



Logistic System Engineering Software

A Project Report Presented to :

Faculty of Engineering

Institute of Higher Education Hatef

In Partial Fulfilment of the Requirement for the degree of Bachelor of Science in

Field Information Technology

zokaeehamid@fastmail.fm

مهندسی نرم افزار سیستم باربری

پروژه دوره کارشناسی

فناوری اطلاعات

ارائه شده به:

دانشکده فنی مهندسی

مؤسسه آموزش عالی هاتف

استاد راهنما:

جناب آقای شیخ زاده

توسط:

آقای حمید ذکائی

۳۰ خرداد، ۱۳۹۰

The Logistic System Engineering Software

A Project Report Presented to :

Faculty of Engineering

Institute of Higher Education Hatef

In Partial Fulfilment of the Requirement for the degree of Bachelor of Science in

Field Information Thechnology

Advisor:

Mr.sheykhzadeh

By:

Mr.zokaei

June 20, 2011

Abstract

Thesis, which is located in front of your presence, the logistic system engineering software is. the first group of disciplines, modeling agencies, requirements management, analysis and design and implementation, and the second group of disciplines; discipline Deployment, discipline to the content process of the R.U.P system software to run on the logistic system. The output of each discipline by the software with the Rational Rose modeling language U.M.L, have been modeled (file attach project is the thesis). In the last essay of the disciplines in the project has been concluded.

صفحه	فهرست مطالب
۶.....	۱.۱ چکیده.....
۷.....	۲ . فصل اول پیشگفتار.....
۸.....	۲.۱ پیشگفتار.....
۹.....	۳ . فصل دوم بررسی منابع علمی.....
۱۰.....	۳.۱ مقدمه.....
۱۱.....	۳.۲ تاریخچه مختصر نرم افزار.....
۱۱.....	۳.۳ مفاهیم مهندسی نرم افزار.....
۱۱.....	۳.۳.۱ تعریف مهندسی نرم افزار.....
۱۲.....	۳.۳.۲ تعریف پروژه موفق.....
۱۲.....	۳.۳.۳ تعریف فرآورده نرم افزاری.....
۱۲.....	۳.۳.۴ مفهوم کیفیت.....
۱۳.....	۳.۳.۵ تعریف فرایند تولید نرم افزار.....
۱۴.....	۳.۳.۶ آمارهای پروژه های نرم افزاری.....
۱۶.....	۳.۴ آر.یو.پی چیست ؟.....
۱۶.....	۳.۴.۱ فرایند یکپارچه رشنال (یا به اختصار، آر.یو.پی) چیست؟.....
۱۷.....	۳.۴.۲ آر.یو.پی به عنوان یک رویکرد مهندسی نرم افزار.....
۱۹.....	۳.۴.۳ دو بُعد دینامیک و ایستاتیک آر.یو.پی.....
۲۱.....	۳.۴.۴ تعریف دیسپیلین.....
۲۲.....	۳.۴.۵ تعریف جریان کاری.....
۲۴.....	۳.۴.۶ دسته های دیسپیلین در آر.یو.پی.....

۴	فصل سوم روش انجام پروژه	۲۶
۴.۱	مقدمه	۲۷
۴.۲	دیسپلین مدل سازی از سازمان	۲۸
۴.۲.۱	درک ساختار و پویایی سازمان باربری	۲۹
۴.۲.۲	درک مسائل و مشکلات جاری سازمان باربری	۳۰
۴.۲.۳	استنتاج و استخراج نیاز مندیهای سازمان باربری	۳۱
۴.۲.۴	چشم انداز سازمان باربری	۳۲
۴.۲.۵	شناسایی اکتورها و اهداف آنها	۳۳
۴.۲.۶	استخراج مدل تحلیل سازمان	۳۴
4.3	دیسپلین مدیریت نیازمندی ها	۳۵
۴.۳.۱	شناسایی اکتورها و موارد کاربرد	۳۶
۴.۳.۲	جزئیات موارد کاربرد (یا همان سناریوها)	۳۷
۴.۳.۳	استخراج مدل موارد کاربرد	۴۲
۴.۳.۴	ایجاد نمونه ای از واسط کاربر	۴۳
۴.۴	دیسپلین تحلیل و طراحی	۴۷
۴.۴.۱	مدل تحلیل	۴۸
۴.۴.۲	دیاگرام های توالی در تحلیل سیستم نرم افزاری باربری	۴۹
۴.۴.۳	مدل طراحی	۵۰
۴.۴.۴	بسته های مدل طراحی سیستم نرم افزاری باربری	۵۱
۴.۴.۵	کلاس های مدل طراحی سیستم نرم افزاری باربری	۵۲
۴.۵	دیسپلین پیاده سازی	۵۹
4.5.1	پیاده سازی کلاسها و اشیاء در قالب یکسری مؤلفه (کامپونتها)	۶۰
۴.۵.۲	کامپونتهای سیستم نرم افزاری باربری	۶۱

تعریف چگونگی سازماندهی و ساختار کدهای برنامه در قالب زیر سیستم ها.....	۴.۵.۳	۶۴
..... دیسیپلین محیط	۴.۶	۷۲
..... مهندس فرایند	4.6.1	۷۳
..... مدل محیط سیستم نرم افزاری باربری	۴.۶.۲	۷۳
..... فصل چهارم نتیجه گیری	۵	۷۵
..... نتیجه گیری	5.1	۷۶
..... سخن آخر	5.2	۷۷
..... منبع کل	۵.۳	۷۷

فهرست شکلها صفحه

..... شکل ۱-۳ مثلث موفقیت پروژه	۱۳
..... شکل ۲-۳ فرآیند تولید محصول نرم افزاری	۱۳
..... شکل ۳-۳ آمار مربوط به وضعیت پروژههای نرم افزاری در سال ۱۹۷۹ میلادی	۱۴
..... شکل ۴-۳ آمارهای مربوط به وضعیت پروژههای نرم افزار در سال ۱۹۹۵ میلادی	۱۴
..... شکل ۵-۳ آمارهای مربوط به وضعیت پروژههای نرم افزاری در سال ۲۰۰۰ میلادی	۱۵
..... شکل ۶-۳ مقایسه میان درصد پروژه های موفق، شکست خورده، و دارای چالش در طول سال های مختلف	۱۵
..... شکل ۷-۳ مقایسه مفاهیم Unified و Intergrated	۱۶
..... شکل ۸-۳ ارتباط میان فرایند تولید و پروژه	۱۷
..... شکل ۹-۳ مدل مهندسی فرایند نرم افزار؛ ارائه شده توسط OMG	۱۸
..... شکل ۱۰-۳ سازماندهی فرایند آریوپپی در دو بعد زمانی و محتوایی (دینامیک و ایستاتیک)	۱۹

- شکل ۱۱-۳ نمای کلی این چهار فاز نشان داده شده است ۲۰
- شکل ۱۲-۳ نمونه ای از ارتباط میان عناصر کلیدی در ساختار محتوایی آر.یو.پی ۲۲
- شکل ۱۳-۳ ارتباط میان دیسپلین های فرایند و جریان کار متناظر با آن ۲۳
- شکل ۱۴-۳ نمونه ایی از یک جریان کاری ۲۳
- شکل ۱۵-۳ یک تکرار در چرخه تولید شامل تمام دیسپلین ها می باشد ۲۵
- شکل ۱-۴ مدل تحلیل سازمان باربری ۳۴
- شکل ۲-۴ نمایی مقطعی از دیسپلین نیازمندی ها در چرخه تولید ۳۵
- شکل ۳-۴ مدل موارد کاربرد سیستم نرم افزاری باربری ۴۲
- شکل ۴-۴ پنجره نخست ۴۳
- شکل ۵-۴ ورود ۴۳
- شکل ۶-۴ سفارش حمل بار ۴۴
- شکل ۷-۴ گرفتن گزارش از موقعیت بار ۴۴
- شکل ۸-۴ گرفتن اطلاعات مسیر و بارنامه ۴۵
- شکل ۹-۴ گرفتن گزارش از بارنامه ها ۴۵
- شکل ۱۰-۴ مدیریت بارنامه ها ۴۶
- شکل ۱۱-۴ مدیریت ایستگاه ها ۴۶
- شکل ۱۲-۴ نمایی مقطعی از دیسپلین تحلیل و طراحی در طول فازهای چرخه تولید ۴۷
- شکل ۱۳-۴ مدل تحلیل سازمان باربری ۴۸
- شکل ۱۴-۴ محل دیاگرام های توالی در فایل رشنال رز پروژه ۴۹
- شکل ۱۵-۴ نمونه یک دیاگرام توالی (دیاگرام توالی شالوده تحلیل) ۵۰
- شکل ۱۶-۴ نمونه ای از ساختار بسته ها در مدل طراحی سیستم نرم افزاری باربری ۵۲
- شکل ۱۷-۴ کلاس مشتری و خصوصیات آن در فایل رشنال رز پروژه ۵۳
- شکل ۱۸-۴ کلاس مشتری و خصوصیات آن در فایل رشنال رز پروژه ۵۴
- شکل ۱۹-۴ محل کلاس بارنامه در فایل رشنال رز پروژه ۵۵
- شکل ۲۰-۴ محل کلاس بارنامه در فایل رشنال رز پروژه ۵۶

شکل ۴-۲۱	محل کلاس بارنامه در فایل رشنال رز پروژه	۵۷
شکل ۴-۲۲	محل کلاس بارنامه در فایل رشنال رز پروژه	۵۷
شکل ۴-۲۳	محل کلاس کاربر در فایل رشنال رز پروژه	۵۸
شکل ۴-۲۴	نمایی مقطعی از دیسپلین پیاده سازی در طول چرخه تولید	۵۹
شکل ۴-۲۵	مدل شالوده کامپونت ها در فایل رشنال رز پروژه	۶۳
شکل ۴-۲۶	دیسپلین محیط و حجم فعالیت های مرتبط با آن در طی فازهای مختلف فرایند تولید	۷۲
شکل ۴-۲۷	مدل محیط سیستم نرم افزاری باربری	۷۳

صفحه

فهرست جدولها

جدول ۴-۱	سناریوی مورد کاربرد ورود	۳۷
جدول ۴-۲	سناریوی مورد کاربرد خروج	۳۷
جدول ۴-۳	سناریوی مورد کاربرد گرفتن گزارش از موقعیت بار	۳۸
جدول ۴-۴	سناریوی مورد کاربرد سفارش دادن برای حمل بار	۳۸
جدول ۴-۵	سناریوی مورد کاربرد اعلام موقعیت بار در هر ایستگاه	۳۹
جدول ۴-۶	سناریوی مورد کاربرد گرفتن اطلاعات مسیر و بارنامه	۳۹
جدول ۴-۷	سناریوی مورد کاربرد گرفتن گزارش از بارنامه	۴۰
جدول ۴-۸	سناریوی مورد کاربرد مدیریت بارنامه ها	۴۰
جدول ۴-۹	سناریوی مورد کاربرد مدیریت ایستگاه ها	۴۱
جدول ۴-۱۰	سناریوی مورد کاربرد مدیریت راننده ها	۴۱

۱.۱ چکیده

موضوع پایان نامه ای که در مقابل حضور شما قرار دارد، مهندسی نرم افزار سیستم باربری است. در فصلهای نخست این پایان نامه به تشریح مفاهیم پایه از مهندسی نرم افزار می پردازیم و در فصل میانی به تشریح روش انجام پروژه در قالب فرایند تولید آر.یو.پی از ۵ دیسپلین استفاده شده است. از دیسپلین های دسته اول؛ مدل سازی سازمان، مدیریت نیازمندیها، تحلیل و طراحی و پیاده سازی، و از دیسپلین های دسته دوم؛ دیسپلین محیط را از بُعد محتوایی فرایند آر.یو.پی برای سیستم نرم افزاری باربری به اجرا در آمده است. خروجی های هر دیسپلین توسط نرم افزار رشنال رز با زبان مدل سازی یو.ام.ال، مدل سازی شده اند (فایل پروژه ضمیمه پایان نامه شده است). در انتهای پایان نامه از اهداف هر دیسپلین که در پروژه به اجرا گذاشته نتیجه گیری شده است. تمامی تعاریف تئوریک این پایان نامه از کتاب، معرفی آر.یو.پی (مشخصات کامل این کتاب در قسمت منبع کلی ذکر شده است) بر گرفته شده است و برای جلوگیری از تحریف مفاهیم پایه، بعضی از مطالب عیناً مثل خود کتاب فوق الذکر بیان شده اند.

۲. فصل اول پیشگفتار

۲.۱ پیشگفتار

به نام یگانه خالق هستی بخش، به نام او که جان را فکرت آموخت. نوشتن، همواره یکی از پیچیده ترین فعالیتهای فکری بشر بوده است. یک نوشته خوب می تواند اثری ماندگار و ابزار ارتباطی مناسبی برای انتقال اندیشه ها و تجارب باشد. در عین حال، یک نوشته بد هم می تواند تاثیرات مخربی بر ذهن ها و دیدگاه ها داشته باشد. در این پایان نامه که در مقابل حضور شما است، یک نمونه پروژه سیستم نرم افزاری در زمینه مهندسی نرم افزار می باشد. از آنجا که ساخت یک نرم افزار کاری بس بسیار مشکل و پرهزینه است، و بسیار ممکن است با شکست روبرو شوند. بشر به فکر روش و چهارچوبی اصولی برای ساخت نرم افزارها افتاد و علمی به نام مهندسی نرم افزار به وجود آمد، تا در به موفقیت رسیدن پروژه ساخت یک نرم افزار کمک کند. البته در این مقاله موفقیت یک نرم افزار را تعریف و چالش هایی که باعث شکست پروژه های نرم افزاری می شود را بررسی می کنیم و در نهایت به راهکارهایی برای این چالش ها ارائه شده است را عنوان می کنیم. سیستم نرم افزاری که در این مقاله در نظر گرفته شده است، سیستم نرم افزاری باربری (لجستیک)^۱ می باشد و فرایند استفاده شده، فرایند توسعه نرم افزار آر.یو.پی^۲ است که در ادامه به توصیف آن می پردازیم. مطالبی که در مورد مهندسی نرم افزار در این پایان نامه آمده است بصورت کلی بیان شده است و فقط برای درک بهتر از این پروژه است، البته منبع کلی مطالب ذکر شده است و در صورت نیاز می توانید به آن رجوع کنید.

^۱ Logistic

^۲ RUP(Rational Unified Process)

۳. فصل دوم

بررسی منابع علمی

۳.۱ مقدمه

بی گمان، نرم افزار یکی از پیچیده ترین و در عین حال قابل انعطاف ترین دستاوردهای بشر می باشد. با وجودی که بیش از چند دهه از پیدایش نرم افزار نمی گذرد، این پدیده شگفت آور قرن بیستم، به عنوان یکی از مؤلفه های کلیدی فناوری های نوین اطلاعات و ارتباطات، تاثیر شگرفی بر کلیه جوانب زندگی بشر داشته است. امروزه نرم افزار، سوخت لازم برای راه اندازی و به حرکت در آوردن موتورهای اقتصاد نوین تلقی می شود. هیچ سازمان و کسب و کار نوینی، نمی تواند بدون نرم افزار به حرکت و تکامل خود ادامه دهد.

به طور کلی فناوری اطلاعات و ارتباطات و به طور خاص نرم افزار، گونه های جدیدی از ارتباطات و تعاملات را شکل داده است. به کمک این پدیده و دستاورد منحصر به فرد، قادر هستیم اطلاعات را به شکل هایی که تاکنون حتی تصور هم نمی کردیم، ایجاد نمود و الگوهای جدیدی را در میان آنها کشف نماییم. روش های درمان بیماری ها، روش های یادگیری، روش های کسب و کار، و به طور خلاصه، کلیه جوانب زندگی به شدت تحت تاثیر قرار گرفته است. به کمک این فناوری ها، بشر توانسته پا را از مرزها و قلمروهای پیشین فراتر نهاده و قدم در دنیای پر رمز و راز هستی نهد. دسترسی به فضای بیکران آسمان ها از یک سو و وارد شدن به دنیای اتم ها (در مقایسه نانو) از سوی دیگر، نمونه های آشنایی از تاثیرات و جلوه های بکارگیری فناوری های نوین اطلاعات و ارتباطات و به خصوص نرم افزار می باشد.

اما به واسطه تغییرات بسیار سریع و غافل گیر کننده فناوری های نوین اطلاعات و ارتباطی و به طور خاص نرم افزار، و به موازات آن، تغییر نیازها، خواسته ها، و انتظارات استفاده کنندگان از نرم افزار و قابلیت های آن، طراحی و تولید نرم افزار، بسیار پیچیده می باشد. عوامل دیگری مانند رقابت شدید، کمبود نیروی متخصص و حرفه ای، عدم دسترسی به دانش و تجربه موفق دیگران، لزوم تولید سریع، لزوم تولید مقرون به صرفه، نیاز روز افزون به همکاری میان رشته های مختلف، و مهم تر از همه، عدم استفاده مناسب از اصول و مبانی مهندسی در طراحی تولید نرم افزار، این صنعت را با چالش های بسیاری روبرو نموده است.

۳.۲ تاریخچه مختصر نرم افزار

حدود ۵۰ سال پیش، یعنی در اوایل پیدایش نرم افزار، استفاده کنندگان این فراورده نوین، همان طراحان و تولیدکنندگان آن بودن. در آن زمان نرم افزار عمدتاً برای محاسبات و حل مسائل ریاضی استفاده می شد. وجود زبان های سطح پایین و محدودیتهای سخت افزاری (کمبود حافظه و سرعت پردازش کم) از دیگر مشخصه های دوران اولیه پیدایش نرم افزار است. در آن روزهای اولیه، نرم افزار چیزی جدا از سخت افزار نبود و حتی برای فروش سخت افزار، بطور رایگان در آن تعبیه می شد! اما با گسترش دامنه کاربرد رایانه و به تبع آن نرم افزار در زمینه های مختلف، به مرور شرایطی به وجود آمد که استفاده کنندگان و کاربران نرم افزار از طراحان و تولید کنندگان آن جدا شدند. سازمان هایی و شرکت هایی به وجود آمدند که کارشان صرفاً تولید نرم افزار بود. حال دیگر نرم افزار قیمت داشت و اتفاقاً بر خلاف روند کاهش قیمت در سخت افزارها، روز به روز بر قیمت آن افزوده می شد.

سرانجام برای اولین بار، در سال ۱۹۶۸ و در یک کنفرانس که توسط ناتو در کشور آلمان برگزار شده بود، لزوم مهندسی این دستاورد جدید بشر، یعنی نرم افزار، تأکید شد. از آن زمان به بعد، با گسترش تکنیکهای مهندسی، ابزارها، دانش و تجربه، صنعت نرم افزار به یکی از صنایع برتر جهان تبدیل شده است.

۳.۳ مفاهیم مهندسی نرم افزار

۳.۳.۱ تعریف مهندسی نرم افزار

مهندسی نرم افزار، شاخه ای است از مهندسی، که با بهره گیری از دانش علمی، به ارائه راه حل هایی مقرون به صرفه، در قالب دستاوردهای نرم افزاری و به منظور حل مسائل و مشکلات علمی و خدمت به جامعه بشری، اقدام می نماید.

فعالیت های مهندسی، عمدتاً دو گونه اند: ۱- طراحی روتین که شامل ارائه راه حل برای مسائلی آشنا و استفاده مجدد از راه حل های قبلی می باشد و ۲- طراحی نوآورانه که عبارت است از ارائه راه حل هایی نو و بدیع برای مسائلی نا آشنا. مهندسی نرم افزار، اغلب با فعالیت های دسته دوم، یعنی طراحی نوآورانه سر و کار دارد.

۳.۳.۲ تعریف پروژه موفق

یک پروژه موفق نرم افزاری، پروژه ای است که در یک محدوده زمانی از قبل برنامه ریزی شده و با بودجه ای از قبل پیش بینی شده، یک فراورده نرم افزاری دارای کیفیت مطلوب (یعنی کیفیتی مطابق با خواسته ها و نیازهای واقعی کاربران) در آن تولید می گردد.

البته برای تکمیل تعریف ارائه شده از یک پروژه موفق، لازم است برای دو مفهوم اساسی در این تعریف، یعنی فراورده نرم افزاری و نیز مفهوم کیفیت، تعاریفی ارائه نماییم.

۳.۳.۳ تعریف فراورده نرم افزاری

به زبان ساده، یک فراورده نرم افزاری عبارت است از یک برنامه نرم افزاری قابل اجرا به علاوه مجموعه مستندات و دست نامه های کاربران آن (البته گاهی مجموعه کدهای برنامه نیز در قالب فراورده نهایی ارائه میگردد). توجه داشته باشید که یک فراورده نرم افزاری باید به اصطلاح، قابلیت خودپشتیبانی داشته باشد؛ بر اساس این قابلیت، همه کاربران یک سیستم نرم افزاری، اعم از کاربران ساده، مدیران سیستم، و یا مسئولین نگهداری و به روز رسانی آن، باید قادر باشند بدون نیاز به حضور تولیدکنندگان و پدیدآوردگان آن، تمام انتظاراتشان را از سیستم بر آورده نمایند.

۳.۳.۴ مفهوم کیفیت

کیفیت، مفهوم بسیار پیچیده ای است. مسلم است که همه کیفیت، عبارتند از: تطابق با نیازمندی ها و متناسب بودن برای استفاده. این دو تعریف که تا حد زیادی به هم مرتبط می باشند، تفاوت کوچکی نیز دارند؛ متناسب بودن برای استفاده، تأکید بیشتری بر نقش نیازمندی ها و انتظارات مشتری و کاربر دارد. حال می توانیم مفهوم عدم موفقیت یک پروژه نرم افزاری را بررسی نماییم. در شکل ۳-۱، مثلث موفقیت نشان داده شده است. بر اساس این شکل، تنها در صورتی یک پروژه، موفق تلقی می گردد که در محدوده زمان و هزینه پیش بینی شده، خاتمه یافته و نتیجه حاصل از آن دربرگیرنده تمام نیازمندیهای مورد توافق با مشتری باشد؛ هر شرایطی غیر از این، عدم موفقیت یا شکست پروژه محسوب می شود.

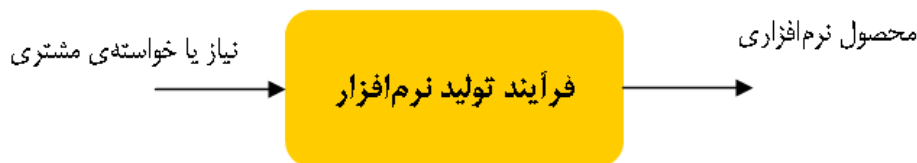


شکل ۱-۳ مثلث موفقیت پروژه

۳.۳.۵ تعریف فرایند تولید نرم افزار

هر پروژه نرم افزاری، بزرگ یا کوچک، موفق یا ناموفق، مرحله‌ای را طی می‌نماید که در طی آن یک خواسته یا نیاز، به فرآورده‌ای نرم افزاری تبدیل می‌شود. الگو و قالبی که چگونگی طی مراحل مختلف یک پروژه را تعریف می‌نماید، اصطلاحاً فرایند تولید نرم افزار نامیده می‌شود.

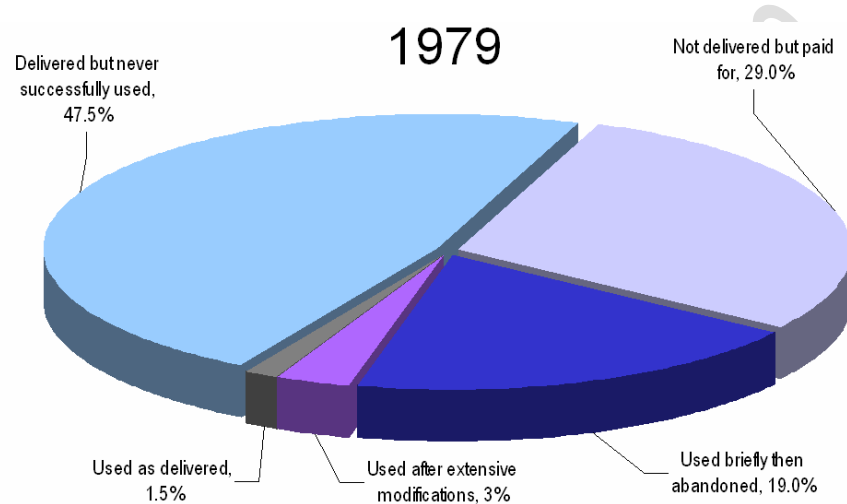
شکل ۲-۳، فرایند تولید نرم افزار و ورودی و خروجی‌های آن را نشان می‌دهد. یک فرایند تولید، به ما می‌گوید که برای دستیابی به هدف مطلوب در یک پروژه که همان تولید فرآورده‌ای نرم افزار با کیفیت مطلوب می‌باشد، چه کسی، چه کاری را، چه موقع، و چگونه باید انجام دهد. در واقع بدون داشتن تعریفی مشترکی از فرایند، هماهنگی و انجام کار تیمی در یک پروژه نرم افزاری، امکان پذیر نخواهد بود.



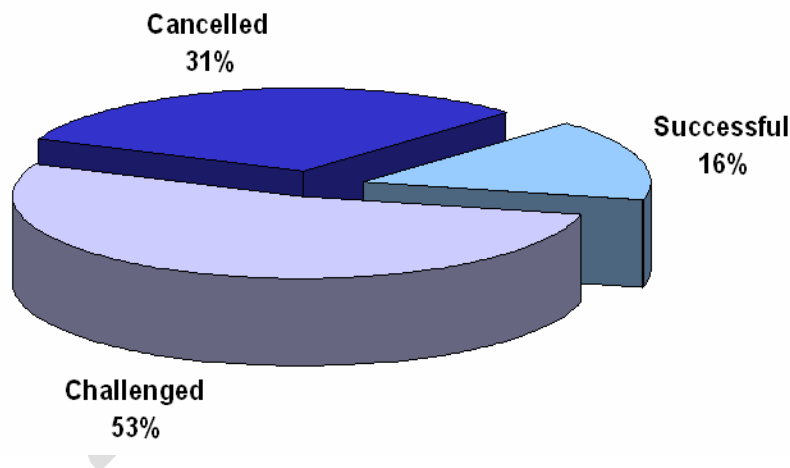
شکل ۲-۳ فرایند تولید محصول نرم افزاری

۳.۳.۶ آمارهای پروژه های نرم افزاری

بر اساس آمارهای معتبری که توسط مؤسسات مانند آی،دی،سی^۳ داس تن ایش گروپ^۴ و در پی بررسی هزاران پروژه تولید نرم افزار در ابعاد و در زمینه های مختلف، تهیه شده است، درصد زیادی از پروژه های نرم افزاری در دنیا با شکست و عدم موفقیت مواجه می شوند.



شکل ۳-۳ آمار مربوط به وضعیت پروژه های نرم افزاری در سال ۱۹۷۹ میلادی



شکل ۳-۴ آمارهای مربوط به وضعیت پروژه های نرم افزار در سال ۱۹۹۵ میلادی

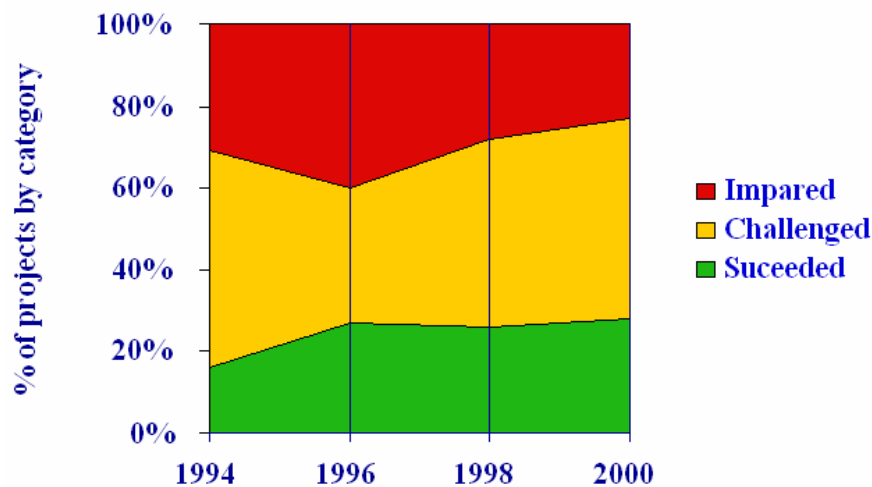
^۳IDC : International Data Corporation

^۴Standish Group



شکل ۵-۳ آمارهای مربوط به وضعیت پروژههای نرم افزاری در سال ۲۰۰۰ میلادی

همانگونه که در نمودار زیر نشان داده شده است، در طول زمان و با بهبود روش های مهندسی و نیز تکامل ابزارها، آمارهای مرتبط با شکست و عدم موفقیت کمتر شده است.



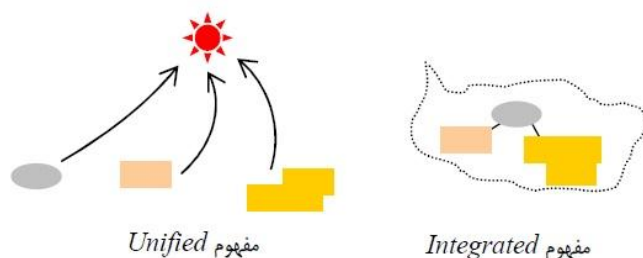
شکل ۶-۳ مقایسه میان درصد پروژه های موفق، شکست خورده، و دارای چالش در طول سال های مختلف

۳.۴ آر.یو.پی چیست ؟

در این بخش مروری اجمالی بر ساختار و مفاهیم کلیدی آر.یو.پی خواهیم داشت. پس از بررسی مفهوم آر.یو.پی ساختار دو بعدی آن معرفی شده و به طور مختصر تشریح می گردد.

۳.۴.۱ فرایند یکپارچه رشنال^۵ (یا به اختصار، آر.یو.پی^۶) چیست؟

اجازه دهید پیش از تعریف آر.یو.پی و آشنایی با معنای دقیق آن، به معنای لغوی آن اشاره ای داشته باشیم. حرف اول این واژه (یعنی حرف آر) مخفف کلمه رشنال می باشد. رشنال نام یکی از شرکت های بزرگ در صنعت نرم افزار است. این شرکت نقش مهمی در توسعه صنعت نرم افزار ایفا نموده است. این شرکت در سال ۲۰۰۳ رسماً توسط شرکت آی.بی.ام^۷ خریداری شد. بنابراین درحال حاضر، مالکیت آر.یو.پی (به عنوان یک محصول) در اختیار شرکت آی.بی.ام می باشد. حرف دوم در واژه آر.یو.پی (یعنی حرف یو) مخفف کلمه یونی فاید^۸ و به معنای تلفیق شده، و متحدالشکل می باشد. توجه داشته باشید که اصطلاح یکپارچه در اینجا به معنای یکی شده بکار می رود و در واقع، یکپارچگی معادل واژه Intergrated مورد نظر نیست.



شکل ۳-۷ مقایسه مفاهیم Unified و Intergrated

⁵ Rational Unified Process

⁶ RUP

⁷ IBM

⁸ Unified

حرف سوم واژه آر.یو.پی (یعنی حرف پی) مخفف واژه پروسس^۹ به معنای فرایند می باشد. البته در این پایان نامه منظور از فرایند، فرایند تولید است؛ فرایند تولید، قالب و الگوی پروژه را تعریف می نماید.

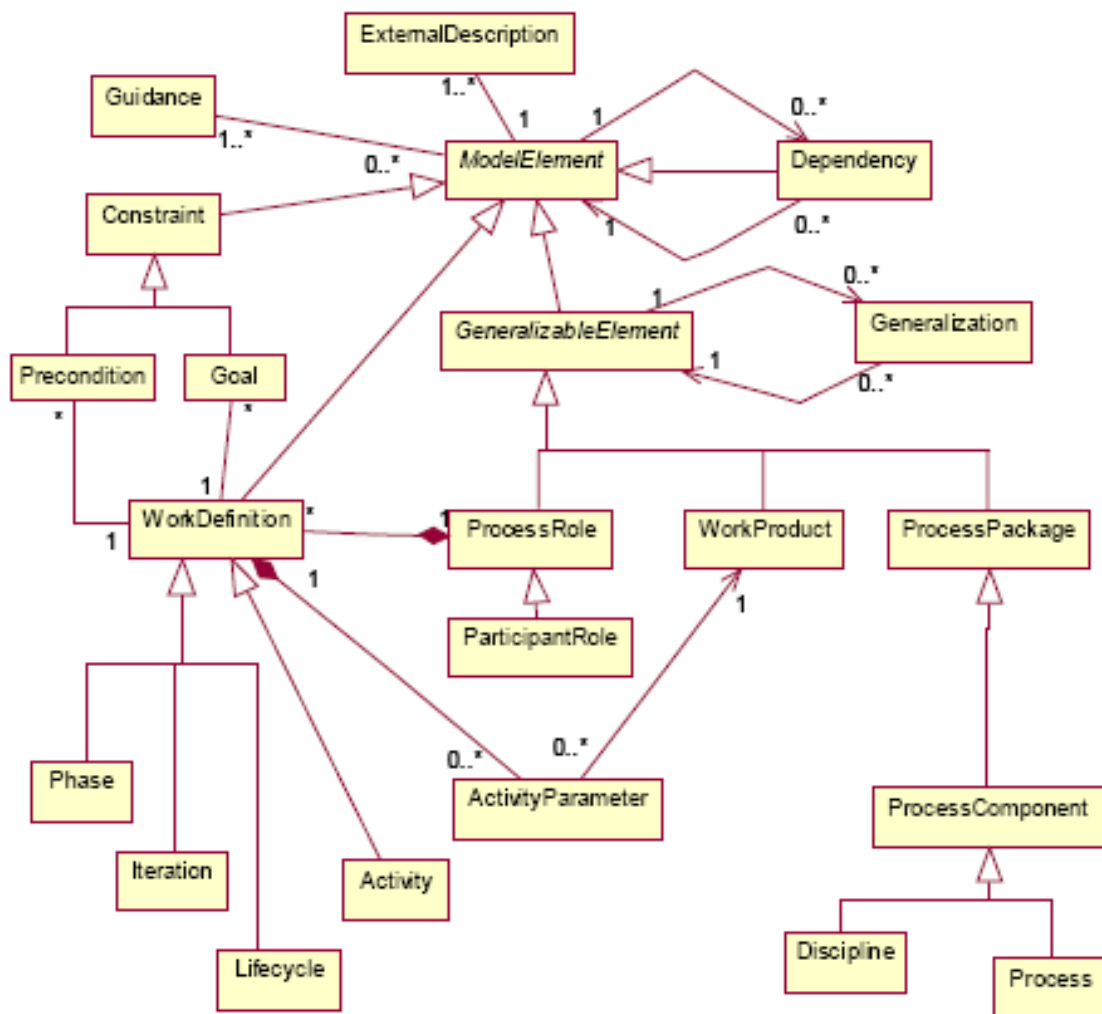


شکل ۳-۸ ارتباط میان فرایند تولید و پروژه

۳.۴.۲ آر.یو.پی به عنوان یک رویکرد مهندسی نرم افزار

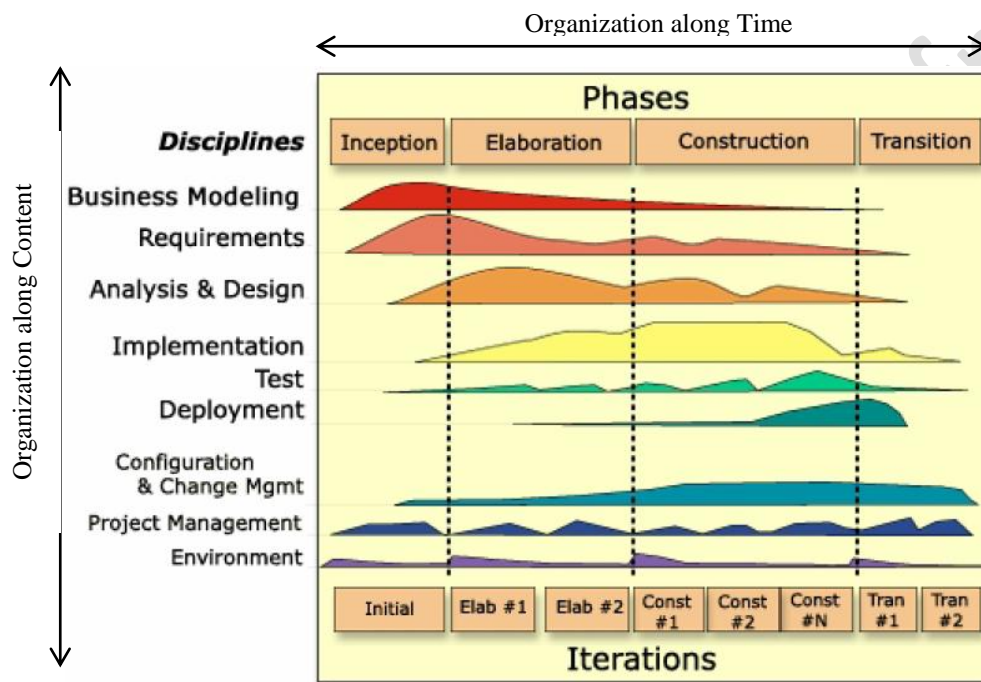
از این منظر آر.یو.پی همانند یک بانک دانش و گنجینه ای از تجارب، الگوها، و راهکارهای موفق در صنعت نرم افزار می باشد. آر.یو.پی خود به عنوان یک فرایند تولید و مهندسی نرم افزار، با استفاده از تکنیک های طراحی و مدل سازی نرم افزار، طراحی شده است. تعریف کامل این فرایند به وسیله ابر مدلی تحت عنوان مدل مهندسی فرایند نرم افزار که استاندارد برای مدل سازی فرایند مبتنی بر زبان مدل سازی استاندارد یو.ام.ال می باشد، صورت پذیرفته است. این مدل مرجع، در شکل ۳-۹ در صفحه بعد نشان داده شده است.

^۹ Process



شکل ۹-۳ مدل مهندسی فرایند نرم افزار؛ ارائه شده توسط OMG

در شکل ۳-۱۰، معماری کلی آریوپی نشان داده شده است. همانگونه که در این شکل ملاحظه می نمایید این فرایند دارای ساختاری دو بعدی است.



شکل ۳-۱۰ سازماندهی فرایند آریوپی در دو بعد زمانی و محتوایی (دینامیک و ایستاتیک)

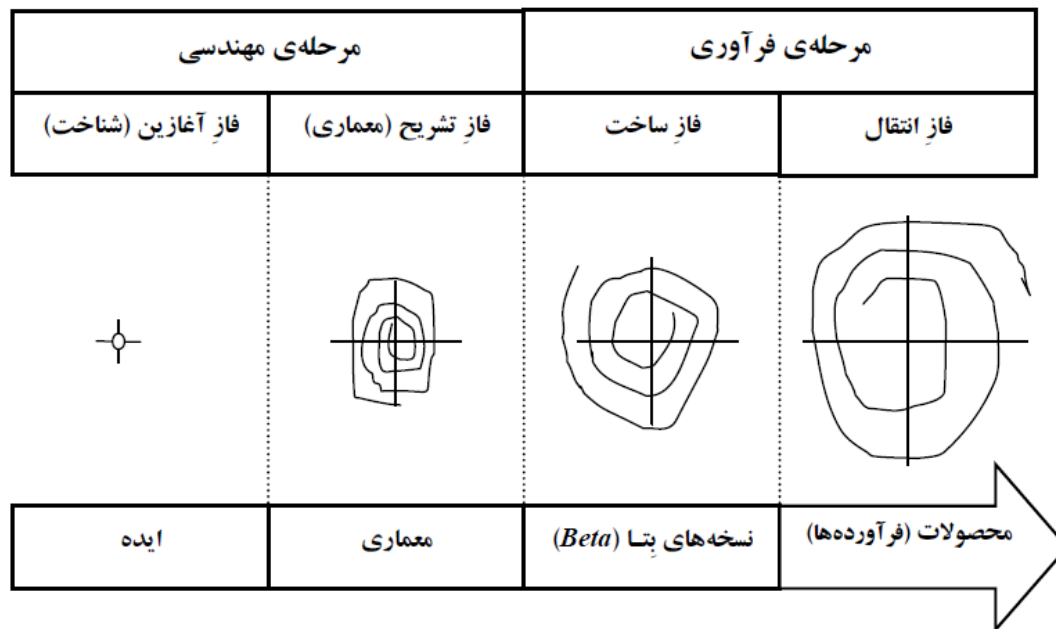
۳.۴.۳ دو بُعد دینامیک و ایستاتیک آریوپی

دو بُعد تشکیل دهنده ساختاری متعامد آریوپی عبارتند از:

ساختار دینامیک^{۱۰} (پویا): ساختار دینامیک آریوپی بعد افقی نشان داده شده در شکل ۳-۱۱ و بیانگر ساختار پویا و ملاحظات مرتبط با زمان در فرایند می باشد. در این بُعد، ملاحظات مانند چرخه

¹⁰ Dynamic

های توسعه (یا چرخه های تولید)، فازها، تکرارها، و نقاط تصمیم گیری کلیدی مطرح می باشد. این مفاهیم در کنار هم، چرخه عمر یک پروژه نرم افزاری (پروژه تولید یک محصول نرم افزاری) را تعریف می نمایند. بر اساس فرایند آر.یو.پی، یک پروژه نرم افزاری که با هدف تولید یک فرآورده نرم افزاری تعریف و اجرا می گردد، از نظر دینامیکی یا بعد زمانی دارای چهار فاز می باشد: فاز آغازین^{۱۱} (یا فاز شناخت)، فاز تشریح^{۱۲} (یا فاز معماری)، فاز ساخت^{۱۳} و فاز انتقال^{۱۴}. همانطور که در شکل قبل شکل ۳-۱۰ ملاحظه می کنید، محور افقی که نشان دهنده زمان انجام پروژه است، از چهار فاز ذکر شده تشکیل می شود.



شکل ۳-۱۱ نمای کلی این چهار فاز نشان داده شده است

¹¹ Inception
¹² Elaboration
¹³ Construction
¹⁴ Transition

ساختار محتوایی^{۱۵} (استاتیک): همانگونه که در شکل ۳-۱۰ نشان داده شده است، ساختار آر.یو.پی دارای یک بعد عمودی نیز می باشد که بیانگر ساختار استاتیک یا محتوایی آن است. در این بعد توصیفی از چگونگی دسته بندی و سازماندهی عناصر محتوایی فرایند یعنی مجموعه فعالیت ها، راهنمایی ها، دستاوردها و نقش ها در قالب دیسپلین ها یا جریانهای منظم و منطقی مجموعه کارها می باشد. لازم به ذکر است که تمامی مؤلفه های ساختار محتوایی آر.یو.پی با زبان مدل سازی استاندارد تحت عنوان یو.ام.ال، مدل سازی شده اند. یک فرایند (و در واقع، یک فرایند تولید) توصیف می نماید که چه کسی، چه کاری را، چگونه، و چه موقع باید انجام دهد تا اینکه به نتیجه مطلوب که یک محصول با کیفیت دست یابیم. از این منظر، نقش ها معادل مؤلفه چه کسی، فعالیت ها معادل مؤلفه چگونگی، دستاوردها معادل مؤلفه چه چیزی، و جریان کار معادل مؤلفه چه موقع در فرایند می باشد.

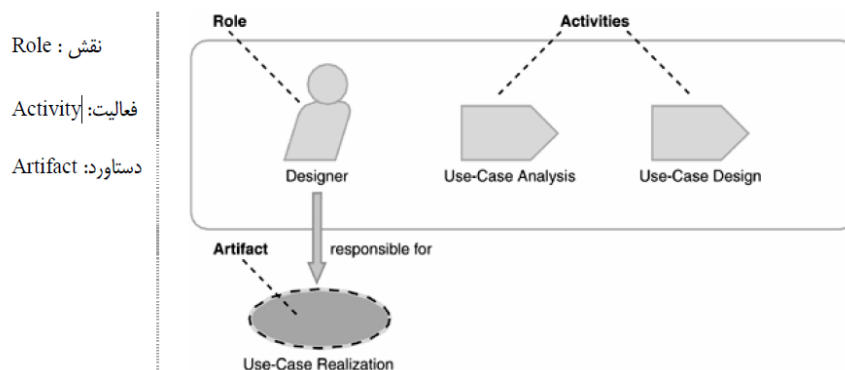
از آنجا که تشریح تمام اجزای ساختار محتوایی؛ نقش ها، فعالیت ها، دستاوردها، دیسپلین ها و جریان کاری از حوصله این مقاله خارج است. فقط توصیفی اجمالی از دیسپلین و جریان کاری خواهیم داشت، که فکر می کنم برای فهم این پایان نامه ضروری است.

۳.۴.۴ تعریف دیسپلین^{۱۶}

دیسپلین در لغت به معنای انضباط و نظم کاری می باشد. در واقع، از آنجایی که بطور مثال مجموعه فعالیت ها، دستاوردها، و نقش هایی که همگی با هدف فراهم نمودن، تجزیه و تحلیل، مدل سازی، و پالایش نیازمندی های نرم افزاری و نیازهای مشتری انجام می شود، دارای یکسری ویژگی های فکری، رفتاری، هدف، و عملکرد مشابهی می باشند که با مجموعه فعالیت ها، دستاوردها، و نقش های دیگر فرایند که هدف آنها تحلیل و طراحی سیستم، پیاده سازی، تست و یا استقرار محصول می باشد، متفاوت است. در واقع مفهوم دیسپلین این ذهنیت را تداعی می نماید که نوعی ارتباط منطقی میان مجموعه فعالیت ها، نقش ها، و دستاوردهای هر دیسپلین وجود دارد.

¹⁵ Content

¹⁶ Disciplines

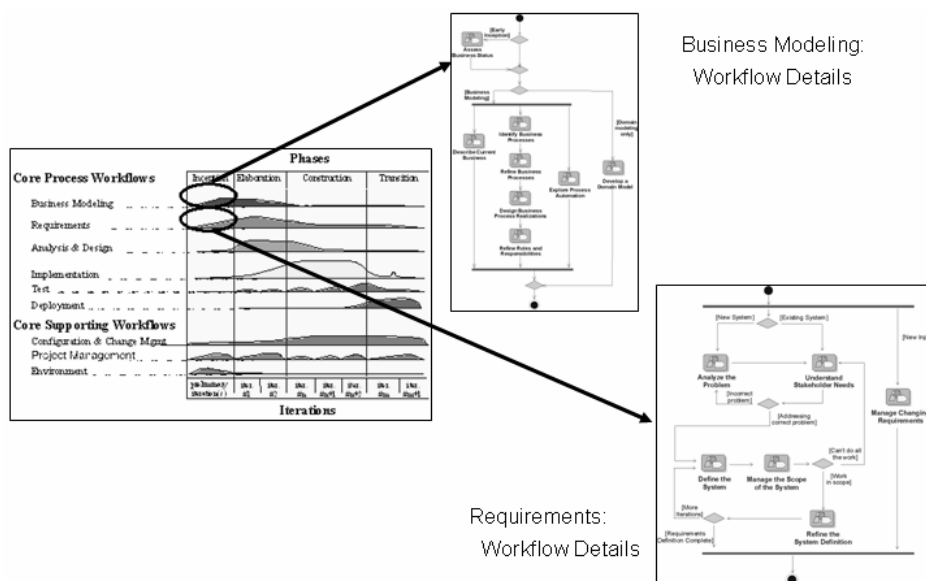


شکل ۳-۱۲ نمونه ای از ارتباط میان عناصر کلیدی در ساختار محتوایی آر.یو.پی

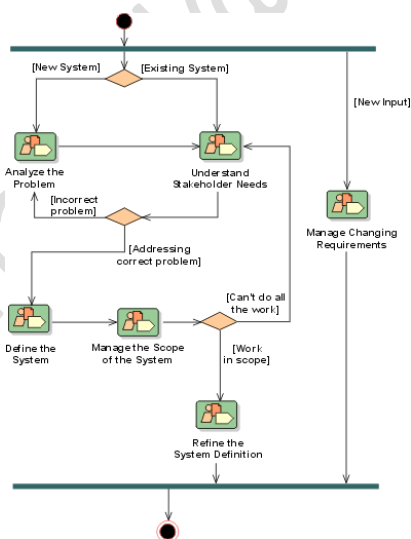
۳.۴.۵ تعریف جریان کاری^{۱۷}

توالی معنادار و منطقی از فعالیت های مختلف است که منجر به ارائه نتیجه ای با ارزش و قابل توجه شده است و در آن چگونگی تعامل میان نقش های مختلف توصیف می شود. دو شکل عمده از جریان کار در آر.یو.پی مطرح می باشد. به ازای هر یک از دیسپلین های تعریف شده، مدلی از یک جریان کار ارائه شده است. این جریان کار که جریان کار اصلی نیز نامیده می شود، بیانگر توالی منطقی مجموعه فعالیت های یک دیسپلین می باشد. از آنجایی که معمولاً فعالیت های متعددی در یک دیسپلین وجود دارد، امکان مدل سازی آنها در قالب یک نمودار فعالیت، درک آن را با مشکل مواجه می کند. بنابراین، آر.یو.پی به منظور مدیریت پیچیدگی، با معرفی مفهوم جزئیات جریان کار یا بسته های جزئی جریان کار، مجموعه فعالیت هایی که نوعی ارتباط منطقی نزدیک تری با هم دارند را در قالب یک بسته کوچک جریان کار دسته بندی می نماید.

¹⁷ Workflow



شکل ۳-۱۳ ارتباط میان دیسپلین های فرایند و جریان کار متناظر با آن



شکل ۳-۱۴ نمونه ایی از یک جریان کاری

۳.۴.۶ دسته های دیسپلین در آر.یو.پی

در آر.یو.پی دیسپلین های موجود به دو دسته تقسیم شده اند. دسته اول، شامل دیسپلین هایی است که ارتباط مستقیمی با شکل گیری ماهیت فراورده نرم افزاری دارند. این دیسپلین ها را دیسپلین های اصلی می نامند. این دیسپلین ها عبارتند از:

- دیسپلین مدل سازی سازمان^{۱۸}
- دیسپلین نیازمندی ها^{۱۹}
- دیسپلین تحلیل و طراحی^{۲۰}
- دیسپلین پیاده سازی^{۲۱}
- دیسپلین تست^{۲۲}
- دیسپلین استقرار^{۲۳}

دسته دوم، سه دیسپلین می باشد که عمدتاً نقش پشتیبانی و مدیریتی دارند. این سه دیسپلین را که دیسپلین های پشتیبانی نیز می نامند، عبارتند از:

- دیسپلین مدیریت پروژه^{۲۴}
- دیسپلین مدیریت تغییرات و پیکربندی^{۲۵}
- دیسپلین محیط^{۲۶}

¹⁸ Business Modeling

¹⁹ Requirements

²⁰ Analysis and Design

²¹ Implementation

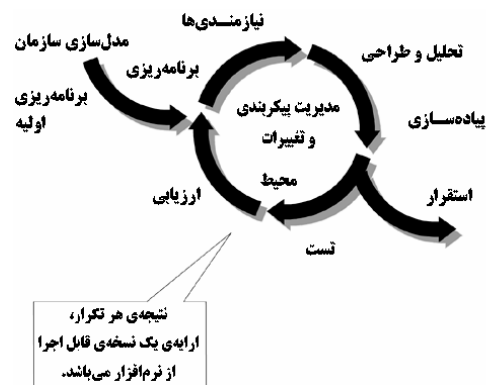
²² Test

²³ Assessment

²⁴ Project Management

²⁵ Change & Configuration Management

²⁶ Deployment



شکل ۳-۱۵ یک تکرار در چرخه تولید شامل تمام دیسپلین ها می باشد

توجه داشته باشید که این دیسپلین ها و جریان فعالیت هایی که متناظر با هر یک از آنها توسط آ.یو.پی مطرح شده است، مفاهیم ثابت و بدون تغییری نیستند. ممکن در شرایط مختلف بر حسب نوع پروژه و ماهیت فعالیت های آن و یا در سازمان های مختلف، با طیف متفاوتی از دیسپلین ها برخورد داشته باشیم.

۴. فصل سوم

روش انجام پروژه

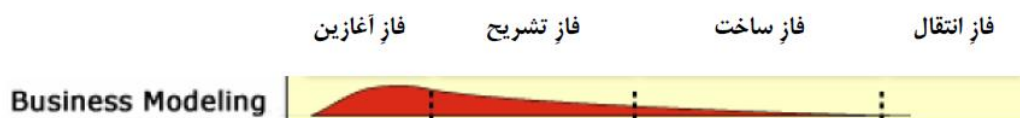
۴.۱ مقدمه

در این فصل، فرایند تولید سیستم نرم افزاری باربری را همگام با دیسپلین های مورد نظر از فرایند تولید آر.یو.پی تشریح می کنیم. در ابتدای هر بخش تعریف اجمالی از خود دیسپلین و اهداف آن خواهیم داشت و در ادامه به نحوه اجرای جریان کار هر دیسپلین در پروژه سیستم نرم افزاری خواهیم پرداخت. پس از بیان مباحث فوق الذکر، خروجی هر دیسپلین (یا دستاوردها) را عنوان و بررسی می کنیم. در انتها، خروجی ها (یا دستاوردها) را که در نرم افزار رشنال رز پیاده سازی شده است، با ذکر نمونه ایی بیان و تشریح می کنیم. قبل از مطالعه این فصل توجه به چند نکته زیر در مورد این پایان نامه ضروری است:

- تمامی تعاریف تئوریک این فصل از کتاب، معرفی آر.یو.پی (مشخصات کامل این کتاب در قسمت منابع ذکر شده است) بر گرفته شده است و برای جلوگیری از تحریف مفاهیم پایه، بعضی از مطالب عیناً مثل خود کتاب فوق الذکر بیان شده اند.
- جای شک نیست که ممکن است؛ تحلیل ها، دستاوردها و در کل استنباط از سیستم نرم افزاری باربری که در این پایان نامه بیان شده است، عاری از نقص نباشد. اما همانطور که می دانید فرایند تولید یک نرم افزار، فرایندی است تکاملی و با تکرار هر قسمت، می توان نقایص را بهینه کرد (در ادامه توضیح بیشتر خواهیم داد).
- نرم افزار استفاده شده برای مدل سازی خروجی ها (یا دستاوردها)، نرم افزار رشنال رز متعلق به شرکت آی.بی.ام می باشد و زبان مدل سازی خروجی ها، زبان مدل سازی یو.ام.ال است.
- دیسپلین هایی که در این پروژه استفاده کرده ایم، متناظر با چهار دیدگاه نرم افزار رشنال رز است و از دیگر دیسپلین ها به علت عملی و تجربی بودن صرف نظر کرده ایم.
- اگر به دقت به شکل ۳-۱۰ نگاه کنید، متوجه می شود که بعضی از دیسپلین ها بصورت تقریباً برابر در هر فاز فعالیت می کنند، ولی بعضی دیگر در یک فاز بیشتر از فاز های دیگر فعالیت می کنند. این امر باعث تکرار دیسپلین ها در طول فرایند تولید می شود، و دلیل این تکرارها؛ مقابله با ضعفها و نقصها است.

۴.۲ دیسیپلین مدل سازی از سازمان

در این بخش، به دیسیپلین مدل سازی سازمان (کسب و کار) در سیستم نرم افزاری باربری می پردازیم. در شکل ۴-۱ همانطور که مشاهده می کنید این دیسیپلین بیشترین فعالیت خود را در فاز آغازین دارد و ممکن است با تکرارهایی که در این دیسیپلین صورت می گیرد، تا فاز ساخت ادامه پیدا کند.



شکل ۴-۱ نمایشی مقطعی از دیسیپلین مدل سازی سازمان در طول فازهای چرخه تولید

اهدافی که در این مدل دنبال می کنیم عبارتند از:

- ۱- درک ساختار و پویایی سازمان باربری
- ۲- درک مسائل و مشکلات جاری سازمان باربری
- ۳- استنتاج و استخراج نیازمندی های سیستم که باید سازمان باربری را در دستیابی به اهدافش پشتیبانی کند.

برای نیل به این اهداف، دیسیپلین مدل سازی سازمان، چگونگی توسعه یک چشم انداز از سازمان جدید را توصیف نموده و بر اساس این چشم انداز، چگونگی تعریف فرایندها، اکتورها، و مسئولیت های سازمان را در یک مدل از سازمان بیان می نماید.

۴.۲.۱ درک ساختار و پویایی سازمان باربری

در سازمان باربری که بصورت سنتی مشتری ها درخواست جابجایی کالای خود را می دهند، تمامی این روند درخواست جابجایی به صورت دستی انجام می شود و لازم است که مشتری در نمایندگی سازمان باربری حضور داشته باشد. درخواست های مشتری بر روی کاغذ ثبت می شوند و همه این درخواست ها بعد از تأیید شدن در یک مجموعه بزرگتری به نام بارنامه قرار می گیرند. در این روند با توجه به زمان و مقصد جابجایی، درخواست ها در بارنامه ها جای می گیرند. در نتیجه، هر بارنامه شامل درخواست هایی است که دارای زمان و مقصد یکسان است. در این سازمان باربری هر راننده در مسیری مشخصی تردد می کند. رانندها موظف اند در ایستگاه های که برای آنها در نظر گرفته شده است توقف کنند و بار مربوط به آن ایستگاه (یا شهر) را تخلیه کنند.

حال بارنامه ها که شامل درخواست هایی با مقصد و زمان یکسانی هستند منتظر می مانند تا تعداد درخواست ها به اندازه ایی برسد که آماده برای جابجایی شده و از لحاظ مادی مقرون به صرفه باشد. در ادامه این روند بارنامه ایی که آماده برای جابجایی شده است در قسمتی دیگر با توجه به مقصد آن که اگر در مسیر راننده ایی باشد بارنامه به لیست بار راننده انتقال پیدا می کند و در ایستگاه خود در لیست بار ثبت می شود. اگر بخواهیم از زمانی که مشتری درخواست جابجایی کالا می دهد این روند را مرور کنیم، می توانیم بگوییم؛ مشتری با حضور در نمایندگی درخواست جابجایی می دهد، درخواست بروی کاغذ مستند می شود، در خواست ها با توجه به مقصد و زمان آنها در بارنامه ها قرار می گیرند. در ادامه بارنامه

ها در لیست بار رانندها درایستگاه متناظر با مقصد خود قرار گرفته، ثبت نهایی می شوند و آماده برای جابجایی خواهند شد.

در این سازمان مدیر می بایست بر قسمت های مختلف این سازمان مدیریت کند. مدیر در واقع کنترل کننده تمام بارنامه ها، ایستگاهها، لیست بار راننده ها و مسیر تردد راننده ها می باشد و تعیین کننده تمام محدودیت های سازمان مثل مسیر تردد آنها، تعیین تعداد درخواست ها در هر بارنامه، قیمت ها، زمان کار و خیلی موارد دیگر است. یکی دیگر از مهم ترین فعالیت های مدیر این سازمان گزارش گیری از قسمت های مختلف است، مثل لیست بار رانندها و بارنامه ها می باشد. حال این گزارش ها بر اساس زمان، مقصد، هزینه، اطلاعات مشتری و دیگر موارد تهیه می شود و به دست مدیر خواهد رسید. همچنین مدیر این سازمان قادر به تغییر و اضافه و یا حذف کردن موردی در درخواست ها، بارنامه ها، لیست بار راننده ها، ایستگاه ها، مسیر تردد راننده ها می باشد. در این قسمت با روند کار سازمان باربری سنتی که هنوز فرایند سفارش گیری آن ماشینی و یا در واقع مبتنی بر یک سیستم نرم افزاری نشده است آشنا شدیم.

۴.۲.۲ درک مسائل و مشکلات جاری سازمان باربری

اولین مسئله ایی که در سازمان باربری سنتی به چشم می خورد، مسئله حضور مشتری در نمایندگی سازمان است. مشتری برای اینکه بتواند درخواست جابجایی کالا بدهد و یا اینکه از دیگر خدمات سازمان استفاده کند باید حتماً در سازمان حضور فیزیکی داشته باشد. این مسئله حضور فیزیکی خود مشکلاتی را به همراه خواهد داشت، که عبارتند از:

- ۱- مشتری برای حضور باید هزینه و انرژی صرف کند
- ۲- حضور مشتری در سازمان بخصوص در زمان هایی که اوج کار کارمندان است باعث اختلال در کار آنها می شود
- ۳- زمان بیشتری نیاز است تا مشتری به هدف خود برسد
- ۴- از بین رفتن رغبت مشتری برای استفاده از دیگر خدمات سازمان به جزء جابجایی کالا که ممکن است سود خوبی برای سازمان داشته باشد
- ۵- اطلاع رسانی ضعیفتر خواهد شد.

مسئله بعدی استفاده از کاغذ است، مصرف کاغذ خود هزینه بسیاری را برای سازمان به همراه خواهد داشت. از طرفی چاپ فرم ها و جابجایی آنها از مرکز به نمایندگی ها برای سازمان هزینه خواهد داشت، که هنوز ما هزینه افراد مسئول این کارها را حساب نکردیم. در مجموع می توان گفت استفاده از کاغذ از لحاظ هزینه، حجم زیاد، انرژی، زمان، حفظ امنیت، افراد مسئول و خیلی موارد دیگر برای سازمان مقرون به صرفه نخواهد بود.

از دیگر مسائلی که در یک سازمان سنتی باربری مشکل ایجاد می کند، مسئله انجام امور و ارائه خدمات به صورت دستی است. مشکل آن است که روند دستی، درصد خطا را در سازمان افزایش می دهد و مانع از کیفیت مطلوب خدمات سازمان می گردد. حال اینکه خطاها ممکن است یا از سوی مشتریان در هنگام پر کردن فرم های سفارش رخ دهد و یا از سوی کارمندان سازمان یا از سوی راننده ها و یا مدیران باشد.

مسائل زیادی در سازمان باربری سنتی وجود دارد که می توان بر آنها تکیه کرد تا دلایلی محکمی برای مدرنیزه شدن سازمان ارائه دهیم. مسائلی همچون عقب ماندگی سازمان از دیگر سازمان های موجود که دیگر قادر به رقابت با آنها نمی باشد، هزینه های بالا روند سنتی، صرف زمان و هزینه زیاد برای آنالیز فعالیت ها، پایین آمدن کیفیت خدمات، کاهش وفاداری مشتری به سازمان، کارا نبودن سازمان سنتی، در دسترس نبودن در هر زمان و هر مکان، مدیریت ضعیف و ناکارآمد، نارضایتی مشتریان از روند سنتی، افزایش خطا در سازمان و.... که این مسائل خود مشکلات بزرگی را برای سازمان به همراه خواهد داشت.

۴.۲.۳ استنتاج و استخراج نیاز مندیهای سازمان باربری

یک سازمان باربری سنتی مسائل و مشکلات زیادی دارد که برای از بین بردن این مشکلات، یکسری نیازها را خواستار می باشد. با پاسخ دادن به این نیازها در واقع مشکلات سازمان را از بین برده ایم، در این قسمت می خواهیم نیازهای سازمان حمل و نقل را با توجه به مشکلات آن دریابیم.

۱- نیاز به بستر اینترنت و سیستم سفارش گیری، که با این امر دیگر به حضور فیزیکی مشتری در سازمان نمی باشد

۲- نیاز به سیستم پایگاه داده هم در بستر شبکه اینترنت و هم در بستر شبکه محلی سازمان، برای مدیریت همه جانبه سازمان حمل و نقل در هر زمان و مکان

۳- نیاز به یکپارچه سازی داده های سازمان باربری، برای انواع گزارش گیری از داده های سازمان و دست کاری آنها. با این سیستم می توان داده های خام سازمان را تبدیل به اطلاعات مفیدی کرد که در راستای بهتر شدن سطح کیفی خدمات، سفارشی کردن خدمات برای هر مشتری و مدیریت خیلی بهتر بر سازمان استفاده نمود

۴- نیاز به سیستمی که تمام آنالیزهای سازمان را در مدت زمان، هزینه و نیروی کار انسانی کمتری انجام دهد

۵- نیاز به سیستمی که درصد خطای سازمان را به اندازه قابل توجه ای کاهش دهد.

ممکن است نیازهای دیگری بتوان برای این سازمان در نظر گرفت اما بیش از این حجم پروژه را افزایش می دهد، که این خود یک تیم مجرب را برای انجام این پروژه می طلبد.

۴.۲.۴ چشم انداز سازمان باربری

بر آنیم تا تمام فرایندهای سنتی باربری را مدرنیزه کنیم، این فرایندها که بصورت دستی اعمال می شد توسط سیستم نرم افزاری باربری انجام شود تا مشکلات سازمان که در گذشته با آن روبرو بود از بین برود. مشتری در هر مکان و زمان با دسترسی به اینترنت سفارش حمل بار می دهد. خدمات ویژه ای مثل گزارش از وضعیت بار برای مشتری در نظر شده است، مشتری با این سیستم آگاه می شود که بارش در حال حاضر در کجا قرار دارد و کی به مقصد می رسد. درون سیستم نرم افزاری باربری الگوریتمی طراحی شده است که درخواست ها با مقصد و زمان یکسان در بارنامه ها قرار می گیرند، و هر بارنامه با توجه به مقصد در یک رشته از مسیر راننده قرار می گیرند. در نهایت سیستم به راننده اعلام می کند که کدام بارنامه ها مطعلق به اوست و چه زمانی باید در کدام مسیر حرکت کند. حال بعد از اینکه راننده در مسیر خود قرار گرفت، به هر ایستگاهی که برسد کد ایستگاه مستقر در آن را برای سیستم از طرق مختلف مثل پیام کوتاه و یا اینترنت ارسال می کند. سیستم متوجه می شود که راننده در کدام ایستگاه قرار دارد موقعیت بار را در پایگاه داده خود ذخیره می کند، حال زمانی که مشتری برای گرفتن گزارش از موقعیت بار به صفحه اینترنتی سیستم باربری مراجعه می کند، اطلاعات بار از مشتری گرفته می شود. در پردازنده ایی که در مرکز سازمان باربری قرار دارد اطلاعات بررسی می شود. سیستم بررسی می کند که

بار شخص در کدام بارنامه قرار دارد و موقعیت بار را با توجه به اطلاعاتی که از طرف راننده ها در پایگاه داده سیستم ذخیره شده است، برای سیستم مشتری (یا کلاینت) می فرستد.

مدیر که یک رکن (یا رُل) اصلی در سیستم باربری است. رایانه مدیر توسط لینکی که به پایگاه داده مرکزی سیستم باربری متصل شده است چه در بستر شبکه اینترنت جهانی و چه در بستر شبکه محلی سازمان، بعد از گذر از موانع امنیتی برای تشخیص هویت مدیر می تواند مدیریت کامل بروی اطلاعات، محدودیتها، قیمتها، مسئولیتها، زمان کاری و بسیاری موارد دیگر داشته باشد. این مدیریت در هر زمان و مکان قابل اجرا است.

در این قسمت، چشم اندازی از سازمان مدرنیزه شده باربری که ذینفعان سازمان از آن انتظار دارند توصیف کردیم. در ضمن به این مهم توجه کنید که این چشم انداز در واقع یک دید مثبت و ایده آل از سیستم است و ممکن است در طی انجام پروژه سیستم باربری با مشکلات روبرو شویم که از کیفیت این توصیفات بکاهد.

۴.۲.۵ شناسایی اکتورها و اهداف آنها

بعد از مستند سازی چشم انداز سازمان، به دنبال تعریف اکتورهای کلیدی، فرایندها و اهداف آنها در این چشم انداز خواهیم بود. البته فرایندهای این اکتورها برای رسیدن به اهدافی که در سیستم دنبال می کنند در قسمت های قبل به تفصیل توصیف شدن، توصیف دوباره آنها ایجاد افزونگی در این پایان نامه خواهد کرد.

اکتور مشتری: شخصی که حضور مجازی در سیستم دارد و درخواست خدمات می کند.

اهداف:

۱- درخواست فرم سفارش برای جابجایی کالا

۲- درخواست گزارش از موقعیت بار.

اکتور راننده: شخصی که مسئول انجام خدمات جابجایی کالا های مشتریان در سازمان باربری می باشد.

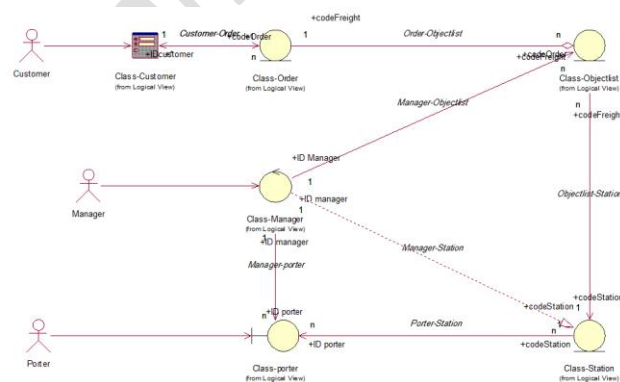
اهداف:

- ۱- درخواست لیست بار
 - ۲- گزارش دادن موقعیت بار.
- اکتور مدیر: شخصی که مدیریت تمام قسمت های سیستم را بر عهده دارد، مدیریت در آنالیز و پایگاه داده سیستم نرم افزاری باربری.

- ۱- گزارش گیری بر اساس زمان، مقصد، قیمت، حجم بار، اطلاعات مشتری و دیگر موارد
- ۲- ویرایش تمام اطلاعات مربوط به بارنامه ها
- ۳- ویرایش تمام اطلاعات مربوط به راننده ها
- ۴- ویرایش تمام اطلاعات مربوط به ایستگاه ها.

۴.۲.۶ استخراج مدل تحلیل سازمان

بعد از اینکه اکتورها و نقش آنها، فرایندها و اهداف آنها را در سازمان باربری یافتیم مدل ریز را ارائه می دهیم. همچنین مدل رشنال رز متناظر با مدل زیر در دیدگاه یوزکیس^{۲۷}، با نام بیزینس مدلینگ^{۲۸} قرار دارد در صورت نیاز به آن مراجعه فرمایید. فایل رشنال رز پروژه ضمیمه پایان نامه شده است.



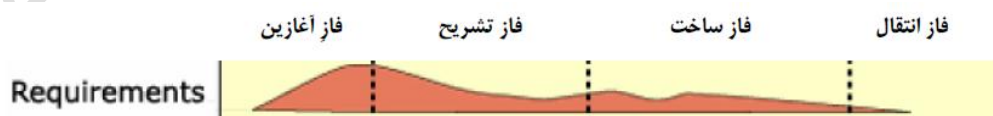
شکل ۴-۱ مدل تحلیل سازمان باربری

²⁷ Usecase view

²⁸ Business Modeling

۴.۳ دیسیپلین مدیریت نیازمندی ها

دیسیپلین نیازمندی ها یا دیسیپلین مدیریت نیازمندیها، مجموعه فعالیتها، دستاوردها، و نقشهایی است که با بهره گیری از روشها و تکنیکهای موفقیت آمیز مانند مدل سازی نیازمندیها به وسیله موارد کاربرد، ما را در جمع آوری، سازماندهی، و مدیریت نیازمندیهای یک سیستم، یاری می دهند.



شکل ۴-۲ نمایی مقطعی از دیسیپلین نیازمندی ها در چرخه تولید

اهداف دیسیپلین نیازمندی ها به شرح زیر می باشد:

- ۱- برقراری و حفظ توافق میان مشتری و سایر ذینفعان با تیم تولید درباره چستی و چرایی سیستم
- ۲- فراهم نمودن درک مناسبی از نیازمندی های سیستم برای تولید کنندگان آن
- ۳- تعریف یک واسط کاربر که بر نیازها و اهداف کاربران متمرکز می باشد.

برای دستیابی به این اهداف، دیسیپلین نیازمندی ها براساس چشم انداز تعریف شده به ترجمه آن به مدل مواردکاربرد و مجموعه مشخصه های تکمیلی که جزئیات نیازمندیهای سیستم (یا سناریوها) را در بر خواهد داشت، تشریح می نماید. فعالیت های که در دیسیپلین نیازمندی ها انجام می شود عبارتند از:

- ۱- شناسایی اکتورها و مواردکاربرد
- ۲- جزئیات مواردکاربرد (یا همان سناریوها)
- ۳- استخراج مدل مواردکاربرد
- ۴- ایجاد نمونه ای از واسط کاربر.

۴.۳.۱ شناسایی اکتورها و مواردکاربرد

این فعالیت در دیسیپلین مدل سازی از سازمان انجام گرفته شد. چون سیستم نرم افزاری که ما در نظر گرفته ایم چندان پیچیده نمی باشد، نیازی به بررسی بیشتر برای شناسایی اکتورها و مواردکاربرد نیست. ولی در پروژه های نرم افزاری بزرگ و پیچیده که از فرایند تولید آر.یو.پی استفاده می کنند، این امر بررسی مجدد ضروری است.

۴.۳.۲ جزئیات موارد کاربرد (یا همان سناریوها)

جدول ۴-۱ سناریوی مورد کاربرد ورود

مورد کاربرد	ورود
هدف	به مشتری اجازه می دهد تا از امکانات سیستم در سطح اختیارت خود استفاده کند
عامل اصلی	مشتری
شرایط اولیه	کاربر گزینه ورود را انتخاب کند
جریان اصلی	<p>۱- این مورد از کاربرد وقتی آغاز می شود که مشتری درخواست ورود به سیستم را داشته باشد .</p> <p>۲- سیستم در خواست اطلاعات لازم برای احراز هویت می کند .</p> <p>۳- مشتری اطلاعات لازم در مورد مشخصات خود را وارد می کند .</p> <p>۴- سیستم اطلاعات وارد شده را پایش می کند .</p> <p>۵- سیستم به مشتری اجازه ورود می دهد .</p>
شرایط نهایی	کاربر وارد سیستم شده است .
جریان فرعی	<p>۱- اطلاعات وارد شده توسط مشتری صحیح نباشد .</p> <p>۲- پیغام هشدار که اطلاعات وارد شده صحیح نمی باشد نمایش داده می شود .</p>

جدول ۴-۲ سناریوی مورد کاربرد خروج

مورد کاربرد	خروج
هدف	مشتری از سیستم خارج می شود و تغییرات ایجاد شده ذخیره می شود .
عامل اصلی	مشتری
شرایط اولیه	مشتری وارد سیستم شده باشد .
جریان اصلی	<p>۱. این مورد از کاربرد وقتی آغاز می شود که مشتری درخواست خروج از سیستم را داشته باشد.</p> <p>۲. پیغام هشدار که آیا می خواهید خارج شوید نمایش داده می شود .</p> <p>۳. در صورت انتخاب گزینه خروج ، سیستم تغییرات در اطلاعات مربوط به مشتری را پایش و ذخیره می کند.</p> <p>۴. مشتری از سیستم خارج می شود.</p>
شرایط نهایی	مشتری از سیستم خارج شده است.
جریان فرعی	<p>۱. در صورتی که سیستم اطلاعات را پایش کند و صحیح نباشد.</p> <p>۲. پیغام هشدار که ، اطلاعات صحیح نمی باشد و گزینه خروج بدون ثبت نمایش داده می شود.</p> <p>۳. اگر مشتری گزینه بدون ثبت را انتخاب کند سیستم بدون تثبیت اطلاعات مشتری را از سیستم خارج می کند.</p> <p>۴. اگر مشتری پیغام را لغو کند به پنجره قبلی بر می گردد.</p>

جدول ۳-۴ سناریوی مورد کاربرد گرفتن گزارش از موقعیت بار

مورد کاربرد	گرفتن گزارش از موقعیت بار
هدف	در این مورد کاربرد مشتری از این که بارش در چه مرحله ای از جابجایی قرار دارد مطلع خواهد شد .
عامل اصلی	مشتری
شرایط اولیه	مشتری وارد سیستم شده باشد .
جریان اصلی	<p>۱- مشتری درخواست گرفتن گزارش از وضعیت بار را داشته باشد .</p> <p>۲- سیستم درخواست کد باربری می کند (کد باربری قبلاً به هنگام سفارش ، توسط سیستم به مشتری اعلام شده است) .</p> <p>۳- در صورت صحیح بودن کد باربری مشتری وارد این قسمت خواهد شد .</p> <p>۴- اطلاعات وضعیت بار مربوط به کد باربری وارد شده نمایش داده خواهد شد .</p>
شرایط نهایی	مشتری از موقعیت بار خود مطلع خواهد شد .
جریان فرعی	<p>۱- در صورتی که کد باربری صحیح نباشد .</p> <p>۲- پیغام هشدار که کد باربری صحیح نمی باشد نمایش داده خواهد شد .</p> <p>۳- سیستم به پنجره گرفتن گزارش از موقعیت بار بر می گردد .</p>

جدول ۴-۴ سناریوی مورد کاربرد سفارش دادن برای حمل بار

مورد کاربرد	سفارش دادن برای حمل بار
هدف	در این مورد کاربرد مشتری می تواند درخواست برای حمل بارش به مقصد مورد نظر را محقق سازد .
عامل اصلی	مشتری
شرایط اولیه	مشتری گزینه سفارش حمل بار را انتخاب کرده باشد .
جریان اصلی	<p>۱- مشتری وارد قسمت سفارش حمل بار خواهد شد .</p> <p>۲- فرم مربوط به مشخصات فردی و اطلاعات بارنامه را تکمیل می کند .</p> <p>۳- گزینه ثبت را انتخاب می کند .</p> <p>۴- سیستم اطلاعات را پردازش می کند و اطلاعات لازم نظیر هزینه جابجایی و زمان تحویل در مبدأ و مقصد و را نمایش می دهد.</p> <p>۵- در صورت انتخاب گزینه ثبت نهایی سفارش حمل بار در سیستم ذخیره خواهد شد .</p>
شرایط نهایی	سفارش حمل بار مشتری در سیستم ثبت خواهد شد .
جریان فرعی	<p>۱- در صورتی که مشتری هنگام ثبت نهایی درخواست انصراف از سفارش را بدهد .</p> <p>۲- پیغام هشدار نمایش داده خواهد شد .</p> <p>۳- سیستم اطلاعات مربوط به سفارش این مشتری را حذف خواهد کرد .</p> <p>۴- سیستم به صفحه نخست بر خواهد گشت .</p>

جدول ۴-۵ سناریوی مورد کاربرد اعلام موقعیت بار در هر ایستگاه

مورد کاربرد	اعلام موقعیت بار در هر ایستگاه
هدف	در این مورد کاربرد راننده به هر ایستگاه که خواهد رسید به سیستم اعلام می کند .
عامل اصلی	راننده
شرایط اولیه	راننده در ایستگاه مورد نظر برای اعلام موقعیت بار، حضور داشته باشد .
جریان اصلی	<p>۱- راننده یک پیام کوتاه حاوی اطلاعات لازم برای احراز هویت به سیستم مرکزی ارسال می کند .</p> <p>۲- سیستم اطلاعات را پردازش می کند اگر صحیح بود یک پیام تأیید به راننده ارسال می کند و منتظر دریافت کد ایستگاه خواهد ماند .</p> <p>۳- راننده کد ایستگاهی که در آنجا قرار دارد و کد بارنامه را توسط یک پیام به سیستم مرکزی اعلام می کند .</p> <p>۴- موقعیت بار در سیستم ذخیره خواهد شد .</p> <p>۵- یک پیام ثبت موقعیت از سیستم به راننده ارسال خواهد شد .</p>
شرایط نهایی	سیستم از موقعیت بار مطلع خواهد شد و در سیستم جهت مشاهده مشتری و مدیر ذخیره خواهد شد .
جریان فرعی	<p>۱- در صورتیکه اطلاعات راننده برای احراز هویت صحیح نباشد و یا کد ایستگاه صحیح یا تکراری باشد.</p> <p>۲- سیستم یک پیام هشدار به راننده ارسال می کند و درخواست ارسال دوباره اطلاعات از راننده می نماید .</p>

جدول ۴-۶ سناریوی مورد کاربرد گرفتن اطلاعات مسیر و بارنامه

مورد کاربرد	گرفتن اطلاعات مسیر و بارنامه
هدف	در این مورد کاربرد اطلاعاتی که راننده برای تعیین مسیر و بار خود لازم دارد در اختیارش قرار می گیرد .
عامل اصلی	راننده
شرایط اولیه	راننده گزینه گرفتن اطلاعات مسیر و بارنامه را انتخاب کند .
جریان اصلی	<p>۱- راننده درخواست گرفتن اطلاعات مسیر و بارنامه به سیستم را داشته باشد .</p> <p>۲- سیستم درخواست اطلاعات لازم از راننده برای احراز هویت می کند .</p> <p>۳- سیستم اطلاعات را پردازش کرده ، در صورت صحیح بودن اطلاعات، وارد این قسمت می شود .</p> <p>۴- اطلاعات مربوط به مسیر و بارنامه نمایش داده خواهد شد .</p>
شرایط نهایی	سیستم راننده ها را اطلاع رسانی کرده است .
جریان فرعی	<p>۱- در صورتی اطلاعات احراز هویت صحیح نباشد .</p> <p>۲- سیستم پیغام هشدار نمایش می دهد .</p> <p>۳- سیستم به پنجره اصلی بر خواهد گشت .</p>

جدول ۴-۷ سناریوی مورد کاربرد گرفتن گزارش از بارنامه

مورد کاربرد	گرفتن گزارش از بارنامه
هدف	در این مورد کاربرد مدیر می تواند انواع گزارش از بارنامه که در روز، یا در ماه و یا در سال سفارش شده اند ، گزارش تهیه کند .
عامل اصلی	مدیر
شرایط اولیه	مدیر گزینه گرفتن گزارش از بارنامه را انتخاب کند .
جریان اصلی	۱- مدیر درخواست گرفتن گزارش از سیستم را داشته باشد . ۲- سیستم درخواست اطلاعات لازم از مدیر برای احراز هویت می کند . ۳- در صورت صحیح بودن اطلاعات مدیر وارد این قسمت خواهد شد . ۴- مدیر با استفاده از امکانات سیستم انواع گزارش گیری را انجام می دهد . ۵- مدیر گزینه خروج را انتخاب می کند .
شرایط نهایی	مدیر انواع گزارش گیری را در مکانی از حافظه به همراه تاریخ ذخیره خواهد کرد .
جریان فرعی	۱- در صورت اینکه اطلاعات لازم برای احراز هویت مدیر صحیح نباشد . ۲- پیغام هشدار نمایش داده می شود . ۳- سیستم به پنجره اصلی بر خواهد گشت .

جدول ۴-۸ سناریوی مورد کاربرد مدیریت بارنامه ها

مورد کاربرد	مدیریت بارنامه ها
هدف	در این مورد کاربرد مدیر می تواند با استفاده از امکانات سیستم بارنامه ها را ویرایش کند .
عامل اصلی	مدیر
شرایط اولیه	مدیر گزینه مدیریت بارنامه ها را انتخاب کند .
جریان اصلی	۱- مدیر درخواست مدیریت بارنامه ها از سیستم را داشته باشد . ۲- سیستم درخواست اطلاعات لازم از مدیر برای احراز هویت می کند . ۳- در صورت صحیح بودن اطلاعات مدیر وارد این قسمت خواهد شد . ۴- مدیر می تواند از امکانات سیستم در این قسمت استفاده کند و تغییرات لازم را اعمال کند . ۵- مدیر گزینه ثبت را انتخاب می کند و سیستم تغییرات لازم را ذخیره می کند . ۶- در صورت انتخاب گزینه خروج پیغام هشدار نمایش داده خواهد شد و مدیر از سیستم خارج می شود .
شرایط نهایی	تغییرات اعمال شده در بارنامه ها توسط مدیر در سیستم ثبت خواهد شد و لیست ویرایش شده بارنامه ها نمایش داده خواهد شد .
جریان فرعی	۱- در صورت اینکه اطلاعات لازم برای احراز هویت مدیر صحیح نباشد . ۲- پیغام هشدار نمایش داده می شود . ۳- سیستم به پنجره اصلی بر خواهد گشت .

جدول ۴-۹ سناریوی مورد کاربرد مدیریت ایستگاه ها

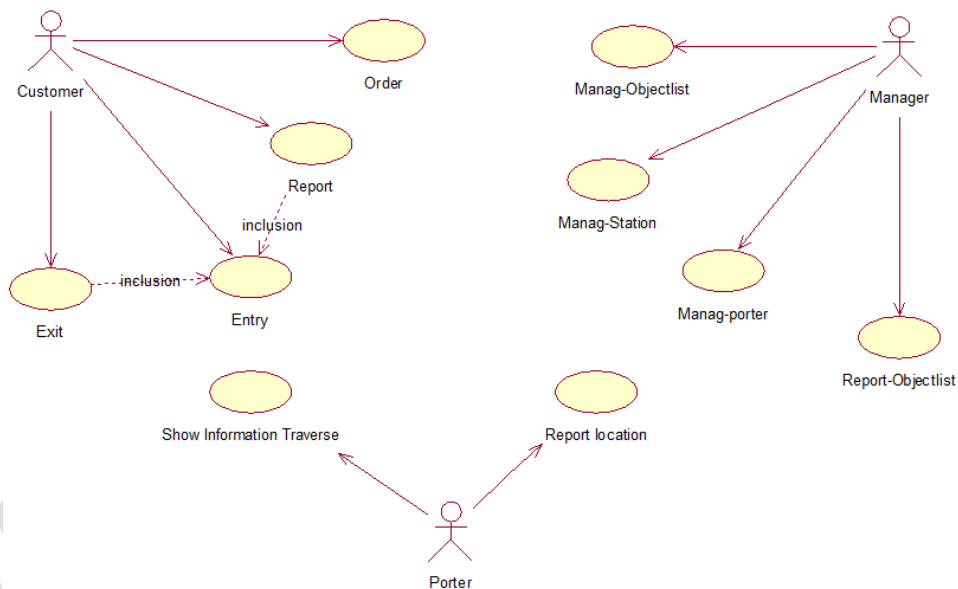
مورد کاربرد	مدیریت ایستگاه ها
هدف	در این مورد کاربرد مدیر می تواند با استفاده از امکانات سیستم کد ایستگاه ها را ویرایش کند .
عامل اصلی	مدیر
شرایط اولیه	مدیر گزینه مدیریت ایستگاه ها را انتخاب کند .
جریان اصلی	<ol style="list-style-type: none"> ۱- مدیر درخواست مدیریت ایستگاه ها از سیستم را داشته باشد . ۲- سیستم درخواست اطلاعات لازم از مدیر برای احراز هویت می کند . ۳- در صورت صحیح بودن اطلاعات مدیر وارد این قسمت خواهد شد . ۴- مدیر می تواند از امکانات سیستم در این قسمت استفاده کند و تغییرات لازم را اعمال کند . ۵- مدیر گزینه ثبت را انتخاب می کند و سیستم تغییرات لازم را ذخیره می کند . ۶- در صورت انتخاب گزینه خروج پیغام هشدار نمایش داده خواهد شد و مدیر از سیستم خارج می شود.
شرایط نهایی	تغییرات، در کد ایستگاه ها توسط مدیر در سیستم ثبت خواهد شد و لیست ویرایش شده کد ایستگاه ها نمایش داده خواهد شد.
جریان فرعی	<ol style="list-style-type: none"> ۱- در صورت اینکه اطلاعات لازم برای احراز هویت مدیر صحیح نباشد . ۲- پیغام هشدار نمایش داده می شود . ۳- سیستم به پنجره اصلی بر خواهد گشت .

جدول ۴-۱۰ سناریوی مورد کاربرد مدیریت راننده ها

مورد کاربرد	مدیریت راننده ها
هدف	در این مورد کاربرد مدیر می تواند با استفاده از امکانات سیستم مسیر راننده ها را ویرایش کند .
عامل اصلی	مدیر
شرایط اولیه	مدیر گزینه مدیریت راننده ها را انتخاب کند .
جریان اصلی	<ol style="list-style-type: none"> ۱. مدیر درخواست مدیریت راننده ها از سیستم را داشته باشد . ۲. سیستم درخواست اطلاعات لازم از مدیر برای احراز هویت می کند . ۳. در صورت صحیح بودن اطلاعات مدیر وارد این قسمت خواهد شد . ۴. مدیر می تواند از امکانات سیستم در این قسمت استفاده کند و تغییرات لازم را اعمال کند . ۵. مدیر گزینه ثبت را انتخاب می کند و سیستم تغییرات لازم را ذخیره می کند . ۶. در صورت انتخاب گزینه خروج پیغام هشدار نمایش داده خواهد شد و مدیر از سیستم خارج می شود.
شرایط نهایی	تغییرات، در مسیر راننده ها توسط مدیر در سیستم ثبت خواهد شد و لیست ویرایش شده مسیر راننده ها نمایش داده خواهد شد .
جریان فرعی	<ol style="list-style-type: none"> ۱. در صورت اینکه اطلاعات لازم برای احراز هویت مدیر صحیح نباشد . ۲. پیغام هشدار نمایش داده می شود . ۳. سیستم به پنجره اصلی بر خواهد گشت .

۴.۳.۳ استخراج مدل موارد کاربرد

مدل موارد کاربرد، مدلی است از سیستم و رفتارهای مطلوب آن از منظر بیرونی، یا در واقع، مدلی است که رفتارهای سیستم را از منظر عناصر و موجودیت های خارج از آن، توصیف می نماید. این مدل دربرگیرنده مجموعه ای از موارد کاربرد و سرویس می باشد که تعریف کننده چگونگی بکارگیری سیستم به وسیله اکتورها می باشد. اکتور عبارت است از شخص یا چیزی بیرون از سیستم و در تعامل با آن. یک مورد کاربرد، دیالوگی است میان اکتور و سیستم. از منظر این مدل، سیستم همانند یک جعبه سیاه دیده می شود. در این مدل تنها به بررسی چیستی و چرایی سرویسها و نیازمندیها می پردازیم. به عبارت دیگر، مدل کاربرد برای توصیف نیازمندیهای وظیفه مندی سیستم بکار می روند. البته برخی ملاحظات مرتبط با نیازمندی ها غیر وظیفه مندی در توصیف متنی موارد کاربرد بیان می شود. همچنین مدل رشنال رز متناظر با مدل زیر در دیدگاه یوزکیس، با نام یوزکیس^{۲۹} قرار دارد در صورت نیاز به آن مراجعه فرمایید. فایل رشنال رز پروژه ضمیمه پایان نامه شده است.



شکل ۴-۳ مدل موارد کاربرد سیستم نرم افزاری باربری

²⁹ Usecase

۴.۳.۴ ایجاد نمونه ای از واسط کاربر

در این قسمت یک نمونه ظاهری از واسطه ها که بین سیستم و کاربران ارتباط برقرار می کنند می سازیم. البته این نمونه ای است برای ذینفعان سازمان باربری و ممکن ظاهر واسطه ها در انتهای پروژه تغییر کنند.



شکل ۴-۴ پنجره نخست



شکل ۴-۵ ورود

سیستم باربری

سفارش حمل بار

نام:
 نام خانوادگی:
 نام کالا:
 استان:
 شهرستان:
 استان مقصد:
 شهرستان مقصد:
 زمان تحویل: ۲۰۱۰/۱۲/۱۲ ☐ ذخیره اطلاعات

بارنامه

کد باربری:
 نام و خانوادگی صاحب بار:
 مبدأ:
 مقصد:
 زمان حرکت از مقصد:
 موقعیت فعلی بار:
 زمان تمویل بار در مقصد (زمان مناسبه شده پیش فرض):
 هزینه پالیه‌ها:

گیت: چاپ بارنامه:

لطفأ برای سفارش حمل بار اطلاعات لازم را وارد کنید خروج:

شکل ۴-۶ سفارش حمل بار

سیستم باربری

گزارش موقعیت بار

کد باربری:
 ذخیره کد باربری: ☐

گزارش موقعیت بار:

سپتامبر ۲۰۱۰

الحد	الحد	الحد	الحد	الحد	الحد	الحد	الحد	الحد	الحد
۲۸	۲۹	۳۰	۳۱	۱	۲	۳	۴	۵	۶
۱۱	۱۲	۱۳	۱۴	۱۵	۱۶	۱۷	۱۸	۱۹	۲۰
۲۱	۲۲	۲۳	۲۴	۲۵	۲۶	۲۷	۲۸	۲۹	۳۰
۱	۲	۳	۴	۵	۶	۷	۸	۹	۱۰

Today: ۲۰۱۰/۱۲/۰۹

گزارش موقعیت بار

کد باربری:
 نام صاحب بار:
 مبدأ:
 مقصد:
 زمان حرکت از مقصد:
 موقعیت فعلی بار:
 زمان تمویل بار در مقصد:

چاپ:

لطفأ کد باربری را برای نمایش گزارش موقعیت بار وارد کنید خروج:

شکل ۴-۷ گرفتن گزارش از موقعیت بار

شکل ۴-۸ گرفتن اطلاعات مسیر و بارنامه

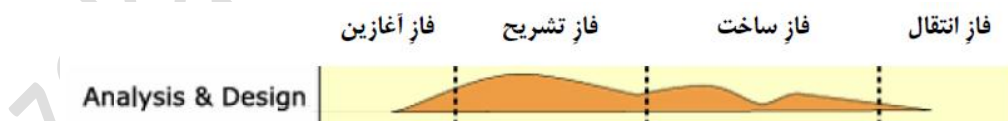
شکل ۴-۹ گرفتن گزارش از بارنامه ها

شکل ۴-۱۰ مدیریت بارنامه ها

شکل ۴-۱۱ مدیریت ایستگاه ها

۴.۴ دیسپلین تحلیل و طراحی

هدف اصلی دیسپلین تحلیل و طراحی، ترجمه نیازمندی ها به توصیف هایی است که چگونگی پیاده سازی سیستم را نشان می دهند.



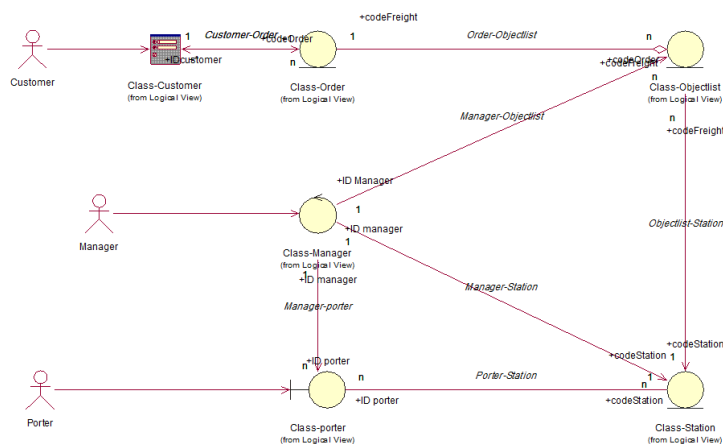
شکل ۴-۱۲ نمایی مقطعی از دیسپلین تحلیل و طراحی در طول فازهای چرخه تولید

در مدل های تحلیل و طراحی به بررسی چگونگی تحقق نیازمندا توجه داریم. برای نیل به این مقصود، ابتدا باید نیازمندا را به خوبی درک نموده و پس از آن با انتخاب بهترین استراتژی پیاده سازی، نیازمندی های را به طراحی سیستم ترجمه نماییم.

۴.۴.۱ مدل تحلیل

تحلیل به توصیف چگونگی تحقق وظیفه مندی ها با معرفی سیستم به صورت یک جعبه خاکستری از اجزا درونی آن، متمرکز می باشد. در واقع بدون وارد شدن به جزئیات خاص پیاده سازی و ملاحظات غیر وظیفه مندی، چگونگی تحقق موارد کاربرد سیستم بررسی می شود. با رفتن از نیازمندا به تحلیل، طراحی، و پس از آن، پیاده سازی، به تدریج به جزئیات مورد نیاز اضافه شده و میزان تجرید کمتر می شود. بنابراین مدل تحلیل مدلی عام از چگونگی رفتارهای سیستم ارائه می دهد، بدون آنکه مثلاً در آن به زبان جاوا، مبتنی بر وب بودن، یا یک بانک اطلاعاتی خاص، اشاره ای داشته باشیم. به عبارت دیگر، مدل تحلیل بیانگر تصویری ایده آل از سیستم می باشد. با توجه به ساده و کوچک بودن پروژه سیستم نرم افزاری باربری این پایان نامه، به مدل تحلیل سازمان که در دیسیپلین مدل سازی سازمان ارائه دادیم و دیاگرام توالی که در ادامه ارائه می دهیم بسنده می کنیم.

در مدل تحلیل، برای بیان نحوه تحقق موارد کاربرد تنها از سه نوع کلاس کلی استفاده می نماییم. این کلاس های به اصطلاح کلیشه ای، عبارتند از: کلاسهای مرزی، کلاسهای کنترلی، و کلاسهای موجودیت.



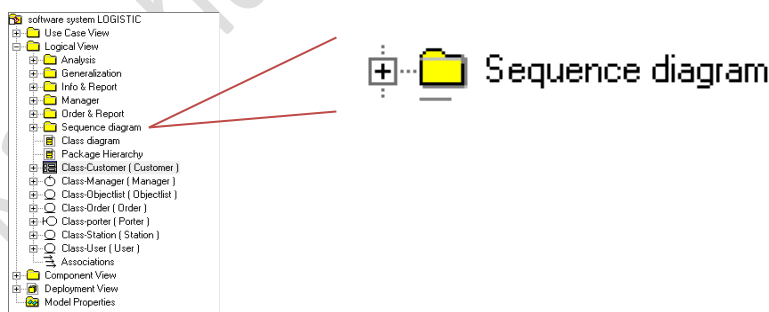
شکل ۴-۱۳ مدل تحلیل سازمان باربری

۴.۴.۲ دیاگرام های توالی در تحلیل سیستم نرم افزاری باربری

برای تحلیل موارد کاربرد سیستم نرم افزاری، از دیاگرام های تعاملی استفاده می کنیم. دیاگرام های تعاملی دو گونه اند، دیاگرام توالی و دیاگرام ارتباط. در این پروژه به علت ساختار ساده از دیاگرام ارتباط چشم پوشی کرده ایم، و تمام بار تحلیل این سیستم را بر عهده دیاگرام توالی انداخته ایم.

دیاگرام توالی؛ بررسی در زمان اجرا به گونه ای که مشخص شود برای انجام یک موردکاربر چه کلاس هایی با یکدیگر همکاری می کنند(کلاسها بصورت ارسال پیام با یکدیگر همکاری می کنند). در این بررسی مشخص می شود که آیا کلاس های تعریف شده موارد کاربرد را به طور کامل انجام می دهند یا خیر. در این دیاگرام اکتور ها همان اکتورهای مدل تحلیل سازمانی است و اشیاء آن همان کلاسهای مدل تحلیل سازمان که در قسمت قبل بیان کردیم می باشد.

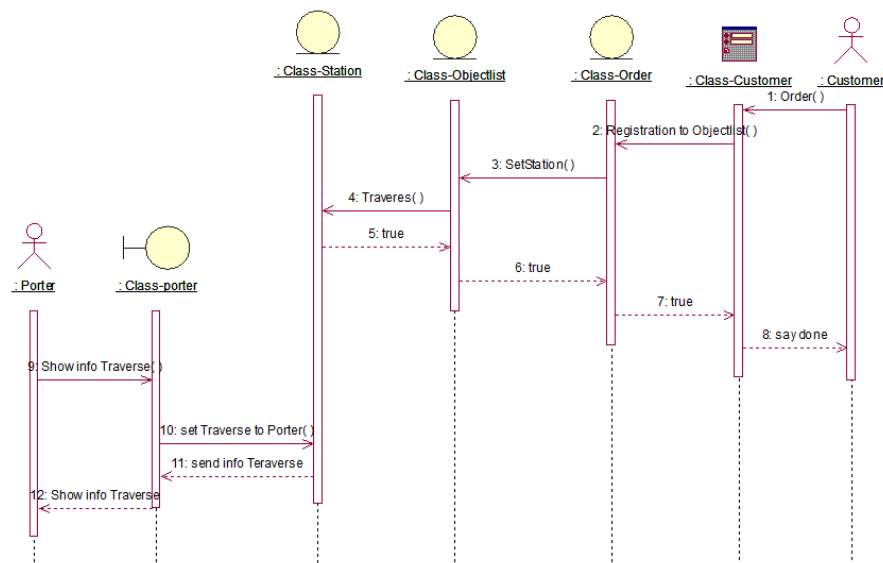
توجه: از آنجا که این پایان نامه تخصصی رشته کارشناسی فناوری اطلاعات است و اشخاصی که با فن مهندسی نرم افزار آشنا هستند در درک این مدل ها مشکلی ندارند ، و کما اینکه این مدل ها و دیاگرام ها خود بیان کننده کامل روند تولید نرم افزار به زبان یو.ام.ال هستند. با استناد به دلایل فوق الذکر از بیان جزئیات دیاگرام های توالی صرف نظر می کنیم. برای مشاهده جزئیات این دیاگرام به فایل رشنال رز پروژه، در قسمت دیدگاه منطقی^{۳۰} و پوشه با نام دیاگرام توالی^{۳۱} مراجعه فرمایید. این فایل ضمیمه پایان نامه شده است.



شکل ۴-۱۴ محل دیاگرام های توالی در فایل رشنال رز پروژه

^{۳۰} Logical view

^{۳۱} Sequence diagram



شکل ۴-۱۵ نمونه یک دیاگرام توالی (دیاگرام توالی شالوده تحلیل)

۴.۴.۳ مدل طراحی

در مقابل، هدف طراحی عبارت است از تطبیق نتایج تحلیل بر اساس محدودیتها و شرایط تحمیل شده توسط نیازمندیهای غیر وظیفه مندی و محیط پیاده سازی. به عبارت دیگر، مدل طراحی در مقایسه با مدل تحلیل، یک توصیف با جزئیات تفصیلی از پیاده سازی و شرایط خاص آن ارائه می دهد و بنابراین آن را می توانیم یک جعبه سفید از سیستم تلقی کنیم.

مدل طراحی، اصلی ترین دستاورد دیسیپلین تحلیل و طراحی می باشد. مدل طراحی شامل مجموعه ای از همکاریهای میان عناصر مدل می باشد که مجموعاً رفتارهای سیستم را توصیف می نمایند. این رفتارها از روی موارد کاربرد و نیازمندی های غیر وظیفه مندی استنتاج می شود. در مدل طراحی شامل توصیفی از همکاریهای میان کلاسها می باشد که ممکن است در قالب بسته ها و زیر سیستم ها در کنار هم تجمیع شوند. یک کلاس توصیف کننده مجموعه ای از اشیاء با مسئولیتها، روابط، عملیاتها، ویژگیها، و معنای مشابه می باشد. یک بسته، مکانیزمی است برای سازماندهی و کنترل پیچیدگی و نیز گروه بندی منطقی عناصر یک مدل، از جمله کلاسها. زیر سیستم بسته ای است شامل مجموعه ای از کلاس های فراهم

کننده رفتارهای خاص، البته لازم به ذکر است که سیستم نرم افزاری باربری ما به علت ساده بودن دارای زیر سیستم نمی باشد.

۴.۴.۴ بسته های مدل طراحی سیستم نرم افزاری باربری

این ساختار، از ۵ بسته تشکیل شده است. بسته همانطور که در قسمت قبل بیان شد مجموعه ایی از کلاسهای به هم وابسته از لحاظ منطقی است و بیان کننده یک رفتار خاص در سیستم نرم افزاری می باشد. بسته های سیستم باربری ما به گونه زیر است:

۱- بسته سفارش و گزارش: در این بسته، رفتار خدمات سیستم به مشتری مد نظر قرار گرفته است خدماتی همچون سفارش برای حمل بار و گزارش از موقعیت بار

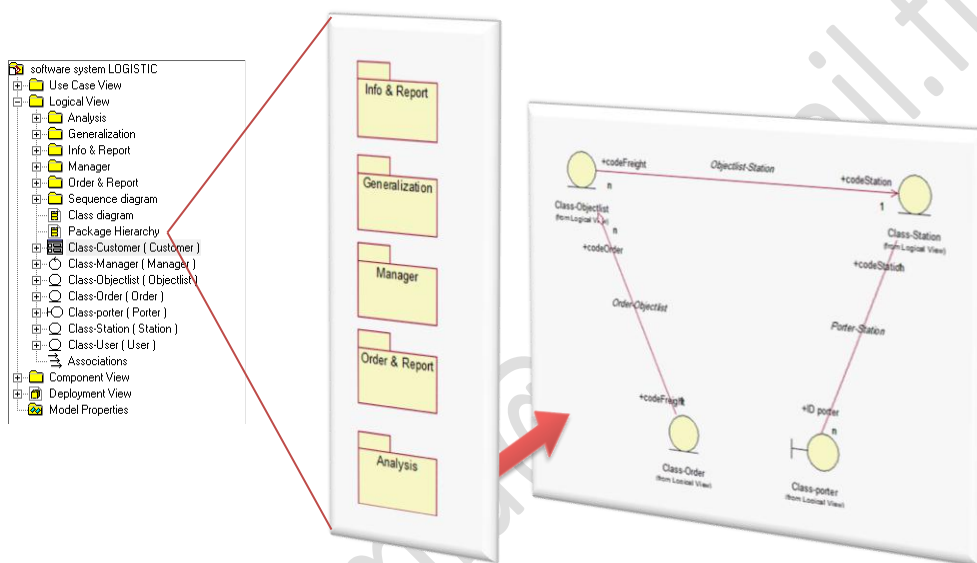
۲- بسته آنالیز: در این بسته، رفتار تراکنشی و تحلیلی سیستم در مقابل درخواست های کاربران مد نظر است. آنالیزهای جای دادن سفارشها در برنامه ها، تخصیص برنامه ها در ایستگاه ها، جای دادن و ترتیب دادن در ایستگاه ها با مسیر راننده ها و آنالیز دادهای راننده ها و تبدیل به اطلاعات مفید برای مشتری

۳- بسته اطلاعات و گزارش: در این بسته، رفتار سیستم با راننده ها مد نظر است. اطلاعات؛ راننده ها نیاز به یکسری اطلاعاتی از مسیر حرکت و لیست برنامه ها دارند، که این اطلاعات بعد از آنالیز در سیستم در اختیار راننده ها قرار می گیرد. گزارش؛ از طرفی راننده ها به هر ایستگاه که خواهند رسید می بایست کد ایستگاه مستقر را برای سیستم باربری ارسال نمایند، در ادامه سیستم بعد از آنالیز این داده ها آنها را تبدیل به اطلاعات مفید برای مشتریان می کند. مشتری از موقعیت بار، زمان سپری شده از شروع جابجایی، زمان باقی مانده تا رسیدن به مقصد و موارد دیگر اطلاع می یابد

۴- بسته مدیریت: در این بسته، رفتار سیستم در حوزه اختیارات مدیر سازمان می باشد. برای اینکه مدیر بتواند محدودیت ها و ویرایش های خود را در سیستم نرم افزاری باربری اعمال کند این بسته را در نظر گرفته ایم. در مجموع می توان گفت؛ بازتاب رفتار مدیر سازمان باربری در سیستم نرم افزاری باربری مد نظر است. که جزئیات این رفتار در فایل رشنال رز این پروژه نهاده شده است

۵- بسته ارث بری: همانطور که قبلاً گفته شد یک بسته می تواند نمایانگر ارتباطی منطقی بین اجزای آن باشد. در این بسته بین کلاسها و کلاسی که دارای خصوصیات عمومی است، یک ارتباط منطقی ارث بری وجود دارد

مدل رشنال رز متناظر با مدل زیر در دیدگاه منطقی، با نام سلسله بسته ها^{۳۲} قرار دارد، در صورت نیاز به آن مراجعه فرمایید. فایل رشنال رز پروژه ضمیمه پایان نامه شده است



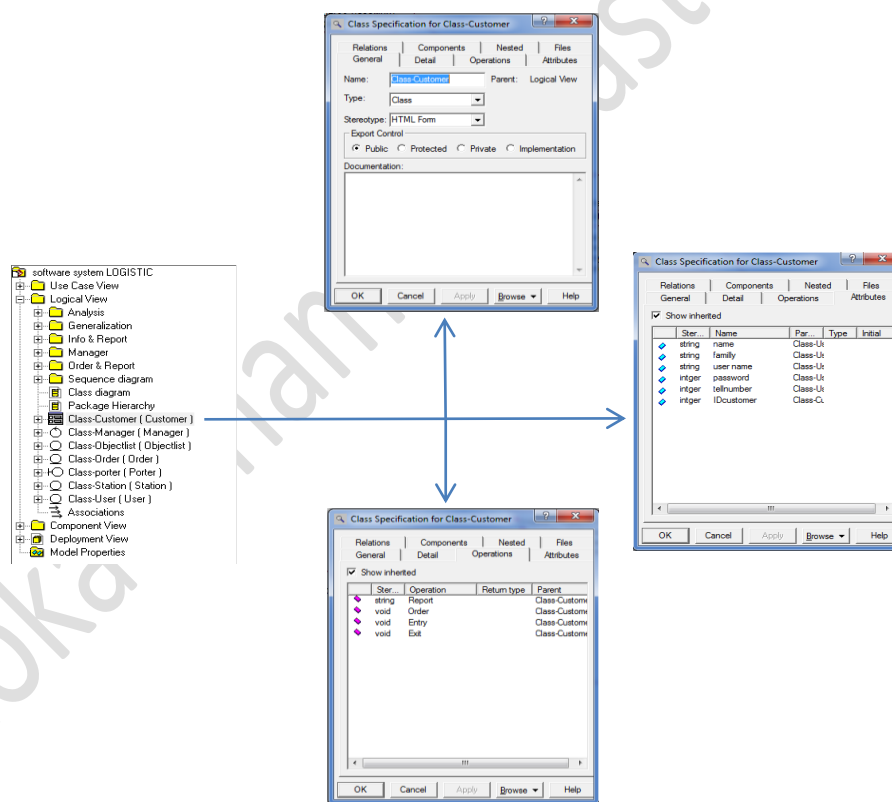
شکل ۴-۱۶ نمونه ای از ساختار بسته ها در مدل طراحی سیستم نرم افزاری باربری

۴.۴.۵ کلاس های مدل طراحی سیستم نرم افزاری باربری

در این قسمت رفتارهای سیستم را، در ۷ کلاس مختلف پیاده سازی کرده ایم، همانطور که قبلاً اشاره شد هر کلاس مجموعه ای از اشیاء با مشخصه های مشترک است و همچنین انواع کلی کلاسها را بیان کردیم. حال به شرح نوع و خصوصیات کلاسهای سیستم نرم افزاری باربری طراحی شده می پردازیم.

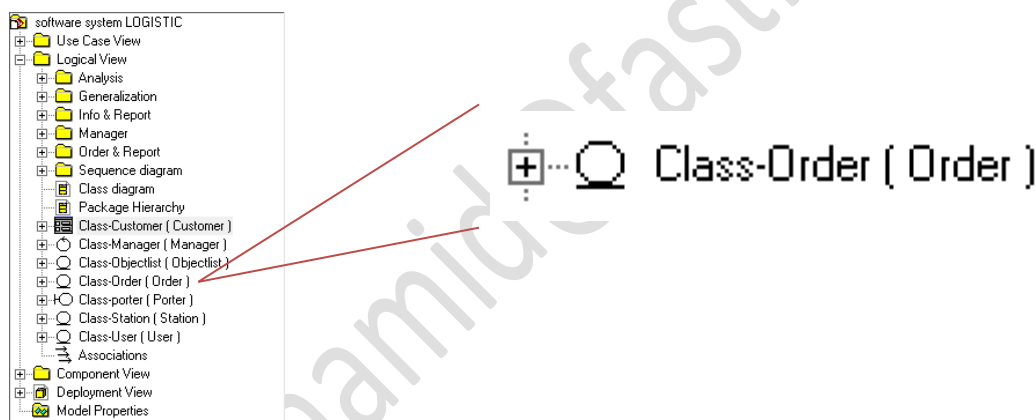
³² Package Hierarchy

۱- کلاس مشتری: این کلاس برای ارتباط با مشتری طراحی شده است. چون مشتری یک اکتور و خارج از سیستم قرار دارد، کلاس برای ارتباط با مشتری در مرز سیستم قرار می گیرد، در نتیجه این کلاس را از نوع مرزی در نظر می گیریم و به صورت یک فرم وب برای مشتری نمایش می دهیم. صفات عمومی این کلاس از کلاس کاربر به ارث برده شده است و صفات خصوصی این کلاس هم، متناسب با اطلاعات لازم برای یک مشتری است. عملیات کلاس مشتری هم مطابق با نیازهایی که در دیسپلین مدیریت نیازمندی ها بیان شد طراحی شده است. برای مشاهده جزئیات این کلاس به فایل رشنال رز پروژه، در قسمت دیدگاه منطقی و کلاس با نام مشتری مراجعه فرمایید. این فایل ضمیمه پایان نامه شده است



شکل ۴-۱۷ کلاس مشتری و خصوصیات آن در فایل رشنال رز پروژه

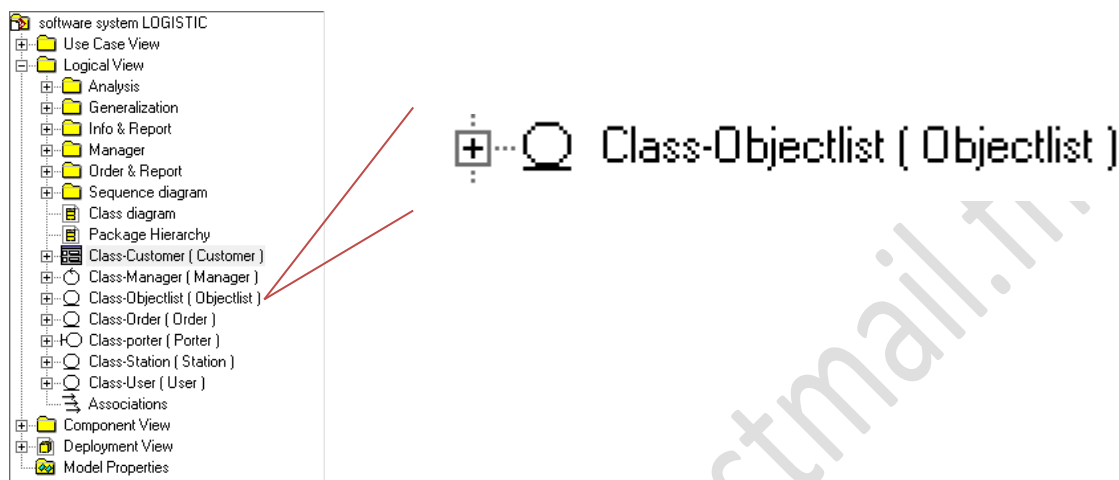
۲- کلاس سفارش: این کلاس برای آنالیز اطلاعات سفارشها، تثبیت سفارشها در بارنامه ها و، ذخیره و بازیابی اطلاعات سفارشها در پایگاه داده سیستم طراحی شده است. از آنجا که این کلاس ارتباط مستقیم با پایگاه داده سیستم دارد به عنوان کلاس موجودیت در نظر گرفته می شود. صفات این کلاس همان اطلاعات لازم از سفارش حمل کالا است، که از مشتری درخواست می شود و عملیات این کلاس، تثبیت اطلاعات سفارش در بارنامه ها می باشد. برای مشاهده جزئیات این کلاس به فایل رشنال رز پروژه، در قسمت دیدگاه منطقی و کلاس با نام سفارش مراجعه فرمایید. این فایل ضمیمه پایان نامه شده است



شکل ۴-۱۸ کلاس مشتری و خصوصیات آن در فایل رشنال رز پروژه

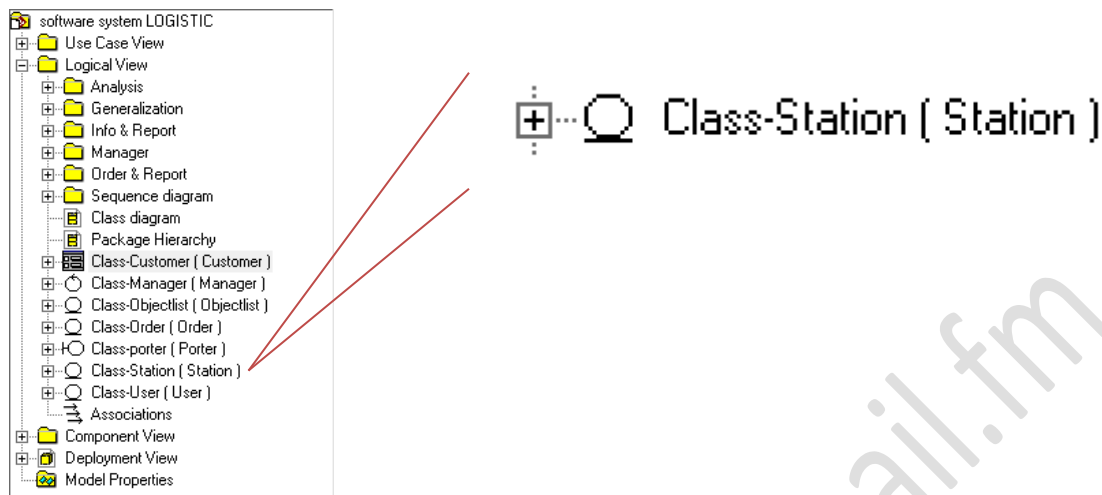
۳- کلاس بارنامه (لیست اشیاء): این کلاس برای آنالیز اطلاعات بارنامه ها، تثبیت بارنامه ها در ایستگاه ها و، ذخیره و بازیابی اطلاعات بارنامه ها در پایگاه داده سیستم طراحی شده است. از آنجا که این کلاس ارتباط مستقیم با پایگاه داده سیستم دارد به عنوان کلاس موجودیت در نظر گرفته می شود. صفات این کلاس همان اطلاعات یک بارنامه واقعی در سازمان است که بعد از آنالیز سیستم در کلاسهای سفارش، این کلاس مطلع خواهد شد. عملیات کلاس بارنامه، تثبیت اطلاعات بارنامه ها در ایستگاه ها می باشد. برای مشاهده جزئیات این کلاس به فایل رشنال رز

پروژه، در قسمت دیدگاه منطقی و کلاس با نام بارنامه (لیست اشیاء) مراجعه فرمایید. این فایل ضمیمه پایان نامه شده است



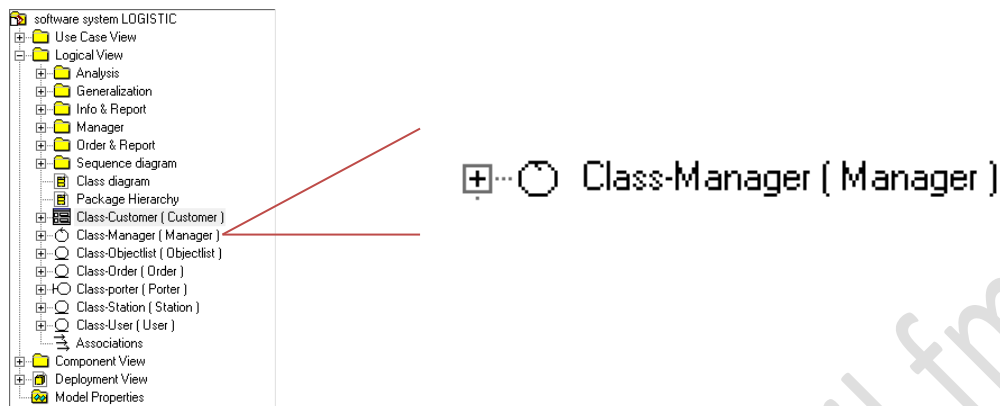
شکل ۴-۱۹ محل کلاس بارنامه در فایل رشنال رز پروژه

۴- کلاس ایستگاه: این کلاس برای آنالیز اطلاعات ایستگاه ها، تثبیت ایستگاه ها در مسیر راننده ها و، ذخیره و بازیابی اطلاعات ایستگاه ها در پایگاه داده سیستم، طراحی شده است. از آنجا که این کلاس ارتباط مستقیم با پایگاه داده سیستم دارد به عنوان کلاس موجودیت در نظر گرفته می شود. صفات این کلاس همان اطلاعاتی است برای شناسایی ایستگاه ها در سازمان که توسط مدیر سازمان به آنها داده می شود. عملیات کلاس ایستگاه، تثبیت اطلاعات ایستگاه ها در مسیر راننده ها می باشد. برای مشاهده جزئیات این کلاس به فایل رشنال رز پروژه، در قسمت دیدگاه منطقی و کلاس با نام ایستگاه مراجعه فرمایید. این فایل ضمیمه پایان نامه شده است



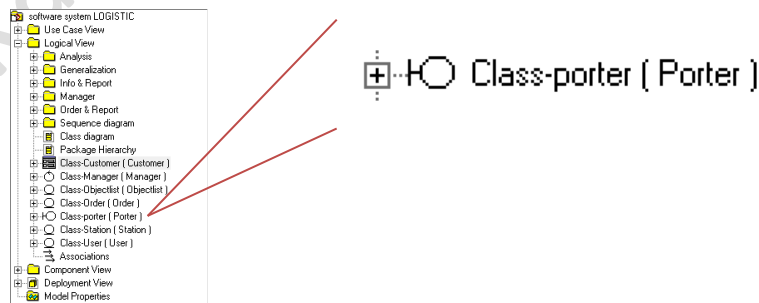
شکل ۴-۲۰ محل کلاس بارنامه در فایل رشنال رز پروژه

۵- کلاس مدیر: این کلاس برای مدیریت، مدیر سازمان بر کلاسهای بارنامه، ایستگاه و راننده می باشد و همچنین گرفتن گزارش از آنها طراحی شده است. اطلاعات این کلاس ها در پایگاه داده سیستم ذخیره می شود و مدیر برای اعمال هر نوع تغییری در آنها می بایست به پایگاه داده سیستم دسترسی داشته باشد. از آنجا که این کلاس نقش مدیریت بر کلاسها دارد، کلاس مدیر را از نوع کنترلی در نظر می گیریم. صفات عمومی کلاس مدیر از کلاس کاربر به ارث برده شده است و صفات خصوصی این کلاس، همان اطلاعات سازمانی مدیر سازمان باربری است. عملیات کلاس مدیر، رفتار و نیازهای پیاده سازی شده مدیر سازمان در سیستم می باشد. برای مشاهده جزئیات این کلاس به فایل رشنال رز پروژه، در قسمت دیدگاه منطقی و کلاس با نام مدیر مراجعه فرمایید. این فایل ضمیمه پایان نامه شده است



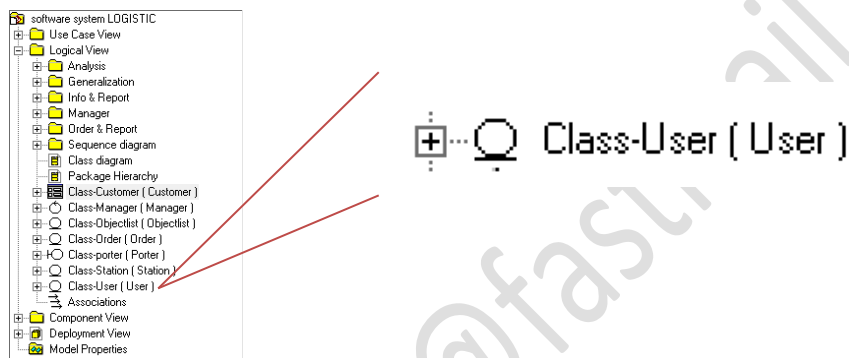
شکل ۴-۲۱ محل کلاس بارنامه در فایل رشنال رز پروژه

۶- کلاس راننده: این کلاس برای ارتباط با راننده طراحی شده است. چون راننده یک اکتور و خارج از سیستم قرار دارد، کلاس برای ارتباط با راننده در مرز سیستم قرار می گیرد، در نتیجه این کلاس را از نوع مرزی در نظر می گیریم و توسط مودم با راننده ارتباط برقرار می کند. صفات عمومی این کلاس از کلاس کاربر به ارث برده شده است و صفات خصوصی این کلاس هم، همان اطلاعات یک راننده در سازمان باربری می باشد. عملیات کلاس راننده هم مطابق با نیازهایی که در دیسپلین مدیریت نیازمندیها بیان شد، طراحی شده است. برای مشاهده جزئیات این کلاس به فایل رشنال رز پروژه، در قسمت دیدگاه منطقی و کلاس با نام راننده مراجعه فرمایید. این فایل ضمیمه پایان نامه شده است.



شکل ۴-۲۲ محل کلاس بارنامه در فایل رشنال رز پروژه

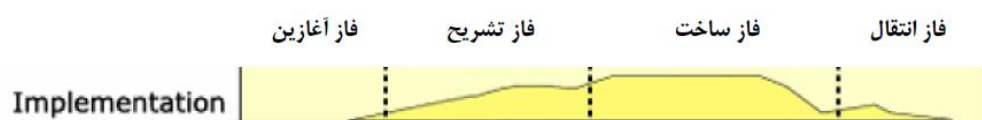
۷- کلاس کاربر: این کلاس اطلاعات مورد نیاز تشخیص هویت کاربران را در پایگاه داده سیستم ذخیره می کند، به همین دلیل این کلاس را از نوع موجودیت تعریف می کنیم. صفات کلاس کاربر، همان اطلاعات شخصی کاربران در سازمان است. برای مشاهده جزئیات این کلاس به فایل رشنال رز پروژه، در قسمت دیدگاه منطقی و کلاس با نام کاربر مراجعه فرمایید. این فایل ضمیمه پایان نامه شده است.



شکل ۴-۲۳ محل کلاس کاربر در فایل رشنال رز پروژه

۴.۵ دیسیپلین پیاده سازی

در این فصل به دیسیپلین پیاده سازی اجزا در سیستم نرم افزاری باربری می پردازیم. در شکل زیر نمایی از حجم فعالیتهای این دیسیپلین در طول چرخه تولید نشان داده شده است. همانگونه که ملاحظه می نمایید، به طور معمول، حجم فعالیتهای این دیسیپلین در فاز ساخت بیشتر از سایر فازها است.



شکل ۴-۲۴ نمایی مقطعی از دیسیپلین پیاده سازی در طول چرخه تولید

به تفاوت معنایی مفاهیم پیاده سازی و ساخت در آ.یو.پی توجه داشته باشید. بخش عمده فعالیت های دیسیپلین پیاده سازی به نوعی به برنامه نویسی مرتبط می باشد. اما، در فاز ساخت برای دستیابی به اهداف تعیین شده، فعالیتها یا دیسیپلینهای دیگری (علاوه بر پیاده سازی) نیز دخیل می باشند. بنابراین، پیاده سازی یکی از اقدامات مهم و عمده فاز ساخت است، نه همه آن. توجه داشته باشید که دیسیپلین پیاده سازی ارتباط نزدیکی با دیسیپلین تست دارد. که البته دیسیپلین تست و دیسیپلین استقرار را به علت عملی و تجربی بودن، از بیانشان در این پایان نامه صرف نظر کردیم.

چهار هدف اصلی دیسیپلین پیاده سازی، عبارتند از:

- ۱- پیاده سازی کلاسها و اشیاء در قالب یکسری مؤلفه (کامپونتها)
- ۲- تعریف چگونگی سازماندهی و ساختار کدهای برنامه در قالب زیر سیستم ها
- ۳- تست پیاده سازی شده به صورت واحد
- ۴- یکپارچه سازی و مجتمع سازی مؤلفه ها در قالب یک سیستم قابل اجرا

در دیسیپلین پیاده سازی، مفهوم تست، محدود به تست واحد هر یک از مؤلفه ها به صورت جداگانه و مستقل می باشد. تست هایی مانند تست سیستم و تست یکپارچگی در دیسیپلین تست قرار دارند. البته به خاطر داشته باشید که در عمل، جریان های کار به هم آمیخته اند و در اجرا مرز کاملاً مشخصی میان آنها وجود نخواهد داشت.

۴.۵.۱ پیاده سازی کلاسها و اشیاء در قالب یکسری مؤلفه (کامپونتها)

ما این مؤلفه ها را کامپونت در نظر می گیریم. کامپونتها از کدهایی تشکیل می شوند که هر کدام، سرویس خاصی را انجام می دهند و این سرویس ها از طریق رابط ها ارائه می شوند. همین امر، امنیت و یوزر فرندی سیستم نرم افزاری را افزایش می دهد. چرا؟ چون؛ از دید امنیتی کاربران بطور مستقیم به کدها دسترسی ندارند و این رابط ها نقش مانع امنیتی را ایفا می کنند، و اما از دید یوزر فرندی (یا کاربر دوستانه)، رابط ها به صورت یک فرم طراحی و پیاده سازی می شوند و برای کاربران کدهای برنامه نویسی شده که جلوه تخصصی یا به تعبیری جلوه زشت کار است به نمایش گذاشته نمی شود.

۴.۵.۲ کامپونتهای سیستم نرم افزاری باربری

۱- کامپونت مشتری: این کامپونت در ارتباط با کلاس مشتری است. زبان برنامه نویسی این کامپونت زبان برنامه نویسی تحت وب است. نوع این کامپونت اکتیوایکس، و اکتیوایکس یک نوع برنامه اجرایی فرم سفارش در سیستم کلاینت (مشتری) است، توسط مرورگروپ دانلود، و سپس در مرورگروپ اجرا می شود (می توان مرورگر را طوری تنظیم کرد که آیا بصورت خودکار اکتیوایکسها را دانلود کند یا نه، قبل از دانلود از کاربر اجازه بگیرد). این فرم ها قابلیت آنالیز اطلاعات فرم سفارش برای قیمت و زمان جابجایی کالا را دارند. این آنالیزها در سیستم کلاینت (مشتری) انجام می شود و نیازی به رفت و برگشت فرم وب به سرور سازمان نمی باشد و همچنین می توان فرمها را برای هر مشتری متناسب با اطلاعات کوکی ها سفارشی کنیم. برای مشاهده این کامپونت به فایل رشنال رز پروژه، در دیدگاه کامپونت^{۳۳} (یا اجزا) و کامپونت با نام مشتری مراجعه فرمایید. فایل پروژه ضمیمه پایان نامه شده است

۲- کامپونت سفارش: این کامپونت در ارتباط با کلاس سفارش است. زبان برنامه نویسی این کامپونت زبان برنامه نویسی ویژال سی پلاس پلاس^{۳۴} است. نوع این کامپونت، یک زیر روال با ارائه سرویس خدمات سازمان به مشتری و گزارش گرفتن است. برای مشاهده این کامپونت به فایل رشنال رز پروژه، در دیدگاه کامپونت (یا اجزا) و کامپونت با نام سفارش مراجعه فرمایید. فایل پروژه ضمیمه پایان نامه شده است

۳- کامپونت بارنامه (لیست اشیاء): این کامپونت در ارتباط با کلاس بارنامه است. زبان برنامه نویسی این کامپونت زبان برنامه نویسی ویژال سی پلاس پلاس است. نوع این کامپونت، یک زیر روال با ارائه سرویس آنالیز، ذخیره و بازیابی اطلاعات در پایگاه داده است. برای مشاهده این کامپونت به فایل رشنال رز پروژه، در دیدگاه کامپونت (یا اجزا) و کامپونت با نام سفارش مراجعه فرمایید. فایل پروژه ضمیمه پایان نامه شده است

³³ Component view

³⁴ VC++

۴- کامپونت ایستگاه: این کامپونت در ارتباط با کلاس ایستگاه است. زبان برنامه نویسی این کامپونت زبان برنامه نویسی ویژال سی پلاس پلاس است. نوع این کامپونت، یک برنامه اصلی (ریشه) با ارائه سرویس آنالیز، ذخیره اطلاعات در پایگاه داده و تثبیت اطلاعات در کلاس های دیگر است. برای مشاهده این کامپونت به فایل رشنال رز پروژه، در دیدگاه کامپونت (یا اجزا) و کامپونت با نام سفارش مراجعه فرمایید. فایل پروژه ضمیمه پایان نامه شده است

۵- کامپونت راننده: این کامپونت در ارتباط با کلاس راننده است. زبان برنامه نویسی این کامپونت زبان برنامه نویسی ویژال سی پلاس پلاس است. نوع این کامپونت، یک اپلیکیشن^{۳۵} (نرم افزار کوچک) با ارائه سرویس ارسال و دریافت اطلاعات به سیستم از طرق مختلف است. این اپلیکیشن باید در سیستم ارتباطی (موبایل یا لپ تاپ و یا هر وسیله رایانه ای متصل به شبکه وب جهانی) راننده نصب شود. با این اپلیکیشن راننده ها تشخیص هویت می شوند؛ برای گزارش دادن از موقعیت بار و یا گرفتن اطلاعات مسیر حرکت. برای مشاهده این کامپونت به فایل رشنال رز پروژه، در دیدگاه کامپونت (یا اجزا) و کامپونت با نام راننده مراجعه فرمایید. فایل پروژه ضمیمه پایان نامه شده است

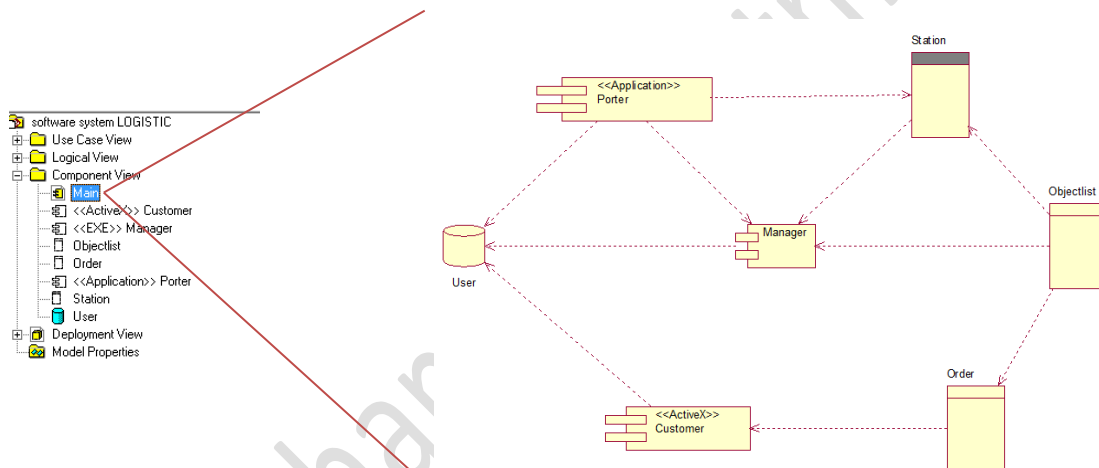
۶- کامپونت مدیر: این کامپونت در ارتباط با کلاس مدیر است. زبان برنامه نویسی این کامپونت زبان برنامه نویسی ویژال سی پلاس پلاس است. نوع این کامپونت، یک فایل قابل اجرا برای ارتباط مدیر با سیستم نرم افزاری است. توسط این فایل مدیر تشخیص هویت می شود، بعد از صحت اطلاعات وارد شده مدیر می تواند کل اطلاعات پایگاه داده سیستم را ویرایش کند. در ضمن این فایل قابل حمل بوده و مدیر در هر کجا می تواند از آن استفاده کند، البته بعد از صحت تشخیص هویت. برای مشاهده این کامپونت به فایل رشنال رز پروژه، در دیدگاه کامپونت (یا اجزا) و کامپونت با نام مدیر مراجعه فرمایید. فایل پروژه ضمیمه پایان نامه شده است

۷- کامپونت کاربر: این کامپونت در ارتباط با کلاس کاربر است. زبان برنامه نویسی آن همان ویژال سی پلاس پلاس است. نوع این کامپونت، پایگاه داده ای است و سرویس ذخیره اطلاعات کاربران در پایگاه داده سیستم، ارائه می دهد. برای مشاهده این کامپونت به فایل رشنال رز پروژه،

³⁵ Application

در دیدگاه کامپونت (یا اجزا) و کامپونت با نام کاربر مراجعه فرمایید. فایل پروژه ضمیمه پایان نامه شده است

توجه: زبان برنامه نویسی این پروژه سی شارپ و سی شارپ دات نت^{۳۶} در محیط ویژال استدیو^{۳۷} است، همچنین پایگاه داده این پروژه اسکینال^{۳۸} و ارتباط با پایگاه داده توسط دیتا آداپتر آڈ دات نت^{۳۹} می باشد. نوع زبانی که در کامپونت ها بیان شد نمادین است.



شکل ۴-۲۵ محل مدل شالوده کامپونت ها در فایل رشنال رز پروژه

³⁶ C#.net

³⁷ Visual Studio

³⁸ SQL

³⁹ Data Adapter ADO.net

۴.۵.۳ تعریف چگونگی سازماندهی و ساختار کدهای برنامه در قالب زیر سیستم ها

۴.۵.۳.۱ کلاس کاربر

User
<pre># string name ; # string family ; # string username ; # intger password ; # intger tellnumber;</pre>

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
namespace system_barbari
{
    class user
    {
        protected string name;
        protected string family;
        protected string username;
        protected int password;
        protected int tellnumber;
    }
}
```

۴.۵.۳.۲ کلاس مشتری

Customer
<pre>intger Idcustomer ;+</pre>
<pre>void RequestOrder();+ void ShowinfoOrder();+ + void getinfoCustomer();</pre>

```
using System;
using System.Collections.Generic;
using System.Linq;
```

```
using System.Text;
namespace system_barbari
{
    class customer:user
    {
        public int IDcustomer=0;
        public string customerID;
        public void getinfoCustomer()
        {
            IDcustomer++;
            customerID = "customer" + IDcustomer.ToString();
            customerID = new customer();
            customerID.name = txtname.text;
            customerID.family = txtfamily.text;
            customerID.username = txtUname.text;
            customerID.password = txtpass.text;
            customerID.tellnumber = txttellnumber.text;
        }
        public void RequestOrder()
        {
            string orderID;
            orderID = "order"+customerID.ToString();
            Order orderID = new Order();
            orderID.codeorder++;
            orderID.nameobject = textbox1.text;
            orderID.typeobject = textbox2.text;
            orderID.weaighobject = textbox3.text;
            orderID.destination = textbox4.text;
            orderID.source = textbox5.text;
            orderID.date = textbox6.text;
        }
        public void ShowinfoOrder()
        {
            label1.text=orderID.codeorder.ToString();
            label2.text=orderID.nameobject.ToString();
            label3.text=orderID.typeobject.ToString();
            label4.text=orderID.weaighobject.ToString();
            label5.text=orderID.destination.ToString();
            label6.text=orderID.source.ToString();
            label7.text=orderID.date.ToString();    }
    }
}
```

Order
+ string nameobject ; + string weighobject ; + string typeobject; + intger date; + intger codeorder; + string destination; + string source ; + void RegistrationtoObjectlist(string);

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
namespace system_barbari
{
    class Order
    {
        public string nameobject;
        public string weaighobject;
        public string typeobject;
        public int date;
        public int codeorder=0;
        public string destination;
        public string source;
        public int i=1;

        public void RegistrationtoObject(string orderID)
        {
            while (Objectlistdate&destintion.Read())
            if(Convert.ToString(Objectlistdate&destintion.["destination"]==orderID.destination.Tostring()){
                i++;
                Objectlistdate&destination.locationDestination[i] = orderID.destination.Tostring();
                Objectlistdate&destination.ListobjectOeder[i] = orderID.codeorder.Tostring();
            }

            string Objectlistdate&destination = "objectlist" + date.ToString()+destination.Tostring();
            Objectlist Objectlistdate&destination = new Objectlist();
            Objectlistdate&destination.locationDestination[i] = orderID.destination.Tostring();
            Objectlistdate&destination.ListobjectOeder[i] = orderID.codeorder.Tostring();

        }
    }
}
    
```

۴.۵.۳.۴ کلاس بارنامه (لیست اشیاء)

Objectlist
+ integer codeFreight; + locationFreight; + locationDestination; + ListobjectOeder; + void setStation(class);

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
namespace system_barbari
{
    class Objectlist
    {
        public int codeFreight;
        public string locationFreight;
        public string[] locationDestination;
        public string[] ListobjectOrder;
        public void setStation(class Objectlistdate&destination)
        {
            station station = new station();
            while (station.Read())
            if(Convert.ToString(station.["location"])==ToString(Objectlistdate&destination.locationFreight)){
                station.StationObjectlist = ToString(Objectlistdate&destination.codefreight);
            }
        }
    }
}
```

۴.۵.۳.۵ کلاس ایستگاه

Station
+ integer codeStation; + string StationObjectlist; + string PorterToStation; + string location; + void Traverse()

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
namespace system_barbari
{
class station
{
public int codeStation;
public string[] StationObjectlist;
public string PorterToStation;
public string location;
public int count = 0;
public void Traverse()
{
int IDporter;
string porter;
IDporter=getIDporterEmpty();
porter = "porter" + IDporter.ToString();
porterID = new porter();
count++;
porterID.TraverseStation[count]=codeStation.ToString();
}}}
```

۴.۵.۳.۶ کلاس راننده

Porter
+ intger IDporter;
+ string TraverseStation;
+ void ReportLocation(class);
+ string showinfoTraverse();

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
namespace system_barbari
{
class porter:user
{
public int IDporter;
public string[] TraverseStation;

public void showinfoTraverse(){
for(int i=1;i<=count;i++){
Map(TraverseStation[i]);
}
public void Reportlocation(class SMS)
```



```
{
    while(station.Read())
        if(station.location == SMS.text)
            station.PorterToStation=SMS.text;
    } } }
```

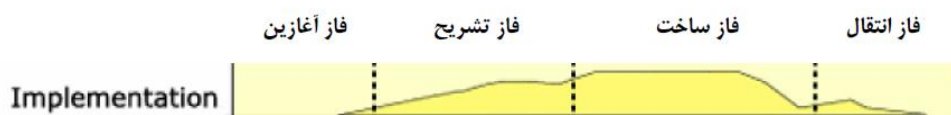
۴.۵.۳.۷ کلاس مدیر

Manager
+ integer IDmanager;
+ integer date;
+ void ReportListObject(integer);
+ void EditListObject(integer);
+ void EditCodeStation(integer);
+ void EditTraverseStation(integer);

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
namespace system_barbari
{
    class Manager:user{
        public int IDmanager;
        public int date;
        int i = 1;
        public void ReportListObject(int date) {
            while (OrderID.date == date)
                show(OrderID.codeorder); }
        public void EditListObject(int codeFreight)
        {
            for (; ; )
                if (Objectlistdate & destination.codeFreight == codeFreight)
                {
                    locationFreight = textbox1.Text;
                    for (; ; )
                        locationDestination[i++] = textbox2; } }
        public void EditCodeStation(int codeStation)
        {
            for(;;)
                if(codeStation == station.codeStation)
                    station.codeStation = textbox.Text;
        }
        public void EditTraverseStation(int TraverseStation)
        {
            for (; ; )
                if (porter.TraverseStation == TraverseStation)
                    porter.TraverseStation = textbox.Text; } } }
```

۴.۶ دیسپلین محیط

دیسپلین محیط به عنوان یکی از سه دیسپلین پشتیبان در فرایند آر.یو.پی است. شکل زیر بیانگر نمایی مقطعی از این دیسپلین و حجم فعالیت های آن در طول فازهای چهارگانه چرخه تولید می باشد.



شکل ۴-۲۶ دیسپلین محیط و حجم فعالیت های مرتبط با آن در طی فازهای مختلف فرایند تولید

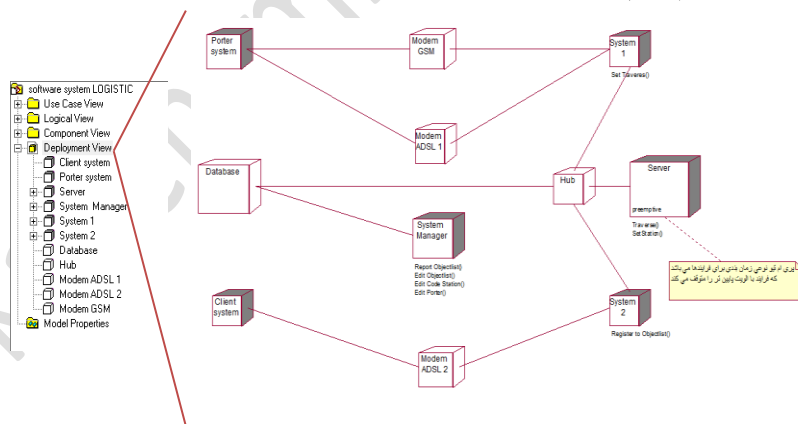
هدفِ دیسیپلین محیط، عبارت است از پشتیبانی سازمان تولید کننده فرآورده، هم از جنبه فرایند مناسب برای تولید و هم از نظر ابزارهای لازم. این پشتیبانی ها عبارتند از:

- ۱- انتخاب و فراهم آوری ابزار ابزارهای مناسب و ضروری
- ۲- نصب و پیکربندی ابزارهای مورد نیاز
- ۳- پیکربندی فرایند تولید
- ۴- بهبود فرایند
- ۵- ارائه خدمات فنی برای پشتیبانی از فرایند: فراهم نمودن زیر ساخت فناوری اطلاعات، راهبری ابزار و حساب های شخصی افراد مختلف تیم، تهیه پشتیبان، و موارد مشابه آن.

۴.۶.۱ مهندس فرایند

نقش کلیدی در این دیسیپلین، مهندس فرایند می باشد. مهندس فرایند، مسئول پیکر بندی اولیه و بهبود مستمر فرایند در طول چرخه تولید و نیز در سازمان می باشد. یکی از مهم ترین دستاوردهای مهم این مهندس مدل محیط سازمان می باشد.

۴.۶.۲ مدل محیط سیستم نرم افزاری باربری



شکل ۴-۲۷ مدل محیط سیستم نرم افزاری باربری

در این مدل همانطور که مشاهده می کنید، سرور به عنوان یک پردازنده و آنالیزگر مرکزی قرار گرفته است. در واقع تمام آنالیزهای مربوط به خدمات در این سیستم انجام می شود. نام فرایندها اصلی آن در زیر پردازنده نوشته شده است و نوع زمان بندی فرایندها هم preemptive (یعنی پردازنده

فرایندهای با الویت پایین تر را متوقف می کند) می باشد و تبادل اطلاعات با سرور از طریق یک هاب انجام می شود.

سیستم مدیر برای انجام عملیات ویرایشی اطلاعات به سیستم پایگاه داده سازمان متصل شده است. سیستم شماره ۲ توسط مودم شماره ۲ با مشتری در شبکه اینترنت ارتباط برقرار می کند و به عنوان یک واسطه بین سرور و سیستم مشتری برای ارائه خدمات سازمان قرار گرفته است. سیستم شماره ۱ از دو طریق با رانندها برای اطلاع رسانی آنالیزهای سرور ارتباط برقرار می کند. یک راه، مودم GSM است که با سیم کارت تلفن های همراه توسط پیام کوتاه با راننده ها تبادل اطلاعات می شود. راه دیگر، مودم ۱ (به نظر من ADSL تنها راه مقرون به صرفه و با سرعت بالا در کشور ما است) است که توسط اینترنت با راننده ها تبادل اطلاعات می شود.

۵. فصل چهارم نتیجه گیری

۵.۱ نتیجه گیری

در نهایت پس از انجام دیسپلین های فرایند آر.یو.پی، در مورد سیستم نرم افزاری باربری، به نتایج زیر براساس اهداف هر دیسپلین باید رسیده باشیم:

در دیسپلین مدل سازی از سازمان باربری:

- ۱- شناخت کامل از ساختار و پویایی سازمان پیدا کرده ایم
 - ۲- مسائل و مشکلات سازمان باربری را درک کردیم و پتانسیل ها و نقاط بهبود سازمان را یافتیم
 - ۳- به استنتاج و استخراج نیازمندی های سیستم نرم افزار باربری که باید سازمان باربری را در دستیابی به اهدافش، پشتیبانی نمایند پرداختیم.
- در دیسپلین مدیریت نیازمندی های سازمان برابری:

- ۱- میان مشتری و سایر ذینفعان با تیم تولید درباره چستی و چرایی سیستم نرم افزاری باربری به توافق رسیدیم
 - ۲- به درک مناسبی از نیازمندی های سیستم نرم افزار باربری رسیدیم
 - ۳- مدیریت مرزهای سیستم نرم افزاری باربری تعریف کردیم
 - ۴- مبنایی مناسب برای برنامه ریزی تکرارهای مختلف در روند تولید سیستم بر اساس موارد کاربرد فراهم کردیم
 - ۵- یک واسط کاربر که بر نیازها و اهداف کاربران متمرکز شده باشد تعریف کردیم
- دیسپلین تحلیل و طراحی:

- ۱- به ترجمه نیازمندی ها به توصیف هایی که چگونگی پیاده سازی سیستم نرم افزاری باربری را نشان دهند پرداختیم
- ۲- مدل های تحلیل و طراحی و تشریح ارتباط آن ها با سایر مدل ها در سیستم نرم افزاری باربری تعریف کردیم
- ۳- کلاس ها، زیر سیستم ها، و همکاری بین آنها در سیستم نرم افزاری باربری طراحی کردیم

در دیسپلین پیاده سازی:

- ۱- سازماندهی و ساختار کدهای برنامه در قالب زیرسیستم ها پرداختیم
- ۲- اشیاء و کلاسها را در قالب یکسری مؤلفه به نام کامپونت بیان کردیم

در دیسپلین محیط:

- ۱- به انتخاب و فراهم آوردن ابزارهای مناسب و ضروری برای سیستم نرم افزاری باربری پرداختیم
- ۲- ابزارهای مورد نیاز را نصب و پیکربندی کردیم
- ۳- خدمات فنی برای پشتیبانی از فرایندها در سیستم نرم افزاری ارائه دادیم

۵.۲ سخن آخر

در اینجا می خواهیم این پایان نامه را با بیان داستانی زیبا و پندآموز به پایان برسانیم؛ آقای پروفیسور اینشتن در یکی از ترم هایی که در دانشگاه کرسی استادی داشتند، سؤالاتی را در پایان ترم در اختیار دانشجویان قرار دادند. چند نفر از دانشجویان که سؤالات سال قبل استاد را مرور نموده بودند، متوجه شدند که سؤالات تکراری است. در این حال، یکی از دانشجویان از پروفیسور پرسید: استاد، به نظر می رسد اشتباهی رخ داده باشد چرا که سؤالات تکراری هستند! فکر می کنید که استاد در پاسخ این دانشجو چه گفتند؟ پاسخ ایشان چنین بود، فرمودند: **سؤالات امروز ما همان هایی است که دیروز هم با آنها مواجه بودیم، اما امروز، جواب ها متفاوت است.**



بهترین راه برای پیش بینی آینده، ساختن آن است (دانشمند علوم کامپیوتر Alan key)

۵.۳ منبع کل

محمد بدری، " معرفی آ.یو.پی (چهارچوب فرایند تولید نرم افزار)"، -

قابل دسترس در: نشانی مانه (www.software-academy.com) در اینترنت، خرداد ماه ۱۳۸۵.