

فصل اول - آشنایی با اتوماسیون

۱-۱- مقدمه

با گسترش روزافزون ارتباطات و پیدایش شکل‌های مختلف جریان‌ات ارتباطی، شبکه‌های ارتباطی در تمامی امور زندگی افراد سایه افکنده است و تجارت هم که در اقصی نقاط جهان به شیوه‌های مختلف (تولید کالا یا خدمات) دیده می‌شود از این قضیه مستثنی نیست. با پیشرفته و پیچیده‌تر شدن ارتباطات تجاری، سازمانها برای از دست ندادن یکی از منابع ورودی (اطلاعات) لاجرم دست به ایجاد سیستم‌هایی در درون خود زدند تا بتوانند از اطلاعات موجود در اطراف خود استفاده بهینه کنند و با پردازش مناسب این اطلاعات را به مشتریان و در جهت جلب رضایت آنان، عرضه کنند. در واقع محیط پرقاب‌ت کسب و کار و دگرگونی‌های این محیط در دهه ۱۹۹۰ (جهانی شدن اقتصاد و تبدیل اقتصادها و جوامع صنعتی به اقتصاد خدماتی برپایه دانش و اطلاعات) لزوم توجه به سیستم‌های اطلاعاتی را دو چندان کرد. به همین جهت، بحث سیستم‌های اتوماسیون که نوعی از انواع سیستم‌های اطلاعاتی است روزبه‌روز اهمیت بیشتری می‌یابد، چنانکه امروز سازمانهایی می‌توانند در اغلب اوقات خود را در بالاترین میزان آمادگی برای مقابله با تغییرات محیطی و داخلی بینند که از درجه بالایی از اتوماسیون سود برده باشند. اما سیستم‌های اتوماسیون اداری چیستند؟ چه مزایا و معایبی دارند؟ نیاز واقعی به تکامل و ایجاد آنها چه بوده است؟ انواع کاربردها و یا زیرسیستم‌های آن چیست؟ چه تفاوتی با سایر سیستم‌های اطلاعاتی دارد؟ و...

۱-۲- اهمیت اطلاعات و سیستم‌های اطلاعاتی

اطلاعات یکی از منابع باارزش اصلی مدیران یک سازمان است. همانطور که منابع انسانی، مواد اولیه و منابع مالی در روند تولید دارای نقش و ارزش خاصی هستند لکن در عصر اطلاعات و ارتباطات، اطلاعات دارای ارزش ویژه‌ای هستند. از طرفی اطلاعات کلید جامعه مردمی است و انتشار و استفاده از آن یک شاخص اجتماعی به‌شمار می‌رود. رشد این شاخص به معنای ارتقای ملی خواهد بود. اطلاعات به‌طور محسوسی بر پیشرفت رفتار ما اثر می‌گذارد.

فناوری و ابزارهای الکترونیک و رایانه‌ای نیز در دهه گذشته پدیده انفجار اطلاعات را موجب شدند و به‌طور حتم تأثیر مهمی را بر جهت‌گیری جوامع و اطلاعات مورد نیاز آنها خواهند گذاشت. امروزه نه تنها مدیران ارشد و مدیران اجرایی، بلکه تمام اقشار اجتماع چون محققان و دانشوران و تجار ناگزیر از استفاده اطلاعات هستند. کاربران نظام اطلاعات، اطلاعات را چون یک منبع

ارزشمند، هم سنگ سرمایه و نیروی کار به کار می‌برند. از آنجا که اطلاعات مهم و ارزشمند هستند و اساسی برای کل فعالیتهای سازمان محسوب می‌شوند، لذا بایستی سیستم‌هایی را برپا کرد تا بتوانند اطلاعات را تولید و آنها را مدیریت کنند. هدف نهایی چنین سیستم‌هایی کسب اطمینان از صحت، اعتبار و روایی اطلاعات در دسترس در زمان مورد نیاز و به شکل قابل استفاده است. امروزه سیستم‌های اطلاعاتی نقش اساسی در همه زمینه‌های فعالیت یک شرکت ایفا می‌کنند. توجه به شرکتهای موفق نشان می‌دهد که همگی آنها به سیستم‌های اطلاعاتی جهت فعالیت روزانه‌شان مجهزند. چالش حقیقی که شرکتهای با آن روبرو هستند، صرفاً به کارگیری سیستم‌های اطلاعاتی متکی به رایانه نیست، بلکه هدف اساسی استفاده اثربخش سیستم‌های اطلاعاتی در مدیریت است. سیستم‌های اطلاعاتی که به‌عنوان منبعی ارزشمند محسوب می‌شوند، توانایی مدیران و کارکنان را افزایش داده و امکان تحقق اثربخش اهداف سازمان را با بهره‌وری بالا موجب می‌گردند.

۱-۳- انواع سیستمهای اطلاعاتی

در نخستین سالهای اختراع رایانه متخصصان توجه چندانی به نیازهای اطلاعاتی مدیران نداشته و از رایانه‌ها فقط جهت پردازش داده‌ها در امور حسابداری استفاده می‌شد. در طول این دوره که تا اواسط دهه ۶۰ به طول انجامید بیشترین تاکید بر قدرت محاسباتی و پردازش رایانه بود. در حال حاضر به این نوع کاربرد رایانه پردازش داده‌ها، گفته می‌شود و باید توجه داشت که سیستم‌های پردازش داده مقداری اطلاعات نیز تولید می‌کنند.

در اوائل دهه ۱۹۶۰ و پس از اختراع انواع رایانه‌های جدید که با هزینه کمتر اطلاعات بیشتری را پردازش می‌کردند روشهای جدیدتری برای معرفی تجهیزات جدید ترویج داده شده؛ یعنی سیستم‌های اطلاعاتی مدیریت. پس از گذشت یک دهه و به دلیل به‌وجود آمدن نیازهای اطلاعاتی جدید، سیستم اطلاعاتی جدید به‌نام سیستم پشتیبانی تصمیم‌گیری یا SYSTEM SUPPORT DECISION (DSS) به‌وجود آمد. پس از تکمیل سیستم پشتیبانی تصمیم‌گیری با توجه به پیشرفتهای حاصله در تجهیزات رایانه‌ای، موضوع افزایش بهره‌وری و کارایی دفاتر مدیران و تسهیل ارتباطات بین مدیر و کارکنان از طریق به کارگیری تجهیزات رایانه‌ای و الکترونیک مطرح گردید و در سال ۱۹۸۰ سیستم‌های اتوماسیون اداری طراحی گردیدند.

اما در واقع اتوماسیون اداری در سال ۱۹۶۴ وقتی که IBM محصول جدید خود، یعنی نوار مغناطیسی/ماشین تایپ سلک توری (MT/ST) را معرفی کرد شروع شد، ماشین تایپی که می‌توانست به‌صورت اتوماتیک از روی حروف ضبط شده بر روی نوار مغناطیسی تایپ کند. این عملیات تایپ اتوماتیک خیلی زود به سیستم‌های کوچک در ریزپردازنده‌ها تبدیل شد. و تکاملهای

بعدی در این زمینه ایجاد شد. و نیاز واقعی به تکامل این بود که طی دهه ۱۹۷۰ کارایی کارخانه‌ها ۸۵-۹۰ درصد افزایش یافت، در حالی که کارایی دفتری تنها ۴ درصد افزایش داشت، پس بایستی سیستم‌هایی به وجود می‌آمدند که موجب افزایش بهره‌وری و کارایی دفاتر هم می‌شدند.

۱-۴- سیر تکاملی فرآیند

سیر تکامل فناوری اداری شامل دوره‌های فناوری اداری، فناوری رایانه‌ای و فناوری ارتباطات است. در دوره اول کارفرمایان در تلاش بودند تا محیطی را به وجود آورند که جدا از مسائل کارخانه‌ای و در معنای عام محیط تولید، کلیه امور اداری، پرسنلی و تجارتي در آن صورت گیرد، از مهمترین شاخصهای این دوره انجام مطالعاتی در مورد جایابی و فضاسازی برای محیطهای اداری بود. (دهه ۲۰ به بعد) در حقیقت در این دوره سعی در جدا ساختن فعالیتهای اداری از فعالیتهای تولیدی بود و در نهایت واحدهای ستادی از لحاظ مکانی از واحدهای اجرایی جدا گردیدند. تفکر این دوره این چنین بود که محیطی با شرایط بهتر برای انجام امور اداری که در آن زمان، بیشتر امور محاسباتی نظیر حسابداری، حقوق و دستمزد و مالی بود، فراهم گردد.

در دوره دوم؛ یعنی فناوری رایانه، با ورود رایانه‌های کوچک که از سرعت و دقت بالایی بهره‌مند بودند، مدیران تصمیم گرفتند که با استفاده از این وسایل و سرمایه‌گذاری در رایانه‌ای کردن سازمان خود از سرعت و دقت رایانه در کاهش زمان کار و حذف اشتباهات انسانی استفاده کرده و پیامد این حرکت به وجود آمده این بود که کاربران را برای انجام امور روزمره‌شان به خوبی یاری کرد.

در دوره سوم؛ یعنی فناوری ارتباطات، با گسترش روزافزون و همه جانبه علوم و فناوری رایانه و استفاده از وسایل جانبی آن و یکپارچه شدن و ادغام هر یک از این سیستم‌ها در یکدیگر. در حقیقت امروزه سیستم‌های اداری سیستم‌های جهانی هستند که وظیفه اصلی‌شان ایجاد ارتباط و بهبود ارتباطات هستند. نوعاً ارتباطات از لحاظ اطلاعات تجاری از اهمیت بسزایی برخوردار است. از مشخصات دیگر این دوره که الان در آن به سر می‌بریم، این است که سازمانها و شرکتهای به ارزش واقعی اطلاعات به هنگام و یا کیفیت پی برده‌اند و در بازارهای پیچیده امروزی نبود ارتباط با منابع اطلاعات به معنای از دور خارج شدن سازمان است. به دست آوردن ارتباطات درست و با کیفیت و مربوط به معنی هماهنگی و سازگار بودن سازمان با محیط و جوابگو بودن نسبت به تغییرات است.

۱-۵- تعریف اتوماسیون

بسیاری عقیده دارند که سیستم و چارچوبی به نام اتوماسیون وجود ندارد، بلکه ترکیبی از وسایل و تجهیزات گوناگون برای تسهیل در امور مرتبط با فعالیتهای اداری را اتوماسیون اداری می‌نامند.

اما از دهه ۱۹۶۰ که جنبه‌های بیشتری از کاربردها و فعالیتهای اداری و بازرگانی گسترش یافت، وجود یک سیستم یکپارچه اداری، مناسب که حجم عظیم اطلاعات، مکاتبات، مراسلات را در برگیرد به وضوح احساس شد که با نامهای مختلفی همچون سیستم‌های اداری، سیستم‌های اطلاعات اداری، سیستم‌های کاربر نهایی و سیستم‌های محاسباتی کاربر نهایی نامیده شده است. ولی عمومی‌ترین و بالاترین درجه از اتوماسیون سیستم‌های اداری به نام اتوماسیون اداری نامیده می‌شود. سیستم‌های فوق تعریف مشخصی ندارند بلکه منحصر به تعیین دیدگاههای کاربر است و این خود بدین معنی است که سیستم‌های اتوماسیون اداری دارای تعاریف بی‌شماری است. در اینجا به برخی از آنها اشاره می‌کنیم.

اتوماسیون اداری، مشتمل بر تمام سیستم‌های الکترونیک رسمی و غیررسمی بوده که به برقراری ارتباط اطلاعات بین اشخاص در داخل و خارج موسسه و بالعکس مربوط می‌شود. کلمه اصلی که اتوماسیون اداری را از داده‌پردازی، سیستم اطلاعات مدیریت و سیستم پشتیبانی تصمیم متمایز می‌سازد ارتباطات است. اتوماسیون اداری به منظور تسهیل انواع ارتباطات به هر دو صورت شفاهی و کتبی است.

سیستم‌های اطلاعات اداری (OIS) از امور اداری از طریق فناوری اطلاعاتی حمایت می‌کند. سیستم‌های اطلاعات مدرن از جمله اجزای سیستم اطلاعاتی مدیریت (MIS) است که ابزارهایی را برای ارتباطات و هماهنگی مابین کارگران دانشی به وسیله ایجاد مدیریت اثربخش اسناد و پیامها و جلسات الکترونیک فراهم می‌کند. اتوماسیون اداری عبارت است از کاربرد وسایل الکترونیک در فعالیتهای دفتری به منظور افزایش کارایی؛ کارایی افزایش یافته ناشی از تکامل تبادل اطلاعات، در داخل دفتر و بین دفاتر و محیط آنها بوده و در نتیجه با ارائه اطلاعات بهتر برای تصمیم‌گیری می‌تواند به مدیر سود برساند.

۱-۵-۱- لزوم استفاده جامع از سیستم اتوماسیون

با توجه به اینکه حجم مکاتبات اداری و گردش نامه‌ها و اسناد در سازمانها و مؤسسات اداری بسیار بالا می‌باشد، وجود یک سیستم نامناسب مکاتبات اداری باعث اتلاف زمان، صرف انرژی بیشتر، افزایش هزینه‌ها و در نتیجه غیر بهره‌ور شدن سازمان

می‌شود. تنها یک سیستم مناسب و سریع اداری می‌تواند حجم عمده‌ای از پاسخ‌ها را کاهش داده و باعث صرفه‌جویی اقتصادی و افزایش بهره‌وری گردد. در هر ارگانی که الگوی سازمان الکترونیکی به‌طور صحیح و کامل پیاده‌سازی شود، آن ارگان تبدیل به سازمانی بدون نیاز به کاغذ، با عکس‌العمل‌هایی سریع و کارآمد خواهد شد.

در یک سازمان الکترونیکی امکانات فراوان اینترنتی و اینترنتی وجود دارد که در سایه یکپارچگی این امکانات، اطلاعات به شکل صحیحی جریان می‌یابد و سازمان با هر حجمی که باشد می‌تواند سرعت و صحت را در کنار هم در اختیار داشته باشد.

با قرارگیری در قالب یک سازمان الکترونیکی کارآمد، زمان و انرژی بیاری حفظ خواهد شد. مثلاً دیگر نیازی به هماهنگی بین مدیران برای برگزاری جلسات مداوم تصمیم‌گیری نخواهد بود و بسیاری از تصمیم‌گیری‌ها در قالب نامه‌های الکترونیکی انجام می‌پذیرد.

۶-۱- مزایا و معایب اتوماسیون

مهمترین شرط کاربرد و استفاده از اتوماسیون در اکثر سازمانها، صرف‌نظر از موارد استفاده دیگر آن، مربوط به لزوم سرعت و دقت و صحت در کار آنها است. و به خاطر توسعه دامنه عملیات سازمان است که این امر احتیاج به گسترش مجاری ارتباطات با سرعت بیشتر است. و اتوماسیون اداری موجب می‌گردد که مجاری ارتباطی کوتاه و ارتباطات کوتاه به‌سهولت در اختیار مدیران قرار گیرد.

به‌طور کلی، مزایای کاربرد سیستم‌های اتوماسیون به دو دسته مزایای مستقیم و مزایای غیرمستقیم تقسیم می‌شود.

۶-۱-۱- مزایای مستقیم

این مزایا عبارتند از: افزایش محصول یا بازده و صرفه‌جویی در وقت یا نیروی کار. معمولاً این مزایا که قابل اندازه‌گیری بوده و ممکن است تأثیر مستقیم و کوتاه‌مدت بر جریان نقدینگی داشته باشد عبارتند از:

(الف) کنترل بهتر بر کار، به خاطر تقسیم کمتر نیروی کار؛

(ب) تبدیل اطلاعات از شکلی به شکلی دیگر کمتر صورت می‌گیرد، مانند نوشتن روی نوار که پس از آن روی کاغذ تایپ می‌شود؛

(ج) فعالیتهای غیرمولد مانند بایگانی، نگهداری سوابق و به‌هنگام‌رسانی کمتر می‌شود؛

د) سازماندهی پرسنل بهتر انجام می‌شود. چرا که با استفاده از امکانات کنفرانس تلفنی، مسافرت و گردهمایی کمتر می‌گردد.

۱-۲- مزایای غیر مستقیم

این مزایای غیر کمی هستند و ممکن است از طریق سودآوری و رشد در بلندمدت سازمان را غنی سازند، این مزایا عبارتند از:

الف) وابستگی کمتر به ادارات دیگر برای تهیه کپی، چاپ و امور مشابه دیگر؛

ب) نیاز کمتر به تشریفات و کنترل جهت نظارت بر جریان کار بین ادارات؛

ج) به دلیل افزایش اثربخشی کارکنان در انجام کارهای خاص، رضایت شغلی آنها افزایش می‌یابد؛

د) به دلیل ارائه بهتر اطلاعات و خدمات به موقع، رضایت مشتریان بیشتر می‌شود؛

ه) رقابت بیشتر سازمانها از طریق استفاده از منابع اطلاعاتی و قابلیت لازم برای عکس العمل نسبت به فشارها و فرصتها.

۱-۳- معایب اتوماسیون

اما کاربرد سیستم‌های اتوماسیون معایبی را نیز دربردارد که عبارتند از: ایجاد تغییرات در محیط انسانی، نادیده گرفته شدن برخی از روابط اجتماعی و انسانی در کارها و به صورت مکانیزه در آمدن فعالیتها، پیچیده شدن و سختی کار با سیستم‌ها، بروز اخلاق توجیه اشتباهات صورت گرفته و نسبت دادن آنها به سیستم، مشکلات جسمانی کارکنان در کار با رایانه، نپذیرفتن سیستم‌های مکانیزه توسط مدیران و یا کارکنان، کم شدن امنیت اطلاعات هم از نظر دسترسی و هم از نظر تخریب.

۱-۷- انواع کاربردها یا زیرسیستمهای اتوماسیون

از دیدگاههای متعددی به انواع کاربردهای اتوماسیون پرداخته شده است که در این جا به دو نمونه از آنها می‌پردازیم. در یک دیدگاه انواع دستگاههای مکانیزه اداری اصلی عبارتند از:

۱- در زمینه کسب اطلاعات و کپی کردن آنها: واژه پردازها، ماشینهای کپی هوشمند، سندخوانها، سیستم‌های صوتی دیجیتالی، حروف چینی نوری.

۲- در زمینه ذخیره سازی اطلاعات: کشورهای بایگانی الکترونیک، زیر نگاهشته‌ها (میکروگرافیک).

۳- در زمینه ارتباطات: سیستم تلفن رایانه‌ای، پست الکترونیک، ارسال فاکس، تسهیلات کنفرانس از راه دور.

۸-۱- ارگونومیک

پیشرفت فناوری در قرن گذشته و بویژه از جنگ جهانی دوم به بعد در رشد اقتصادی و پیشرفت اجتماعی جهان صنعتی اثر چشمگیر داشته است. بی تردید پیشرفت فناوری علاوه بر ارتقای سطح زندگی بشر در کاهش شدید بسیاری از منابع حوادث، آسیبها و تنشهای کاری، نقش برجسته‌ای داشته است. با این حال، فناوری پیشرفته به همراه خود منابع جدیدی از آسیب و تنش ناشی از کار آورده است. و بایستی در نظر داشته باشیم که ما به شرطی می‌توانیم عملکرد را بهینه کرده و به بهره‌وری بیشتر و تامین بهداشت و سلامت کارکنان برسیم که اصول ارگونومی را در طراحی، عرضه و استفاده از فناوری منظور کنیم.

اصطلاح ارگونومیک مطالعه عوامل را که راحتی، ارضاء و کارایی افرادی که با سیستم‌ها و وسایل تولید، کار می‌کنند را تشریح می‌کند.

وقتی کارکنان در وضعیتی تحت فشار قرار گیرند و برای مدت زیادی به صفحه مانیتور نگاه کنند اغلب آنها دچار سردرد، کمردرد و چشم درد و... می‌شوند با در نظر گرفتن بسیاری از مسائل ایمنی و بهداشتی در ساخت و تولید و عرضه وسایل اتوماسیون اداری در واقع ما در نظر گرفته‌ایم که کارکنان دفتری قسمت مهمی از داراییهای سازمان هستند و اگر خسته و تحت-تاثیر باشند، نمی‌توانند به بهترین وجه کار کرده و کارایی سیستم و فراهم ساختن اطلاعات لازم جهت پشتیبانی مدیران با مشکل مواجه می‌شود.

از طرف دیگر، می‌توان گفت اتوماسیون اداری فراتر از اضافه کردن سیستم‌های نرم‌افزاری جدید است و بایستی مسائل مربوط به تغییر و مقاومت در برابر تغییر را هم مدنظر داشت، یک تحقیق پیمایشی انجام شده در سوئد نشان داده است که هر تغییر روانی - اجتماعی در محیط کاری، همانند رایانه‌ای شدن، سیستم نرواندوکرین بدن را فعال کرده و موجب واکنشهای فیزیولوژیک و روانی او می‌شود که فرد را برای مبارزه یا فرار تحریک می‌کند و در واقع فرد با مکانیزاسیون که نوعی تغییر محسوب می‌شود مقاومت می‌کند. و مدیران و سایر طراحان و مسئولان ذیربط بایستی با کاربرد تکنیک‌های غلبه بر مقاومت که مهمترین آنها مشارکت و آموزش کاربران است، بر این مقاومت غلبه کرده و یا میزان آن را کاهش دهند.

۹-۱- مقایسه انواع سیستمهای اطلاعاتی

اتوماسیون اداری برای جریان اطلاعات در داخل سازمان مناسب‌تر است تا بین سازمان و محیط؛ زیرا خیلی از کاربردهای اتوماسیون اداری به فرستنده و دریافت‌کننده برای استفاده از وسایل سازگار و یا پروتکل‌های ارتباطات نیاز دارد. البته باید مدنظر

داشت که اتوماسیون اداری تمام ارتباطات را فراهم نمی‌سازد. مدیران هنوز از تلفن، جلسات حضوری و خواندن گزارشهای اداری استفاده می‌کنند. اتوماسیون اداری راهی برای تکمیل و ایجاد ارتباطات خاص است، ولی هنوز مدیران به استفاده از گزارشهای غیررسمی ادامه می‌دهند.

اتوماسیون اداری عمدتاً یک سیستم رسمی با رویه تعریف شده است که بایستی در ارتباط با دیگر سیستم‌های رسمی مانند سیستم پشتیبانی تصمیم‌گیری مورد استفاده قرار گیرد. کاربرد اتوماسیون اداری بسیار ساده است. بنابراین، احتمال استفاده مدیر از آن بیشتر است. یک مدیر بایستی هم رایانه و هم دانش MIS را به منظور استفاده از بعضی از ابزارهای پیچیده سیستم پشتیبانی تصمیم‌گیری دارا باشد.

ولی استفاده از اتوماسیون اداری احتیاج به چنین ابزار پیچیده توسط مدیر را ندارد به همین دلیل اتوماسیون اداری از جذابیت خاصی برخوردار است.

از لحاظ حل مسئله و فضای حل مسئله نیز می‌توان کاربردهای انواع سیستم‌های اطلاعاتی را با هم مورد مقایسه قرار داد، شکل شماره ۲ نشان‌دهنده این تفاوت است. کاربردهای اتوماسیون اداری با شخص مدیران انطباق یافته و بهترین شکل برای مسائل کمتر ساختار یافته که ارتباطات غیررسمی فراوان است وفق می‌یابد. با وجود آنکه سیستم اتوماسیون اداری اطلاعات حل مسئله را در اختیار قرار می‌دهد، تصمیم در مورد نحوه به کارگیری آن به مدیران محول می‌شود.

۱۰-۱- نتیجه‌گیری

طی دهه ۱۹۷۰ کارایی کارخانه‌ها ۸۵-۹۰ درصد بالا رفت، در حالی که کارایی دفتری تنها ۴ درصد افزایش داشت، پس بایستی سیستم‌هایی به وجود می‌آمدند که موجب افزایش بهره‌وری و کارایی دفاتر هم می‌شدند، از این‌رو، بعد از تکامل سیستم‌های پردازش داده (DPS)، سیستم‌های اطلاعاتی مدیریت و سیستم‌های پشتیبانی تصمیم‌گیری، سیستم‌های اتوماسیون اداری به وجود آمدند. این سیستم‌ها از طریق فناوری اطلاعاتی از امور اداری حمایت کرده و موجب افزایش کارایی دفاتر می‌شوند. کارایی افزایش یافته ناشی از تکامل تبادل اطلاعات و سرعت و صحت اطلاعات در داخل دفتر و بین دفاتر و محیط آنها بوده و در نتیجه با ارائه اطلاعات بهتر برای تصمیم‌گیری می‌تواند به مدیر سود برساند.

اما بایستی توجه داشت که در حل یک مسئله بخش عمده‌ای از آن از طریق ارتباطات بین فردی حاصل می‌شود. مدیر و کارشناس اطلاعات باید به اتوماسیون اداری به‌عنوان ابزارهای مکمل چنین ارتباطات بین فردی بنگرند و در عین حالی که کاربردهای اتوماسیون اداری موجب کارایی و صرفه‌جویی هزینه‌ها و افزایش سرعت و ایجاد اداره بدون کاغذ می‌شود، از ابعاد

دیگر موجب درهم و برهم شدن اسناد و پوشه‌ها به وسیله چاپگر رایانه، یا افزایش پیامهای بی‌اهمیت و جزئی و پخش وسیع پیامها در مورد پست الکترونیک می‌شود. متأسفانه مطالعات آماری فقط نشان‌دهنده این هستند که کاربرد اتوماسیون اداری در حال گسترش زیادی است ولی این امر به ما نشان نمی‌دهد که آیا واقعاً اتوماسیون اداری برای ادارات مفید است یا فقط طبق فناوری روز ورود آن به ادارات دیده می‌شود. یک مسئله مشخص است: کمپانی‌هایی که طبق مطالعات آماری اقدام به جایگزینی سیستم‌های اتوماسیون اداری در طیف وسیع می‌کنند لزوماً موفق نیستند بلکه مطالعات نشان داده است بهتر است در هر اداره از اتوماسیون اداری در جایی که مفید واقع می‌شود و واقعاً به آن نیاز است استفاده بشود. در مقاله بیزینس و یک اشاره شده کمپانی‌هایی که بهترین اهداف را از به کارگیری اتوماسیون اداری کسب کرده‌اند و بالاترین کارآمدی را دارند آنها هستند که مشخصاً از اتوماسیون اداری در جای مربوط استفاده می‌کنند.

نکته آخر و مهم اینکه در کاربرد سیستم‌های اتوماسیون اداری و اعمال تغییر و تحولات از طریق آنها بایستی به مباحث ارگونومیک (مهندسی انسانی) و مقاومت در برابر تغییر کارکنان و تکنیک‌های مناسب غلبه بر مقاومت آنها نیز توجه اساسی داشته باشیم و با استفاده از نگرش سیستمی، اقدام به تغییر و تحولات و به کارگیری سیستم‌های اتوماسیون اداری کنیم و بدانیم که نیروی انسانی هم به‌عنوان مهمترین عامل ایجادکننده تغییر و هم به‌عنوان پذیرنده و اجراکننده آن است.

فصل دوم- زیرساخت‌های انتخابی

۲-۱- سبک‌های برنامه نویسی:

به طور کلی سه سبک برنامه نویسی تا کنون مورد استفاده قرار گرفته اند شامل سبک سنتی و سبک برنامه نویسی ساخت یافته و سبک برنامه نویسی شیء گرا.

۲-۱-۱- سبک برنامه نویسی سنتی:

در سبک برنامه نویسی سنتی نوشتن برنامه ها و ردیابی اجرای آنها بسیار دشوار بود زیرا با استفاده از دستور goto کنترل اجرای برنامه از نقطه ای به نقطه دیگر منتقل می شد. نگهداری این برنامه ها و انجام تغییرات در آنها برای پاسخگویی به وضعیت و نیازهای جدید بسیار دشوار بود. این سبک برنامه نویسی کاملاً منسوخ شده است.

برنامه نویسی ساخت یافته:

برنامه نویسی ساخت یافته روش منظمی برای نوشتن برنامه هاست و منجر به نوشتن برنامه هایی میشود که خوانایی آنها بالاست و تست و اشکال زدایی راحت تر و اصلاح آنها آسانتر است. در برنامه نویسی ساخت یافته به صورت مجموعه ای از فعالیت ها تصور می شود که باید بر روی داده ها انجام شوند. در این روش هر کار پیچیده ای به مجموعه ای از کارهای کوچکتر تجزیه میشود تا اینکه کارهای ایجاد شده قابل درک باشند. به عبارت دیگر برنامه نویس سعی میکند توابعی بنویسد که نیازمندیهای سیستم را برآورده کنند.

برنامه نویسی ساخت یافته روش موفقتری برای حل مسائل پیچیده است اما مشکلات خاص خود را دارد. در این روش داده ها از فعالیت ها (توابعی) که آنها را پردازش میکنند جداست. وقتی حجم داده ها زیاد میشوند نگهداری آنها مشکل می شود. هرچه بیشتر با داده ها کار میکنید با مشکلات بیشتری مواجه میشوید.

بعضی از زبانهای برنامه سازی که برای برنامه نویسی ساخت یافته طراحی شده اند عبارتند از زبان `pascal` و `Ada`.

۲-۱-۲- برنامه نویسی شیئی گرا:

برنامه نویسی شیئی گرا شیوه نوینی است که در آن می توان قطعاتی را ایجاد کرد و در برنامه های مختلف مورد استفاده قرار داد. قابلیت خوانایی برنامه هایی که در این روش نوشته میشوند بالا بوده و تست و عیب یابی و اصلاح آنها آسان است. شیئی گرایی بر اشیا تکیه دارد. تعدادی از زبانها برای برنامه نویسی شیئی گرا طراحی شده اند. زبان `smalltalk` یک زبان برنامه نویسی کاملا شیئی گراست. زبان `C++` که تکامل یافته زبان `C` است دارای خصوصیات است که برنامه نویسی شیئی گرا در آن امکان پذیر است. در این زبان برنامه نویسی ساخت یافته نیز ممکن است. `Visual Basic` زبانی کاملا شیئی گراست.

۲-۲- تکنولوژی NET

قبل از هر چیز بهتر است تعریفی دقیقی از کلمات فریم ورک یا چارچوب و هم چنین پلتفرم ارائه دهیم. در تعریف NET می-توانیم بگوییم که چارچوب NET یک پلتفرم جدید است که توسط مایکروسافت برای طراحی و توسعه نرم افزار ایجاد شده است.

نکته جالبی که در این تعریف وجود دارد ابهام زیادی است که در این تعریف به کار برده ام. اما برای این کار دلیل خوبی وجود دارد برای شروع توجه کنید که در این تعریف نگفته ام طراحی و توسعه نرم افزار برای سیستم عامل ویندوز اگرچه مایکروسافت چارچوب NET¹ را برای اجرا بر روی سیستم عامل ویندوز منتشر کرده است. به زودی نسخه های دیگری از این چارچوب را مشاهده خواهید کرد که بر روی سیستم عامل های دیگر مانند لینوکس نیز اجرا می شوند. یکی از این نسخه ها مونو² است. موتو

¹. NET Framework

². Mono

یک نسخه متن باز از چارچوب NET است. (که شامل یک کامپایلر C# نیز هست) که برای سیستم عامل‌های گوناگونی مانند نسخه‌های مختلف لینوکس و مکینتاش منتشر شده است.

پروژه‌های بسیار دیگری مشابه موتو در حال اجرا هستند که ممکن است هنگام انتشار این کتاب در اختیار شما قرار گرفته باشند. به علاوه می‌توانید با استفاده از نسخه فشرده این چارچوب به نام NEW Microsoft. Compact Framework زیر مجموعه‌ای از چارچوب NET است برای وسایل هوشمند مانند دستیار دیجیتال شخصی^۳ و یا موبایل‌ها نیز برنامه بنویسید. (با این چارچوب در فصل بیست و دوم بیشتر آشنا خواهیم شد).

اگر به تعریفی که در بالا برای چارچوب NET آورده شده است دقت کنید مشاهده می‌کنید که این تعریف محدود به نوع خاصی از برنامه‌ها نیست. در حقیقت در مورد نوع برنامه‌هایی که می‌توان با NET نوشت هیچ محدودیتی وجود ندارد که بخواهیم آن را ذکر کنیم از چارچوب NET می‌توانید برای طراحی برنامه‌های تحت ویندوز، تحت وب سرویس‌های مبتنی بر وب و... استفاده کنید.

چارچوب NET یک چارچوب کلی است و محدود به زبان برنامه‌نویسی خاصی نیست. شما می‌توانید برنامه خودتان را به هر زبانی که بخواهید بنویسید. در این کتاب برنامه‌نویسی به زبان C# را بررسی می‌کنیم. اما علاوه بر این زبان می‌توانید از زبان‌هایی مانند ++C و ویژوال بیسیک. جاوا و حتی زبان‌های قدیمی مانند COBOL نیز استفاده کنید. برای هر کدام از این زبان‌ها یک کامپایلر خاص NET ارائه می‌شود. به وسیله این کامپایلر برنامه‌های نوشته شده به این زبان‌ها نه تنها می‌توانند با چارچوب NET ارتباط داشته باشند. بلکه می‌توانند با برنامه‌های زبان‌های دیگر که تحت NET نوشته شده‌اند نیز ارتباط داشته باشند. برای مثال یک برنامه که به زبان C# نوشته شده است به راحتی می‌تواند از کدی استفاده کند که به زبان ویژوال بیسیک نوشته شده است و یا برعکس.

مواردی که تاکنون گفتیم سطح بالای تنوع در NET را نشان می‌دهند. این تنوع یکی از دلایلی است که باعث می‌شود چارچوب NET چنین دورنمایی جذابی داشته باشد.

پس دقت داشته باشید که NET یک زبان برنامه‌نویسی یک مدل برنامه‌نویسی مانند برنامه‌نویسی تحت ویندوز، یک نوع برنامه‌نویسی برای سیستم عاملی خاص مانند برنامه‌نویسی تحت ویندوز و یا مواردی از این قبیل نیست بلکه NET یک روش برای

³ . Personal Digital Assistant

طراحی و توسعه نرم افزار است که به وسیله مایکروسافت معرفی شده است و می تواند در تمامی مواردی که در بالا ذکر شد مورد استفاده قرار گیرد.

۲-۳- چارچوب NET از چه اجزایی تشکیل شده است؟

یکی از اجزای چارچوب NET کتابخانه کلاس عظیم آن است که می توانید از آن در برنامه های خود استفاده کنید. کتابخانه کلاس^۴ یک مجموعه از توابع و کلاس ها است که برای انجام امور مختلف مورد استفاده قرار می گیرد. برای مثال یک کتابخانه کلاس شامل توابعی برای کنترل ورودی و خروجی استفاده از امکانات چاپ، کار با انواع مختلف شبکه ها و... است. این توابع و کلاس ها که با استفاده از تکنیک های برنامه نویسی شیء گرا نوشته شده اند. در NET به گروه ها و با فضای نام های مختلفی دسته بندی می شوند. با مفهوم فضای نام در فصل ۹ بیشتر آشنا خواهیم شد. در نوشتن یک برنامه، می توانید هر کدام از فضای نامها را که نیاز داشتید به برنامه اضافه کنید. برای مثال یکی از این فضای نامها برای برنامه نویسی تحت ویندوز به کار می رود، یکی دیگر برای برنامه نویسی شبکه مورد استفاده قرار می گیرد، فضای نام دیگری برای برنامه نویسی تحت وب به کار می رود. بعضی از این فضای نامها خود به فضای نامهای کوچکتری تقسیم می شوند که برای کاربرد خاصی در آن قسمت استفاده می-شوند. برای مثال فضای نام برنامه نویسی تحت وب شامل یک فضای نام کوچکتر است که برای نوشتن سرویس های تحت وب بکار می رود.

باید توجه داشته باشید که تمام سیستم عامل ها، همه توابع موجود ر این فضای نامها را پشتیبانی نمی کنند. برای مثال یک دستیار دیجیتال شخصی از توابع اصلی چارچوب NET، پشتیبانی می کند، اما یک سری از توابع که در این وسایل کاربردی ندارد به-وسیله آنها پشتیبانی نمی شود.

بخش دیگری از چارچوب NET، یک سری نوع های داده ای ابتدایی را تعریف می کند. نوع های داده ای برای نگهداری اطلاعات یک برنامه در طول اجرای آن مورد استفاده قرار می گیرند.

نوع های داده ای که در این قسمت از NET، تعریف می شوند به صورت بسیار پایه ای هستند (مانند «عدد صحیح علامت دار ۳۲ بیتی»).

4 . CLASS LIBRARY

نوع‌های داده‌ای پیشرفته‌تری که در زبانهای برنامه‌نویسی مبتنی بر NET مانند C# و با ویژوال بیسیک وجود دارند باید بر اساس یکی از این نوع‌های داده‌ای تعریف شده در این قسمت از چارچوب NET باشند.

این مورد باعث هماهنگی بین زبانهای برنامه‌نویسی می‌شود که از چارچوب NET استفاده می‌کنند. این قسمت از چارچوب NET، سیستم نوع داده‌ای عمومی و یا به اختصار CTS^۵ نامیده می‌شود. با نوع‌های داده‌ای در فصل بعد بیشتر آشنا خواهیم شد.

علاوه بر کتابخانه کلاسی که ذکر شد چارچوب NET شامل بخشی به نام زبان عمومی زمان اجرا و یا به اختصار CLR^۶ است. این بخش از چارچوب NET (که مهمترین بخش آن نیز محسوب می‌شود) مسئول کنترل و مدیریت اجرای تمام برنامه‌هایی است که با استفاده از کتابخانه کلاس NET نوشته شده‌اند.

چگونه با استفاده از چارچوب NET برنامه بنویسیم؟

نوشتن برنامه با استفاده از چارچوب NET به معنی نوشتن کد به هر کدام از زبان‌هایی که توسط NET پشتیبانی می‌شوند، با استفاده از کتابخانه کلاس NET. همانطور که گفتیم در طول این کتاب از محیط طراحی مجتمع ویژوال استودیو (IDE) برای طراحی و برنامه‌نویسی استفاده می‌کنیم. مزیت استفاده از این محیط این است که می‌توانید به راحتی از ویژگی‌هایی که بر بخش‌های قبلی از چارچوب NET معرفی کردیم استفاده کنید. کدی که شما برای نوشتن برنامه‌ها در طول این کتاب استفاده می‌کنید کلا به زبان C# است، اما در طول برنامه‌ها از چارچوب NET. و همچنین از یک سری ویژگی‌های و ابزارهایی که محیط ویژوال استودیو در اختیار ما قرار می‌دهد استفاده خواهیم کرد.

یک برنامه که به زبان C# نوشته شده است قبل از اجرا باید به کدی تبدیل شود که برای سیستم عامل قابل فهم باشد. به این کد، کد محلی می‌گویند.^۷ تبدیل یک کد از هر زبانی به کد محلی که برای سیستم عامل قابل فهم باشد را کامپایل کردن می‌گویند و عملی است که به وسیله کامپایلر انجام می‌شود. چارچوب NET این بخش از دو مرحله تشکیل شده است.

^۵ .COMMON TYPE SYSTEM

^۶ .Common language Runtime

^۷ .Native code

۲-۴- JIT , MSIL

هنگامی که برنامه‌ای که در آن از توابع موجود در کتابخانه کلاس NET است داده شده است را کامپایل می‌کنید، بلافاصله کد قابل فهم برای سیستم عامل و یا کد محلی تولید نمی‌شود. در عوض کد شما به زبانی به نام زبان سطح میانی مایکروسافت و یا به اختصار MSIL^۸ تبدیل می‌شود. این کد برای سیستم عامل خاصی نیست و همچنین منحصر به زبان C# نیز نیست. به عبارت دیگر کد زبان‌های دیگر نیز می‌تواند به MSIL تبدیل شود (و البته باید تبدیل شود). کدهای زبان‌های دیگری که از چارچوب NET استفاده می‌کنند نیز (مانند ویژوال بیسیک) هنگامی کامپایل ابتدا به زبان MSIL تبدیل می‌شوند. هنگام استفاده از ویژوال استودیو برای نوشتن برنامه، این مرحله از کامپایل توسط ویژوال استودیو انجام می‌شود.

اما برای اجرای یک برنامه توسط سیستم عامل یک مرحله دیگر نیز مورد نیاز است. این مرحله وظیفه‌ی کامپایلر JUST – In Time – و یا به اختصار JIT کامپایلر است. این کامپایلر کد MSIL یک برنامه را دریافت کرده و آن را به کدی تبدیل می‌کند که به وسیله سیستم عامل قابل اجرا باشد. بعد از اینکه این تبدیل توسط JIT انجام شد، سیستم عامل می‌تواند برنامه را اجرا کند.

همانطور که از اسم این قسمت نیز مشخص است (JUST – In - Time) کدهای زبانی MSIL فقط هنگامی به زبان محلی قابل فهم برای سیستم عامل تبدیل می‌شوند، که بخواهند اجرا شوند.

در گذشته برای اینکه بتوانید برنامه خود را بر روی سیستم عامل‌های مختلف اجرا کنید که برای هر نسخه از سیستم عامل، آن کد را یک مرتبه به‌طور کامپایل کنید. اما در چارچوب NET. نیازی به این کار نیست، زیرا برای هر نوع پردازنده و نیز هر نوع سیستم عامل یک نسخه از JIT وجود دارد. برنامه شما در هر سیستم عاملی که اجرا شود، کامپایلر JIT موجود در آن سیستم عامل، کد MSIL برنامه‌ی شما را که از مستقل سیستم عامل و نوع پردازنده است دریافت کرده و کد محلی مناسبی تولید می‌کنند که برای سیستم عامل قابل فهم باشد.

فایده استفاده از این روش در این است که وظیفه برنامه‌نویس را به شدت کاهش می‌دهد. در حقیقت می‌توان گفت که به‌عنوان برنامه‌نویس، هنگام نوشتن کد می‌توانید سیستم عاملی که قرار است برنامه وی اجرا شود را فراموش کرده و فکر خود را بر روی کد و منطق برنامه متمرکز کنید.

^۸ . Microsoft Intermediate Language

۲-۵- اسمبلی‌ها

هنگامی که یک برنامه را کامپایل می‌کنید، کد MSIL تولید شده در فایل‌هایی به نام اسمبلی^۹ ذخیره می‌شوند. فایل‌های اسمبلی می‌توانند شامل برنامه‌هایی باشند که بدون نیاز به برنامه‌ای دیگر بتوانند بر روی سیستم عامل اجرا شوند. (این‌گونه فایل‌ها دارای پسوند exe) هستند و یا شامل کتابخانه‌هایی از کلاسها و توابع برای استفاده در دیگر برنامه‌ها باشند (این‌گونه فایل‌ها دارای پسوند dll هستند)

فایل‌های اسمبلی علاوه بر کدهای MSIL شامل اطلاعات متا^{۱۰}

(اطلاعاتی راجع به اطلاعات ذخیره شده در فایل اسمبلی) و همچنین منابع اختیاری (اطلاعات اضافی که به وسیله کدهای MSIL استفاد می‌شوند، همانند فایل‌های صوتی و یا فایل‌های تصویری) نیز هستند. اطلاعات متا باعث می‌شوند که یک فایل اسمبلی بتواند اطلاعات داخل خود را به‌طور کامل توصیف کند. به عبارت دیگر برای استفاده از یک اسمبلی به هیچ اطلاعات و یا کارهای اضافی مانند ثبت آن در رجیستری سیستم نیازی ندارید، به این ترتیب از مشکلاتی که عموماً هنگام کار با این نوع فایل‌ها در محیط‌های دیگر به وجود می‌آمد نیز جلوگیری می‌شود.

یکی دیگر از خاصیت‌های این مورد در این است که توزیع یک نرم‌افزار به سادگی کپی کردن تمام فایل‌های آن بر روی کامپیوتر مقصد است. به علت اینکه برای اجرای یک فایل اسمبلی به هیچ مورد دیگری نیاز نیست، می‌توانید به سادگی فولدر حاوی برنامه را بر روی کامپیوتر مقصد کپی کنید و سپس با کلیک کردن بر روی فایل اجرایی آن، برنامه را اجرا کنید و از آن استفاده کنید (با فرض اینکه CLR که مهمترین بخش NET است. قبلاً در آن کامپیوتر نصب شده باشد). در مورد چگونگی توزیع یک نرم‌افزار در فصل بیست و یکم بیشتر صحبت خواهیم کرد.

البته ممکن است در بعضی مواقع بخواهید از توابع موجود در یک فایل DLL، در چند برنامه استفاده کنید. برای این کار لازم نیست فایل مذکور را در فولدر تمام برنامه‌هایی که از آن استفاده می‌کنند قرار دهید. بلکه می‌توانید آن را یک بار در یک مکان مشخص قرار دهید و سپس تمام برنامه‌هایی که به آن نیاز دارند از آن استفاده کنند. در چارچوب NET، این مکان مشخص که برای قرار گرفتن فایل‌های اسمبلی عمومی در نظر گرفته شده است، Global Assembly Cache و یا GAC نام دارد، برای اینکه یک فایل اسمبلی را در این قسمت قرار دهید، کافی است به سادگی فایل مورد نظر را در فولدر مشخص شده برای

^۹ . مفهوم اسمبلی در این کتاب کاملاً با زبان برنامه‌نویسی اسمبلی تفاوت دارد.

^{۱۰} . Metadata

GAC کپی کنید زیرا برنامه‌ها می‌توانند علاوه بر اسمبلی‌های خود به همه اسمبلی‌های موجود در این فولدر نیز دسترسی داشته باشند.

کدهای مدیریت شده :

همانطور که گفتیم برنامه‌ی شما هنگام کامپایل ابتدا به کد MSIL تبدیل می‌شود، سپس این کد قبل از اجرا به وسیله JIT، به کد محلی تبدیل شده و کد محلی به وسیله سیستم عامل اجرا می‌شود. تمام این قسمت‌ها بخشی از وظایف CLR است. اما وظایف CLR به این موارد ختم نمی‌شود. برنامه‌ای که به وسیله NET نوشته شده است در طول زمان اجرا توسط CLR مدیریت می‌شود. به عبارت دیگر در طول اجرای برنامه‌های نوشته شده CLR.NET مسئول کنترل امنیت آنها، مدیریت حافظه برنامه‌ها، کنترل بخش‌های خطایابی در برنامه‌ها و... است.

به همین دلیل به برنامه‌هایی که با NET نوشته شده‌اند، برنامه‌های مدیریت شده می‌گویند. در مقابل، برنامه‌هایی که تحت کنترل CLR اجرا نمی‌شوند به برنامه‌های مدیریت نشده¹¹ معروف هستند و زبانهای مشخصی مانند C++ می‌توانند چنین برنامه‌هایی را تولید کنند. از کدهای مدیریت نشده بیشتر در مواقعی استفاده می‌شود که قابلیت استفاده از کدهای مدیریت شده نباشد، همانند فراخوانی توابع سطح پایین سیستم عامل، البته با استفاده از زبان C# نمی‌توان کدهای مدیریت نشده تولید کرد و تمام کدهای تولید شده به وسیله کامپایلر C# تحت کنترل CLR اجرا می‌شوند.

۲-۶- مدیریت حافظه در NET

یکی از مهمترین ویژگیهای کدهای مدیریت شده، بخش مدیریت حافظه در این نوع کدها است که به وسیله سیستمی به نام Garbage Collection و یا به اختصار GC انجام می‌شود. چارچوب NET با استفاده از این سیستم می‌تواند اطمینان حاصل کند که حافظه‌ای که به یک برنامه اختصاص داده می‌شود با پایان برنامه به‌طور کامل بازیابی می‌شود. در زبانهای برنامه‌نویسی قبل از NET، این مورد بوسیله برنامه‌نویس کنترل می‌شود و امکان داشت که با یک اشتباه کوچک در کد برنامه، مقدار زیادی از فضای حافظه غیرقابل استفاده بماند و برنامه با کمبود حافظه مواجهه شود. اینگونه مشکلات باعث کاهش سرعت برنامه‌ها و حتی در بعضی شرایط باعث توقف سیستم می‌شد. نحوه کار GC در NET، به این صورت است که در زمانهای مشخصی به بررسی حافظه می‌پردازد و داده‌هایی را که دیگر استفاده نمی‌شوند از حافظه پاک می‌کند.

¹¹ . Unmanaged Code

البته بررسی حافظه توسط GC در فاصله‌های زمانی ثابت صورت نمی‌گیرد بلکه ممکن است در شرایطی در هر ثانیه چندین هزار بار اجرا شود در شرایط دیگر در هر چند ثانیه یک بار اجرا شود.

۲-۷- مراحل اجرای برنامه در NET

قبل از ادامه، مراحل لازم برای ایجاد یک برنامه با NET را قسمتهایی قبلی توضیح داده شد جمع‌بندی می‌کنیم:

- (۱) کد برنامه به وسیله یکی از زبانهای سازگار با NET مانند C# نوشته می‌شود. (شکل ۱-۲)
- (۲) این کد به زبان MSIL کامپایل می‌شود و سپس در یک فایل اسمبلی ذخیره می‌شود. (شکل ۲-۲).
- (۳) هنگامی که کد بخواهد اجرا شود (چه خود فایل اجرایی شود، چه یک فایل حاوی توابع مورد استفاده باشد و به وسیله دیگر برنامه‌ها احضار شود) ابتدا باید به وسیله یک کامپایلر دیگر به کد محلی تبدیل شود. این کامپایلر JIT نام دارد (شکل ۳-۲)
- (۴) کد محلی تولید شده به وسیله JIT به همراه دیگر برنامه‌های در حال اجرا که به وسیله NET نوشته شده‌اند، تحت کنترل CLR به اجرا در می‌آید. (شکل ۴-۲)

لینک دادن:

در تکمیل گفته‌های قبلی، فقط یک بخش دیگر باقی مانده است. کد C# یک برنامه که در اولین مرحله به زبان MSIL تبدیل می‌شود حتماً نباید در یک فایل باشد، بلکه می‌توانیم برنامه را در چندین فایل سورس کد قرار دهیم و سپس آنها را در یک فایل اسمبلی کامپایل کنیم. به این عمل لینک کردن گفته می‌شود که در برنامه‌ها کاربرد زیادی دارد. فایده این روش در این است که معمولاً برای برنامه‌نویس کار یا چند فایل کوچک راحت‌تر از کار با یک فایل بزرگ است. برای مثال می‌توانید سورس یک برنامه را به چندین فایل مجزا تقسیم کنید و سپس به‌طور جداگانه بر روی هر یک از آنها کار کنید. به این ترتیب در مواقع مورد نیاز، پیدا کردن قسمت خاصی از کد نیز بسیار راحت‌تر خواهد شد. یک دیگر از قابلیت‌های این روش در این است که گروه‌های برنامه‌نویسی می‌توانند یک برنامه را به چند قسمت تقسیم کنند. به این ترتیب هر کدام از برنامه‌نویسان می‌توانند بر روی یک قسمت خاص کار کنند بدون اینکه در مورد نحوه پیشرفت قسمتهای دیگر نگران باشند.

۲-۸- C# چیست؟

همانطور که در قسمتهای قبلی نیز ذکر شد C# یکی از زبانهای برنامه‌نویسی است که به‌وسیله آن می‌توان برنامه‌هایی با قابلیت اجرا در .NET CLR تولید کرد. زبان C# در حقیقت نسخه کامل شده‌ی زبانهای برنامه‌نویسی C و C++ که به‌وسیله مایکروسافت برای کار با چارچوب .NET بوجود آمده است. با توجه به جدید بودن این زبان برنامه‌نویسی، در ایجاد آن سعی شده است که از ویژگی‌های خوب زبانهای برنامه‌نویسی دیگر الهام گرفته شود و نیز کاستی‌های آن زبان‌ها برطرف شود.

ایجاد یک برنامه در محیط C# بسیار راحت‌تر از ایجاد یک برنامه در محیط C++ است. علاوه بر این سادگی، C# زبان قدرتمندی نیز محسوب می‌شود به‌نحوی که اغلب کارهایی که در C++ امکان‌پذیر است در C# هم می‌توان انجام داد. بعضی از ویژگی‌های C# که هم‌سطح با ویژگی‌های پیشرفته C++ هستند، همانند قابلیت دسترسی مستقیم به حافظه و نیز تغییر آن، باعث می‌شوند که کدهای یک برنامه به‌عنوان کد نامن در نظر گرفته شود. استفاده از این تکنیک‌های پیشرفته‌ی برنامه‌نویسی، عموماً خطرناک هستند زیرا ممکن است باعث شوند قسمت‌های مهم حافظه که اطلاعات سیستم عامل در آن قرار دارد به‌طور ناخواسته تغییر کند و سیستم متوقف شود. به‌همین دلیل این مباحث در این کتاب مورد بررسی قرار نمی‌گیرد.

بعضی مواقع کدهای زبان C# طولانی‌تر از کدهای زبان C++ هستند. علت این طولانی بودن کدها به خاطر این است که C# بر خلاف C++ یک زبان نوع - امن^{۱۲} است و در اصطلاح این لغت به‌معنی این است که هنگامی که نوع داده‌ای یک متغیر مشخص شد، آن متغیر نمی‌تواند به یک نوع داده‌ای دیگر که در آن مرتبط نیست تبدیل شود. علاوه بر این مورد یک‌سری محدودیت‌های دیگر نیز هنگام تبدیل یک نوع داده‌ای به نوع داده‌ای دیگر نیز وجود دارد که باعث می‌شود کدهای C# طولانی‌تر از کدهای C++ شوند، اما در مقابل کدهای C# از پایداری بیشتری برخوردارند و نیز خطایابی در آنها ساده‌تر است. البته C# فقط یکی از زبانهای برنامه‌نویسی است که برای طراحی برنامه تحت .NET ایجاد شده است، اما به نظر من مطمئناً بهترین زبان برای این کار است. یکی از دلایل این امر این است که زبان C# از پایه برای استفاده در محیط .NET ایجاد شده است و معمولاً در پروژه‌هایی که در رابطه با انتقال .NET به سیستم عامل‌های دیگر است، مانند Mono از این زبان استفاده می‌کنند. در زبان‌های دیگر، مانند نسخه .NET. زبان ویژوال بیسیک، برای اینکه شباهت با نسل‌های قبلی خود را حفظ کنند، یکسری از قسمتهای CRL پشتیبانی نمی‌شود. در مقابل با استفاده از زبان C# می‌توان از تمام ویژگی‌های ارائه شده به‌وسیله .NET در برنامه استفاده کرد.

¹². Type - safe

چه نوع برنامه‌هایی را می‌توان با استفاده از C# انجام داد؟

همانطور که گفتیم، در NET هیچ محدودیتی برای نوع برنامه‌های قابل اجرا وجود ندارد، زبان C# نیز از چارچوب NET استفاده می‌کند، بنابراین هیچ محدودیتی در نوع برنامه‌هایی که می‌توان با این زبان انجام داد وجود ندارد، اما بیشتر برنامه‌هایی که با C# نوشته می‌شوند جزء یکی از دسته برنامه‌های زیر هستند:

برنامه‌های مبتنی بر ویندوز: این نوع برنامه‌ها همانند office برنامه‌هایی هستند که دارای ظاهر آشنای برنامه‌های ویندوزی هستند. این نوع برنامه‌ها به وسیله فضای نام مربوط به برنامه‌ای ویندوزی در چارچوب NET نوشته می‌شوند. این فضای نام شامل کنترل‌هایی از قبیل دکمه‌های فرمان، نوار ابزارها، منوها و ... است که بوسیله آنها می‌توان رابط گرافیکی برنامه را طراحی کرد.

برنامه‌های مبتنی بر وب: این نوع برنامه‌ها شامل یکسری صفحات وب هستند که ممکن است تاکنون به وسیله مرورگرهای مختلف اینترنت آنها را مشاهده کرده باشید. چارچوب NET دارای یک سیستم قوی برای ایجاد اتوماتیک صفحات وب و تامین امنیت آنها و ... است. این سیستم ASP.NET نامیده می‌شود و شما می‌توانید با استفاده از زبان C# و سیستم NET.¹³ ASP برنامه‌هایی مبتنی بر وب ایجاد کنید.

سرویس‌های وب: وب سرویس‌ها یک روش جدید و جالب برای ایجاد برنامه‌های توزیع شدنی مبتنی بر وب هستند. با استفاده از وب سرویس‌ها می‌توانید هر نوع اطلاعاتی را از طریق اینترنت بین برنامه‌ها منتقل کنید. در این مورد زبان مورد استفاده در برنامه و یا سیستم عاملی که برنامه در آن اجرا می‌شود اهمیتی ندارد. به عبارت دیگر برنامه شما که تحت NET و سیستم عامل ویندوز نوشته شده است می‌تواند با برنامه‌هایی دیگر که تحت سیستم عامل‌های دیگر عمل می‌کند تبادل اطلاعات داشته باشد. در هر کدام از این برنامه‌هایی که در بالا ذکر شد ممکن است به دسترسی به یک بانک اطلاعاتی نیاز پیدا کنید. برای این منظور در NET باید از سیستمی به نام NET.¹⁴ ADO استفاده کنید.

¹³ . Active server pages . NET

¹⁴ . Active Data objects.NET

۲-۹- ویرال استودیو 2005

در طراحی برنامه‌های NET. استفاده از ویرال استودیو موردی ضروری نیست. اما با استفاده از آن سرعت طراحی برنامه‌ها افزایش شدیدی پیدا می‌کند. برای نوشتن برنامه با استفاده از C# تحت NET. می‌توان حتی از یک ویرایشگر ساده‌ی متن مانند Notepad نیز استفاده کنید و سپس با استفاده از کامپایلر خط - فرمان NET. برای C# آن را به یک برنامه قابل اجرا تبدیل کنید.

در زیر ویژگی‌هایی از ویرال استودیو NET. که باعث می‌شود این محیط انتخابی مناسب برای برنامه‌نویسی تحت NET. محسوب شود را بررسی خواهیم کرد:

ویرال استودیو تمام مراحل کامپایل یک سورس کد به یک برنامه قابل اجرا به صورت اتوماتیک انجام می‌دهد و همچنین به برنامه‌نویس اجازه می‌دهد هر قسمتی را که بخواهد تغییر داده و تنظیم کنند.

ویرایشگر کد ویرال برای کد نویسی زبان‌های پشتیبانی شده در NET. بسیار هوشمند است و می‌تواند هنگام نوشتن این کدها خطاهای آنها را تشخیص داده و در تصحیح آنها به برنامه‌نویس کمک کند.

ویرال استودیو شامل محیط‌هایی برای طراحی برنامه‌های ویندوزی و نیز برنامه‌های مبتنی بر وب است که به کم آنها می‌توانید به سادگی محیط برنامه خود را طراحی کنید.

برای اینکه در NET. یک برنامه تحت ویندوز ایجاد کنید باید مقدار زیادی کد را که در اغلب برنامه‌ها به صورت تکراری هستند بنویسید این مورد در اجرای برنامه‌هایی از نوع‌های دیگر مانند برنامه‌های تحت وب نیز وجود دارد ویرال استودیو با نوشتن اتوماتیک این کدها در سرعت بخشیدن به طراحی برنامه‌ها کمک قابل توجهی می‌کند.

ویرال استودیو دارای ابزارهای زیادی است که بسیاری از کارهای عمومی را برای شما انجام داده و کد مربوط به آنها را در برنامه قرار می‌دهد به این ترتیب دیگر نیازی نیست در مورد نحوه نوشتن کد آنها نگران باشید.

ویرال استودیو دارای ابزارهای قدرتمندی برای کنترل قسمت‌های مختلف یک پروژه از قبیل سورس C# و با فایل‌های مورد نیاز برنامه از قبیل فایل‌های صوتی و تصویری است.

علاوه بر سادگی طراحی برنامه‌ها در NET. توزیع آنها نیز بسیار ساده است و به راحتی می‌توان آن را بر روی کامپیوترهای مقصد اجرا کرد و یا به روز رساند.

ویژوال استودیو دارای ابزارهای قوی خطایابی در برنامه است. برای مثال با استفاده از این ابزارها می‌توان برنامه را خط به خط اجرا کرد و در اجرای هر خط موقعیت برنامه را بررسی کرد.

ویژگی‌های ویژوال استودیو بسیار بیشتر از موارد ذکر شده است اما همین مقدار برای نمایش مزایای استفاده از آن کافی به نظر می‌رسد.

راه‌حل‌های ویژوال استودیو:

هنگامی که بخواهید یک برنامه را با ویژوال استودیو بنویسید ابتدا باید یک راه‌حل ایجاد کنید یک راه‌حل در اصطلاح ویژوال استودیو، از بیش از یک پروژه تشکیل می‌شود. راه‌حل‌ها می‌توانند شامل چندین پروژه از انواع مختلف باشند، برای مثال تصور کنید می‌خواهید برنامه‌ای برای یک شرکت تجاری بنویسید که از دو قسمت تشکیل می‌شود: در قسمت اول باید یک برنامه تحت ویندوز ایجاد کنید که امور مختلف آن شرکت را کنترل کند، در قسمت دوم نیز باید یک برنامه تحت وب ایجاد کنید تا اطلاعات مربوط به آن شرکت را در یک وب سایت نمایش دهد. برای هر کدام از این قسمت‌ها به یک پروژه مجزا نیاز دارید بنابراین در کل باید دو پروژه ایجاد کنید. برای در یک گروه قرار دادن این دو پروژه می‌توانید از راه‌حل‌ها استفاده کنید، به این ترتیب می‌توانید کدهای مرتبط به هم را در یک جا گروه‌بندی کنید، حتی اگر پروژه‌های آنها در قسمت‌های مختلف هارددیسک باشند و هنگام کامپایل چندین فایل اسمبلی مختلف در قسمت‌های متفاوت هارددیسک ایجاد شود.

۲-۱۰- مزایای Ado.Net

شرکت مایکروسافت در ویژوال بیسیک ورژن 3 که در سال 1993 ارائه داد از Jet که هسته اصلی MS Access بود برای ایجاد ارتباط با بانکهای اطلاعاتی استفاده نمود البته برای استفاده بهتر برنامه‌نویسان، Jet را در غالب مجموعه object‌هایی تحت عنوان (Data Access object) آورد. در نسخه 4 و 5 ویژوال بیسیک، مایکروسافت روشی دیگر برای ارتباط با بانکهای اطلاعاتی تحت عنوان Remote Data object = RDO را مطرح ساخت این روش برای برنامه‌نویسی Client / Server روش مناسبی بود.

در دوم سپتامبر سال 1998 وقتی مایکروسافت ویژوال بیسیک ورژن 6 را ارائه داد روش جدیدی تحت عنوان (ActiveX Data object) ADO را مطرح ساخت در این روش که پایه و اساس آن OLEDB بود مایکروسافت برخلاف RDO، DAO که ساختاری پیچیده و سلسله مراتبی داشتند، از ساختاری ساده و مجزا (غیرسلسله مراتبی) استفاده کرد و در واقع ADO

هم شامل object هائی است که برنامه‌نویس بتواند از آنها برای ایجاد ارتباط با بانکهای اطلاعاتی و انجام عملیات روی آنها استفاده کند و در تاریخ 13 February سال 2002 که میکروسافت نسخه نهایی Visual Basic.Net را ارائه داده روشی جدید برای کار با بانکهای اطلاعاتی تحت عنوان ADO.Net را آورده است. این نسخه از ویژوال بیسیک بر خلاف نسخه‌های قبل بطور کامل مباحث OOP را پشتیبانی می‌کند)

(FULL OOP در واقع ADO.Net هم مانند ADO , RDO , ADO شامل ساختاری برای ارتباط و انجام عملیات روی بانکهای اطلاعاتی است ADO.net همانند DAO , RDO و بر خلاف ADO دارای ساختار سلسله مراتبی می‌باشد. در این مقابله می‌خواهیم به مقایسه ADO و ADO.net پردازیم ADO. در ورژنهای مختلفی در این چند سال اخیر به بازار آمده است با آمدن SQL Server 2000 ورژن جدید ADO یعنی ورژن 2.6 از این محصول ارائه شد و اکنون هم ADO 2.6 در سایت میکروسافت قابل Download می‌باشد.

آخرین ورژن ADO شامل 9 تا object است که عبارتند از:

Connection
Command
Recordset
Parameter
Field
Error
Property
Record
Stream

آبجکت Connection امکان ارتباط با Data Source که شامل بانک اطلاعاتی است را فراهم می‌سازد. بعنوان مثال اگر بخواهیم از آبجکت Recordset برای اضافه و یا حذف و یا تغییر در محتوای رکوردی استفاده کنیم این آبجکت از Connection برای ایجاد ارتباط با بانک اطلاعاتی مثلاً SQL Server استفاده می‌کند اما همانطور که گفته شد و در شکل ملاحظه می‌شود ساختار آبجکتهای ADO بصورت سلسله مراتبی نیست و می‌توان مثلاً Recordset ای ایجاد نمود که مستقل از آبجکت Connection بتواند با بانک اطلاعاتی ارتباط برقرار نماید.

در ADO علاوه بر آبجکتهای، چهار Collection هم دیده می‌شود که عبارتند از:

Parameters

Fields
Properties
Errors

که هر کدام از آنها شامل آبجکتهایی از همان نوع هستند بعنوان مثال ساختاری که برای Recordset کشیده شده بیان کننده آن است که این آبجکت شامل Collection های Fields و Properties بوده و مثلاً Collection مربوط به Fields شامل آبجکتهای فیلد است.

آبجکت فیلد اطلاعاتی را راجع به یک ستون از Recordset در خود نگه داشته است.

در ADO آبجکت Command هم وجود دارد که از آن می توان برای اجرای یک فرمان SQL استفاده نمود البته بهترین روش برای اجرای Stored Procedure نیز استفاده از متد Execute مربوط به همین آبجکت است.

از آبجکت Parameter هم برای مشخص کردن مقادیر پارامترهای ورودی یک Stored Procedure استفاده می کنیم.

آبجکت Property برای استفاده از Dynamic Property استفاده می شود که Property هایی هستند که وابسته به Provider بوده و استاندارد نیستند.

از Record برای دسترسی به یک سطر از Recordset و ویژگیهای مربوط به آن سطر استفاده می کنیم و آبجکت Stream هم به منظور ذخیره و بازیابی اطلاعات بصورت باینری در یک فیلد از Record در نظر گرفته شده است.

تا اینجا با ساختار ADO آشنا شدیم حال به تشریح ADO.net می پردازیم.

ADO.Net مجموعه ای از Class های Interface ها و دستوراتی جهت مدیریت و کار با بانکهای اطلاعاتی است. در بحث

.NET مجموعه ای از Class های مربوط به هم در یک غالب تحت عنوان Name Space ارائه شده است. NET شامل

تعدادی Name Space است که در غالب Net Framework قرار دارد. تمام ساختار ADO.net در چهار

Namespaces قرار دارد که عبارتند از:

System.Data.SqlClient.

System.Data.OleDb.

System.Data.Odbc.

System.Data.

اگر بخواهیم از بانکهای SQL Server استفاده کنیم System.Data.SqlClient بهترین انتخاب است در این NameSpace امکاناتی فراهم شده تا بتوانیم Application های بنویسیم و در آنها با SQL Server ارتباط برقرار نمائیم بطوریکه برنامه‌ها از Performance بالایی برخوردار باشند.

در صورتی که کاربر بخواهد از بانکهای رابطه‌ای دیگر نظیر Oracle استفاده کند می‌تواند از System.Data.OleDb بهره گیرد.

System.Data.Odbc برای ارتباط با بانکهای اطلاعاتی از طریق ODBC می‌باشد System.Data هم شامل Provider های خاص نظیر DataSet و DataTable است.

ADO.net نیز شامل object ها و Collection هائی است که از مهمترین آنها می‌توان موارد ذیل را شمرد:

OleDbConnection

OleDbCommand

OleDbDataReader

OleDbDataAdapter

SelectCommand

InsertCommand

UpdateCommand

DeleteCommand

DataTable

DataRelation

OleDbConnection آبیجکتی است شبیه به آبیجکت Connection در ADO، که امکان ایجاد ارتباط با بانک را فراهم

می‌سازد این آبیجکت متدهایی برای اجرای فرامین SQL نیز داراست OleDbCommand هم مشابه آبیجکت Command

در ADO می‌باشد.

از این آبیجکت نیز برای اجرای Stored Procedure و اجرای فرامین SQL استفاده می‌شود. آبیجکت بعد

OleDbDataReader است که برای گرفتن اطلاعات از Database استفاده می‌شود. البته لازم به ذکر است Resultset

ای که با این روش ساخته می‌شود بصورت Forwardonly , ReadOnly می‌باشد OleDbDataAdapter. آبیجکت

جدیدی در Ado.net است که خود شامل چهار آبیجکت برای انجام فرامین SQL می‌باشد که عبارتند از:

SelectCommand

InsertCommand

UpdateCommand

DeleteCommand.

که به ترتیب برای واکنشی اطلاعات، اضافه نمودن اطلاعات جدید به داخل بانک، به هنگام سازی اطلاعات و حذف اطلاعات استفاده می شود. این آبجکت به همراه آبجکت Dataset استفاده می شود و برای پر کردن Recordset موجود در Dataset بکار می رود.

آبجکت بعدی Dataset است که مهمترین آبجکت در Ado.net می باشد این آبجکت شامل دو Collection به نام های DataTables و DataRelations است. همانطور که می دانید در ADO هم آبجکت Recordset داشتیم اینجا مجموعه ای از Recordset ها در داخل یک DataTables نگه داشته می شود و می توان بین DataTable های مختلف رابطه نیز ایجاد نمود که این روابط هم داخل یک مجموعه ای با نام DataRelations نگه داشته می شود. در واقع DataTable در Ado.net معادل همان RecordSet در ADO می باشد.

فصل سوم- پیاده سازی پایگاه داده SQL :

۳-۱- تشریح جدول ها (Tables)

• ساختار داده جدول bargiri :

در این جدول اطلاعات فرم بارگیری ثبت می شود.

فیلد های این جدول:

Bar_id : شماره بارنامه.

Com_name : نام شرکت.

Gen_date : تاریخ صدور بارنامه .

Give_time : ساعت صدور یا تحویل بارنامه.

Ezhar_num : شماره اظهارنامه.

Bar_owner : نام صاحب بار.

Vagon_id : شماره واگن.

Kala_type : نوع کالا.

UN_code : کد UN

Source : مبدا

Dest_keshvar : کشور مقصد

Dest_station : ایستگاه مقصد

Bar_weigh : وزن بار

Keraye : کرایه انتقال بار

Keraye_type : نوع پرداخت کرایه

Haghe_tavaghof : حق توقف

hazineTavizBoozhi : هزینه تعویض بوژی

hazineSodurBarname : هزینه صدور بارنامه

hazineGharantine : هزینه قرنطینه

other_pay : سایر هزینه ها

• ساختار داده جدول cancel_ghabz :

این جدول اطلاعات قبض های کنسلی واگن را در خود نگه می دارد.

ghabz_num : شماره قبض کنسلی

Price : مبلغ قبض

req_num : شماره نلمه درخواست

• ساختار داده جدول company_folder :

در این table اطلاعات فرم ثبت شرکت ها نگه داشته میشود. فیلد های این جدول به صورت زیر است:

com_id : کد مخصوص شرکت

com_name : نام شرکت

manager_name : نام مدیر عامل شرکت

activity_field : رشته فعالیت

master_motahari : نام نماینده شرکت در مطهری

master_motahari_pn : شماره تلفن نماینده مطهری

master_motahari_fax : فکس نماینده مطهری

master_motahari_cp : شماره همراه نماینده مطهری

master_mashad : نام نماینده شرکت در مشهد

master_mashad_pn : شماره تلفن نماینده مشهد

master_mashad_fax : فکس نماینده مشهد

master_mashad_cp : تلفن همراه نماینده مشهد

master_sarakhs : نام نماینده شرکت در سرخس

master_sarakhs_pn : شماره تلفن نماینده سرخس

master_sarakhs_fax : فکس نماینده سرخس

master_sarakhs_cp : شماره همراه نماینده سرخس

phon_man : شماره تلفن مدیر عامل شرکت

fax_man : فکس مدیر

fax_master : فکس نماینده

phon_office : شماره تلفن دفتر شرکت

fax_office : فکس دفتر

com_add : آدرس شرکت

validdat_vekalat_motahari : تاریخ خاتمه اعتبار و کالت محضری نماینده مطهری

validdat_vekalat_mashad : تاریخ خاتمه اعتبار و کالت محضری نماینده مشهد

validdat_vekalat_sarakhs : تاریخ خاتمه اعتبار و کالت محضری نماینده سرخس

validdate_card : تاریخ خاتمه اعتبار کارت بازرگانی مدیر عامل شرکت

chek_num : شماره چک ضمانت واردات/صادرات

chek_some : مبلغ چک ضمانت واردات/صادرات

account_owner : صاحب حساب چک ضمانت واردات/صادرات

account_number : شماره حساب چک ضمانت واردات/صادرات

bank_name : نام بانک چک ضمانت واردات/صادرات

bank_shahr : شهر بانک

bank_shobe : نام شعبه بانک چک ضمانت واردات/صادرات

• ساختار داده جدول enter_vagon :

این جدول برای ثبت اطلاعات واگن های ورودی می باشد. فیلد های آن به صورت زیر است:

vagon_id : شماره واگن

enter_date : تاریخ ورود

vagon_state : وضعیت واگن (تعمیری / سالم)

fix_date : تاریخ خاتمه تعمیر

• ساختار داده جدول واگذاری واگن (give_vagon):

فیلد های این جدول به صورت زیر است:

current_mojoodi : آخرین موجودی مالی شرکت

vagon_id : شماره واگن

vagon_type : نوع واگن

req_num : شماره نامه درخواست

give_date : تاریخ واگذاری

give_time : ساعت واگذاری

cancel_req_num : شماره نامه درخواست کنسلی

cancel_req_date : تاریخ تقاضای کنسلی

cancel_req_time : ساعت تقاضای کنسلی

com_id : کد مخصوص شرکت

Cancel : آیا کنسل شده

• ساختار داده جدول ورود کاربران به برنامه (LogIntoProgram) :

در این جدول لیستی از کاربران که حق استفاده از برنامه را دارند تعریف شده است. فیلدهای این جدول به صورت زیر است:

Username : نام کاربری

Password : کلمه عبور

FirstName : نام کاربر

LastName : نام خانوادگی کاربر

• ساختار داده جدول درخواست های واگن (vagon_req):

هر شرکت که درخواست واگن میدهد اطلاعات درخواستی آن شرکت در این جدول نگه داشته می شود که فیلدهای آن به

صورت زیر است:

com_id : کد مخصوص شرکت

com_name : نام شرکت

req_num : شماره نامه درخواست

date_reqform : تاریخ فرم درخواست

time_reqform : ساعت فرم درخواست

vagon_type : نوع واگن

kala_type : نوع کالا

rec_con : کشور مقصد

rec_station : ایستگاه مقصد

date_req : تاریخ درخواست جهت واگذاری

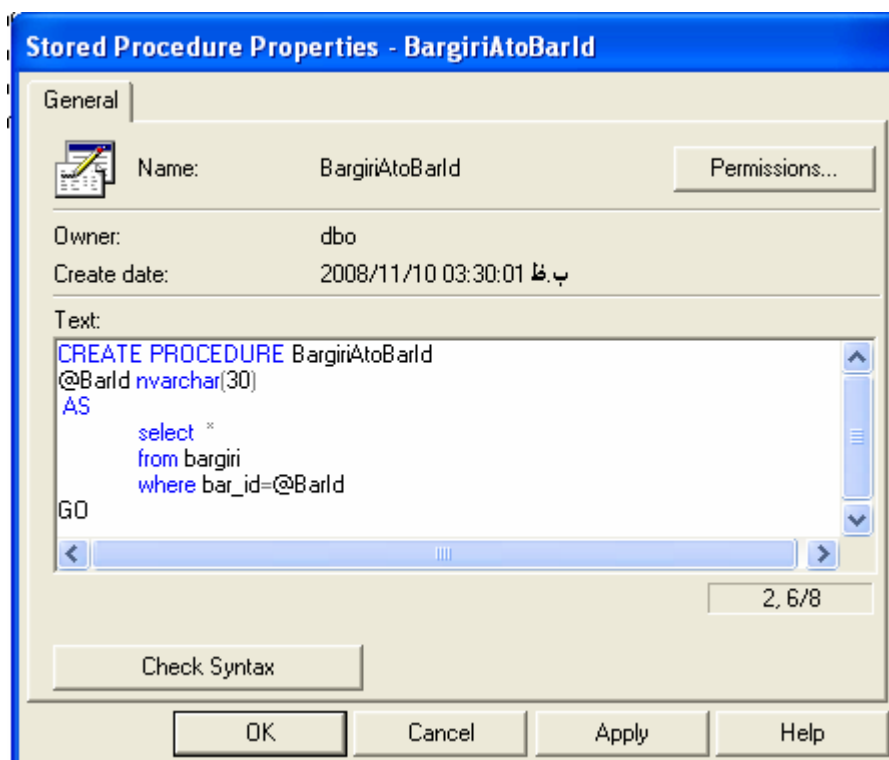
Description : توضیحات

Vagozari : واگذاری شده یا خیر

Enteghali : انتقالی داشته یا خیر

۳-۲- تشریح روال های ذخیره شده (Stored Procedures)

• BargiriAtoBarId :

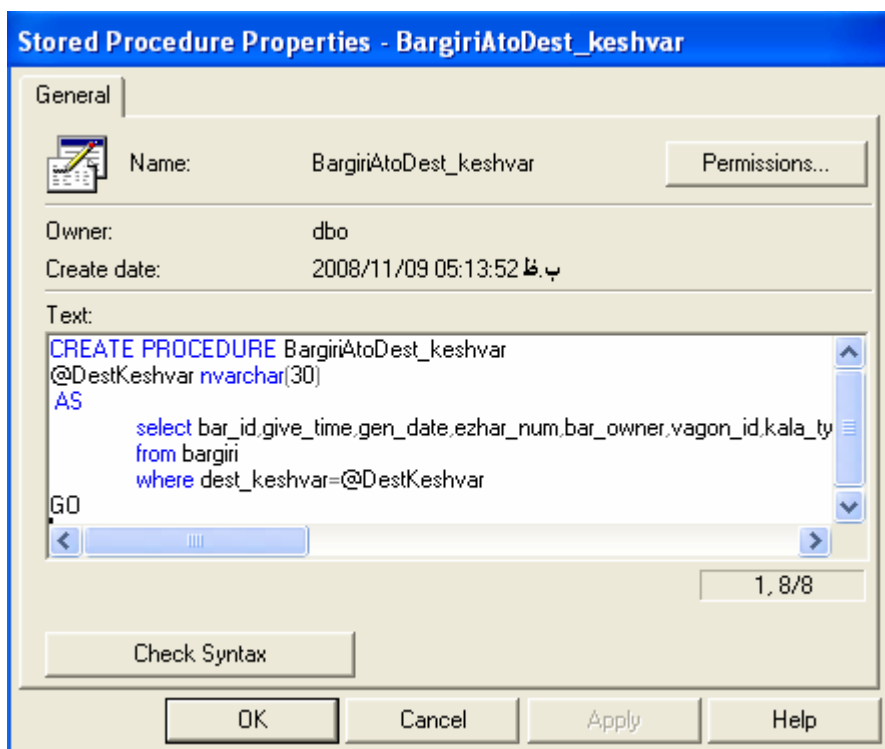


شکل ۳-۱- روال bargiriAtoBaId

این روال رکورد های جدول bargiri را بر حسب شماره بارنامه انتخاب می کند. شماره بارنامه پارامتر ورودی این روال می باشد.

• **BargiriAtoDest_keshvar** :

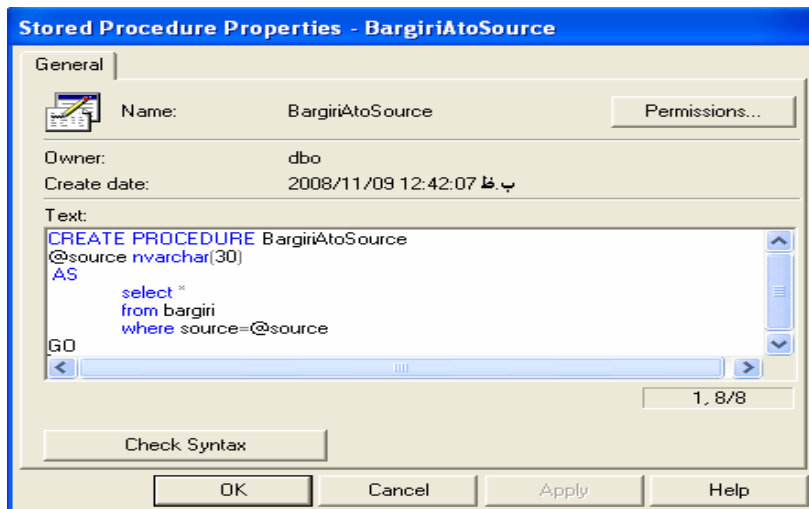
این روال هو مانند روال قبلی است با این تفاوت که رکوردها را بر مبنای کشور مقصد انتخاب میکند که کشور مقصد پارامتر ورودی میباشد .



شکل ۳-۲- روال بارگیری بر مبنای کشور مقصد

• **BargiriAtoSource** :

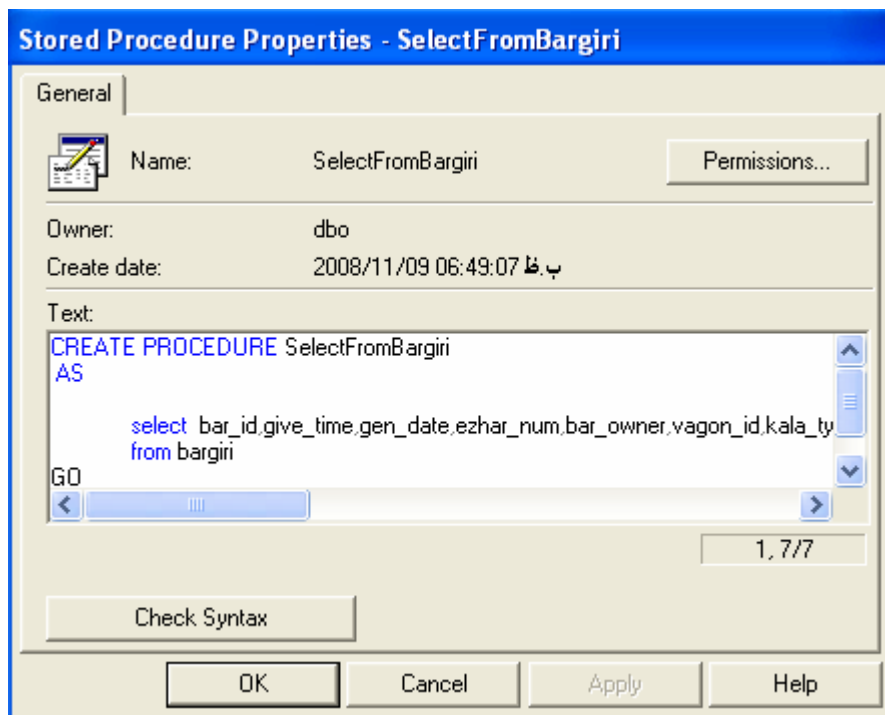
این روال هو مانند روال قبلی است با این تفاوت که رکوردها را بر مبنای مبدا انتخاب میکند که مبدا پارامتر ورودی میباشد . در این روال تمام ستون ها به عنوان خروجی انتخاب می شوند.



شکل ۳-۳- روال بارگیری بر مبنای مبدا

• **:SelectFromBargiri**

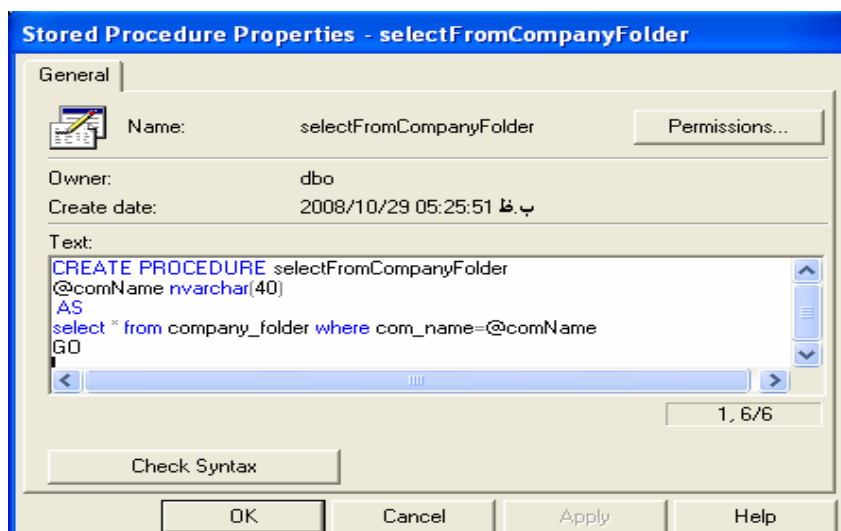
این روال تمام رکورد های جدول بارگیری را انتخاب می کند.



شکل ۳-۴- روال selectFromBargiri

• **:selectFromCompanyFolder**

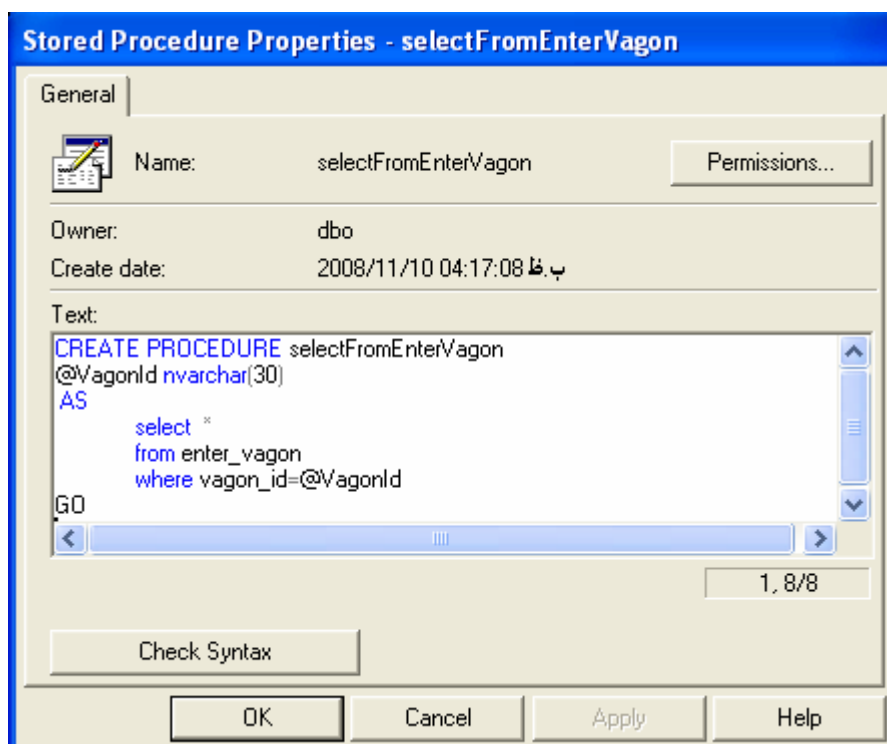
در این روال تمام رکوردها بر حسب نام شرکت انتخاب می شود.



شکل ۳-۵- روال selectFromCompanyFolder

• :selectFromEnterVagon

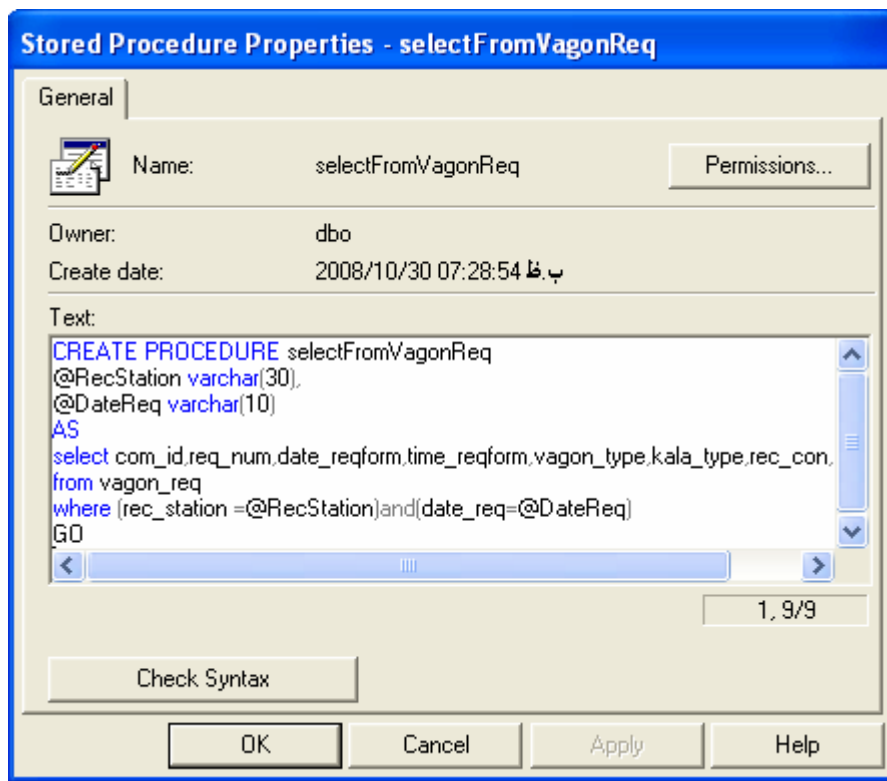
در این روال رکوردهای جدول enter_vagon بر حسب شماره واگن که پارامتر ورودی است انتخاب میشود.



شکل ۳-۶- روال selectFromEnterVagon

• :selectFromVagonReq

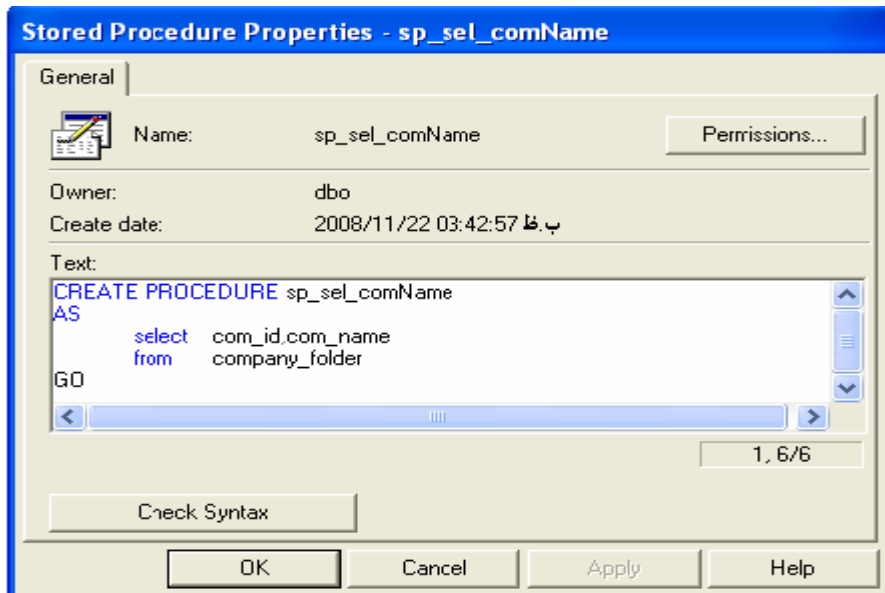
این روال برای انتخاب رکوردهای جدول vagon_req استفاده شده است که رکوردها بر حسب دو پارامتر ورودی ایستگاه مقصد و تاریخ درخواست جهت واگذاری می باشد. این دو پارامتر به StoredProcedure پاس داده می شود.



شکل ۳-۷- روال selectFromVagonReq

• sp_sel_comName :

از این روال برای comboBox نام شرکت در فرمها استفاده میشود. به این صورت که فیلد com_id و com_name را از داخل جدول company_folder استخراج می کند.



شکل ۳-۸- روال sel_sel_comName

۳-۳- تشریح Defaults

- Description :

این Defaults برای فیلد Description در جدول vagon_req تعبیه شده است که در صورتی که در این فیلد داده ای وارد نشود مقدار واقع در این default در این فیلد قرار میگیرد.

- ForBit :

این Default برای فیلدهای از نوع Bit تعبیه شده است و در صورتی که در این فیلدها مقداری قرار نگیرد مقدار صفر را به صورت خودکار در داخل فیلد قرار میدهد.

- otherPay :

این default برای فیلد other_pay تعبیه شده است و در صورتی که مقدار این فیلد وارد نشود مقدار آن را که سایر هزینه ها می باشد صفر قرار میدهد.

۳-۴- تشریح ساختار پایگاه داده امور مالی :

پایگاه داده امور مالی شامل یک جدول است که برای بررسی وضعیت مالی هر شرکت که شروع به فعالیت می کند استفاده می شود و نام این پایگاه داده Omur_Mali می باشد.

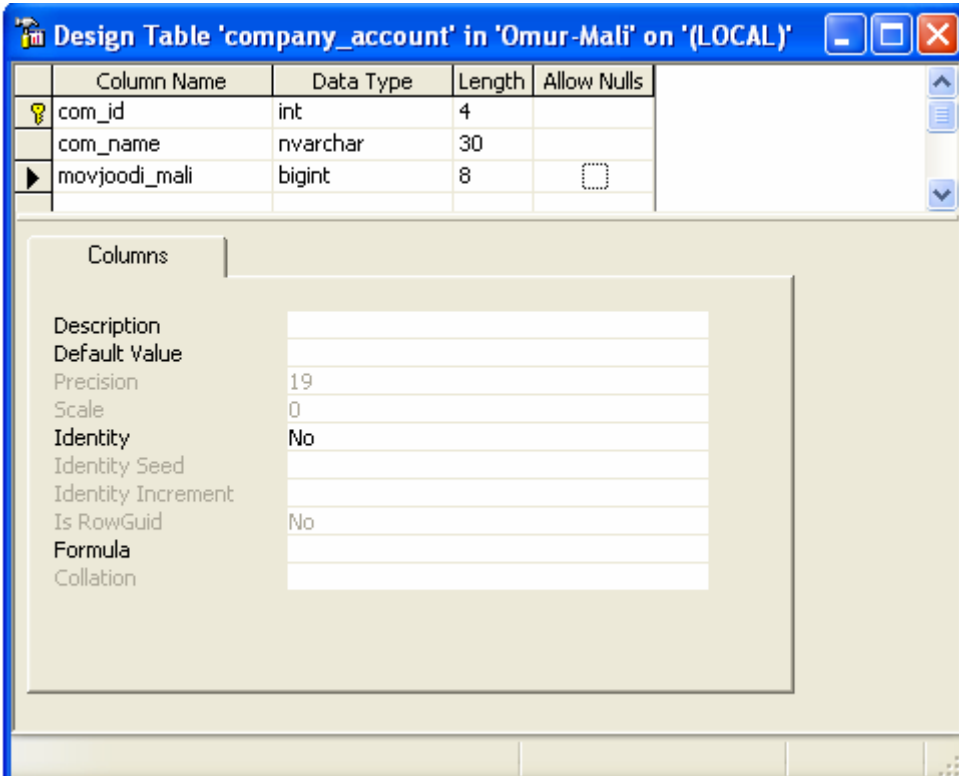
ما از این پایگاه داده در دو مورد استفاده می کنیم:

۱. بررسی آخرین موجودی مالی هر شرکت.

۲. برداشتن کد مخصوص هر شرکت از این پایگاه داده.

کد مخصوص هر شرکت باید بین اداره بازرگانی و اداره امور مالی و بایگانی یکی باشد به همین خاطر است که برای ثبت کد مخصوص هر شرکت به این پایگاه مراجعه می کنیم.

اطلاعاتی که ما نیاز داریم در جدولی به نام company_account قرار گرفته است که ساختار فرضی آن به شکل زیر است:



Column Name	Data Type	Length	Allow Nulls
com_id	int	4	
com_name	nvarchar	30	
movjoodi_mali	bigint	8	<input type="checkbox"/>

Description	
Default Value	
Precision	19
Scale	0
Identity	No
Identity Seed	
Identity Increment	
Is RowGuid	No
Formula	
Collation	

شکل ۳-۹: جدول Company_account

فیلدهای آن شامل سه فیلد زیر است:

com_id: کد مخصوص شرکت

com_name: نام شرکت

فصل چهارم - پیاده‌سازی سیستم

سیستم پیاده‌سازی شده به طور خلاصه امور زیر در آن صورت می‌گیرد:

۱- ثبت اطلاعات ۲- گزارشگیری ۳- چاپ ۴- جستجو

۴-۱- طراحی فرم های سیستم

۴-۱-۱- فرم اصلی برنامه (frmmain):

این فرم شامل یک منو طراحی شده است که دسترسی به تمام فرم ها از همین منو صورت می‌گیرد.



شکل ۴-۲: فرم اصلی برنامه

Item های منو:

- ثبت اطلاعات: فرم های مربوط به ثبت اطلاعات را باز می‌کند.

گزارشات: فرم مربوط به گزارشات را باز می کند که با انتخاب گزینه از این منو سربرگ مربوط به همان گزینه در فرم گزارشات باز خواهد شد.

جستجو: این گزینه فرم مربوط به جستجو را باز می کند. با انتخاب هر گزینه سربرگ مربوط به همان گزینه در فرم جستجو باز خواهد شد.

تنظیمات کاربر: فرم مربوط به تنظیمات کاربر را باز می کند با انتخاب هر گزینه سربرگ مربوط به همان گزینه در فرم تنظیمات کاربر باز خواهد شد.

زیر منوهای این گزینه:

۱. تعریف کاربر جدید

۲. حذف یک کاربر

۳. تغییر رمز عبور

پنجره مربوط به ورود (Login):

در این پنجره هر کاربر که برای این سیستم تعریف شده باشد می تواند با وارد کردن نام کاربری و کلمه عبور خود وارد برنامه شود اگر کسی که مدیریت است یا از طریق Administrator وارد سیستم شود با زدن تیک مربوط مدیر که زیر دکمه Enter است می تواند وارد نرم افزار شود.

- دکمه ثبت: این دکمه همان کارهای مربوط به آیتم ثبت اطلاعات را انجام می دهد.

با کلیک کردن بر روی این دکمه چهار دکمه دیگر نشان داده خواهد شد:

- دکمه تشکیل پرونده شرکت: که فرم ثبت اطلاعات شرکت را باز می کند.

- دکمه ورودی واگن: که فرم ثبت اطلاعات ورودی واگن را باز می کند.

- دکمه درخواست واگن: که فرم ثبت اطلاعات درخواست واگن را باز می کند.

- دکمه بارگیری و تحویل واگن: که فرم ثبت اطلاعات بارگیری و تحویل واگن را باز می کند.

- دکمه جستجو: فرم جستجو را باز می کند.



- دکمه :

که در سمت چپ پایین صفحه واقع شده است این که تست می کند که آیا برنامه می تواند به شبکه متصل شود
وامکان وصل شدن به Server را دارد یا خیر.



- دکمه :

- این دکمه همان کار LogOff در Windows را انجام می دهد. با زدن این دکمه تمام دکمه ها و منوی
برنامه غیر فعال می شود و هیچ کس بدون وارد کردن نام کاربری و کلمه عبور خود نمی تواند دوباره وارد
برنامه شود. برای امنیت بالاتر نرم افزار برای کاربران تعبیه شده است. از موارد کاربرد این دکمه می توان
گفت زمانی که کاربر زمان کوتاهی از پشت سیستم خود بلند می شود با زدن این دکمه برنامه به حالت
خاموش می رود و تا بازگشت خود کاربر کس دیگر نمی تواند وارد برنامه شود.

۴-۱-۲- فرم ثبت اطلاعات شرکتها :

در این فرم اطلاعات مربوط به شرکتهایی که قرار است فعالیت بازرگانی داشته باشند ثبت می شود.

تشکیل پرونده شرکت

اطلاعات را وارد نمایید و سپس گزینه "ثبت" را کلیک نمایید. برای دریافت کد مخصوص شرکت نام شرکت را وارد نمایید و سپس کلید Enter را بزنید تا سیستم شما را راهنمایی کند. برای حذف یک شرکت نام آن را وارد نمایید و سپس دکمه حذف را کلیک نمایید.

نام شرکت: نام مدیر عامل شرکت:

کد مخصوص شرکت: رشته فعالیت:

نام نماینده شرکت در مطهری:

نام نماینده شرکت در مشهد:

نام نماینده شرکت در سرخس:

تلفن مدیر: تلفن دفتر:

فکس مدیر: فکس دفتر:

آدرس شرکت:

تاریخ خاتمه اعتبار وکالت محضری نماینده مطهری: YYYY/MM/DD

تاریخ خاتمه اعتبار وکالت محضری نماینده مشهد: YYYY/MM/DD

تاریخ خاتمه اعتبار وکالت محضری نماینده سرخس: YYYY/MM/DD

تاریخ خاتمه اعتبار کارت بازرگانی: YYYY/MM/DD

مشخصات چک

شماره چک: شماره حساب: شعبه بانک:

مبلغ چک: نام بانک:

صاحب حساب: شهر بانک:

ثبت شرکت حذف شرکت درخواست ویرایش بازگشت

ب.ا.ب 01:34 Paint - frmMain2 تشکیل پرونده شرکت سیستم ثبت تقاضای و Debug start

شکل ۴-۱: فرم ثبت اطلاعات شرکت

دکمه هایی که در این فرم طراحی شده اند:

- دکمه ثبت: اطلاعات وارد شده را ثبت می کند که در این فرم اکثر اطلاعات مربوط به شرکت باید وارد شود

در غیر این صورت پیغام خطا داده می شود.

مثلاً اطلاعات مربوط به چک و تاریخ ها جزء اطلاعاتی هستند که حتماً برای هر شرکت باید وارد شود.

- دکمه بازگشت: این دکمه فرم باز شده را می بندد.

۴-۱-۳- فرم درخواست های واگن (VagonReg):

شرکت هایی که برای امور بازرگانی قبلا اطلاعات آنها ثبت شده است می توانند برای واردات یا صادرات تقاضای واگن دهند این فرم برای ثبت اطلاعات درخواستی واگن تعبیه شده است که در آن هر شرکت نوع واگن و نوع کالا و مقصد خود را و همچنین تاریخی که می خواهد واگن به آن شرکت واگذار شود را مشخص می کند.

The screenshot shows a web application window titled "درخواست های واگن" (Wagon Requests). The window contains a form with the following fields and controls:

- Text input: نام شرکت را وارد نمایید و سپس کلید اینتر را بزنید تا کد شرکت را ببینید و سپس یک اینتر دیگر بزنید تا اطلاعات کنترلی شرکت بررسی شود.
- Dropdown menu: نام شرکت متقاضی (Azeri)
- Text input: نوع واگن (مسقف)
- Text input: کد شرکت
- Text input: نوع کالا
- Text input: کشور گیرنده
- Text input: ایستگاه مقصد
- Text input: تاریخ درخواست جهت واگذاری
- Text input: شماره ردیف در لیست تقاضا
- Text input: شماره درخواست
- Text input: تاریخ فرم درخواست
- Text input: ساعت فرم درخواست
- Text input: توضیحات
- Buttons: ثبت درخواست, انصراف واگن, انتقال درخواست, واگذاری واگن
- Button: بررسی اطلاعات کنترلی شرکت

شکل ۴-۳: فرم درخواست های واگن

دکمه های فرم درخواست های واگن :

- دکمه بررسی اطلاعات کنترل شرکت

پس از اینکه نام هر شرکت به همراه که مخصوص آن شرکت مشخص شد باردن این دکمه بررسی می شود که آیا شرکت نامبرده اجازه درخواست واگن را دارد یا خیر.

شرکتی می تواند درخواست واگن دهد که دارای شرایط زیر باشد

۱. تمام اطلاعات مربوط به چک شرکت ثبت شده باشد.
۲. تاریخ و کالت محضری نمایندگان نباید ارتاریخ روز درخواست گذشته باشد.

۳. هر شرکت برای درخواست واگن باید ۱۰۰/۰۰۰ تومان موجودی داشته باشد. موجودی مالی شرکت در سیستم امور مالیت شده است که باید آن سیستم متصل شود و با توجه به که مخصوص شرکت که در هر دو سیستم یکتاست چک شود که آیا شرکت موجودی مالی کافی دارد یا خیر.

۴. و آخرین مورد تاریخ کارت بازرگانی شرکت است که باید اعتبار داشته باشد.

در صورتی که در هر کدام از موارد بالا مشکلی بود نباید اجازه ثبت درخواست به کاربر داده شود.

- دکمه ثبت درخواست :

پس از اینکه کاربر بتواند اطلاعات فرم را وارد نماید یعنی شرکت هیچ مشکلی نداشته باشد با زدن دکمه ثبت اطلاعات وارد شده در جدول vagon-Req در پایگاه داده ثبت می شود و پس از ثبت شدن اطلاعات فرم به حالت اول باز می گردد.

- دکمه انصراف واگن:

وقتی کاربر شماره نامه درخواست را وارد نماید و سپس این دکمه را بزند واگنی که قبلا اطلاعات درخواست آن ثبت شده در صورتی که ۲۴ ساعت قبل از تاریخ واگذاری باشد می تواند انصراف داده شود.

- دکمه انتقال درخواست:

در صورتی که ۲۴ ساعت به تاریخ جهت واگذاری مانده باشد با زدن این گزینه فرمی باز می شود که در آن می توان تاریخ جهت واگذاری را تغییر داد.

- دکمه واگذاری واگن:

با زدن این دکمه فرم واگذاری واگن باز شده و واگنی که قبلا درخ. است داده شده است با مشخص شدن شماره واگن به شرکت درخواست دهنده پس از وارد کردن اطلاعات واگذاری واگن واگذار می شود.

- دکمه بازگشت :

با زدن این دکمه فرم درخواست های واگن بسته می شود.

دکمه شماره ردیف در لیست تقاضا: وارد کردن تاریخ واگذاری و زدن این دکمه تمام درخواست هایی که در این تاریخ باید واگذاری شوند را نشان می دهد در فرم Row Number و می گوید مه شماره ردیف این درخواست درایت تاریخ در صورت ثبت چند است.

۴-۱-۴- فرم واگذاری واگن (frm Give Vagon) :

توضیحات:

زمانی که یک درخواست واگن ثبت می شود به هنگام واگذاری این فرم باز می شود که با وارد کردن اطلاعات فرم مثل شماره واگن نوع واگن تاریخ وساعت واگذاری می توان واگنی رابع درخواستی اریک شرکت واگذار کرد. در صورتی که آخرین موجودی مالی شرکت به اندازه لازم نباشد نباید به کاربر اجازه داد که واگنی رابه درخواستی واگذار کند. (با وارد کردن نام شرکت و زدن کلید Enter می توان آخرین موجودی مالی شرکت را مشاهده کرد.

شکل ۴-۴: فرم واگذاری واگن

دکمه های روی فرم :

- دکمه ثبت:

پس از وارد نمودن اطلاعات کلیدها و زدن این دکمه اطلاعات در بانک ثبت می شود (جدول give-vagon) و واگن به شرکت واگذار می شود.

- دکمه کنسل نمودن واگن واگذار شده :

بازدن این دکمه در صورتی که شرکت واگنی را قبلاً درخواست داده باشد و این واگن و گذار شده باشد واگن مورد نظر را کنترل کرده و فرمی را باز می کند که در آن اطلاعات قبض کنسلی وارد می شود. قبض کنسلی هم قبض است که به ازای جریمه کنسل نمودن واگن و گذار شده باید تهیه شود. اطلاعات قبض کنسلی در جدول CancelGhabz وارد می شود.

- دکمه بازگشت:

- فرم واگذاری واگن بسته می شود.

۴-۱-۵- فرم تغییر تاریخ واگذاری :

وقتی در فرم درخواست واگن شماره درخواست را وارد کردیم و سپس گزینه انتقال درخواست را زدیم این فرم نمایش داده می شود که برای تغییر تاریخ جهت واگذاری می باشد. در این فرم فقط فیلر تاریخ جهت واگذاری را می توان تغییر داد و بقیه فیلرها غیر قابل تغییر و فقط خواندنی هستند. پس از تغییر فیلر تاریخ جهت واگذاری گزینه ثبت را می زنیم که باعث تغییر تاریخ در پایگاه داده می شود.

تغییر تاریخ واگذاری	
تاریخ را تغییر دهید و گزینه ثبت را بزنید .	
کد شرکت :	1234
شماره نامه درخواست :	102
تاریخ فرم درخواست :	1387/09/19
زمان فرم درخواست :	13:00
نوع واگن :	مسئف
توضیحات :	
نوع کالا :	طبیبل
کشور مقصد :	بیابیل
ایستگاه مقصد :	سلیب
تاریخ درخواست جهت واگذاری :	1387/09/20

بازگشت ثبت تغییر

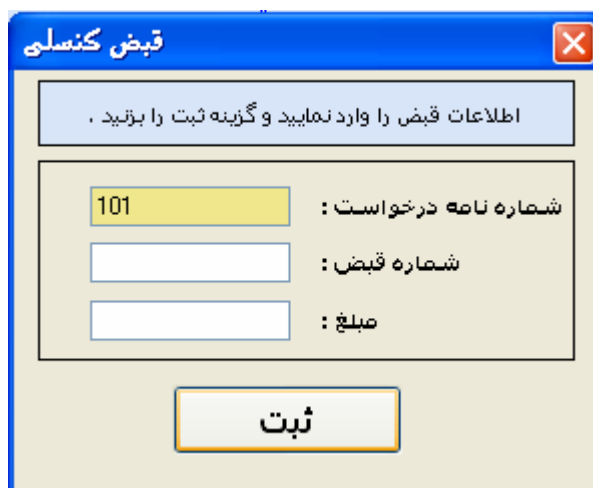
شکل ۴-۵: فرم تغییر تاریخ واگذاری

- دکمه ثبت : تاریخ واگذاری را به تاریخ جدید تغییر می دهد.

- دکمه بازگشت: پنجره یافرم را می بندد.

۴-۱-۶- فرم قبض کنسلی :

هنگامی که یک واگن که به یک شرکت واگذار شده است کنسل می شود این کنسلی شامل یک قبض کنسلی است که به ازای کنسل کردن واگن باید تهیه شود که شامل شماره قبض و مبلغ آن می باشد.



شکل ۴-۶: فرم قبض کنسلی

- دکمه ثبت : این دکمه پس از وارد کردن فیلدها کار ثبت اطلاعات را درجدول

CancelGhabz انجام می دهد و همچنین فیلد (cancel) درواگذاری واگن ۱ میشود به این معنی

که این واگن کنسل شده است.

۴-۱-۷- فرم بارگیری :

پس ازاینکه تمام امور درمراحل قبل انجام شد. مثل درخواست واگن و واگذاری واگن شرکتی که واگن

راتحویل گرفته است باید مرحله بارگیری راانجام دهد. فرم بارگیری شامل اطلاعات بارنامه است که بارنامه شامل

اطلاعات وفیلدهایی ازقبیل : نوع کالا کد UN مبدا وکشور مقصد ایستگاه مقصد و وزن بار و کرایه و نوع

پرداخت کرایه و حق توقف و هزینه تعویض بوژی و...می باشد.

بارگیری و تحویل واگن

مشخصات را وارد نمایید و سپس گزینه "ثبت" را کلیک نمایید. برای مشاهده اطلاعات ثبت شده شماره بارنامه را وارد نمایید و کلید Enter را بزنید. برای حذف یک بارنامه شماره آن را وارد نمایید و دکمه حذف را کلیک نمایید.

شماره واگن	: First Field	ساعت تحویل بارنامه	:
شماره بارنامه	:	شماره اظهارنامه	:
تاریخ صدور	:	نام فرستنده	:

مشخصات بارنامه

نوع کالا	:	کرایه	:
کد UN	:	نوع کرایه	: پس کرایه
مبدا	:	حق توقف	:
کشور مقصد	:	هزینه تعویض بوزگ	:
ایستگاه مقصد	:	هزینه صدور بارنامه	:
وزن بار (کیلو)	:	هزینه قرنطینه	:
نام شرکت	: تسانس	سایر هزینه ها	:

بازگشت حذف بارنامه ثبت بارنامه

شکل ۴-۷: فرم بارگیری و تحویل واگن

- دکمه ثبت:

پس از وارد کردن تمام اطلاعات فرم بارگیری گزینه ثبت اطلاعات را وارد جدول Bargiri در پایگاه داده می کند.

- دکمه بازگشت:

فرم رامی بندد.

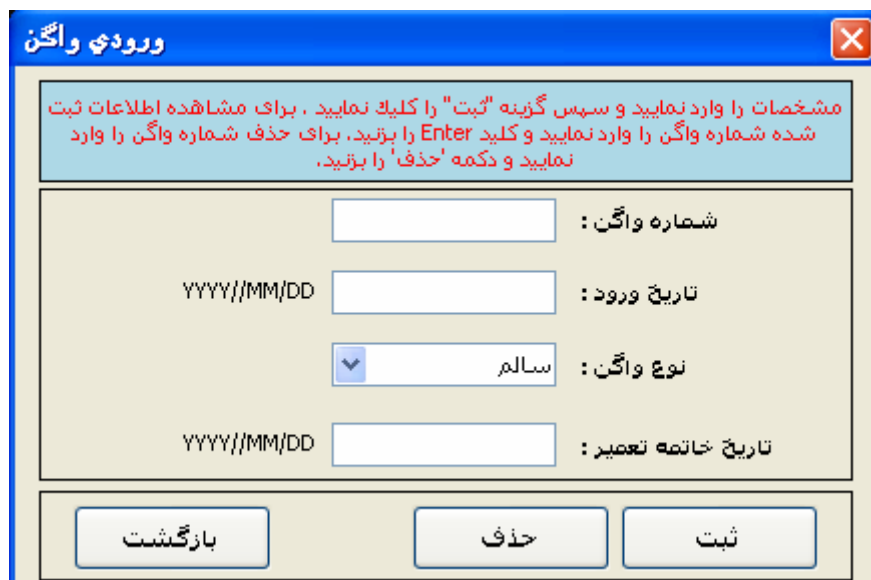
فیلد شماره بارنامه در این فرم فیلد اصلی جدول در پایگاه داده است یعنی وقتی که کاربر شماره بارنامه را وارد کند و کلید اینتر را بزند اگر بارنامه ای قبلا ثبت شده باشد اطلاعات آن رادر فیلد هانمایش می دهد و همچنین اگر شماره بارنامه ای که کاربر وارد کرده است در داخل جدول Bargiri وجود نداشته باشد این پیغام به کاربر نمایش داده می شود به این معنی که شماره بارنامه وارد شده اشتباه می باشد.

دکمه بازگشت هم برای بسته شدن فرم وانصراف از ثبت اطلاعات بارگیری تعیبه شده است و در صورتی که بر روی هر کدام از فیلدها باشیم و دکمه ESC روی صفحه کلید را بزنیم فرم بسته می شود. فرم بارگیری و اطلاعاتش در اصل جزء فعالیتهای اصلی شرکت محسوب می شود یعنی تا قبل از مرحله بارگیری در اصل شرکت فعالیتی انجام نداده است بارگیری جزء فعالیت اصلی و عمل شرکت در سیستم محسوب می شود.

۴-۱-۸- فرم ورودی واگن :

توضیحات :

در این فرم هر واگنی که آماده برای بارگیری است اطلاعات آن وارد این فرم می شود که فیلدهای آن شامل شماره واگن و تاریخ ورود و نوع واگن و تاریخ خاتمه تعمیر می باشد.



شکل ۴-۸ : فرم ورودی واگن

دکمه ثبت :

پس از وارد کردن این اطلاعات و زدن این دکمه اطلاعات در جدول Enter_vagon در پایگاه داده نگهداری می شود. SAVE

دکمه بازگشت برای انصراف از ثبت و بستن فرم می باشد.

۴-۱-۹- فرم گزارشات :

توضیحات:

این فرم شامل سربرگ های مختلفی است که در هر سربرگ گزارشگیری براساس پارامتر خاصی صورت می گیرد.

۱. سربرگ " براساس نام شرکت " :

این سربرگ شامل سه فیلد است که یکی نام شرکت و دو فیلد دیگر ***تاریخ را نشان می دهد. دکمه گزارش هم کار نمایش دادن اطلاعات در جدول را نشان می دهد. اطلاعات شامل فعالیت های شرکت است که در جدول Bargiri در پایگاه داده ثبت شده است.

گزارشات

س درخواست های کنسلی | بر اساس تاریخ واگذاری | بر اساس نوع واگن | بر اساس تاریخ صدور بارنامه | بر اساس کشور مقصد | بر اساس ایستگاه مبدأ | بر اساس درخواست های روزانه | بر اساس نام شرکت

نام شرکت: از تاریخ: 1387/05/01 تا تاریخ: 1388/05/01

هیچ خروجی برای جدول وجود ندارد!!!

ردیف	نام شرکت	شماره بارنامه	تاریخ صدور بارنامه	ساعت تحویل بارنامه	شماره اظهارنامه	نام صاحب بار	شماره واگن	نوع ک
------	----------	---------------	--------------------	--------------------	-----------------	--------------	------------	-------

تعداد فعالیت های شرکت:

شکل ۴-۹: فرم گزارشات/بر اساس نام شرکت

۲. سربرگ " گزارش براساس درخواست های روزانه و " :

در این سربرگ تاریخ را وارد می کنیم و سپس یکی از انواع درخواست را که روزانه ، هفتگی ، ماهانه و سالانه می باشد مشخص می شود و پس از زدن دکمه گزارش اطلاعات در جدول نشان داده می شود و تعداد رکوردهای بدست آمده نیز در زیر جدول نمایش داده می شود .

ردیف	کد شرکت	نام شرکت	تاریخ درخواست واگن	شماره نامه درخواست	تاریخ فرم درخواست	زمان فرم درخواست	نوع واگن	نوع
------	---------	----------	--------------------	--------------------	-------------------	------------------	----------	-----

شکل ۴-۱۰: فرم گزارشات/بر اساس درخواست های روزانه

۳. گزارش براساس ایستگاه مبدأ:

این صفحه شامل سه فیلد است :

در یک فیلد ایستگاه مبدأ را وارد می کنیم و دو فیلد دیگر زنج تاریخ را مشخص می کند که این تاریخ مربوط به تاریخ صدور بازنامه می باشد . با زدن دکمه گزارش که کنار فیلدهاست اطلاعات در جدول نمایش داده می شود و در پایین جدول هم فیلد تعداد رکوردها نشان داده خواهد شد .

گزارشات

س درخواست های کنسلی | بر اساس تاریخ واگذاری | بر اساس نوع واگن | بر اساس تاریخ صدور بازنامه | بر اساس کشور مقصد | بر اساس ایستگاه مبدأ | بر اساس درخواست های روزانه | بر اساس نام شرکت

ایستگاه مبدأ: ایران | از تاریخ: 1387/05/01 | تا تاریخ: 1388/05/01 | Print

ردیف	نام شرکت	شماره بازنامه	تاریخ صدور بازنامه	ساعت تحویل بازنامه	شماره اظهارنامه	نام صاحب بار	شماره واگن	نوع ک
------	----------	---------------	--------------------	--------------------	-----------------	--------------	------------	-------

تعداد رکوردهای جدول:

شکل ۴-۱۱: گزارش بر اساس ایستگاه مبدأ

۴. گزارش گیری براساس کشور مقصد :

در این صفحه سه فیلد برای وارد کردن اطلاعات داریم :

- فیلد کشور مقصد : برای وارد کردن کشور مقصد

- فیلد تاریخ اول : تاریخ اول را برای زنج تاریخ مشخص می کند

- فیلد تاریخ دوم : تاریخ دوم را برای زنج تاریخ مشخص می کند

با زدن دکمه گزارش گیری انجام شده و اطلاعات در جدول نمایش داده می شود و تعداد فعالیت ها در انتها زیر

جدول نمایش داده می شود .

ردیف	نام شرکت	شماره بارنامه	تاریخ صدور بارنامه	ساعت تحویل بارنامه	شماره اظهارنامه	صاحب بار	شماره واگن	نوع
------	----------	---------------	--------------------	--------------------	-----------------	----------	------------	-----

شکل ۴-۱۲: گزارش بر اساس کشور مقصد

۵. گزارشگیری براساس تاریخ صدور بارنامه :

توضیحات: در این سربرگ دوفیلد برای اطلاعات داریم که فقط برای مشخص کردن زنج تاریخ تعبیه شده است.

- فیلد تاریخ اول:

- فیلد تاریخ دوم:

با زدن دکمه گزارش گیری اطلاعاتی که تاریخ صدور بارنامه آنها بین این دو تاریخ باشند در جدول نشان داده

خواهند شد. و در زیر جدول هم تعداد رکوردهایی که پیدا شده اند در داخل فیلدی نشان داده می شود.

گزارشات

س درخواست های کنسلی | بر اساس تاریخ واگذاری | بر اساس نوع واگن | بر اساس تاریخ صدور بارنامه | بر اساس کشور مقصد | بر اساس ایستگاه مبدأ | بر اساس درخواست های روزانه | بر اساس نام شرکت

تاریخ صدور بارنامه

از تاریخ: 1387/05/01 تا تاریخ: 1388/05/01

Print

ردیف	نام شرکت	شماره بارنامه	تاریخ صدور بارنامه	ساعت تحویل بارنامه	شماره اظهارنامه	صاحب بار	شماره واگن	نو
------	----------	---------------	--------------------	--------------------	-----------------	----------	------------	----

تعداد فعالیت هاگ انجام شده در این تاریخ:

شکل ۴-۱۳: فرم گزارش بر اساس تاریخ صدور بارنامه

۶. گزارش گیری براساس نوع واگن :

این سربرگ شامل سه فیلد برای ورود اطلاعات است که در فیلد اول نوع واگن مشخص می شود و دو فیلد دیگر رنج تاریخ را مشخص می کند .

یک دکمه هم برای گزارش گیری داریم .

در هر کدام از سربرگ های فرم گزارشات در کنار دکمه گزارش گیری یک دکمه هم برای print قرار گرفته است

که با زدن این دکمه اطلاعات داخل Data Grid به فرم print برای چاپ اطلاعات فرستاده می شود .

س درخواست های کنسلی | بر اساس تاریخ واگذاری | بر اساس نوع واگن | بر اساس تاریخ صدور بارنامه | بر اساس کشور مقصد | بر اساس ایستگاه مبدأ | بر اساس درخواست های روزانه | بر اساس نام شرکت

نوع واگن: مسقف | از تاریخ: 1387/05/01 | تا تاریخ: 1388/05/01 | Print

ردیف	کد شرکت	نام شرکت	شماره نامه درخواست	تاریخ فرم درخواست	زمان فرم درخواست	نوع واگن	نوع کالا	کشور
۱	۱۲۲۴	hossein	۰۱۲	۱۳۸۷/۰۹/۱۱	۱۲:۰۰		jghj	
۲	۱۲۲۴	hossein	۰۱۲۲۴	۱۳۸۷/۰۹/۰۵	۱۲:۰۰		ایرانی	ا
۳	۱۲۲۴	hossein	۰۱۲۲۴۵۶	۱۳۸۷/۰۹/۰۶	۱۲:۰۰		kjklk	h
۴	۱۲۲۴	حافظان رایانه	۱۰۰	۱۳۸۷/۰۹/۱۸	۱۲:۰۰		نیمنی	سیت
۵	۱۲۲۴	حافظان رایانه	۱۰۱	۱۳۸۷/۰۹/۱۹	۱۲:۰۰		سششپ	س
۶	۱۲۲۴	حافظان رایانه	۱۰۲	۱۳۸۷/۰۹/۱۹	۱۲:۰۰		طویل	پر
۷	۱۲۲۴	hossein	۱۲۲۴				۱۲۵۴۷	
۸	۱۲۲۴	hossein	۱۲۲۴۵۶۷	۱۳۸۷/۰۹/۰۶	۱۲:۰۰		jjkddddddd	www
۹	۱۲۲۴	حافظان رایانه	۱۲۲۴۵۶۷۸۹	۱۳۸۷/۰۹/۱۸	۱۲:۰۰		لییب	س
۱۰	۱۲۲۴	hossein	۱۲۲۵۴۴۶۹۸	۱۳۸۷/۰۹/۰۵	۱۲:۰۰		پستتتش تشتش	متطیغ
۱۱	۱۲۲۴	حافظان رایانه	۱۲۸۷	۱۳۸۷/۰۹/۱۸	۱۲:۰۰		بیل	ع

تعداد درخواست ها: 11

شکل ۴-۱۴: فرم گزارشات/ بر اساس نوع واگن

۷. گزارش گیری براساس تاریخ واگذاری :

در این سربرگ پس از وارد کردن رنج تاریخ که همان تاریخ واگذاری می باشد لیست تمام درخواست هایی که در این رنج به آنها واگذار شده است در این سر برگ نمایش داده می شود:

گزارشات

س درخواست های کنتملی | بر اساس تاریخ واگذاری | بر اساس نوع واگن | بر اساس تاریخ صدور بارنامه | بر اساس کشور مقصد | بر اساس ایستگاه مبدأ | بر اساس درخواست های روزانه | بر اساس نام شرکت

از تاریخ : 1387/05/01 تا تاریخ : 1388/05/01

Print

ردیف	کد شرکت	نام شرکت	شماره نامه درخواست	تاریخ فرم درخواست	زمان فرم درخواست	تاریخ درخواست جهت واگذاری	کشور مقصد
------	---------	----------	--------------------	-------------------	------------------	---------------------------	-----------

شکل ۴-۱۵: فرم گزارشات/بر اساس تاریخ واگذاری

۸ گزارشگیری بر اساس درخواست های کنسلی:

در این سربرگ مشخص می کنیم که کدام درخواست ها را برای ما نمایش دهد آنهایی که کنسل شده اند یا نشده اند و یا انتقال تاریخ داشته اند یا نداشته اند. همچنین رنجی برای تاریخ داریم که آن تاریخ تاریخ واگذاری می باشد.

ردیف	کد شرکت	نام شرکت	شماره نامه درخواست	تاریخ فرم درخواست	زمان فرم درخواست	تاریخ درخواست جهت واگذاری	کشور مقصد
۱	۱۲۲۴	hossein	+۱۲۲۲۵۶	۱۳۸۷/۰۹/۰۶	۱۲:۰۰	۱۳۸۷/۰۹/۰۷	ایران
۲	۱۲۲۴	حافظان رایانه	۱۲۲۲۵۶/۸۹	۱۳۸۷/۰۹/۱۸	۱۲:۰۰	۱۳۸۷/۰۹/۱۸	سویس
۳	۱۲۲۴	حافظان رایانه	۱۲۸۷	۱۳۸۷/۰۹/۱۸	۱۲:۰۰	۱۳۸۷/۰۹/۱۸	یمن
۴	۱۲۲۴	حافظان رایانه	۱۰۱	۱۳۸۷/۰۹/۱۹	۱۲:۰۰	۱۳۸۷/۰۹/۱۹	سویس

شکل ۴-۱۶: فرم گزارشات/بر اساس درخواستهای کنسلی

۹. گزارش واگنهای تعمیری :

در این سربرگ لیست تمام واگنهای تعمیری و سالم که در رنج تایخی که کاربر وارد کرده است وارد سیستم شده اند نمایش داده می شود.

گزارشات

گزارش واگنهای تعمیری | بر اساس درخواست های کنسلی | بر اساس تاریخ واگذاری | بر اساس نوع واگن | بر اساس تاریخ صدور بارنامه | بر اساس کشور مقصد | بر اساس ایستگاه مبدأ

تاریخ ورود از تاریخ : 1387/05/01 تا تاریخ : 1388/05/01

Print

واگن های سالم :				واگن های تعمیری :					
تاریخ ورود	شماره واگن	ردیف		تاریخ خاتمه تعمیر	وضعیت واگن	تاریخ ورود	شماره واگن	ردیف	
۱۳۸۷/۰۹/۱۰	۱۱۱۱	۱		۱۳۸۷/۱۲/۰۲	تعمیری	۱۳۸۷/۱۲/۰۱	۱۱۱۱۱	۱	
۱۳۸۷/۰۹/۱۰	۱۳۳۴	۲		۱۳۸۷/۱۲/۲۵	تعمیری	۱۳۸۷/۱۵/۰۲	۱۲۴۵	۲	
۱۳۸۷/۰۵/۰۱	۱۳۳۴۵۶	۳		۱۳۸۷/۰۲/۰۲	تعمیری	۱۳۸۷/۰۹/۰۱	۱۲۵۴	۳	
۱۳۸۷/۰۸/۲۹	۱۳۳۴۵۶۷۸۹	۴		۱۳۸۷/۱۲/۰۱	تعمیری	۱۳۸۷/۱۰/۱۵	۱۲۵۸۷	۴	
۱۳۸۷/۰۹/۰۱	۱۳۳۴۵۸۷	۵		۱۳۸۷/۰۲/۰۲	تعمیری	۱۳۸۷/۰۹/۰۱	۱۲۶۹	۵	
۱۳۸۷/۰۵/۰۱	۲۱۲۵۴۸۷	۶		۱۳۸۷/۰۲/۰۲	تعمیری	۱۳۸۷/۰۹/۰۱	۱۲۸۷	۶	
				۱۳۸۷/۰۲/۰۲	تعمیری	۱۳۸۷/۰۹/۰۱	۱۲۸۷۷	۷	
				۱۳۸۷/۱۲/۲۵	تعمیری	۱۳۸۷/۱۵/۰۲	۱۵۴۸۹	۸	
				۱۳۸۷/۱۲/۲۵	تعمیری	۱۳۸۷/۱۵/۰۲	۱۶۵۴	۹	
				۱۳۸۷/۱۲/۰۵	تعمیری	۱۳۸۷/۱۲/۰۵	۲۲۲۲	۱۰	
				۱۳۸۷/۰۲/۰۲	تعمیری	۱۳۸۷/۰۲/۰۲	۵۴۹۸	۱۱	

شکل ۴-۱۷: فرم گزارشات/گزارش واگنهای تعمیری

۴-۱-۱۰- فرم جستجو:

این فرم شامل سه سربرگ می باشد که هر کدام جستجو براساس معیار خاصی می باشد. که شامل سه سربرگ می باشد.

۱. جستجو براساس نام شرکت:

در این سربرگ یک فیلد برای وارد کردن اطلاعات بیشتر نداریم که در داخل فیلد نام شرکت را وارد میکنیم.

The screenshot shows a web-based search form with the following sections:

- اطلاعات مخصوص شرکت:** Includes fields for company name, code, manager name, phone number, fax number, branch, and address.
- مشخصات چک:** Includes fields for check number, name, amount, and bank details.
- اطلاعات نماینده مطهری:** Includes fields for representative name, date, and contact information.
- اطلاعات نماینده مشهود:** Includes fields for representative name, date, and contact information.

شکل ۴-۱۸: فرم جستجو/بر اساس نام شرکت

دکمه " search " : بعد از وارد کردن نام شرکت و زدن این دکمه تمام اطلاعات ثبت شده شرکت در

جدول Company - Folder را نمایش می دهد.

۲. جستجو بر اساس شماره بارنامه :

در این سربرگ یک فیلد برای وارد کردن اطلاعات داریم که در آن شماره بارنامه را وارد میکنیم و بعد از وارد کردن شماره بارنامه و زدن دکمه search تمام اطلاعات بارنامه در فرم نمایش داده می شود .

The screenshot shows a web application window titled "جستجو" (Search). At the top, there are three tabs: "بر اساس نام شرکت" (By company name), "بر اساس شماره بارنامه" (By bill of lading number), and "بر اساس شماره واگن" (By wagon number). The "بر اساس شماره بارنامه" tab is selected. Below the tabs is a search bar with a "Search" button and a text input field. The main content area is divided into two sections. The upper section contains four input fields with labels: "تاریخ صدور بارنامه:" (Date of issuance), "ساعت تحویل بارنامه:" (Delivery time), "شماره اظهارنامه:" (Manifest number), and "نام صاحب بار:" (Shipper name). The lower section, titled "مشخصات بارنامه" (Bill of lading details), contains twelve input fields arranged in a grid, with labels: "شماره بارنامه:" (Bill of lading number), "نام شرکت:" (Company name), "هزینه تعویض بوزگ:" (Switching cost), "شماره واگن:" (Wagon number), "ایستگاه مقصد:" (Destination station), "هزینه صدور بارنامه:" (Issuance cost), "وزن بار:" (Weight), "نوع کالا:" (Goods type), "کد UN:" (UN code), "مبدأ:" (Origin), "نوع کرایه:" (Freight type), "سایر هزینه ها:" (Other costs), "حقی توقف:" (Actual stop), and "کشور مقصد:" (Destination country).

شکل ۴-۱۹: فرم جستجو/بر اساس شماره بارنامه

۳. جستجو براساس شماره واگن

در این فرم شماره واگن را در فیلد مخصوص که در بالای صفحه است وارد می کنیم و بعد از زدن دکمه search تمام اطلاعات واگن در فرم نمایش داده می شود .

شکل ۴-۲۰: فرم جستجو/بر اساس شماره واگن

۴-۱-۱۱- فرم وضعیت اتصال (Connection status) :

این فرم هنگامی باز می شود که در فرم اصلی برنامه دکمه " تست اتصال " را کلیک کنیم و وضعیت اتصال به شبکه را با توجه به Connection string تعریف شده به صورت یک متن (text) نشان می دهد. دکمه Ok هم فرم را می بندد.



شکل ۴-۲۱: فرم وضعیت اتصال

۴-۱-۱۲- فرم تنظیمات کاربر (frmuser Account):

این فرم برای تعریف کاربرانی که قرار است از این نرم افزار استفاده کنند طراحی شده است که شامل سه سربرگ است که به شرح زیر می باشد:

۱. سربرگ " تعریف کاربر ":

شکل ۴-۲۲: فرم تنظیمات کاربر/تعریف کاربر

در این صفحه می توان یک کاربر جدید تعریف نمود که برای این کار باید نام ، نام خانوادگی ، نام کاربری و کلمه عبور را وارد نمایید . پس از وارد کردن اطلاعات و زدن دکمه " ایجاد کاربر " در صورتی که این کاربر قبلاً برای سیستم تعریف نشده باشد به لیست کاربران تعریف شده اضافه می شود .

۲. سربرگ " حذف کاربر " :

تنظیمات کاربر

تعریف کاربر حذف کاربر تغییر رمز عبور

اطلاعات کاربر

نام کاربری :

کلمه عبور :

تکرار کلمه عبور :

انصراف حذف

شکل ۴-۲۳: فرم تنظیمات کاربر/حذف کاربر

در این صفحه کاربر پس از وارد کردن اطلاعات در قسمت اطلاعات کاربر و زدن دکمه "حذف" میتواند کاربری که قبلاً به لیست کاربران اضافه شده است را حذف کند. دکمه انصراف برای بستن فرم استفاده شده است.

۳. سربرگ " تغییر رمز عبور " :

تنظیمات کاربر

تعریف کاربر حذف کاربر تغییر رمز عبور

اطلاعات کاربر

رمز عبور قدیمی :

رمز عبور جدید :

تکرار رمز عبور جدید :

رمز عبور خودتان را وارد نمایید و سپس کلمه عبور جدید را وارد نمایید و بعد از تکرار کلمه عبور جدید دکمه تغییر را بزنید.

انصراف تغییر رمز عبور

شکل ۴-۲۴: تنظیمات کاربر/تغییر رمز عبور

در این صفحه پس از وارد نمودن اطلاعات کاربر وزدن دکمه " تغییر رمز عبور " کاربر میتواند رمز عبور خود را تغییر دهد .

ع-۱-۱۳- فرم print (frmPrint) :

The screenshot shows a window titled "Print" with a toolbar and a "Main Report" tab. The report content is as follows:

اداره کل راه آهن خراسان

2009 / 2009

نام شرکت	تاریخ صدور بارنامه	شماره بارنامه	تاریخ واکارگی	شماره اظهارنامه	نام صاحب بار	شماره وگن	مبلغ	کد	مبدأ	کشور مبدا	ایستگاه مقصد	وزن	کریه	حق توقف	هریبه عمومی بزرگ	هریبه صدور بارنامه	هریبه فرجه
شرکت راه آهن	1387/05/01	100		100	مشهد	1111	1000000.00	1200	مشهد	ایران	مشهد	1000			1,000,000.00	1,000,000.00	1,000,000.00

تعداد فعالیت های انجام شده : 1.00 هریبه صدور بارنامه : 1,000,000.00 هریبه عمومی بزرگ: 1,000,000.00

1

Current Page No.: 1 Total Page No.: 1 Zoom Factor: Whole Page

شکل ۴-۲۵: فرم پرینت

این فرم هنگامی باز می شود که در فرم گزارشات دکمه print که در کنار دکمه گزارش گیری واقع شده است زده شود ، پس از زدن این دکمه هر جدولی را که انتخاب کرده باشیم اطلاعاتش در این فرم (print) نمایش داده می شود واز اینجا می توانیم بر روی اطلاعات داخل فرم ویرایش داشته باشیم و یا آن را چاپ کنیم یا به صورت فایل در قسمتی از حافظه save کنیم . و تنظیمات مربوط به چاپ اطلاعات را از قبیل تنظیمات مربوط به اندازه صفحه ، نوع print و نصب آن و اندازه صفحه آن را از همین قسمت انجام دهیم و سپس از ویرایش

صفحه در صورتی که printer در اختیار داشته باشیم میتوانیم با زدن دکمه print اطلاعات را برای print به printer بفرستیم.

۴-۱-۱۴- فرم لیست تقاضاهای این تاریخ (Show Row Number) :

این فرم با زدن دکمه " شماره ردیف در لیست تقاضا " واقع در فرم درخواست واگن نشان داده می شود . پس از وارد کردن تاریخ در فرم درخواست واگن تمام تقاضاهایی که در آن تاریخ مشخص قرار است واگذار شوند را در این لیست نشان می دهد . یعنی تاریخ که وارد می شود همان تاریخ جهت واگذاری واگن است و در این فرم یک label قرار دارد که شماره ردیف را نشان میدهد .

ردیف	کد شرکت	نام شرکت	تاریخ درخواست جهت واگذاری	شماره نامه درخواست	تاریخ فرم درخواست
۱	۱۲۲۲	حافظان رایانه	۱۳۸۷/۰۹/۱۸	۱۰۰	۱۳۸۷/۰۹/۱۸

شماره ردیف در لیست تقاضا ۱ میباشد

شکل ۴-۲۶: فرم لیست تقاضاهای این تاریخ

۴-۱-۱۵- فرم " درخواست های این تاریخ " (frm show Reqs) :

در فرم اصلی برنامه درمنوی ثبت اطلاعات یکی از گزینه های آن نمایش درخواست های امروز می باشد ، با کلیک بر روی این گزینه این فرم باز می شود .

با زدن این گزینه تاریخ امروز سیستم به شمسی به فرم داده می شود و تمام درخواست های این روز در این فرم نمایش داده می شود و پس از خواندن اطلاعات هر درخواست کاربر میتواند با زدن تیک واکگذاری مربوط به هر درخواست واگن و واگن را به درخواست مربوط واکگذار نماید .

پس از زدن هر تیک فرم واکگذاری باز می شود و اطلاعات آن را کاربر باید وارد کند و گزینه ثبت را بزند .

درخواست							
ردیف	کد شرکت	نام شرکت	تاریخ درخواست جهت واکگذاری	شماره نامه درخواست	تاریخ فرم درخواست	زمان فرم درخواست	نوع
۱	۱۲۳۲	حافظات رایانه	۱۳۸۷/۰۹/۱۸	۱۰۰	۱۳۸۷/۰۹/۱۸	۱۲:۰۰	۰
۲	۱۲۳۲	حافظات رایانه	۱۳۸۷/۰۹/۱۸	۱۲۳۲۵۶۷۸۹	۱۳۸۷/۰۹/۱۸	۱۲:۰۰	۰
۳	۱۲۳۲	حافظات رایانه	۱۳۸۷/۰۹/۱۸	۱۳۸۷	۱۳۸۷/۰۹/۱۸	۱۳:۰۰	۰

شکل ۴-۲۷: فرم درخواست های این تاریخ