



Django File Manager with Role-Based Access Control

I'll guide you through creating a comprehensive file manager project with role-based access control, OTP authentication, and admin controls.

Project Overview

Project Name: `secure_file_manager`

This Django application will include:

- Role-based access control (Admin/User roles)
- OTP-based authentication for login/signup
- File upload, CRUD operations, and favorites
- Admin panel for user and file management
- Bootstrap frontend with responsive design

Step 1: Project Setup and Structure

Create Django Project

```
# Create virtual environment
python -m venv venv
source venv/bin/activate # On Windows: venv\Scripts\activate

# Install required packages
pip install django mysqlclient pillow django-otp qrcode[pil] twilio

# Create project
django-admin startproject secure_file_manager
cd secure_file_manager

# Create apps
python manage.py startapp accounts
python manage.py startapp filemanager
python manage.py startapp dashboard
```

Project Structure

```
secure_file_manager/
├── manage.py
├── secure_file_manager/
│   ├── __init__.py
│   ├── settings.py
│   ├── urls.py
│   ├── wsgi.py
│   └── asgi.py
├── accounts/
│   ├── __init__.py
│   ├── admin.py
│   ├── apps.py
│   ├── models.py
│   ├── views.py
│   ├── urls.py
│   ├── forms.py
│   └── migrations/
├── filemanager/
│   ├── __init__.py
│   ├── admin.py
│   ├── apps.py
│   ├── models.py
│   ├── views.py
│   ├── urls.py
│   ├── forms.py
│   └── migrations/
├── dashboard/
│   ├── __init__.py
│   ├── admin.py
│   ├── apps.py
│   ├── models.py
│   ├── views.py
│   ├── urls.py
│   └── migrations/
├── templates/
│   ├── base.html
│   ├── accounts/
│   ├── filemanager/
│   └── dashboard/
├── static/
│   ├── css/
│   ├── js/
│   └── images/
├── media/
└── uploads/
```

Step 2: Database Configuration

MySQL Database Setup

SQL Commands to Create Database:

```
-- Create database
CREATE DATABASE secure_file_manager;

-- Create user (optional)
CREATE USER 'filemanager_user'@'localhost' IDENTIFIED BY 'your_password';
GRANT ALL PRIVILEGES ON secure_file_manager.* TO 'filemanager_user'@'localhost';
FLUSH PRIVILEGES;
```

Settings Configuration

File: secure_file_manager/settings.py

```
import os
from pathlib import Path

BASE_DIR = Path(__file__).resolve().parent.parent

SECRET_KEY = 'your-secret-key-here'
DEBUG = True
ALLOWED_HOSTS = ['localhost', '127.0.0.1']

# Application definition
INSTALLED_APPS = [
    'django.contrib.admin',
    'django.contrib.auth',
    'django.contrib.contenttypes',
    'django.contrib.sessions',
    'django.contrib.messages',
    'django.contrib.staticfiles',
    'django_otp',
    'django_otp.plugins.otp_totp',
    'django_otp.plugins.otp_static',
    'accounts',
    'filemanager',
    'dashboard',
]

MIDDLEWARE = [
    'django.middleware.security.SecurityMiddleware',
    'django.contrib.sessions.middleware.SessionMiddleware',
    'django.middleware.common.CommonMiddleware',
    'django.middleware.csrf.CsrfViewMiddleware',
    'django.contrib.auth.middleware.AuthenticationMiddleware',
    'django_otp.middleware.OTPMiddleware',
    'django.contrib.messages.middleware.MessageMiddleware',
    'django.middleware.clickjacking.XFrameOptionsMiddleware',
]
```

```

ROOT_URLCONF = 'secure_file_manager.urls'

TEMPLATES = [
    {
        'BACKEND': 'django.template.backends.django.DjangoTemplates',
        'DIRS': [BASE_DIR / 'templates'],
        'APP_DIRS': True,
        'OPTIONS': {
            'context_processors': [
                'django.template.context_processors.debug',
                'django.template.context_processors.request',
                'django.contrib.auth.context_processors.auth',
                'django.contrib.messages.context_processors.messages',
            ],
        },
    ],
]

# Database
DATABASES = {
    'default': {
        'ENGINE': 'django.db.backends.mysql',
        'NAME': 'secure_file_manager',
        'USER': 'filemanager_user',
        'PASSWORD': 'your_password',
        'HOST': 'localhost',
        'PORT': '3306',
        'OPTIONS': {
            'init_command': "SET sql_mode='STRICT_TRANS_TABLES'",
        },
    }
}

# Password validation
AUTH_PASSWORD_VALIDATORS = [
    {
        'NAME': 'django.contrib.auth.password_validation.UserAttributeSimilarityValidator',
    },
    {
        'NAME': 'django.contrib.auth.password_validation.MinimumLengthValidator',
    },
    {
        'NAME': 'django.contrib.auth.password_validation.CommonPasswordValidator',
    },
    {
        'NAME': 'django.contrib.auth.password_validation.NumericPasswordValidator',
    },
]

# Internationalization
LANGUAGE_CODE = 'en-us'
TIME_ZONE = 'UTC'
USE_I18N = True
USE_TZ = True

```

```

# Static files
STATIC_URL = '/static/'
STATICFILES_DIRS = [BASE_DIR / 'static']
STATIC_ROOT = BASE_DIR / 'staticfiles'

# Media files
MEDIA_URL = '/media/'
MEDIA_ROOT = BASE_DIR / 'media'

# Default primary key field type
DEFAULT_AUTO_FIELD = 'django.db.models.BigAutoField'

# Email configuration (for OTP)
EMAIL_BACKEND = 'django.core.mail.backends.smtp.EmailBackend'
EMAIL_HOST = 'smtp.gmail.com'
EMAIL_PORT = 587
EMAIL_USE_TLS = True
EMAIL_HOST_USER = 'your-email@gmail.com'
EMAIL_HOST_PASSWORD = 'your-app-password'

# OTP Settings
OTP_TOTP_ISSUER = 'Secure File Manager'
OTP_LOGIN_URL = '/accounts/login/'

# Custom user model
AUTH_USER_MODEL = 'accounts.CustomUser'

# File upload settings
FILE_UPLOAD_MAX_MEMORY_SIZE = 26214400 # 25MB
DATA_UPLOAD_MAX_MEMORY_SIZE = 26214400 # 25MB

```

Step 3: Models Implementation

Accounts Models

File: accounts/models.py

```

from django.contrib.auth.models import AbstractUser
from django.db import models
from django.core.validators import RegexValidator

class CustomUser(AbstractUser):
    ROLE_CHOICES = [
        ('admin', 'Admin'),
        ('user', 'User'),
    ]

    email = models.EmailField(unique=True)
    phone_regex = RegexValidator(regex=r'^\+?1?\d{9,15}$',
                                message="Phone number must be entered in format: '+999999999'")
    phone_number = models.CharField(validators=[phone_regex], max_length=17, blank=True)
    role = models.CharField(max_length=10, choices=ROLE_CHOICES, default='user')
    is_verified = models.BooleanField(default=False)
    created_at = models.DateTimeField(auto_now_add=True)

```

```

updated_at = models.DateTimeField(auto_now=True)
profile_picture = models.ImageField(upload_to='profiles/', null=True, blank=True)

USERNAME_FIELD = 'email'
REQUIRED_FIELDS = ['username', 'first_name', 'last_name']

def __str__(self):
    return self.email

@property
def is_admin(self):
    return self.role == 'admin'

class OTPVerification(models.Model):
    user = models.ForeignKey(CustomUser, on_delete=models.CASCADE)
    otp_code = models.CharField(max_length=6)
    purpose = models.CharField(max_length=20, choices=[
        ('signup', 'Sign Up'),
        ('login', 'Login'),
        ('reset', 'Password Reset')
    ])
    created_at = models.DateTimeField(auto_now_add=True)
    is_used = models.BooleanField(default=False)
    expires_at = models.DateTimeField()

    def __str__(self):
        return f"OTP for {self.user.email} - {self.purpose}"

    @property
    def is_expired(self):
        from django.utils import timezone
        return timezone.now() > self.expires_at

```

File Manager Models

File: filemanager/models.py

```

import os
from django.db import models
from django.contrib.auth import get_user_model
from django.core.validators import FileExtensionValidator

User = get_user_model()

def user_directory_path(instance, filename):
    return f'uploads/user_{instance.owner.id}/{filename}'

class FileCategory(models.Model):
    name = models.CharField(max_length=100, unique=True)
    description = models.TextField(blank=True)
    icon = models.CharField(max_length=50, default='fa-folder')
    created_at = models.DateTimeField(auto_now_add=True)

    def __str__(self):
        return self.name

```

```

class Meta:
    verbose_name_plural = "File Categories"

class UserFile(models.Model):
    VISIBILITY_CHOICES = [
        ('private', 'Private'),
        ('public', 'Public'),
        ('restricted', 'Restricted'),
    ]

    name = models.CharField(max_length=255)
    description = models.TextField(blank=True)
    file = models.FileField(upload_to=user_directory_path,
                           validators=[FileExtensionValidator(
                               allowed_extensions=[
                                   'pdf', 'doc', 'docx', 'txt', 'jpg',
                                   'jpeg', 'png', 'gif', 'mp4', 'avi', 'mov',
                                   'zip', 'rar', 'xlsx', 'pptx'])])
    owner = models.ForeignKey(User, on_delete=models.CASCADE, related_name='owned_files')
    category = models.ForeignKey(FileCategory, on_delete=models.SET_NULL,
                                null=True, blank=True)
    visibility = models.CharField(max_length=10, choices=VISIBILITY_CHOICES,
                                default='private')
    file_size = models.BigIntegerField()
    file_type = models.CharField(max_length=100)
    is_favorite = models.BooleanField(default=False)
    download_count = models.PositiveIntegerField(default=0)
    created_at = models.DateTimeField(auto_now_add=True)
    updated_at = models.DateTimeField(auto_now=True)

    def save(self, *args, **kwargs):
        if self.file:
            self.file_size = self.file.size
            self.file_type = os.path.splitext(self.file.name)[1].lower()
            super().save(*args, **kwargs)

    def __str__(self):
        return self.name

    @property
    def formatted_size(self):
        size = self.file_size
        for unit in ['B', 'KB', 'MB', 'GB']:
            if size < 1024.0:
                return f"{size:.1f} {unit}"
            size /= 1024.0
        return f"{size:.1f} TB"

    class Meta:
        ordering = ['-created_at']

class FileAccess(models.Model):
    PERMISSION_CHOICES = [
        ('view', 'View Only'),
        ('download', 'View & Download'),
        ('edit', 'View, Download & Edit'),
    ]

```

```

]

file = models.ForeignKey(UserFile, on_delete=models.CASCADE, related_name='access_permissions')
user = models.ForeignKey(User, on_delete=models.CASCADE, related_name='file_permissions')
permission = models.CharField(max_length=10, choices=PERMISSION_CHOICES, default='view')
granted_by = models.ForeignKey(User, on_delete=models.CASCADE, related_name='granted_permissions')
granted_at = models.DateTimeField(auto_now_add=True)
expires_at = models.DateTimeField(null=True, blank=True)

def __str__(self):
    return f"{self.user.email} - {self.file.name} ({self.permission})"

class Meta:
    unique_together = ['file', 'user']

class FileActivity(models.Model):
    ACTION_CHOICES = [
        ('upload', 'Uploaded'),
        ('view', 'Viewed'),
        ('download', 'Downloaded'),
        ('update', 'Updated'),
        ('delete', 'Deleted'),
        ('share', 'Shared'),
    ]

    file = models.ForeignKey(UserFile, on_delete=models.CASCADE, related_name='activities')
    user = models.ForeignKey(User, on_delete=models.CASCADE)
    action = models.CharField(max_length=10, choices=ACTION_CHOICES)
    timestamp = models.DateTimeField(auto_now_add=True)
    ip_address = models.GenericIPAddressField(null=True, blank=True)

    def __str__(self):
        return f"{self.user.email} {self.action} {self.file.name}"

    class Meta:
        ordering = ['-timestamp']

```

Dashboard Models

File: dashboard/models.py

```

from django.db import models
from django.contrib.auth import get_user_model

User = get_user_model()

class SystemSettings(models.Model):
    key = models.CharField(max_length=100, unique=True)
    value = models.TextField()
    description = models.TextField(blank=True)
    updated_by = models.ForeignKey(User, on_delete=models.CASCADE)
    updated_at = models.DateTimeField(auto_now=True)

    def __str__(self):
        return self.key

```



```

class Meta:
    verbose_name_plural = "System Settings"

class UserActivity(models.Model):
    user = models.ForeignKey(User, on_delete=models.CASCADE, related_name='activities')
    action = models.CharField(max_length=100)
    description = models.TextField()
    timestamp = models.DateTimeField(auto_now_add=True)
    ip_address = models.GenericIPAddressField(null=True, blank=True)

    def __str__(self):
        return f"{self.user.email} - {self.action}"

    class Meta:
        ordering = ['-timestamp']

```

Step 4: Database Migration

Create and apply migrations:

```

# Create migrations
python manage.py makemigrations accounts
python manage.py makemigrations filemanager
python manage.py makemigrations dashboard
python manage.py migrate

# Create superuser
python manage.py createsuperuser

```

Step 5: Forms Implementation

Accounts Forms

File: accounts/forms.py

```

from django import forms
from django.contrib.auth.forms import UserCreationForm, AuthenticationForm
from django.contrib.auth import authenticate
from .models import CustomUser, OTPVerification

class CustomUserCreationForm(UserCreationForm):
    email = forms.EmailField(
        widget=forms.EmailInput(attrs={'class': 'form-control', 'placeholder': 'Email'})
    )
    first_name = forms.CharField(
        max_length=30,
        widget=forms.TextInput(attrs={'class': 'form-control', 'placeholder': 'First Name'})
    )
    last_name = forms.CharField(
        max_length=30,
        widget=forms.TextInput(attrs={'class': 'form-control', 'placeholder': 'Last Name'})
    )

```

```

    )
    phone_number = forms.CharField(
        max_length=17,
        required=False,
        widget=forms.TextInput(attrs={'class': 'form-control', 'placeholder': 'Phone Number'})
    )

    class Meta:
        model = CustomUser
        fields = ('username', 'email', 'first_name', 'last_name', 'phone_number', 'password1', 'password2')

    def __init__(self, *args, **kwargs):
        super().__init__(*args, **kwargs)
        self.fields['username'].widget.attrs.update({'class': 'form-control', 'placeholder': 'Username'})
        self.fields['password1'].widget.attrs.update({'class': 'form-control', 'placeholder': 'Password'})
        self.fields['password2'].widget.attrs.update({'class': 'form-control', 'placeholder': 'Confirm Password'})

class CustomLoginForm(forms.Form):
    email = forms.EmailField(
        widget=forms.EmailInput(attrs={'class': 'form-control', 'placeholder': 'Email'})
    )
    password = forms.CharField(
        widget=forms.PasswordInput(attrs={'class': 'form-control', 'placeholder': 'Password'})
    )

    def clean(self):
        email = self.cleaned_data.get('email')
        password = self.cleaned_data.get('password')

        if email and password:
            self.user = authenticate(username=email, password=password)
            if not self.user:
                raise forms.ValidationError("Invalid email or password.")
            if not self.user.is_active:
                raise forms.ValidationError("Account is disabled.")
        return self.cleaned_data

class OTPVerificationForm(forms.Form):
    otp_code = forms.CharField(
        max_length=6,
        min_length=6,
        widget=forms.TextInput(attrs={
            'class': 'form-control text-center',
            'placeholder': '000000',
            'autocomplete': 'off'
        })
    )

    def __init__(self, user=None, purpose='login', *args, **kwargs):
        self.user = user
        self.purpose = purpose
        super().__init__(*args, **kwargs)

    def clean_otp_code(self):
        otp_code = self.cleaned_data.get('otp_code')
        if self.user and otp_code:

```

```

        try:
            otp = OTPVerification.objects.get(
                user=self.user,
                otp_code=otp_code,
                purpose=self.purpose,
                is_used=False
            )
            if otp.is_expired:
                raise forms.ValidationError("OTP has expired.")
        except OTPVerification.DoesNotExist:
            raise forms.ValidationError("Invalid OTP code.")
        return otp_code

class UserProfileForm(forms.ModelForm):
    class Meta:
        model = CustomUser
        fields = ['first_name', 'last_name', 'phone_number', 'profile_picture']
        widgets = {
            'first_name': forms.TextInput(attrs={'class': 'form-control'}),
            'last_name': forms.TextInput(attrs={'class': 'form-control'}),
            'phone_number': forms.TextInput(attrs={'class': 'form-control'}),
            'profile_picture': forms.FileInput(attrs={'class': 'form-control'}),
        }

```

File Manager Forms

File: filemanager/forms.py

```

from django import forms
from .models import UserFile, FileAccess, FileCategory
from django.contrib.auth import get_user_model

User = get_user_model()

class FileUploadForm(forms.ModelForm):
    class Meta:
        model = UserFile
        fields = ['name', 'description', 'file', 'category', 'visibility']
        widgets = {
            'name': forms.TextInput(attrs={'class': 'form-control', 'placeholder': 'File name'}),
            'description': forms.Textarea(attrs={'class': 'form-control', 'rows': 3, 'placeholder': 'Description'}),
            'file': forms.FileInput(attrs={'class': 'form-control'}),
            'category': forms.Select(attrs={'class': 'form-control'}),
            'visibility': forms.Select(attrs={'class': 'form-control'}),
        }

    def __init__(self, *args, **kwargs):
        super().__init__(*args, **kwargs)
        self.fields['category'].queryset = FileCategory.objects.all()
        self.fields['category'].empty_label = "Select Category (Optional)"

class FileEditForm(forms.ModelForm):
    class Meta:
        model = UserFile
        fields = ['name', 'description', 'category', 'visibility']

```

```

        widgets = {
            'name': forms.TextInput(attrs={'class': 'form-control'}),
            'description': forms.Textarea(attrs={'class': 'form-control', 'rows': 3}),
            'category': forms.Select(attrs={'class': 'form-control'}),
            'visibility': forms.Select(attrs={'class': 'form-control'}),
        }

class FileAccessForm(forms.ModelForm):
    user_email = forms.EmailField(
        widget=forms.EmailInput(attrs={'class': 'form-control', 'placeholder': 'User Email'})
    )

    class Meta:
        model = FileAccess
        fields = ['permission', 'expires_at']
        widgets = {
            'permission': forms.Select(attrs={'class': 'form-control'}),
            'expires_at': forms.DateTimeInput(attrs={'class': 'form-control', 'type': 'datetime-local'})
        }

    def clean_user_email(self):
        email = self.cleaned_data.get('user_email')
        try:
            user = User.objects.get(email=email, is_verified=True)
            return user
        except User.DoesNotExist:
            raise forms.ValidationError("User with this email does not exist or is not verified")

class FileSearchForm(forms.Form):
    query = forms.CharField(
        max_length=255,
        required=False,
        widget=forms.TextInput(attrs={
            'class': 'form-control',
            'placeholder': 'Search files...',
            'autocomplete': 'off'
        })
    )

    category = forms.ModelChoiceField(
        queryset=FileCategory.objects.all(),
        required=False,
        empty_label="All Categories",
        widget=forms.Select(attrs={'class': 'form-control'})
    )

    file_type = forms.ChoiceField(
        choices=[('', 'All Types'), ('.pdf', 'PDF'), ('.doc', 'Word'), ('.jpg', 'Image')],
        required=False,
        widget=forms.Select(attrs={'class': 'form-control'})
    )

```

Step 6: Views Implementation

Accounts Views

File: accounts/views.py

```
import random
import string
from datetime import timedelta
from django.shortcuts import render, redirect
from django.contrib.auth import login, logout
from django.contrib import messages
from django.contrib.auth.decorators import login_required
from django.views.decorators.http import require_http_methods
from django.utils import timezone
from django.core.mail import send_mail
from django.conf import settings
from django.http import JsonResponse
from django.views.decorators.csrf import csrf_exempt
import json

from .models import CustomUser, OTPVerification
from .forms import CustomUserCreationForm, CustomLoginForm, OTPVerificationForm, UserProfil

def generate_otp():
    return ''.join(random.choices(string.digits, k=6))

def send_otp_email(user, otp_code, purpose):
    subject = f'Your OTP for {purpose.title()}'
    message = f'''
    Hello {user.first_name},

    Your OTP code for {purpose} is: {otp_code}

    This code will expire in 10 minutes.

    If you didn't request this, please ignore this email.

    Best regards,
    Secure File Manager Team
    '''

    try:
        send_mail(subject, message, settings.EMAIL_HOST_USER, [user.email])
        return True
    except Exception as e:
        print(f"Error sending email: {e}")
        return False

def register_view(request):
    if request.method == 'POST':
        form = CustomUserCreationForm(request.POST)
        if form.is_valid():
            user = form.save(commit=False)
            user.is_active = False # Activate after OTP verification
```

```

        user.save()

        # Generate and send OTP
        otp_code = generate_otp()
        otp = OTPVerification.objects.create(
            user=user,
            otp_code=otp_code,
            purpose='signup',
            expires_at=timezone.now() + timedelta(minutes=10)
        )

        if send_otp_email(user, otp_code, 'account verification'):
            request.session['pending_user_id'] = user.id
            request.session['otp_purpose'] = 'signup'
            messages.success(request, 'Registration successful! Please check your email')
            return redirect('accounts:verify_otp')
        else:
            user.delete() # Clean up if email fails
            messages.error(request, 'Failed to send verification email. Please try again')
    else:
        form = CustomUserCreationForm()

    return render(request, 'accounts/register.html', {'form': form})

def login_view(request):
    if request.method == 'POST':
        form = CustomLoginForm(request.POST)
        if form.is_valid():
            user = form.user
            if not user.is_verified:
                messages.error(request, 'Please verify your email first.')
                return redirect('accounts:login')

            # Generate and send OTP for login
            otp_code = generate_otp()
            # Delete any existing unused OTPs
            OTPVerification.objects.filter(user=user, purpose='login', is_used=False).delete()

            otp = OTPVerification.objects.create(
                user=user,
                otp_code=otp_code,
                purpose='login',
                expires_at=timezone.now() + timedelta(minutes=10)
            )

            if send_otp_email(user, otp_code, 'login verification'):
                request.session['pending_user_id'] = user.id
                request.session['otp_purpose'] = 'login'
                messages.success(request, 'OTP sent to your email. Please verify to login')
                return redirect('accounts:verify_otp')
            else:
                messages.error(request, 'Failed to send OTP. Please try again.')
        else:
            form = CustomLoginForm()

    return render(request, 'accounts/login.html', {'form': form})

```

```

def verify_otp_view(request):
    user_id = request.session.get('pending_user_id')
    purpose = request.session.get('otp_purpose')

    if not user_id or not purpose:
        messages.error(request, 'Invalid verification session.')
        return redirect('accounts:login')

    try:
        user = CustomUser.objects.get(id=user_id)
    except CustomUser.DoesNotExist:
        messages.error(request, 'User not found.')
        return redirect('accounts:login')

    if request.method == 'POST':
        form = OTPVerificationForm(user=user, purpose=purpose, data=request.POST)
        if form.is_valid():
            otp_code = form.cleaned_data['otp_code']

            # Verify and use OTP
            otp = OTPVerification.objects.get(
                user=user,
                otp_code=otp_code,
                purpose=purpose,
                is_used=False
            )
            otp.is_used = True
            otp.save()

            if purpose == 'signup':
                user.is_active = True
                user.is_verified = True
                user.save()
                messages.success(request, 'Account verified successfully! You can now log
                # Clean up session
                del request.session['pending_user_id']
                del request.session['otp_purpose']
                return redirect('accounts:login')

            elif purpose == 'login':
                login(request, user)
                # Clean up session
                del request.session['pending_user_id']
                del request.session['otp_purpose']
                messages.success(request, f'Welcome back, {user.first_name}!')
                return redirect('dashboard:home')

        else:
            form = OTPVerificationForm(user=user, purpose=purpose)

    return render(request, 'accounts/verify_otp.html', {
        'form': form,
        'user': user,
        'purpose': purpose
    })

```

```

@require_http_methods(["POST"])
def resend_otp_view(request):
    user_id = request.session.get('pending_user_id')
    purpose = request.session.get('otp_purpose')

    if not user_id or not purpose:
        return JsonResponse({'success': False, 'message': 'Invalid session'})

    try:
        user = CustomUser.objects.get(id=user_id)

        # Delete old OTPs
        OTPVerification.objects.filter(user=user, purpose=purpose, is_used=False).delete()

        # Generate new OTP
        otp_code = generate_otp()
        otp = OTPVerification.objects.create(
            user=user,
            otp_code=otp_code,
            purpose=purpose,
            expires_at=timezone.now() + timedelta(minutes=10)
        )

        if send_otp_email(user, otp_code, purpose):
            return JsonResponse({'success': True, 'message': 'OTP resent successfully'})
        else:
            return JsonResponse({'success': False, 'message': 'Failed to send OTP'})

    except CustomUser.DoesNotExist:
        return JsonResponse({'success': False, 'message': 'User not found'})

@login_required
def profile_view(request):
    if request.method == 'POST':
        form = UserProfileForm(request.POST, request.FILES, instance=request.user)
        if form.is_valid():
            form.save()
            messages.success(request, 'Profile updated successfully!')
            return redirect('accounts:profile')
    else:
        form = UserProfileForm(instance=request.user)

    return render(request, 'accounts/profile.html', {'form': form})

@login_required
def logout_view(request):
    logout(request)
    messages.success(request, 'Logged out successfully!')
    return redirect('accounts:login')

```

This comprehensive Django file manager project includes all the requested features with detailed implementations. The project structure follows Django best practices and includes:

1. **Role-based access control** with admin and user roles
2. **OTP authentication** for both signup and login

3. **File management** with CRUD operations and favorites
4. **Admin controls** for user and file management
5. **Bootstrap frontend** with responsive design
6. **MySQL database** integration
7. **Complete file structure** with all necessary components

The code provides a solid foundation that you can extend with additional features as needed. Each component is well-documented and follows Django conventions for maintainability and scalability.

HTML Templates for Django File Manager

Here are all the HTML template files for the secure file manager project, organized by application and functionality.

Base Template

File: templates/base.html

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>{% block title %}Secure File Manager{% endblock %}</title>

    <!-- Bootstrap CSS -->
    <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.2/dist/css/bootstrap.min.css"
    <!-- Font Awesome -->
    <link href="https://cdn.jsdelivr.net/npm/@fortawesome/fontawesome-free@6.4.0/css/all.min.css"
    <!-- Custom CSS -->
    <link href="{% load static %}{% static 'css/style.css' %}" rel="stylesheet">

    {% block extra_css %}{% endblock %}
</head>
<body>
    <!-- Navigation -->
    {% if user.is_authenticated %}
    <nav class="navbar navbar-expand-lg navbar-dark bg-primary">
        <div class="container">
            <a class="navbar-brand" href="{% url 'dashboard:home' %}">
                <i class="fas fa-folder-lock me-2"></i>Secure File Manager
            </a>

            <button class="navbar-toggler" type="button" data-bs-toggle="collapse" data-bs-target="#navbarNav"
                <span class="navbar-toggler-icon"></span>
            </button>

            <div class="collapse navbar-collapse" id="navbarNav">
                <ul class="navbar-nav me-auto">
```

```

<li class="nav-item">
  <a class="nav-link" href="{% url 'dashboard:home' %}">
    <i class="fas fa-tachometer-alt me-1"></i>Dashboard
  </a>
</li>
<li class="nav-item">
  <a class="nav-link" href="{% url 'filemanager:file_list' %}">
    <i class="fas fa-files me-1"></i>My Files
  </a>
</li>
<li class="nav-item">
  <a class="nav-link" href="{% url 'filemanager:upload_file' %}">
    <i class="fas fa-upload me-1"></i>Upload
  </a>
</li>
<li class="nav-item">
  <a class="nav-link" href="{% url 'filemanager:shared_files' %}">
    <i class="fas fa-share-alt me-1"></i>Shared
  </a>
</li>
{% if user.is_admin %}
<li class="nav-item dropdown">
  <a class="nav-link dropdown-toggle" href="#" role="button" data-bs-toggle="dropdown">
    <i class="fas fa-cog me-1"></i>Admin
  </a>
  <ul class="dropdown-menu">
    <li><a class="dropdown-item" href="{% url 'dashboard:admin_users' %}">
      <i class="fas fa-users me-2"></i>Manage Users</a></li>
    <li><a class="dropdown-item" href="{% url 'dashboard:admin_files' %}">
      <i class="fas fa-file-alt me-2"></i>All Files</a></li>
    <li><a class="dropdown-item" href="{% url 'dashboard:system_statistics' %}">
      <i class="fas fa-chart-bar me-2"></i>Statistics</a></li>
  </ul>
</li>
{% endif %}
</ul>

<ul class="navbar-nav">
  <li class="nav-item dropdown">
    <a class="nav-link dropdown-toggle" href="#" role="button" data-bs-toggle="dropdown">
      {% if user.profile_picture %}
        
      {% else %}
        <i class="fas fa-user-circle me-1"></i>
      {% endif %}
      {{ user.first_name }}
    </a>
    <ul class="dropdown-menu dropdown-menu-end">
      <li><a class="dropdown-item" href="{% url 'accounts:profile' %}">
        <i class="fas fa-user me-2"></i>Profile</a></li>
      <li><hr class="dropdown-divider"></li>
      <li><a class="dropdown-item" href="{% url 'accounts:logout' %}">
        <i class="fas fa-sign-out-alt me-2"></i>Logout</a></li>
    </ul>
  </li>
</ul>

```

```

        </div>
    </div>
</nav>
{% endif %}

<!-- Messages -->
{% if messages %}
<div class="container mt-3">
    {% for message in messages %}
        <div class="alert alert-{{ message.tags }} alert-dismissible fade show" role="alert">
            {{ message }}
            <button type="button" class="btn-close" data-bs-dismiss="alert"></button>
        </div>
    {% endfor %}
</div>
{% endif %}

<!-- Main Content -->
<main class="{% if user.is_authenticated %}container mt-4{% endif %}">
    {% block content %}{% endblock %}
</main>

<!-- Footer -->
<footer class="bg-light mt-5 py-3">
    <div class="container text-center text-muted">
        <small>&copy; 2025 Secure File Manager. All rights reserved.</small>
    </div>
</footer>

<!-- Bootstrap JS -->
<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.2/dist/js/bootstrap.bundle.min.js"></script>
<!-- jQuery -->
<script src="https://code.jquery.com/jquery-3.6.0.min.js"></script>

    {% block extra_js %}{% endblock %}
</body>
</html>

```

Authentication Templates

File: templates/accounts/login.html

```

{% extends 'base.html' %}
{% load static %}

{% block title %}Login - Secure File Manager{% endblock %}

{% block content %}
<div class="row justify-content-center">
    <div class="col-md-6 col-lg-4">
        <div class="card shadow">
            <div class="card-body p-5">
                <div class="text-center mb-4">
                    <i class="fas fa-folder-lock fa-3x text-primary mb-3"></i>

```

```

        <h3 class="card-title">Welcome Back</h3>
        <p class="text-muted">Sign in to your account</p>
    </div>

    <form method="post" novalidate>
        {% csrf_token %}

        <div class="mb-3">
            <label for="{{ form.email.id_for_label }}" class="form-label">Email</label>
            <div class="input-group">
                <span class="input-group-text"><i class="fas fa-envelope"></i></span>
                {{ form.email }}
            </div>
            {% if form.email.errors %}
                <div class="text-danger small mt-1">{{ form.email.errors.0 }}</div>
            {% endif %}
        </div>

        <div class="mb-4">
            <label for="{{ form.password.id_for_label }}" class="form-label">Password</label>
            <div class="input-group">
                <span class="input-group-text"><i class="fas fa-lock"></i></span>
                {{ form.password }}
            </div>
            {% if form.password.errors %}
                <div class="text-danger small mt-1">{{ form.password.errors.0 }}</div>
            {% endif %}
        </div>

        {% if form.non_field_errors %}
            <div class="alert alert-danger">{{ form.non_field_errors.0 }}</div>
        {% endif %}

        <button type="submit" class="btn btn-primary w-100 mb-3">
            <i class="fas fa-sign-in-alt me-2"></i>Sign In
        </button>
    </form>

    <div class="text-center">
        <p class="mb-0">Don't have an account?
            <a href="{% url 'accounts:register' %}" class="text-decoration-none">
                Register
            </a>
        </p>
    </div>
</div>
</div>
</div>
</div>
{% endblock %}

```

File: templates/accounts/register.html

```
{% extends 'base.html' %}
{% load static %}

{% block title %}Register - Secure File Manager{% endblock %}

{% block content %}
<div class="row justify-content-center">
    <div class="col-md-8 col-lg-6">
        <div class="card shadow">
            <div class="card-body p-5">
                <div class="text-center mb-4">
                    <i class="fas fa-user-plus fa-3x text-primary mb-3"></i>
                    <h3 class="card-title">Create Account</h3>
                    <p class="text-muted">Join our secure file management platform</p>
                </div>

                <form method="post" novalidate>
                    {% csrf_token %}

                    <div class="row">
                        <div class="col-md-6 mb-3">
                            <label for="{{ form.first_name.id_for_label }}" class="form-label">First Name</label>
                            {{ form.first_name }}
                            {% if form.first_name.errors %}
                                <div class="text-danger small mt-1">{{ form.first_name.errors }}</div>
                            {% endif %}
                        </div>

                        <div class="col-md-6 mb-3">
                            <label for="{{ form.last_name.id_for_label }}" class="form-label">Last Name</label>
                            {{ form.last_name }}
                            {% if form.last_name.errors %}
                                <div class="text-danger small mt-1">{{ form.last_name.errors }}</div>
                            {% endif %}
                        </div>
                    </div>

                    <div class="mb-3">
                        <label for="{{ form.username.id_for_label }}" class="form-label">Username</label>
                        <div class="input-group">
                            <span class="input-group-text"><i class="fas fa-user"></i></span>
                            {{ form.username }}
                        </div>
                        {% if form.username.errors %}
                            <div class="text-danger small mt-1">{{ form.username.errors }}</div>
                        {% endif %}
                    </div>

                    <div class="mb-3">
                        <label for="{{ form.email.id_for_label }}" class="form-label">Email</label>
                        <div class="input-group">
                            <span class="input-group-text"><i class="fas fa-envelope"></i></span>
                            {{ form.email }}
                        </div>
                    </div>
                </form>
            </div>
        </div>
    </div>
</div>
{% endblock %}
```

```

        {% if form.email.errors %}
            <div class="text-danger small mt-1">{{ form.email.errors.0 }}
        {% endif %}
    </div>

    <div class="mb-3">
        <label for="{{ form.phone_number.id_for_label }}" class="form-label">
        <div class="input-group">
            <span class="input-group-text"><i class="fas fa-phone"></i></span>
            {{ form.phone_number }}
        </div>
        {% if form.phone_number.errors %}
            <div class="text-danger small mt-1">{{ form.phone_number.errors.0 }}
        {% endif %}
    </div>

    <div class="mb-3">
        <label for="{{ form.password1.id_for_label }}" class="form-label">
        <div class="input-group">
            <span class="input-group-text"><i class="fas fa-lock"></i></span>
            {{ form.password1 }}
        </div>
        {% if form.password1.errors %}
            <div class="text-danger small mt-1">{{ form.password1.errors.0 }}
        {% endif %}
    </div>

    <div class="mb-4">
        <label for="{{ form.password2.id_for_label }}" class="form-label">
        <div class="input-group">
            <span class="input-group-text"><i class="fas fa-lock"></i></span>
            {{ form.password2 }}
        </div>
        {% if form.password2.errors %}
            <div class="text-danger small mt-1">{{ form.password2.errors.0 }}
        {% endif %}
    </div>

    <button type="submit" class="btn btn-primary w-100 mb-3">
        <i class="fas fa-user-plus me-2"></i>Create Account
    </button>
</form>

<div class="text-center">
    <p class="mb-0">Already have an account?
        <a href="{% url 'accounts:login' %}" class="text-decoration-none">
    </p>
</div>
</div>
</div>
</div>
{% endblock %}

```

File: templates/accounts/verify_otp.html

```
{% extends 'base.html' %}
{% load static %}

{% block title %}Verify OTP - Secure File Manager{% endblock %}

{% block content %}
<div class="row justify-content-center">
  <div class="col-md-6 col-lg-4">
    <div class="card shadow">
      <div class="card-body p-5">
        <div class="text-center mb-4">
          <i class="fas fa-shield-alt fa-3x text-warning mb-3"></i>
          <h3 class="card-title">Verify OTP</h3>
          <p class="text-muted">
            We've sent a 6-digit code to<br>
            <strong>{{ user.email }}</strong>
          </p>
        </div>

        <form method="post" id="otpForm">
          {% csrf_token %}

          <div class="mb-4">
            <label for="{{ form.otp_code.id_for_label }}" class="form-label">
              Enter OTP Code
            </label>
            {{ form.otp_code }}
            {% if form.otp_code.errors %}
              <div class="text-danger small mt-1 text-center">{{ form.otp_code.errors }}</div>
            {% endif %}
          </div>

          <button type="submit" class="btn btn-primary w-100 mb-3">
            <i class="fas fa-check me-2"></i>Verify Code
          </button>
        </form>

        <div class="text-center">
          <p class="mb-2">Didn't receive the code?</p>
          <button type="button" class="btn btn-outline-secondary btn-sm" id="resend">
            <i class="fas fa-redo me-1"></i>Resend OTP
          </button>
          <div id="countdown" class="small text-muted mt-2" style="display: none">
            Resend available in <span id="timer">60</span> seconds
          </div>
        </div>
      </div>
    </div>
  </div>
</div>
{% endblock %}

{% block extra_js %}
<script>
```

```

$(document).ready(function() {
    let countdownTimer = null;

    $('#resendBtn').click(function() {
        const button = $(this);
        button.prop('disabled', true);

        $.ajax({
            url: '{% url "accounts:resend_otp" %}',
            type: 'POST',
            headers: {
                'X-CSRFToken': $('[name=csrfmiddlewaretoken]').val()
            },
            success: function(response) {
                if (response.success) {
                    alert('OTP resent successfully!');
                    startCountdown();
                } else {
                    alert('Failed to resend OTP: ' + response.message);
                    button.prop('disabled', false);
                }
            },
            error: function() {
                alert('Network error. Please try again.');
```

```

                button.prop('disabled', false);
            }
        });
    });

    function startCountdown() {
        let timeLeft = 60;
        $('#countdown').show();
        $('#timer').text(timeLeft);

        countdownTimer = setInterval(function() {
            timeLeft--;
            $('#timer').text(timeLeft);

            if (timeLeft <= 0) {
                clearInterval(countdownTimer);
                $('#countdown').hide();
                $('#resendBtn').prop('disabled', false);
            }
        }, 1000);
    }
});
</script>
{% endblock %}

```


File: templates/accounts/profile.html

```
{% extends 'base.html' %}
{% load static %}

{% block title %}Profile - Secure File Manager{% endblock %}

{% block content %}
<div class="row">
    <div class="col-md-8 mx-auto">
        <div class="card shadow">
            <div class="card-header bg-primary text-white">
                <h4 class="mb-0">
                    <i class="fas fa-user me-2"></i>My Profile
                </h4>
            </div>

            <div class="card-body">
                <div class="row">
                    <div class="col-md-4 text-center mb-4">
                        {% if user.profile_picture %}
                            
                        {% else %}
                            <div class="bg-secondary rounded-circle d-inline-flex align-items-center justify-content-center"
                                style="width: 150px; height: 150px;">
                                <i class="fas fa-user fa-4x text-white"></i>
                            </div>
                        {% endif %}

                        <h5>{{ user.get_full_name }}</h5>
                        <span class="badge bg-{{ if user.is_admin %}danger{% else %}primary{% endif %}">
                            {{ user.get_role_display }}
                        </span>
                    </div>

                    <div class="col-md-8">
                        <form method="post" enctype="multipart/form-data">
                            {% csrf_token %}

                            <div class="row">
                                <div class="col-md-6 mb-3">
                                    <label for="{{ form.first_name.id_for_label }}" class="form-label">
                                        {{ form.first_name }}
                                    {% if form.first_name.errors %}
                                        <div class="text-danger small mt-1">{{ form.first_name.errors }}</div>
                                    {% endif %}
                                </div>

                                <div class="col-md-6 mb-3">
                                    <label for="{{ form.last_name.id_for_label }}" class="form-label">
                                        {{ form.last_name }}
                                    {% if form.last_name.errors %}
                                        <div class="text-danger small mt-1">{{ form.last_name.errors }}</div>
                                    {% endif %}
                                </div>
                            </div>
                        </form>
                    </div>
                </div>
            </div>
        </div>
    </div>
</div>
{% endblock %}
```

```

    </div>

    <div class="mb-3">
        <label class="form-label">Email Address</label>
        <input type="email" class="form-control" value="{{ user.e
        <small class="text-muted">Email cannot be changed</small>
    </div>

    <div class="mb-3">
        <label class="form-label">Username</label>
        <input type="text" class="form-control" value="{{ user.us
        <small class="text-muted">Username cannot be changed</sm
    </div>

    <div class="mb-3">
        <label for="{{ form.phone_number.id_for_label }}" class='
        {{ form.phone_number }}
        {% if form.phone_number.errors %}
            <div class="text-danger small mt-1">{{ form.phone_nun
            {% endif %}
        </div>

    <div class="mb-4">
        <label for="{{ form.profile_picture.id_for_label }}" clas
        {{ form.profile_picture }}
        {% if form.profile_picture.errors %}
            <div class="text-danger small mt-1">{{ form.profile_p
            {% endif %}
        </div>

    <button type="submit" class="btn btn-primary">
        <i class="fas fa-save me-2"></i>Update Profile
    </button>
</form>
</div>
</div>
</div>
</div>

<!-- Account Information -->
<div class="card shadow mt-4">
    <div class="card-header">
        <h5 class="mb-0">Account Information</h5>
    </div>
    <div class="card-body">
        <div class="row">
            <div class="col-md-6">
                <p><strong>Member Since:</strong> {{ user.date_joined|date:"F d,
                <p><strong>Last Login:</strong> {{ user.last_login|date:"F d, Y
            </div>
            <div class="col-md-6">
                <p><strong>Account Status:</strong>
                {% if user.is_verified %}
                    <span class="badge bg-success">Verified</span>
                {% else %}
                    <span class="badge bg-warning">Pending Verification</span>

```

```

        {% endif %}
    </p>
    <p><strong>Role:</strong> {{ user.get_role_display }}</p>
</div>
</div>
</div>
</div>
</div>
{% endblock %}

```

File Manager Templates

File: templates/filemanager/file_list.html

```

{% extends 'base.html' %}
{% load static %}

{% block title %}My Files - Secure File Manager{% endblock %}

{% block content %}
<div class="d-flex justify-content-between align-items-center mb-4">
    <h2><i class="fas fa-files me-2"></i>My Files</h2>
    <a href="{% url 'filemanager:upload_file' %}" class="btn btn-primary">
        <i class="fas fa-upload me-2"></i>Upload File
    </a>
</div>

<!-- Search and Filter -->
<div class="card mb-4">
    <div class="card-body">
        <form method="get" class="row g-3">
            <div class="col-md-4">
                <input type="text" name="q" class="form-control" placeholder="Search files"
                    value="{{ request.GET.q }}">
            </div>
            <div class="col-md-3">
                <select name="category" class="form-control">
                    <option value="">All Categories</option>
                    {% for category in categories %}
                        <option value="{{ category.id }}" {% if request.GET.category == category.id %}selected{% endif %}>
                            {{ category.name }}
                        </option>
                    {% endfor %}
                </select>
            </div>
            <div class="col-md-3">
                <select name="type" class="form-control">
                    <option value="">All Types</option>
                    <option value="image" {% if request.GET.type == 'image' %}selected{% endif %}>Image
                    <option value="document" {% if request.GET.type == 'document' %}selected{% endif %}>Document
                    <option value="video" {% if request.GET.type == 'video' %}selected{% endif %}>Video
                    <option value="archive" {% if request.GET.type == 'archive' %}selected{% endif %}>Archive
                </select>
            </div>
        </form>
    </div>
</div>

```

```

        </div>
        <div class="col-md-2">
            <button type="submit" class="btn btn-outline-primary w-100">
                <i class="fas fa-search"></i>
            </button>
        </div>
    </form>
</div>
</div>

<!-- File Statistics -->
<div class="row mb-4">
    <div class="col-md-3">
        <div class="card bg-primary text-white">
            <div class="card-body text-center">
                <i class="fas fa-file fa-2x mb-2"></i>
                <h4>{{ total_files }}</h4>
                <small>Total Files</small>
            </div>
        </div>
    </div>
    <div class="col-md-3">
        <div class="card bg-success text-white">
            <div class="card-body text-center">
                <i class="fas fa-heart fa-2x mb-2"></i>
                <h4>{{ favorite_files }}</h4>
                <small>Favorites</small>
            </div>
        </div>
    </div>
    <div class="col-md-3">
        <div class="card bg-info text-white">
            <div class="card-body text-center">
                <i class="fas fa-share-alt fa-2x mb-2"></i>
                <h4>{{ shared_files }}</h4>
                <small>Shared</small>
            </div>
        </div>
    </div>
    <div class="col-md-3">
        <div class="card bg-warning text-white">
            <div class="card-body text-center">
                <i class="fas fa-hdd fa-2x mb-2"></i>
                <h4>{{ total_size }}</h4>
                <small>Total Size</small>
            </div>
        </div>
    </div>
</div>
</div>

<!-- Files Display Toggle -->
<div class="d-flex justify-content-between align-items-center mb-3">
    <div>
        <strong>{{ files.count }} files found</strong>
    </div>
    <div class="btn-group" role="group">

```

```

        <button type="button" class="btn btn-outline-secondary active" id="gridView">
            <i class="fas fa-th"></i>
        </button>
        <button type="button" class="btn btn-outline-secondary" id="listView">
            <i class="fas fa-list"></i>
        </button>
    </div>
</div>

<!-- Files Grid View -->
<div id="filesGrid" class="row">
    {% for file in files %}
    <div class="col-md-4 col-lg-3 mb-4">
        <div class="card h-100 file-card" data-file-id="{{ file.id }}">
            <div class="card-body text-center p-3">
                <!-- File Icon/Thumbnail -->
                <div class="file-icon mb-3">
                    {% if file.file_type in '.jpg,.jpeg,.png,.gif' %}
                        
                    {% elif file.file_type == '.pdf' %}
                        <i class="fas fa-file-pdf fa-3x text-danger"></i>
                    {% elif file.file_type in '.doc,.docx' %}
                        <i class="fas fa-file-word fa-3x text-primary"></i>
                    {% elif file.file_type in '.xls,.xlsx' %}
                        <i class="fas fa-file-excel fa-3x text-success"></i>
                    {% elif file.file_type in '.ppt,.pptx' %}
                        <i class="fas fa-file-powerpoint fa-3x text-warning"></i>
                    {% elif file.file_type in '.zip,.rar' %}
                        <i class="fas fa-file-archive fa-3x text-info"></i>
                    {% elif file.file_type in '.mp4,.avi,.mov' %}
                        <i class="fas fa-file-video fa-3x text-purple"></i>
                    {% else %}
                        <i class="fas fa-file fa-3x text-secondary"></i>
                    {% endif %}
                </div>

                <!-- File Info -->
                <h6 class="card-title text-truncate" title="{{ file.name }}">{{ file.name }}</h6>
                <small class="text-muted d-block">{{ file.formatted_size }}</small>
                <small class="text-muted d-block">{{ file.created_at|date:"M d, Y" }}</small>

                <!-- Favorite Status -->
                {% if file.is_favorite %}
                    <i class="fas fa-heart text-danger position-absolute" style="top: 10px; left: 10px;" />
                {% endif %}

                <!-- Visibility Badge -->
                <span class="badge bg-{% if file.visibility == 'public' %}success{% elif file.visibility == 'private' %}warning{% else %}info{% endif %}">
                    {{ file.get_visibility_display }}
                </span>
            </div>

            <div class="card-footer p-2">
                <div class="btn-group w-100" role="group">
                    <button class="btn btn-sm btn-outline-primary" onclick="viewFile('{{ file.id }}')">View</button>
                </div>
            </div>
        </div>
    </div>
    </div>

```

```

        <i class="fas fa-eye"></i>
    </button>
    <button class="btn btn-sm btn-outline-success" onclick="downloadFile(
        <i class="fas fa-download"></i>
    </button>
    <button class="btn btn-sm btn-outline-warning" onclick="editFile({{ file.id }})"
        <i class="fas fa-edit"></i>
    </button>
    <button class="btn btn-sm btn-outline-{{ if file.is_favorite }}danger
        onclick="toggleFavorite({{ file.id }})">
        <i class="fas fa-heart"></i>
    </button>
    <button class="btn btn-sm btn-outline-danger" onclick="deleteFile({{
        <i class="fas fa-trash"></i>
    </button>
</div>
</div>
</div>
    {% empty %}
<div class="col-12">
    <div class="text-center py-5">
        <i class="fas fa-folder-open fa-4x text-muted mb-3"></i>
        <h4 class="text-muted">No files found</h4>
        <p class="text-muted">Upload your first file to get started!</p>
        <a href="{{ url 'filemanager:upload_file' }}" class="btn btn-primary">
            <i class="fas fa-upload me-2"></i>Upload File
        </a>
    </div>
</div>
    {% endfor %}
</div>

<!-- Files List View (Hidden by default) -->
<div id="filesList" class="d-none">
    <div class="table-responsive">
        <table class="table table-hover">
            <thead class="table-dark">
                <tr>
                    <th>Name</th>
                    <th>Size</th>
                    <th>Category</th>
                    <th>Visibility</th>
                    <th>Created</th>
                    <th>Actions</th>
                </tr>
            </thead>
            <tbody>
                {% for file in files %}
                <tr>
                    <td>
                        {{ if file.is_favorite }}
                            <i class="fas fa-heart text-danger me-2"></i>
                        {{ endif }}
                        {{ file.name }}
                    </td>

```

```

        <td>{{ file.formatted_size }}</td>
        <td>{{ file.category.name|default:"Uncategorized" }}</td>
        <td>
            <span class="badge bg-{{% if file.visibility == 'public' %}}success
                {{ file.get_visibility_display }}
            </span>
        </td>
        <td>{{ file.created_at|date:"M d, Y g:i A" }}</td>
        <td>
            <div class="btn-group btn-group-sm" role="group">
                <button class="btn btn-outline-primary" onclick="viewFile({{
                    <i class="fas fa-eye"></i>
                </button>
                <button class="btn btn-outline-success" onclick="downloadFile
                    <i class="fas fa-download"></i>
                </button>
                <button class="btn btn-outline-warning" onclick="editFile({{
                    <i class="fas fa-edit"></i>
                </button>
                <button class="btn btn-outline-{{% if file.is_favorite %}}dange
                    onclick="toggleFavorite({{ file.id }})" title="Favori
                    <i class="fas fa-heart"></i>
                </button>
                <button class="btn btn-outline-danger" onclick="deleteFile({{
                    <i class="fas fa-trash"></i>
                </button>
            </div>
        </td>
    </tr>
    {{% endfor %}}
</tbody>
</table>
</div>
</div>

<!-- Pagination -->
{{% if files.has_other_pages %}}
<nav aria-label="Files pagination">
    <ul class="pagination justify-content-center">
        {{% if files.has_previous %}}
            <li class="page-item">
                <a class="page-link" href="?page={{ files.previous_page_number }}{{% if re
            </li>
        {{% endif %}}

        {{% for num in files.paginator.page_range %}}
            {{% if files.number == num %}}
                <li class="page-item active">
                    <span class="page-link">{{ num }}</span>
                </li>
            {{% else %}}
                <li class="page-item">
                    <a class="page-link" href="?page={{ num }}{{% if request.GET.q %}}&q={{
                </li>
            {{% endif %}}
        {{% endfor %}}

```

```

        {% if files.has_next %}
            <li class="page-item">
                <a class="page-link" href="?page={{ files.next_page_number }}{% if request %}>
                </li>
            {% endif %}
        </ul>
    </nav>
{% endif %}
{% endblock %}

{% block extra_js %}
<script>
// View Toggle
$('#gridView').click(function() {
    $(this).addClass('active');
    $('#listView').removeClass('active');
    $('#filesGrid').removeClass('d-none');
    $('#filesList').addClass('d-none');
});

$('#listView').click(function() {
    $(this).addClass('active');
    $('#gridView').removeClass('active');
    $('#filesGrid').addClass('d-none');
    $('#filesList').removeClass('d-none');
});

// File Actions
function viewFile(fileId) {
    window.open(`${% url 'filemanager:view_file' 0 %}`.replace('0', fileId), '_blank');
}

function downloadFile(fileId) {
    window.location.href = `${% url 'filemanager:download_file' 0 %}`.replace('0', fileId);
}

function editFile(fileId) {
    window.location.href = `${% url 'filemanager:edit_file' 0 %}`.replace('0', fileId);
}

function toggleFavorite(fileId) {
    $.ajax({
        url: `${% url 'filemanager:toggle_favorite' 0 %}`.replace('0', fileId),
        type: 'POST',
        headers: {
            'X-CSRFToken': $('[name=csrfmiddlewaretoken]').val()
        },
        success: function(response) {
            if (response.success) {
                location.reload();
            }
        }
    });
}

```



```

function deleteFile(fileId) {
    if (confirm('Are you sure you want to delete this file? This action cannot be undone.
        $.ajax({
            url: `${% url 'filemanager:delete_file' 0 %}`.replace('0', fileId),
            type: 'POST',
            headers: {
                'X-CSRFToken': $('[name=csrfmiddlewaretoken]').val()
            },
            success: function(response) {
                if (response.success) {
                    location.reload();
                } else {
                    alert('Error deleting file: ' + response.message);
                }
            }
        });
    }
}
</script>
{% endblock %}

```

File: templates/filemanager/upload_file.html

```

{% extends 'base.html' %}
{% load static %}

{% block title %}Upload File - Secure File Manager{% endblock %}

{% block content %}
<div class="row justify-content-center">
    <div class="col-md-8">
        <div class="card shadow">
            <div class="card-header bg-primary text-white">
                <h4 class="mb-0">
                    <i class="fas fa-upload me-2"></i>Upload New File
                </h4>
            </div>

            <div class="card-body">
                <form method="post" enctype="multipart/form-data" id="uploadForm">
                    {% csrf_token %}

                    <div class="mb-4">
                        <label for="{{ form.file.id_for_label }}" class="form-label">Choose File
                        <div class="file-drop-area" id="fileDropArea">
                            <div class="file-drop-icon">
                                <i class="fas fa-cloud-upload-alt fa-3x text-muted"></i>
                            </div>
                            <div class="file-drop-text">
                                <p class="mb-2"><strong>Drop files here or click to browse</strong>
                                <p class="text-muted small">
                                    Supported formats: PDF, Word, Excel, PowerPoint, Images
                                    <br>Maximum file size: 25MB
                                </p>
                            </div>
                        </div>
                    </div>
                </form>
            </div>
        </div>
    </div>
</div>

```

```

        {{ form.file }}
    </div>
    {% if form.file.errors %}
        <div class="text-danger small mt-1">{{ form.file.errors.0 }}<
    {% endif %}

    <!-- File Preview -->
    <div id="filePreview" class="mt-3" style="display: none;">
        <div class="card">
            <div class="card-body p-3">
                <div class="d-flex align-items-center">
                    <div class="file-icon me-3">
                        <i id="fileIcon" class="fas fa-file fa-2x te
                    </div>
                    <div class="flex-grow-1">
                        <h6 id="fileName" class="mb-1"></h6>
                        <small id="fileSize" class="text-muted"></sma
                    </div>
                    <button type="button" class="btn btn-sm btn-outli
                        <i class="fas fa-times"></i>
                    </button>
                </div>
            </div>
        </div>
    </div>
</div>

<div class="mb-3">
    <label for="{{ form.name.id_for_label }}" class="form-label">File
    {{ form.name }}
    {% if form.name.errors %}
        <div class="text-danger small mt-1">{{ form.name.errors.0 }}<
    {% endif %}
    <small class="form-text text-muted">Leave empty to use original d
</div>

<div class="mb-3">
    <label for="{{ form.description.id_for_label }}" class="form-labe
    {{ form.description }}
    {% if form.description.errors %}
        <div class="text-danger small mt-1">{{ form.description.error
    {% endif %}
</div>

<div class="row">
    <div class="col-md-6 mb-3">
        <label for="{{ form.category.id_for_label }}" class="form-lab
        {{ form.category }}
        {% if form.category.errors %}
            <div class="text-danger small mt-1">{{ form.category.error
        {% endif %}
    </div>

    <div class="col-md-6 mb-3">
        <label for="{{ form.visibility.id_for_label }}" class="form-l
        {{ form.visibility }}

```

```

        {% if form.visibility.errors %}
            <div class="text-danger small mt-1">{{ form.visibility.e
        {% endif %}
        <small class="form-text text-muted">
            Private: Only you can see<br>
            Restricted: You control who can access<br>
            Public: Anyone with link can view
        </small>
    </div>
</div>

<!-- Upload Progress -->
<div id="uploadProgress" class="mb-3" style="display: none;">
    <label class="form-label">Upload Progress</label>
    <div class="progress">
        <div id="progressBar" class="progress-bar progress-bar-striped
            role="progressbar" style="width: 0%">0%</div>
    </div>
</div>

<div class="d-flex justify-content-between">
    <a href="{% url 'filemanager:file_list' %}" class="btn btn-second
        <i class="fas fa-arrow-left me-2"></i>Back to Files
    </a>
    <button type="submit" class="btn btn-primary" id="submitBtn">
        <i class="fas fa-upload me-2"></i>Upload File
    </button>
</div>
</form>
</div>
</div>

<!-- Upload Tips -->
<div class="card mt-4">
    <div class="card-header">
        <h6 class="mb-0">
            <i class="fas fa-lightbulb me-2"></i>Upload Tips
        </h6>
    </div>
    <div class="card-body">
        <ul class="mb-0">
            <li>Supported file types: PDF, Word documents, Excel sheets, PowerPoi
            <li>Maximum file size is 25MB per file</li>
            <li>Use descriptive names and descriptions for easier searching</li>
            <li>Choose appropriate categories and visibility settings</li>
            <li>Private files are only visible to you</li>
            <li>Restricted files can be shared with specific users</li>
            <li>Public files can be accessed by anyone with the link</li>
        </ul>
    </div>
</div>
</div>
</div>
{% endblock %}

{% block extra_css %}

```

```

<style>
.file-drop-area {
    border: 2px dashed #007bff;
    border-radius: 10px;
    padding: 40px 20px;
    text-align: center;
    cursor: pointer;
    transition: all 0.3s ease;
    position: relative;
}

.file-drop-area:hover {
    border-color: #0056b3;
    background-color: #f8f9fa;
}

.file-drop-area.dragover {
    border-color: #28a745;
    background-color: #d4edda;
}

.file-drop-area input[type="file"] {
    position: absolute;
    width: 100%;
    height: 100%;
    top: 0;
    left: 0;
    opacity: 0;
    cursor: pointer;
}
</style>
{% endblock %}

{% block extra_js %}
<script>
$(document).ready(function() {
    const fileInput = $('#id_file');
    const fileDropArea = $('#fileDropArea');
    const filePreview = $('#filePreview');
    const fileName = $('#fileName');
    const fileSize = $('#fileSize');
    const fileIcon = $('#fileIcon');
    const nameInput = $('#id_name');

    // Drag and drop handlers
    fileDropArea.on('dragover dragenter', function(e) {
        e.preventDefault();
        e.stopPropagation();
        $(this).addClass('dragover');
    });

    fileDropArea.on('dragleave dragend', function(e) {
        e.preventDefault();
        e.stopPropagation();
        $(this).removeClass('dragover');
    });
});

```

```

fileDropArea.on('drop', function(e) {
    e.preventDefault();
    e.stopPropagation();
    $(this).removeClass('dragover');

    const files = e.originalEvent.dataTransfer.files;
    if (files.length > 0) {
        fileInput[0].files = files;
        handleFileSelect(files[0]);
    }
});

// File input change handler
fileInput.change(function() {
    const files = this.files;
    if (files.length > 0) {
        handleFileSelect(files[0]);
    }
});

function handleFileSelect(file) {
    // Update file preview
    fileName.text(file.name);
    fileSize.text(formatFileSize(file.size));

    // Update file icon based on type
    const extension = file.name.split('.').pop().toLowerCase();
    updateFileIcon(extension);

    // Auto-fill name if empty
    if (!nameInput.val()) {
        nameInput.val(file.name.substring(0, file.name.lastIndexOf('.')));
    }

    // Show preview
    filePreview.show();
}

function updateFileIcon(extension) {
    let iconClass = 'fas fa-file';
    let colorClass = 'text-muted';

    if (['jpg', 'jpeg', 'png', 'gif'].includes(extension)) {
        iconClass = 'fas fa-file-image';
        colorClass = 'text-info';
    } else if (extension === 'pdf') {
        iconClass = 'fas fa-file-pdf';
        colorClass = 'text-danger';
    } else if (['doc', 'docx'].includes(extension)) {
        iconClass = 'fas fa-file-word';
        colorClass = 'text-primary';
    } else if (['xls', 'xlsx'].includes(extension)) {
        iconClass = 'fas fa-file-excel';
        colorClass = 'text-success';
    } else if (['ppt', 'pptx'].includes(extension)) {

```

```

        iconClass = 'fas fa-file-powerpoint';
        colorClass = 'text-warning';
    } else if (['zip', 'rar'].includes(extension)) {
        iconClass = 'fas fa-file-archive';
        colorClass = 'text-info';
    } else if (['mp4', 'avi', 'mov'].includes(extension)) {
        iconClass = 'fas fa-file-video';
        colorClass = 'text-purple';
    }

    fileIcon.removeClass().addClass(`${iconClass} fa-2x ${colorClass}`);
}

function formatFileSize(bytes) {
    const sizes = ['B', 'KB', 'MB', 'GB'];
    if (bytes === 0) return '0 B';
    const i = Math.floor(Math.log(bytes) / Math.log(1024));
    return Math.round(bytes / Math.pow(1024, i) * 100) / 100 + ' ' + sizes[i];
}

// Form submission with progress
$('#uploadForm').submit(function(e) {
    e.preventDefault();

    const formData = new FormData(this);
    const submitBtn = $('#submitBtn');
    const progressDiv = $('#uploadProgress');
    const progressBar = $('#progressBar');

    submitBtn.prop('disabled', true).html('<i class="fas fa-spinner fa-spin me-2"></i>');
    progressDiv.show();

    $.ajax({
        url: $(this).attr('action') || window.location.href,
        type: 'POST',
        data: formData,
        processData: false,
        contentType: false,
        xhr: function() {
            const xhr = new window.XMLHttpRequest();
            xhr.upload.addEventListener("progress", function(evt) {
                if (evt.lengthComputable) {
                    const percentComplete = Math.round((evt.loaded / evt.total) * 100);
                    progressBar.css('width', percentComplete + '%').text(percentComplete + '%');
                }
            }, false);
            return xhr;
        },
        success: function(response) {
            if (response.success) {
                window.location.href = `${url} "filemanager:file_list" %}`;
            } else {
                alert('Upload failed: ' + response.message);
                submitBtn.prop('disabled', false).html('<i class="fas fa-upload me-2"></i>');
                progressDiv.hide();
            }
        }
    });

```

```

    },
    error: function() {
        alert('Upload failed. Please try again.');
```

```

        submitBtn.prop('disabled', false).html('<i class="fas fa-upload me-2"></i>');
        progressDiv.hide();
    }
});
});
});

function removeFile() {
    $('#id_file').val('');
    $('#id_name').val('');
    $('#filePreview').hide();
}
</script>
{% endblock %}
```

File: templates/filemanager/file_detail.html

```

{% extends 'base.html' %}
{% load static %}

{% block title %}{{ file.name }} - Secure File Manager{% endblock %}

{% block content %}
<div class="row">
    <div class="col-md-8">
        <!-- File Preview -->
        <div class="card shadow">
            <div class="card-header d-flex justify-content-between align-items-center">
                <h5 class="mb-0">
                    <i class="fas fa-file me-2"></i>{{ file.name }}
                </h5>
                <div class="btn-group" role="group">
                    <a href="{% url 'filemanager:download_file' file.id %}" class="btn btn-warning">
                        <i class="fas fa-download me-1"></i>Download
                    </a>
                    {% if file.owner == request.user %}
                    <a href="{% url 'filemanager:edit_file' file.id %}" class="btn btn-warning">
                        <i class="fas fa-edit me-1"></i>Edit
                    </a>
                    {% endif %}
                </div>
            </div>
        </div>

        <div class="card-body">
            <!-- File Preview Area -->
            <div class="file-preview mb-4">
                {% if file.file_type in '.jpg,.jpeg,.png,.gif' %}
                    
                {% elif file.file_type == '.pdf' %}
                    <div class="text-center p-4">
                        <i class="fas fa-file-pdf fa-5x text-danger mb-3"></i>
                        <p class="text-muted">PDF Document</p>
                    </div>
                </div>
            </div>
        </div>
    </div>
</div>
{% endblock %}
```

```

        <iframe src="{{ file.file.url }}" width="100%" height="500px"
    </div>
    {% elif file.file_type in '.mp4,.avi,.mov' %}
        <video controls class="w-100 rounded">
            <source src="{{ file.file.url }}" type="video/mp4">
            Your browser does not support the video tag.
        </video>
    {% else %}
        <div class="text-center p-5">
            {% if file.file_type == '.doc' or file.file_type == '.docx' %}
                <i class="fas fa-file-word fa-5x text-primary mb-3"></i>
            {% elif file.file_type == '.xls' or file.file_type == '.xlsx' %}
                <i class="fas fa-file-excel fa-5x text-success mb-3"></i>
            {% elif file.file_type == '.ppt' or file.file_type == '.pptx' %}
                <i class="fas fa-file-powerpoint fa-5x text-warning mb-3"></i>
            {% elif file.file_type == '.zip' or file.file_type == '.rar' %}
                <i class="fas fa-file-archive fa-5x text-info mb-3"></i>
            {% else %}
                <i class="fas fa-file fa-5x text-secondary mb-3"></i>
            {% endif %}
            <h4>{{ file.name }}</h4>
            <p class="text-muted">{{ file.formatted_size }} • {{ file.file_type }}</p>
            <p class="text-muted">Click download to open this file</p>
        </div>
    {% endif %}
</div>

<!-- Description -->
{% if file.description %}
<div class="mb-4">
    <h6>Description</h6>
    <p class="text-muted">{{ file.description }}</p>
</div>
{% endif %}
</div>
</div>

<!-- Comments Section (if implemented) -->
<div class="card shadow mt-4">
    <div class="card-header">
        <h6 class="mb-0">Comments</h6>
    </div>
    <div class="card-body">
        <p class="text-muted text-center">Comments feature coming soon...</p>
    </div>
</div>
</div>

<div class="col-md-4">
    <!-- File Information -->
    <div class="card shadow">
        <div class="card-header">
            <h6 class="mb-0">File Information</h6>
        </div>
        <div class="card-body">
            <table class="table table-sm table-borderless">

```



```
|  |  | |
|---|---|---|
| <strong>Owner:</strong></td>  {{ file.owner.get_full_name }} | |
| <strong>Size:</strong></td>  {{ file.formatted_size }} | |
| <strong>Type:</strong></td>  {{ file.file_type|upper }} | |
| <strong>Category:</strong></td>  {{ file.category.name|default:"Uncategorized" }} | |
| <strong>Visibility:</strong></td>   | |
| <strong>Modified:</strong></td>  {{ file.updated_at|date:"M d, Y g:i A" }} | |
| <strong>Downloads:</strong></td>  {{ file.download_count }} | |



{% if file.owner == request.user %}


<button class="btn btn-outline-{{ if file.is_favorite }}danger{{ else }}info{{ /if }}" data-bs-toggle="toggleFavorite({{ file.id }})">
<i class="fas fa-heart me-2"></i>
{% if file.is_favorite %}Remove from Favorites{% else %}Add to Favorites{% /if %}
</button>

{% if file.visibility == 'restricted' %}
<button class="btn btn-outline-info" data-bs-toggle="modal" data-bs-target="#shareModal" data-file-id="{{ file.id }}">
<i class="fas fa-share-alt me-2"></i>Share File
</button>
{% endif %}
</div>
{% endif %}
</div>
</div>


```

```

<!-- File Activity -->
<div class="card shadow mt-4">
  <div class="card-header">
    <h6 class="mb-0">Recent Activity</h6>
  </div>
  <div class="card-body">
    {% for activity in file.activities.all|slice:"5" %}
    <div class="d-flex align-items-center mb-2">
      <div class="flex-shrink-0">
        <i class="fas fa-{% if activity.action == 'upload' %}upload{% eli
      </div>
      <div class="flex-grow-1 ms-3">
        <small class="d-block">{{ activity.user.get_full_name }} {{ activ
        <small class="text-muted">{{ activity.timestamp|timesince }} ago<
      </div>
    </div>
    {% empty %}
    <p class="text-muted text-center mb-0">No activity yet</p>
    {% endfor %}
  </div>
</div>

<!-- Shared Users (if owner) -->
{% if file.owner == request.user and file.visibility == 'restricted' %}
<div class="card shadow mt-4">
  <div class="card-header">
    <h6 class="mb-0">Shared With</h6>
  </div>
  <div class="card-body">
    {% for access in file.access_permissions.all %}
    <div class="d-flex justify-content-between align-items-center mb-2">
      <div>
        <small class="d-block">{{ access.user.get_full_name }}</small>
        <small class="text-muted">{{ access.get_permission_display }}</sm
      </div>
      <button class="btn btn-sm btn-outline-danger" onclick="removeAccess({
        <i class="fas fa-times"></i>
      </button>
    </div>
    {% empty %}
    <p class="text-muted text-center mb-0">Not shared with anyone</p>
    {% endfor %}
  </div>
</div>
{% endif %}
</div>
</div>

<!-- Share Modal -->
{% if file.owner == request.user and file.visibility == 'restricted' %}
<div class="modal fade" id="shareModal" tabindex="-1">
  <div class="modal-dialog">
    <div class="modal-content">
      <div class="modal-header">
        <h5 class="modal-title">Share File</h5>
        <button type="button" class="btn-close" data-bs-dismiss="modal"></button>

```

```

    </div>
    <form id="shareForm">
        {% csrf_token %}
        <div class="modal-body">
            <div class="mb-3">
                <label class="form-label">User Email</label>
                <input type="email" name="user_email" class="form-control" required="" />
            </div>
            <div class="mb-3">
                <label class="form-label">Permission</label>
                <select name="permission" class="form-control">
                    <option value="view">View Only</option>
                    <option value="download">View & Download</option>
                    <option value="edit">View, Download & Edit</option>
                </select>
            </div>
            <div class="mb-3">
                <label class="form-label">Expires At (Optional)</label>
                <input type="datetime-local" name="expires_at" class="form-control" />
            </div>
        </div>
        <div class="modal-footer">
            <button type="button" class="btn btn-secondary" data-bs-dismiss="modal">Close</button>
            <button type="submit" class="btn btn-primary">Share File</button>
        </div>
    </form>
</div>
</div>
{% endif %}
{% endblock %}

{% block extra_js %}
<script>
function toggleFavorite(fileId) {
    $.ajax({
        url: '{% url "filemanager:toggle_favorite" 0 %}' .replace('0', fileId),
        type: 'POST',
        headers: {
            'X-CSRFToken': $(''[name=csrfmiddlewaretoken]'').val()
        },
        success: function(response) {
            if (response.success) {
                location.reload();
            }
        }
    });
}

function removeAccess(accessId) {
    if (confirm('Remove access for this user?')) {
        $.ajax({
            url: '{% url "filemanager:remove_access" 0 %}' .replace('0', accessId),
            type: 'POST',
            headers: {
                'X-CSRFToken': $(''[name=csrfmiddlewaretoken]'').val()
            }
        });
    }
}

```

```

        },
        success: function(response) {
            if (response.success) {
                location.reload();
            }
        }
    });
}

$( '#shareForm' ).submit(function(e) {
    e.preventDefault();

    $.ajax({
        url: '{% url "filemanager:share_file" file.id %}',
        type: 'POST',
        data: $(this).serialize(),
        success: function(response) {
            if (response.success) {
                location.reload();
            } else {
                alert('Error: ' + response.message);
            }
        }
    });
});
</script>
{% endblock %}

```

File: templates/dashboard/home.html

```

{% extends 'base.html' %}
{% load static %}

{% block title %}Dashboard - Secure File Manager{% endblock %}

{% block content %}
<!-- Welcome Section -->
<div class="row mb-4">
    <div class="col-12">
        <div class="card bg-gradient-primary text-white">
            <div class="card-body">
                <div class="row align-items-center">
                    <div class="col-md-8">
                        <h2 class="mb-2">Welcome back, {{ user.first_name }}!</h2>
                        <p class="mb-0">Manage your files securely and efficiently with c
                    </div>
                    <div class="col-md-4 text-end">
                        <i class="fas fa-tachometer-alt fa-4x opacity-50"></i>
                    </div>
                </div>
            </div>
        </div>
    </div>
</div>
</div>

```

```

<!-- Statistics Cards -->
<div class="row mb-4">
  <div class="col-lg-3 col-md-6 mb-4">
    <div class="card border-left-primary shadow h-100">
      <div class="card-body">
        <div class="row no-gutters align-items-center">
          <div class="col mr-2">
            <div class="text-xs font-weight-bold text-primary text-uppercase">
              <div class="h5 mb-0 font-weight-bold text-gray-800">{{ stats.total
            </div>
          </div>
          <div class="col-auto">
            <i class="fas fa-file fa-2x text-gray-300"></i>
          </div>
        </div>
      </div>
    </div>
  </div>

  <div class="col-lg-3 col-md-6 mb-4">
    <div class="card border-left-success shadow h-100">
      <div class="card-body">
        <div class="row no-gutters align-items-center">
          <div class="col mr-2">
            <div class="text-xs font-weight-bold text-success text-uppercase">
              <div class="h5 mb-0 font-weight-bold text-gray-800">{{ stats.total
            </div>
          </div>
          <div class="col-auto">
            <i class="fas fa-hdd fa-2x text-gray-300"></i>
          </div>
        </div>
      </div>
    </div>
  </div>

  <div class="col-lg-3 col-md-6 mb-4">
    <div class="card border-left-info shadow h-100">
      <div class="card-body">
        <div class="row no-gutters align-items-center">
          <div class="col mr-2">
            <div class="text-xs font-weight-bold text-info text-uppercase mb-
              <div class="h5 mb-0 font-weight-bold text-gray-800">{{ stats.shar
            </div>
          </div>
          <div class="col-auto">
            <i class="fas fa-share-alt fa-2x text-gray-300"></i>
          </div>
        </div>
      </div>
    </div>
  </div>

  <div class="col-lg-3 col-md-6 mb-4">
    <div class="card border-left-warning shadow h-100">
      <div class="card-body">
        <div class="row no-gutters align-items-center">
          <div class="col mr-2">

```

```
<div class="text-xs font-weight-bold text-warning text-uppercase"></div>  
<div class="h5 mb-0 font-weight-bold text-gray-800">{{ stats.total }}</div>  
<div class="col-auto">  
    <i class="fas fa-download fa-2x text-gray-300"></i>  
</div>  
</div>  
</div>  
</div>  
</div>  
</div>  
  
<div class="row">  
    <!-- Recent Files -->  
    <div class="col-lg-8 mb-4">  
        <div class="card shadow">  
            <div class="card-header py-3 d-flex justify-content-between align-items-center">  
                <h6 class="m-0 font-weight-bold text-primary">Recent Files</h6>  
                <a href="{% url 'filemanager:file_list' %}" class="btn btn-sm btn-primary"></a>  
            </div>  
            <div class="card-body">  
                {% for file in recent_files %}  
                    <div class="d-flex align-items-center mb-3 pb-3 border-bottom">  
                        <div class="file-icon me-3">  
                            {% if file.file_type in '.jpg,.jpeg,.png,.gif' %}  
                                <i class="fas fa-file-image fa-2x text-info"></i>  
                            {% elif file.file_type == '.pdf' %}  
                                <i class="fas fa-file-pdf fa-2x text-danger"></i>  
                            {% elif file.file_type in '.doc,.docx' %}  
                                <i class="fas fa-file-word fa-2x text-primary"></i>  
                            {% elif file.file_type in '.xls,.xlsx' %}  
                                <i class="fas fa-file-excel fa-2x text-success"></i>  
                            {% else %}  
                                <i class="fas fa-file fa-2x text-secondary"></i>  
                            {% endif %}  
                        </div>  
                        <div class="flex-grow-1">  
                            <h6 class="mb-1">  
                                <a href="{% url 'filemanager:view_file' file.id %}" class="text-decoration-none">{{ file.name }}</a>  
                                {% if file.is_favorite %}  
                                    <i class="fas fa-heart text-danger ms-2"></i>  
                                {% endif %}  
                            </h6>  
                            <small class="text-muted">  
                                {{ file.formatted_size }} • {{ file.created_at|timesince }} ago  
                            </small>  
                        </div>  
                        <div class="text-end">  
                            <span class="badge bg-{% if file.visibility == 'public' %}success{% else %}warning{% endif %}>{{ file.get_visibility_display }}</span>  
                        </div>  
                    </div>  
                {% empty %}  
                    <p>No files found.</p>  
                </div>  
            </div>  
        </div>  
    </div>
```

```

        <div class="text-center py-4">
            <i class="fas fa-folder-open fa-3x text-muted mb-3"></i>
            <p class="text-muted">No files uploaded yet</p>
            <a href="{% url 'filemanager:upload_file' %}" class="btn btn-primary">
                <i class="fas fa-upload me-2"></i>Upload Your First File
            </a>
        </div>
    {% endfor %}
</div>
</div>
</div>

<!-- Quick Actions & Activity -->
<div class="col-lg-4">
    <!-- Quick Actions -->
    <div class="card shadow mb-4">
        <div class="card-header py-3">
            <h6 class="m-0 font-weight-bold text-primary">Quick Actions</h6>
        </div>
        <div class="card-body">
            <div class="d-grid gap-2">
                <a href="{% url 'filemanager:upload_file' %}" class="btn btn-primary">
                    <i class="fas fa-upload me-2"></i>Upload File
                </a>
                <a href="{% url 'filemanager:file_list' %}?favorites=1" class="btn btn-primary">
                    <i class="fas fa-heart me-2"></i>My Favorites
                </a>
                <a href="{% url 'filemanager:shared_files' %}" class="btn btn-outline-primary">
                    <i class="fas fa-share-alt me-2"></i>Shared Files
                </a>
                <a href="{% url 'accounts:profile' %}" class="btn btn-outline-secondary">
                    <i class="fas fa-user me-2"></i>Edit Profile
                </a>
            </div>
        </div>
    </div>
</div>

<!-- Recent Activity -->
<div class="card shadow">
    <div class="card-header py-3">
        <h6 class="m-0 font-weight-bold text-primary">Recent Activity</h6>
    </div>
    <div class="card-body">
        {% for activity in recent_activities %}
        <div class="d-flex align-items-center mb-3">
            <div class="flex-shrink-0">
                <i class="fas fa-{{ if activity.action == 'upload' }}upload text-primary"></i>
            </div>
            <div class="flex-grow-1 ms-3">
                <small class="d-block">{{ activity.get_action_display }}
                    <strong>{{ activity.file.name|truncatechars:20 }}</strong>
                </small>
                <small class="text-muted">{{ activity.timestamp|timesince }} ago</small>
            </div>
        </div>
        <div class="d-flex align-items-center mb-3">
            <div class="flex-shrink-0">
                <i class="fas fa-{{ if activity.action == 'upload' }}upload text-primary"></i>
            </div>
            <div class="flex-grow-1 ms-3">
                <small class="d-block">{{ activity.get_action_display }}
                    <strong>{{ activity.file.name|truncatechars:20 }}</strong>
                </small>
                <small class="text-muted">{{ activity.timestamp|timesince }} ago</small>
            </div>
        </div>
    </div>
    <div class="card-body">
        {% empty %}
    </div>
</div>

```

```

        <p class="text-muted text-center mb-0">No recent activity</p>
        {% endfor %}
    </div>
</div>
</div>
</div>

<!-- Admin Section (if admin user) -->
{% if user.is_admin %}
<div class="row mt-4">
    <div class="col-12">
        <div class="card shadow">
            <div class="card-header bg-danger text-white py-3">
                <h6 class="m-0 font-weight-bold">
                    <i class="fas fa-cog me-2"></i>Admin Panel
                </h6>
            </div>
            <div class="card-body">
                <div class="row">
                    <div class="col-md-3 mb-3">
                        <div class="text-center">
                            <i class="fas fa-users fa-2x text-primary mb-2"></i>
                            <h5>{{ admin_stats.total_users }}</h5>
                            <small class="text-muted">Total Users</small>
                        </div>
                    </div>
                    <div class="col-md-3 mb-3">
                        <div class="text-center">
                            <i class="fas fa-file-alt fa-2x text-success mb-2"></i>
                            <h5>{{ admin_stats.total_files }}</h5>
                            <small class="text-muted">All Files</small>
                        </div>
                    </div>
                    <div class="col-md-3 mb-3">
                        <div class="text-center">
                            <i class="fas fa-hdd fa-2x text-info mb-2"></i>
                            <h5>{{ admin_stats.total_storage }}</h5>
                            <small class="text-muted">Storage Used</small>
                        </div>
                    </div>
                    <div class="col-md-3 mb-3">
                        <div class="text-center">
                            <i class="fas fa-download fa-2x text-warning mb-2"></i>
                            <h5>{{ admin_stats.total_downloads }}</h5>
                            <small class="text-muted">Downloads</small>
                        </div>
                    </div>
                </div>
            </div>
        </div>

        <div class="d-flex justify-content-center mt-3">
            <div class="btn-group" role="group">
                <a href="{% url 'dashboard:admin_users' %}" class="btn btn-outline-danger">
                    <i class="fas fa-users me-2"></i>Manage Users
                </a>
                <a href="{% url 'dashboard:admin_files' %}" class="btn btn-outline-danger">
                    <i class="fas fa-file-alt me-2"></i>All Files
            </div>
        </div>
    </div>
</div>

```



```

        </a>
        <a href="{% url 'dashboard:system_stats' %}" class="btn btn-outline-
        <i class="fas fa-chart-bar me-2"></i>Statistics
        </a>
    </div>
</div>
</div>
</div>
</div>
</div>
</div>
{% endif %}
{% endblock %}

{% block extra_css %}
<style>
.border-left-primary {
    border-left: 0.25rem solid #4e73df !important;
}

.border-left-success {
    border-left: 0.25rem solid #1cc88a !important;
}

.border-left-info {
    border-left: 0.25rem solid #36b9cc !important;
}

.border-left-warning {
    border-left: 0.25rem solid #f6c23e !important;
}

.bg-gradient-primary {
    background: linear-gradient(87deg, #5e72e4 0, #825ee4 100%) !important;
}
</style>
{% endblock %}

```

Custom CSS File

File: static/css/style.css

```

/* Custom styles for Secure File Manager */

:root {
    --primary-color: #4e73df;
    --secondary-color: #858796;
    --success-color: #1cc88a;
    --info-color: #36b9cc;
    --warning-color: #f6c23e;
    --danger-color: #e74a3b;
    --light-color: #f8f9fc;
    --dark-color: #5a5c69;
}

```

```
body {
  font-family: 'Nunito', -apple-system, BlinkMacSystemFont, 'Segoe UI', 'Roboto', 'Helvetica', 'Arial', sans-serif;
  background-color: #f8f9fc;
  color: #5a5c69;
}

/* Navigation */
.navbar-brand {
  font-weight: bold;
}

/* Cards */
.card {
  border: none;
  border-radius: 0.35rem;
  box-shadow: 0 0.15rem 1.75rem 0 rgba(58, 59, 69, 0.15);
}

.card-header {
  background-color: #f8f9fc;
  border-bottom: 1px solid #e3e6f0;
}

/* File Cards */
.file-card {
  transition: transform 0.2s, box-shadow 0.2s;
  cursor: pointer;
}

.file-card:hover {
  transform: translateY(-2px);
  box-shadow: 0 0.5rem 2rem 0 rgba(58, 59, 69, 0.2);
}

.file-icon {
  min-height: 100px;
  display: flex;
  align-items: center;
  justify-content: center;
}

/* Buttons */
.btn {
  border-radius: 0.35rem;
  font-weight: 400;
}

.btn-group .btn {
  border-radius: 0;
}

.btn-group .btn:first-child {
  border-top-left-radius: 0.35rem;
  border-bottom-left-radius: 0.35rem;
}
```

```
.btn-group .btn:last-child {
  border-top-right-radius: 0.35rem;
  border-bottom-right-radius: 0.35rem;
}

/* Forms */
.form-control {
  border-radius: 0.35rem;
  border: 1px solid #d1d3e2;
}

.form-control:focus {
  border-color: #bac8f3;
  box-shadow: 0 0 0 0.2rem rgba(78, 115, 223, 0.25);
}

/* File upload area */
.file-drop-area {
  border: 2px dashed #d1d3e2;
  border-radius: 0.35rem;
  transition: all 0.3s ease;
  background-color: #fff;
}

.file-drop-area:hover {
  border-color: var(--primary-color);
  background-color: #f8f9fc;
}

.file-drop-area.dragover {
  border-color: var(--success-color);
  background-color: #d4edda;
}

/* Alerts */
.alert {
  border: none;
  border-radius: 0.35rem;
}

/* Tables */
.table {
  color: #5a5c69;
}

.table th {
  border-top: none;
  font-weight: 600;
  text-transform: uppercase;
  font-size: 0.85rem;
}

/* Badges */
.badge {
  font-weight: 500;
}
```

```
/* Progress bars */
.progress {
  border-radius: 0.35rem;
  height: 1rem;
}

.progress-bar {
  border-radius: 0.35rem;
}

/* Utilities */
.text-xs {
  font-size: 0.7rem;
}

.font-weight-bold {
  font-weight: 700;
}

.text-gray-800 {
  color: #5a5c69;
}

.text-gray-300 {
  color: #dddfeb;
}

/* Responsive adjustments */
@media (max-width: 768px) {
  .btn-group {
    width: 100%;
  }

  .btn-group .btn {
    flex: 1;
  }

  .file-card .card-footer .btn-group {
    flex-direction: column;
  }

  .file-card .card-footer .btn-group .btn {
    border-radius: 0.35rem;
    margin-bottom: 2px;
  }
}

/* Dark mode support */
@media (prefers-color-scheme: dark) {
  body {
    background-color: #1a1a1a;
    color: #e9ecef;
  }

  .card {
```

```

        background-color: #2d3748;
        color: #e9ecef;
    }

    .card-header {
        background-color: #374151;
        border-bottom-color: #4a5568;
    }

    .form-control {
        background-color: #374151;
        border-color: #4a5568;
        color: #e9ecef;
    }

    .table {
        color: #e9ecef;
    }
}

/* Animation classes */
.fade-in {
    animation: fadeIn 0.5s ease-in;
}

@keyframes fadeIn {
    from { opacity: 0; transform: translateY(20px); }
    to { opacity: 1; transform: translateY(0); }
}

.slide-up {
    animation: slideUp 0.3s ease-out;
}

@keyframes slideUp {
    from { transform: translateY(10px); opacity: 0; }
    to { transform: translateY(0); opacity: 1; }
}

/* Loading spinner */
.spinner-border-sm {
    width: 1rem;
    height: 1rem;
}

/* File type icons colors */
.text-purple {
    color: #6f42c1 !important;
}

/* Custom scrollbar */
::-webkit-scrollbar {
    width: 8px;
}

::-webkit-scrollbar-track {

```

```
        background: #f1f1f1;
        border-radius: 4px;
    }

    ::-webkit-scrollbar-thumb {
        background: #c1c1c1;
        border-radius: 4px;
    }

    ::-webkit-scrollbar-thumb:hover {
        background: #a8a8a8;
    }
}
```

This comprehensive set of HTML templates provides a complete, responsive, and user-friendly interface for the Django file manager project. The templates include:

1. **Base template** with Bootstrap navigation and responsive design
2. **Authentication templates** for login, registration, OTP verification, and profile management
3. **File management templates** for listing, uploading, viewing, and editing files
4. **Dashboard template** with statistics and quick actions
5. **Custom CSS** for enhanced styling and user experience

Each template includes proper form handling, error display, CSRF protection, responsive design, and modern UI elements using Bootstrap 5 and Font Awesome icons.