

فهرست

4	.....مقدمه
5	.....فصل اول : معرفی
6	.....(۱-۱) ضعف برنامه های انبار موجود
7	.....(۲-۱) انگیزه تحقیق
7	.....(۳-۱) هدف پروژه
7	.....(۴-۱) روش انجام پروژه
8	.....فصل دوم : مروری بر فرآیند انبارداری
9	.....(۱-۲) تعریف انبار
9	.....(۲-۲) دلایل استفاده از انبار
10	.....(۳-۲) انواع موجودیهای انبار
11	.....(۴-۲) سازمان انبار
11	.....(۵-۲) وظایف انباردار
12	.....(۶-۲) دیاگرام جریان اطلاعات در انبار
14	.....فصل سوم : طراحی نرم افزار
15	.....(۱-۳) گزارش شناخت سیستم
۱۶	.....(۲-۳) گزارش کامل آنالیز سیستم
30	.....(۳-۳) پیاده سازی بانک اطلاعاتی

34	فصل چهارم : پیاده سازی نرم افزار.....
35	۱-۴) مقدمه.....
36	۲-۴) تعریف چند واژه.....
36	۳-۴) چند نکته پیرامون پیاده سازی نرم افزار.....
37	۴-۴) ارزیابی پروژه.....
38	۵-۴) قسمتهایی از کد برنامه.....

46	پیوست.....
47	فرم های طراحی شده برنامه.....
55	منابع و مراجع.....

#### فهرست اشکال :

11	۱-۲- نمودار ارتباط سازمان انبار با مخازن و اداره امور انبارها.....
12	۲-۲- فرآیند انبار (ورودی ها و خروجی ها).....
13	۳-۲- دیاگرام جریان اطلاعات انبار.....
47	۱-۱- صفحه بارگذاری که در ابتدای برنامه ظاهر می شود.....

- 47 ..... ۲-۱- فرم رمز عبور برنامه
- 48 ..... ۳-۱- صفحه اصلی برنامه
- 48 ..... ۴-۱- فرم ثبت نام پرسنل
- 49 ..... ۵-۱- فرم ثبت کارهای روزانه پرسنل
- 49 ..... ۶-۱- فرم سفارشات شرکت
- 50 ..... ۷-۱- فرم مواد اولیه انبار ورودی
- 50 ..... ۸-۱- فرم تولیدات نهایی شرکت
- 51 ..... ۹-۱- فرم مواد اولیه انبار خروجی
- 52 ..... ۱۰-۱- فرم لیست کارهای ثبت شده
- 52 ..... ۱۱-۱- فرم جستجوی تولیدات نهایی
- 53 ..... ۱۲-۱- فرم جستجوی مواد اولیه انبار ورودی
- 53 ..... ۱۳-۱- فرم جستجوی سفارشات
- 54 ..... ۱۴-۱- فرم میزان دریافتی سفارشات
- 54 ..... ۱۵-۱- فرم گزارش مشخصات پرسنل
- 54 ..... ۱۵-۱- فرم ارتباط با ما



### چکیده

#### طراحی و پیاده سازی سیستم انبارداری

با توجه به وجود برنامه های انبار که تا کنون نیازهای انبار داری را پاسخ می دهد ولی در بیشتر موارد با توجه به نوع انبار و محل آن نیاز به برنامه خصوصی که نیاز های کامل انبار دار را پاسخ بدهد نیز وجود دارد . در برنامه های انبار داری جدید مسئول انبار یا مدیر ، با دادن فاکتورهای خاص به برنامه تصمیم گیری برای مقدار ورود و خروج کالا را برای تولید و غیره را به عهده خود برنامه می گذارد . این پروژه در نظر دارد که ایده های شرکت کفش مشکین مشهد را جهت استفاده بهینه از برنامه انبارداری برآورده سازد که مهم آنها عبارتند از :

❖ خصوصی بودن برنامه برای شرکت کفش مشکین مشهد

❖ محاسبه حقوق و دستمزد پرسنل شرکت

❖ بالا بردن امنیت برنامه در مقابل دستکاری داده ها و کاربران غیر مجاز

❖ یکپارچه سازی برنامه

مراحل به کارگرفته شده برای کسب و جمع آوری اطلاعات مورد نیاز این پروژه عبارتند از :

▪ مشاهده

▪ مصاحبه

▪ پرسشنامه

پس از انجام مراحل امکان سنجی و آنالیز ؛ در فاز طراحی و پیاده سازی این برنامه از ۶ جدول اطلاعاتی توسط نرم افزار Access استفاده شده است و برای پیاده سازی فرم ها از بسته نرم افزاری Visual Studio ؛ نرم افزار Visual Basic 6 استفاده شده است .

کلمات کلیدی :

## انبار - انبارداری - پایگاه داده - Visual Basic - Access - ODBC

### مقدمه

فنون و تخصصهای اداری در زمره مسائلی است که آگاهی از آن برای کارشناسان و صاحبان مشاغل امری ضروری است. هر اندازه دانش مدیران و کارگزاران در زمینه مدیریت و فنون اداری بیشتر و ابعادی گسترده تر داشته باشد، اعمال مدیریت و سرپرستی با کمال و نیروی بیشتری تحقق خواهد پذیرفت و نتیجتاً کارایی سازمان ها و کارکنان در جهت نیل به هدفهای سازمانی فزونی خواهد یافت.

برای دستیابی به هدف فوق تهیه متون آموزشی مناسب مورد تاکید خاصی قرار گرفته و با توجه به اهمیت کامپیوتر در دنیای امروز، بخصوص در تصمیم گیری های کشوری با حجم زیاد اطلاعات و نقش مهمی که این تکنولوژی در دستیابی به سایر تکنولوژی های پیشرفته ایفا می کند، استفاده از آن و ایجاد بانک های اطلاعاتی به کمک نرم افزار های پایگاه داده مفید واقع خواهد شد.

در این پایان نامه به مباحث زیر اشاره شده است:

- فصل اول: معرفی، که به معرفی کل سیستم می پردازد.
- فصل دوم: مروری بر فرآیند انبارداری، که به نکاتی در مورد اصول انبارداری اشاره می کند.
- فصل سوم: طراحی نرم افزار، که ساختار و چگونگی تشکیل پایگاه داده سیستم را مورد بررسی قرار می دهد.

○ فصل چهارم : پیاده سازی نرم افزار ، که در آن به نحوه پیاده سازی نرم افزار پرداخته ایم .

○ فصل پنجم : راهنمای استفاده از نرم افزار انبار ، که به کاربر مهارت های استفاده از این نرم افزار را می آموزد .

○ فصل ششم : قسمت هایی از کد برنامه که در آن با توابع مهم برنامه آشنا می شویم .

فصل اول :

معرفی

#### ۱-۱) ضعف برنامه های انبار موجود

با آمدن سیستم عامل ویندوز ، سیستم عامل داس رفته رفته از دور خارج شده است و برنامه های قدیمی از جمله برنامه های انبارداری تحت سیستم عامل داس بوده اند قابل مقایسه با برنامه های تحت ویندوز نمی باشد بنا به دلایل زیر :

- محیط گرافیکی ویندوز



محیط متن داس در مقابل محیط گرافیکی و دوستانه که کاربر کمتر احساس خستگی می کند و با سلیقه خود زمینه کاری را عوض می کند که تاثیرات روانی بهینه ای بر روی کاربر دارد .

- قابلیت چند وظیفه ای ویندوز

در محیط داس در هر لحظه فقط یک عمل را می توانید انجام دهید ولی در محیط ویندوز شما قابلیت چند وظیفه ای را دارید و چند کار را با هم انجام می دهید . مثلاً در حین کار با برنامه خود نامه های الکترونیکی خود را دریافت کنید ، فایل ها را از اینترنت دریافت کنید ، منابع خود را برای دیگران به اشتراک گذاشت و موسیقی نیز گوش کرد بدون اینکه برنامه خود را قطع کنید .

- استفاده از موتور های پایگاه داده

نرم افزار های تحت ویندوز امکاناتن بهتری برای استفاده از پایگاه داده مختلف دارد مثلاً در محیط Visual Foxpro علاوه بر استفاده از جداول Foxpro امکان استفاده از جداول Access ، Excell ، Oracle ، DB و ... را دارد . که این قابلیت برای برنامه نویس مفید واقع می شود و جداول طراحی شده در هر نرم افزار دیگر قابل استفاده است . که این امکان در محیط داس خیلی ضعیف می باشد .

- استفاده از قابلیت بهینه شبکه

در محیط شبکه ویندوز هر ایستگاه کاری می تواند داده ها و منابع خود را در اختیار دیگر ایستگاه ها بگذارد ولی در محیط شبکه ناول این امکان فقط در سرور وجود دارد و امکانات بیشمار شبکه ویندوز قابل قیاس با شبکه های قدیمی نیست .

## ۱-۲) انگیزه تحقیق

با توجه به وجود برنامه های انبار که تا حدودی نیازهای انبارداری را پاسخ می دهد ولی در بیشتر موارد با توجه به نوع انبار و محل آن نیاز به برنامه خصوصی که نیاز های کامل انباردار را پاسخ دهد نیز وجد دارد . در برنامه های انبارداری جدید مسؤل انبار یا مدیر با دادن فاکتور های خاص به برنامه تصمیم گیری برای مقدار ورود و خروج کالا را برای تولید و غیره را به عهده خود برنامه می گذارد .

## ۱-۳) هدف پروژه

این پروژه در نظر دارد که ایده های این شرکت را جهت استفاده بهینه از برنامه انبارداری برآورده سازد . که مهم آنها عبارتند از :

- ❖ خصوصی بودن برنامه برای شرکت کفش مشکین مشهد
- ❖ محاسبه حقوق و دستمزد پرسنل شرکت
- ❖ بالا بردن امنیت برنامه در مقابل دستکاری داده ها و کاربران غیر مجاز
- ❖ یکپارچه سازی برنامه

## ۱-۴) روش انجام پروژه

در ابتدا با دریافت فرمها و فاکتور ها و همچنین نیاز های برنامه از سوی کارفرما مراحل آنالیز برنامه صورت گرفته است . برای تهیه این برنامه از شش جدول استفاده شده است که برای تهیه این جداول از نرم افزار Access استفاده شده است و برای پیاده سازی فرم ها از بسته نرم افزاری Visual Basic 6 استفاده شده است .

فصل دوم :

مروری بر فرآیند انبار داری

### ۲-۱) انبار

به محل یا فضایی که یک یا چند نوع کالای بازرگانی، صنعتی، مواد اولیه و یا فرآورده های مختلف که بر اساس یک سیستم صحیح طبقه بندی می گردد، انبار گفته می شود.

انبار از نظر فرم به سه شکل می باشد:

- انبار های پوشیده: این امکان از تمام اطراف بسته و دارای سقف و وسایل ایمنی کامل می باشد.

- انبار های سر پوشیده: این انبار دارای سقف است ولی چهار طرف آن باز و فاقد حفاظ جانبی می باشد. این نوع انبار ها کالاها را از باران و آفتاب حفظ می کند.

- انبار های باز یا محوطه: این انبارها به صورت محوطه می باشد و معمولاً برای نگهداری ماشین آلات و لوازم سنگین مورد استفاده قرار می گیرد.

### ۲-۲) دلایل استفاده از انبار

در امور تدارکات و کار پردازی ، سیستم انبارداری از اهمیت خاصی برخوردار است و همکاری این دو واحد خدماتی با یکدیگر اجتناب ناپذیر است . سیستم صحیح انبارداری دارای متضمن مزایای زیر می باشد :

- دریافت ، حفاظت و دسترسی قرار دادن کالا ، مواد و وسایل مورد نظر به سهولت و سرعت انجام می شود .

- با اعمال کنترل دقیق ، از ازدیاد خارج از حد موجودی ها که ممکن است بر اثر تغییر قیمت ها باعث زیان گردد ، جلوگیری می شود .

- با استفاده صحیح از سیستم انبارداری میزان موجودی کالا در انبار و میزان مصرف آن در هر واحد ، که پایه و اساس حسابداری صنعتی می باشد ، محاسبه می گردد .

- کنترل میزان موجودی در انبار از نظر قیمت به سهولت صورت می گیرد .

- صدور قبض انبار ( رسید انبار ) موجبات تسهیل پرداخت مبلغ کالا به فروشنده و عملیات حسابداری می گردد .

- چون میزان موجودی ها معمولاً به قیمت تمام شده در انبار ها نگهداری می شود ، لذا در صورت خسارت ناشی از آتش سوزی و سایر ضایعات تعیین خسارت وارد شده به سهولت محاسبه می گردد .

### ۲-۳) انواع موجودی های انبار

اجناس و کالاهای موجود در انبار اعم از موسسات تولیدی را می توان به پنج دسته تقسیم نمود:

(۱) مواد خام و یا مواد اولیه

از این مواد برای تولید کالا استفاده می شود .

(۲) مواد و لوازم مصرفی و غیر مصرفی

مواد و لوازم مصرفی به موادی اطلاق می شود که به طور غیر مستقیم در انجام کار موسسه دخالت دارد و یا در سازمان های تولیدی برای انجام خدمات جانبی مورد استفاده قرار داده می شود . این مواد بر اثر مصرف از بین می روند مانند کاغذ و مواد و وسایل دسته بندی در سازمان های تولیدی .  
مواد غیر مصرفی که در انجام کار دخالت غیر مستقیم دارد ولی مصرف نمی شود و به مرور زمان کهنه و فرسوده می شود . مانند گوشی تلفن ، میز و صندلی .

(۳) مواد و لوازم در جریان ساخت

به موادی که مقداری تغییرات در مواد اولیه آن داده شده ولی هنوز به صورت کامل ساخته نشده است ، مواد و لوازم در جریان ساخت گویند .

(۴) کالایی که از نظر ساخت به مرحله تکمیلی رسیده و قابل عرضه می باشد .

(۵) اجناس خریداری شده جهت فروش

این سری اجناس شامل اجناسی هستند که بدون هیچ گونه تغییری به بازار جهت فروش عرضه می گردد و به طور موقت در انبار نگهداری می شود ، مانند فروشگاه ها و موسسات بازرگانی .

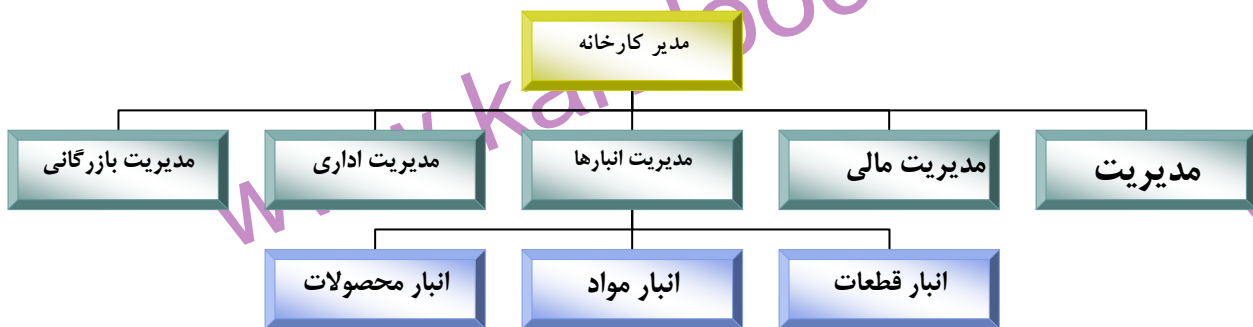
(۲-۴) سازمان انبار

انبار قطعات و مواد اولیه در واحد های تولیدی معمولاً زیر نظر مدیر فنی کارخانه قرار دارد و در بعضی از موسسات ، انبار مستقیماً زیر نظر مدیر موسسه انجام وظیفه می نماید .

در سازمان های بزرگ ، واحد انبار داری مدیریت مستقلی است که زیر نظر مدیر عامل قرار دارد ، و تحت نظر مدیر انبار نیز انبارهای مختلفی که هر یک از آنها دارای مسؤل می باشد ، انجام وظیفه می کند . مثلاً در واحد های تولیدی ، انبار مواد ، قطعات و ابزار آلات و در دانشگاه ها ، انبار دانشکده ها تحت نظر مدیر انبار می باشند .

در سازمان های دولتی و غیر انتفاعی ، معمولاً واحد انبار تحت نظر تدارکات و یا کار پردازی که خود این ها نیز یکی از واحد تابعه امور اداری محسوب می شود ، می باشد .

نمونه ای از نمودار سازمانی که موقعیت انبار و تدارکات را نشان می دهد ، در نمودار شکل زیر نشان داده شده است .



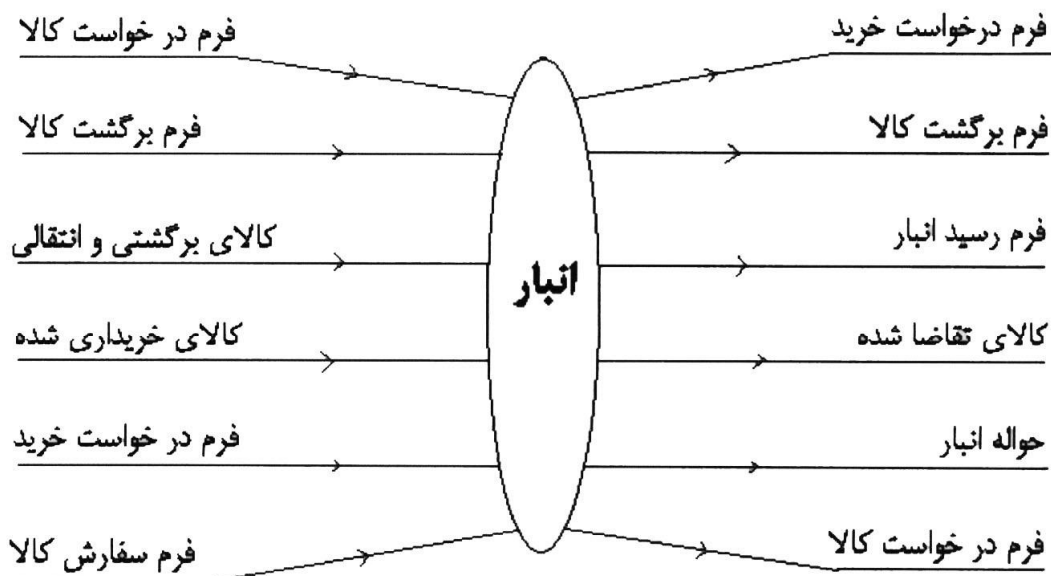
شکل (۲-۱) نمودار ارتباط سازمان انبار با مخازن و اداره امور انبار ها

### ۲-۵) وظایف انبار دار

انبار دار کسی است که عهده دار وظایف زیر می باشد .

- تحویل گرفتن اجناس و کالا های خریداری شده طبق اسناد و مدارک خرید
- صدور قبض انبار یا برگ رسید جنس به انبار پس از تحویل گرفتن کالا
- صدور حواله انبار هنگام تحویل دادن کالا
- صدور فرم های مرجوعی ، برگشت از خرید و سایر فرم های مشابه
- صدور برگ درخواست کالا در صورت لزوم
- ثبت مشخصات و تعداد اجناس وارد و صادر شده در دفاتر و کارت های انبار
- طبقه بندی ، تنظیم و کد گذاری کالا و اجناس
- همکاری با حسابداری صنعتی و واحد سفارش ها در مورد تعیین حداقل ، حداکثر و نقطه سفارش
- حفظ و حراست اجناس و رعایت اصول ایمنی در انبار
- بایگانی اسناد و مدارک انبار
- ارائه گزارش های لازم به مقامات ذیربط

## ۶-۲) دیاگرام جریان اطلاعات در انبار







فصل سوم :

طراحی نرم افزار

### ۳-۱) گزارش شناخت سیستم

شرکت تولیدی کفش مشکین در شهر مشهد - بلوار ابوطالب - ابوطالب ۲۳ - پلاک ۱۴ با شماره تلفن ۰۵۱۱-۷۲۴۵۷۸۴ واقع می باشد

مالکین شرکت آقایان مهندس فرزاد امینی خراسانی به عنوان مدیر عامل و مهندس سامان بیات می باشند. که سهام شرکت به طور مساوی بین آنها تقسیم شده است. شرکت دارای مجوز از اتحادیه صنفی کفشان می باشد و ملک مربوطه دارای جواز کاربری کارگاه است. و خدماتی چون تولید انواع کیف و کفش مردانه و زنانه را ارائه می دهند.

مواد اولیه این شرکت انواع چرم، انواع استری، انواع شبرو، انواع اشوالت، انواع قالب، ریزه، پارچه و... می باشد که این مواد از کاخانجات تولیدی چرم مانند کارخانه تولید چرم آفتاب و نیز فروشگاههای عرضه ی مواد اولیه تولیدی های کفش خریداری می شوند.

تمام خریدهای مواد اولیه از داخل کشور می باشد و خرید خارجی تنها محدود به مدل ها و قالب های جدید می باشد.

حمل مواد اولیه خریداری شده در داخل شهر مشهد توسط ماشین های باری خود شرکت و خریدهای خارج از شهر به عهده فروشنده است که باز هم توسط ماشین های باری فروشگاهی که از آن خرید شده حمل می شوند و پرداخت حساب مواد اولیه اغلب بصورت چکی و نیز بعضا نقدی می باشد.

مواد اولیه بعد از خریداری و حمل در انبارهای مخصوص که هوای خشک و تهویه دارای

و دمای متعادل ۲۵ (درجه سانتیگراد) است ، نگهداری می شوند که محیط انبار در زمانهای بخصوصی نظافت و سمپاشی می گردد .

مواد مورد نیاز برای هر قسمت توسط شخص شاغل در این قسمت بطور دستی از انبار برای تولید حمل می شوند .

شرکت بطور متوسط روزانه ۸۰ جفت کفش تولید می کند که تا حدود ۱۲۰ جفت نیز ظرفیت تولید دارد .

برای تولید هر کفش ابتدا چرم طبق مدل و الگوی از پیش تعیین شده و در سایزهای مختلف برش خورده و سپس دوخته می شوند (پستی) در مرحله ی بعد پستی ها روی قالب کشیده می شوند و در خشک کن حالت می گیرند و سپس ریزه کفش نصب و پرس می گردد و در آخر آماده سازی نهایی و بسته بندی صورت می گیرد.

نقطه ی سفارش و قطع سفارش رعایت نمی شود . و تنوع محصولات بستگی به فصول سال و تقاضا دارد به عنوان مثال در فصل پاییز کفش های بوت و نیمه بوت و در اواخر زمستان بیشتر کفشهای مجلسی و رسمی و... تولید می گردد . انبار مواد اولیه دارای دو قسمت است که در یک قسمت مواد اولیه و در قسمت بعدی محصولات انبار می شوند .

لوازم یدکی برای چرخ های خیاطی استفاده می شود و در صورت بروز مشکل و سرویس دستگاه های دیگر ، به تعمیرکار متخصص مراجعه می گردد .

فروش محصولات با استفاده از بازاریابی است که بازاریابان در تهران و سایر شهرستانها با مراجعه به فروشگاه های عرضه ی کفش و ارایه مدل های در حال تولید برای شرکت سفارش می گیرند . و نیز در شهر مشهد یک نمایندگی فروش دارد که فقط مسئول سفارشات است و پنخش محصولات و تسویه حساب به عهده ی خود شرکت است و در صورت اختلاف حساب با مراجعه به دفاتر فروش و سفارشات و نیز دفاتر پرسنل و... بررسی مجدد آنها اختلاف حساب رفع می شود در حال حاضر در این شرکت ۱۵ نفر مشغول کارند که در مواقع خاصی این تعداد بیشتر می شوند

این افراد نیاز به تخصص خاص خود را دارند که اغلب بصورت تجربی کسب کرده اند.

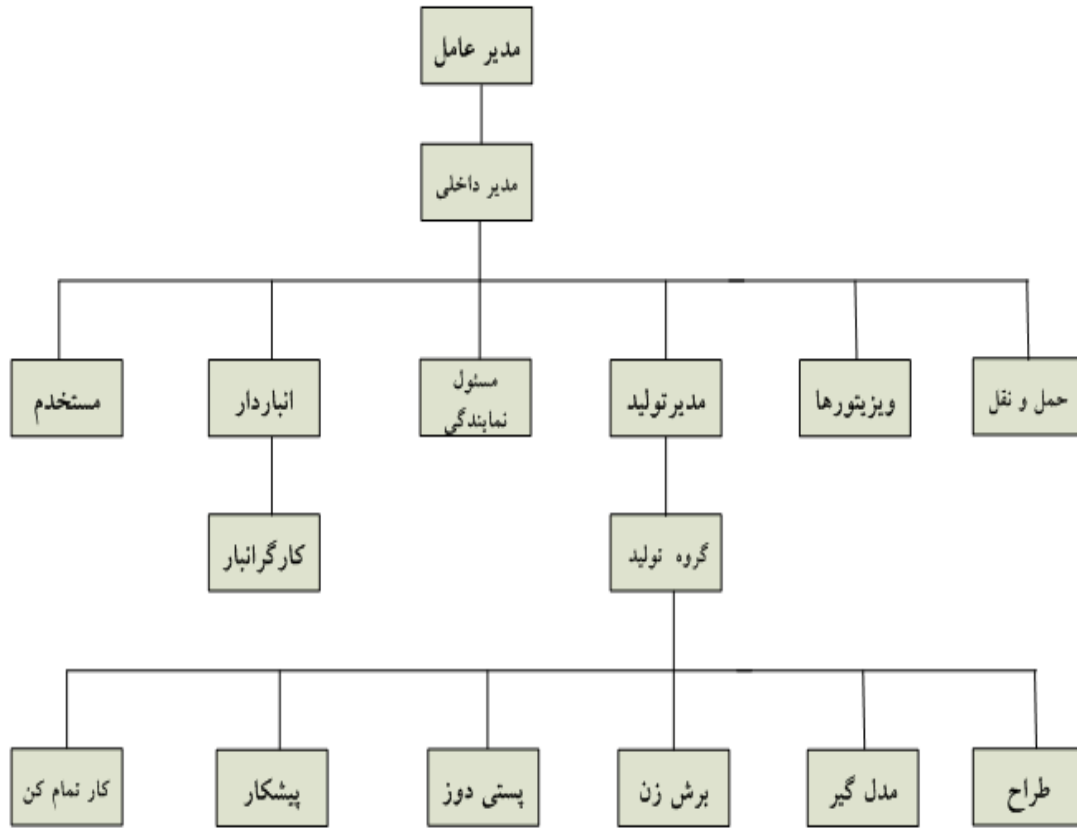
حقوق و دستمزد پرسنل بر اساس تعداد کاری که در طول هفته تحویل داده اند محاسبه و در پایان هفته با آنان تسویه حساب می گردد. بجز پرسنل اصلی بقیه مشمول مواردی همچون بیمه و... نمی باشند زیرا بطور دایم در تبلیغات هم توسط بازاریابان و هم هدایای تبلیغاتی انجام می پذیرد.

با عنایت به مطالب فوق نکات زیر به چشم می خورد:

- نقطه ی سفارش و نقطه ی قطع سفارش در این شرکت رعایت نمی شود.
- در صورت بروز اختلاف حساب باید تمام دفاتر فروش و سفارشات و نیز دفاتر پرسنل و... بررسی شود.

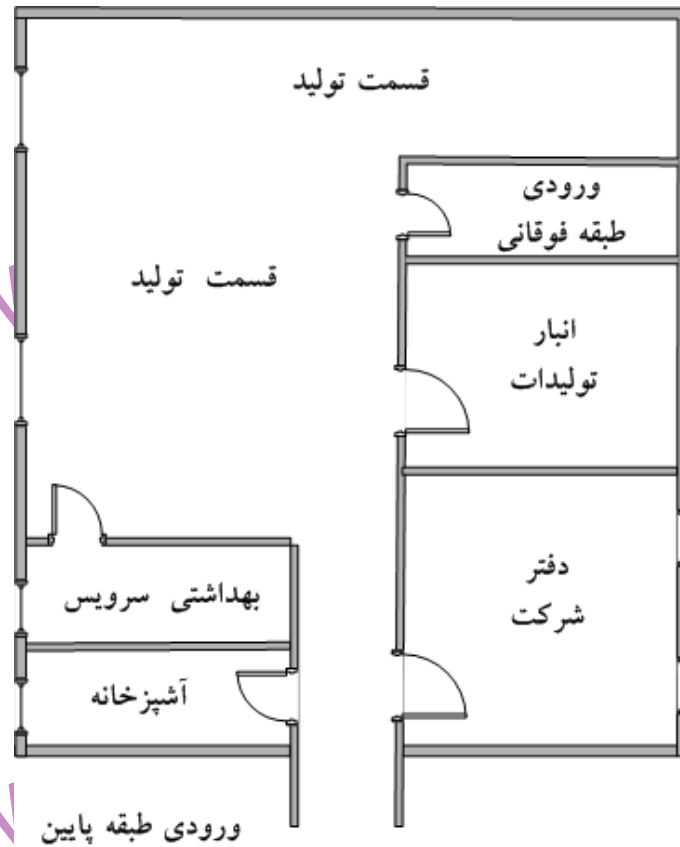
۲-۳) گزارش کامل آنالیز سیستم

۳-۲-۱ چارت سازمانی



۳-۱ نمودار چارت سازمانی شرکت کفش مشکین

۳-۲-۲) نمودار جانمایی (Schematic Layout)

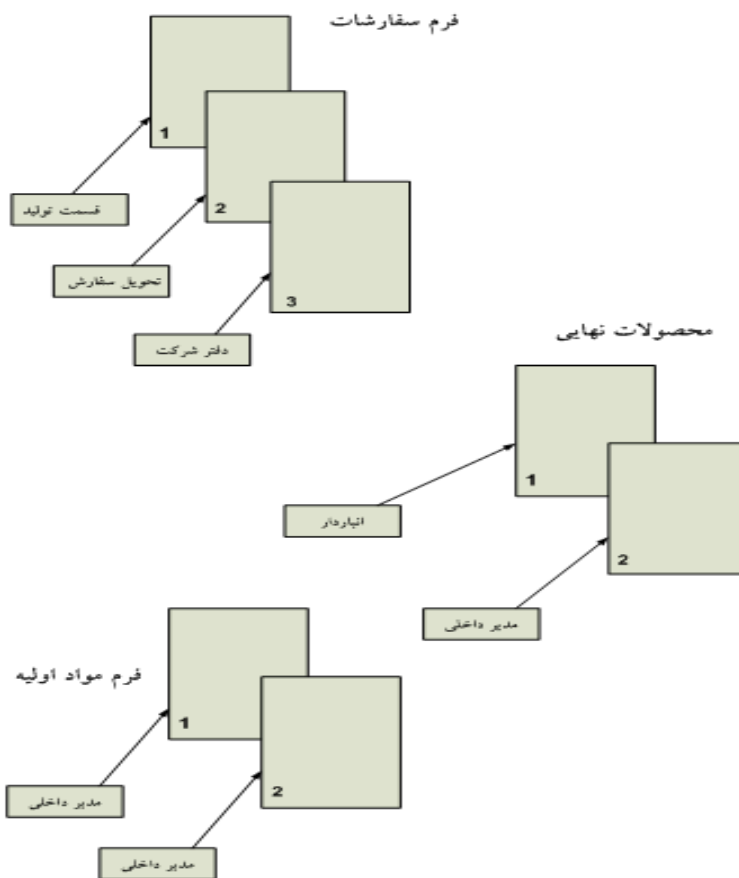


۳-۲) نمودار جانمایی دفتر و تولید شرکت کفش مشکین

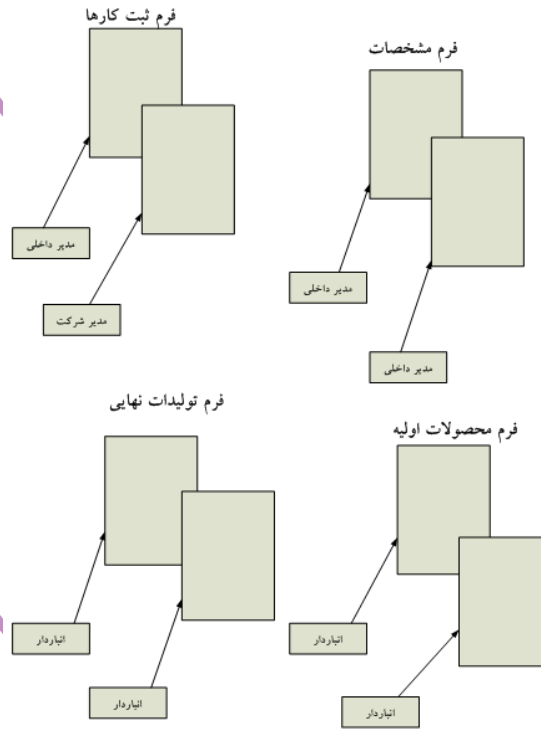


۳-۳) نمودار جانمایی انبار شرکت کفش مشکین

۳-۲-۳) آنالیز فرم ها

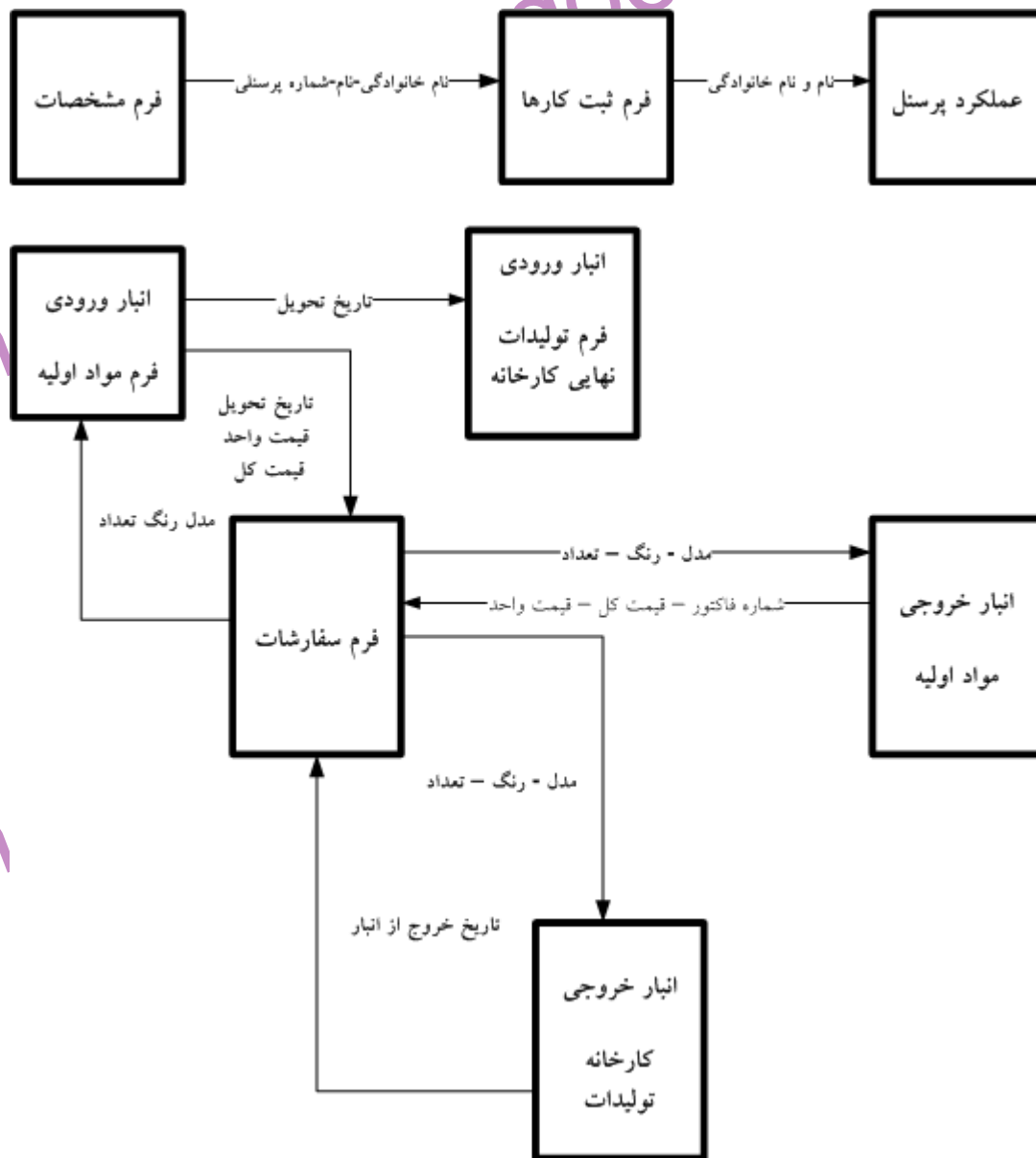






Form Flow نمودار (۳-۴)

FRD نمودار (۳-۲-۴)

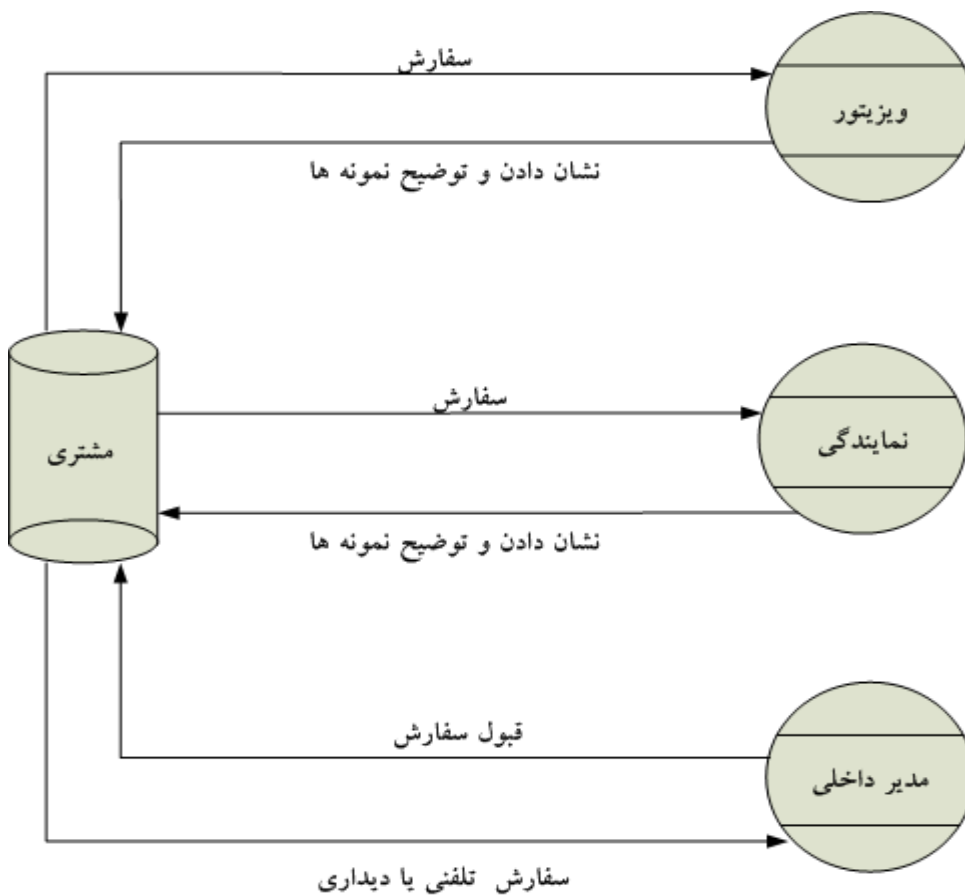


۳-۵) نمودار FRD سیستم کفش مشکین

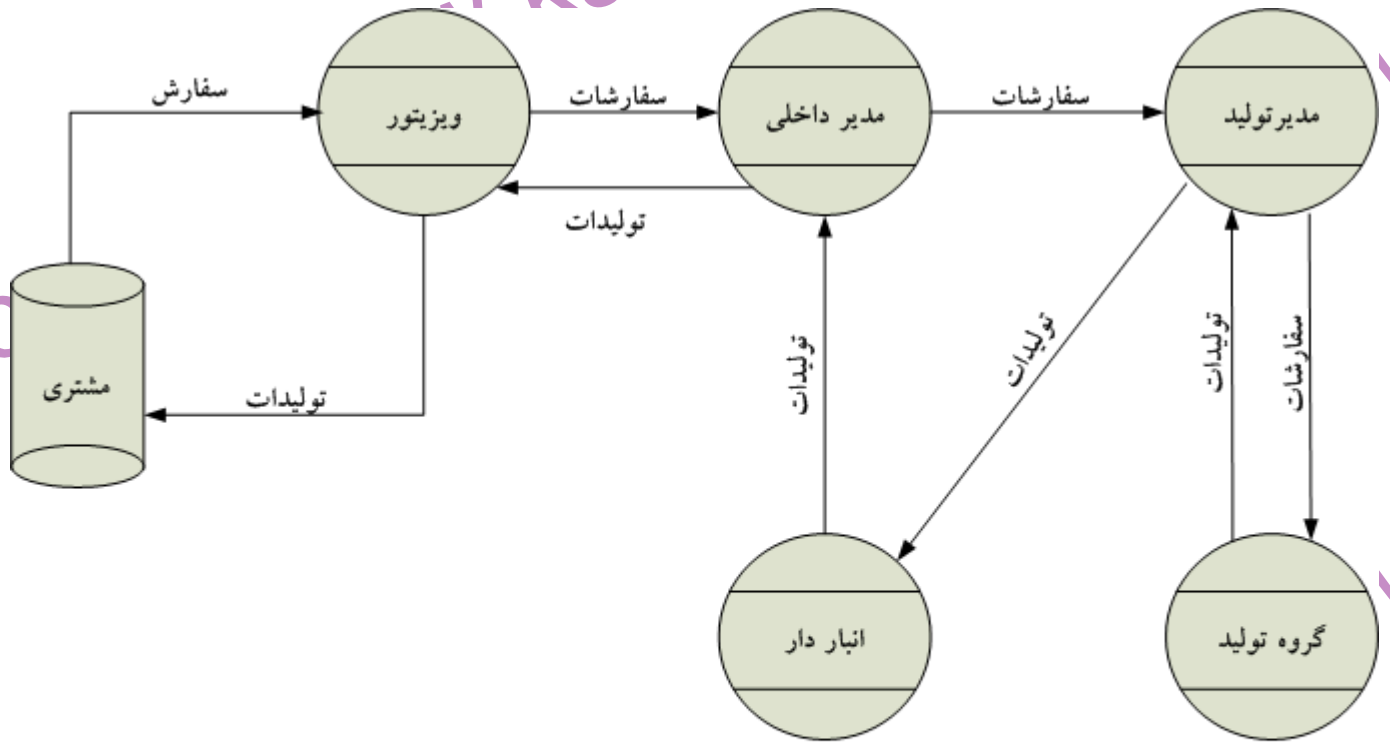
DFD نمودار های (۳-۲-۵)



L0: نمودار سطح صفر مربوط به ناظر خارجی

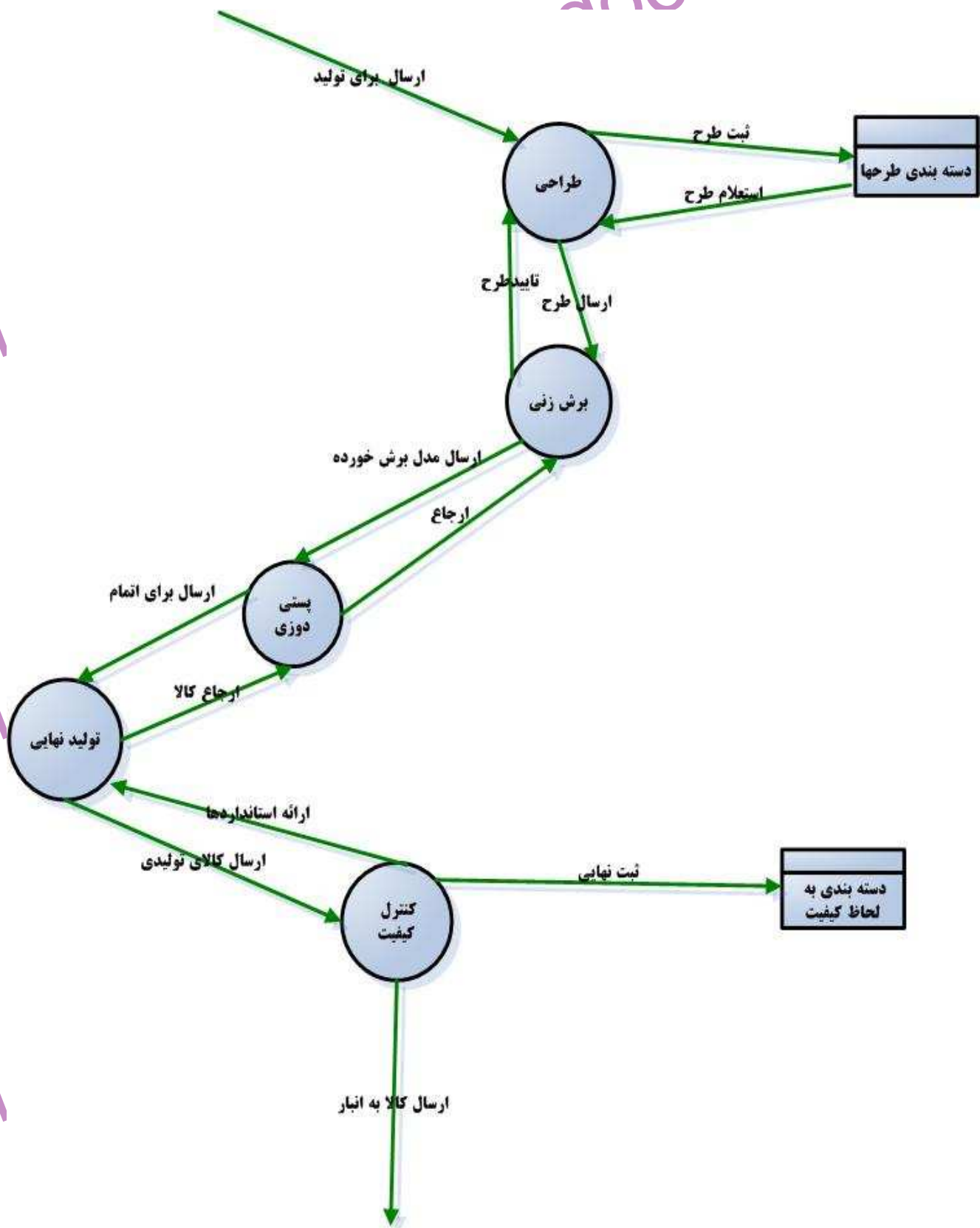


L1: نمودار سطح یک مربوط به داخل سیستم کفش مشکین



L2: نمودار سطح دوم مربوط به ویزیتور





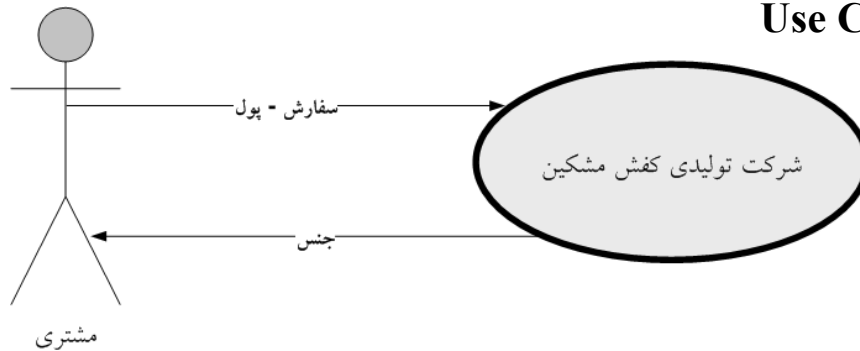
DFD :L4 گروه تولید

DFD نمودار های (۳-۶)

## Best Case

نمودارهای UML

Use Case (۳-۲-۶)

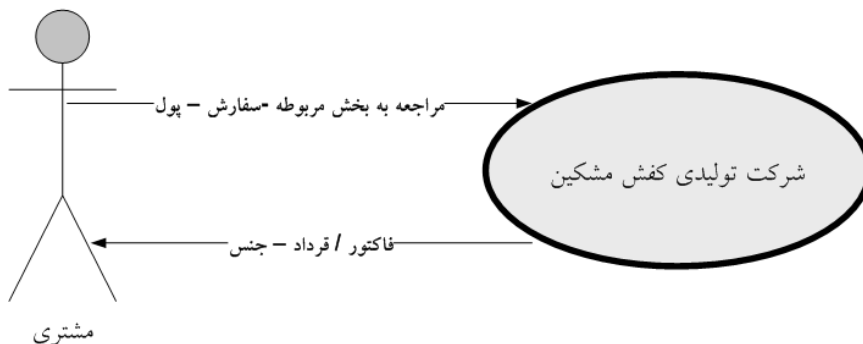


سناریو

- (1) وارد شرکت شوید.
- (2) سفارش خود را ابلاغ کنید.
- (3) مبلغی که باید پرداخت شود را پرداخت نمایید.
- (4) جنس خود را تحویل گرفته و خارج شوید

(۳-۷) نمودار Use Case ( Best Case )

## Jeneral Case



سناریو

- (1) وارد شرکت شوید.
- (2) برای ابلاغ سفارش با مدیر شرکت ملاقات کنید.
- (3) اگر مدیر شرکت حضور نداشت سفارش خود را به معاون وی ابلاغ نمایید.
- (4) چنانچه به توافق رسیدید قرارداد را منعقد کرده و پس واریز مبلغ رسید یا فاکتور را دریافت نمایید.
- (5) اگر اجناس درخواستی شما در انبار شرکت موجود نبود طبق تاریخ قرارداد می روید و در تاریخ معین شده برای دریافت اجناس خود مراجعه می نمایید.

( Jeneral Case ) Use Case نمودار (۳-۷)

Class Diagram نمودار های (۳-۲-۷)

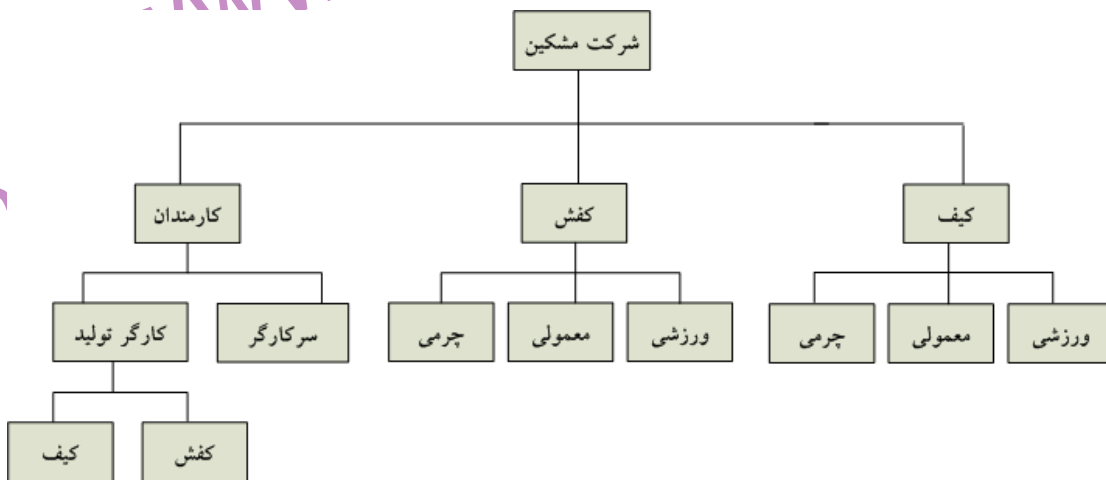
Class Component نمودار (۳-۲-۷-۱)

نام کلاس: کارمندان	نام کلاس: کیف	نام کلاس: کفش
نام نام خانوادگی محل سکونت تلفن شماره شناسنامه	نام مدل رنگ جنس تعداد	نام مدل رنگ جنس تعداد
سمت کارمند زمان کاری کارمند تخصص کارمند	جای دادن اشیا قابلیت حمل اشیا ...	پوشیدن ورزش کردن ...

(۳-۸) نمودار Class Component سیستم کفش مشکین

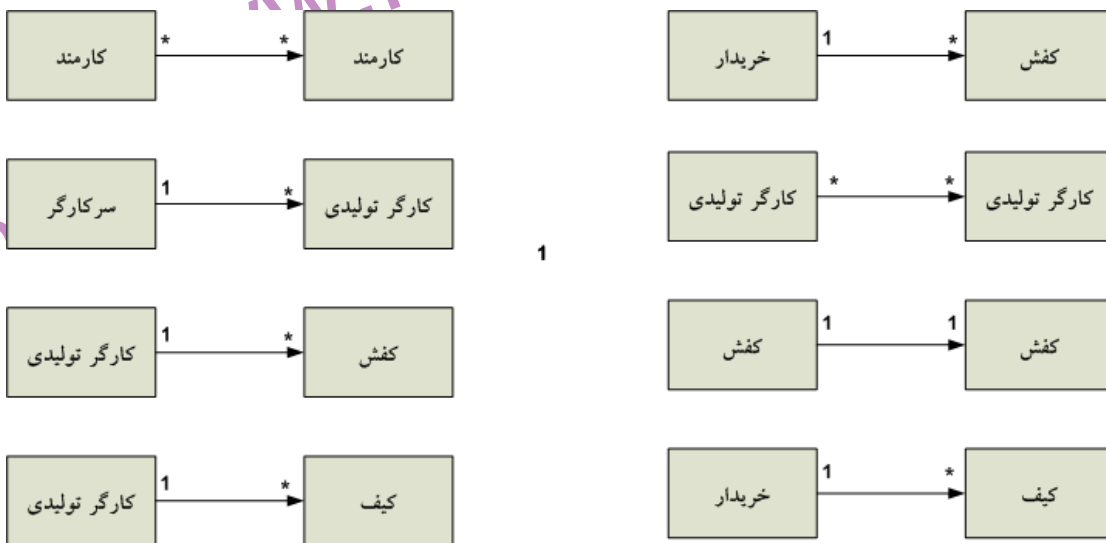


**Class Call Caboration** نمودار (۳-۲-۷-۲)



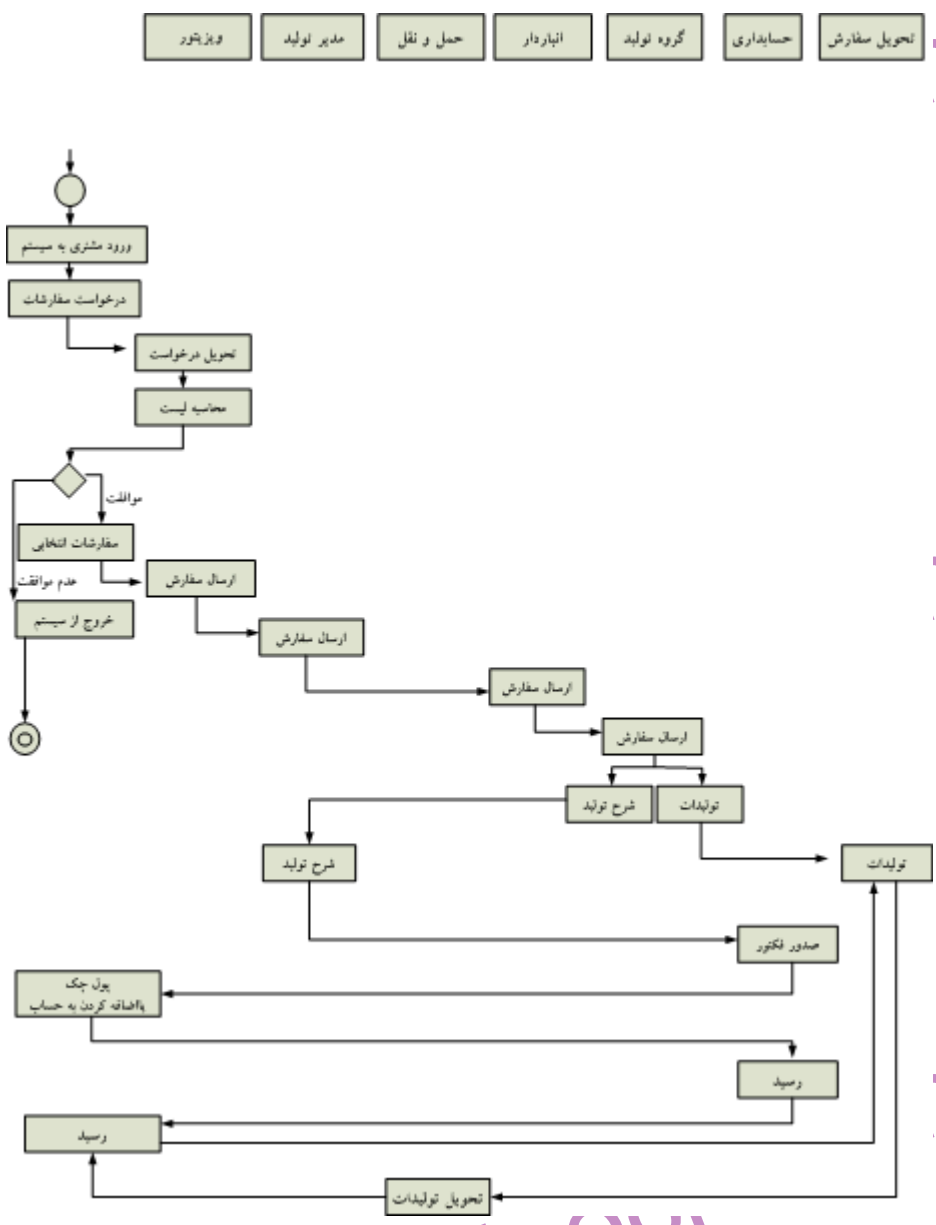
**Class Call Caboration** سیستم کفش مشکین نمودار (۳-۹)

**Class Relation** نمودار (۳-۲-۷-۳)



۳-۱۰ نمودار Class Relation سیستم کفش مشکین

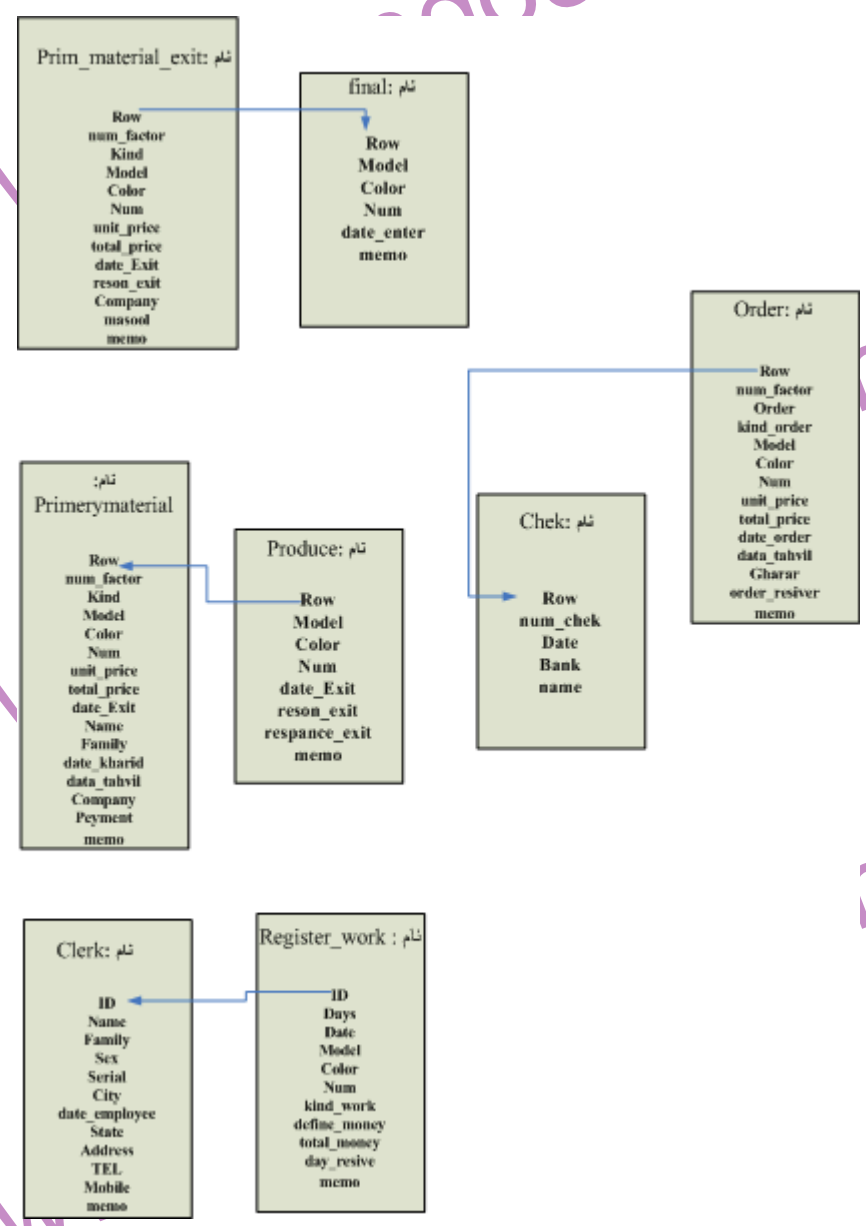
۳-۲-۷-۴ نمودار State diagram



۳-۱۱ نمودار State diagram سیستم کفش مشکین

www.kandooch.com

ERD نمودار (۳-۲-۷-۵)



www.kandooch.com

www.kandooch.com

## ۱۲-۳) نمودار ERD سیستم کفش مشکین

### ۸-۲-۳) پیاده سازی بانک اطلاعاتی

بانک اطلاعاتی مورد استفاده : Access

نام جداول مورد استفاده :

- ۱- جدول مربوط به ثبت مشخصات پرسنل ( Clerk )
- ۲- جدول مربوط به لیست چک ها ( Chek )
- ۳- جدول مربوط به ثبت تولیدات نهایی کارخانه ( final )
- ۴- جدول مربوط به ثبت سفارشات ( Order )
- ۵- جدول مربوط به ثبت مواد اولیه انبار خروجی ( Prim\_material\_exit )
- ۶- جدول مربوط به ثبت مواد اولیه انبار ورودی ( Primery\_material )
- ۷- جدول مربوط به ثبت تولیدات انبار خروجی ( Produce )
- ۸- جدول مربوط به ثبت عملکرد پرسنل ( Register\_work )
- ۹- جدول مربوط به رمز عبور ورودی برنامه ( Usersinfo )

نام بانک اطلاعاتی : DB Meshkin

طراحی جداول اطلاعاتی :

✓ جدول Clerk :

Caption	Field Size	Data Type	نام فیلد
شماره پرسنلی	Long Integer	Number	ID
نام	۲۰	Text	name
نام خانوادگی	۳۰	Text	family
جنسیت	۵	Text	sex
شماره شناسنامه	Long Integer	Number	serial
محل تولد	۲۰	Text	city
تاریخ استخدام	۱۵	Text	date_employe
محل سکونت	۲۰	Text	state
آدرس	۲۰۰	Text	address
تلفن تماس	Long Integer	Number	TEL
تلفن همراه	Long Integer	Number	mobile
توضیحات	-	Memo	memo

✓ جدول Chek

Caption	Field Size	Data Type	نام فیلد
ردیف	Long Integer	Number	row
شماره چک	Long Integer	Number	num_check

تاریخ	۲۰	Text	Date
نزد بانک	۲۰	Text	bank
نام صاحب حساب	۲۰	Text	name

جدول final ✓

Caption	Field Size	Data Type	نام فیلد
ردیف	Long Integer	Number	row
مدل	۳۰	Text	model
رنگ	۲۰	Text	color
تعداد	Integer	Number	num
تاریخ ورود به انبار	۲۰	Text	date_enter
توضیحات	-	Memo	memo

جدول Order ✓

Caption	Field Size	Data Type	نام فیلد
ردیف	Long Integer	Number	row
شماره فاکتور	Long Integer	Number	num_factor
سفارش دهنده	۳۰	Text	order

نوع سفارش	۲۰	Text	kind_order
مدل	۲۰	Text	model
رنگ	۲۰	Text	color
تعداد	Integer	Number	num
قیمت واحد	-	Currency	unit_price
قیمت کل	-	Currency	total_price
تاریخ سفارش	۱۰	Text	date_order
تاریخ تحویل	۱۰	Text	data_tahvil
قرار تحویل	۱۰	Text	gharar
سفارش گیرنده	۵۰	Text	order_resiver
توضیحات	-	Memo	memo

جدول Prim\_material\_exit ✓

Caption	Field Size	Data Type	نام فیلد
ردیف	Long Integer	Number	row

شماره فاکتور	Long Integer	Number	num_factor
نوع جنس	۲۰	Text	kind
مدل	۲۰	Text	model
رنگ	۲۰	Text	color
تعداد	Integer	Number	num
قیمت واحد	-	Currency	unit_price
قیمت کل	-	Currency	total_price
تاریخ خروج	۱۰	Text	date_Exit
علت خروج	۳۰	Text	reason_exit
فروشگاه طرف قرارداد	۳۰	Text	company
مسؤل خروج	۲۰	Text	masool
توضیحات	-	Memo	memo

جدول Primery\_material ✓

Caption	Field Size	Data Type	نام فیلد
ردیف	Long Integer	Number	row
شماره فاکتور	Long Integer	Number	num_factor
نوع جنس	۲۰	Text	kind
مدل	۲۰	Text	model
رنگ	۲۰	Text	color



تعداد	Integer	Number	num
قیمت واحد	-	Currency	unit_price
قیمت کل	-	Currency	total_price
تاریخ خروج	۱۰	Text	date_Exit
نام خریدار	۲۰	Text	name
نام خانوادگی خریدار	۳۰	Text	family
تاریخ خرید	۱۰	Text	date_kharid
تاریخ تحویل	۱۰	Text	data_tahvil
فروشگاه طرف قرارداد	۳۰	Text	company
نوع پرداخت	۱۰	Text	payment
توضیحات	-	Memo	memo

جدول Produce ✓

Caption	Field Size	Data Type	نام فیلد
ردیف	Long Integer	Number	row
مدل	۲۰	Text	model
رنگ	۲۰	Text	color
تعداد	Integer	Number	num

تاریخ خروج از انبار	۱۰	Text	date_Exit
علت خروج	۵۰	Text	reson_exit
مسئول خروج	۳۰	Text	respace_exit
توضیحات	-	Memo	memo

✓ جدول Register\_work

Caption	Field Size	Data Type	نام فیلد
شماره پرسنلی	Long Integer	Number	ID
ایام هفته	۱۰	Text	days
تاریخ	۱۰	Text	date
مدل	۲۰	Text	model
رنگ	۲۰	Text	color
تعداد	Integer	Number	num
نوع کار	۲۰	Text	kind_work
دستمزد تعیین شده	۱۰	Currency	define_money
دستمزد کل	۵۰	Currency	total_money
دریافتی روزانه	۳۰	Currency	day_reserve

توضیحات	-	Memo	memo
---------	---	------	------

جدول Usersinfo ✓

Caption	Field Size	Data Type	نام فیلد
ردیف	Long Integer	AutoNumber	autoNo
نام کاربر	۵۰	Text	tUserName
ID	۲۰	Text	tUserId
رمز عبور	۲۰	Text	tPassword

## پیاده سازی نرم افزار

### ۱-۴) مقدمه

با توجه به فرضیه های تحلیل نیازمندی ها و طراحی که قبلاً مورد اشاره قرار گرفت اینک به مرحله پیاده سازی می رسیم .

پیاده سازی این نرم افزار با بسته نرم افزاری Visual Basic 6 و تحت سیستم عامل Windows 98 صورت گرفته است . اکنون به تشریح مرحله پیاده سازی می پردازیم . قبل از آن مروری داریم بر ویژگیها و امکانات سودمند آن .

برنامه نویسی با Visual basic 6 می تواند کار پر زحمت و خسته کننده برنامه نویسی را به یک تفریح تبدیل کند ؛ بازی با اشیا گرافیکی و کشیدن و انداختن آن ها روی یک پنجره .

محبوبیت Visual Basic ، علاوه بر گرافیکی و ساده بودن ، از آنجا ناشی می شود که زبانی اینترپرتری ( interpreter ) و کامپایلری ( compiler ) است . با ترکیب این

دو ویژگی، **Visual Basic** به سهولت تولید برنامه و سرعت اجرا در محصول نهایی دست یافته است .

روزی که میکروسافت اولین ویرایش **Visual Basic** را روانه بازار کرد، بسیاری از منتقدان شکست آن و مرگ خاندان بیسیک را پیش بینی می کردند . این مخالفان می گفتند که اساساً یک زبان جدی نیست و ساختار آن ( برخلاف C و پاسکال ) به درد محیط ویندوز نمی خورد . اما میکروسافت با ارایه **Visual Basic** درس های زیادی به جامعه برنامه نویسان داد :

- زبانی مانند بیسیک می تواند قابل فهم قوی باشد .
- زبانی مانند بیسیک با داشتن واسط مناسب می تواند برای محیط ویندوز هم کارایی داشته باشد .
- **Visual Basic** بسته به نیاز برنامه نویس می تواند کامپایلری یا

اینترپرتی باشد

- و **Visual Basic** نه تنها فراموش نشد بلکه به محبوبترین زبان برنامه نویسی دنیا تبدیل شد .

**Visual Basic** در چندین مدل مختلف عرضه می شود :

- مدل فوق العاده : محیط برنامه نویسی مناسب کارهای شبکه و برنامه های توزیع شده .
- مدل حرفه ای : محیط برنامه نویسی مناسب کارهای حرفه ای به همراه مجموعه کاملی از ابزارهای جنبی .

- مدل آموزشی : مجموعه ای از ابزارهای استاندارد و هر آنچه برای شروع برنامه نویسی بدان نیاز دارید .

- مدل استودیوی ویژوال : در این مدل **Visual Basic** به همراه چند زبان ویژوال دیگر ( مانند ویژوال C++ و ویژوال J++ ) در یک بسته نرم افزاری گرد آمده اند تا آموختن آن ها متضمن کمترین زحمت برای برنامه نویسان باشد . که ما در پیاده سازی این نرم افزار از این مدل استفاده کرده ایم .

## ۲-۴) تعریف چند واژه

اکنون به تشریح چند واژه که در سایر قسمت ها زیاد استفاده شده اند ی پردازیم :

**Component** : یک جزء Component مثل برچسب متن ، کنترل ویرایش یا جعبه لیست تکه ای از نرم افزار است که دارای عملکرد خاص و از پیش تعیین شده است .

**Property** : یک خاصیت (Property) عملکرد جزئی را تعیین می کند ، مثلاً خاصیت رنگ ، رنگ یک شی ویندوز را معین می کند .

**Event** : یک Event یا رویداد چیزی است که در نتیجه واکنش (Interactive) یک جزء با کاربر یا ویندوز رخ می دهد .

**Module** : کدها و روتین های عمومی برنامه ، که به هیچ کنترل خاصی مربوط نیستند ، باید در مدول (module) ها قرار داده شوند . از یک مدول به دلیل خصلت عمومی آن می توان در برنامه های متعددی استفاده کرد . پسوند نام فایل مدول BAS است .

**Form** : هر پروژ می تواند فرم های متعددی داشته باشد . پسوند نام فایل فرم ها همیشه FRM است .

## ۳-۴) چند نکته پیرامون پیاده سازی نرم افزار

در پیاده سازی این نرم افزار علاوه بر بسته نرم افزاری Visual Basic 6 از نرم افزار های Adobe Photoshop و Paint به منظور تهیه آرم شرکت و تصاویر بر روی فرم ها و نیز عکس های تهیه شده از برنامه ، استفاده شده است .

با توجه به حجم بالای برنامه و طول کشیدن مدت زمان اجرا معمولاً در ابتدای برنامه تصویری را در نظر می گیرند که بعد از لود شدن کامل برنامه این تصور می رود که برای برنامه تصویری طراحی شده است مربوط به کار شرکت می باشد . اولین چیزی که هنگام لود شدن برنامه حائز اهمیت است مجوز کاربر برای استفاده از برنامه می باشد که برای این کار فرم رمز عبور در نظر گرفته شده است .

## ۵-۴) مستندات مشترک کلیه فرم ها

دستور *Option Explicit* : این دستور باعث می شود که چنانچه متغیری تعریف نشده باشد برنامه خطا می گیرد .

اگر دستور فوق وجود نداشته باشد و نیز در صورت عدم تعریف متغیر ، برنامه خود متغیر را از نوع *varent* در نظر می گیرد .

متغیر های سراسری برنامه :

۱- *mboolError (Dim mboolError As Boolean)*

این متغیر از نوع بولین است و برای تشخیص خطا به کار می رود .

۲- *mboolLoadError (Dim mboolLoadError As Boolean)*

این متغیر از نوع بولین است و در صورت ایجاد خطا به کمک این متغیر از اجرای برنامه جلوگیری می کنیم

زیر برنامه ها :

۱- رویداد **Error** دیتا کنترل

( در صورتی که *Data Control* با خطا مواجه شود از ظهور فرم جلوگیری می نماید . )

```
Private Sub Adodc1_Error(ByVal ErrorNumber As Long,  
Description As String, ByVal Scode As Long, ByVal Source As  
String, ByVal HelpFile As String, ByVal HelpContext As Long,  
fCancelDisplay As Boolean)
```

```
Call DCErrorCheck(ErrorNumber, Description,  
fCancelDisplay  
End Sub
```

دلایل متفاوتی برای بروز خطا وجود دارد :

قطع بودن شبکه ( *Network* )

مشترک نبودن بانک اطلاعات در شبکه

در دسترس نبودن سرویس دهنده ( *Server* )

قفل شدن انحصاری جدول توسط یک کاربر

و ...

در این زیر برنامه روال *DCErrorCheck* فراخوانی شده که به منظور پردازش خطای واقع شده در *Data Control* نوشته است .

۲- زیر برنامه *DCErrorCheck* (به منظور جلوگیری از نمایش یک فرم خالی در صورت بروز خطا توسط *Data control*)

```
Private Sub DCErrorCheck(ByVal ErrorNumber As Long,  
Description As String, fCancelDisplay As Boolean)  
If Not mboolLoadError Then  
MsgBox Description  
mboolLoadError = True  
fCancelDisplay = True  
Else  
fCancelDisplay = True  
End If  
End Sub
```

پارامترهای موجود در این زیر برنامه عبارتند از :

*ErrorNumber* که از نوع *Long* می باشد و شماره ی خطا در این متغیر قرار می گیرد

*Description* که از نوع *String* می باشد و شرح خطا در آن قرار می گیرد  
*fCancelDisplay* از نوع *Boolean* می باشد و برای جلوگیری از نمایش خطای *Data control* استفاده می شود با استفاده از این پارامتر به *data Control* می

گوییم که خطای خودش را نشان ندهد چون ما آن را پردازش می کنیم و خطای مناسب را نشان خواهیم داد.

در دستور شرطی ابتدای روال به وسیله متغیر *mboolLoadError* چک می کنیم که قبلا خطایی رخ داده یا خیر اگر خطای رخ نداده باشد خودمان همان خطا را به وسیله یک *MsgBox* اعلام می کنیم و سپس مقدار متغیر *mboolLoadError* را به *True* تنظیم می کنیم تا نشان دهیم که خطایی واقع شده است و پارامتر



*Data* را *True* کردیم که از نمایش پیغام خطای خود *CancelDisplay* جلوگیری کنیم .

۳- زیر برنامه ابتدای رکورد

```
Private Sub cmdfirst_Click()  
    Adodc1.Recordset.MoveFirst  
    cmdprevious.Enabled = False  
    cmdnext.Enabled = True  
End Sub
```

بوسیله دستور اول به ابتدای رکورد ها می رویم چون قبل از ابتدایی ترین رکورد رکورد دیگری نداریم خاصیت *Enabled* دکمه قبلی را غیر فعال می کنیم و پس از آن خاصیت *Enabled* دکمه بعدی را فعال می کنیم تا کاربر بتواند به رکوردهای بعدی دسترسی داشته باشد .

۴- زیر برنامه رکورد قبلی

```
Private Sub cmdPrevious_Click()  
    Adodc1.Recordset.MovePrevious  
    If Adodc1.Recordset.BOF Then  
        cmdprevious.Enabled = False  
    Adodc1.Recordset.MoveNext  
    End If  
    cmdnext.Enabled = True  
End Sub
```

اگر در ابتدای فایل قرار داشته باشیم خاصیت *Enabled* دکمه قبلی غیر فعال می شود و بر روی رکورد بعدی قرار می گیریم در غیر این صورت به رکورد قبلی می رویم و خاصیت *Enabled* دکمه بعدی فعال می کنیم .

۵- زیر برنامه رکورد بعدی

```
Private Sub cmdNext_Click()  
    Adodc1.Recordset.MoveNext  
    If Adodc1.Recordset.EOF Then  
        cmdnext.Enabled = False
```

```
Adodc1.Recordset.MovePrevious  
End If  
cmdprevious.Enabled = True  
End Sub
```

بوسیله این روال به رکورد بعدی می رویم و خاصیت Enabled دکمه قبلی را فعال می کنیم اگر در انتهای فایل قرار داشته باشیم خاصیت Enabled دکمه بعدی را غیر فعال می کنیم و بر روی رکورد قبلی قرار می گیریم .

۶- زیربرنامه انتهای رکورد

```
Private Sub cmdlast_Click()  
Adodc1.Recordset.MoveLast  
cmdprevious.Enabled = True  
cmdnext.Enabled = False  
End Sub
```

بوسیله این زیر برنامه به انتهای فایل رفته و خاصیت Enabled دکمه قبلی را فعال و دکمه بعدی را غیر فعال می کنیم .

۷- زیر برنامه دکمه جدید

```
Private Sub cmdNew_Click()  
Adodc1.Recordset.AddNew  
txtID.SetFocus  
Call ToggleButtons  
End Sub
```

بوسیله این زیر برنامه رکورد جدیدی ایجاد می شود سپس با دستور بعد فوکس به txtID منتقل می شود و در آخر زیر برنامه ToggleButtons فراخوانی می شود .

۸- زیر برنامه ToggleButtons

```
Private Sub ToggleButtons()  
If cmdclose.Caption = "....." Then  
cmdclose.Caption = "....."  
Else  
cmdclose.Caption = "....."  
End If  
cmdsave.Enabled = Not cmdsave.Enabled
```

*cmdnew.Enabled = Not cmdnew.Enabled*  
*cmddelete.Enabled = Not cmddelete.Enabled*  
*cmdfind.Enabled = Not cmdfind.Enabled*  
*cmdprevious.Enabled = Not cmdprevious.Enabled*  
*cmdnext.Enabled = Not cmdnext.Enabled*  
*cmdfirst.Enabled = Not cmdfirst.Enabled*  
*cmdlast.Enabled = Not cmdlast.Enabled*  
*lstname.Enabled = Not lstname.Enabled*  
*Adodc1.Enabled = Not Adodc1.Enabled*

*End Sub*

بوسیله این روال از تغییر یافتن داده ها آگاه شده و به برنامه پاسخ مقتضی را می دهیم و نیز وضعیت دکمه ها را تغییر می دهیم ابتدا در یک عبارت شرطی وضعیت دکمه *cmdclose* را مورد بررسی قرار می دهیم در صورتی خاصیت *Caption* این کنترل "خروج" باشد آن را به "لغو" تغییر می دهیم و چنانچه به "لغو" تغییر یافته باشد آن را به "خروج" تغییر نام می دهیم و در واقع دکمه بین این دو وضعیت سوئیچ می کند .

در قسمت بعد وضعیت کنترل های *cmddelete cmdnew cmdsave lstname cmdlast cmdfirst cmdnext cmdprevious cmdfind Adodc1* را تغییر می دهیم چنانچه فعال باشند ، غیر فعال می شوند و چنانچه غیر فعال باشند ، فعال می شوند بدین منظور از عملگر *Not* استفاده کرده ایم در صورتی که ما یک مقدار *True* را به این عملگر بدهیم ، به ما مقدار *False* برمی گرداند و در صورتی که ما یک مقدار *False* را به این عملگر بدهیم ، به ما مقدار *True* برمی گرداند .

۹- زیر برنامه **TextChange**

```
Private Sub textChange()  
If Adodc1.Recordset.BOF Or Adodc1.Recordset.EOF Then  
do nothing'  
Else
```

```
If Not cmdsave.Enabled Then  
    Call ToggleButtons  
End If  
End If  
End Sub
```

با وقوع رویداد *Change* هر *Text Box* این روال فراخوانی شده در این رویداد چنانچه کنترل داده بر روی رکورد معتبری باشد خاصیت *Enabled* دکمه ذخیره فعال شده تا کاربر بتواند اطلاعات را ذخیره نماید و سپس زیر برنامه *ToggleButtons* فراخوانی می شود .

#### ۱۰- رویداد *Change* کلیه *Text Box* ها :

```
Private Sub TextBox_Change()  
If TextBox.Tag <> TextBox.Text Then  
    Call textChange  
End If  
End Sub
```

برای تشخیص مقدار قبلی هر کنترل باید این مقدار را در جایی ذخیره کنیم بدین منظور همه کنترل ها خاصیتی به نام *Tag* را دارا می باشند که این عمل را انجام می دهند

با وقوع رویداد *Change* هر *Text Box* ابتدا چک می کنیم که آیا مقدار فعلی کنترل با مقدار قبلی برابر است یا خیر چنانچه برابر نباشد بدین معنی است که مقدار آن تغییر کرده است روال *textChange* فراخوانی می کنیم .

#### ۱۱- رویداد *MoveComplete*

```
Private Sub Adodc1_MoveComplete(ByVal adReason As  
ADODB.EventReasonEnum, ByVal pError As ADODB.Error,  
adStatus As ADODB.EventStatusEnum, ByVal pRecordset As  
ADODB.Recordset)  
If pRecordset.EOF Or pRecordset.BOF Then  
Else
```

*Call ControlsTagLoad(pRecordset)*

*End If*

*End Sub*

این رویداد مختص کنترل داده *Ado* می باشد که بعد از انجام این رویداد ، محتویات کنترل ها را به صورت بروز آوری شده در اختیار داریم بنا براین در این رویداد عمل مقدار دهی خاصیت *Tag* کنترل را انجام می دهیم در این روال ابتدا چک می کنیم که در ابتدا یا انتهای فایل هستیم یا خیر بنابراین اگر در ابتدا یا انتهای فایل باشیم یعنی بر روی یک رکورد مجازی باشیم نیازی نیست که کاری را انجام دهیم در غیر این صورت روال *ControlsTagLoad* را فراخوانی می کنیم که به این روال یک پارامتر که در واقع همان رکوردستی است که ما در اختیار داریم و به وسیله آن می توان به رکورد جاری دسترسی پیدا کرد .

#### ۱۲- زیر برنامه **ControlsTagLoad**

*Private Sub ControlsTagLoad(pRecordset As*

*ADODB.Recordset)*

*Dim oCtrl As Control*

*If pRecordset.EOF Or pRecordset.BOF Then*

*For Each oCtrl In Me.Controls*

*If TypeOf oCtrl Is TextBox Then*

*oCtrl.Tag = ""*

*ElseIf TypeOf oCtrl Is CheckBox Then*

*oCtrl.Tag = vbUnchecked*

*End If*

*Next*

*Else*

*For Each oCtrl In Me.Controls*

*If TypeOf oCtrl Is TextBox Then*

*oCtrl.Tag = )*

*Field2Str(pRecordset(oCtrl.DataField).Value*

*ElseIf TypeOf oCtrl Is CheckBox Then*

```
oCtrl.Tag =
Field2CheckBox(pRecordset(oCtrl.DataField).Value)
End If
Next
End If
End Sub
```

در این زیر برنامه یک پارامتر به نام *pRecordset* از نوع رکوردست داریم که بوسیله آن رکوردست را دریافت می کنیم همچنین یک متغیر از نوع کنترل با نام *oCtrl* تعریف کردیم بعد از این مرحله در یک دستور شرطی چک کردیم که آیا در ابتدا یا انتهای رکوردست قرار داریم یا خیر در صورتی که این شرط صادق باشد برای هر یک از کنترل های موجود در فرم که بوسیله *Controls Collection* فرم به آن دسترسی داریم یک حلقه را تکرار کرده و در این حلقه وضعیت هر یک از کنترل ها را چک می کنیم

اگر کنترل از نوع *Text Box* باشد مقدار خاصیت *Tag* آن را به رشته خالی ست می کنیم و اگر کنترل از نوع *ChekBox* باشد مقدار خاصیت *Tag* آن را به *vbUnchecked* یعنی حالتی که *ChekBox* انتخاب نشده تنظیم می کنیم اما قسمت دوم شرط زمانی اجرا می شود که ما بر روی یک رکورد واقعی و معتبر قرار داشته باشیم در این حالت باز هم برای کلیه کنترل های موجود بر روی فرم چک کردیم که اگر کنترل *Text Box* یا *ChekBox* باشد خصوصیت *Tag* به مقدار فیلد مورد نظر آن کنترل تنظیم شود ( به وسیله فراخوانی تابع *Field2Str* )

### ۱۳- زیر برنامه حذف

```
Private Sub cmddelete_Click( )
Dim BoolOk As Integer
Beep
BoolOk = MsgBox( ..... & Adodc1.Recordset.Fields(2) _&
....., vbYesNo + vbQuestion, ..... )
If BoolOk = vbYes Then
Adodc1.Recordset.Delete
```

*Call DataReposition*

*End If*

*End Sub*

این زیر برنامه به منظور حذف یک رکورد عمل می کند در ابتدا یک متغیر با نام *BoolOk* از نوع *Integer* به منظور نگهداری پاسخ کاربر تعریف شده است و سپس قبل از نمایش پیغام به کاربر بوق سیستم توسط دستور *Beep* به صدا در می آید و در ادامه پیغامی با مضمون این که « آیا مایلید اطلاعات مربوط به فیلد مورد نظر را حذف نمایید ؟ » و سپس در یک دستور کنترلی چنانچه کاربر پاسخ بله را انتخاب کند رکورد مورد نظر حذف شده و زیر برنامه *DataReposition* فراخوانی می شود .

۱۴- زیر برنامه جستجوی فرم پرسنل

*Private Sub cmdfind\_Click()*

*lbllist.Visible = True*

*Image1.Visible = True*

*Image2.Visible = True*

*lstname.Visible = True*

*End Sub*

این زیر برنامه برای جستجو و نمایش کلیه فیلدهای یک پرسنل به وسیله انتخاب نام خانوادگی منظور گردیده است . و خاصیت *Visible* کنترل های *Image1* ، *Image2* ، *lstname* ، به *True* تنظیم شده است تا نمایش داده شوند .

۱۵- زیر برنامه خروج

*Private Sub cmdclose\_Click()*

*If cmdclose.Caption = "....." Then*

*Unload Me*

*Else*

*Adodc1.Recordset.CancelUpdate*

*If Adodc1.Recordset.EditMode <> adEditAdd Then*

*Adodc1.Recordset.Move 0*

*End If*

*Call ToggleButtons*

```
End If  
End Sub
```

۱۶- زیر برنامه مرتب سازی بر اساس شماره پرسنلی

```
Private Sub optsortid_Click()  
If optsortid.Value Then  
Adodc1.Recordset.Sort = " ID "  
End If  
End Sub
```

در این روال در یک دستور شرطی در صورتیکه *Option button* مربوطه فعال شود کنترل داده ، داده ها را بر اساس فیلد *ID* (شماره پرسنلی) مرتب می کند .

۱۷- زیر برنامه مرتب سازی بر اساس نام

```
Private Sub optsortnm_Click()  
If optsortnm.Value Then  
Adodc1.Recordset.Sort = " name "  
End If  
End Sub
```

در این روال در یک دستور شرطی در صورتیکه *Option button* مربوطه فعال شود کنترل داده ، داده ها را بر اساس فیلد *name* ( نام) مرتب می کند .

۱۸- زیر برنامه مرتب سازی بر اساس نام خانوادگی

```
Private Sub optsortfmly_Click()  
If optsortfmly.Value Then  
Adodc1.Recordset.Sort = " family "  
End If  
End Sub
```

در این روال در یک دستور شرطی در صورتیکه *Option button* مربوطه فعال شود کنترل داده ، داده ها را بر اساس فیلد *family* ( نام خانوادگی) مرتب می کند .

۱۹- زیر برنامه دریافت شماره پرسنلی فقط به صورت عدد صحیح

```
Private Sub Txtid_KeyPress(KeyAscii As Integer)
```



```
Select Case KeyAscii  
Case vbKeyDelete ' Delete Key  
Case vbKeyBack ' Backspace Key  
Case 48 To 57 ' Number 0-9  
Case Else  
Beep  
KeyAscii = 0 ' Cancels the keystroke  
End Select  
End Sub
```

در این روال در یک دستور *Select* در صورتی که کلید فشرده شده توسط کاربر یکی از کلید های *Delete* ، *Backspace* ، اعداد بین ۰ تا ۹ باشد ان را نمایش می دهد در غیر این صورت بوق سیستم به صدا در آمده و در جعبه متن مربوطه کاراکتری نمایش داده نمی شود و در واقع کد اسکی صفر قرار داده می شود .

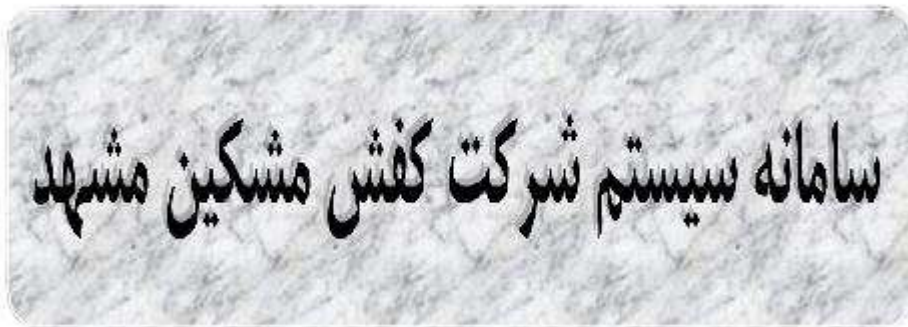
#### ۶-۴) حداقل امکانات مورد نیاز برای اجرای نرم افزار

برای راه اندازی این نرم افزار به امکانات زیادی نیاز نیست . یک سیستم عامل ویندوز ۹۸ یا بالاتر به همراه مدیر موتور پایگاه داده ای Access و فایل اجرایی برنامه . حداقل امکانات سخت افزاری مورد نیاز عبارتند از :

CPU : Pentium  
RAM : 64 MB  
H.D.D : 1GB  
VGA : 16MB

پیوست

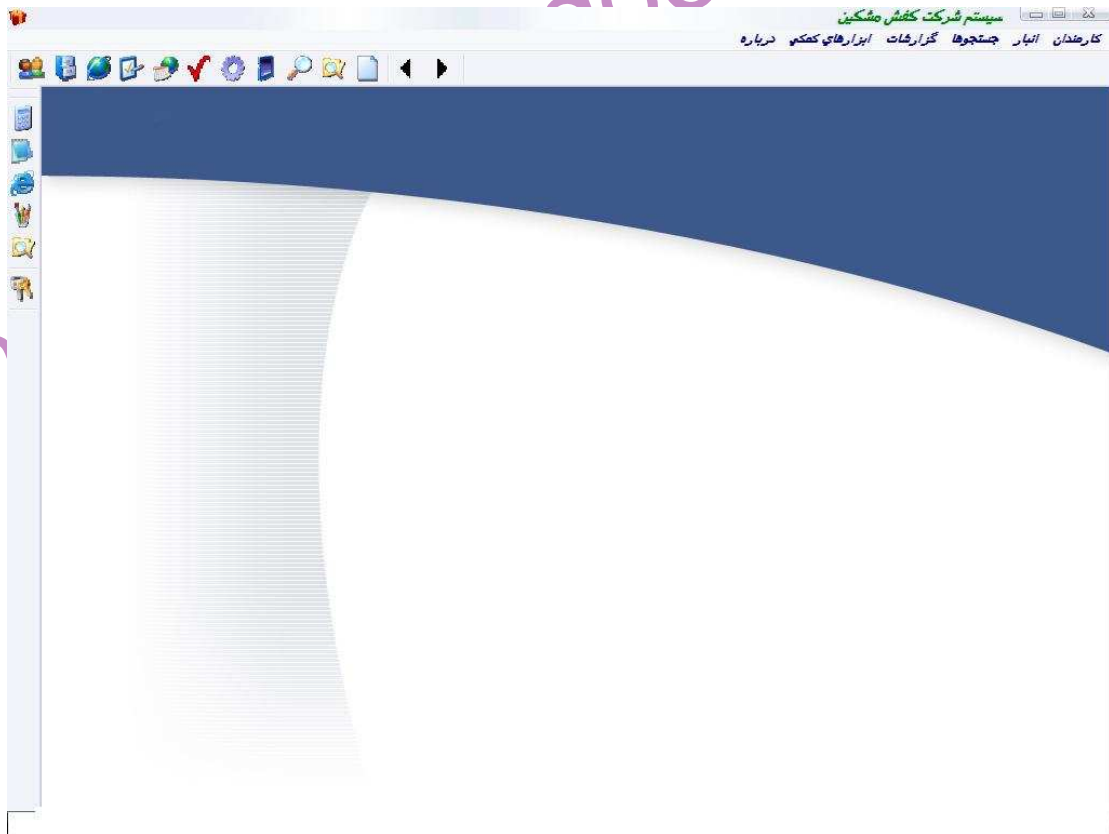
فرم های طراحی شده برنامه



لطفاً کمی صبر کنید ... 60

فرم ۱-۱) فرم بارگذاری برنامه

فرم ۱-۲) فرم رمز عبور برنامه



فرم ۱-۳) فرم اصلی برنامه

ورود اطلاعات

شماره پرسنلی

نام

نام خانوادگی

جنسیت

شماره شناسنامه

محل تولد

آدرس

توضیحات

تاریخ استخدام

محل سکونت

تلفن

تلفن همراه

مرتب سازی بر اساس

شماره پرسنلی  نام  نام خانوادگی

جديد  لغو  حذف  جستجو  خروج

ابتدا  قبلی  بعدي  انتها

### فرم ۴-۱) فرم ثبت مشخصات پرسنل

ثبت کارها

ایام هفته: [ ]

تاریخ: [ ]

کد پرسنلی: [ ]

نوع کار: [ ]

مدل: [ ]

رنگ: [ ]

تعداد: [ ]

دستمزد تعیین شده برای هر جفت: [ ]

دستمزد کل: [ ]

دریافتی روزانه: [ ]

توضیحات: [ ]

لغو ذخیره

انتها قبلی بعدی

### فرم ۵-۱) فرم ثبت کار های روزانه پرسنل

سیستم شرکت گلش مشکین - [Form1]

کارمندان انبار جستجوها گزارشات ابزارهای کمکی درباره

### فرم سفارشات

<input type="text"/>	قیمت واحد	<input type="text"/>	ردیف
<input type="text"/>	قیمت کل	<input type="text"/>	شماره فاکتور
<input type="text"/>	تاریخ سفارش	<input type="text"/>	سفارش دهنده
<input type="text"/>	تاریخ تحویل	<input type="text"/>	نوع سفارش
<input type="text"/>	قرار تحویل	<input type="text"/>	مدل
<input type="text"/>	سفارش گیرنده	<input type="text"/>	رنگ
		<input type="text"/>	تعداد
		<input type="text"/>	توضیحات

فرم ۶-۱) فرم سفارشات شرکت

سیستم شرکت گلش مشکین

کارمندان انبار جستجوها گزارشات ابزارهای کمکی درباره

### مواد اولیه انبار ورودی

<input type="text"/>	نام خریدار	<input type="text"/>	شماره ردیف
<input type="text"/>	نام خانوادگی خریدار	<input type="text"/>	شماره فاکتور
<input type="text"/>	تاریخ خرید	<input type="text"/>	نوع جنس
<input type="text"/>	تاریخ تحویل	<input type="text"/>	رنگ
<input type="text"/>	شرکت یا فروشگاه طرف قرارداد	<input type="text"/>	تعداد
<input type="text"/>	صورت پرداخت	<input type="text"/>	قیمت واحد
		<input type="text"/>	قیمت کل
		<input type="text"/>	توضیحات

مشاهده لیست چک ها

فرم ۷-۱) فرم مواد اولیه انبار ورودی

سیستم شرکت کفش مشکین  
کارمندان انبار جستجوها گزارشات ابزارهای کمکی درباره

### تولیدات نهایی کارخانه

<input type="text"/>	ردیف
<input type="text"/>	مدل
<input type="text"/>	رنگ
<input type="text"/>	تعداد
<input type="text"/>	تاریخ ورود به انبار
<input type="text"/>	توضیحات

لغو جستجو ذخیره حذف

انتها بعدی قبلی ابتدا

### فرم ۸-۱) فرم تولیدات نهایی شرکت

سیستم شرکت کفش مشکین  
کارمندان انبار جستجوها گزارشات ابزارهای کمکی درباره

### مواد اولیه انبار خروجی

<input type="text"/>	شماره ردیف
<input type="text"/>	شماره فاکتور
<input type="text"/>	نوع جنس
<input type="text"/>	رنگ
<input type="text"/>	تعداد
<input type="text"/>	قیمت واحد
<input type="text"/>	قیمت کل
<input type="text"/>	توضیحات

<input type="text"/>	شرکت یا فروشگاه طرف قرارداد
<input type="text"/>	مسؤل خروج
<input type="text"/>	تاریخ خروج
<input type="text"/>	علت خروج

خروج جستجو حذف ذخیره جدید

انتها بعدی قبلی ابتدا

### فرم (۹-۱) فرم مواد اولیه انبار ورودی

The screenshot shows a software window titled 'تولیدات کارخانه' (Factory Production) with a menu bar containing 'سیستم شرکت گلش مشکین' and 'کارندان انبار جستجوها گزارشات ابزارهای کمکی درباره'. The main area contains a form with the following fields:

- شماره ردیف (Row Number)
- مدل (Model)
- رنگ (Color)
- تعداد (Quantity)
- تاریخ خروج از انبار (Date of exit from warehouse)
- علت خروج (Reason for exit)
- مسئول خروج (Responsible for exit)
- توضیحات (Comments)

At the bottom of the form, there are two rows of buttons:

- Row 1: جدید (New), لغت (Cancel), حذف (Delete), جستجو (Search), خروج (Exit)
- Row 2: ابتدا (First), قبلی (Previous), بعدی (Next), انتها (Last)

### فرم (۹-۱) فرم تولیدات شرکت



سیستم شرکت کفش مشکین - کارمندان انبار جستجوها گزارشات ابزارهای کمکی درباره

### لیست کارهای ثبت شده

لطفا نام خانواده‌گی پرسنل مورد نظر را انتخاب نمایید

مظفری

عملکرد پیمان مظفری

ایام هفته	تاریخ	نوع کار	مدل	رنگ	تعداد	دستمزد تعیین شده	دستمزد کل	دریافتی را
پنج شنبه					۱۰۰	ریال ۱۰۰۰۰	ریال ۲۰۰۰۰	ریال ۶۰۰۰۰
چهارشنبه			ایتالیایی	سفید	۲۰	ریال ۵۰۰۰۰	ریال ۱۰۰۰۰۰۰	ریال ۰۰۰۰۰

از تاریخ تا تاریخ

مجموع دستمزد کل ۱۰۲۰۰۰۰  
مجموع دریافتی ۵۰۰۰۰۰  
ریال بدهکار است ۴۹۰۲۰۰۰

ثبت خروج

### فرم (۱-۱۰) فرم کارهای ثبت شده برای پرسنل

سیستم شرکت کفش مشکین - جستجوی سفارشات - کارمندان انبار جستجوها گزارشات ابزارهای کمکی درباره

جستجو بر اساس تاریخ

در تاریخ 13/ / / از تاریخ 13/ / / تا تاریخ 13/ / /

جستجو بر اساس مدل

مدل

جستجو بر اساس مدل و رنگ

مدل رنگ

ردیف	مدل	رنگ	تعداد	تاریخ	توضیحات
۱	کفش	مشکی	۱۰۰	۱۳۸۴/۵/۴	
۲	کیف	مشکی	۵۰	۱۳۸۴/۱۰/۱۵	
۳	کیف	سفید	۸۰	۱۳۸۴/۱۲/۲۵	
۴	کیف	آبی	۱۰۰	۱۳۸۴/۱۰/۵	

جستجو خروج

### فرم (۱-۱۱) فرم جستجوی تولیدات نهایی شرکت

سیستم شرکت کفش مشکی - جستجوی انبار ورودی

کارمندان انبار جستجوها گزارشات ابزارهای کمکی درباره

جستجو بر اساس نوع جنس  
 لطفا نوع جنس را وارد نمایید

جستجو بر اساس شماره فاکتور  
 لطفا شماره فاکتور را وارد نمایید

جستجو بر اساس جنس و رنگ  
 لطفا نوع جنس را وارد نمایید  
 لطفا رنگ جنس را وارد نمایید

جستجو بر اساس نام شرکت یا فروشگاه طرف قرارداد  
 لطفا نام شرکت یا فروشگاه طرف قرارداد را وارد نمایید

جستجو بر اساس تاریخ خرید و تحویل  
 لطفا تاریخ خرید را وارد نمایید  
 لطفا تاریخ تحویل را وارد نمایید

ردیف	شماره فاکتور	نوع جنس	رنگ	تعداد	قیمت واحد	قیمت کل	نام خریدار	نام خانوادگی خریدار	تاریخ
۱	۲۱۲	کیف	مشکی					رضوی	/۵/۸
۲	۵۴۴	کیف	مشکی					مظفری	/۳/۴
۳	۵۶۶	کفش	مشکی					رضایی	
۴	۲۱,۵۶۱							اسماعیل	

جستجو خروج

فرم ۱۲-۱) فرم جستجوی مواد اولیه انبار ورودی

سیستم شرکت کفش مشکی

کارمندان انبار جستجوها گزارشات ابزارهای کمکی درباره

فرم جستجوهای سفارشات

جمع مبلغ کل بر اساس...

جستجو بر اساس تاریخ

در تاریخ: / /  
 از تاریخ: / /  
 تا تاریخ: / /

جستجو بر اساس مدل

مدل: \_\_\_\_\_

جستجو بر اساس مدل و رنگ

مدل: \_\_\_\_\_ رنگ: \_\_\_\_\_

سفارش دهنده	نوع سفارش	شماره فاکتور	مدل	رنگ	تعداد	قیمت واحد	قیمت

جستجو

### فرم ۱۳-۱) فرم جستجوی سفارشات

The screenshot shows a web browser window with the title "سیستم شرکت کفش مشکین" and a URL "www.kandoocn.com". The main content area is titled "فرم جستجوی سفارشات" (Order Search Form). It contains several search filters and buttons:

- جستجو بر اساس تاریخ** (Search by date): Includes fields for "در تاریخ" (On date), "از تاریخ" (From date), and "تا تاریخ" (To date), each with a dropdown menu set to "13 / /". A "جمع مبلغ کل" (Total amount) button is below.
- جستجو بر اساس نام سفارش دهنده و تاریخ** (Search by order name and date): Includes a "نام خانوادگی سفارش دهنده" (Orderer's last name) field, a "جمع مبلغ کل" button, and date fields for "در تاریخ", "از تاریخ", and "تا تاریخ".
- جستجو بر اساس مدل** (Search by model): Includes a "مدل" (Model) field, a "جمع مبلغ کل" button, and another "مدل" field.
- جستجو بر اساس مدل و رنگ** (Search by model and color): Includes "مدل" (Model) and "رنگ" (Color) fields, a "جمع مبلغ کل" button, and another "مدل" field.

### فرم ۱۴-۱) فرم میزان دریافتی سفارشات



فرم ۱۵-۱) فرم گزارش مشخصات پرسنل

### منابع و مراجع

- آموزش ویزوال بیسیک ۶ در ۲۱ روز گریک پری؛ [مترجم] علیرضا زارع پور  
تهران؛ نص، ۱۳۸۳