

باسمه تعالی



دانشگاه آزاد اسلامی
واحد اصفهان (خوراسگان)
دانشکده فنی و مهندسی

پایان نامه برای دریافت درجه کارشناسی در رشته مهندسی کامپیوتر
گرایش نرم افزار

عنوان

سیستم اتوماسیون مشاور املاک

استاد راهنما

مهندس علی سلیمانی

نگارنده

سارا نصری

مهر ۱۳۹۹

تقدیم بہ

پدر بزرگوار و مادر مہربانم

کہ از نگاہشان صلابت

از رفقا رشان محبت

و از صبرشان ایستادگی آموختم

پاسکزاری

پروردگار مایاری کن تا دانش اندکم نه زردبانی باشد برای فزونی و تکبر و غرور، نه حلقه‌ای برای اسارت و نه دستایه‌ای برای تجارت، بلکه گامی باشد برای تجلیل از تو و تعالی ساختن زندگی خود و دیگران.

قبل از هر چیز، خداوند بزرگ را به خاطر لطفی که همواره شامل حال من نموده شاکرم. سپس، از زحمات استاد محترم راهنما، جناب آقای مهندس علی سلیمانی که نه تنها به عنوان استاد بلکه، همچون بھاری در تمام مراحل انجام این تحقیق از ر، بنموده و کمک های بی دریغ ایشان بهره مند شده ام، تشکر و قدردانی می کنم.

چکیده

این پروژه با عنوان سیستم اتوماسیون مشاوره املاک ، پروژه ای است که با بهترین نرم افزار های رایج طراحی و آماده شده است. هدف از ایجاد این پروژه مکانیزه کردن ثبت اطلاعات املاک، مالکین، خریداران، مستاجرین و قرارداد داد ها می باشد.

در فصل اول نگاه کوتاه و گذرا به هر یک نرم افزار های مورد کاربرد در این پروژه که شامل C# و SQL Server است دارد و در مرحله بعد سیستم را مورد تحلیل قرار میدهد چرا که در صورت نبود یک تحلیل مناسب از یک سیستم ، نمیتوان از آن سیستم به نحو اساسی استفاده نمود . در انتها با به کار گرفتن از علم نرم افزار و طراحی سیستم بانک اطلاعاتی ، این پروژه طراحی و ایجاد شده است.

ابتدا سیستم را تحلیل کرده و موجودیت ها و صفات و روابط بین آنها را در می آوریم و جداول آنها را در دیتابیس طراحی کرده و برای هر جدول فرم مربوطه را طبق صفات آنها طراحی کرده و برای هر کدام کد نویسی مربوط را انجام می دهیم.

در این پروژه به راحتی میتوانیم انواع محصولات را در سیستم ثبت ، ویرایش و حذف کنیم ، همچنین منشی سیستم میتواند لیست مالکین ، خریداران، مستاجرین ، املاک (خانه آپارتمان و زمین) و قرارداد ها را ببیند .

فهرست مطالب

عنوان

صفحه

۱- فصل اول کلیات پروژه	۳
۱-۱- بیان کلیات مسئله	۳
۱-۲- نیازمندی های پروژه	۱۰
۱-۳- نرم افزارهای استفاده شده در پیاده سازی سیستم	۱۰
۱-۳-۱- گذری بر نرم افزار C#	۱۱
۱-۳-۲- گذری بر نرم افزار پایگاه داده sql server	۱۲
۲- فصل دوم تحلیل و طراحی سیستم	۱۴
۲-۱- مقدمه	۱۴
۲-۲- نمودار مورد کاربرد (use case)	۱۵
۲-۳- نمودار فعالیت (Activity)	۱۶
۲-۴- نمودار توالی (Sequence Diagram)	۲۱
۲-۵- سناریوها	۲۵
۲-۶- نمودار کلاس (Class Diagram)	۲۹
۳- فصل سوم پیاده سازی سیستم	۳۰
۳-۱- مقدمه	۳۰
۳-۲- پایگاه داده SQL در پروژه	۳۲
۳-۲-۱- جداول تعریف شده	۳۴
۳-۲-۲- چگونگی ارتباط پروژه با پایگاه داده	۴۳
۳-۲-۳- نحوه پیاده سازی پروژه	۴۷
۴- فصل چهارم نصب و راهنمای پروژه	۷۱
۴-۱- نحوه نصب و راه اندازی پروژه	۷۱
۴-۲- راهنمای نرم افزار	۷۴
۴-۲-۱- صفحه ورود به برنامه	۷۴
۴-۲-۲- منوهای اصلی	۷۴
۴-۲-۳- پنجره های فرعی نرم افزار	۷۵
۴-۳- نتیجه گیری و پیشنهادات	۷۶
۴-۴- منابع و مراجع	۷۷

فهرست جداول

عنوان	صفحه
جدول (۱-۲) سناریو وارد کردن اطلاعات.....	۲۵
جدول (۲-۲) سناریو مدیریت کردن اطلاعات.....	۲۶
جدول (۳-۲) سناریو گزارش گیری.....	۲۷
جدول (۴-۲) سناریو جستوجو.....	۲۸

فهرست شکل ها

عنوان	صفحه
شکل (۱-۱) فرم نمونه ثبت قرار داد.....	۳
شکل (۱-۲) نمودار UseCase.....	۱۶
شکل (۲-۲) نمودار فعالیت ثبت مالکین.....	۱۸
شکل (۳-۲) نمودار فعالیت ویرایش مالکین.....	۱۹
شکل (۴-۲) نمودار فعالیت دیدن مالکین.....	۲۰
شکل (۵-۲) نمودار جستوجو کردن.....	۲۱
شکل (۶-۲) نمودار توالی ثبت نام.....	۲۲
شکل (۷-۲) نمودار توالی نمایش مالکین.....	۳۱
شکل (۸-۲) مودار تولی جستوجو.....	۲۳
شکل (۹-۲) نمودار توالی گزارش گیری.....	۲۴
شکل (۱۰-۲) نمودار کلاس.....	۲۹
شکل (۳-۳) روش اضافه کردن یک آیتم جدید به پروژه.....	۳۰
شکل (۴-۳) نحوه انتخاب صفحه با توجه به کاربرد مورد نیاز.....	۳۱
جدول (۷-۳) جدول مالکین.....	۳۴
جدول (۸-۳) جدول خانه.....	۳۶
جدول (۹-۳) جدول قرار داد خرید خانه.....	۳۷
جدول (۱۰-۳) جدول قرار داد خرید زمین.....	۳۸
جدول (۱۱-۳) جدول قرارداد اجاره خانه.....	۳۹
جدول (۱۲-۳) جدول قرارداد اجاره آپارتمان.....	۳۹
جدول (۱۳-۳) جدول آپارتمان.....	۴۰
جدول (۱۴-۳) جدول مستاجر.....	۴۱
جدول (۱۵-۳) جدول زمین.....	۴۱
جدول (۱۶-۳) جدول ورود.....	۴۲
شکل (۱۷-۳) اضافه کردن item جدید.....	۴۳
شکل (۱۸-۳) Ado.Net Entity Data Model اضافه کردن.....	۴۴
شکل (۱۹-۳) اضافه کردن مدل شی گرایى از Data base.....	۴۴
شکل (۲۰-۳) اضافه کردن new connection.....	۴۵
شکل (۲۱-۳) انتخاب نام Data base.....	۴۵
شکل (۲۲-۳) اضافه کردن جداول به پایگاه داده.....	۴۶

۴۶	شکل (۳-۲۳) مدل جدول
۴۷	شکل (۳-۲۴) صفحه ورود
۵۰	شکل (۳-۲۵) صفحه اصلی نرم افزار
۵۲	شکل (۳-۲۶) صفحه مدیریت مالکین
۵۵	شکل (۳-۲۷) صفحه افزودن مالک
۵۸	شکل (۳-۲۸) صفحه مدیریت خریداران
۶۱	شکل (۳-۲۹) صفحه افزودن خریدار
۶۳	شکل (۳-۳۰) صفحه مدیریت مستاجر
۶۶	شکل (۳-۳۱) صفحه افزودن مستاجر
۷۱	شکل (۴-۱) اقدام به Publish پروژه
۷۲	شکل (۴-۲) اجرا فایل Setup
۷۳	شکل (۴-۳) پنجره نصب نرم افزار
۷۴	شکل (۴-۲) صفحه ورود نرم افزار
۷۵	شکل (۴-۳) صفحه اصلی
۷۵	شکل (۴-۴) اطلاعات مالکین
۷۶	شکل (۴-۵) افزودن مالکین

فصل اول

کلیات پروژه

۱-۱- بیان کلیات مسئله

آنچه شما پیش رو دارید پروژه‌ای است برای اتوماسیون مشاور املاک که در این بخش به توجیه و توصیف این موضوع پرداخته خواهد شد.

هدف از طراحی و پیاده‌سازی این سیستم، نرم افزاریست که بتوان با استفاده و بهره‌گیری از آن در انجام امور املاک‌ها و ثبت قراردادها را سرعت بخشیده و از اتلاف وقت بیش از حد جلوگیری شود، همچنین راحتی بیشتری برای مالکین به ارمغان آورد.

این سیستم کلیه عملیات و امکانات مورد نیاز را که قبلاً در سیستم سنتی و کاغذی انجام می‌گرفت را با قابلیت‌های بسیار بیشتر و راحت‌تر و سریع‌تر در اختیار کاربران قرار می‌دهد.

استفاده از این سیستم مزایای زیادی دارد. قسمتی از این مزایا عبارت است از:

- ✓ جلوگیری از اتلاف وقت
- ✓ انجام هرچه سریع‌تر کارها و سرعت بخشیدن به امور گزارش‌گیری
- ✓ دسترسی سریع به اطلاعات ثبت شده و تغییر آنها به صورت آنی
- ✓ جلوگیری از اتلاف کاغذ و هدر دادن منابع طبیعی
- ✓ جلوگیری از بایگانی کردن اطلاعات مورد نیاز

۱-۲- نیازمندی های پروژه

برای انجام این پروژه ابتدا به بررسی سیستم ثبت قرار داد ها پرداخته شد و چندین فرم نمونه ثبت بررسی شد که شکل (۱-۱) نمونه ی مورد بررسی می باشند:

شماره قرار داد	
کد ملی مالک	
کد ملی خریدار	
توضیحات	
تاریخ	

شکل (۱-۱) فرم نمونه ثبت قرار داد

در طی بررسی های انجام شده به این نتیجه رسیدم که بهترین حالت برای یک نرم افزار مدیریت مشاور املاک استفاده از سیستمی است که کاربر بتواند به سادگی اقدام به ثبت ، حذف و ویرایش مالکین و املاک و..... نماید و با استفاده از این سیستم اقدام به گزارش گیری نموده، که اینکار به صورت خودکار است.

۱-۳- نرم افزارهای استفاده شده در پیاده سازی سیستم

برای ساخت نرم افزار، از متدولوژی های خاصی می توان کمک گرفت که کار مهندسی نرم افزار را هماهنگ و منظم و تحت کنترل مدیر و کارشناس در آورد. از جمله متدولوژی ها، متدولوژی RUP می باشد که این متدولوژی نیز همانند سایر متدولوژی ها شامل چهار مرحله اصلی اکتساب، تشریح، ایجاد و تحول است. ما با بهره گیری از این متدولوژی در مرحله ایجاد یا پیاده سازی سیستم از نرم افزار Visual Studio استفاده کردیم. برای پایگاه داده در این پروژه SQLServer استفاده کنیم همچنین نرم افزار Visual Studio نصب خواهد شد.

۱-۳-۱ - گذری بر نرم افزار C#

زبان سی شارپ، یک زبان برنامه‌نویسی چند الگویی و منظم شده مدل‌های تابعی، امری، عمومی، شیء‌گرا و جزگرا و در بستر دات نت می‌باشد. این زبان توسط مایکروسافت و جزئی از دات نت به وجود آمد و بعداً استانداردهای ECMA و ISO را نیز دربر گرفت. سی شارپ یکی از ۴۴ زبان برنامه‌نویسی است که توسط زمان اجرای زبان مشترک از چارچوب دات نت پشتیبانی می‌شوند و در همه جا به وسیله مایکروسافت ویژوال استودیو شناخته می‌شود.

زبان سی شارپ با قدرت و در عین حال سطح بالایی خود توانسته توجه بسیاری از برنامه نویسان را به خود جلب کند.

این زبان برپایه سادگی، مدرن بودن، همه منظوره و شیء‌گرا بودن ساخته شد. آندرس هجلزبرگ، طراح زبان برنامه‌نویسی دلفی، سرپرستی تیم طراحان زبان سی شارپ را بر عهده داشت. این زبان دارای دستوری شیء‌گرا مشابه C++ است و به شدت از زبان‌های جاوا و دلفینیاژمندمدرک تأثیر پذیرفته است. در ابتدا نام این زبان COOL بود که مخفف C like Object Oriented Language بود، هر چند در ژوئیه ۲۰۰۰، زمانی که مایکروسافت پروژه را عمومی اعلام کرد، اسم آن به سی شارپ تغییر پیدا کرد.

استاندارد ECMA این اهداف طراحی زبان را برای سی شارپ برآورده می‌سازد:

سی شارپ یک زبان برنامه‌سازی ساده، مدرن، برای اهداف عمومی و شیء‌گرا است.

به دلیل اهمیت داشتن موضوع نیرومندی و دوام و بهره‌وری برنامه‌نویس، زبان دارای چک‌کننده Strong Type، چک‌کننده مرزهای آرایه، تشخیص حالت‌هایی که یک متغیر مقداردهی اولیه نشده است، قابلیت انتقال کدها و Garbage Collection خودکار است.

این زبان برای استفاده در اجزای توسعه نرم‌افزار برای دستیابی به مزایای سامانه‌های توزیعی در نظر گرفته شده است.

قابلیت انتقال برنامه‌نویس بسیار مهم است، خصوصاً برای آن دسته از برنامه‌نویسانی که با زبان‌های C و C++ آشنا هستند.

پشتیبانی از این زبان برای بین‌المللی شدن بسیار مهم است. زبان سی شارپ برای نوشتن برنامه‌ها برای سامانه‌های تعبیه شده و میزبان در نظر گرفته شده است، سیستم عامل‌های پیچیده بسیار بزرگ گرفته تا توابع اختصاصی بسیار کوچک. هر چند برنامه‌های نوشته شده با سی شارپ طوری هستند که از لحاظ حافظه و پردازنده مورد نیاز مقرون به صرفه باشند، ولی خود زبان از لحاظ اندازه و کارایی به خوبی زبان‌های C و اسمبلی نیست. در هر نرم افزاری که بتوان کد های سی شارپ را وارد کرد بخشی وجود دارد که برای افراد مبتدی در برنامه نویسی است این بخش console.aplication نام دارد.

۲-۳-۱ - گذری بر نرم افزار پایگاه داده sqlserver

SQL Server یک سیستم مدیریت پایگاه داده رابطه ای از مایکروسافت است که هدف اصلی آن ذخیره و مدیریت اطلاعات و داده هاست. این سیستم از عملیات مختلف تجاری، عملیات تحلیل داده و پردازش معاملات پشتیبانی می کند. هر دیتابیس محلی برای ذخیره اطلاعات است اما SQL Server امکان مدیریت آن ها را نیز فراهم می کند. SQL مخفف Structured Query Language است.

مایکروسافت اس کیو ال سرور یا به عبارت صحیح تر سی کو ال سرور کاربرد های زیادی در کسب و کار دارد. اولین و بارزترین کاربرد، پایگاه داده برای ذخیره و مدیریت اطلاعات است. البته کسب و کار هایی که اطلاعات حساس مانند اطلاعات شخصی و کارت های بانکی را نگهداری می کنند، از امنیت پیشرفته آن نیز بهره خواهند برد. همچنین این سیستم امکان به اشتراک گذاری فایل های دیتا در شبکه را فراهم کرده است. استفاده از اس کیو ال سرور برای

پردازش داده ها، باعث افزایش سرعت پردازش می شود و بدین وسیله، امکان اجرای پردازش های بزرگتر در زمان کمتر ایجاد می شود.

اس کیو ال سرور محصولی از کمپانی مایکروسافت است که برای ذخیره سازی و مدیریت اطلاعات در نظر گرفته شده است. به شکل دقیق تر، اس کیو ال سرور یک "سیستم مدیریت پایگاه داده های رابطه ای" (RDSM) است. می توان این نام را به دو بخش مجزا تقسیم نمود. اول اینکه اطلاعات در پایگاه داده های رابطه ای ذخیره می شود و دوم اینکه اس کیو ال سرور یک سیستم مدیریت کامل است و تنها یک دیتابیس نیست. SQL خود به معنای **Structured Query Language** (زبان ساختارمند کوئری ها) است. اس کیو ال، زبانی برای مدیریت و اداره کردن سرور دیتابیس است.

فصل دوم

تحلیل و طراحی سیستم

۲-۱- مقدمه

در فرهنگ رایانه، واسط گرافیکی کاربر یا واسط نگاره‌ای کاربر، مجموعه‌ای از نشانه‌های گرافیکی نمایش داده شده بر روی یک نرم‌افزار است که در آن کاربر به جای تایپ فرمان‌های بلند و پیچیده از اعلان فرمان، با اشاره بر نمایشهای تصویری بر روی صحنه تصویر، پرونده‌ها، برنامه‌ها یا فرمان‌های گوناگون را انتخاب می‌کند.

برنامه‌های کاربردی که در ویندوز اجرا می‌شوند، از مجموعه ثابتی از فهرست‌های گزینش عمودی، جعبه‌های محاوره‌ای و سایر عناصر گرافیکی مانند نوارهای مرور (به انگلیسی: scroll bar) و شمایل‌ها (به انگلیسی: icon) استفاده می‌کنند.

در این پروژه از نرم‌افزار visual studio برای طراحی نرم افزار و از نرم افزار sql server برای طراحی سایر جداول استفاده شده است.

۲-۲- نمودار مورد کاربرد (use case)

این نمودار جز نمودارهای پویا می باشد. بسیاری از پروژه ها با این نمودار شروع می شوند که برای شناخت نیازمندیهای موجود و برای شناخت افعال و کنندگان کار (کاربران) در سیستم است که نشان می دهد. با ترسیم نمودار Use Case می توان یک سیستم را به زیر سیستم های کوچکی تقسیم نمود و برای هر یک، نمودار Use Case جداگانه ترسیم کرد و سپس ارتباط بین زیر سیستم ها را برقرار نمود. اجزاء نمودار مورد کاربرد:

۱- بازیگر (Actor)

فرد، فرایند یا موجودیتی خارجی که با سیستم مادر ارتباط است گفته می شود. برای نمایش آن در نمودار از نماد آدمک استفاده می کنیم.

۲- مورد کاربرد (use case)

به عملی که در سیستم انجام می گیرد، گفته می شود. عبارتی انتظاراتی که از سیستم داریم توسط use case نشان داده میشوند. از نماد بیضی برای use case استفاده می کنیم و در داخل آن یاد رزیبی عملی که انجام میشود را می آوریم.

روابط Use Case ها

۱- رابطه تناظر : توسط خط مستقیم

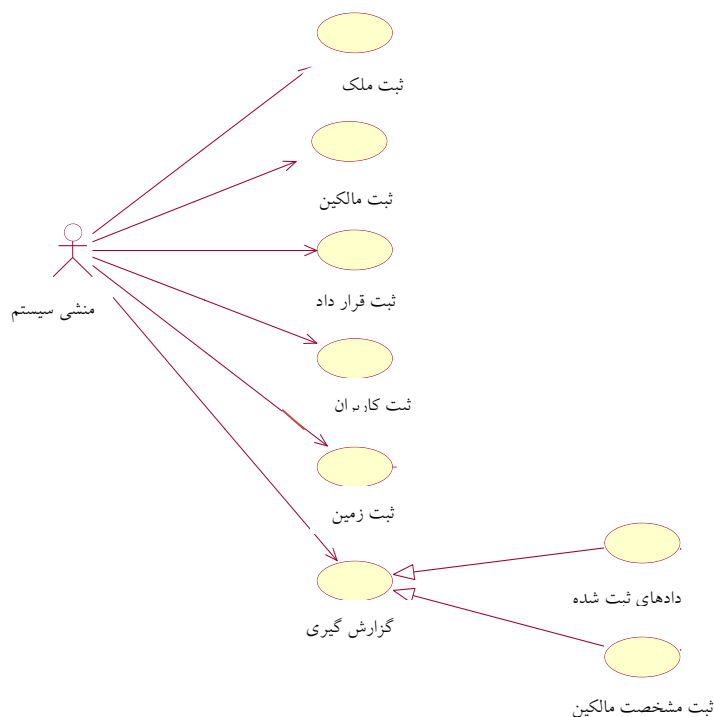
۲- **رابط بسط:** نشان میدهد یک use case ممکن است به طور اختیاری توسط قابلیت های موجود در use case دیگر بسط داده شود. این رابطه توسط یک پیکان بریده بریده جهت دار نشان داده میشود و حتما کلمه extend در بالای پیکان نوشته شود.

۳- **رابطه مشمول:** این رابطه بهین معناست که انجام این عمل شامل انجام عملیات های دیگر است و همانند رابطه قبل توسط پیکان بریده بریده جهت دار نمایش داده میشود با این تفاوت که روی پیکان کلمه include نوشته میشود.

۴- **رابطه تعمیم:** گاهی ممکن است بیشتر از یک نسخه از use case موجود باشد و نسخه های مختلف فعالیت های مشترکی داشته باشند که با راهکار وراثت پیاده سازی میشود. رابطه تعمیم هم بین عامل ها و هم بین use case ها وجود دارد.

باعلامت پیکان جهت دار نمایش داده میشود و طرفای ساده سازی استفاده میگردد.

در شکل (۱-۲) نمودار Use Case اتوماسیون مشاور املاک آمده است.



شکل (۱-۲) نمودار use case

۲-۳- نمودار فعالیت (Activity)

در این نمودار چگونگی جریان انجام یک کار صرف نظر از فاعل آن مشخص می شود. بر خلاف نمودارهای همکاری که فاعلان کار (Actors) در جریان انجام کار وجود دارند. این نمودار را می توان برای شرح User و یا هر یک از افعال (Operation) کلاسها ترسیم نمود. نمودارهای فعالیت بیشتر برای مدل کردن یک عملیات مورد استفاده قرار می گیرد، یعنی گاهی اوقات که یک عملیات پیچیده می شود، می توان از این مدل برای توضیح بیشتر استفاده

کرد. این نمودار شباهت فراوانی به فلوچارت دارد و از لحاظ معنایی نیز همان مفهوم را دنبال می‌کند.

در مدل شی گزایی از این نمودار کمتر استفاده میشود زیرا همانطور که گفتیم بیشتر برای مدل سازی عملیاتها از این نمودار استفاده میشود، حال آنکه تمرکز برنامه شی گرا عمدتاً روی اشیاء است. با این وجود شما به عنوان یک طراح، هر گاه لازم دانستید از این نمودار برای شرح یک UseCase یا متد از آن استفاده کنید. این نمودار برای افرادی که به روش Process Oriented برنامه مینویسند بیشترین کاربرد را در مدل سازی سیستم پیدا میکند.

هر یک از اجزای نمودار فعالیت به صورت زیر است:

۱- حالت شروع و پایان (start/end state): محلی که جریان کار از آنجا آغاز میگردد و با مرحله ای که جریان کار در آن پایان می‌رسد و نماد آغاز یا شروع کار با یک دایره توپر و گره پایان با دایره تو خالی نمایش داده میشود.

۲- Swim lane: برای مشخص کردن این که هر قسمت از فعالیت توسط چه کسی یا چه چیزی انجام میشود از Swim lane ها استفاده می‌گردد که به شکل نوارهای عمودی صفحه نمودار را تقسیم بندی میکند. هر فعالیتی که باید توسط مجری خاص انجام گیرد در نوار مربوط به آن مجری رسم میگردد.

۳- خطوط همگام سازی (synchronization bar): به منظور نمایش فعالیت هایی که به طور همزمان و موازی یکدیگر انجام می‌گیرند، از این خطوط استفاده میشود که به دو دسته fork و join تقسیم بندی میشوند. fork مکانی را نشان میدهد که در آن جریان کار به چند شاخه تقسیم میگردد. join مکانی را نشان میدهد که در آن چند شاخه کنترل به یکدیگر ملحق میگردد تا جریان کنترل ادامه یابد.

۴- انتقال (Transition): به منظور مشخص کردن ترتیب اجرا فعالیت ها از Transition استفاده میشود.

۵- شاخه (Branch): به منظور فراهم آمدن امکانی برای اجرای شرطی فعالیت ها از نمادی به نام شاخه استفاده میگردد.

۶-عمل (Action):یک واحد انجام کار است که بایستی اجرا شود و با علامت بیضی نمایش داده میشود.

در شکل (۲-۲) نمودار فعالیت ثبت مالکین آمده است ،همان طور که مشاهده مینمائید کاربر پس از زدن بر روی گزینه ثبت مالکین اقدام به ثبت مالکین مینماید.

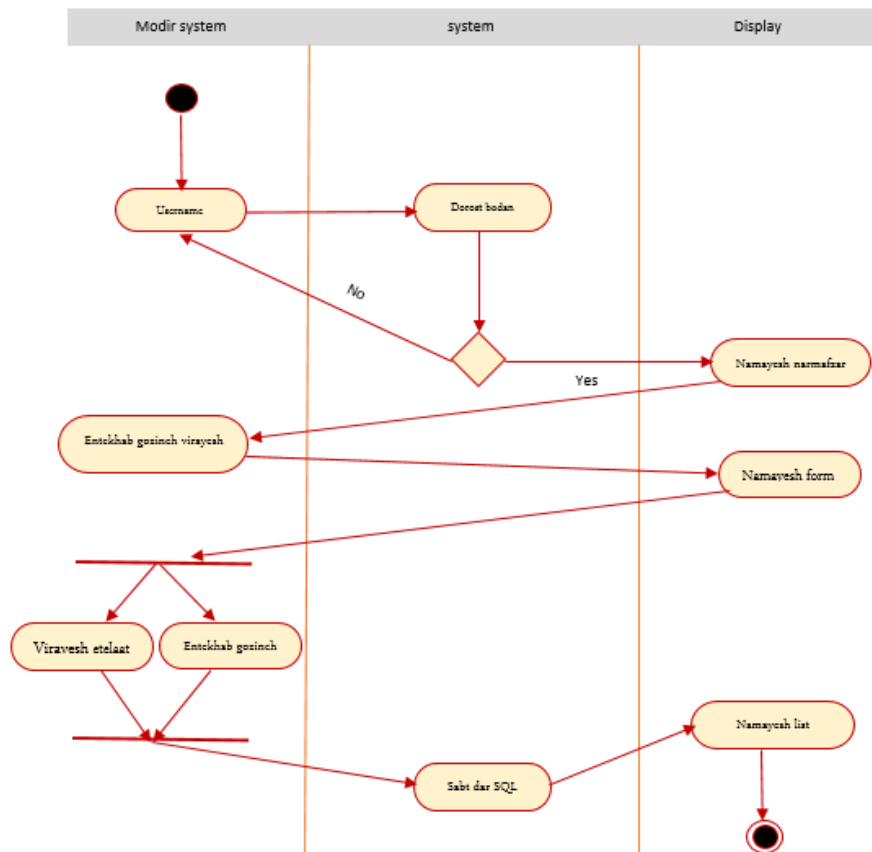
نمودار فعالیت ثبت (مالکین ، خریداران ، مستاجرین)و ثبت تمامی املاک(زمین ، خانه ، آپارتمان)نیز مانند نمودار ثبت مالکین می باشد.



شکل ۲-۲ نمودار فعالیت ثبت مالکین

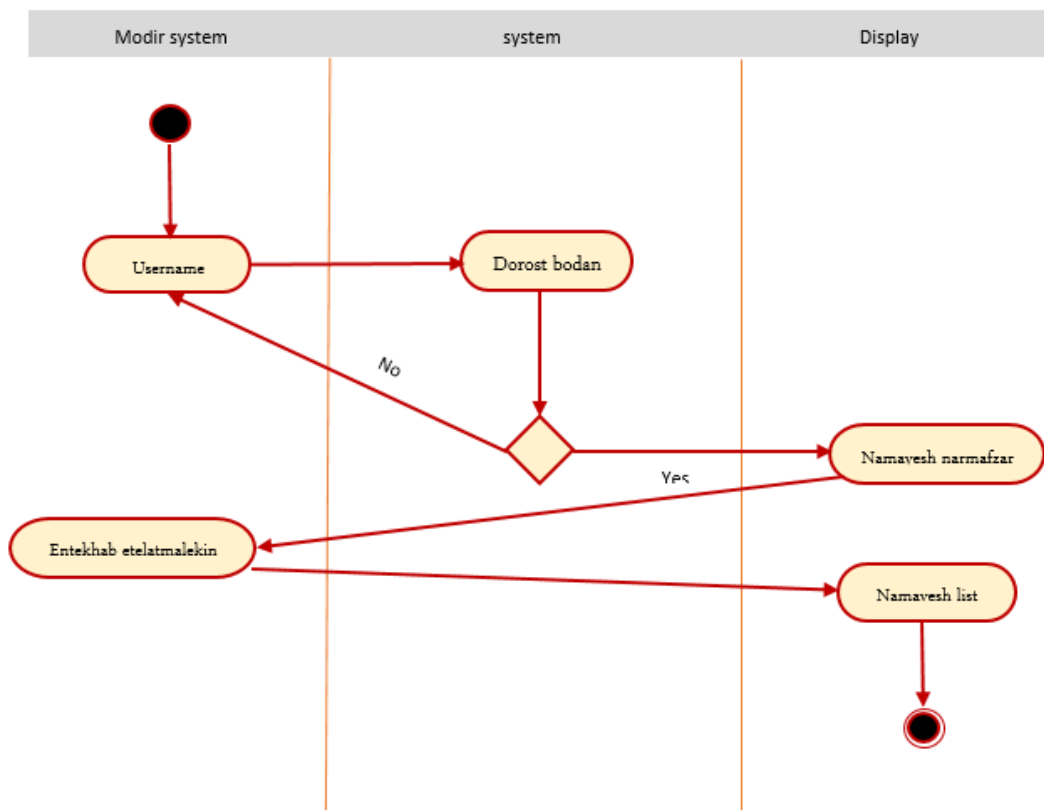
در شکل (۳-۲) نمودار فعالیت گزارش گیری از مالکین آمده است ، همانطور که مشاهده مینمائید.

کاربر پس از زدن بر روی گزینه گزارش گیری اقدام به گزارش گیری از مالکین مینماید.



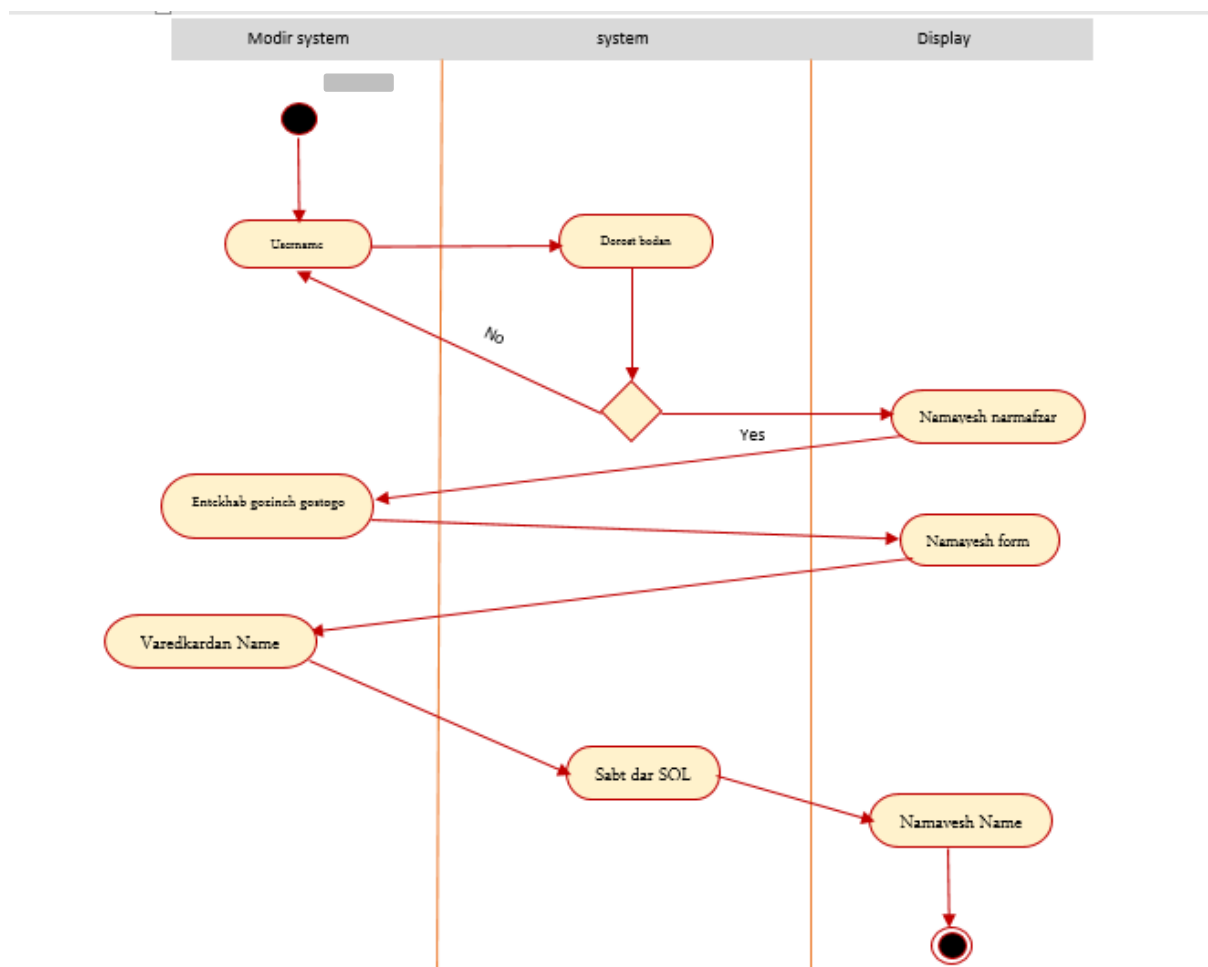
شکل (۳-۲) نمودار فعالیت ویرایش مالکین

در شکل (۴-۲) نمودار فعالیت دیدن مالکین آمده است، همانطور که مشاهده مینمائید منشی سیستم پس از زدن بر روی گزینه دیدن مالکین میتواند مالکین ثبت شده در نرم افزار را مشاهده کند.



شکل ۴-۲ نمودار فعالیت دیدن مالکین

در شکل (۵-۲) نمودار فعالیت جستجو کردن آمده است، همانطور که مشاهده مینمائید منشی سیستم پس از زدن بر روی گزینه جستجو کردن و وارد کردن نام یا کد ملی میتواند خروجی مورد نظر خود را ببیند.



شکل ۲-۵ نمودار جستجو کردن

۲-۴- نمودار توالی (Sequence Diagram)

این نمودار به طور گرافیکی نمایش می‌دهد که اشیاء چگونه از طریق پیام‌ها با یکدیگر تعامل می‌کنند. به عبارت دیگر نشان می‌دهد که پیام‌ها چگونه میان اشیاء ارسال و دریافت شده، و این تبدلات با چه ترتیبی صورت می‌گیرند. این نمودار جزء نمودارهای پویا می‌باشد و به طور کلی برای مدل‌سازی تعاملات اشیایی که برحسب توالی زمان ترتیب یافته‌اند مورد استفاده قرار می‌گیرند.

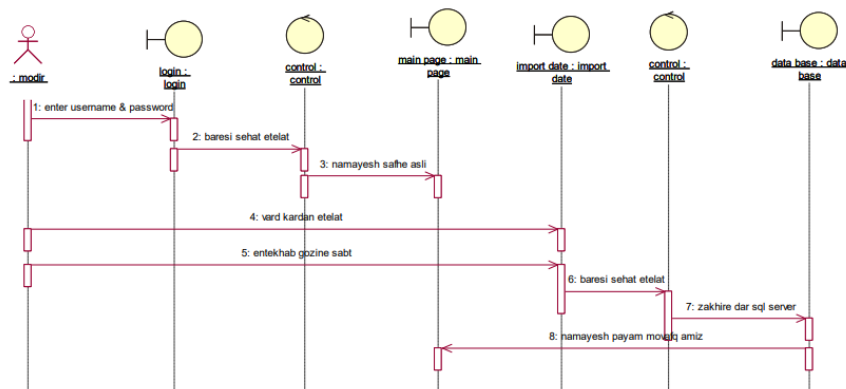
اجزای نمودار توالی عبارتند از:

- ۱- خطوط حیات: نمونه‌های شرکت‌کننده، در یک نمودار ترتیبی با یک خط عمودی نقطه چین و یک چهار گوش در بالای آن نمایش داده می‌شوند که نشان‌دهنده یک شیء است.

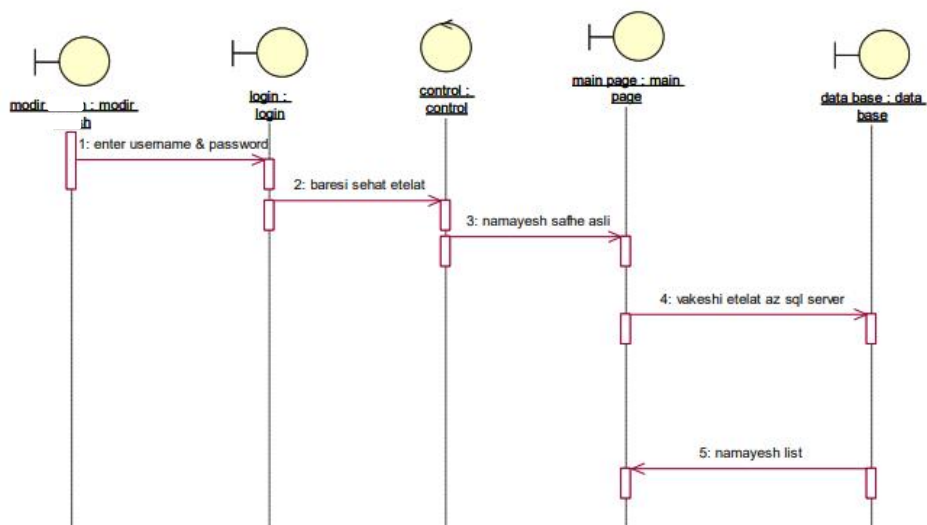
- ۲- پیام (message): نوعی برقراری ارتباط است که زمانی رخ می‌دهد که یک شیء به منظور درخواست اطلاعات یا عملکردی خاص، اقدام به فراخوانی متدی (رفتاری) از شیء دیگری مینماید همچنین پیام‌ها با استفاده از پیکان نشان داده میشوند.
- ۳- قاب: نمودارهای توالی در قاب رسم میشوند. قاب‌ها که در گوشه بالایی سمت چپ آن یک عنوان (نام تعامل) قرار دارد. پارامترهای تعامل و مقدار برگشتی نیز میتواند در عنوان قرار گیرد.

رسم نمودار توالی (Sequence Diagram)

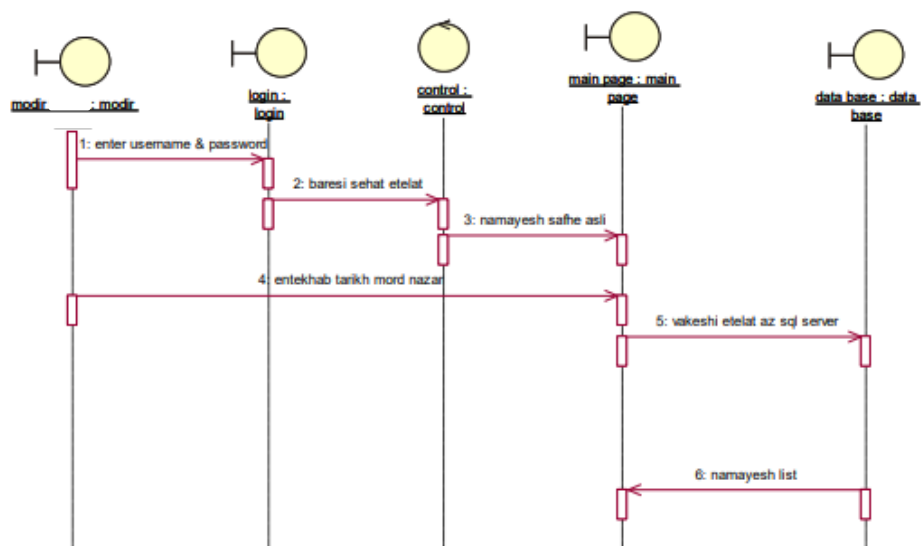
نمودار توالی ثبت مالکین، ثبت انواع ملک‌ها و تعریف قرار داد‌های طرف قرار داد مشابه یکدیگر است یا این تفاوت که اطلاعات وارد شده با هم متفاوت هستند.



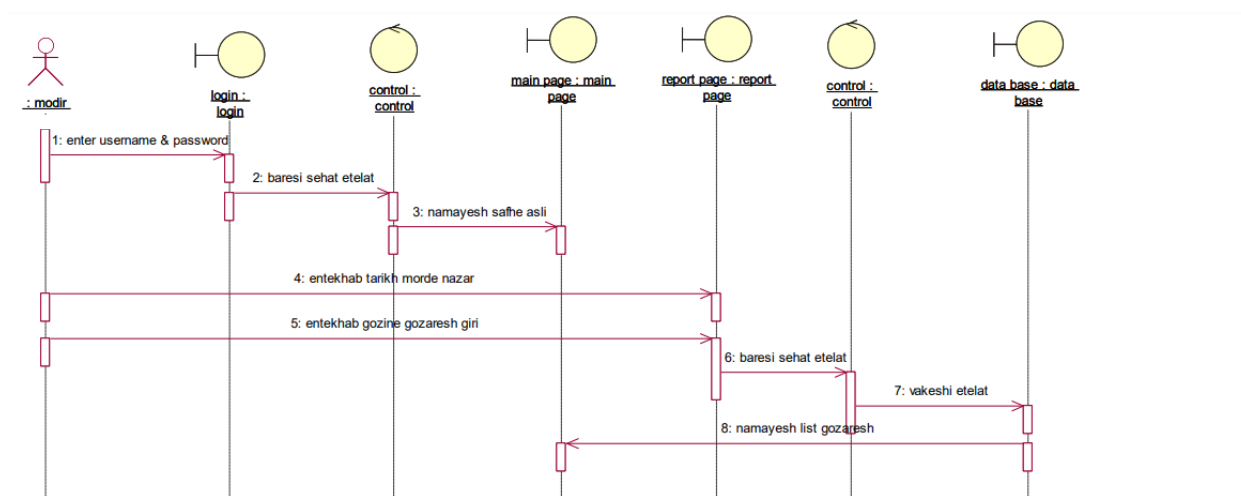
شکل ۲-۶ نمودار توالی ثبت نام



شکل ۲-۷ نمودار توالی نمایش مالکین



شکل ۲-۸ نمودار توالی جستجو



شکل ۹-۲ نمودار توالی گزارش گیری

۲-۵- سناریوها

سناریو اولین قدمی است که برای انجام تجزیه و تحلیل سیستم برداشته می شود. در سناریو ما شمای کلی سیستم و نحوه کار را توضیح می دهیم. این سناریوها در جدول (۲-۱) و (۲-۲) آورده شده است.

جدول (۲-۱) سناریو وارد کردن اطلاعات

سناریو شماره ۱	سیستم اتوماسیون مشاور املاک	نام: وارد کردن اطلاعات مالکین Actor: منشی سیستم
شرح سناریو: منشی سیستم قادر به ثبت مالکین میباشد.		
Assumption: منظور از منشی سیستم، افرادی هستند که با نرم افزار کار می کنند.		
Pre-Condition: اطلاعات موجود باشد		
Include		Extended
مراحل	Actor	System
	۱) منشی سیستم نام کاربری و رمز عبور را وارد میکند	۲) سیستم صحت اطلاعات وارد شده را شده را بررسی میکند
	۴) ورود اطلاعات مورد نظر	۳) نمایش صفحه اصلی
	۶) انتخاب گزینه ثبت	۵) بررسی صحت اطلاعات
		۷) ثبت اطلاعات در پایگاه داده
		۸) نمایش پیغام اطلاعات با موفقیت ثبت گردید
Constraints		
Post-Condition		

جدول (۱-۲) سناریو مدیریت کردن اطلاعات

نام: مدیریت کردن	سیستم اتوماسیون	سناریو شماره ۲
Actor: منشی سیستم	مشاور املاک	
شرح سناریو: منشی سیستم قادر به حذف و ویرایش اطلاعات می باشد.		
Assumption: منظور از منشی سیستم، افرادی هستند که با نرم افزار کار میکنند.		
Pre-Condition: اطلاعات موجود باشد		
Extended		Include
System	Actor	مراحل
(۲) سیستم صحت اطلاعات وارد شده را شده را بررسی میکند	(۱) منشی سیستم نام کاربری و رمز عبور را وارد میکند	
(۳) نمایش صفحه اصلی		
(۵) نمایش صفحه مورد نظر	(۴) انجام عمل حذف یا ویرایش	
(۶) ثبت اطلاعات در پایگاه داده	(۶) وارد کردن اطلاعات مورد نظر	
(۸) بررسی اطلاعات وارد شده	(۷) انتخاب گزینه ثبت	
(۹) ثبت اطلاعات در پایگاه داده		
(۱۰) نمایش پیغام اطلاعات با موفقیت ثبت شد		

جدول (۲-۳) سناریو گزارش گیری

نام: گزارش گیری Actor: منشی سیستم	سیستم اتوماسیون مشاور املاک	سناریو شماره ۱
شرح سناریو: منشی سیستم قادر به گزارش گیری میباشد.		
Assumption: منظور از منشی سیستم، افرادی هستند که با نرم افزار کار می کنند.		
Pre-Condition: اطلاعات موجود باشد		
Include		Extended
System	Actor	مراحل
(۲) سیستم صحت اطلاعات وارد شده را شده را بررسی میکند	(۱) منشی سیستم نام کاربری و رمز عبور را وارد میکند	
(۳) نمایش صفحه اصلی	(۴) انتخاب گزینه گزارش گیری	
(۵) نمایش صفحه گزارش گیری	(۶) ورود تاریخ	
(۷) انتخاب گزینه جستجو		
(۸) نمایش اطلاعات مورد نظر		
Constraints		
Post-Condition		

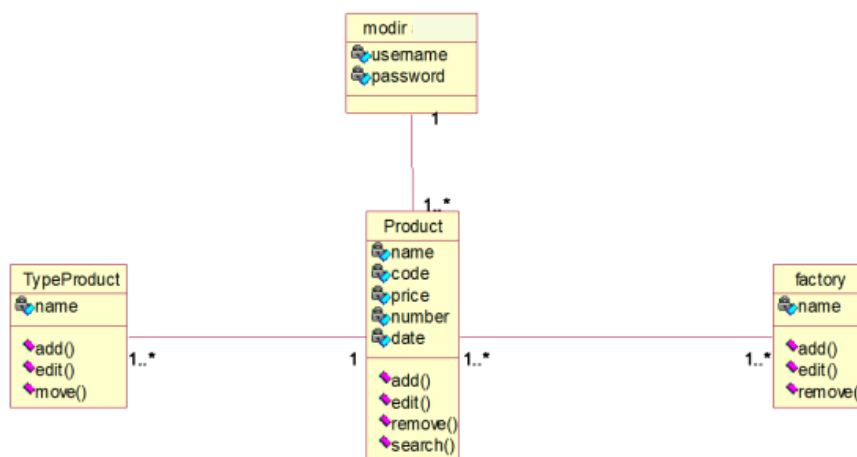
جدول (۴-۲) سناریو جستجو

نام: جستجو Actor: منشی سیستم	سیستم اتوماسیون مشاور املاک	سناریو شماره ۱
شرح سناریو: منشی سیستم قادر به جستجو اطلاعات میباشد.		
Assumption: منظور از منشی سیستم، افرادی هستند که با نرم افزار کار می کنند.		
Pre-Condition: اطلاعات موجود باشد		
Extended:		Include:
System:	Actor:	مراحل
(۲) سیستم صحت اطلاعات وارد شده را شده را بررسی میکند	(۱) منشی سیستم نام کاربری و رمز عبور را وارد میکند	
(۳) نمایش صفحه اصلی	(۴) انتخاب گزینه لیست مالکین	
(۵) نمایش صفحه لیست مالکین	(۶) وارد کردن کد ملی مالکین	
(۸) جستجو در پایگاه داده	(۷) زدن دکمه جستجو	
(۹) نمایش اطلاعات درخواستی		
Constraints:		
Post-Condition: -----		

۲-۶- نمودار کلاس (Class Diagram)

دیاگرام کلاس یکی از دیاگرامهای مهم و اساسی در متدلوژیهای شی گرا می باشد که هر متدلوژی حالات مختلفی از آن را استفاده می کند. دیاگرام کلاس شامل اشیاء و روابط ما بین آنها می باشد. همچنین دیاگرام کلاس شامل صفات و رفتار کلاسها می باشد. این نمودار، کلاسها، واسطه ها و همکاری و روابط بین آنها را نمایش میدهد و نمودار اصلی و مرکزی UML میباشد که بیان کننده ساختار ایستای سیستم نرم افزاری میباشد. در دیاگرام کلاس اگر فقط اشیاء (یعنی نمونه های کلاس ها) و روابط آنها نشان داده میشود، آنرا دیاگرام شی (Object Diagram) گویند.

واسطه ها (Interface) گونه ای از کلاسهای تغییر یافته هستند که به جای پیاده سازی کارکرد تنها آن را تعریف می نمایند. واسطه ها همه ویژگی های کلاسها را دارند و تنها تفاوتشان این است که در واسطه متدها فقط تعریف می شوند و پیاده سازی در کلاسها انجام می شود. در شکل (۲-۵) نمودار کلاس پروژه آمده است.



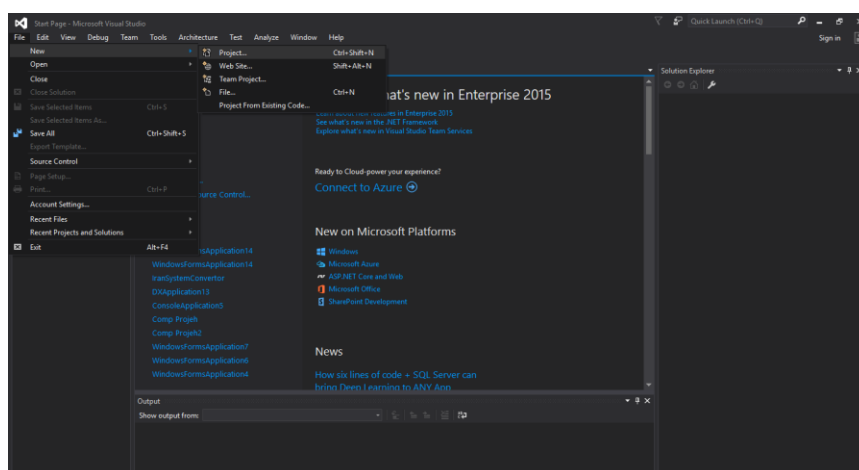
شکل (۲-۱۰) نمودار کلاس

فصل سوم

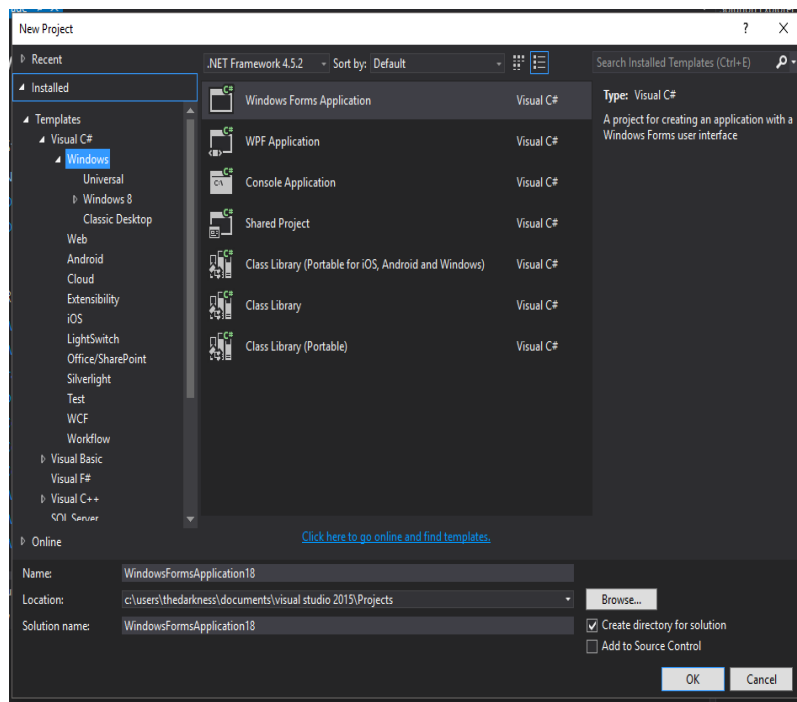
پیاده سازی سیستم

۳-۱- مقدمه

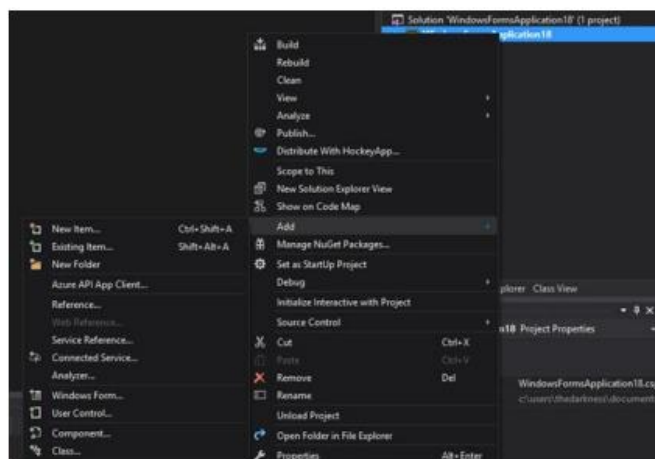
برای ایجاد یک پروژه جدید، ابتدا در محیط Visual Studio، از منوی File، گزینه New و سپس زیرمنوی Project را انتخاب می‌کنیم (شکل ۳-۱). از پنجره‌ایی که باز می‌شود گزینه Windows Form Application را انتخاب کرده (۳-۲) و پس از تعیین مسیر و نام پروژه، صفحه ایجاد می‌گردد.



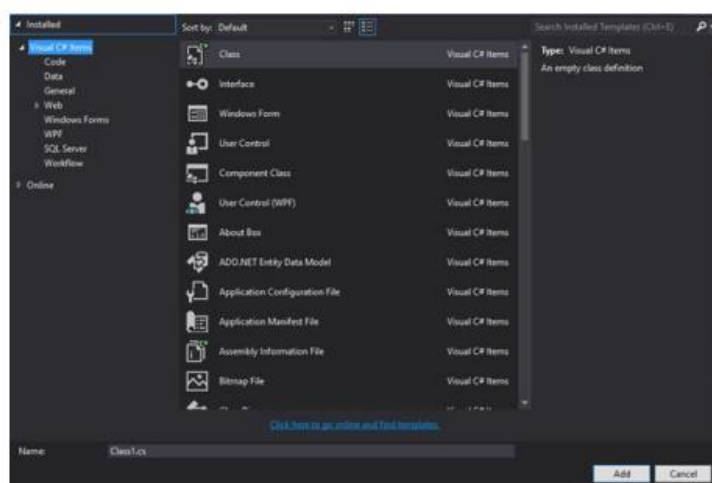
شکل (۳-۱) نحوه ایجاد یک پروژه جدید



شکل (۳-۲) انتخاب یک پروژه خالی



شکل (۳-۳) روش اضافه کردن یک آیتم جدید به پروژه

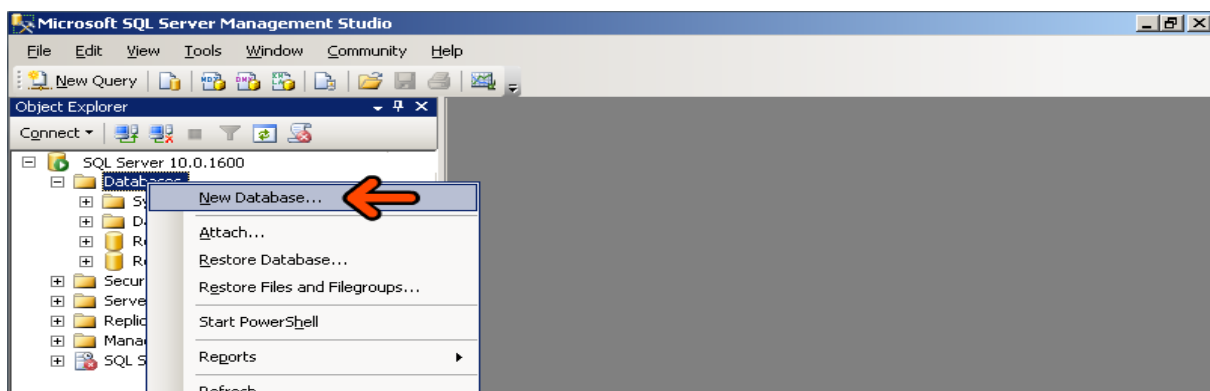


شکل (۳-۴) نحوه انتخاب صفحه با توجه به کاربرد مورد نیاز

۳-۲- پایگاه داده SQL در پروژه

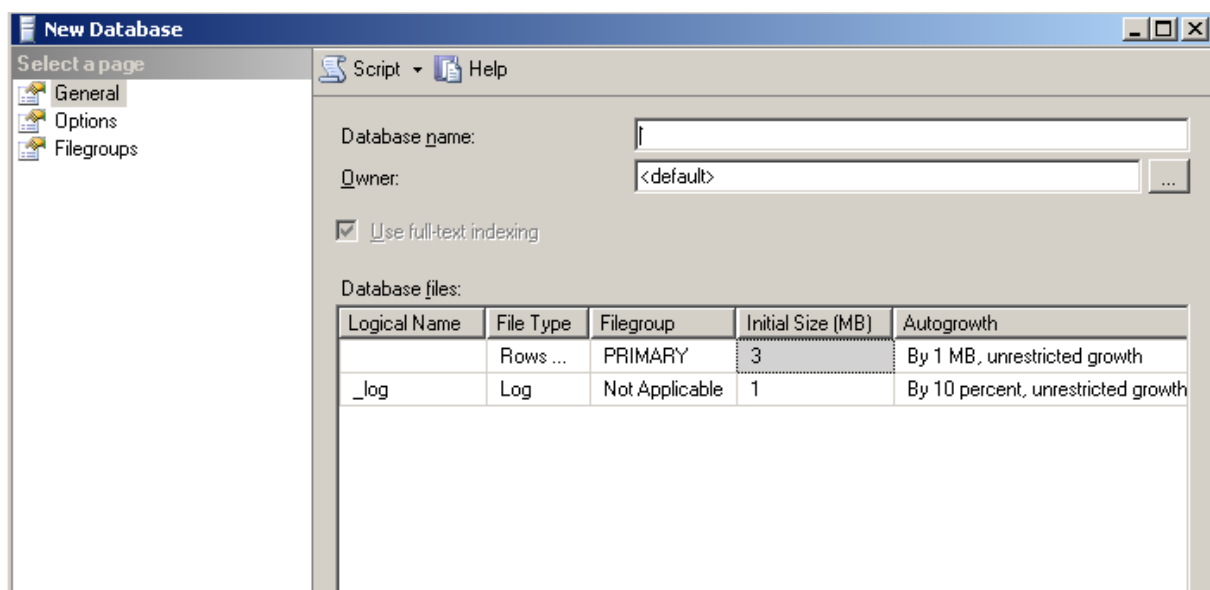
اولین قدم در ایجاد اتوماسیون ایجاد پایگاه داده است. در این بخش مرحله به مرحله نحوه ایجاد پایگاه داده پروژه بیان شده است.

پس از باز کردن برنامه SQL Server Management Studio و ایجاد اتصال و انتخاب سرور یا نمونه مورد نظر، برای ایجاد یک پایگاه داده جدید روی گزینه Databases در پنجره Object Explorer راست کلیک کرده و سپس گزینه New Databases را انتخاب می‌کنیم. (شکل ۳-۳)



شکل (۳-۳) انتخاب گزینه New Database

در پنجره ظاهر شده در قسمت Database Name نام پایگاه را وارد کرده و گزینه Ok را انتخاب می‌کنیم تا پایگاه داده ما ایجاد شود. در شکل ۳-۴ پنجره New Database مشاهده می‌شود.

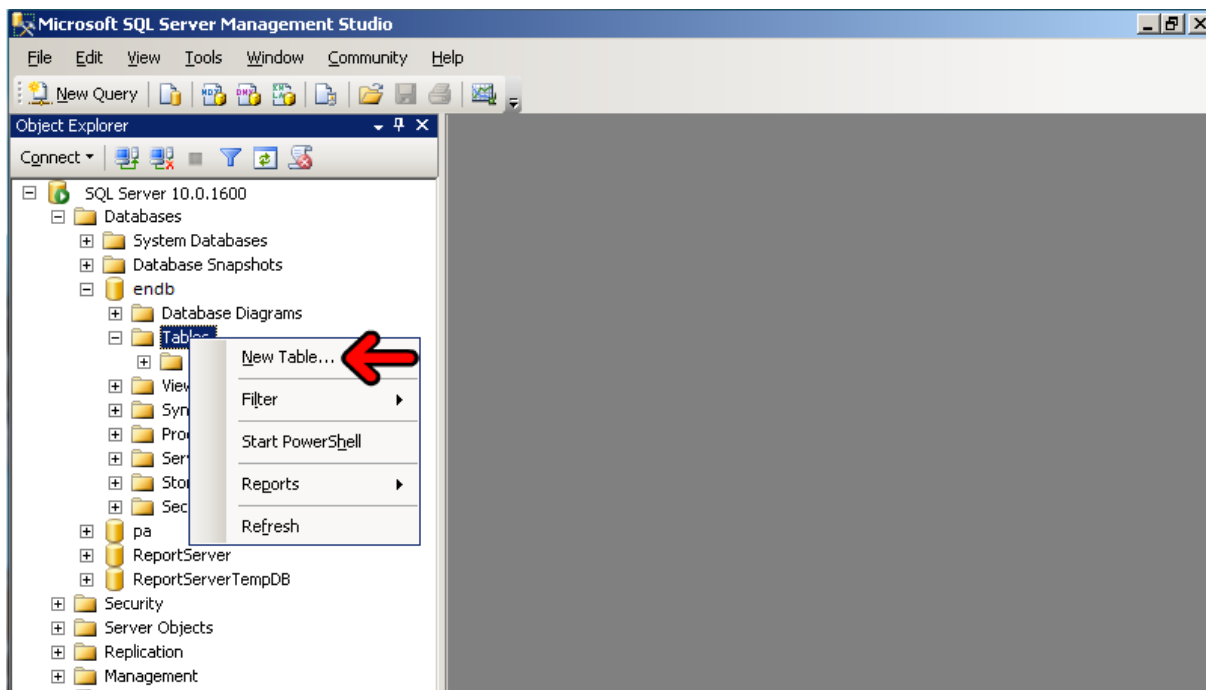


شکل (۴-۳) پنجره New Database

در این پروژه نام پایگاه داده را moshaveramlak^۲ در نظر گرفته‌ام.

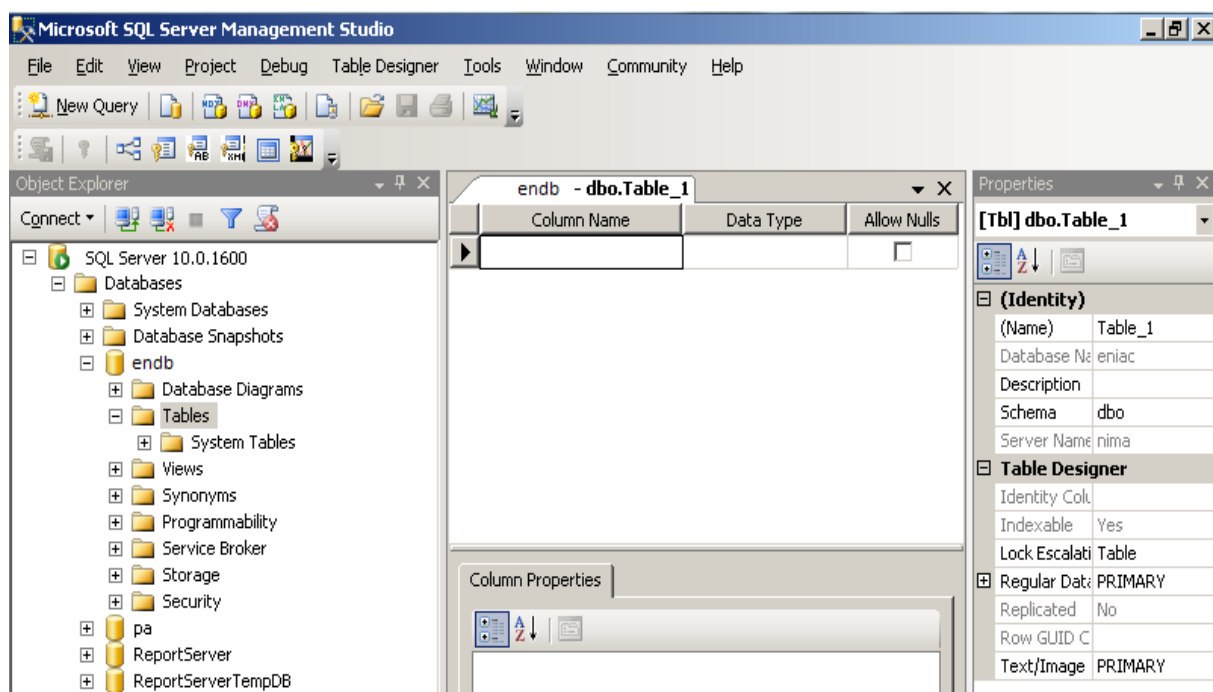
۳-۲-۱- جداول تعریف شده

حال می‌خواهیم جداول مورد نیاز وب‌سایت را ایجاد کنیم. برای ایجاد یک جدول در SQL Server از طریق Wizard در پنجره Object Explorer روی گزینه Tables واقع در پایگاه داده‌مان راست کلیک کرده و گزینه New Table... را انتخاب می‌کنیم. (شکل ۳-۵)



شکل (۳-۵) انتخاب گزینه New table

در پنجره باز شده فیلدهای هر جدول را تعریف می‌کنیم. برای تعریف هر فیلد باید نام فیلد را در قسمت Column Name و نوع داده را در قسمت Data Type وارد کنیم و در صورتیکه که فیلد مورد نظر نباید Null باشد، علامت تیک مربوط به خصوصیت Allow Nulls آن را برمی‌داریم. (شکل ۳-۶)



شکل (۳-۶) تعریف فیلدها

در نهایت دکمه Save را از نوار ابزار انتخاب کرده و نام جدول از ما پرسیده می شود که نام را وارد کرده و جدول ایجاد می گردد.

در ادامه جداول مربوط به این پروژه همراه با توضیح هر یک از فیلدهای آن آمده است:
نحوه ایجاد جدول مالکین در زیر آورده شده و اطلاعات مربوط به هر مالک در جدول (۳-۱) آورده شده است.

۳-۱-۲-۱ جدول malekin

Name: نام

Lsat name: نام خانوادگی

Num meli: کد ملی

Tellhome: تلفن منزل

Mobile: موبایل

Addhome: آدرس منزل

DESKTOP-PFRES50....mlak2 - Diagram_0*			DESKTOP-PFRES50....egareh aparteman	DESKTOP-PFRES50....- dbo.Ta
Column Name	Data Type	Allow Nulls		
id	int	<input type="checkbox"/>		
name	nvarchar(20)	<input type="checkbox"/>		
lastname	nvarchar(50)	<input type="checkbox"/>		
numbermeli	nvarchar(10)	<input type="checkbox"/>		
telhome	nvarchar(20)	<input type="checkbox"/>		
mobile	nvarchar(20)	<input type="checkbox"/>		
addhome	ntext	<input type="checkbox"/>		
		<input type="checkbox"/>		

جدول (۷-۳) مالکین

۳-۱-۲-۲ جدول khaneh

Address: آدرس

Metreaj: متر اژ

Room: اتاق خواب

Forosh: قیمت فروش

Pish: قیمت پیش

Ejareh: قیمت اجاره

Numbermelimalak: کد ملی مالک

Codekhaneh: کد خانه

DESKTOP-PFRES50.m....dbo.Table_khaneh			
Column Name	Data Type	Allow Nulls	
id	int	<input type="checkbox"/>	
address	ntext	<input type="checkbox"/>	
metraj	nvarchar(10)	<input type="checkbox"/>	
room	int	<input type="checkbox"/>	
fosh	nvarchar(20)	<input checked="" type="checkbox"/>	
pish	nvarchar(20)	<input checked="" type="checkbox"/>	
ejareh	nvarchar(20)	<input checked="" type="checkbox"/>	
numbermelimalak	nvarchar(10)	<input type="checkbox"/>	
codekhaneh	nvarchar(10)	<input type="checkbox"/>	
		<input type="checkbox"/>	

جدول (۸-۳) جدول خانه

gh kharid khaneh ۳-۲-۱-۳ جدول

Numermelimalek: کد ملی مالک

Numbermelikharidar: کد ملی خریدار

Codekhaneh: کد خانه

Codekgarardad: کد قرارداد

Date: تاریخ و زمان

DESKTOP-PFRESS0...._gh kharid khaneh		
Column Name	Data Type	Allow Nulls
id	int	<input type="checkbox"/>
numermelimalek	nvarchar(10)	<input type="checkbox"/>
numbermelikharidar	nvarchar(10)	<input type="checkbox"/>
codekhaneh	nvarchar(10)	<input type="checkbox"/>
codekgarardad	nvarchar(10)	<input type="checkbox"/>
date	nvarchar(15)	<input type="checkbox"/>
		<input type="checkbox"/>

Column Properties

جدول (۳-۹) جدول قرار داد خرید خانه

gh kharid zamin ۴-۲-۱-۳ جدول

Numermelimalek: کد ملی مالک

Numbermelikharidar: کد ملی خریدار

Codezamin: کد زمین

Codekharardad: کد قرارداد

DESKTOP-PFRES50....e_gh kharid zamin			
	Column Name	Data Type	Allow Nulls
▶	id	int	<input type="checkbox"/>
	numbermelimalek	nvarchar(10)	<input type="checkbox"/>
	numbermelikharidar	nvarchar(10)	<input type="checkbox"/>
	codezamin	nvarchar(10)	<input type="checkbox"/>
🔑	codekharardad	nvarchar(10)	<input type="checkbox"/>
			<input type="checkbox"/>

جدول (۳-۱۰) جدول قرار داد خرید زمین

gh egareh khaneh ۵-۲-۱-۳ جدول

Numermelimalek: کد ملی مالک

Codekhaneh: کد خانه

Numbermelimostajer: کد ملی مستاجر

Tozih: توضیحات

Data: تاریخ و زمان

DESKTOP-PFRES50...._gh egareh khaneh		
Column Name	Data Type	Allow Nulls
id	int	<input type="checkbox"/>
numbermelimalek	nvarchar(10)	<input type="checkbox"/>
codekhaneh	nvarchar(10)	<input type="checkbox"/>
codegharardad	nvarchar(10)	<input type="checkbox"/>
numbermostager	nvarchar(10)	<input type="checkbox"/>
tozih	ntext	<input checked="" type="checkbox"/>
date	nvarchar(15)	<input type="checkbox"/>
		<input type="checkbox"/>

جدول (۱۱-۳) جدول قرار داد اجاره خانه

۳-۱-۲-۶ جدول gh egareh aparteman

Numermelimalek: کد ملی مالک

Codeaparteman: کد آپارتمان

Numbermelimostajer: کد ملی مستاجر

Tozih: توضیحات

Data: تاریخ و زمان

DESKTOP-PFRES50....egareh aparteman		
Column Name	Data Type	Allow Nulls
id	int	<input type="checkbox"/>
numbermelimalek	nvarchar(10)	<input type="checkbox"/>
numbermelimostajer	nvarchar(10)	<input type="checkbox"/>
codegharardad	nvarchar(10)	<input type="checkbox"/>
codeaparteman	nvarchar(10)	<input type="checkbox"/>
tozih	ntext	<input checked="" type="checkbox"/>
date	nvarchar(15)	<input type="checkbox"/>
		<input type="checkbox"/>

جدول (۱۲-۳) جدول قرار داد اجاره آپارتمان

۳-۱-۲-۷ جدول aparteman

Address: آدرس

Metrej: متراژ

Room: اتاق خواب

Forosh: قیمت فروش

Pish: قیمت پیش

Ejareh: قیمت اجاره

Numbermelimalak: کد ملی مالک

Codeaparteman: کد آپارتمان

DESKTOP-PFRES50.....o.Table_aparteman			
	Column Name	Data Type	Allow Nulls
▶	id	int	<input type="checkbox"/>
	address	ntext	<input type="checkbox"/>
	metraj	nvarchar(10)	<input type="checkbox"/>
	frosh	nvarchar(20)	<input checked="" type="checkbox"/>
	pish	nvarchar(20)	<input checked="" type="checkbox"/>
	ejareh	nvarchar(20)	<input checked="" type="checkbox"/>
	room	int	<input type="checkbox"/>
	numbermelimalak	nvarchar(10)	<input type="checkbox"/>
🔑	codeaparteman	nvarchar(10)	<input type="checkbox"/>
			<input type="checkbox"/>

جدول (۳-۱۳) جدول آپارتمان

۳-۱-۲-۸ جدول mostager

Name: نام

Lsat name: نام خانوادگی

Num meli: کد ملی

Tellhome: تلفن منزل

Mobile: موبایل

Addhome: آدرس منزل

Column Name	Data Type	Allow Nulls
id	int	<input type="checkbox"/>
fistname	nvarchar(20)	<input type="checkbox"/>
lastname	nvarchar(50)	<input type="checkbox"/>
numbermeli	nvarchar(10)	<input type="checkbox"/>
mobile	nvarchar(20)	<input type="checkbox"/>
addhome	ntext	<input type="checkbox"/>
telhome	nvarchar(20)	<input type="checkbox"/>
		<input type="checkbox"/>

جدول (۱۴-۳) جدول مستاجر

zamin ۹-۲-۱-۳ جدول

Name: نام

Matrej: متراژ

Forosh: فروش

Numbermelimalek: کد ملی مالک

Codezamin: کد زمین

Column Name	Data Type	Allow Nulls
id	int	<input type="checkbox"/>
address	ntext	<input type="checkbox"/>
metraj	nvarchar(10)	<input type="checkbox"/>
frosh	nvarchar(40)	<input type="checkbox"/>
numbermelimalek	nvarchar(10)	<input type="checkbox"/>
codezamin	nvarchar(10)	<input type="checkbox"/>
		<input type="checkbox"/>

جدول (۱۵-۳) جدول زمین

login جدول ۱۰-۲-۱-۳

Username: نام کاربری

Password: پسورد

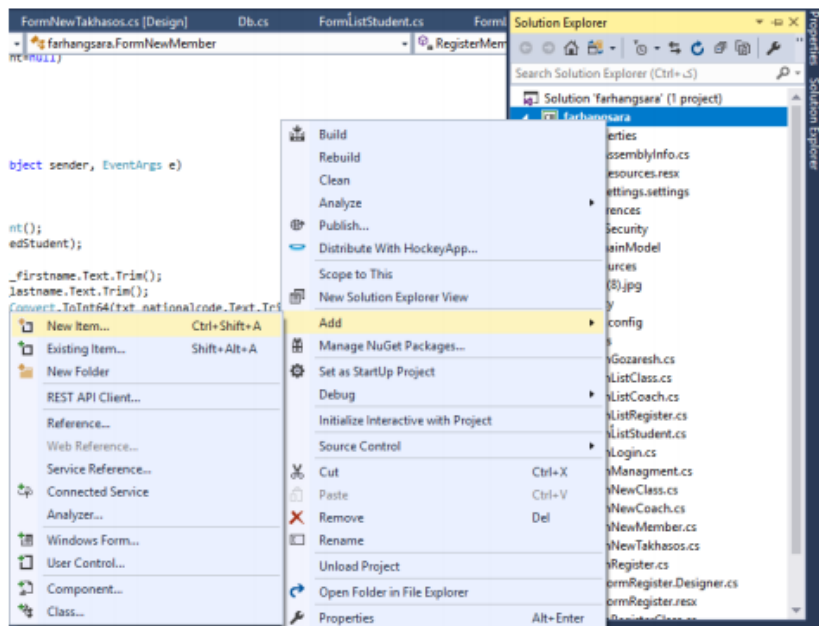
DESKTOP-PFRES50.m... - dbo.Table_login			Properties	
Column Name	Data Type	Allow Nulls	[Tbl] dbo.Table_login	
	int	<input type="checkbox"/>	(Identity)	
username	nvarchar(20)	<input type="checkbox"/>	(Name) Table_login	
password	nvarchar(20)	<input type="checkbox"/>	Database Name moshaveramlak2	
		<input type="checkbox"/>	Description	
		<input type="checkbox"/>	Schema dbo	
		<input type="checkbox"/>	Server Name desktop-pfres50	
		<input type="checkbox"/>	Table Designer	
		<input type="checkbox"/>	Identity Column id	
		<input type="checkbox"/>	Indexable Yes	

جدول (۱۶-۳) جدول ورود

۳-۲-۲- چگونگی ارتباط پروژه با پایگاه داده

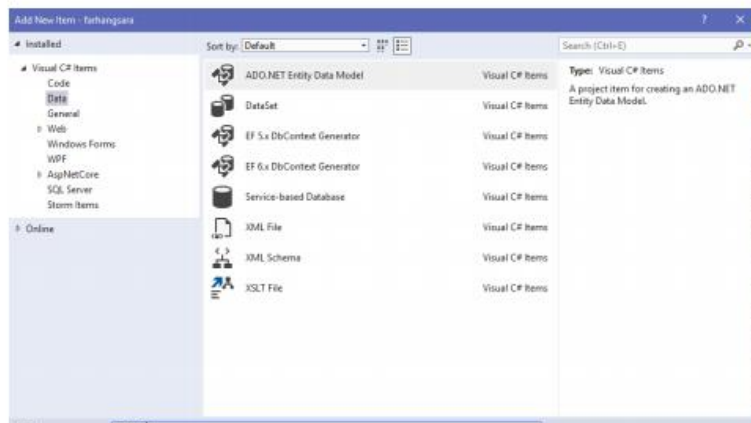
در این پروژه از مفهوم Entity Framework استفاده شده است که جهت ارتباط پایگاه داده و C# مراحل زیر طی میشود.

۱- ابتدا از نوار Solution Explorer بر روی نام پروژه راست کلیک کرده و گزینه Add و بعد از آن گزینه New Item را مانند تصویر زیر انتخاب میکنیم.



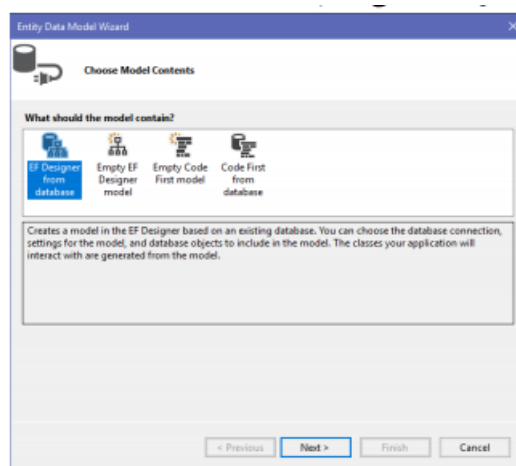
شکل (۳-۱۷) اضافه کردن item جدید

۲- در پنجره باز شده از قسمت چپ Data را انتخاب میکنیم و از گزینه های باز شده گزینه Add.Net Entity Data Model را انتخاب کرده و گزینه Add را میزنیم مانند تصویر زیر:



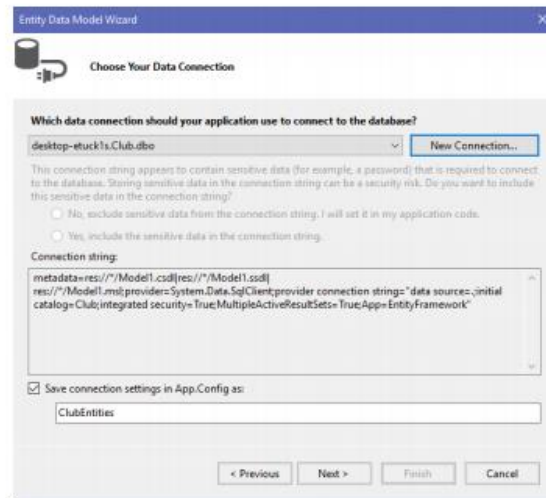
شکل (۳-۱۸) اضافه کردن ADO.NET Entity Data Model

۳- در این مرحله گزینه اول که مفهوم آن این است که می‌خواهیم از دیتابیس مدل شی‌گرای را انتخاب کنیم می‌زنیم و دکمه **Next** را انتخاب می‌کنیم.



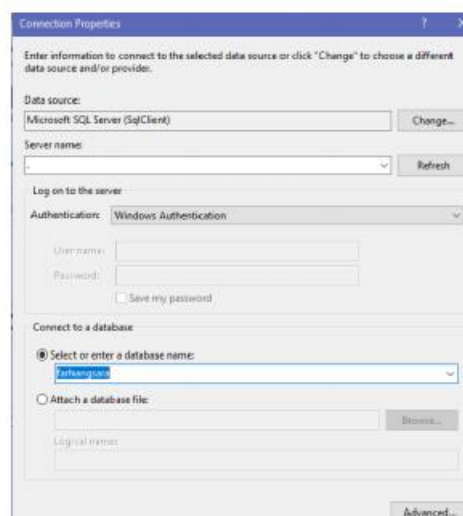
شکل (۳-۱۹) اضافه کردن مدل شی‌گرای از Data base

۴- در این مرحله باید **Connection** را انتخاب کنیم پس روی گزینه **new connection** می‌زنیم.



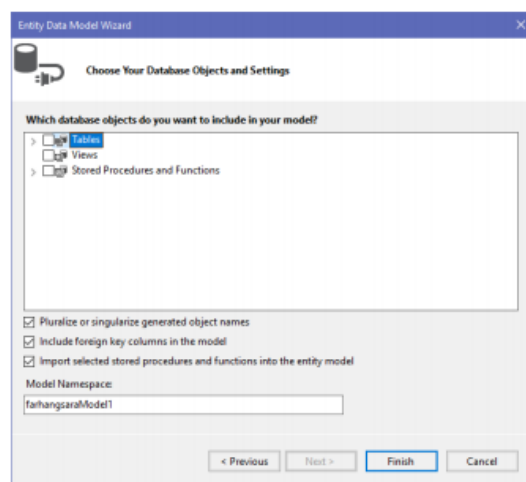
شکل (۳-۲۰) اضافه کردن new connection

۵- در قسمت جدید نام Data base خود را انتخاب کنیم به همین علت در Server Name نقطه وارد میکنیم تا نام پایگاه داده های موجود در سیستم ما در قسمت بعد بیاید و بعد از آن بین آن ها پایگاه داده مورد نظر خود را انتخاب میکنیم.



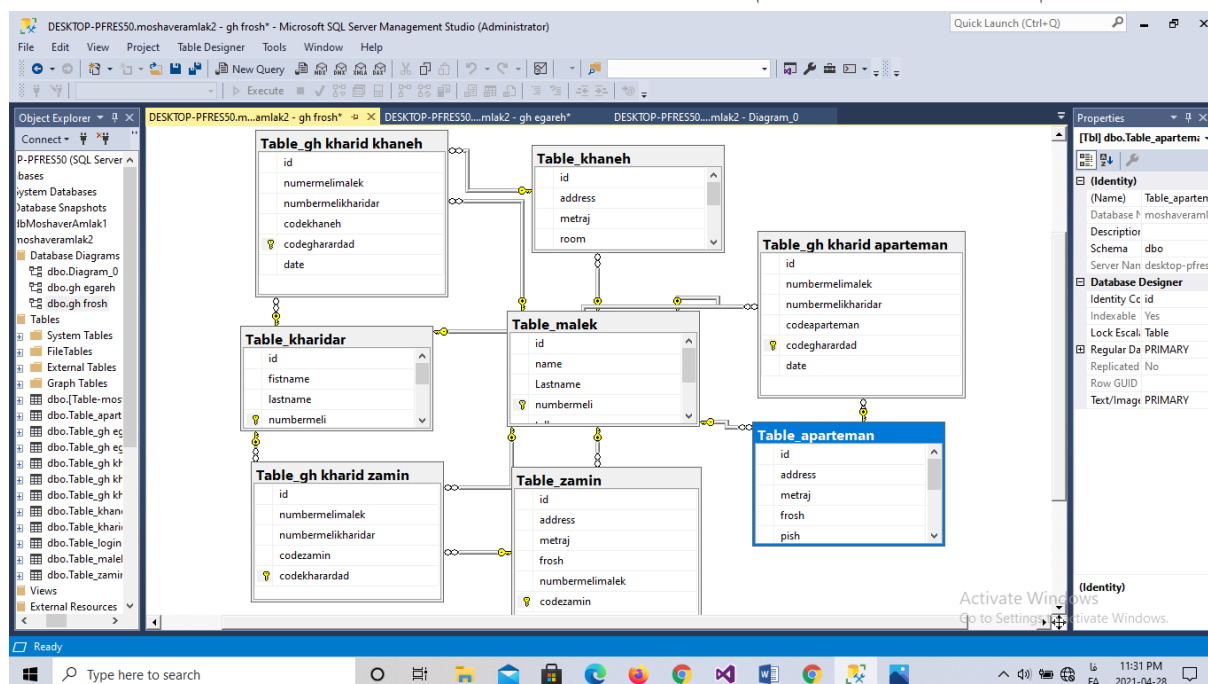
شکل (۳-۲۱) انتخاب نام Data base

۶- در این مرحله باید نام جداول که در پایگاه داده است را انتخاب کنیم و بعد از آن روی گزینه finish می زنیم.



شکل (۳-۲۲) اضافه کردن جداول به پایگاه داده

۷- بعد از این نام جداول را انتخاب کردیم تمامی جداول به ویژوال ما اضافه می شود. به طور مثال :

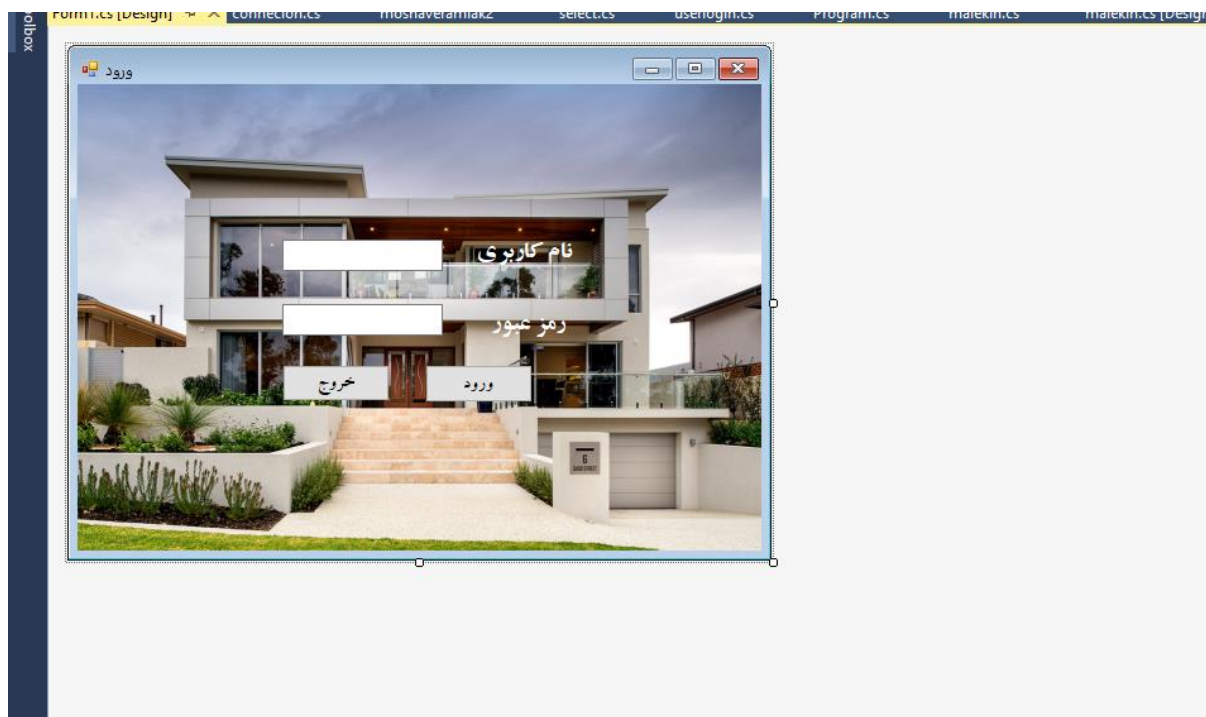


شکل (۳-۲۳) مدل جدول

۳-۳- نحوه پیاده سازی پروژه

با ایجاد پروژه، صفحه اصلی نمایش داده می شود و در واقع، اولین صفحه ای است که ما با آن مواجه می شویم. این صفحه را با windows form طراحی کرده سپس کدها را به آن اضافه می نماییم.

شکل (۳-۲۴) صفحه ورود



شکل (۳-۲۴) صفحه ورود

در این کد زمانی که شخصی وارد سیستم می شود تشخیص می دهیم که مدیر سیستم است یا کاربر پذیرش، بعد از تشخیص قوانین مربوط به آن را از پایگاه داده واکشی می کنیم.

```
public static void checkauthenticate(string Username,string Password)
{
    useronline = Db.Context.logins.Include("Roles").SingleOrDefault(p =>
p.username == Username && p.password==Password);
    if(useronline==null)
    {
        throw new AuthenticatedException();
    }
    logintime = DateTime.Now;
}
```

از این تابع در طول کد جهت دسترسی دادن به کاربر و مدیر سیستم استفاده کرده ایم که این دسترسی طبق قوانین تعریف شده در پایگاه داده است.

```
public static bool checkauthorizeforonlineUser(int RoleID)
{
    return useronline.Roles.Any(p => p.id == RoleID);
}
```

کدهای C# صفحه ورود

کدهای این قسمت مربوط به این میشود که آیا نام کاربری یا رمز عبور درست است یا نه. در صورت درست بودن منشی سیستم وارد برنامه میشود.

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Data.SqlClient;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Windows.Forms;

namespace moshaveramlak2
{
    public partial class login : Form
    {
        public login()
        {
            InitializeComponent();
        }

        private void login_Load(object sender, EventArgs e)
        {
        }

        private void btnvorod_Click(object sender, EventArgs e)
        {
            SqlConnection conn = new SqlConnection(connecion.connect);
            string sql = "select * from Table_login where username ='" + textBox1.Text
+ "' and password = '" + textBox2.Text + "'";
            conn.Open();
            SqlCommand cmd = new SqlCommand(sql, conn);
```



```

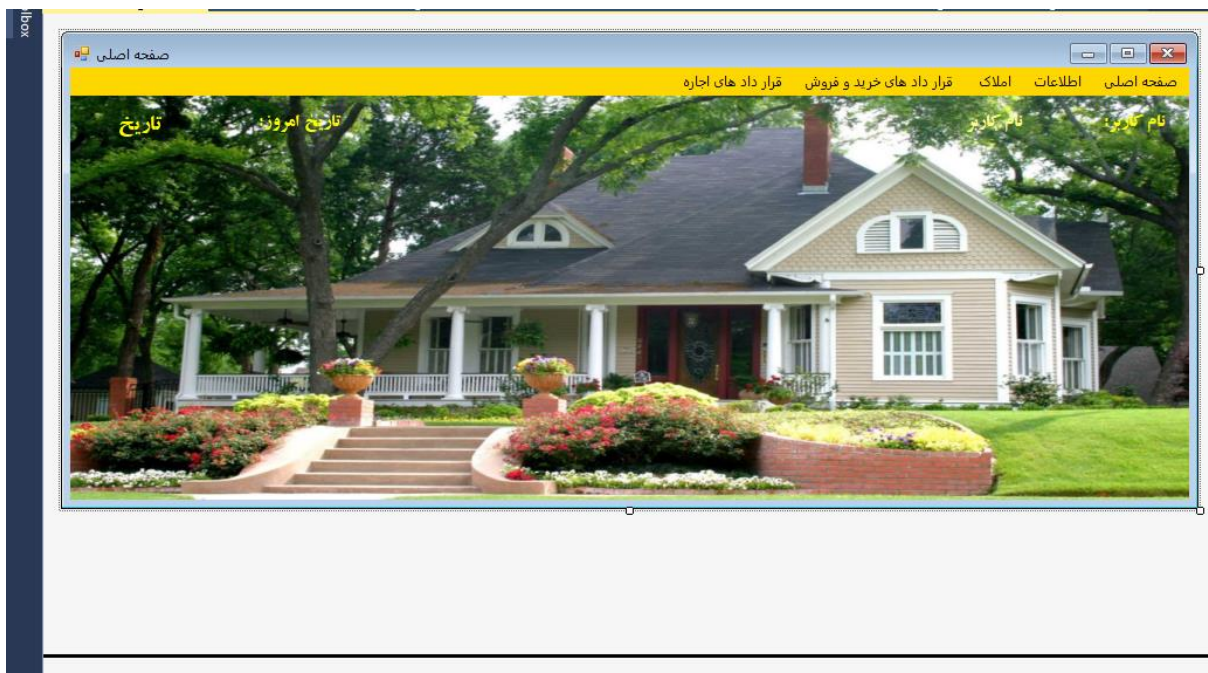
SqlDataReader re = cmd.ExecuteReader();
if(re.Read())
{
    userlogin.login = textBox1.Text;
    Form2 cb = new Form2();
    cb.Show();
    this.Hide();

}
else
{
    MessageBox.Show("است اشتباه عبور رمز یا کاربری نام");
    textBox1.Text = "";
    textBox2.Text = "";
}
conn.Close();
}

private void btnkhurog_Click(object sender, EventArgs e)
{
    Close();
}
}
}

```

کاربر با کلیک برروی گزینه ورود در منوی فایل از نرم افزار ورود پیدا میکند اگر رمز عبور و نام کار
بری درست وارد شده باشد، در غیر این صورت پیام رمز عبور یا نام کاربری اشتباه است.



شکل (۳-۲۵) صفحه اصلی نرم افزار

کدهای C# صفحه اصلی نرم افزار:

در این قسمت نیز کدهای مربوط به صفحه اصلی نرم افزار گذاشته شده است. در این صفحه نام کار بری و تاریخ ورود به نرم افزار نمایش داده می شود و همچنین می تواند از منو بالا به نقاط مختلف اپلیکیشن رود.

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Windows.Forms;
using System.Globalization;
namespace moshaveramlak2
{
    public partial class Form2 : Form
    {
        public Form2()
        {
            InitializeComponent();
        }
        public string datameladi(DateTime _date)
        {

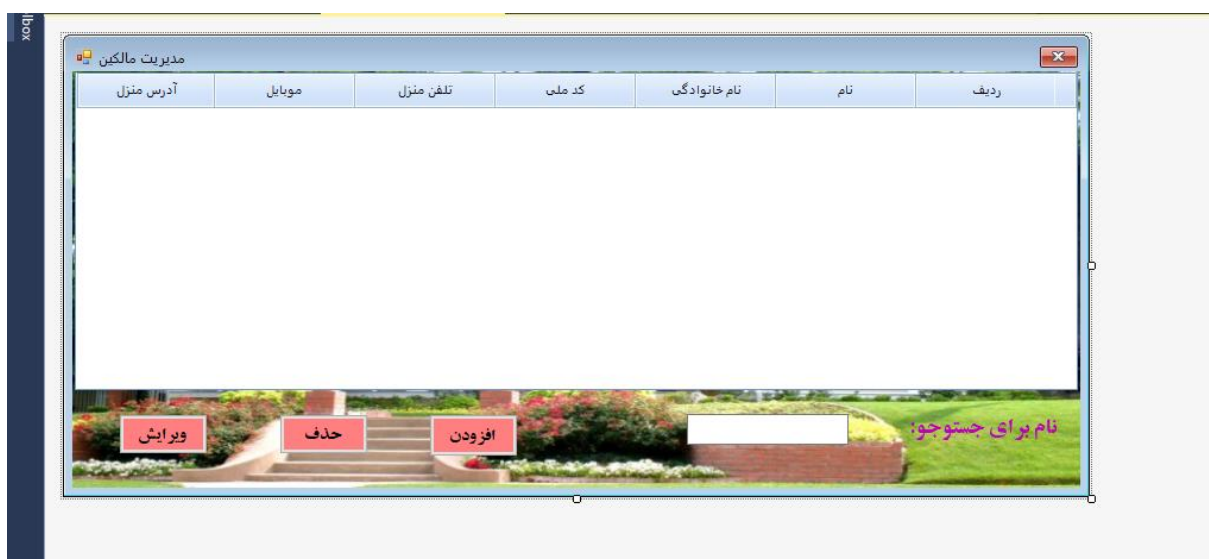
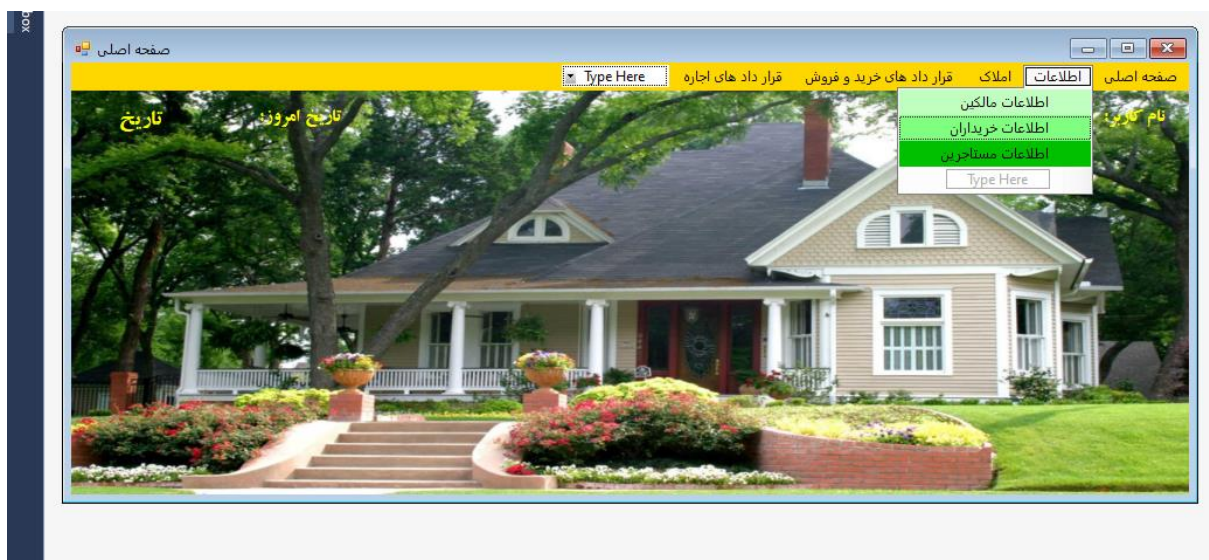
```

کد مربوط به تاریخ ورود کاربر به نرم افزار

```
PersianCalendar pc = new PersianCalendar();
StringBuilder br = new StringBuilder();
br.Append(pc.GetYear(_date).ToString("0000"));
br.Append("/");
br.Append(pc.GetMonth(_date).ToString("00"));
br.Append("/");
br.Append(pc.GetDayOfMonth(_date).ToString("00"));
return br.ToString();
}

private void Form2_Load(object sender, EventArgs e)
{
    IsMdiContainer = true;
    label1.Text = userlogin.login;
    string date = datameladi(DateTime.Now);
    lbltime.Text = date;
}

private void اطلاعاتمالکینToolStripMenuItem_Click(object sender, EventArgs e)
{
    malekin db = new malekin();
    db.MdiParent = this;
    db.Show();
}
}
```



شکل (۳-۲۶) صفحه مدیریت مالکین

کدهای C# صفحه مدیریت مالکین:

با انتخاب گزینه اطلاعات مالکین کاربر وارد صفحه مدیریت مالکین میشود . در اینجا کاربر میتواند تمام عملیات (حذف ، درج ، ویرایش و جستوجو) انجام دهد و در پایگاه داده ثبت شود. تمام عملیات حذف ، ویرایش و جستوجو با استفاده از 'radgridvieo' انجام میشود.

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Data.SqlClient;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Windows.Forms;

namespace moshaveramlak2
{
    public partial class malekin : Form
    {
        public malekin()
        {
            InitializeComponent();

            private void malekin_Load(object sender, EventArgs e)
            {
                // TODO: This line of code loads data into the
                'moshaveramlak2DataSet1.Table_malek' table. You can move, or remove it, as needed.

                this.table_malekTableAdapter.Fill(this.moshaveramlak2DataSet1.Table_malek);
                SqlConnection con = new SqlConnection(conecion.connect);
                con.Open();
                string sql = "select * from Table_malek";
                SqlDataAdapter da = new SqlDataAdapter(sql, con);
                DataTable dt = new DataTable();
                da.Fill(dt);
                con.Close();
                radGridView1.DataSource = dt;

            }
            private void showmalekin()
            {

            }

            private void button1_Click(object sender, EventArgs e)
            {
                addmalek df = new addmalek();
                df.ShowDialog();
                this.Hide();

            }
        }
    }
}
```

```

private void radGridView1_SelectionChanged(object sender, EventArgs e)
{
    label1.Text = radGridView1.SelectedRows[0].Cells[0].Value.ToString();
}

private void serg_TextChanged(object sender, EventArgs e)
{
    string sql = "select * from Table_malek where name like '%" +
txtsearch.Text + "%'";
    SqlDataAdapter da = new SqlDataAdapter(sql, connecion.conn);
    DataTable dt = new DataTable();
    connecion.conn.Open();
    da.Fill(dt);
    connecion.conn.Close();
    radGridView1.DataSource = dt;
}

private void label1_Click_1(object sender, EventArgs e)
{
}

private void hazfe_Click(object sender, EventArgs e)
{
    DialogResult yes;
    yes = MessageBox.Show("هستید؟ مطمئن آیا", "پیغام", MessageBoxButtons.YesNo);
    if (yes == DialogResult.Yes)
    {
        string sql = "delete from Table_malek where id=" + label1.Text;
        SqlCommand cmd = new SqlCommand(sql, connecion.conn);
        connecion.conn.Open();
        cmd.ExecuteNonQuery();
        connecion.conn.Close();
        MessageBox.Show("شد انجام موفقیت با حذف");
        SqlConnection con = new SqlConnection(connecion.connect);
        con.Open();
        string sql2 = "select * from Table_malek";
        SqlDataAdapter da = new SqlDataAdapter(sql2, con);
        DataTable dt = new DataTable();
        da.Fill(dt);
        con.Close();
        radGridView1.DataSource = dt;
    }
    else
    {
        MessageBox.Show("نشد انجام حذف");
    }
}

private void viraesh_Click(object sender, EventArgs e)
{
}

```

```

        select.id = Convert.ToInt32(label11.Text);
        edit_malek db = new edit_malek();
        db.Show();
    }
}

```

The screenshot shows a Windows application window with a title bar that says 'Ibox'. The main area of the window is a form with a light orange background. The form contains six text input fields arranged in two columns. The labels for these fields are in Persian: 'نام' (Name), 'نام خانوادگی' (Family Name), 'کد ملی' (National ID), 'تلفن منزل' (Home Phone), 'موبایل' (Mobile), and 'آدرس منزل' (Home Address). At the bottom of the form, there are two orange buttons with white text: 'ثبت' (Save) on the left and 'خروج' (Exit) on the right. The window has a standard Windows XP-style title bar with minimize, maximize, and close buttons.

شکل (۳-۲۷) صفحه افزودن مالک

کدهای C# صفحه افزودن مالکین:

در این صفحه منشی سیستم میتواند عملیات درج مالک را انجام دهد. با زدن دکمه ثبت اگر عملیات درست انجام شد عبارت ثبت با موفقیت انجام شد نمایش می دهد.

```

using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Data.SqlClient;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Windows.Forms;

namespace moshaveramlak2
{
    public partial class addmalek : Form
    {
        public addmalek()
        {
            InitializeComponent();

            private void label1_Click(object sender, EventArgs e)
            {

            }

            private void button5_Click(object sender, EventArgs e)
            {

            }

            private void sabt_Click(object sender, EventArgs e)
            {
                string sql = string.Format("INSERT INTO Table_malek
VALUES('{0}','{1}','{2}','{3}','{4}','{5}')" ,name1.Text,last.Text,meli1.Text,telhom.Te
xt,mobile1.Text,addhome1.Text);
                SqlCommand cmd = new SqlCommand(sql, connecion.conn);
                connecion.conn.Open();
                cmd.ExecuteNonQuery();
                connecion.conn.Close();
                MessageBox.Show("شد ذخيره اطلاعات");
                name1.Text = "";
                last.Text = "";
                meli1.Text = "";
                telhom.Text = "";
                mobile1.Text = "";
                addhome1.Text = "";

            }

            private void name1_TextChanged(object sender, EventArgs e)
            {

            }

            private void button2_Click(object sender, EventArgs e)
            {
                this.Hide();
            }
        }
    }
}

```



```
    }  
    private void addmalek_Load(object sender, EventArgs e)  
    {  
    }  
}  
}
```

شکل (۳-۲۸) صفحه مدیریت خریداران



کدهای C# صفحه مدیریت خریداران:

با انتخاب گزینه اطلاعات خریداران کاربر وارد صفحه مدیریت خریداران میشود . در اینجا کاربر میتواند تمام عملیات (حذف ، درج ، ویرایش و جستجو) انجام دهد و در پایگاه داده ثبت شود. تمام عملیات حذف ، ویرایش و جستجو با استفاده از `radgridvieo`^۱ انجام میشود.

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Data.SqlClient;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Windows.Forms;

namespace moshaveramlak2
{
    public partial class malekin : Form
    {
        public malekin()
        {
            InitializeComponent();
        }

        private void dataGridView1_CellContentClick(object sender,
        DataGridViewCellEventArgs e)
        {
        }
    }
}
```

```

private void malekin_Load(object sender, EventArgs e)
{
    // TODO: This line of code loads data into the
'moshaveramlak2DataSet1.Table_kharidar' table. You can move, or remove it, as needed.
    this.table_kharidarTableAdapter.Fill(this.moshaveramlak2DataSet1.Table_
kharidar');
    SqlConnection con = new SqlConnection(conecion.connect);
    con.Open();
    string sql = "select * from Table_kharidar ";
    SqlDataAdapter da = new SqlDataAdapter(sql, con);
    DataTable dt = new DataTable();
    da.Fill(dt);
    con.Close();
    radGridView1.DataSource = dt;
}
private void showmalekin()
{
}

private void button1_Click(object sender, EventArgs e)
{
    addkharidar df = new addkharidar ();
    df.ShowDialog();
    this.Hide();
}
private void radGridView1_SelectionChanged(object sender, EventArgs e)
{
    label1.Text = radGridView1.SelectedRows[0].Cells[0].Value.ToString();
}

private void serg_TextChanged(object sender, EventArgs e)
{
    string sql = "select * from Table_kharidar' where name like '%" +
txtsearch.Text + "%'";
    SqlDataAdapter da = new SqlDataAdapter(sql, conecion.conn);
    DataTable dt = new DataTable();
    conecion.conn.Open();
    da.Fill(dt);
    conecion.conn.Close();
    radGridView1.DataSource = dt;
}

private void label1_Click_1(object sender, EventArgs e)
{
}

private void hazfe_Click(object sender, EventArgs e)
{
    DialogResult yes;
    yes = MessageBox.Show("هستید مطمئن آیا", "پیغام", MessageBoxButtons.YesNo);
    if (yes == DialogResult.Yes)
    {

```

```

string sql = "delete from Table_ kharidar' where id=" + label1.Text;
SqlCommand cmd = new SqlCommand(sql, connecion.conn);
connecion.conn.Open();
cmd.ExecuteNonQuery();
connecion.conn.Close();
MessageBox.Show("شدد انجام موفقیت با حذف");

SqlConnection con = new SqlConnection(connecion.connect);
con.Open();
string sql2 = "select * from Table_ kharidar' ";
SqlDataAdapter da = new SqlDataAdapter(sql2, con);
DataTable dt = new DataTable();
da.Fill(dt);
con.Close();
radGridView1.DataSource = dt;

}
else
{
    MessageBox.Show("شدد انجام حذف");
}
}
private void label1_TabIndexChanged(object sender, EventArgs e)
{
}

private void viraesh_Click(object sender, EventArgs e)
{
    select.id = Convert.ToInt32(label1.Text);
    edit_ kharidar' db = new edit_ kharidar' ();
    db.Show();
}

}
}

```

شکل (۵-۲۹) صفحه افزودن خریدار

کدهای C# صفحه افزودن خریدار:

در این صفحه منشی سیستم میتواند عملیات درج خریدار را انجام دهد. با زدن دکمه ثبت اگر عملیات درست انجام شد عبارت ثبت با موفقیت انجام شد نمایش می دهد.

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Data.SqlClient;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Windows.Forms;

namespace moshaveramlak2
{
    public partial class addmalek : Form
    {
        public addkharidar()
        {
            InitializeComponent();
        }

        private void label1_Click(object sender, EventArgs e)
        {
        }
    }
}
```

```

private void button5_Click(object sender, EventArgs e)
{

}

private void sabt_Click(object sender, EventArgs e)
{
    string sql = string.Format("INSERT INTO Table_kharidar
VALUES('{0}','{1}','{2}','{3}','{4}','{5}')" ,name1.Text,last.Text,meli1.Text,telhom.Te
xt,mobile1.Text,addhome1.Text);
    SqlCommand cmd = new SqlCommand(sql, connecion.conn);
    connecion.conn.Open();
    cmd.ExecuteNonQuery();
    connecion.conn.Close();
    MessageBox.Show("شد ذخيره اطلاعات");
    name1.Text = "";
    last.Text = "";
    meli1.Text = "";
    telhom.Text = "";
    mobile1.Text = "";
    addhome1.Text = "";

}

private void name1_TextChanged(object sender, EventArgs e)
{

}

private void button2_Click(object sender, EventArgs e)
{
    this.Hide();
}

}

```



شکل (۵-۳۰) صفحه مدیریت مستاجر

کدهای C# صفحه مدیریت مستاجر:

با انتخاب گزینه اطلاعات مستاجر کاربر وارد صفحه مدیریت مستاجر میشود. در اینجا کاربر میتواند تمام عملیات (حذف، درج، ویرایش و جستجو) انجام دهد و در پایگاه داده ثبت شود. تمام عملیات حذف، ویرایش و جستجو با استفاده از `radgridvieo`^۱ انجام میشود.

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Data.SqlClient;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Windows.Forms;

namespace moshaveramlak2
{
    public partial class mostager : Form
    {
        public mostager ()
        {
            InitializeComponent();
        }

        private void dataGridView1_CellContentClick(object sender,
        DataGridViewCellEventArgs e)
        {
        }

        private void mostager_Load(object sender, EventArgs e)
        {
        }
    }
}
```

```

        // TODO: This line of code loads data into the
'moshaveramlak2DataSet1.Table_ mostager ' table. You can move, or remove it, as
needed.
        this.table_malekTableAdapter.Fill(this.moshaveramlak2DataSet1.Table_
mostager);
        SqlConnection con = new SqlConnection(conecion.connect);
        con.Open();
        string sql = "select * from Table_ mostager ";
        SqlDataAdapter da = new SqlDataAdapter(sql, con);
        DataTable dt = new DataTable();
        da.Fill(dt);
        con.Close();
        radGridView1.DataSource = dt;

    }
    private void showmostager()
    {

    }

    private void button1_Click(object sender, EventArgs e)
    {
        addmostager df = new addmostager();
        df.ShowDialog();
        this.Hide();
    }

    private void radGridView1_SelectionChanged(object sender, EventArgs e)
    {
        label1.Text = radGridView1.SelectedRows[0].Cells[0].Value.ToString();
    }

    private void dataGridView1_CellContentClick_1(object sender,
DataGridViewCellEventArgs e)
    {

    }
}
private void dataGridView1_SelectionChanged(object sender, EventArgs e)
{

}

private void serg_TextChanged(object sender, EventArgs e)
{
    string sql = "select * from Table_ mostager where name like '%" +
txtsearch.Text + "%'";
    SqlDataAdapter da = new SqlDataAdapter(sql, conecion.conn);
    DataTable dt = new DataTable();
    conecion.conn.Open();
    da.Fill(dt);
    conecion.conn.Close();
    radGridView1.DataSource = dt;

}

private void label1_Click_1(object sender, EventArgs e)
{

}

```



```

private void hazfe_Click(object sender, EventArgs e)
{
    DialogResult yes;
    yes = MessageBox.Show("هستید؟ مطمئن آیا", "پیغام", MessageBoxButtons.YesNo);
    if (yes == DialogResult.Yes)
    {

        string sql = "delete from Table_ mostager where id=" + label11.Text;
        SqlCommand cmd = new SqlCommand(sql, connection.conn);
        connection.conn.Open();
        cmd.ExecuteNonQuery();
        connection.conn.Close();
        MessageBox.Show("شد انجام موفقیت با حذف");

        SqlConnection con = new SqlConnection(connection.connect);
        con.Open();
        string sql2 = "select * from Table_ mostager ";
        SqlDataAdapter da = new SqlDataAdapter(sql2, con);
        DataTable dt = new DataTable();
        da.Fill(dt);
        con.Close();
        radGridView1.DataSource = dt;

    }
    else
    {
        MessageBox.Show("شد انجام حذف");
    }
}

private void label11_TabIndexChanged(object sender, EventArgs e)
{
}

private void viraesh_Click(object sender, EventArgs e)
{
    select.id = Convert.ToInt32(label11.Text);
    edit_malek db = new edit_malek();
    db.Show();
}

}

```



شکل (۵-۳۱) صفحه افزودن مستاجر

کدهای C# صفحه افزودن مستاجر:

در این صفحه منشی سیستم میتواند عملیات درج مستاجر را انجام دهد. با زدن دکمه ثبت اگر عملیات درست انجام شد عبارت ثبت با موفقیت انجام شد نمایش می دهد.

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Data.SqlClient;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Windows.Forms;

namespace moshaveramlak2
{
    public partial class addmostager : Form
    {
        public addmostager()
        {
            InitializeComponent();
        }

        private void label1_Click(object sender, EventArgs e)
        {
        }

        private void button5_Click(object sender, EventArgs e)
        {
        }
    }
}
```

```

{
}

private void sabt_Click(object sender, EventArgs e)
{
    string sql = string.Format("INSERT INTO Table_mostager
VALUES('{0}','{1}','{2}','{3}','{4}','{5}')" ,name1.Text,last.Text,meli1.Text,telhom.Te
xt,mobile1.Text,addhome1.Text);
    SqlCommand cmd = new SqlCommand(sql, connecion.conn);
    connecion.conn.Open();
    cmd.ExecuteNonQuery();
    connecion.conn.Close();
    MessageBox.Show("شدد ذخيره اطلاعات");
    name1.Text = "";
    last.Text = "";
    meli1.Text = "";
    telhom.Text = "";
    mobile1.Text = "";
    addhome1.Text = "";

}

private void name1_TextChanged(object sender, EventArgs e)
{
}

private void button2_Click(object sender, EventArgs e)
{
    this.Hide();
}
}
}

```

کدهای C# صفحه گزارش گیری:

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Threading.Tasks;
using System.Windows.Forms;

namespace Anbar
{
    public partial class FormGozaresh : Form
    {
        public FormGozaresh()
        {
            InitializeComponent();

            private void search_click(object sender, EventArgs e)
            {
                Grid_product.AutoGenerateColumns = false;

                var _list = Db.Context.Products.Where(p=> p.DateManufacturing>
txt_startDate.SelectedDateInDateTime && p.DateManufacturing<
txt_EndDate.SelectedDateInDateTime).Select(p => new { p.ID, p.ProductName, p.Code,

p.TypeProduct.TypeName, p.Factory.FactoryName, p.DateManufacturing, p.EndDate,
p.Number, p.Price }).ToList();
                Grid_product.DataSource = _list.ToList();
            }
        }
    }
}
```

در این کد زیر طبق تاریخ های وارد شده اطلاعات مربوط را از پایگاه داده واکشی می کند و در لیست مربوطه نمایش میدهد.

```
private void search_click(object sender, EventArgs e)
{
    Grid_product.AutoGenerateColumns = false;

    var _list = Db.Context.Products.Where(p=> p.DateManufacturing>
txt_startDate.SelectedDateInDateTime && p.DateManufacturing<
txt_EndDate.SelectedDateInDateTime).Select(p => new { p.ID, p.ProductName, p.Code,
p.TypeProduct.TypeName, p.Factory.FactoryName, p.DateManufacturing, p.EndDate,
p.Number, p.Price }).ToList();
    Grid_product.DataSource = _list.ToList();
}
```

کدهای C# مربوط به منشی سیستم و کاربر:

```
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;
using Anbar.DomainModel;
using Anbar.appsecurity;

namespace
{
    public partial class FormMangment : Form
    {
        public FormMangment()
        {
            InitializeComponent();
        }

        private void Sabt_click(object sender, EventArgs e)
        {
            if(!SecurityManagment.checkauthorizeforonlineUser(2))
            {
                MyMessageBox.ShowAuthorizeError();
                return;
            }
            FormRegisterProduct form = new FormRegisterProduct();
            form.ShowDialog();
        }

        private void NoeRegister_click(object sender, EventArgs e)
        {
            if (!SecurityManagment.checkauthorizeforonlineUser(4))
            {
                MyMessageBox.ShowAuthorizeError();
                return;
            }
            FormTypeProduct form = new FormTypeProduct();
            form.ShowDialog();
        }

        private void Exit_click(object sender, EventArgs e)
        {
            FormExit form = new FormExit();
            form.ShowDialog();
        }

        private void Factory_click(object sender, EventArgs e)
        {
            if (!SecurityManagment.checkauthorizeforonlineUser(3))
            {
                MyMessageBox.ShowAuthorizeError();
                return;
            }
            FormFactory form = new FormFactory();
            form.ShowDialog();
        }
    }
}
```

```

    }

    private void FormMangment_Shown(object sender, EventArgs e)
    {
        FormLogin form = new FormLogin();
        form.ShowDialog();
    }

    private void Gozaresh_Click(object sender, EventArgs e)
    {
        if (!SecurityManagment.checkauthorizeforonlineUser(5))
        {
            MyMessageBox.ShowAuthorizeError();
            return;
        }
        FormGozaresh form = new FormGozaresh();
        form.ShowDialog();
    }

    private void List_Click(object sender, EventArgs e)
    {
        FormListProduct form = new FormListProduct();
        form.ShowDialog();
    }
}
}

```

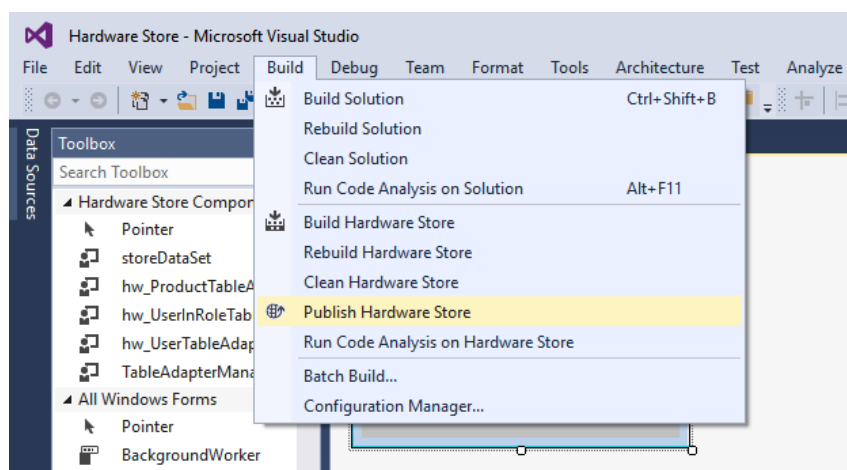
فصل چهارم

نصب و راهنمای پروژه

۶-۱- نحوه نصب و راه اندازی پروژه

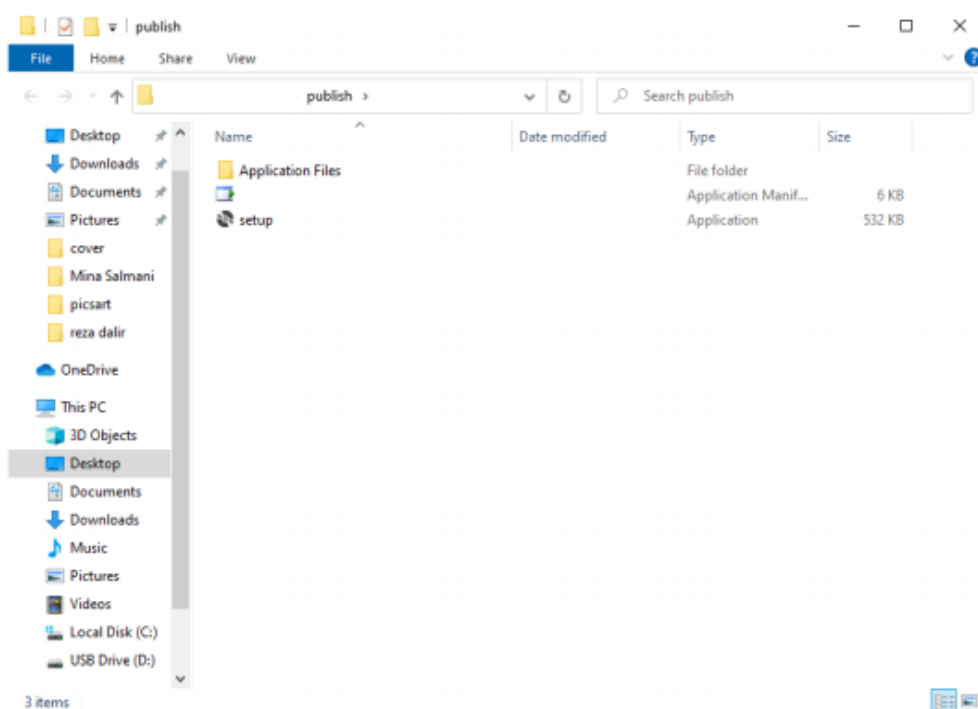
- اقدام به Publish پروژه درون ویژوال استادیو
- نصب SQL بروی سیستم
- نصب برنامه از طریق Setup ایجاد شده توسط ویژوال استادیو
- باز کردن برنامه از طریق منو استارت

برای نصب نرم افزار با استفاده از گزینه Publish در Visual Studio ابتدا اقدام به ایجاد فایل نصبی پروژه می نماییم، تا بتوانیم آن را بروی سی دی قرار داده و اقدام به نصب آن بروی سیستم ها نماییم:



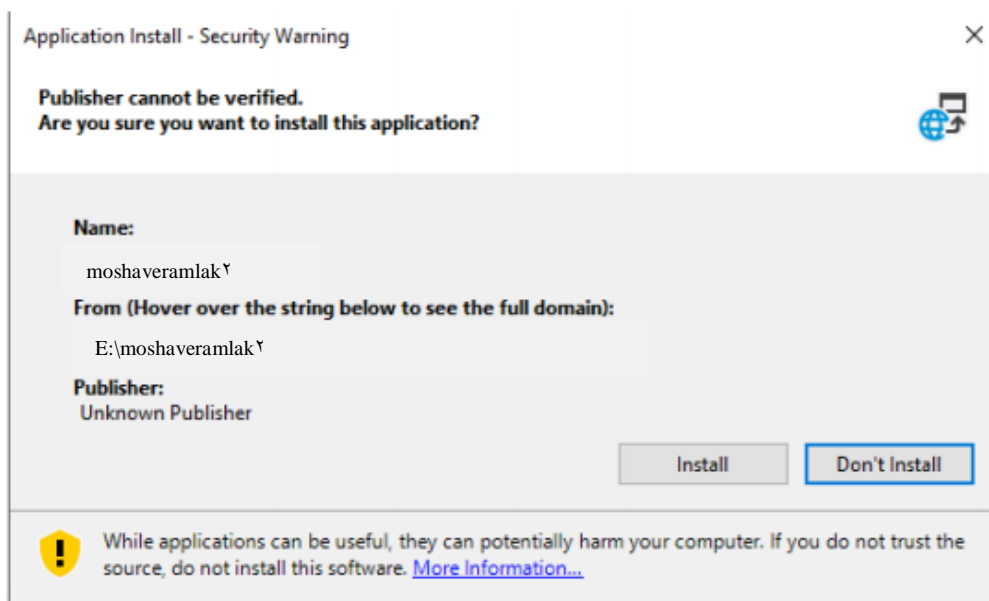
شکل (۶-۱) اقدام به Publish پروژه

سپس از Publish از فایل Setup ایجاد شده اقدام به نصب پروژه بر روی سیستم می نماییم. لازم به ذکر است برای نصب این پروژه به نسخه 4,5 NetFramework و SQL 2014 نیاز می باشد.



شکل (۶-۲) اجرا فایل Setup

با زدن بر روی گزینه Setup با پنجره زیر روبرو میشویم که با کلیک بر روی Install نرم افزار نصب می شود .



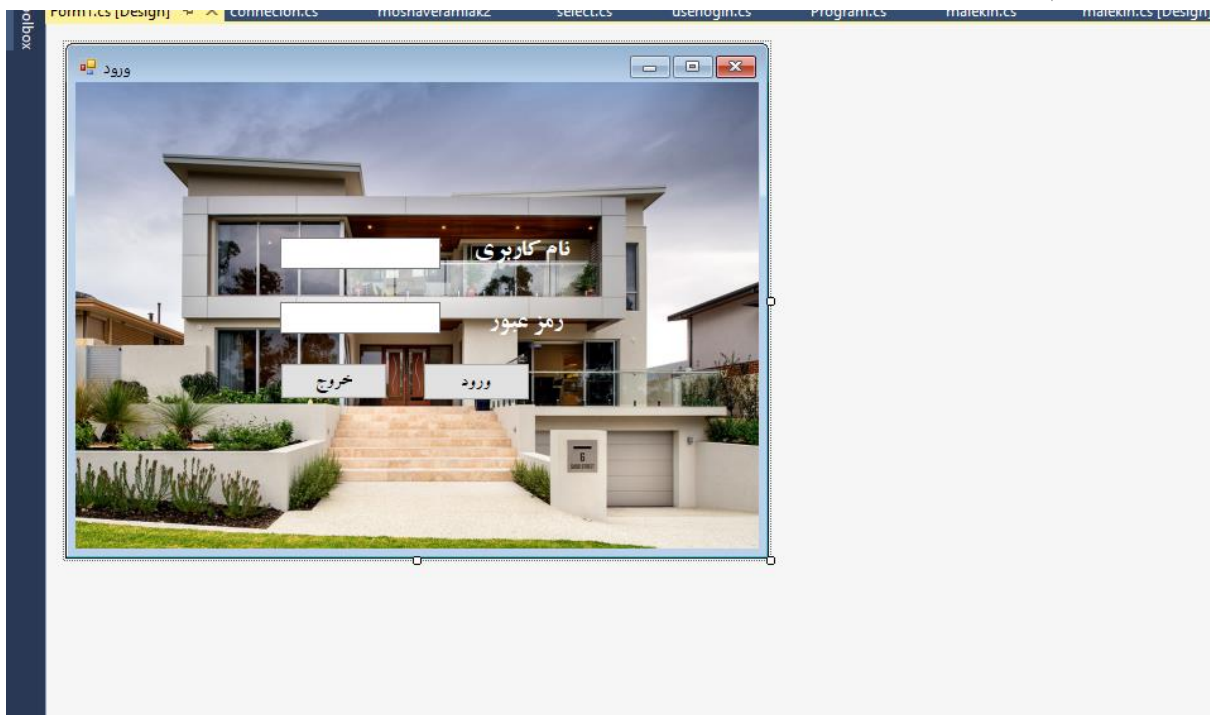
شکل (۳-۶) پنجره نصب نرم افزار

۲-۶- راهنمای نرم افزار

نرم افزار مشاور املاک ، قابلیت ثبت اطلاعات مالکین، خریداران و و ساعت ورود و را دارا می باشد، که در کنار آن، کاربر توانایی گزارش گیری و همچنین جستجو درون پایگاه داده ها را دارد.

۱-۲-۶ صفحه ورود به برنامه

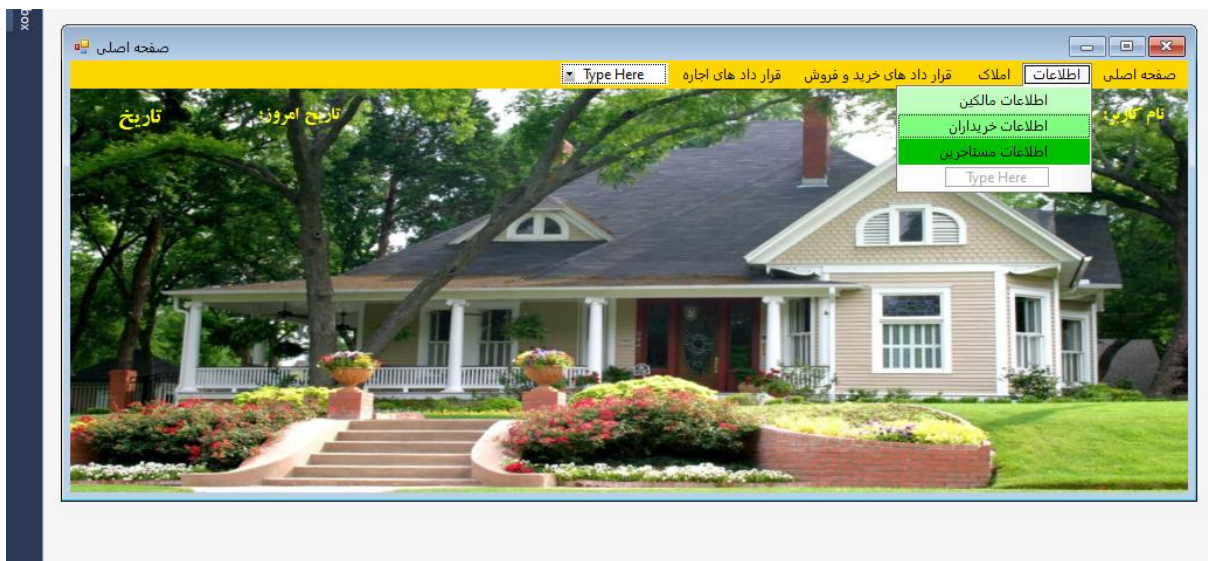
با ایجاد پروژه، صفحه ورود، شکل (۲-۴) نمایش داده می شود و در واقع، اولین صفحه ای است که ما با آن مواجه می شویم.



شکل (۲-۴) صفحه ورود نرم افزار

۲-۲-۶ منوهای اصلی

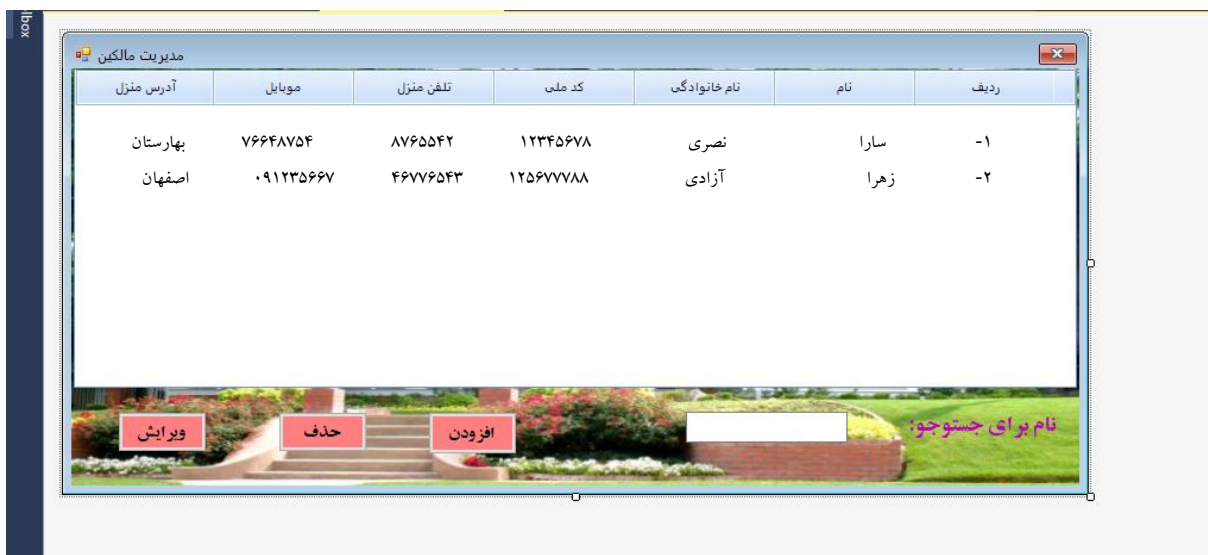
این نرم افزار شامل منوهای صفحه اصلی ، اطلاعات، قرارداد ها و گزارش گیری می باشد. کاربر، با زدن بر روی گزینه اطلاعات، با زیرمنوهای اطلاعات مالکین ،اطلاعات خریداران ، اطلاعات مستاجرین روبرو می شود. در شکل (۳-۴) نشان داده شده است.



شکل (۳-۶) صفحه اصلی

۳-۲-۶- پنجره های فرعی نرم افزار

کاربر، با زدن بر روی گزینه اطلاعات و سپس کلیک بر روی گزینه اطلاعات مالکین، وارد صفحه مالکین می شود.



شکل (۴-۶) اطلاعات مالکین

کاربر با زدن بر روی گزینه افزودن شکل (۴-۵) نشان داده می شود.

The screenshot shows a web form titled 'افزودن مالکین' (Add Owners). The form is displayed in a browser window. It contains the following fields and labels:

- نام (Name): Input field
- نام خانوادگی (Family Name): Input field
- کد ملی (National ID): Input field
- تلفن منزل (Home Phone): Input field
- موبایل (Mobile Phone): Input field
- آدرس منزل (Home Address): Input field

At the bottom of the form, there are two buttons: 'ثبت' (Save) and 'خروج' (Exit).

شکل (۵-۶) افزودن مالکین

مدیر با استفاده از صفحه ویرایش و حذف مالکین توانایی تغییر در اطلاعات وارد شده و بروزرسانی یا رفع اشکال را دارا می‌باشد، که برای دسترسی به آن کافیست از منوی اطلاعات، اطلاعات مالکین، قسمت ویرایش و حذف را باز نماید.

۳-۶- نتیجه گیری

مشاور املاک، این امکان را به کاربران می‌دهد تا اطلاعات مربوط مالکین و ... ساعت ورود و قیمت املاک را مدیریت نمایند و همچنین، کاربران، می‌توانند به راحتی از سیستم، گزارش‌گیری نمایند و اطلاعات مفیدی به دست آورند.

۴-۶- منابع و مراجع

- ۱- جونز، برادلی. گوهری، پریسا. آموزش C#. تهران: نص، ۱۳۹۱، چاپ ششم، ۱۳۹۱
- ۲- ساهاکیان، تالین. اصول و طراحی پایگاه داده، اصفهان: جهاد دانشگاهی واحد اصفهان، چاپ هشتم، ۱۳۸۹.
- ۳- فیلم های آموزشی you tube
- ۴- فیلم های کمک آموزشی اساتید دانشگاه خوراسگان