Arquivo recebido: projeto\_software\_quantico.zip. Agora vou extrair o conteúdo e analisar os arquivos principais para entender do que se trata o sistema de contabilidade.

Aguarde um momento enquanto faço a extração.

O arquivo ZIP contém um sistema desenvolvido em **Python com Flask**, provavelmente voltado à gestão contábil. Aqui estão os principais componentes encontrados:

**🧩 Estrutura do Projeto:**

* **app.py**: Arquivo principal para iniciar a aplicação Flask.
* **config.py**: Configurações da aplicação (como conexão com banco de dados, chaves secretas, etc.).
* **requirements.txt**: Lista de bibliotecas Python necessárias.
* **README.md**: Provavelmente traz instruções de uso.
* **todo.md**: Tarefas pendentes ou planejadas para o sistema.

**📂 Diretório src/:**

* **templates/**: Contém páginas HTML (base, login, cadastro, dashboard...).
* **routes/**: Código com rotas (como usuarios.py, tarefas.py, main.py).
* **models/**: Modelos de dados, como usuario.py.

**🧪 Virtual Environment:**

A pasta venv/ indica que o projeto já vem com um ambiente virtual Python incluso.

**✅ Próximos Passos**

Posso te ajudar a:

* Executar o sistema localmente no seu computador.
* Fazer modificações ou integrações.
* Explicar a estrutura e funcionamento do código.
* Documentar ou melhorar o sistema.

Você quer **rodar o sistema localmente** agora ou prefere **analisar e entender seu funcionamento primeiro**?

4o

**Você disse:**

gostaria de rodar localmente para teste

**O ChatGPT disse:**

Ótimo! Vou te guiar passo a passo para **rodar o sistema localmente** no seu computador Windows.

**✅ Pré-requisitos**

Antes de tudo, verifique se você já tem instalado:

1. **Python 3.11** (ou compatível com o projeto)
2. **Git Bash** ou **Prompt de Comando** (CMD)
3. **Visual Studio Code** (opcional, mas recomendado)

Se não tem o Python instalado ainda, posso te ajudar com isso também.

**🛠️ Passo a Passo para Rodar o Sistema**

**1. Extraia o arquivo ZIP**

* Extraia o projeto\_software\_quantico.zip para uma pasta, por exemplo:  
  C:\Users\SeuNome\projeto\_software\_quantico

**2. Abra o terminal na pasta do projeto**

* No Windows, vá até a pasta extraída.
* Segure Shift + clique com o botão direito e escolha **"Abrir janela do PowerShell aqui"** ou **"Abrir no Terminal"**.

**3. Crie um ambiente virtual (recomendado)**

Se o venv/ já estiver presente e funcional, você pode pular esta etapa, mas se quiser criar do zero:

bash

CopiarEditar

python -m venv venv

**4. Ative o ambiente virtual**

bash

CopiarEditar

.\venv\Scripts\activate

Você verá algo como (venv) no começo da linha.

**5. Instale as dependências**

Se o ambiente virtual não funcionar ou preferir garantir a instalação limpa:

bash

CopiarEditar

pip install -r requirements.txt

**6. Rode a aplicação**

bash

CopiarEditar

python app.py

Se tudo der certo, você verá algo como:

csharp

CopiarEditar

\* Running on http://127.0.0.1:5000/ (Press CTRL+C to quit)

Abra o navegador e acesse:  
👉 <http://127.0.0.1:5000>

Se ocorrer algum erro, me envie a mensagem exata aqui que te ajudo a resolver.  
Você quer que eu te diga o que o sistema faz (funcionalidades)?

4o

**Você disse:**

PS C:\projeto1> python app.py >> Traceback (most recent call last): File "C:\projeto1\app.py", line 1, in <module> from src import create\_app File "C:\projeto1\src\\_\_init\_\_.py", line 2, in <module> from flask\_sqlalchemy import SQLAlchemy ModuleNotFoundError: No module named 'flask\_sqlalchemy'

**O ChatGPT disse:**

O erro indica que **o pacote flask\_sqlalchemy não está instalado** no seu ambiente Python. Isso é comum quando o ambiente virtual não está ativado corretamente ou os pacotes ainda não foram instalados.

**✅ Como resolver**

**1. Ative o ambiente virtual**

No terminal, digite:

bash

CopiarEditar

.\venv\Scripts\activate

Se não der certo ou a pasta venv não existir, crie novamente com:

bash

CopiarEditar

python -m venv venv

.\venv\Scripts\activate

**2. Instale as dependências**

Depois de ativar, rode:

bash

CopiarEditar

pip install -r requirements.txt

Isso vai instalar todos os pacotes necessários, incluindo o flask\_sqlalchemy.

Se quiser instalar apenas o que está faltando rapidamente, rode:

bash

CopiarEditar

pip install flask\_sqlalchemy

**3. Rode novamente a aplicação**

bash

CopiarEditar

python app.py

Se aparecer outro erro, cole aqui que eu continuo te ajudando. Deseja que eu revise o requirements.txt para garantir que ele tem tudo necessário?

4o

**Você disse:**

.\venv\Scripts\activate : O arquivo C:\projeto1\venv\Scripts\Activate.ps1 não pode ser carregado porque a execução de scripts foi desabilitada neste sistema. Para obter mais informações, consulte about\_Execution\_Policies em https://go.microsoft.com/fwlink/?LinkID=135170. No linha:2 caractere:1 + .\venv\Scripts\activate + ~~~~~~~~~~~~~~~~~~~~~~~ + CategoryInfo : ErrodeSegurança: (:) [], PSSecurityException + FullyQualifiedErrorId : UnauthorizedAccess

**O ChatGPT disse:**

Esse erro acontece porque o PowerShell está com a **política de execução de scripts desabilitada** por padrão, o que impede a ativação do ambiente virtual do Python.

**✅ Como resolver o erro “execution of scripts is disabled”**

Você tem duas opções simples:

**🟢 Opção 1: Use o cmd (Prompt de Comando) em vez do PowerShell**

1. Na pasta C:\projeto1, segure Shift e clique com o botão direito.
2. Selecione **“Abrir janela do comando aqui”**.
3. No terminal que abrir, digite:

cmd

CopiarEditar

venv\Scripts\activate.bat

Isso vai ativar normalmente o ambiente virtual.

**🔵 Opção 2: Habilitar scripts no PowerShell (com cuidado)**

**Só use se você entender os riscos.** Você pode permitir temporariamente a execução com este comando (abra o PowerShell como administrador):

powershell

CopiarEditar

Set-ExecutionPolicy -Scope Process -ExecutionPolicy Bypass

Depois disso, tente novamente:

powershell

CopiarEditar

.\venv\Scripts\activate

**✅ Em seguida:**

Depois que o ambiente estiver ativado, instale os pacotes com:

bash

CopiarEditar

pip install -r requirements.txt

E rode o sistema:

bash

CopiarEditar

python app.py

Quer que eu verifique se o requirements.txt está completo com os pacotes necessários?

4o

**Você disse:**

(venv) C:\projeto1> (venv) C:\projeto1>python app.py \* Serving Flask app 'src' \* Debug mode: on WARNING: This is a development server. Do not use it in a production deployment. Use a production WSGI server instead. \* Running on all addresses (0.0.0.0) \* Running on http://127.0.0.1:5000 \* Running on http://192.168.100.32:5000 Press CTRL+C to quit \* Restarting with stat \* Debugger is active! \* Debugger PIN: 105-830-703

**O ChatGPT disse:**

Perfeito! 🎉 Seu sistema Flask está rodando com sucesso em ambiente local!

**✅ Agora você pode acessar:**

* <http://127.0.0.1:5000> (local)
* Ou <http://192.168.100.32:5000> (outros dispositivos da sua rede, como celular)

Se ao acessar aparecer algum erro ou página em branco, posso te ajudar a corrigir.  
Quer que eu te explique o que esse sistema faz (com base no código)?

4o

**Você disse:**

Login Email Senha Lembrar-me Não tem uma conta? Registre-se aqui nao aparece as caixas para inserir os dados

**O ChatGPT disse:**

Ótimo que a tela de **login** apareceu — agora, o problema parece estar relacionado com o **formulário HTML não exibindo os campos de entrada (inputs)** para e-mail e senha.

Esse tipo de erro geralmente ocorre por três motivos principais:

**🔍 Causas possíveis**

**1. ❌ HTML mal formatado no template (login.html)**

Os campos <input> podem estar ausentes ou com erro de sintaxe.

**2. 🧱 Falha no carregamento do CSS/JS**

Se o projeto usa Bootstrap ou outro framework e os arquivos não estão sendo carregados corretamente, o layout pode quebrar.

**3. 🔗 Erros na extensão do template base (base.html)**

Se o login.html depende de base.html, e essa base está mal formatada, o formulário pode nem aparecer.

**✅ Vamos verificar agora**

Vou abrir e analisar o conteúdo do arquivo src/templates/login.html.

O arquivo login.html está corretamente estruturado e espera que um **objeto form seja passado para o template**, usando Flask-WTF (Formulários com validação).

⚠️ **Mas como o formulário é exibido com {{ form.email(...) }}, se o form não for passado, nada aparece.**

**✅ Diagnóstico: o template depende de um objeto form, mas ele não está sendo enviado para o HTML na rota.**

**💡 Como corrigir**

Precisamos verificar o arquivo usuarios.py (em src/routes/usuarios.py) para ver se ele está criando e passando o formulário para o template.

Vou verificar esse arquivo agora.

Achei o problema! 👇

Na rota de login (@usuarios\_bp.route("/login")), **o formulário não está sendo criado nem passado para o template**.  
Mas o login.html exige um objeto form, senão os campos não aparecem.

**✅ Correção recomendada**

Você precisa criar um formulário com Flask-WTF e passá-lo no render\_template.

**1. Verifique se existe um arquivo forms.py (geralmente em src/forms.py).**

Se existir, ótimo — vamos usá-lo.

Se **não existir**, você precisa criar uma classe de formulário. Aqui está um exemplo funcional mínimo:

**🔧 Crie ou edite src/forms.py:**

python

CopiarEditar

from flask\_wtf import FlaskForm

from wtforms import StringField, PasswordField, BooleanField, SubmitField

from wtforms.validators import DataRequired, Email

class LoginForm(FlaskForm):

email = StringField('Email', validators=[DataRequired(), Email()])

senha = PasswordField('Senha', validators=[DataRequired()])

lembrar = BooleanField('Lembrar-me')

submit = SubmitField('Entrar')

**2. No usuarios.py, importe o formulário e use ele:**

Adicione no topo do arquivo:

python

CopiarEditar

from src.forms import LoginForm

E edite a função login() assim:

python

CopiarEditar

@usuarios\_bp.route("/login", methods=["GET", "POST"])

def login():

if current\_user.is\_authenticated:

return redirect(url\_for("main.index"))

form = LoginForm()

if form.validate\_on\_submit():

user = Usuario.query.filter\_by(email=form.email.data).first()

if user and check\_password\_hash(user.senha, form.senha.data):

login\_user(user, remember=form.lembrar.data)

return redirect(url\_for("main.index"))

flash("Email ou senha incorretos", "danger")

return render\_template("login.html", form=form)

Depois de salvar essas alterações, reinicie o app (python app.py) e recarregue a página.  
As caixas de e-mail e senha devem aparecer corretamente.

Quer que eu verifique se o forms.py já existe no seu projeto