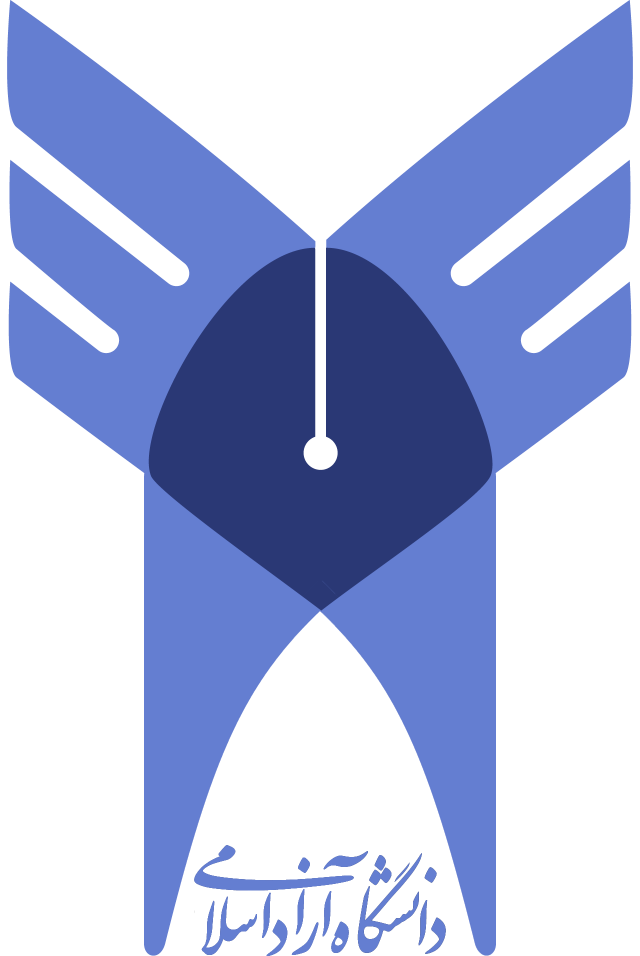
**به نام زیباترین زیبایی‌ها**



**نام دانشجو:** علی اسمعیلی

**شماره دانشجویی:** ۹۶۲۲۴۵۶۸۹

**استاد :** سرکار خانم مهندس شبنم اسبقی

**عنوان درس:** پروژه پایانی

**لینک کد‌های پروژه:** <https://github.com/realxoman/bachelor-project>

**مقدمه:**

این پروژه با هدف ساخت یک پنل کاربری برای معرفی، سفارش و پشتیبانی محصولات تهیه شده است. از امکانات این پروژه، به داشتن پنل کاربری اختصاصی برای کاربران و مدیران هستش و آمار سفارشات، تعداد محصولات و تعداد کاربران قابل مشاهده است.

**نمایی از پروژه:**

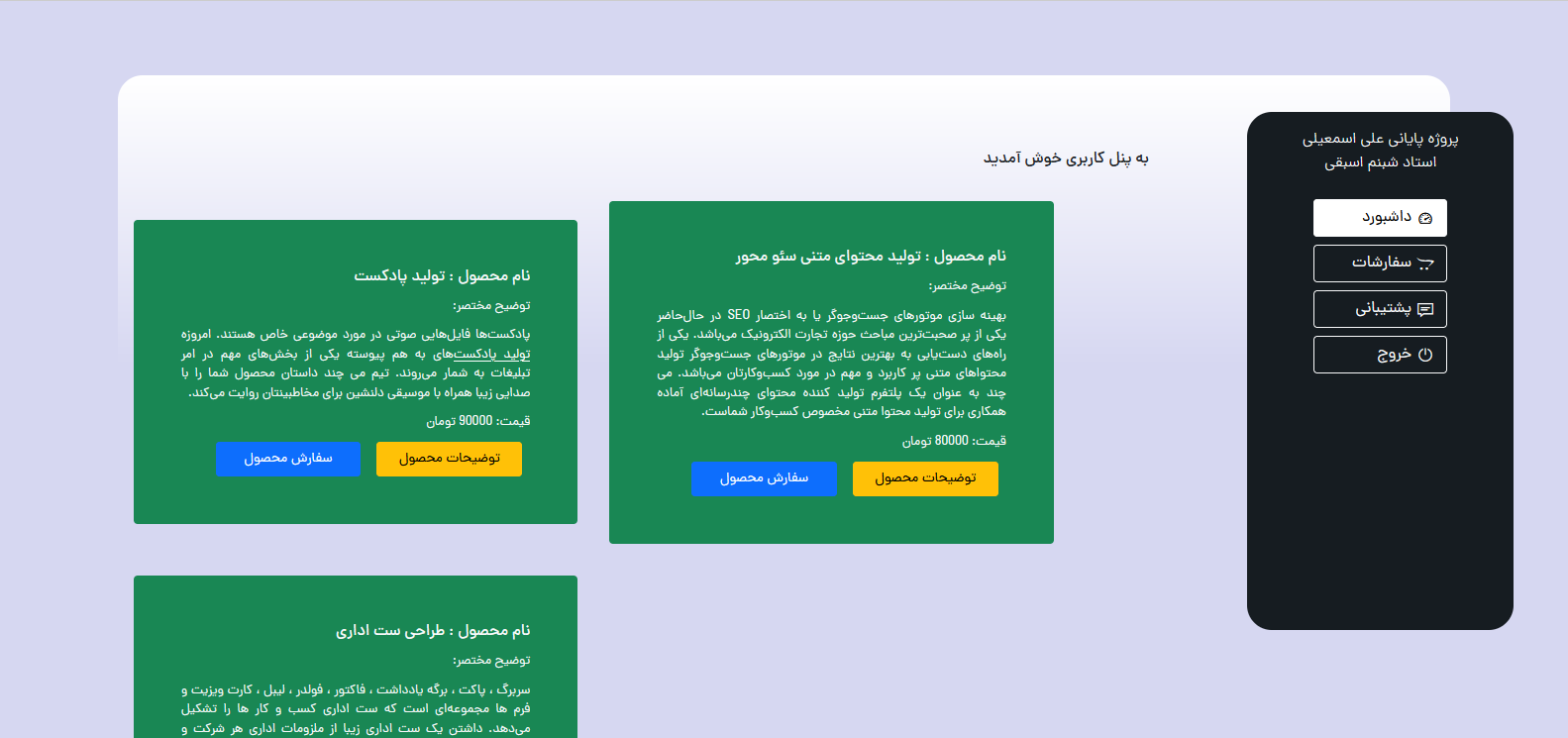
**بخش ورودی:**

****

**بخش ثبت نام:**

****

**پنل کاربران:**

****

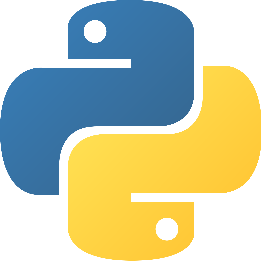
**پنل مدیریت:**

****

**تکنولوژی‌های مورد استفاده:**

**سمت سرور – Back-end :**

از زبان برنامه نویسی پایتون - Python استفاده شده است و فریمورک یا کتابخانه مورد استفاده نیز فریمورک جنگو – Django هست.



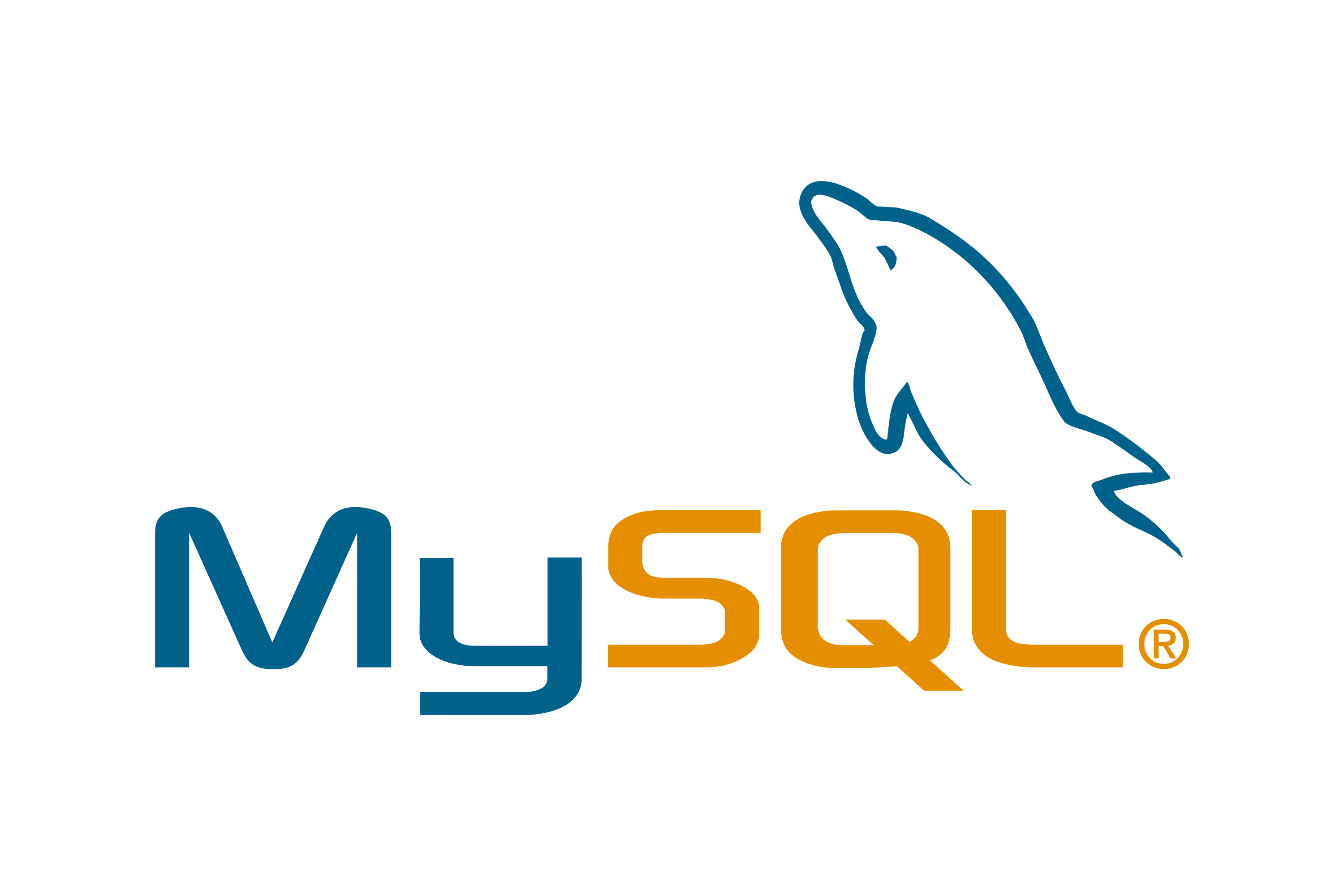
**سمت کاربر – Front-end :**

از زبان‌های HTML – CSS – Javascript استفاده شده است و در CSS برای خلاصه سازی کد‌ها از تکنولوژی Sass استفاده شده است و در نهایت کد‌های Sass به CSS تبدیل می‌شوند.



**پایگاه داده – Database :**

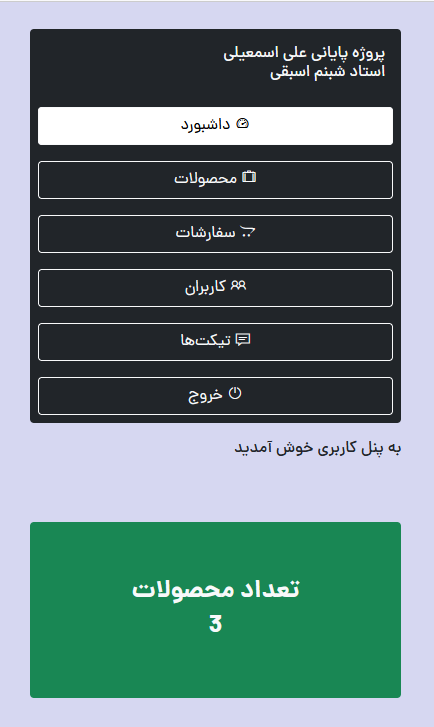
از زبان SQL استفاده شده است و از نرم افزار مدیریت پایگاه داده MySQL به عنوان DBMS استفاده شده است.



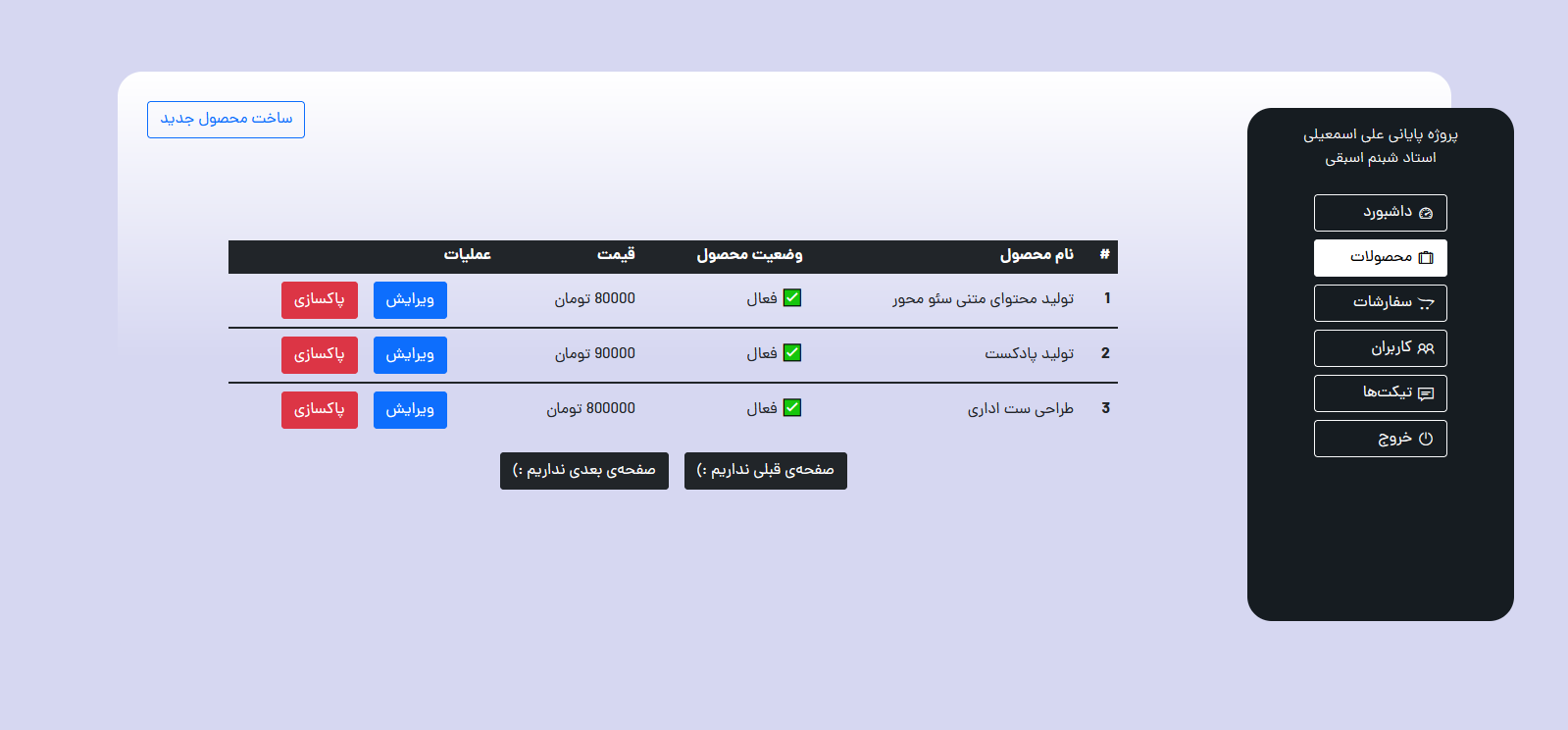
**رابط کاربری:**

تلاش کردم این پنل کاربری را به صورت اصولی طراحی کنم و تمامی تلاشم را کردم که علاوه بر زیبایی برای تمامی پلتفرم ها نیز مناسب باشد.

**رابط کاربری داشبورد مدیریت:**

****

**رابط کاربری بخش محصولات:**

****

**رابط کاربری بخش کاربران:**

****

**سمت سرور:**

برای مشاهده کد‌های کامل پروژه می‌توانید به رپوزیتوری گیت‌هاب پروژه که به صورت عمومی در دسترس است مراجعه کنید:

<https://github.com/realxoman/bachelor-project>

همانطور که گفته شد، برای پیاده سازی این پروژه از زبان برنامه نویسی پایتون و وب فریمورک Django استفاده شده است.

به سرعت‌بخشی و همچنین راحت‌تر کردن درک کد‌ها برای توسعه‌هایی که شاید در آینده روی این پروژه داده شود، از سیستم شیء گرای فریمورک Django یا به اصطلاح ‌Class Base View استفاده شده است.

سیستم کدنویسی مدل‌ها در جنگو به‌جای MVC یک مدل جایگزین به نام MVT هستش. یعنی به جای Model View Control از سیستم Model View Template استفاده می‌کند. برای مطالعه کد‌های رابط کاربری شما می‌توانید به آدرس paneluser/templates مراجعه کنید و کد‌های استاتیک پروژه را مشاهده کنید. برای مطالعه کد‌های مخصوص به URL های پروژه می‌توانید به آدرس paneluser/urls.py مراجعه کنید و آن فایل را مطالعه کنید. کد‌های View پروژه در آدرس paneluser/views.py موجود است.

بخش چالش بر‌انگیز این پروژه، بخش سیستم پشتیبانی یا همان تیکت‌ها بود که نمونه کد بروزرسانی آن را در زیر آورده‌ام.

class UserUpdateTicketView(LoginRequiredMixin,SuccessMessageMixin,View):

    template\_name="paneluser/ticket\_update.html"

    form\_class = TicketForm

    def get(self,request,ticketid):

        ticketid = ticket.objects.get(id=ticketid)

        ticketpms = ticketpm.objects.filter(ticketid=ticketid)

        form = self.form\_class

        context = {'form':form,'ticketpms':ticketpms,'ticketid':ticketid}

        return render(request,self.template\_name,context=context)

    def post(self,request,ticketid):

        ticketid = ticket.objects.get(id=ticketid)

        ticketpms = ticketpm.objects.filter(ticketid=ticketid)

        form = self.form\_class(request.POST, request.FILES)

        if form.is\_valid():

            desc = request.POST.get('text')

            attach = request.FILES['attachment']

            ticketpm1 = ticketpm(user=request.user,ticketid=ticketid,text=desc,attachment=attach)

            ticketid.status = "customer\_answer"

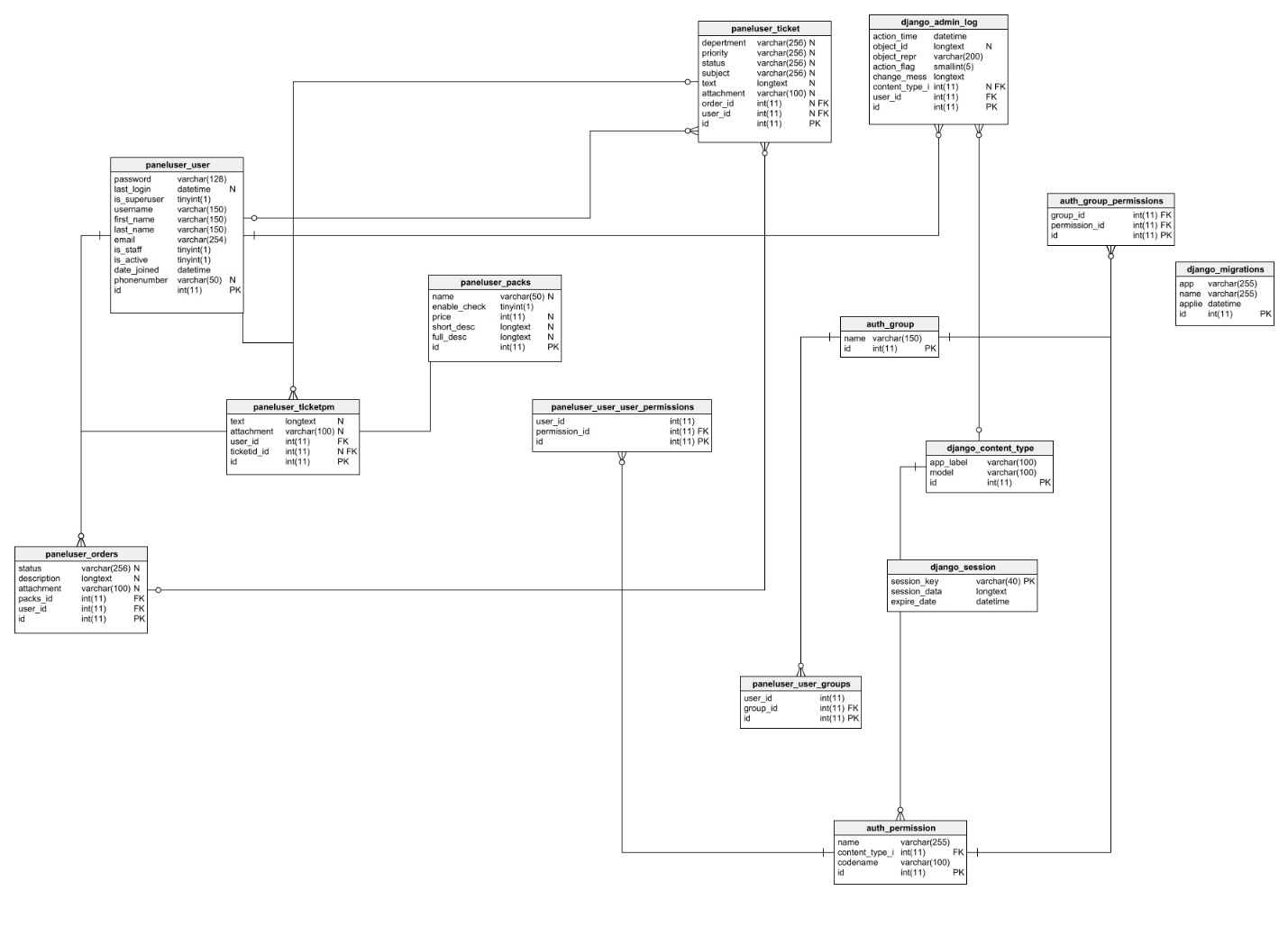
            ticketid.save()

            ticketpm1.save()

        context = {'form':form,'ticketpms':ticketpms,'ticketid':ticketid}

        return render(request,self.template\_name,context=context)

**ساختار پایگاه داده:**



شما می‌توانید ساختار دیتابیس را در لینک زیر با قابلیت زوم مشاهده کنید:

<https://my.vertabelo.com/public-model-view/UVtEVZB98axL9TjNtI48jpdCl2DDeHhpSKVvc1N2uR4AabXdtIzJX788wSoIVD00?x=-361&y=1521&zoom=0.2819>

در ادامه مدل دیتابیس را به صورت کامل شرح می‌دهم:

# 1. Model details

**Model name:**database-ali

**Version:**2.4

**Database engine:**MySQL

# 2. Tables

## 2.1. Table auth\_group

2.1.1. Columns

|  |  |  |  |
| --- | --- | --- | --- |
| **Column name** | **Type** | **Properties** | **Description** |
| name | varchar(150) |  |  |
| id | int(11) | PK |  |

2.1.2. Alternate keys

|  |  |  |
| --- | --- | --- |
| **Key name** | **Columns** | **Description** |
| name | name |  |

## 2.2. Table auth\_group\_permissions

2.2.1. Columns

|  |  |  |  |
| --- | --- | --- | --- |
| **Column name** | **Type** | **Properties** | **Description** |
| group\_id | int(11) |  |  |
| permission\_id | int(11) |  |  |
| id | int(11) | PK |  |

2.2.2. Alternate keys

|  |  |  |
| --- | --- | --- |
| **Key name** | **Columns** | **Description** |
| auth\_group\_permissions\_group\_id\_permission\_id\_0cd325b0\_uniq | group\_id, permission\_id |  |

2.2.3. Indexes

|  |  |  |
| --- | --- | --- |
| **Index name** | **Columns** | **Description** |
| auth\_group\_permissio\_permission\_id\_84c5c92e\_fk\_auth\_perm | permission\_id (ASC) |  |

## 2.3. Table auth\_permission

2.3.1. Columns

|  |  |  |  |
| --- | --- | --- | --- |
| **Column name** | **Type** | **Properties** | **Description** |
| name | varchar(255) |  |  |
| content\_type\_id | int(11) |  |  |
| codename | varchar(100) |  |  |
| id | int(11) | PK |  |

2.3.2. Alternate keys

|  |  |  |
| --- | --- | --- |
| **Key name** | **Columns** | **Description** |
| auth\_permission\_content\_type\_id\_codename\_01ab375a\_uniq | content\_type\_id, codename |  |

## 2.4. Table django\_admin\_log

2.4.1. Columns

|  |  |  |  |
| --- | --- | --- | --- |
| **Column name** | **Type** | **Properties** | **Description** |
| action\_time | datetime |  |  |
| object\_id | longtext | null |  |
| object\_repr | varchar(200) |  |  |
| action\_flag | smallint(5) |  |  |
| change\_message | longtext |  |  |
| content\_type\_id | int(11) | null |  |
| user\_id | int(11) |  |  |
| id | int(11) | PK |  |

2.4.2. Indexes

|  |  |  |
| --- | --- | --- |
| **Index name** | **Columns** | **Description** |
| django\_admin\_log\_content\_type\_id\_c4bce8eb\_fk\_django\_co | content\_type\_id (ASC) |  |
| django\_admin\_log\_user\_id\_c564eba6\_fk | user\_id (ASC) |  |

## 2.5. Table django\_content\_type

2.5.1. Columns

|  |  |  |  |
| --- | --- | --- | --- |
| **Column name** | **Type** | **Properties** | **Description** |
| app\_label | varchar(100) |  |  |
| model | varchar(100) |  |  |
| id | int(11) | PK |  |

2.5.2. Alternate keys

|  |  |  |
| --- | --- | --- |
| **Key name** | **Columns** | **Description** |
| django\_content\_type\_app\_label\_model\_76bd3d3b\_uniq | app\_label, model |  |

## 2.6. Table django\_migrations

2.6.1. Columns

|  |  |  |  |
| --- | --- | --- | --- |
| **Column name** | **Type** | **Properties** | **Description** |
| app | varchar(255) |  |  |
| name | varchar(255) |  |  |
| applied | datetime |  |  |
| id | int(11) | PK |  |

## 2.7. Table django\_session

2.7.1. Columns

|  |  |  |  |
| --- | --- | --- | --- |
| **Column name** | **Type** | **Properties** | **Description** |
| session\_key | varchar(40) | PK |  |
| session\_data | longtext |  |  |
| expire\_date | datetime |  |  |

2.7.2. Indexes

|  |  |  |
| --- | --- | --- |
| **Index name** | **Columns** | **Description** |
| django\_session\_expire\_date\_a5c62663 | expire\_date (ASC) |  |

## 2.8. Table paneluser\_orders

2.8.1. Columns

|  |  |  |  |
| --- | --- | --- | --- |
| **Column name** | **Type** | **Properties** | **Description** |
| status | varchar(256) | null |  |
| description | longtext | null |  |
| attachment | varchar(100) | null |  |
| packs\_id | int(11) |  |  |
| user\_id | int(11) |  |  |
| id | int(11) | PK |  |

2.8.2. Indexes

|  |  |  |
| --- | --- | --- |
| **Index name** | **Columns** | **Description** |
| paneluser\_orders\_packs\_id\_c671b3c9\_fk | packs\_id (ASC) |  |
| paneluser\_orders\_user\_id\_eb06b92e\_fk | user\_id (ASC) |  |

## 2.9. Table paneluser\_packs

2.9.1. Columns

|  |  |  |  |
| --- | --- | --- | --- |
| **Column name** | **Type** | **Properties** | **Description** |
| name | varchar(50) | null |  |
| enable\_check | tinyint(1) |  |  |
| price | int(11) | null |  |
| short\_desc | longtext | null |  |
| full\_desc | longtext | null |  |
| id | int(11) | PK |  |

## 2.10. Table paneluser\_ticket

2.10.1. Columns

|  |  |  |  |
| --- | --- | --- | --- |
| **Column name** | **Type** | **Properties** | **Description** |
| depertment | varchar(256) | null |  |
| priority | varchar(256) | null |  |
| status | varchar(256) | null |  |
| subject | varchar(256) | null |  |
| text | longtext | null |  |
| attachment | varchar(100) | null |  |
| order\_id | int(11) | null |  |
| user\_id | int(11) | null |  |
| id | int(11) | PK |  |

2.10.2. Indexes

|  |  |  |
| --- | --- | --- |
| **Index name** | **Columns** | **Description** |
| paneluser\_ticket\_order\_id\_75bfbbec\_fk | order\_id (ASC) |  |
| paneluser\_ticket\_user\_id\_96555a29\_fk | user\_id (ASC) |  |

## 2.11. Table paneluser\_ticketpm

2.11.1. Columns

|  |  |  |  |
| --- | --- | --- | --- |
| **Column name** | **Type** | **Properties** | **Description** |
| text | longtext | null |  |
| attachment | varchar(100) | null |  |
| user\_id | int(11) |  |  |
| ticketid\_id | int(11) | null |  |
| id | int(11) | PK |  |

2.11.2. Indexes

|  |  |  |
| --- | --- | --- |
| **Index name** | **Columns** | **Description** |
| paneluser\_ticketpm\_ticketid\_id\_dd452fed\_fk | ticketid\_id (ASC) |  |
| paneluser\_ticketpm\_user\_id\_8ef85a7a\_fk | user\_id (ASC) |  |

## 2.12. Table paneluser\_user

2.12.1. Columns

|  |  |  |  |
| --- | --- | --- | --- |
| **Column name** | **Type** | **Properties** | **Description** |
| password | varchar(128) |  |  |
| last\_login | datetime | null |  |
| is\_superuser | tinyint(1) |  |  |
| username | varchar(150) |  |  |
| first\_name | varchar(150) |  |  |
| last\_name | varchar(150) |  |  |
| email | varchar(254) |  |  |
| is\_staff | tinyint(1) |  |  |
| is\_active | tinyint(1) |  |  |
| date\_joined | datetime |  |  |
| phonenumber | varchar(50) | null |  |
| id | int(11) | PK |  |

2.12.2. Alternate keys

|  |  |  |
| --- | --- | --- |
| **Key name** | **Columns** | **Description** |
| username | username |  |

## 2.13. Table paneluser\_user\_groups

2.13.1. Columns

|  |  |  |  |
| --- | --- | --- | --- |
| **Column name** | **Type** | **Properties** | **Description** |
| user\_id | int(11) |  |  |
| group\_id | int(11) |  |  |
| id | int(11) | PK |  |

2.13.2. Alternate keys

|  |  |  |
| --- | --- | --- |
| **Key name** | **Columns** | **Description** |
| paneluser\_user\_groups\_user\_id\_group\_id\_db7a92ca\_uniq | user\_id, group\_id |  |

2.13.3. Indexes

|  |  |  |
| --- | --- | --- |
| **Index name** | **Columns** | **Description** |
| paneluser\_user\_groups\_group\_id\_eeea64cf\_fk\_auth\_group\_id | group\_id (ASC) |  |

## 2.14. Table paneluser\_user\_user\_permissions

2.14.1. Columns

|  |  |  |  |
| --- | --- | --- | --- |
| **Column name** | **Type** | **Properties** | **Description** |
| user\_id | int(11) |  |  |
| permission\_id | int(11) |  |  |
| id | int(11) | PK |  |

2.14.2. Alternate keys

|  |  |  |
| --- | --- | --- |
| **Key name** | **Columns** | **Description** |
| paneluser\_user\_user\_perm\_user\_id\_permission\_id\_2eda581a\_uniq | user\_id, permission\_id |  |

2.14.3. Indexes

|  |  |  |
| --- | --- | --- |
| **Index name** | **Columns** | **Description** |
| paneluser\_user\_user\_\_permission\_id\_fc81276c\_fk\_auth\_perm | permission\_id (ASC) |  |

**جمع‌بندی:**

این پروژه به عنوان اولین پروژه طراحی پنل کاربری من بسیار برام مفید بود و چالش بسیار زیادی که برای من داشت پیاده‌سازی بخش تیکت‌های این پنل کاربری بود که نوشتن الگوریتم و بروزرسانی کردن آن کمی پیچیدگی داشت.

**زمان شروع پیاده سازی این پروژه:** ۲۱/۰۶/۱۴۰۰

**تاریخ تکمیل این سند:** ۰۲/۰۹/۱۴۰۰

**میزان تسلط به پروژه در ابتدای راه:** ۳۰٪

**زمان برای یادگیری:** ۳ ساعت در هفته مازاد کلاس‌های دانشگاه

**لینک‌های مفیدی که از آن‌ها برای پیاده سازی این پروژه استفاده کردم:**

<https://www.w3schools.com>

<https://www.python.org>

<https://docs.djangoproject.com/en/3.2/>

<https://stackoverflow.com/>

<https://medium.com/>

با تشکر از شما برای مطالعه این فایل توضیحات پروژه

**پایان – آذر ۱۴۰۰**