

Form1

سامانه مدیریت بیمارستان



شماره شناسایی:

رمز عبور:

نوع دسترسی:

[حساب کاربری ندارید؟ ایجاد حساب کاربری](#)

گزارش کار پروژه پایانی

درس آزمایشگاه پایگاه داده ها

سامانه مدیریت بیمارستان

علی فاضلی

دانشگاه تبریز، بهار ۹۶

مقدمه:

یک سیستم یکپارچه مدیریت بیمارستان (Hospital Management System) و یا سیستم اطلاعات بهداشتی عبارت است از یک پایگاه داده ها و نرم افزار جامع برای یکپارچه سازی اطلاعات مربوط به بیمار جهت ارسال و تبادلات اطلاعات جامع بیمار بین بخشها و سایر مراکز درمانی بمنظور تسريع در فرايند مراقبت و درمان بیمار، بهبود کیفیت، افزایش رضایتمندی، و کاهش هزینه.

سیستم اطلاعات بیمارستانی، یک سیستم مکانیزه مدیریت اطلاعات و اسناد در بیمارستانها می باشد. سیستم اطلاعات بیمارستانی، یک ابزار قدرتمند اطلاعاتی است که می تواند مدیران بیمارستانها را در فرايند اداره بیمارستانها و اتخاذ تصمیم های صحیح یاری نماید و عملکرد مثبت بیمارستانها را بنحو چشمگیری افزایش دهد. از این رو سیستم اطلاعات بیمارستانی نه تنها موجب بهبود عملکرد و تصمیم گیری بهتر می شود، بلکه با ارایه اطلاعات و سوابق بیمار به ارایه کنندگان خدمات در تصمیم گیری های بهداشتی - درمانی نیز نقش به سزا و موثری دارد.

سیستم اطلاعات بیمارستانی، برای خودکار نمودن امور بیمارستان ها مانند؛ گزارش دهی نتایج آزمایشات، وارد نمودن دستورات پزشک، تجویز دارو، کنترل موجودی داروخانه، انبار مرکزی، واحد تغذیه، و غیره طراحی شده است. در سیستم اطلاعات بیمارستانی، برای هر بیمار یک پرونده الکترونیک تشکیل می گردد، بطوریکه کلیه فعالیتهای بیمارستانی (شامل؛ درمانی، تشخیصی، مالی، و غیره) بیمار از پذیرش تا ترخیص تحت پوشش قرار می دهد. در این سیستم، کلیه اقدامات درمانی، دستورات دارویی و خدمات تشخیصی از طریق سیستم به کلینیکها و پاراکلینیکها و حتی مراکز اداری از قبیل حسابداری، داروخانه، انبارها، و سایر واحدها ارسال شده و پاسخ آنها دریافت می شود. بنابراین، زمان شروع و خاتمه همه اقدامات در سیستم مشخص و قابل پیگیری است. بنابراین سیستم، سیستم اطلاعاتی است که در آن اطلاعات HIS اطلاعات بیمارستانی در بانک اطلاعاتی به نحو جامعی ذخیره شده و از آنجا در زمان و مکان مورد نیاز به اطلاعات در فرمهای ویژه در دسترس مصرف کنندگان قرار می گیرد.

با توصیفات فوق اهمیت موضوع پروژه مشخص شده و علت انتخاب موضوع پروژه نیز قابل توجه می باشد.

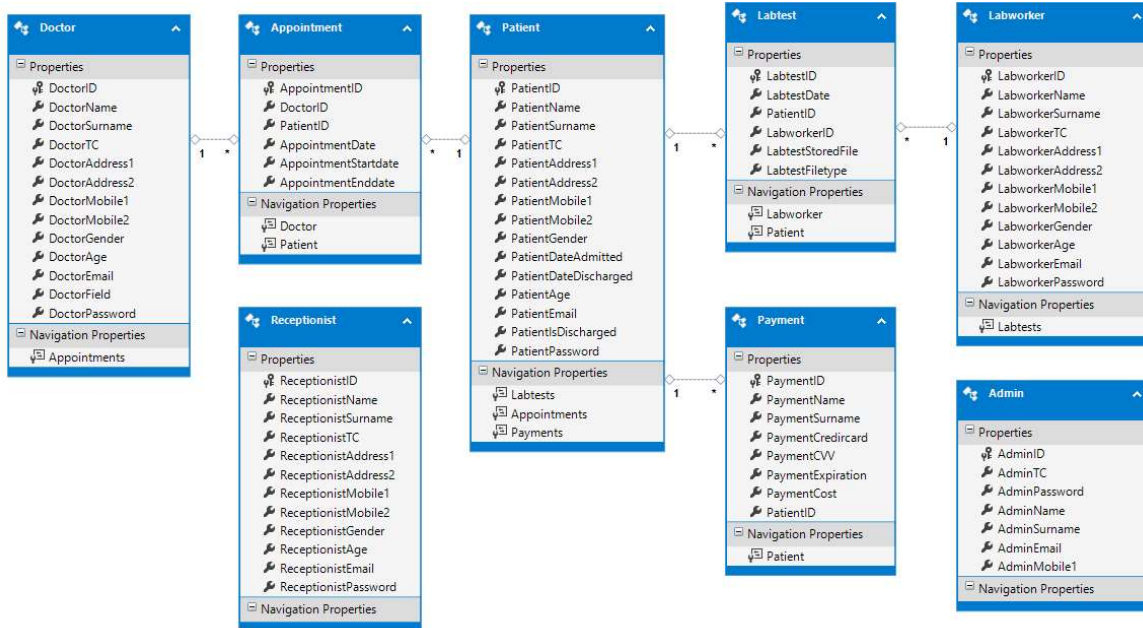
از مزایای یک سیستم اطلاعات بیمارستانی می توان به موارد زیر اشاره کرد:

- ایجاد مدیریت علمی و صحیح در اداره بیمارستان
- ایجاد نظم منطقی در واحدهای مختلف
- افزایش سرعت و دقت در ارائه خدمات شامل: پذیرش، بستری، ترخیص، خدمات کلینیکی و پاراکلینیکی، خدمات اداری و مالی
- افزایش دقت در تهیه، ثبت، نگهداری و ارسال به موقع گزارشهای درمانی و یا نتایج آزمایش ها
- سامان دادن به جمع آوری، تفکیک و استخراج اطلاعات بیمارستانی
- افزایش امکان کنترل مدیریت بر فعالیت های روزمره بیمارستانی
- امکان دسترسی به آخرین اطلاعات بخش های مختلف و نظارت مستقیم بر امور بیمارستان
- امکان تهیه گزارشهای مختلف مدیریت

و در نهایت تسريع فرايند درمان و بازیابی سلامت بیمار که هدف اصلی هر برنامه مدیریت بیمارستان می باشد.

هدف کلی پروژه، نوع و نحوه پیاده سازی:

در ابتدا، نمودار ER پروژه، با توجه به جزئیات لازم برای هر موجودیت موجود در سناریو برنامه، ویژگی ها و ... طراحی شد. نمودار ER پروژه مدیریت الکترونیکی بیمارستان ما به شرح زیر است:



تصویر ۱ نمودار ER پروژه

همانطور که در تصویر مشخص است پروژه دارای هشت جدول جداگانه می باشد که از فسمت طراحی مدل نرم افزار Visual Studio برداشته شده است.

لیست جداول مورد استفاده به شرح زیر است:

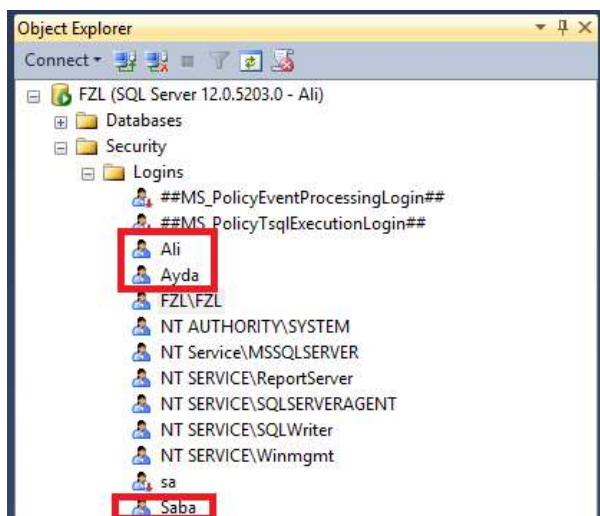
Patient	Receptionist	Labtest	Labworker	Payment	Appointment	doctor	Admin

- جدول doctor دارای کلید اصلی doctor id است که شامل مشخصات دکترهای بیمارستان است.
 - جدول appointment دارای کلید اصلی appointment id است و کلیدهای خارجی این جدول doctor id patient id هستند که این جدول حاوی اطلاعات زمان ملاقات بیمار توسط دکتر است.
 - جدول payment دارای کلید اصلی payment id و کلید خارجی patient id از جدول patient است که حاوی اطلاعات مربوط به حسابداری و امور مالی بیمارستان است.
 - جدول patient دارای کلید اصلی patient id است که حاوی اطلاعات بیمار بستری شده در بیمارستان است.
 - جدول labworker دارای کلید اصلی labworker id است که این جدول حاوی اطلاعات مسئول آزمایشگاه بیمارستان است.
 - جدول labtest دارای کلید اصلی labtest id و کلید خارجی patient id , labworker id است که این جدول حاوی اطلاعات آزمایش گرفته شده از بیمار است.
 - جدول receptionist با کلید اصلی receptionist id حاوی اطلاعات بخش پذیرش بیمارستان است.
 - جدول admin با کلید اصلی admin id حاوی اطلاعات ادمین کل این برنامه است.
- از آنجاییکه طراحی نرم افزار پروژه ما از نوع Data Base First بود، در ابتدا این موجودیت ها در محیط پایگاه داده MSSQL طراحی شده اند. نوع و مقادیر قابل قبول جداول وارد شده و و اقدامات و تمهیدات اولیه لازم برای طراحی پایگاه داده انجام گرفت.

Column Name	Data Type	Allow Nulls
DoctorID	int	<input type="checkbox"/>
DoctorName	nvarchar(50)	<input type="checkbox"/>
DoctorSurname	nvarchar(50)	<input type="checkbox"/>
DoctorTC	nvarchar(11)	<input type="checkbox"/>
DoctorAddress1	nvarchar(400)	<input type="checkbox"/>
DoctorAddress2	nvarchar(400)	<input checked="" type="checkbox"/>
DoctorMobile1	nvarchar(25)	<input type="checkbox"/>
DoctorMobile2	nvarchar(25)	<input checked="" type="checkbox"/>
DoctorGender	nvarchar(3)	<input type="checkbox"/>
DoctorAge	int	<input type="checkbox"/>
DoctorEmail	nvarchar(100)	<input type="checkbox"/>
DoctorField	nvarchar(100)	<input type="checkbox"/>
DoctorPassword	nvarchar(50)	<input type="checkbox"/>

تصویر ۲_ تصویر از محیط طراحی جداول پایگاه داده

سپس دسترسی های مربوط به کاربر های سیستمی بطور جداگانه طراحی شد تا وظیفه طراحی، مشاهده و یا مدیریت از یکدیگر جدا شوند. این عمل به منظور بیشتر کردن سطح امنیت پروژه منظور گردید. تصویر مربوط به سطوح دسترسی اعضای گروه نیز در ذیل آمده است:

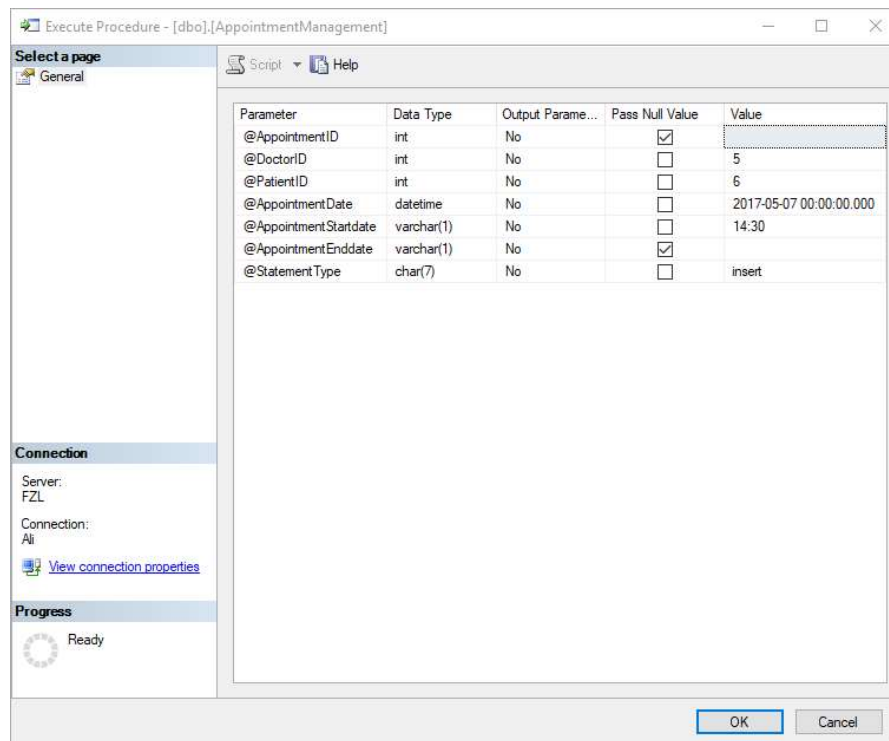


تصویر ۳ سطوح دسترسی جداگانه اعضای گروه به پایگاه داده

پس از طراحی پایگاه داده نوبت به طراحی رویه های ذخیره سازی یا همان **Store Procedure** ها می رسد. برای راحتی بیشتر برنامه نویسی تعدادی رویه ذخیره سازی مفید طراحی گشت که به اختصار به یکی از آنها اشاره می کنیم. رویه ذخیره سازی مورد نظر **AppointmentManagement** می باشد که وظیفه مدیریت رکورد های جدول قرار ملاقات ها را به عهده دارد. به نحوی که این رویه قادر به اضافه، حذف و یا بروز رسانی مقادیر جدول را بصورت کارآمد تری می باشد. نحوه اجرای رویه هم بدین گونه است که برای عمل اضافه کردن، پارامتر **StatementType** را برابر **insert** قرار می دهیم، سپس تمامی ستون های مورد نیاز جدول قرار ملاقات را به جز مقدار **AppointmentID** که یک مقدار یکتا و از نوع **Identity Specification** می باشد را در پنجره اجرای رویه وارد میکنیم. پس از اجرای رویه شاهد اضافه شدن رکورد به جدول در پایگاه داده خواهیم بود.

متد دیگر رویه ذخیره سازی **delete** می باشد که با وارد کردن **AppointmentID** رکورد مورد نظر در پارامتر ورودی رویه قادر به حذف رکورد هستیم.

عملکرد متد بروز رسانی جدول هم تقریباً مشابه متد اضافه کردن (**insert**) می باشد و از توضیح اضافه در این رابطه خودداری می کنیم.



تصویر ۴ عمل اضافه کردن رکورد به جدول قرار ملاقات

سپس این رویه برای جداول مورد نیاز دیگر نیز طراحی شد و اسکریپت ها از نوع Function و Trigger نیز طراحی و به پایگاه داده اضافه شدند که در پایگاه داده قابل مشاهده است.

سپس در محیط ویژوال استودیو یک پروژه بنام Hospitool ایجاد گردید. نحوه نام گذاری پروژه نیز در زیر آورده شده است:



سپس مدل (و به تبع آن اشیاء و Property های آنان) با استفاده از ADO.net Entity Data Model از پایگاه داده طراحی شده استخراج گردید و به پروژه اضافه گشت.

تصویر ۵_ استخراج مدل از روی پایگاه داده در محیط ویژوال استودیو

در ادامه از Entity Framework نسخه پنجم استفاده شد و برای طراحی صفحات یا اصطلاحا Win Form ها از Microsoft ۵,۲ Dot.Net Framework استفاده گشت. نحوه طراحی در Entity Framework نیز سراسر (اصطلاحا Straight Forward می باشد).

نحوه اجرای برنامه:

محیط اصلی برنامه صفحه ورودی برنامه می باشد که در شکل شماره شش قابل مشاهده می باشد.

تصویر ۶ صفحه ورود به برنامه

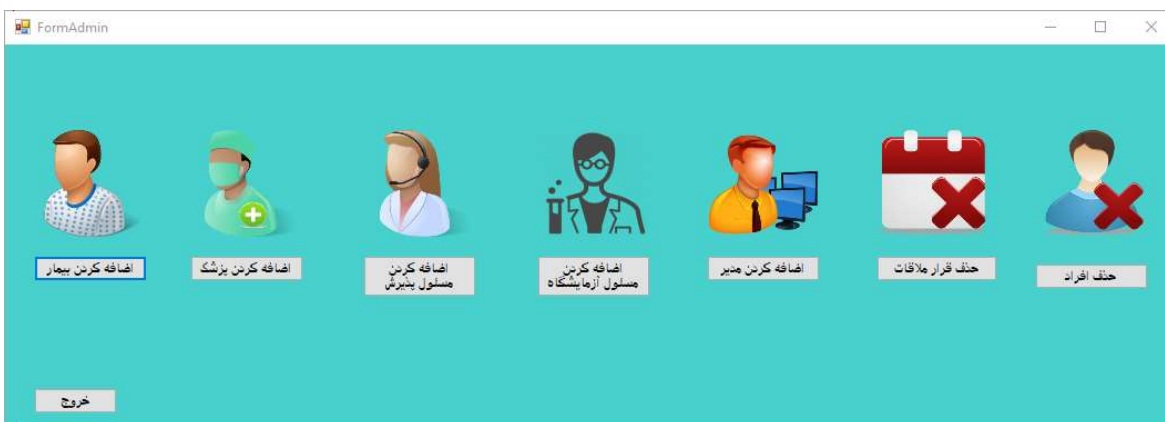
همانطور که در شکل قابل مشاهده است هر فردی (جدای از نوع دسترسی و موقعیت خود در سامانه بیمارستان) بایستی یک شماره شناسایی ثبت شده در سامانه (به عبارت بهتر در پایگاه داده) سامانه داشته باشد. در غیر اینصورت این فرد قطعاً بیماری خواهد بود که تا به حال مراجعه ای به بیمارستان (و به تبع آن سامانه مدیریت بیمارستان) نداشته است. که در آن صورت می تواند از آخرین گزینه استفاده کرده و برای خود یک حساب کاربری بیمار ایجاد کند.

در صورت داشتن حساب کاربری هر فرد میتواند با در دست داشتن شماره شناسایی و رمز عبور خود و مشخص کردن نوع دسترسی خود به سامانه وارد شود.

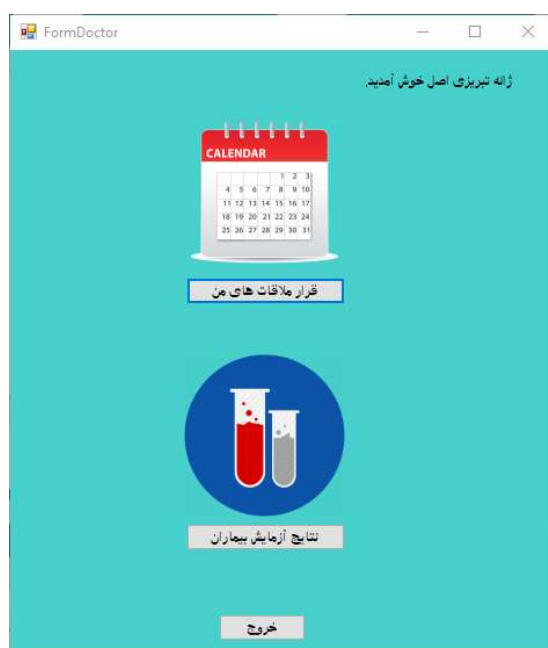
انواع دسترسی های ممکن به سامانه به پنج دسته پزشک، کارکنان آزمایشگاه، مسئول پذیرش، بیمار و یا مدیر کل تقسیم بندی میشود که هر فرد با توجه به موقعیت خود در بیمارستان با سطح دسترسی خود به سامانه مدیریتی وارد میشود.

نکته قابل توجه در امنیت بیماران ، کارکنان و پزشکان در آن سامانه این است که در هنگام ثبت نام و یا عضو شدن آنها در سامانه و تعیین رمز عبور خود، رمز عبور آنها با الگوریتم رمزگذاری MD۵ رمز گذاری شده و رمز عبور آن ها بصورت رمز شده در پایگاه داده ذخیره می گردد. و در هنگام احراز هویت آنها در سامانه مقادیر MD۵ شده ورودی وارد شده توسط کابر مربوطه با MD۵ ذخیره شده در پایگاه داده مقایسه می شود. و از آنجاییکه رمز گذاری MD۵ بصورت brute force قابل شکسته شدن نمی باشد این امر ویژگی مهم این سامانه در رابطه با امنیت و حفظ امنیت کاربران می باشد.

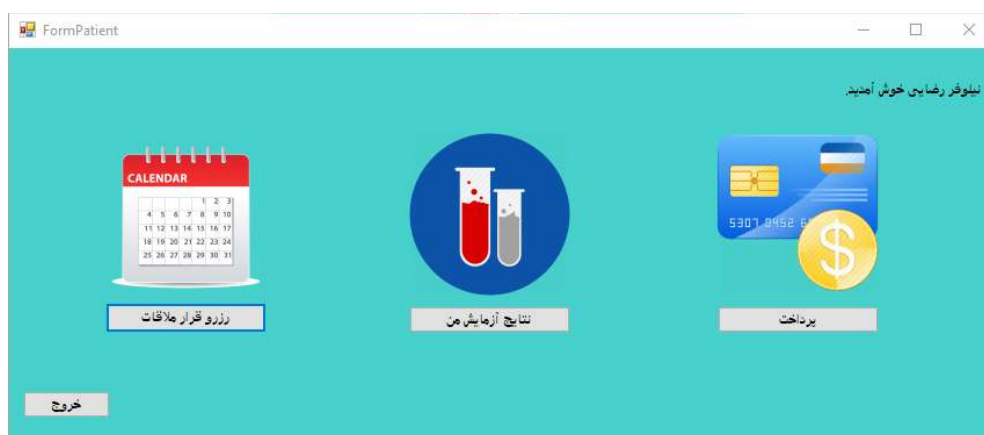
در ادامه تصاویر دسترسی های مختلف آورده شده است، لازم به ذکر است که گزینه های در دسترس در هر نوع دسترسی همراه با شکل در منو قابل مشاهده است و از توضیح قابلیت ها در هر حالت خودداری میکنیم.



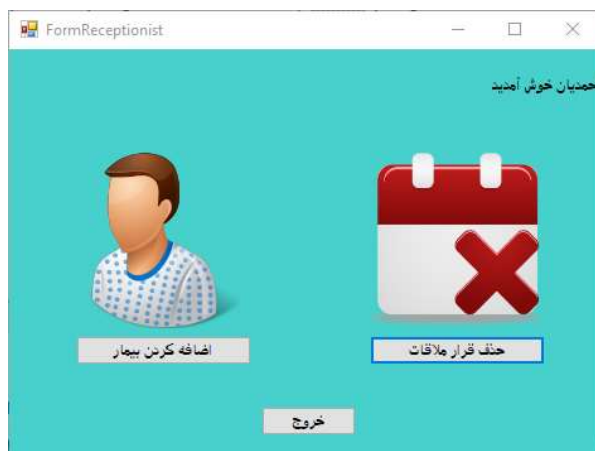
تصویر ۷ دسترسی مدیریت به سامانه



تصویر ۸ دسترسی پزشک به سامانه



تصویر ۹ دسترسی بیمار به سامانه



تصویر ۱۰ دسترسی مسئول پذیرش به سامانه



تصویر ۱۱ دسترسی مسئول آزمایشگاه به سامانه

نتیجه گیری:

هدف پروژه زیر در حالت کلی پیاده سازی موارد بوده است:

۱. حذف رویه های تکراری و غیرضروری
۲. استفاده از کامپیوتر بعنوان ابزار کار
۳. استخراج آمار و اطلاعات به روشهای سریع تر و دقیق تر
۴. ایجاد یک روش و سیستم کاری مدرن و استاندارد بیمارستانی
۵. برقراری ارتباط داده ها با سیستمهای مهندسی پزشکی
۶. برقراری ارتباط اطلاعاتی بین بیمارستانها و مراکز درمانی در سطح کشور
۷. رسیدن به یک بانک اطلاعاتی توزیع شده در سطح کشور و ایجاد ارتباط آن با شبکه های بهداشت جهانی
۸. ارتقای سطح بهداشت جامعه