

گزارش پیاده سازی پایگاه داده سیستم اطلاعاتی آچاره

محمد مهدی رحیم سیرت ۹۷۱۲۳۵۸۰۱۶

محمد حسین رستمی ۹۷۱۲۳۵۸۰۱۷

دیپارتمان مهندسی کامپیوتر، دانشکده فنی مهندسی بوعلی سینا، ایران

E-mail: rostamee1@gmail.com

چکیده: تکنولوژی روز و پایگاه داده ها امروزه نقشی مهم در دسته بندی و مدیریت جوامع انسانی برای مدیران و گسترش عدالت ایفا می کنند، زبانی که برای پایگاه داده ها در نظر گرفته شده، استاندارد SQL است که البته بخشی از پایگاه داده ها از استاندارد آن پیروی نمی کنند، سیستم اطلاعاتی آچاره که در حوزه خدمات خانگی به مردم کمک می کند، نیازمند پایگاه داده ای برای دسته بندی و سرویس اتوماتیک به کاربران خود است، در این گزارش به بررسی پیاده سازی پایگاه داده ای برای ارائه این خدمت پرداخته شده است، با استفاده از رابط کاربری قدرتمندتر نیز می توان این سیستم را در عمل جایگزین سیستم دستی کرد و از مزایای سرعت و ماندگاری آن بهره مند شد.

کلمات کلیدی:

جدول، نمودار موجودیت رابطه، کوئری، رابط کاربری، داده های تست

۱- مقدمه:

قرن بیست و یکم که از آن به عنوان قرن انفجار اطلاعات یاد می شود، پیشرفت های خود را مدیون گسترش صنعت کامپیوتر و دیجیتال است، به طوری که هم اکنون تکنولوژی به جزء جدایی ناپذیر زندگی بشر تبدیل شده است، که اگر لحظه ای از کار بیفتد موجب خسارت های بیشماری به زندگی جوامع می شود.

یکی از مهمترین تکنولوژی های قرن ۲۱، که با سرعت بالایی در حال گسترش است، توانایی ذخیره اطلاعات به صورت دیجیتالی بر روی یک وسیله ذخیره اطلاعات از جمله HDD و SSD می باشد، که نیازمندی جوامع به کاغذ و مشکلات نگهداری آن را به شدت کاهش داده است.

چالش جدیدی که در حال کار با تکنولوژی های جدید ذخیره اطلاعات به وجود آمد، موجب شد تا دانشمندان و مهندسان کامپیوتر، به فکر ساخت ابزاری به منظور دسته بندی و مرتب سازی اطلاعات، برای سهولت دسترسی و سرعت بالای آن باشند.

امروزه پایگاه داده ها نقش دسته بندی، کنترل و دیگر امور مربوط به نگهداری و دسترسی اطلاعات را ارائه می دهند، که بخشی از آن ها از استاندارد SQL استفاده می کنند و بخشی دیگر از آن استفاده نمی کنند.

"پایگاه داده به مجموعه ای از داده های مرتبط، ساختارمند یا سازمان یافته گفته می شود که دسترسی به این اطلاعات معمولاً از طریق سیستم مدیریت پایگاه داده یا (DBMS) صورت می گیرد. سیستم مدیریت پایگاه داده متشکل از مجموعه یکپارچه از نرم افزارهای رایانه ای است که به کاربران اجازه می دهد تا با یک یا چند پایگاه داده ارتباط یافته و به اطلاعات موجود در پایگاه داده دسترسی یابند" [۱]

امروزه با توجه به زیاد شدن فواصل و عدم به صرفه بودن عزیمت به سوی کارگاه یا مغازه متخصص مورد نظر ما، و هم چنین گسترش خدمات اپلیکیشن موبایل و سایت ها، نیاز به یک برنامه که بتواند متخصص ترین افراد را که ابتدا مغازه ای برای کار در اختیار ندارند و در گام بعد، کار خدماتی را شغل دوم خود می دانند و قصد وقت صرف کردن برای پیدا کردن مشتری ندارند، بسیار حس می شود.

مشتریان همیشه دوست می دارند که یک بار برای هر کاری هزینه کنند و بهترین خدمت را دریافت کنند، مقایسه متخصصان حاضر در یک حوزه همواره راه حل هر مشتری می باشد، اما چون هر مشتری نمی تواند هر دفعه یک متخصص جدید را تست کند، ترجیح می دهد اگر دومی از اولی بهتر بود او را انتخاب کند، اما با توجه تکنولوژی روز که پایگاه داده ها و هوش مصنوعی در اختیار ما گذاشته است لازم نیست هر مشتری تمام متخصصان را تست کند تا میزان مهارت آن ها را متوجه شود، کافیست امتیازی که دیگر مشتریان به او را داده اند بررسی کند تا بهترین متخصص را انتخاب کند.

اپلیکیشن ارائه خدمت، می تواند به راحتی تخصص هر متخصصی را ثبت کند و با نشان دادن امتیاز هر متخصص و توضیح درباره هر تخصص، به مشتری این کمک را بکند که بهترین انتخاب را داشته باشد.

امروزه دیگر به شیوه سنتی درآمد کمتری می توان کسب کرد، زیرا علاقه جامعه به استفاده از اپلیکشین ها و درخواست خدمت و کار مورد نیاز خود از راه دور بدون ایاب ذهاب و دردرس های خرید و فروش، یکی از دلایل مراجعه کمتر به بازارهای سنتی است .

۲- شرح :

۲-۱- طرح پیشنهادی سیستم اطلاعاتی آچاره :

۲-۱-۱-وظایف سیستم اطلاعاتی آچاره :

- ۱- ثبت اطلاعات مشتری و خدمات مورد نیاز او
- ۲- ثبت نام پرسنل و متخصصان خواستار کار کردن در سازمان به همراه خدماتی که می توانند ارائه بدهند
- ۳- محاسبه هزینه خدمات برای مشتری
- ۴- ذخیره رزومه متخصصان برای ارائه به مشتری
- ۵- حفاظت از آدرس های منزل و شماره تلفن های مشتری
- ۶- ارائه تمامی خدمات موجود برای مشتری

۲-۱-۲-امکانات :

مشتریان :

- امکان اضافه نمودن مشتری جدید به همراه آدرس و شماره تلفن آن
- امکان ویرایش آدرس مشتری
- ثبت سابقه استخدام مشتری
- ثبت کیف پول مشتری و اضافه و کم نمودن مبلغ آن
- حذف خدمت خواسته شده مشتری
- حذف حساب کاربری مشتری
- ویرایش زمان و مکان ارائه خدمت
- انتخاب آپشن خدمات خود
- امکان مشاهده رزومه متخصصان
- ثبت شکایت از متخصص

متخصصان:

- ثبت تخصص خود
- ثبت مهارت خود در تخصص (آپشن های تخصص)
- ثبت نتایج آزمون راستی آزمایی
- ثبت اطلاعات نام و سابقه کار
- امکان اضافه شدن هر خدمت ارائه شده به سابقه کار
- ثبت و ویرایش رضایت مشتری از متخصص و ثبت امتیاز متخصص
- امکان انتخاب و حذف یک خدمت در صف به مشتری
- امکان مشاهده صورت حساب خدمت ارائه داده خود
- امکان ثبت و ویرایش ساعت های کاری خود
- ثبت شکایت از مشتری برای رخداد یک اتفاق ضد قانون

تخصص :

- تعریف آپشن های هر تخصص

- ثبت تعداد بار درخواست هر تخصص
- ثبت اطلاعات و عکس تخصص برای معرفی به مشتری
- ثبت و ویرایش قیمت تخصص
- ویرایش تعداد بار درخواست هر تخصص
- اضافه نمودن آپشن به هر تخصص

نمایندگی ها :

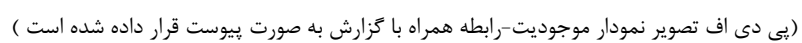
- ثبت اطلاعات و آدرس ساختمان نمایندگی جدید در شهر مشخص
- ثبت ساعات کاری هر نمایندگی در هر شهر
- ثبت شماره تلفن برای هر نمایندگی
- تعریف و ویرایش و حذف پشتیبانان حاضر در نمایندگی
- ثبت میزان دستمزد هر پشتیبان در هر نمایندگی
- مشخص نمودن وظیفه هر پشتیبان در نمایندگی
- ثبت بودجه واریز شده برای هر نمایندگی
- تشکیل درخت مدیریت برای نمایندگی ها

پشتیبان :

- ثبت اطلاعات پشتیبان
- ثبت و ویرایش ساعات کاری هر پشتیبان برای خدمت رسانی
- ثبت و ویرایش سابقه پشتیبانی و رضایت مشتری از پاسخگویی او
- ثبت و ویرایش میزان دانش پشتیبان
- ثبت فیش حقوقی هر پشتیبان
- ثبت محل مورد خدمت پشتیبان

۲-۲- نمودار موجودیت رابطه :

پس از تعیین نیازمندی های مسئله، و طرح پیشنهادی برای شروع کار، برای طراحی پایگاه داده، یک نمودار موجودیت-رابطه نیاز است، تا با کمک آن روابط ترسیم شوند و با تکنیک های تبدیل نمودار موجودیت-رابطه به جدول، به جدول هایی جهت درج در پایگاه داده برسیم.



۳-۲- تبدیل به جدول :

پس از ترسیم نمودار نمودار موجودیت رابطه، با استفاده از تکنیک ها و قرارداد های موجود در کتاب پایگاه داده ها^۱، جدول مربوط به هر موجودیت و رابطه را تعیین می کنیم .

تخصص					
کد تخصص	تعداد بار درخواست	دستمزد	شماره عکس تخصص	نام	نمره حد نصاب آزمون
مهارت های تخصص					
تخصص					
مهارت های لازم	کد تخصص				
تخصص خودرو					
تخصص					
کد تخصص خودرویی	شماره جعبه ابزار خودرو				
تخصص کامپیوتر					
تخصص					
کد تخصص کامپیوتری	شماره لپ تاپ سازمانی				
تخصص آب و فاضلاب					
تخصص					
کد تخصص آبی	شماره جعبه ابزار باز کردن فاضلاب				

جدول شماره ۱

^۱ Fundamentals of Database Systems, Pearson , Ramez Elmasri and Shamkant B. Navathe, Seventh Edition, Chapter ۹

متخصص								
شماره پرسنلی	نام و نام خانوادگی	شماره موبایل	ساعت کاری	میانگین امتیاز	جنسیت	پلاک	خیابان	شهر
کوچه								
سابقه استخدام								
متخصص								
شماره پرسنلی	تخصص انجام شده	تعداد	امتیاز					
نماینده								
کد نمایندگی								
تعداد پشتیبان ها	بودجه	پلاک	خیابان	شهر	کوچه	شماره پرسنلی	مدت تاریخ شروع	مدت تاریخ شروع
ساعت های کاری نماینده								
نماینده								
کد نمایندگی	ساعت کار							
مشتری								
شماره عضویت	شماره موبایل	اعتبار حساب	جنسیت	نام و نام خانوادگی	پلاک	خیابان	شهر	کوچه
پشتیبان								
نماینده								
شماره پرسنلی	نام و نام خانوادگی	جنسیت	میزان حقوق	نمره دانشی	شماره موبایل	کد نمایندگی		
آدرس های پشتیبان								
پشتیبان								
شماره پرسنلی پشتیبان	پلاک	خیابان	شهر	کوچه				
تماس با پشتیبان								
مشتری								
شماره عضویت مشتری	شماره پرسنلی پشتیبان	امتیاز پشتیبان	کد تماس					
متخصص بودن								
متخصص								
شماره پرسنلی	کد تخصص	سطح تخصص						


```

(
    CarProfCode int not null check(CarProfCode <= 10000 ) primary key,
    CarToolBoxNo int unique,
    Foreign key ( CarProfCode ) References Profession( ProfCode ),
);

Create Table CompProf
(
    CompProfCode int not null check(CompProfCode <= 10000 ) primary key,
    CompToolBoxNo int unique,
    Foreign key ( CompProfCode ) References Profession( ProfCode ),
);

Create Table WaterProf
(
    WaterProfCode int not null check(WaterProfCode <= 10000 ) primary key,
    WaterToolBoxNo int unique,
    Foreign key ( WaterProfCode ) References Profession( ProfCode ),
);

Create Table Expert
(
    PersenelCode int not null primary key,
    NameAndFamily varchar( 100 ) not null,
    PhoneNumber varchar(11) not null,
    WorkingHours varchar( 10 ) not null,
    AvgPoints int not null,
    Sex varchar(1) not null check( Sex in ( 'F', 'M' ) ),
    HouseNo varchar(12),
    StreetNo varchar(20),
    City varchar(20),
    Alley varchar(20)
);

Create Table EmpHistory
(
    PersenelCode int not null,
    foreign key ( PersenelCode ) References Expert( PersenelCode ),
    ProfessionType varchar(100) not null,
    Primary Key ( PersenelCode, ProfessionType ),
    ProfNumber int,
    Point int,
);

Create Table Agency
(
    AgencyCode int primary key,
    SupporterNum int not null,
    Budget int not null check( Budget <= 100000 and Budget >= 0 ),
    HouseNo varchar(12),
    StreetNo varchar(20),
    City varchar(20),
    Alley varchar(20),
    ManagerCode int not null,
    StartDate Date,
);

Create Table AgencyWorkingHours
(
    AgencyCode int,
    WorkingHour varchar(12),

```



```

Foreign Key ( AgencyCode ) References Agency( AgencyCode ),
Primary Key ( AgencyCode, WorkingHour ),
);

Create Table Client
(
    MemberShipCode int not null Primary Key,
    PhoneNumber varchar(11) not null,
    AccountCredit int,
    Sex varchar(1) not null check( Sex in ( 'F', 'M' ) ),
    NameAndFamily varchar(100) not null,
    HouseNo varchar(12),
    StreetNo varchar(20),
    City varchar(20),
    Alley varchar(20),
);

Create Table Supporter
(
    PersenelCode int primary key,
    NameAndFamily varchar(100) not null,
    Sex varchar(1) not null check( Sex in ( 'F', 'M' ) ),
    Wage int not null,
    KnowledgePoint int not null check( KnowledgePoint >= 0 and KnowledgePoint <= 100 ),
    PhoneNumber varchar(11) not null,
    AgencyCode int not null,
    Foreign Key (AgencyCode) References Agency( AgencyCode )
);

Alter Table Agency
Add Foreign Key ( ManagerCode ) References Supporter( PersenelCode );

Create Table SupportersAddress
(
    PersenelCode int not null,
    Foreign Key ( PersenelCode ) References Supporter( PersenelCode ),
    HouseNo varchar(12),
    StreetNo varchar(20),
    City varchar(20),
    Alley varchar(20),
    Primary Key ( PersenelCode, HouseNo, StreetNo, City, Alley ),
);

Create Table CallingSupporter
(
    MemberShipCode int not null Primary Key,
    PersenelCode int not null,
    Foreign Key ( PersenelCode ) References Supporter( PersenelCode ),
    Foreign Key ( MemberShipCode ) References Client( MemberShipCode ),
    CallPoint int check( CallPoint <= 100 and CallPoint >= 0 ),
    CallCode int not null,
);

Create Table HaveProfession
(
    PersenelCode int not null Primary Key,
    Foreign Key ( PersenelCode ) References Expert( PersenelCode ),
    ProfCode int not null,
    Foreign Key ( ProfCode ) References Profession( ProfCode ),
    ProfLevel varchar( 8 ) Check( ProfLevel in ( 'High', 'Low', 'Medium' ) ),
);

```

```
);

Create Table TakeService
(
    PersenelCode int not null,
    Foreign Key ( PersenelCode ) References Expert( PersenelCode ),
    MemberShipCode int not null,
    Foreign Key ( MemberShipCode ) References Client( MemberShipCode ),
    Primary Key ( PersenelCode, MemberShipCode ),
    AddressDesc varchar( 100 ) not null,
    TimeDesc time not null, --09:12:22
    TrafficCost varchar( 20 ),
    ProfessionCost varchar( 20 ),
    ExpertPoint int check( ExpertPoint >= 0 and ExpertPoint <= 100 ),
    ServiceCode int not null,
);

Create Table Complaining
(
    PersenelCode int not null,
    Foreign Key ( PersenelCode ) References Expert( PersenelCode ),
    MemberShipCode int not null,
    Foreign Key ( MemberShipCode ) References Client( MemberShipCode ),
    Primary Key ( PersenelCode, MemberShipCode ),
    ComplainDesc varchar( 100 ),
    ComplainCode int not null,
);
```

کد شماره ۱

۲-۵-درج داده های تست در جدول ها :

زمانی که پایگاه داده و رابط کاربری آن تحویل ادمین می شود، داده های واقعی سازمان توسط ادمین وارد پایگاه داده می شود و گزارش هایی که کاربران و ادمین ها به آن نیاز دارند توسط همان رابط کاربری که درج داده صورت گرفته، دریافت می شود که در این قسمت طراحی رابط کاربری مناسب حائز اهمیت می باشد.

به جهت تست عملکرد پایگاه داده لازم بود داده های تست وارد پایگاه داده شوند، برای وارد کردن این داده ها دو راه بصری و نوشتاری در SQL Server وجود دارد، که راه نوشتاری آن گزیده شد تا به کار سهولت ببخشد .

نمونه کدی که برای وارد کردن داده در جدول های پایگاه داده استفاده می شود در کد شماره ۲ نمایش داده شده است .

Use AcharehDB

```
Insert Into TakeService
(
    PersenelCode,
    MemberShipCode,
    AddressDesc,
    TimeDesc,
    TrafficCost,
    ProfessionCost,
    ExpertPoint,
    ServiceCode
)
Values ( 3, 102, 'Hamedan Dast Rast', '09:22:20', 200, 4000, 66, 1111 );
```

کد شماره ۲

پس از درج داده های تست، با استفاده از دستورات Select Top ۱۰۰ که به صورت بصری با حرکت نشان گر و کلیک راست بر روی هر جدول امکان پذیر است، می توانیم داده های درج شده در جداول را ببینیم .

	ProfCode	ReqNum	ProfWage	ImageNo	ProfName	TestQuorum
1	1	3	2000000	NULL	Fazelab	60
2	2	5	1500000	NULL	Computer	70
3	3	10	4000000	NULL	Khodro	90

شکل شماره ۲ - جدول متخصص

100 %		
Results Messages		
	Skill	ProfCode
1	shostane fazelab	1
2	Baz kardan soorakh	1
3	gereftegi va baz kardan fazelab	1
4	how to crack	2
5	install photoshop	2
6	install Windows	2
7	nasb vs code	2
8	Pak kardan Fazelab	1
9	Raditor Taemir	3
10	repair tire	3
11	Sakht machine	3
12	taemire radiator	3

شکل شماره ۳ - جدول مهارت های تخصص

Results Messages		
	CarProfCode	CarToolBoxNo
1	3	6505

شکل شماره ۴ - جدول تخصص خودرو

100 %		
Results Messages		
	CompProfCode	CompToolBoxNo
1	2	7404

شکل شماره ۵ - جدول تخصص کامپیوتر

100 %

Results Messages

	WaterProfCode	WaterToolBoxNo
1	1	3033

شکل شماره ۶ - جدول تخصص آب و فاضلاب

100 %

Results Messages

	PersenelCode	NameAndFamily	PhoneNumber	WorkingHours	AvgPoints	Sex	HouseNo	StreetNo	City	Alley
1	1	Jack Wiliam	0992999222	10-12	66	M	103	Haft Lashgar	Ghazvin	Boolandian
2	2	Donald Blinkton	0932229222	10-14	70	M	102	Ostad Moein	Tehrun	Khoram
3	3	Michael Johnny	0918999222	12-14	50	M	101	Haft Lashgar	Ghazvin	Boolandian
4	4	Joe Biden	09155529222	08-14	44	M	099	Ostad Moein	Tehrun	Khoram
5	5	David Banker	0900999222	10-12	90	M	109	Haft Lashgar	Ghazvin	Boolandian
6	6	Henry Scallate	0911229222	12-14	77	M	112	Ostad Moein	Tehrun	Khoram

شکل شماره ۷ - جدول متخصص

100 %

Results Messages

	PersenelCode	ProfessionType	ProfNumber	Point
1	1	Fazelab Ghazvin	4	55
2	3	Repair Damaged Computer	2	99
3	5	Made Raditor	1	92

شکل شماره ۸ - جدول سابقه استخدام

100 %

Results Messages

	AgencyCode	SupporterNum	Budget	HouseNo	StreetNo	City	Alley	ManagerCode	StartDate
1	11	2	100000	114	Rafsanjani	Tehrun	Tork	1001	2000-01-26
2	12	2	60000	113	Torghooz	Ghazvin	Ubaba	1002	2001-01-26
3	13	2	50000	120	Fereshteh	Tehrun	Feresht	1003	2010-01-26

شکل شماره ۹ - جدول نمایندگی

100 %

Results Messages

	AgencyCode	WorkingHour
1	11	10-12
2	11	14-16
3	12	11-16
4	12	15-17
5	13	10-15

شکل شماره ۱۰ - جدول ساعت کاری نمایندگی

100 %

Results Messages

	MemberShipCode	PhoneNumber	AccountCredit	Sex	NameAndFamily	HouseNo	StreetNo	City	Alley
1	101	09186120555	700	M	Akbar Gholami	122	Khezr	Hamedan	Ostad
2	102	09306120555	900	M	Asghar Garichi	123	Kedr	Ghazvin	Shagerd
3	103	09226120555	1000	M	Gholam Shamsi	101	Ostadan	Hamedan	Sourati
4	104	09026120555	9500	M	Ezzat Solati	182	Niavaran	Tehrun	daneshjoo

شکل شماره ۱۱ - جدول مشتری

100 %

Results Messages

	PersenelCode	NameAndFamily	Sex	Wage	KnowledgePoint	PhoneNumber	AgencyCode
1	1001	Samira Mahmudi	F	4000	90	09189999999	11
2	1002	Somayye Afrasiaby	F	22000	99	09189129999	12
3	1003	Dorsa Jahany	F	4000	90	09929999999	13
4	1004	Mahla Khoshghadam	F	4000	90	09129999999	11
5	1005	Parisa Sarikhani	F	3000	30	09449999999	12

شکل شماره ۱۲ - جدول پشتیبان

100 %

Results Messages

	PersenelCode	HouseNo	StreetNo	City	Alley
1	1001	133	Vakil	Mashhad	Bahrami
2	1001	143	Vakil	Ghazvin	chahrami
3	1002	155	Mahdieh	Tehrun	Usef
4	1003	133	Vakil	Yazd	Bahrami
5	1004	133	Vakil	Kerman	Bahrami
6	1005	133	Vakil	Esfahan	Bahrami

شکل شماره ۱۳ - جدول آدرس های پشتیبان

100 %

Results Messages

	MemberShipCode	PersenelCode	CallPoint	CallCode
1	101	1003	86	1113
2	102	1001	88	1112
3	103	1003	85	1111
4	104	1002	5	1115

شکل شماره ۱۴ - جدول تماس با پشتیبان

100 %

Results Messages

	PersenelCode	ProfCode	ProfLevel
1	1	1	High
2	2	3	Low
3	3	2	Medium
4	4	2	Low
5	5	2	Low
6	6	2	Low

شکل شماره ۱۵ - جدول متخصص بودن

100 %

Results Messages

	PersenelCode	MemberShipCode	AddressDesc	TimeDesc	TrafficCost	ProfessionCost	ExpertPoint	ServiceCode
1	1	102	Hamedan Dast Chap	09:21:20.0000000	200	4000	1	11112
2	2	101	Hamedan DastChap va Rast	09:25:20.0000000	200	4000	85	11113
3	3	102	Hamedan Dast Rast	09:22:20.0000000	200	4000	66	11111
4	4	102	Ghazvin	08:22:20.0000000	200	6000	66	11114

شکل شماره ۱۶ - جدول استخدام کردن

100 %

Results Messages

	PersenelCode	MemberShipCode	ComplainDesc	ComplainCode
1	1	102	Soorakh Fazelab Basteh Shod	12001
2	3	103	Khoob Nist	12002
3	6	101	Az Ghiafash Khosham Nemiad	12003

شکل شماره ۱۷ - جدول شکایت کردن

۲-۶- نوشتن و تست گزارش های مورد نیاز کاربر:

از آن جایی که این یک سیستم اطلاعاتی واقعی نیست و گزارش های مورد نیاز واقعی کاربران مجهول می باشد، تعدادی گزارش انتخاب شدند تا عملکرد پایگاه داده تست بشود، گزارش هایی که از پایگاه داده گرفته می شوند عبارتند از :

۱- نام و نام خانوادگی مشتریانی که از شهر همدان هستند .

۲- شرح شکایت هایی که از متخصص شماره ۳ شده است .

۳- میانگین امتیاز و و امتیاز کار متخصص شماره ۲ که برای مشتری شماره ۱۰۱ خدمت کرده است

۴- نام و نام خانوادگی پشتیبانی که عضو شماره ۱۰۳ با او تماس گرفته است .

۵- سطح تخصص متخصصی که در مکان قزوین به مشتری ای که شهر او قزوین است خدمت ارائه داده است .

۶- ساعت کار نمایندگی های شهر تهران

۷- تعداد باری که متخصصی که تخصصش آب و فاضلاب است، خدمت ارائه داده است .

۸- نام و نام خانوادگی و نمره دانشی مدیری که در نمایندگی شهر تهران با بودجه صد هزار تومان ، مدیریت می کند .

۹- آدرس پشتیبانی که مشتری ای به نام اکبر غلامی با اون تماس گرفته است

۱۰- نام و نام خانوادگی مشتری ای که از متخصص به نام مایکل جانی خدمت گرفته است و از متخصص آب و فاضلاب به نام جک ویلیام شکایت کرده است و با پشتیبان به نام

سمیرا محمودی صحبت کرده است

گزارشات فوق، به زبان محاوره ای بیان شده اند و قابل فهم برای پایگاه داده نیستند، به همین جهت لازم است تا با استفاده از زبان SQL، با سیستم مدیریت پایگاه داده SQL

Server ارتباط برقرار شود و پاسخ مورد نظر دریافت شود .

```
Use AcharehDB

Select NameAndFamily
From Client
Where City = 'Hamedan';

Select ComplainDesc
From Complaining
Where PersenelCode = 3;

Select Expert.AvgPoints, TakeService.ExpertPoint
From TakeService, Expert
Where Expert.PersenelCode = 2 and Expert.PersenelCode = TakeService.PersenelCode and TakeService.MemberShipCode = 101 ;

Select Supporter.NameAndFamily
From Supporter, CallingSupporter
Where Supporter.PersenelCode = CallingSupporter.PersenelCode and CallingSupporter.MemberShipCode = 103;

Select HaveProfession.ProfLevel, Expert.NameAndFamily
From HaveProfession, TakeService, Client, Expert
Where HaveProfession.PersenelCode = TakeService.PersenelCode and TakeService.MemberShipCode = Client.MemberShipCode and
TakeService.AddressDesc = 'Ghazvin' and Client.City = 'Ghazvin' and Expert.PersenelCode = TakeService.PersenelCode ;

Select WorkingHour
From AgencyWorkingHours, Agency
Where AgencyWorkingHours.AgencyCode = Agency.AgencyCode and Agency.City = 'Tehrur';

Select ProfNumber
From EmpHistory, Expert, HaveProfession, Profession
Where Expert.PersenelCode = HaveProfession.PersenelCode and HaveProfession.ProfCode = Profession.ProfCode and
Profession.ProfName = 'Fazelab' and EmpHistory.PersenelCode = Expert.PersenelCode

Select NameAndFamily ,KnowledgePoint
From Supporter, Agency
where Supporter.PersenelCode = Agency.ManagerCode and Agency.Budget = 100000 and Agency.City = 'Tehrur'
```

شکل شماره ۱۸ - کد SQL برای گزارش های مذکور به ترتیب از بالا به پایین

```

Select SupportersAddress.City, SupportersAddress.StreetNo, SupportersAddress.HouseNo, SupportersAddress.Alley
From SupportersAddress, CallingSupporter , Client
Where Client.NameAndFamily = 'Akbar Gholami' and Client.MembershipCode = CallingSupporter.MembershipCode and
CallingSupporter.PersenelCode = SupportersAddress.PersenelCode

Select Distinct Client.NameAndFamily
From Client, Expert, TakeService, HaveProfession, Complaining, Profession
Where Client.MembershipCode = TakeService.MembershipCode and Expert.PersenelCode = TakeService.PersenelCode and
Expert.NameAndFamily = 'Michael Johnny' and (
Select Expert2.NameAndFamily
From Expert As Expert2, TakeService As TakeService2
Where Expert2.PersenelCode = TakeService2.PersenelCode and TakeService2.MembershipCode = Client.MembershipCode and
Expert2.PersenelCode = HaveProfession.PersenelCode and
Profession.ProfCode = HaveProfession.ProfCode and Profession.ProfName = 'Fazelab'
) in ( 'Jack Wiliam' )

```

شکل شماره ۱۹ – کد SQL برای گزارش های مذکور (ادامه شکل شماره ۱۸)

پس از اجرای دستورات SQL فوق، می بایست خروجی مورد نظر خود را از پایگاه داده دریافت کنیم.

همانطور که در شکل شماره ۲۰ مشخص است، و با بررسی روابط در شکل های شماره ۲ تا ۱۷ می توانیم به درستی دستورات خود پی ببریم.

Results				
NameAndFamily				
1	Akbar Gholami			
2	Gholam Shamsi			
ComplainDesc				
1	Khoob Nist			
AvgPoints		ExpertPoint		
1	70	85		
NameAndFamily				
1	Dorsa Jahany			
ProfLevel		NameAndFamily		
1	Low	Joe Biden		
WorkingHour				
1	10-12			
2	14-16			
3	10-15			
ProfNumber				
1	4			
NameAndFamily		KnowledgePoint		
1	Samira Mahmudi	90		
City	StreetNo	HouseNo	Alley	
1	Yazd	Vakil	133	Bahrami
NameAndFamily				
1	Asghar Garichi			

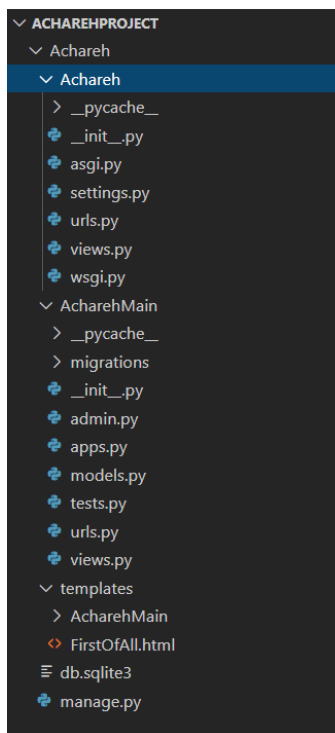
شکل شماره ۲۰ – خروجی گزارش ها از نرم افزار SQL Server Management Studio

۷-۲- رابط کاربری و ارتباط آن با پایگاه داده :

یکی از امکاناتی که سیستم های مدیریت پایگاه داده معروف دارند، امکان ارتباط آن ها با زبان های مختلف است، به طوری که با اکثر زبان های برنامه نویسی می توان به SQL Server متصل شد و گزارش های مورد نظر را از آن استخراج کرد.

اینجا با استفاده از زبان پایتون برنامه ای کد زده شده که می تواند با پایگاه داده ارتباط برقرار کند و گزارش های فوق را از پایگاه داده بگیرد و به کاربر نمایش بدهد .

هنگامی با استفاده از دستورات Django Framework یک پروژه را ساختیم، پوشه های زیر را در آن اضافه کردیم و یا وجودشان بررسی شد.

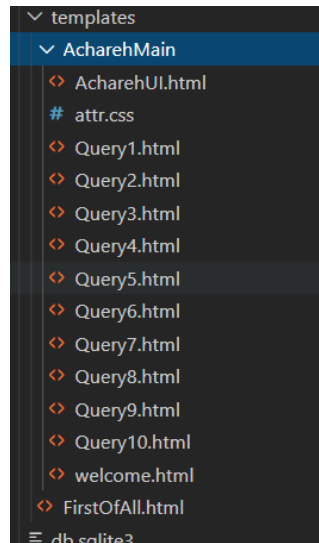


شکل شماره ۲۱ - پوشه های پروژه

پوشه اصلی Achareh در شکل شماره ۲۱ شامل خود برنامه می باشد که با استفاده از فایل های موجود در پوشه های پیشفرض خود و فایل manage.py، پروژه را شروع می کند .

پوشه AcharehMain در شکل شماره ۲۱ ، یکی از اپلیکیشن ها یا بخش های برنامه اصلی می باشد، که میتوانیم برای سهولت کار و افزایش مازولیتی برنامه از چنین امکانی با وارد کرد دستورات Django در خط فرمان استفاده کنیم .

پوشه Template در شکل شماره ۲۱ ، شامل فایل های Html می باشد که برای نمایش هر یک از بخش های برنامه استفاده می کنیم، اولین صفحه html با عنوان FirstOfAll دیده می شود که هنگام اجرای پروژه این صفحه به عنوان اولین صفحه تعیین شده است .



شکل شماره ۲۲ – پوشه template

هنگام اجرای هر گزارش، صفحه خاصی باید نمایش داده بشود تا اطلاعات خواسته شده به کاربر نمایش داده شود، و ستون ها مرتبط باشند، به همین دلیل به تعداد گزارش ها، صفحه Html کد زده شده است .

فایل AcharehUi.html نیز برای صفحه پیشفرض اپلیکیشن در نظر گرفته شده است.

```

1 <html>
2   <head>
3
4   </head>
5   <body>
6     <center>
7       <h1>1 گزارش شماره </h1>
8       <h2> نام و نام خانوادگی مشتریانی که از شهر همدان هستند </h2>
9       <hr/>
10      <table border = "1" >
11        <tr>
12          <th>نام و نام خانوادگی</th>
13        </tr>
14        {%for datadisplay in sqlserverconn%}
15        <tr>
16          <td>{{datadisplay.NameAndFamily}}</td>
17        </tr>
18        {%endfor%}
19      </table>
20      <br>
21      <a href="./AcharehUI" > بازگشت به عقب </a>
22    </body>
23  </html>

```

شکل شماره ۲۳ – نمونه کد html برای گزارشی از پایگاه داده

همانطور که در شکل شماره ۲۳ مشاهده می شود، عنوان گزارش و شماره آن بالای صفحه آورده می شود، که لازم است ستونی که خروجی پایگاه داده نمایش می دهد با خروجی گزارش یکی باشد، به همین جهت عنوان گزارش نام و نام خانوادگی تعیین شده است.

کدی که برای نمایش داده های دریافت شده از پایگاه داده استفاده می شود درون آکولاد ها نوشته شده است، که دیتا های دریافت شده و ذخیره شده در آبجکت sqlserverconn که در قسمت modules پروژه و اپلیکیشن تعریف کردیم را با پیمایش یک حلقه نمایش می دهد .

```
def Query1( request ) :
    conn = pyodbc.connect( 'Driver={SQL Server Native Client 11.0};'
                           'Server=DESKTOP-71AV8RA;'
                           'Database=AcharehDB;'
                           'Trusted_Connection=yes;' )
    cursor = conn.cursor()
    cursor.execute( "Select NameAndFamily From Client Where City = 'Hamedan';" )
    result = cursor.fetchall()
    return render( request, 'AcharehMain/Query1.html', { 'sqlserverconn' : result } )
```

شکل شماره ۲۴ – تابع تعریف شده در فایل views.py برای نمایش داده ها

در شکل شماره ۲۴، پس از استفاده از پکیج pyodbc و استفاده از توابع موجود در آن، یک اتصال به سیستم مدیریت پایگاه داده، با syntax فوق برقرار می کنیم و نام پایگاه داده مطلوب نظر که در اینجا AcharehDB است را در آن ذکر می کنیم تا به همان وصل بشود .

سپس از آبجکتی که برای اتصال به پایگاه داده ساخته شد، تابع cursor را فراخوانی می کنیم تا در آبجکت cursor انتقال داده شود، بعد از این تابع execute از آبجکت cursor را که گزارش خواسته شده به زبان SQL به عنوان ورودی آن داده می شود را فراخوانی می کنیم تا برای پایگاه داده ارسال شود.

پس از این اتفاقات، تمامی داده های واکنشی شده در آبجکت result ریخته شده و خصوصیات این آبجکت در sqlserverconn کپی می شود تا همراه با صفحه html مشخص شده برای آن، و با استفاده از تابع render، گزارش مطلوب را نمایش بدهد .

```
urls.py
Achareh > AcharehMain > urls.py > ...
1  from django.contrib import admin
2  from django.urls import path
3  from . import views
4
5  urlpatterns = [
6      path( 'AcharehUI', views.index ),
7      path( '', views.welcome ),
8      path( 'Query1', views.Query1 ),
9      path( 'Query2', views.Query2 ),
10     path( 'Query3', views.Query3 ),
11     path( 'Query4', views.Query4 ),
12     path( 'Query5', views.Query5 ),
13     path( 'Query6', views.Query6 ),
14     path( 'Query7', views.Query7 ),
15     path( 'Query8', views.Query8 ),
16     path( 'Query9', views.Query9 ),
17     path( 'Query10', views.Query10 )
18 ]
19
```

شکل شماره ۲۵ – فایل urls در اپلیکیشن AcharehMain

به جهت توانایی در استفاده مناسب از url ها در مرورگر اینترنتی، همانطور که در شکل شماره ۲۵ مشاهده می شود، باید مسیر های لازم و مشتق شده از مسیر فعلی را تعریف کنیم تا بتوانیم آدرس مشخصی برای هر صفحه html داشته باشیم و از آن برای تعریف رابط کاربری خود استفاده کنیم .

```
models.py ×
Achareh > AcharehMain > models.py > ...
1 from django.db import models
2
3 # Create your models here.
4
5 class sqlserverconn( models.Model ) :
```

شکل شماره ۲۶- فایل models.py برای تعریف کلاس اتصال

برای ذخیره کردن اطلاعات دریافتی از پایگاه داده، لازم است تا آبجکتی تعریف شود و آن آبجکت داده هایش را در صفحه html به ترتیب چاپ کند تا خروجی مناسبی به کاربر نمایش داده شود، نوشتن یک کلاس برای این آبجکت در شکل شماره ۲۶ انجام شده است .

به اچاره خوش آمدید

[برای ورود کلیک کنید](#)

شکل شماره ۲۷ - صفحه پیشفرض هنگام شروع پروژه

هنگامی که با دستور python manage.py runserver، پروژه شروع می شود، لینکی برای ورود که با استفاده از تگ a پیش بینی شده است، در وسط صفحه ملاحظه می شود، با کلیک بر روی آن وارد گزارش های برنامه می شود .

رابط کاربری اچاره

[گزارش شماره ۱](#)
[گزارش شماره ۲](#)
[گزارش شماره ۳](#)
[گزارش شماره ۴](#)
[گزارش شماره ۵](#)
[گزارش شماره ۶](#)
[گزارش شماره ۷](#)
[گزارش شماره ۸](#)
[گزارش شماره ۹](#)
[گزارش شماره ۱۰](#)

شکل شماره ۲۸ - گزارش های برنامه

پس از ورود به منوی اصلی برنامه، با انتخاب هر یکی از گزارش های مطلوب، می توان به محتوای مورد نظر که از پایگاه داده دریافت شده است دسترسی داشت .

گزارش شماره 1

نام و نام خانوادگی مشتریانی که از شهر همدان هستند

نام و نام خانوادگی
Akbar Gholami
Gholam Shamsi

بازگشت به عقب

شکل شماره ۲۹ - گزارش شماره ۱ در رابط کاربری

همانطور که در شکل شماره ۲۹ مشاهده می شود، گزارشی که برنامه برای کاربر به نمایش گذاشته است، همان خروجی هایی را دارد که در شکل شماره ۲۰ توسط سیستم مدیریت پایگاه داده به کاربر نمایش داده شد.

با توجه به نتایج دریافتی از رابط کاربری، می توان به صحت پروژه پی برد و آن را پایان یافته تلقی کرد .

۳- نتیجه گیری :

برای شروع طراحی یک پایگاه داده، رسم نمودار موجودیت رابطه و بررسی و درک آن گام اول می باشد، که معمولاً باید با مدیر سازمان بحث هایی صورت بگیرد تا تمامی نیازمندی های سازمان پوشش داده شود، سپس تبدیل این نمودار ها به جدول کار بعدی متخصص طراحی پایگاه داده است، پیاده سازی این جدول ها در پایگاه داده، بدون استفاده از DBMS ها امکان پذیر نیست و هزینه زیادی برای سازمان دارد، به همین دلیل با انتخاب یک DBMS مناسب همچون SQL Server می توان پایگاه داده و جدول های سازمان را به خوبی مدیریت کرد، متخصص پیاده سازی پایگاه داده می بایست به زبان SQL مسلط باشد تا هر گزارشی که مدیر سازمان نیاز داشت را بر آورده کند، هر چند این گزارش ها از پیش طلب شده است و گزارش جدیدی به صورت روزانه قرار نیست اضافه بشود، زیرا نیازمند تغییر در رابط کاربری نیز هست، هم اکنون در شرکت های برنامه نویسی فردی که این کار را انجام می دهد backend کار نامیده می شود که از احترام بالایی برخوردار است .

۴- منابع :

- کانال آپارات مریم قنبری، آموزش Django
- سایت Microsoft
- Fundamentals of Database Systems, Pearson , Ramez Elmasri and Shamkant B. Navathe, Seventh Edition
- [۱] سایت ویکی پدیا، پایگاه داده