Secure File Sharing Application

Project Overview

Purpose

The Secure File Sharing Application is a Djangobased web application designed to provide a secure and userfriendly platform for file management and sharing. The primary goal is to ensure file confidentiality, integrity, and controlled access through robust encryption and authentication mechanisms.

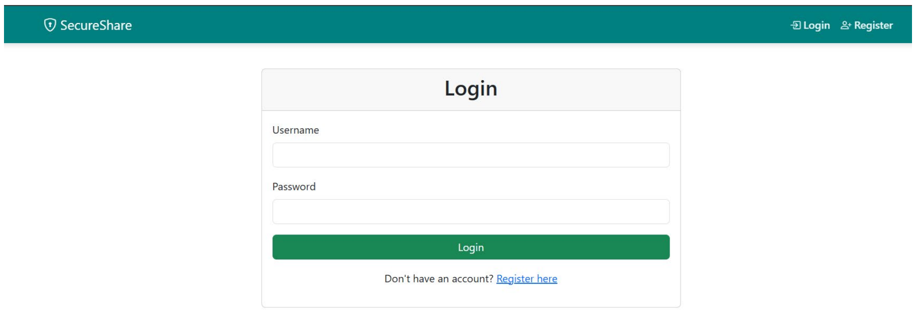
Key Objectives

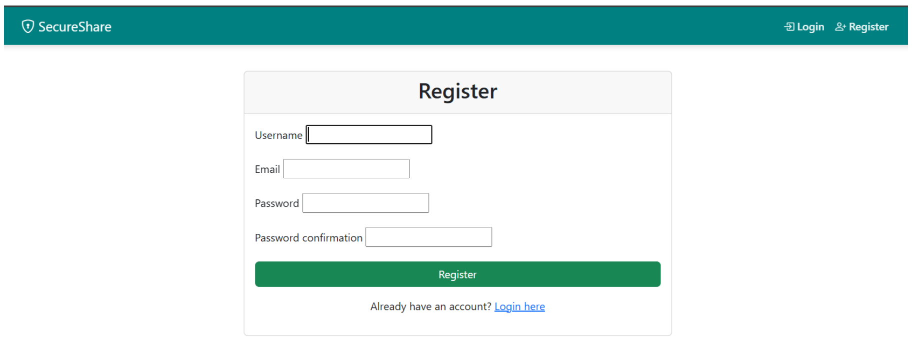
Provide secure user registration and authentication
Enable encrypted file uploads and downloads
Implement secure file sharing between users
Ensure file integrity through cryptographic hashing
Protect against common web vulnerabilities

Features

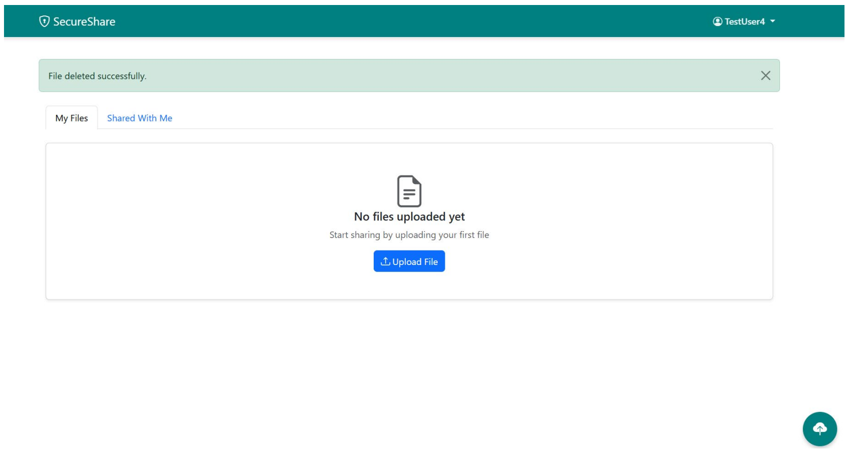
User Authentication

- Secure user registration
- Login/logout functionality
- Password protection





File Upload

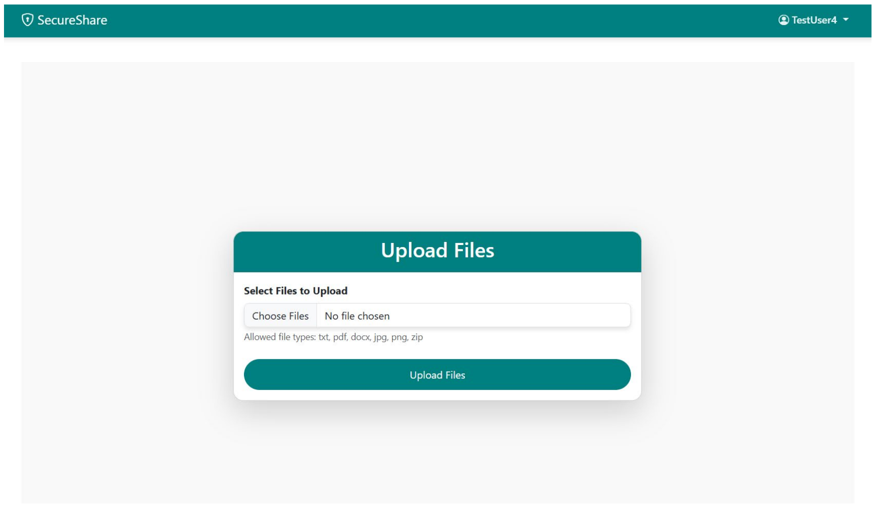


File deleted successfully.

No files uploaded yet

Start sharing by uploading your first file

- Support for multiple file types (txt, pdf, docx, jpg, png, zip)

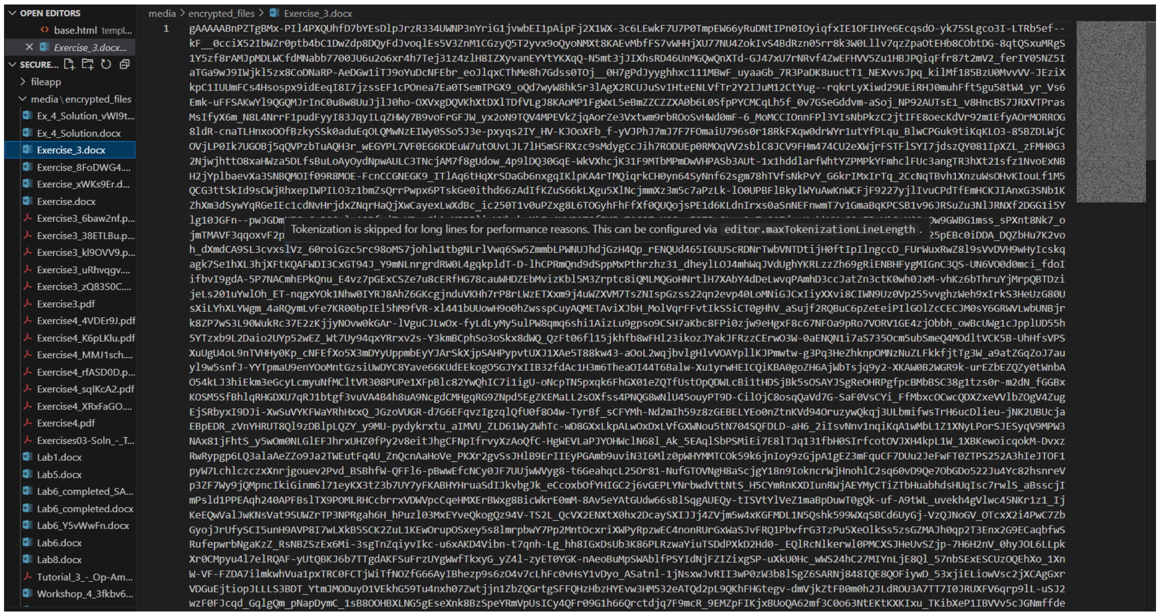


Select Files to Upload

No file chosen

Allowed file types: txt, pdf, docx, jpg, png, zip

- Automatic file encryption before storage



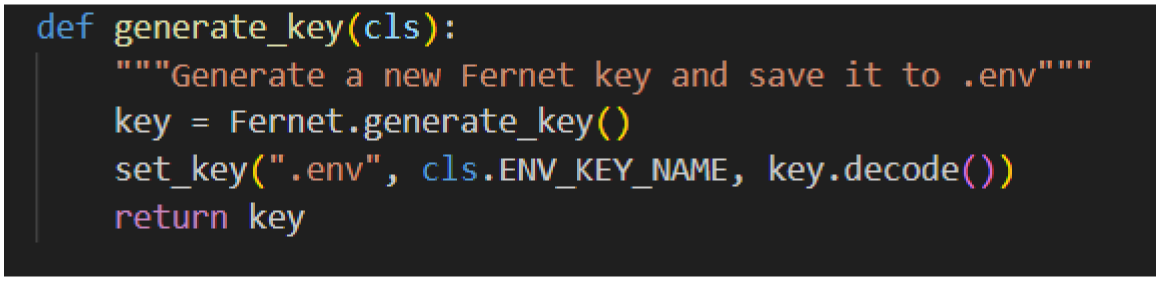
```
gAAAAABnPZTgBMx-PIl4PXQUhfD7bYEsDlpJrzR334UWNP3nYriG1jvwbEI1pAipFj2X1WX-3c6LEwkF7U7P0TmpEW66yRuDNtIPn0IOyiqfxIE1OFIHYe6EcqsdO-yk75SLgco3I-LTRb5ef--
kF__0cciX52IbWZr0ptb4bC1DwZdp8DQyFdJvoqlEs5V3ZnM1CGzyQ5T2yvx9oQyoNMXt8KAEvMbfFS7vWHHjXU77NU4ZokIvS4BdRzn05rr8k3W0Lllv7qzZpaOtEHb8CObtDG-8qtQSxuMRgS
1Y5zf8rAMJpMDLWCfdMNabb7700JU6u2o6xr4h7Tej31z4zlH8IZXyvanEYYtYKXqQ-N5mt3jJIXhsRD46UnMGQwQnXTd-GJ47xU7rNRvf4ZwEFHVV5Zu1HBJPQiqFfr87t2mV2_ferIY05NZ5I
aTGa9wJ9IWjkl5zx8CoDNaRP-AeDGw1iTJ9oYuDcNFEbr_eoJlqxCThMe8h7Gdss0TOj__0H7gPdJyyghhxc111MBwF_uyaaGb_7R3PaDK8uuctT1_NEXvvsJpq_kilMf185BzU0MvvVV-JEziX
kpC1IUUmFCs4Hsospx9idEeqI8I7jzssEF1cPOnea7Ea0TSemTPGX9_oQd7wyW8hk5r3lAgX2RCUJuSvIHteENLVfTr2Y2IJuM12CtYug--rqkrLyXiwd29UEiRHJ0muhFft5gu58tW4_yr_Vs6
Emk-uFFSAKwYl9QGQMJrInC0u8w8UuJjlJ0ho-OXVxgDQVKhXtDXlTDfVLgJ8KAoMP1FgWxL5eBmZZCZZXA0b6L0SfpPYCMCqLh5f_0v7GSeGddvm-aSoj_NP92AUTsE1_v8HncBS7JRXVTPras
MsIfyX6m_N8L4NrrF1pudFyyI83JqyILqZHWy7B9voFrGFJW_yx2oN9TQV4MPEVkZjqAorZe3Vxtwm9rbROoSvHWd0mF-6_MoMCCIOnnFPl3YIsNbPkzC2jtIFE8oecKdVr92m1EfyAOrMORROG
8ldR-cnaTLHnxoOOfBzkySSk0aduEqOLQMwNzEIWy0SSo5J3e-pxyqs2IY_HV-KJOoXFb_f-yVJPhJ7mJ7F7FOmaiU796s0r18RkFXqw0drWYr1utYfPLqu_BlwCPGuk9tiKqKLO3-85BZDLWjC
OVjLP0Ik7UGOBj5qQVPzbTuAQH3r_wEGYPL7VF0EG6KDEuW7utOUvLJL7lH5mSFRXzc9sMdygCcJih7RODUEp0RMOqVV2sblC8JCV9FHm474CU2eXWjrFSTFlSYI7jdszQY081IpXZL_zFMH0G3
2NjwjhttO8xaHWza5DLfsBuLoAyOydNpwAULC3TNcjAM7f8gUdow_4p9lDQ30GqE-WkVXhcjK31F9MTbMPmDwVHPASb3AUt-1x1hddlarfWhtYZPMPkYFmhclFUc3angTR3hXt21sfz1NvoExNB
H2jYplbaevXa3SNBQMOIf09R8MOE-FcnCCGNEGK9_ITlAq6tHqXrSDaGb6nxgqIKlpKA4rTMQiqrkCH0yn64SyNnf62sgm78hTVfsNkPvY_G6krIMxIrTq_2CcNqTBvh1XnzuWsOHvKIouLf1M5
QCG3ttSkId9sCWjRhxepIWPILO3z1bmZsQrrPwpx6PTskGe0ithd66zAdIfKZuS66kLXgu5XlNcjmmXz3m5c7aPzLk-lO0UPBFlBkylWYuAwKnWCFjF9227yjlIvuCPdTfEmHCKJIAnxG3SNb1K
ZhXm3dSywYqRGeIEc1cdNvHrjdxZNqrHaQjXwCayexLwXdBc_ic250T1v0uPZxg8L6TOGyhFhFfXf0QUQojsPE1d6KLdnIrxs0aSnNEFnwmT7v1GmaBqKPCSB1v96JRSuZu3NlJRNXf2DGG1i5Y
lg10JGFn--pwJGDm[illegible]Dw9GWBG1mss_sPXnt8Nk7_o
jmTMAVF3qqoxvF2p[illegible]25pEBc0iDDA_DQZbHu7K2vo
h_dXmdCA9SL3cvxslVz_60roiGzc5rc98oMS7johlw1tbgNLrlVwq6Sw5ZmmbLPWNUJhdjGzH4Qp_rENQUd465I6UUScRDNrTwbVNTDtijH0ftIpIlngccD_FUrWuxRwZ8l9sVvDVH9wHyIcskq
agk7Se1hXL3hjXFtKQAFWDI3CxGT94J_Y9mNLnrgrdRW0L4gqkpldT-D-lhCPRmQnd9dSppMxPthrzhz31_dheylLOJ4mhWqJVdUghYKRLzzZh69gRiENBHFygMIGnC3QS-UN6VO0d0mci_fdoI
ifbvI9gdA-5P7NACmhEPkQnu_E4vz7pGExCSZe7u8cERfHG78cauWHDZEbMvizKbl5M3Zrptc8iQMLMQGoHNrtlH7XAbY4dDeLwvqPAmhD3ccJatZn3ctK0wh0JxM-vhKz6bThruYjMrpQBTDzi
jeLs201uYwlOh_ET-nqgxYOk1Nhw0IYRJ8AhZ6GKcgjnduVKHh7rP8rLWzETXxm9j4uWZXVM7TsZNIspGzss22qn2evp40LoMNiGJCxIiyXXvi8CIWN9Uz0Vp255vvghzWeh9xIrkS3HeUzG80U
sXiLYhXLYWgm_4aRQymLvFe7KR00bpIEl5hM9fVR-xl441bUUowH9o0hZwsspCuyAQMETAviXJbH_MolVqrFFvtIkSSiCT0gHhV_aSujf2RQBuC6pZeEeiPIlGOlZcCECJM0sY6GRWVLwbUNBjr
k8ZP7wS3L90WukRc37E2zKjjyNOvw0kGAr-lVguCJLwOx-fyLdLyMy5ulPW8qmq6shi1AizLu9gpso9CSH7aKbc8FPi0zjw9eHgxF8c67NFOa9pRo7VORV1GE4zjObbh_owBcUWg1cJpplUD55h
5YTzxb9L2Daio2UYp52wEZ_Wt7Uy94qxYRrxv2s-Y3kmBCphSo3oSkx8dWQ_QzFt06fl15jkhfb8wFHl23ikozJYakJFRzzCErwO3W-0aENQN1i7aS735Ocm5ubSmeQ4MOdltVCK5B-UhHfsVPS
XuUgU4oL9nTVHHy0Kp_cNFEfXo5X3mDYyUppmbEyYJArSkXjpSAHPypvtUXJ1XAe5T88kw43-aOoL2wqjbvlgHlvVOAYpllKJPmwtw-g3Pq3HeZhknpOMNzNuZLFkkfjtTg3W_a9atZGqZoJ7au
yl9w5snfJ-YYTpmaU9enYOoMntGzsiUwDYC8Yave66KUdEEkogO5GJYxIIB32fdAc1H3m6TheaOI44T6Balw-Xu1yrwHEICQiKBA0goZH6AjWbTsjq9y2-XKAW0B2WGR9k-urEZbEZQZy0tWnbA
O54kLJ3hiEkm3eGcyLcmyuNfMCltVR308PUPe1XFpBlc82YwQhIC7i1igU-oNcpTN5pxqk6FhGX01eZQTfUstOpQDWLcBi1tHDSjBk5sOSAYJSgReOHRPgfpcBMbBSC38g1tzs0r-m2dN_fGGBx
KOSM5SfBhlqRHGDXU7qRJ1btgf3vuVA4B4h8uA9NcgdCMHgqRG9ZNpd5EgZKEMaLL2sOXfss4PNQG8wNlU45ouyPT9D-CilOjC8osqQaVd7G-SaF0VsCYi_FfMbxcOCwcQDXZxeVVlbZOgV4Zug
EjSRbyxI9DJi-XwSuVYKFWaYRhHxxQ_JGzoVUGR-d7G6EFqvzIgzqlQfU0f8O4w-TyrBf_sCFYMh-Nd2mIh59z8zGEBELYEo0nZtnKVd94OruzywQkqj3ULbmifwsTrH6ucDlieu-jNK2UBUcja
EBpEDR_zVnYHRUT8Ql9zDBlpLQZY_y9MU-pydykrxtu_aIMVU_ZLD61Wy2WhTc-wD8GXxLkpALwOxDxLVfGXWNou5tN704SQFDLD-aH6_2iIsvNnv1nqiKqA1wMbL1Z1XNyLPorSJESyqV9MPW3
NAx81jFhtS_y5wOm0NLGlEFJhrxUHZ0fPy2v8eitJhgCFNpIfrvyXzAoQfC-HgWEVLaPJYOHWclN68l_Ak_5EAqlSbPSMiEi7E8lTJq131fbH0SIrfcotOVJXH4kpL1W_1XBKewoicqokM-Dvxz
RwRypgp6LQ3alaAeZZo9Ja2TWEutFq4U_ZnQcnAaHoVe_PKXr2gvSsJHlB9ErIIEyPGAmb9uviN3I6Mlz0pWHYMMTCOk59k6jnIoy9zGjpA1gEZ3mFquCF7DUu2JeFwFT0ZTPS252A3hIeJTOF1
pyW7LchlczczxXnrjgouev2Pvd_BSBhfW-QFFl6-pBwwEfcNCy0JF7UUjwWVyg8-t6GeahqcL25Or81-NufGTOVNgH8aScjgY18n9IokncrWjHnohlC2sq60vD9Qe7ObGDo522Ju4Yc82hsnreV
p3ZF7Wy9jQMpncIkiGinm6l71eyKX3tZ3b7UY7yFKABHYHruaSdIJkvbgJk_eCcoxbOfYHIGC2j6vGEPLYNrbwdVttNtS_H5CYmRnKXDIunRWjAEYMyCTiZTbHuabhdsHUqIsc7rwlS_aBsscjI
mPsld1PPEAqh240APFBslTX9POMLRHCcbrrxVDWVpcCqeHMXErBWxg8BicWkrE0mM-8Av5eYAtGUdw66sBlSqgAUEQy-tISVtYlVeZ1maBpDuwT0gQk-uf-A9tWL_uvekh4gVlwc45NKr1z1_Ij
KeEQwValJwKNsVat9SUWZrTP3NPRgah6H_hPuzl03MxEYveQkogQz94V-TS2L_QcVX2ENXtX0hx2DcaySXIJJj4ZVjm5w4xKGFMDL1N5Qshk599WXqSBCd6UyGj-VzQJNoGV_OTcxX2i4PwC7Zb
GyojJrUfySCI5unH9AVP8I7wLXkB5SCK2ZuL1KEwOrupOSxey5s8lmrpbwY7Pp2MntOcxriXWPyRpzwEC4nonRUrGxWaSJvFRQ1PbvfrG3TzPu5XeOlkSs5zsGZMAJh0qp2T3Enx2G9ECaqbfwS
RufepwrbNgaKzZ_RsNBZSzEx6Mi-3sgTnZqiyvIkc-u6xAKD4Vibn-t7qnh-Lg_hh8IGxDsUb3K86PLRzwaYiuTSDdPXkD2Hd0-_EQlRcNlkerwl0PMCXSJHeUvSZjp-7H6H2nV_0hyJOL6LLpk
Xr0CMpyu4l7elRQAF-yUtQBKJ6b7TTgdAKFSuFrzUYgWwfTkxyG_yZ4l-zyET0YGK-nAeoBuMpSWAblfPSYIdNjFZIZixgSP-uXkU0Hc_wWS24hC27MIYnLjE8Ql_57nbSExESCUzOQEhXo_1Xn
W-VF-FZDA7ilmkwhVua1pxTRC0FCTjWiTfNOZfG66AyIBhezp9s6zO4v7cLhFc0vHsY1vDyo_ASatnl-1jNsxwJvRII3wP0zW3b8lSgZ6SARNj848IQE8QOFiywD_53xjiELiowVsc2jXCAgGxr
VDGuEjtiopJLLLS3BDT_YtmJMODuyD1VEkhG59Tu4nxh07Zwtjjn1ZbZQGrtgSFFQHzHbzHYEvw3HM532eATQd2pL9QKhFHGtegv-dmVjkZtFB0m0h2JLdROU3A7TT7I0JRUXFV6qrp9lL-uSJ2
wzF0FJcqd_GqlgQm_pNapDymC_1sB8OOHBXLNG5gEseXnk8BzSpeYRmVpUsICy4QFr09G1h66Qrctdjq7F9mcR_9EMZpFIKjxBUoQA62mf3C0o63NtEKtKXKIxu_TKibXeP1IBVVv5cJGNmffde
```

Tokenization is skipped for long lines for performance reasons. This can be configured via `editor.maxTokenizationLineLength`.

- File type validation

File Encryption

- Uses Fernet symmetric encryption

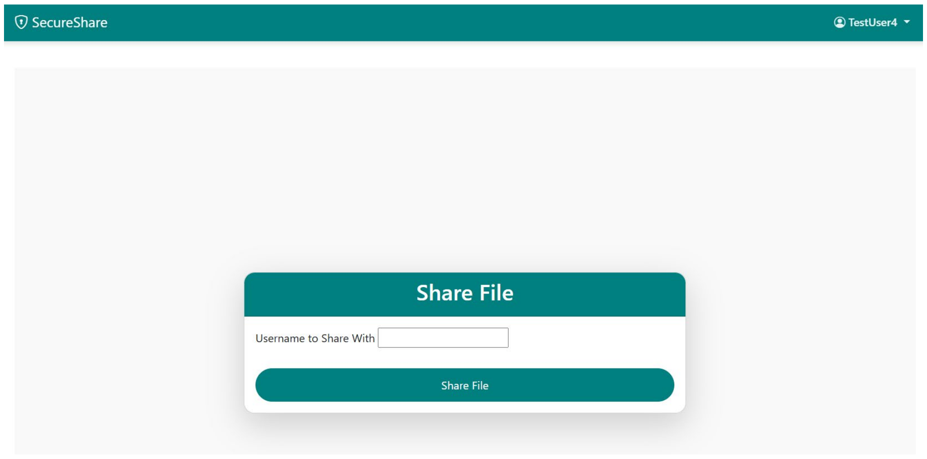


```python
def generate_key(cls):
    """Generate a new Fernet key and save it to .env"""
    key = Fernet.generate_key()
    set_key(".env", cls.ENV_KEY_NAME, key.decode())
    return key
```

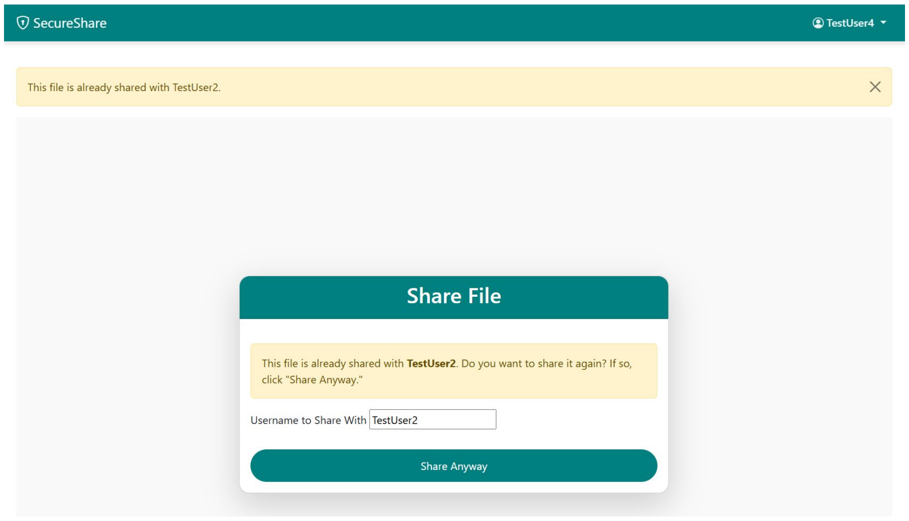
- Generates and manages encryption keys
- Encrypts files before storage
- Decrypts files only during download

File Sharing

- Share files with specific users



- Track file sharing history


SecureShare

TestUser4

This file is already shared with TestUser2.

Share File

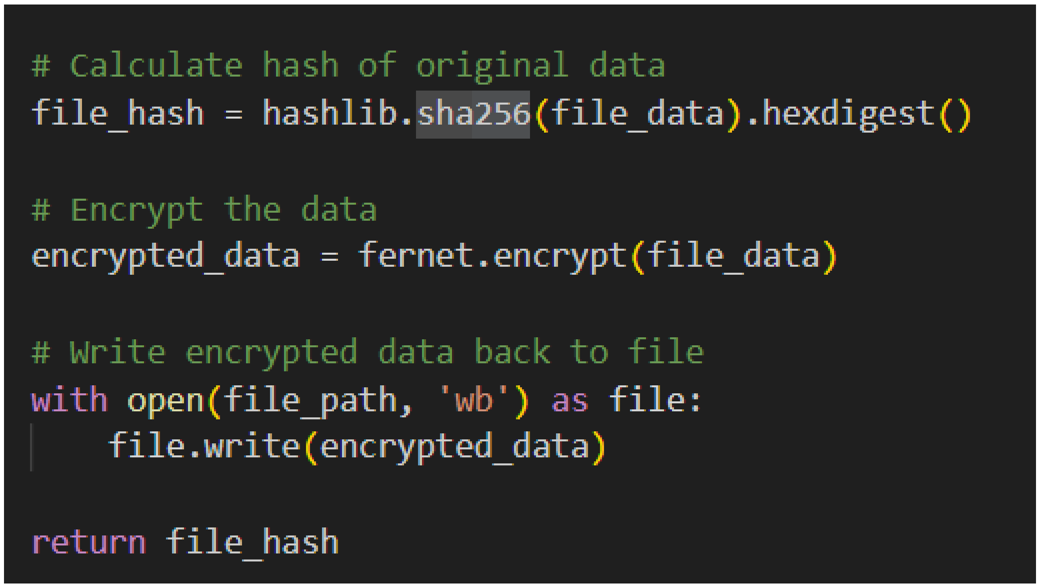
This file is already shared with **TestUser2**. Do you want to share it again? If so, click "Share Anyway."

Username to Share With: TestUser2

- Prevent unauthorized access

File Integrity Checks

- SHA256 hash generation



```
# Calculate hash of original data
file_hash = hashlib.sha256(file_data).hexdigest()

# Encrypt the data
encrypted_data = fernet.encrypt(file_data)

# Write encrypted data back to file
with open(file_path, 'wb') as file:
    file.write(encrypted_data)

return file_hash
```

- Verification during file upload and download
- Detects file tampering

Installation

Software Requirements

Python 3.8+
Django 3.2+
pip package manager

Required Dependencies

1. Django
2. python-dotenv
3. cryptography

Installation Dependencies

Run the following command to install dependencies:

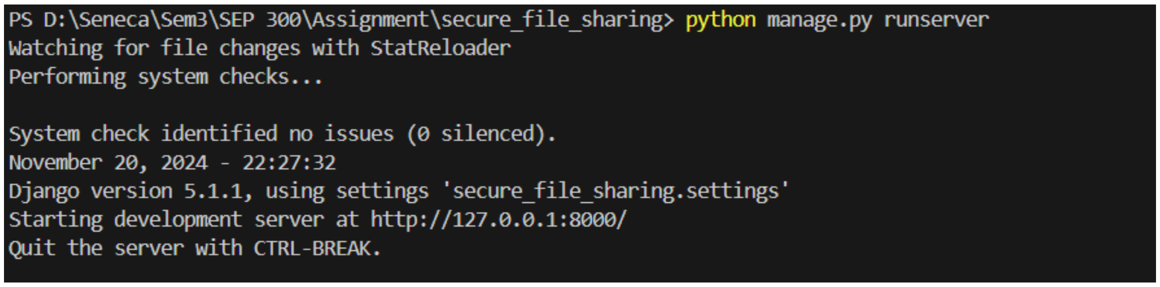
```
pip install django  cryptography python-dotenv
```

Run Development Server

To run the server, use the following command:

```
python manage.py runserver
```

After running the command, you should see output similar to this:



```
PS D:\Seneca\Sem3\SEP 300\Assignment\secure_file_sharing> python manage.py runserver
Watching for file changes with StatReloader
Performing system checks...

System check identified no issues (0 silenced).
November 20, 2024 - 22:27:32
Django version 5.1.1, using settings 'secure_file_sharing.settings'
Starting development server at http://127.0.0.1:8000/
Quit the server with CTRL-BREAK.
```

Edge Cases – Demo Videos for all different cases are uploaded on the repository.