ۱ در انتهای جهت ارتباط دارند.

۲. برای هر ارتباط از این نوع ، یک یا بیشتر عناصر دسترسی برای مشخص کردن ناوبری انتخاب کنید.

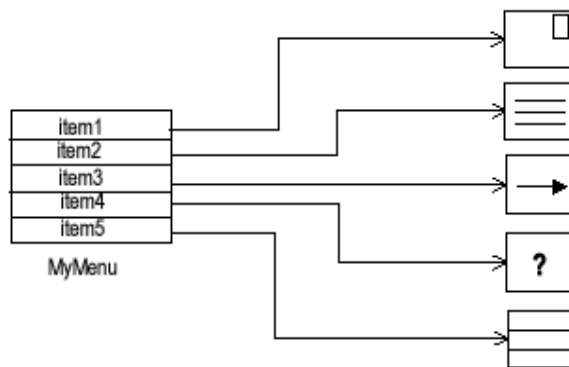
۳. بهبود مدل فضای ناوبری به صورت یک به یک است. به این صورت که نامهای نقش های ناوبری در مدل فضای ناوبری به عناصر دسترسی منتقل می شوند. (شکل ۳ و ۸ را مقایسه کنید). در گام ۲ این وظیفه طراح است که عناصر دسترسی مناسب را انتخاب کند.

۳-۲-۵-منوها

منو به وسیله شیء مرکب که شامل یک تعداد ثابت از عناصر منو است مدل می شود. هر عنصر منو دارای یک نام ثابت و یک پیوند به یک مورد از یک کلاس ناوبری یا به یک عنصر دسترسی است. هر منو یک مورد از تعدادی کلاسهای منو است که به وسیله <<Menu>> با آیکن مربوطه مشخص می شود. کلاس منو بایستی مطابق با ساختار مرکب کلاسهای نشان داده شده در شکل ۹ ساخته شود:



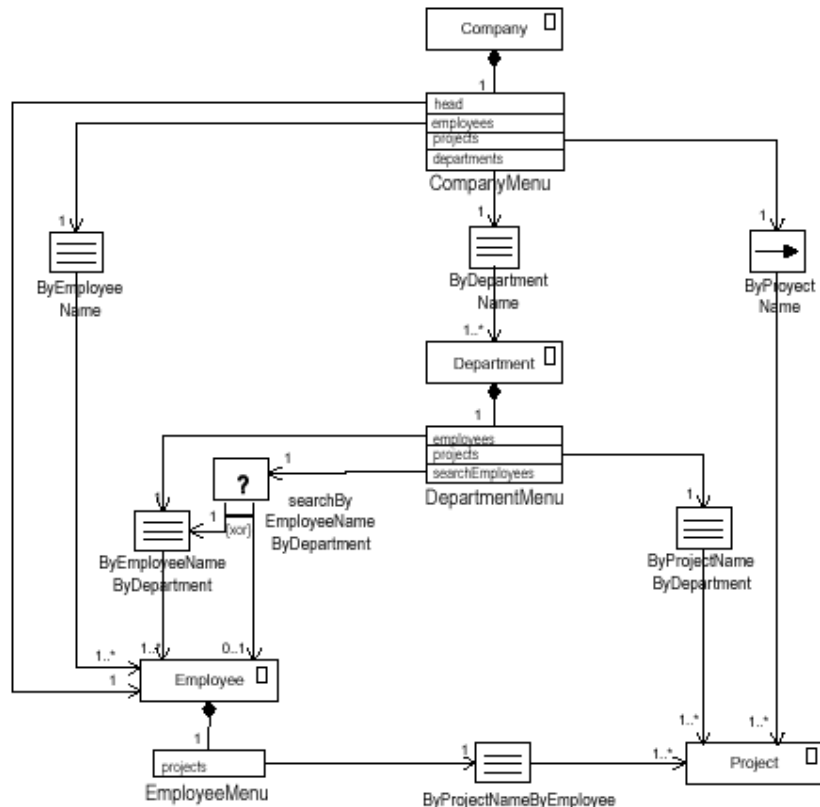
شکل ۹: کلاس منو



شکل ۱۰: نماد خلاصه برای کلاس منو

از آنجائیکه فرض شده عناصر منو نامهای ثابت دارند ویژگی^{۳۱}، {frozen} به هر نام ویژگی در یک کلاس منو متصل می شود. با این وجود همان کلاس عضو منو ممکن است مثالهای متفاوت داشته باشد یعنی با یک نام به شیءهای متفاوت پیوند شود. برای یک نماد مناسب از کلاسهای منو در مدلهای ساختار ناوبری ما نماد خلاصه نشان داده شده در شکل ۱۰ را استفاده می کنیم.

شکل ۱۱ نشان می دهد چگونه مدل ناوبری بخش قبلی به وسیله منوها کامل می شود. به طوریکه که هر کلاس منو با یک ارتباط مرکب به یک کلاس ناوبری مرتبط می شود. توجه کنید که نامهای وظایف در مدل قبلی حالا نامهای عناصر منو مربوطه هستند.

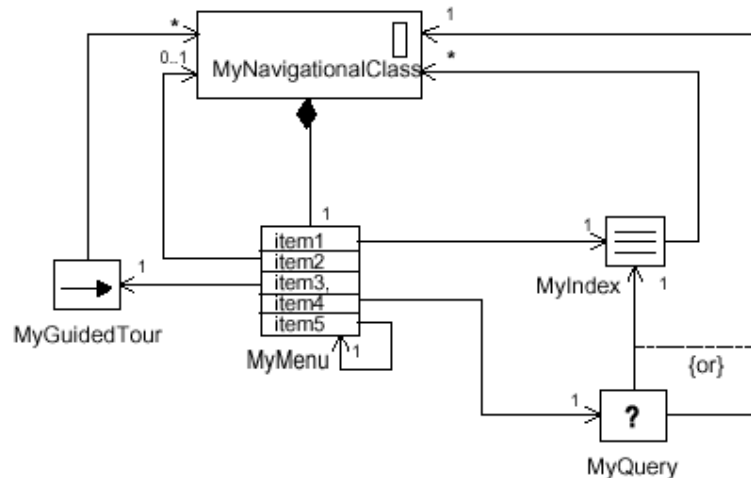


شکل ۱۱: مدل ساختاری ناوبری

تکمیل مدل فضای ناوبری به وسیله عناصر دسترسی از نوع منو با قوانین مشخصی است که به صورت زیر خلاصه می شوند:

۱. ارتباطاتی را در نظر بگیرید که به عنوان منبع ، یک کلاس ناوبری دارند.
۲. به هر کلاس ناوبری (در مدل قبلی) که حداقل یک ارتباط خروجی به یک کلاس منو مربوطه دارد، ارتباط برقرار کنید. ارتباط بین کلاس ناوبری و کلاس منو مربوطه اش مرکب است.
۳. هر وظیفه که در مدل قبلی اتفاق می افتد (در انتهای یک ارتباط جهت دار) یک عضو منو مربوطه برای آن معرفی کنید. به صورت پیش فرض نام وظیفه به عنوان نام ثابت عضو منو استفاده می شود.

۴. هر ارتباط از مدل قبلی که به عنوان مبدا یک کلاس ناوبری را دارد حالا یک ارتباط از عنصر منو مربوطه در مرحله ۳ دارد. توجه کنید که همه مراحل روش بالا می تواند با یک روش اتوماتیک انجام شود. به عنوان نتیجه یک مدل ساختاری ناوبری جامع به دست می آید.



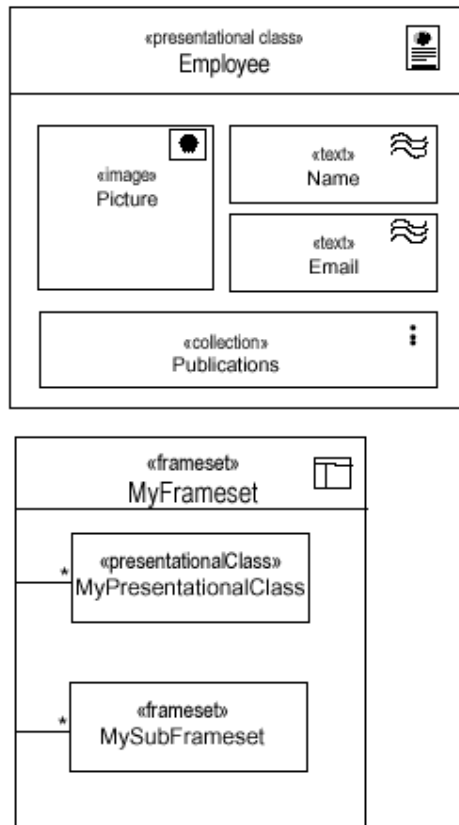
شکل ۱۲: الگوی طراحی برای ساختارهای دسترسی

۴-۲-۵- از یک مدل ساختاری ناوبری به یک مدل نمایش

مدل ساختاری ناوبری نشان می دهد چگونه از طریق فضای ناوبری با استفاده از عناصر دسترسی تعریف شده در بخش قبلی، هدایت شویم. در گام بعدی شرح داده می شود که چگونه اطلاعات داخل فضای ناوبری و ساختارهای دسترسی برای کاربر نمایش داده می شوند که به وسیله ساختن مدل نمایش انجام می شود، و طراحی اینترفیس انتزاعی مشابه با طرح اینترفیس کاربر را نشان می دهد. مدل نمایش روی سازمان ساختاری نمایش تکیه دارنده روی ظاهر فیزیکی به اصطلاح فرمت های خاص، رنگها و غیره. چنین تصمیماتی برای ساختن مدل پیش فرض رابط کاربر یا برای فاز پیاده سازی باقی می ماند که هدف ما نیست.

۱-۴-۲-۵- مدل کردن قطعات

هدف از مدل نمایش تصمیم گیری درباره این مطلب است که چه عناصر نمایشی برای نمایش کلاسهای ناوبری و یا ساختارهای دسترسی استفاده می شود. برای این منظور چندین عنصر مدل نمایش (با قالب های مربوطه) استفاده می شود. یک عنصر سطح بالا برای نمایش ، یک فریم ست^{۳۲} است که به وسیله یک شیء مرکب مدل می شود که شامل (در سطح پایین تر) شیء های نمایش دیگر است. اما همچنین ممکن است شامل یک تعداد از فریم ست ها باشد که در داخل آن قرار گرفته اند. فریم ست یک مثال از یک کلاس فریم ست هست که به وسیله <<frameset>> با آیگون مربوطه مشخص می شود. کلاسهایی که فریم ست هارا مدل می کنند بایستی مطابق با ساختار مرکب نشان داده شده در شکل ۱۳ ساخته شوند:



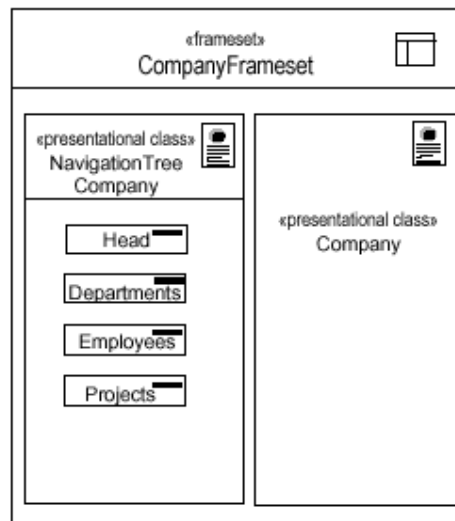
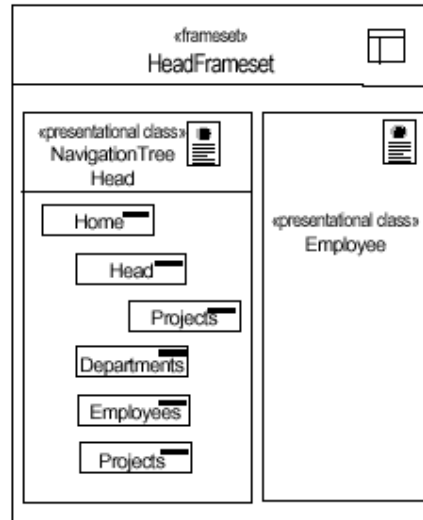
شکل ۱۳ :

شکل ۱۴: کلاس نمایش برای کارمند

Frameset

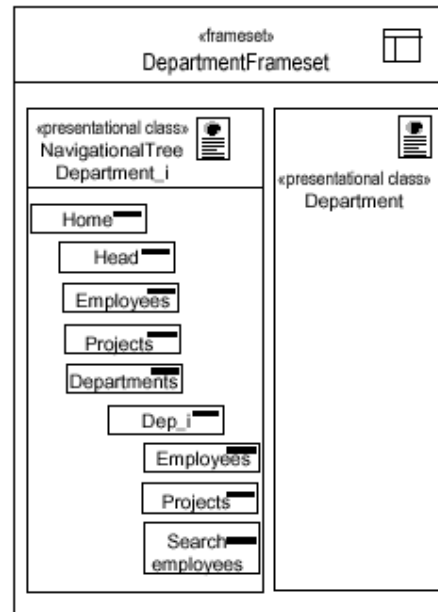
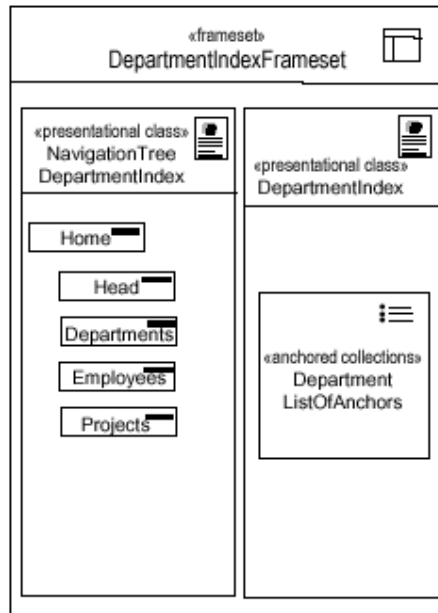
شیء‌های نمایشی که در فریم ست وجود دارند نمونه هایی از کلاس نمایشی هستند که به وسیله <<Presentational class>> با آیکون مربوطه مشخص می شوند. چنین شیء‌هایی شامل عناصر اصلی نمایش شبیه متن^{۳۳} ها، مجموعه ها (لیست هایی از متن ها)، تصاویر، anchor ها، مجموعه هایی از anchorها و غیره هستند. شکل ۱۴ نشان می دهد چگونه از اینها برای ساختن الگو برای نمایش کارمندان استفاده می شوند.

شکل ۱۵ تا ۱۸ قسمتی از یک مدل نمایش برای برنامه کاربردی را نشان می دهد. در این مثال فریم ست ها برای تقسیم نمایش به فریم ها استفاده می شوند ، که به موجب آن فریم چپ درخت واقعی ناوبری را نشان می دهد و فریم راست Context مربوطه را نشان می دهد. شکل ۱۵ نمایش کمپانی را نشان می دهد ، شکل ۱۶ رئیس کمپانی را نشان می دهد (بعد از انتخاب کردن Head) ، شکل ۱۷ ایندکسی از سازمان ها را به وسیله لیستی از Anchor ها نشان می دهد (بعد از انتخاب شدن سازمان^{۳۴}) و شکل ۱۸ یک سازمان انتخاب شده را نشان می دهد (برای مثال، Dep_i، سازمان i-ام) نمایش داده شده است.



شکل ۱۶: فریم ست برای Head

شکل ۱۵: فریم ست برای کمپانی



شکل ۱۸: فریم ست برای

شکل ۱۷: فریم ست برای Department

DepartmentIndex

البته احتمالات و انتخابهای زیادی برای ساختن مدل نمایش بر اساس مدل ساختاری ناوبری وجود دارند. در هر مورد، تعریف کردن شیوه نمایش برای هر کلاس ناوبری ضروری است و همچنین ساختار ناوبری بایستی پشتیبانی شود. روشی که استفاده می شود بر اساس استفاده از فریم ست هاست که اجازه می دهد ساختار ناوبری را تصور کنیم. به این وسیله ایده نمایش به دو قسمت اصلی تقسیم می شود: یک قسمت نمایش درخت ناوبری را فراهم می کند (که مسیر واقعی ناوبری کاربر را نشان می دهد) و قسمت دیگر محتوای مربوطه را نشان می دهد.

بر این اساس پروسیجر زیر برای بدست آوردن مدل نمایش از مدل ساختاری ناوبری تعریف می شود:

۱. ساختن یک نمایش برای هر کلاس ناوبری و برای هر کلاس ایندکس که در مدل ساختاری ناوبری اتفاق می افتد. نمایش کلاس ناوبری بایستی یک الگو را برای نمایش نمونه ها پی از کلاس فراهم کند که ویژگیهای داده شده را ارائه می دهند. برای مثال شکل ۱۴ یک نمایش را برای کلاس ناوبری، کارمند نشان می دهد. (نمایش کلاس ایندکس معمولاً به وسیله یک لیست از Anchor ها داده می شود). برای مثال نمایش کلاس ایندکس سازمان^{۳۵} در فریم راست شکل ۱۷ یک نمایش از کلاس ایندکس By Department name شکل ۱۱ است.

۲. کلاس ناوبری را به عنوان یک ریشه برای ناوبری انتخاب کنید. در مثال ذکر شده کلاس کمپانی انتخاب شده است.

۳. برای هر کلاس ناوبری و برای هر کلاس ایندکس همه مسیر های ممکن (در مدل ساختاری ناوبری) از کلاس ریشه به کلاس واقعی را در نظر بگیرید. برای هر مسیر یک نمایش از درخت ناوبری مربوطه بسازید.

(برای مثال با کلاس نمایشی درخت ناوبری کمپانی در فریم چپ شکل ۱۵ شروع می شود. از آنجائیکه کمپانی ریشه درخت ناوبری مربوطه است، تنها منو مربوط به کمپانی را نشان می دهد.) کلاس درخت ناوبری Head در شکل ۱۶ درخت ناوبری را برای کسی که به رئیس کمپانی حرکت می کند نشان می دهد. توجه کنید که ریشه این درخت به وسیله لینک، Home برای برگشت به

کمپانی نشان داده شده است. پیوند ، Projects در عمق ۲ به درخت برای نمایش منو مرتبط شده به Employees اضافه شده است.

کلاس نمایشی ، NavigationtreeDepartmentIndex در شکل ۱۷ ، درخت ناوبری را برای کسی که از کمپانی به ایندکس سازمان^{۳۳} حرکت کند نشان می دهد و کلاس نمایشی ، department_i در شکل ۱۸ ، درخت ناوبری را برای کسی که در آینده به یک سازمان خاص هدایت شود را نشان می دهد. (برای مثال سازمان ، i ام). توجه کنید که سپس Anchor های Employee ، Projects و Search Employees در عمق ۳ از درخت برای نمایش منو مرتبط شده بایک سازمان درج می شوند.

۴. نتایج گام ۱ و گام ۳ را بر طبق فریم ست ها ترکیب کنید. هر فریم ست دو قسمت دارد به طوری که فریم راست شامل نمایش کلاس ناوبری یا کلاس ایندکس (بدست آمده در مرحله ۱) و فریم چپ درخت ناوبری (بدست آمده در مرحله ۳) و یک امکان ناوبری برای این کلاس را نشان می دهد.

۵-۲-۵- نتیجه گیری

در این متد هر صفحه وب به عنوان یک شیء با Stereotype صفحه وب در نظر گرفته شده است. برای نمایش ارتباطات بین صفحات از شیء ای با Stereotype پیوند استفاده می شود. همچنین روش از چندین فاز که از طراحی مفهومی آغاز شده و در نهایت ارتباطات بین صفحات سایت را در قالب مدل ارتباطی کلاسها تولید می کند ، تشکیل می شود.

در پایان طرح کلی صفحات سایت در قالب اجزایی ساختاری مدل می شود .

توانایی این روش تا حدی به وسیله تولید اتوماتیک ساختار ناوبری و نمایش برنامه کاربردی می باشد.

چگونگی مدل سازی گام به گام به شرح زیر می باشد:

- مدل فضای ناوبری بر اساس مدل مفهومی
- مدل ساختاری ناوبری از مدل فضای ناوبری
- مدل نمایش از مدل ساختاری ناوبری

