

**DJANGO PROJECT**

**ONLINE INVENTORY  
MANAGEMENT SYSTEM**

**TASK: DATABASE AND ADMIN INTERFACE**

**DATE: 23 FEBRUARY 2023**

**CREATED BY: AMIL DEV S**

# DATABASE

- Models.py

```
from django.db import models

# Created the category choice.
category_choice=(
    ('Food','Food'),
    ('Clothes','Clothes'),
    ('Electronics','Electronics'),
    ('Home','Home'),
    ('Grocery','Grocery'),
)

# Created the Category model.
class Category(models.Model):
    name=models.CharField(max_length=50,blank=True,null=True)

    def __str__(self):
        return self.name

# Created the Stock Model.
class Stock(models.Model):
    category=models.ForeignKey(Category, on_delete=models.CASCADE, blank=True)
    item_name=models.CharField(max_length=50,blank=True,null=True)
    quantity=models.IntegerField(default='0',blank=False,null=True)
    receive_quantity=models.IntegerField(default='0',blank=True,null=True)
    receive_by=models.CharField(max_length=50,blank=True,null=True)
    issue_quantity=models.IntegerField(default='0',blank=True,null=True)
    issue_by=models.CharField(max_length=50,blank=True,null=True)
    issue_to=models.CharField(max_length=50,blank=True,null=True)
    phone_number=models.CharField(max_length=50,blank=True,null=True)
    created_by=models.CharField(max_length=50,blank=True,null=True)
    reorder_level=models.IntegerField(default='0',blank=True,null=True)
    last_updated=models.DateTimeField(auto_now_add=False,auto_now=True)
    timestamp=models.DateTimeField(auto_now_add=True,auto_now=False)
    #date=models.DateTimeField(auto_now_add=False,auto_now=False)

    def __str__(self):
        return self.item_name + " " +(str(self.quantity))
```

# Created the Stock History Model

```
class StockHistory(models.Model):
```

```
    category=models.ForeignKey(Category, on_delete=models.CASCADE,  
blank=True,null=True)
```

```
    item_name=models.CharField(max_length=50,blank=True,null=True)
```

```
    quantity=models.IntegerField(default='0',blank=True,null=True)
```

```
    receive_quantity=models.IntegerField(default='0',blank=True,null=True)
```

```
    receive_by=models.CharField(max_length=50,blank=True,null=True)
```

```
    issue_quantity=models.IntegerField(default='0',blank=True,null=True)
```

```
    issue_by=models.CharField(max_length=50,blank=True,null=True)
```

```
    issue_to=models.CharField(max_length=50,blank=True,null=True)
```

```
    phone_number=models.CharField(max_length=50,blank=True,null=True)
```

```
    created_by=models.CharField(max_length=50,blank=True,null=True)
```

```
    reorder_level=models.IntegerField(default='0',blank=True,null=True)
```

```
    last_updated=models.DateTimeField(auto_now_add=False,auto_now=False, null=True)
```

```
    timestamp=models.DateTimeField(auto_now_add=False,auto_now=False,null=True)
```

```
    def __str__(self):
```

```
        return self.item_name + " " +(str(self.quantity))
```

# Created the Client Model

```
class Client(models.Model):
```

```
    place=models.CharField(max_length=50)
```

```
    description=models.CharField(max_length=100)
```

```
    def __str__(self):
```

```
        return self.place
```

```
class Transaction(models.Model):
```

```
    trans_id=models.AutoField(primary_key=True)
```

```
    quantity=models.IntegerField()
```

```
    time=models.DateTimeField(auto_now=True)
```

```
    item=models.ForeignKey(Stock,on_delete=models.CASCADE,blank=False,null=False)
```

```
    client=models.ForeignKey(Client,on_delete=models.CASCADE,blank=False,null=False)
```

```
    def __str__(self):
```

```
        return ("%s , %s , %s" % (self.item.name,self.client.place,self.time))
```

- **Forms.py**

```
from django import forms
from .models import *

class StockCreateForm(forms.ModelForm):
    class Meta:
        model = Stock
        fields=['category','item_name','quantity','issue_by']

    def clean_category(self):
        category = self.cleaned_data.get('category')
        if not category:
            raise forms.ValidationError('Please select a category')
        for instance in Stock.objects.all():
            if instance.category == category:
                raise forms.ValidationError('Category already exists')
        return category

    def clean_item_name(self):
        item_name = self.cleaned_data.get('item_name')
        if item_name:
            return item_name
        else:
            raise forms.ValidationError('Please enter an item name')

class StockSearchForm(forms.ModelForm):
    export_to_CSV = forms.BooleanField(required=False)
    start_date=forms.DateTimeField(required=False)
    end_date=forms.DateTimeField(required=False)
    class Meta:
        model = StockHistory
        fields=['category','item_name','start_date','end_date']

class StockUpdateForm(forms.ModelForm):

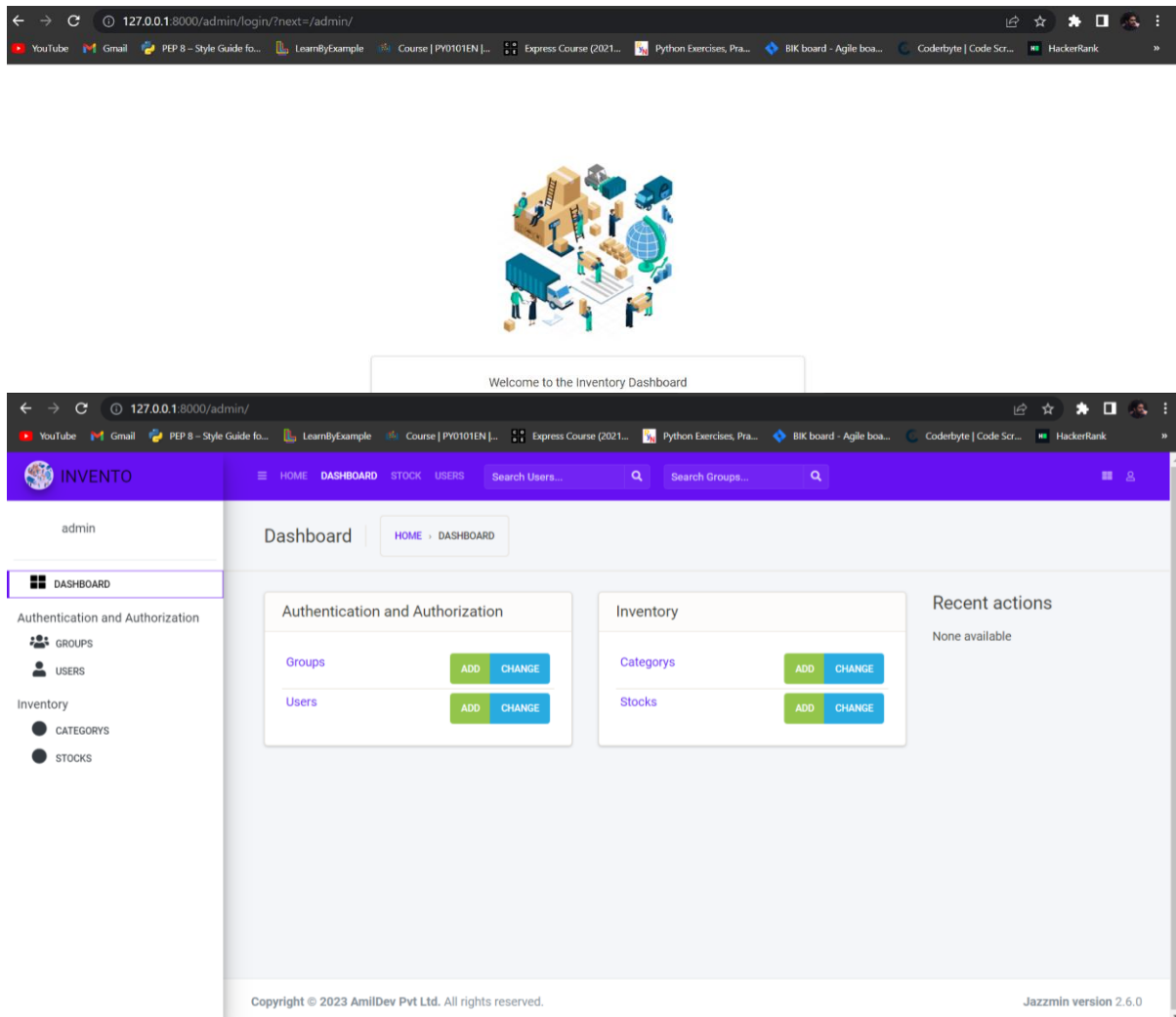
    class Meta:
        model = Stock
        fields=['category','item_name','quantity']

class IssueForm(forms.ModelForm):
```

```
class Meta:
    model = Stock
    fields=['issue_quantity','issue_to']

class ReceiveForm(forms.ModelForm):
    class Meta:
        model = Stock
        fields=['receive_quantity','receive_by']
```

## ADMIN INTERFACE



127.0.0.1:8000/admin/inventory/category/

INVENTO

admin

DASHBOARD

Authentication and Authorization

GROUPS

USERS

Inventory

CATEGORYS

STOCKS

Categories

HOME • INVENTORY • CATEGORYS

ADD CATEGORY

✓ The category "Smartphones" was added successfully.

category 0 of 2 selected

Category

Smartphones

Laptops

2 categorys

Copyright © 2023 AmilDev Pvt Ltd. All rights reserved.

Jazzmin version 2.6.0

127.0.0.1:8000/admin/inventory/stock/

INVENTO

admin

DASHBOARD

Authentication and Authorization

GROUPS

USERS

Inventory

CATEGORYS

STOCKS

Stocks

HOME • INVENTORY • STOCKS

ADD STOCK

✓ The stock "HP Laptops 10" was added successfully.

category SEARCH

0 of 1 selected

Category	Item name	Quantity	Issue by
Laptops	HP Laptops	10	Amil Dev

1 stock

Copyright © 2023 AmilDev Pvt Ltd. All rights reserved.

Jazzmin version 2.6.0

# CLASS DIAGRAM

