



Instituição: UniCeub
Participantes do Grupo:

Ricardo Ribeiro Reis
Rafael Coura
Guilherme Santiago

Introdução

Este documento detalha o projeto da "Biblioteca de Alexandria", uma página web para uma instituição de ensino, no contexto da disciplina de Programação para Web. Serão abordados o código, a preparação do ambiente, um guia de instalação de dependências e a execução da aplicação. Adicionalmente, será fornecido um guia de testes, incluindo orientações para popular o banco de dados com dados iniciais e um roteiro para testes funcionais

Criação do Ambiente Virtual: A IDE utilizada é o VSCode (Visual Studio Code). Para isolar as dependências (bibliotecas e suas versões) e evitar conflitos entre projetos, é essencial criar um ambiente virtual (ver Figura 1).

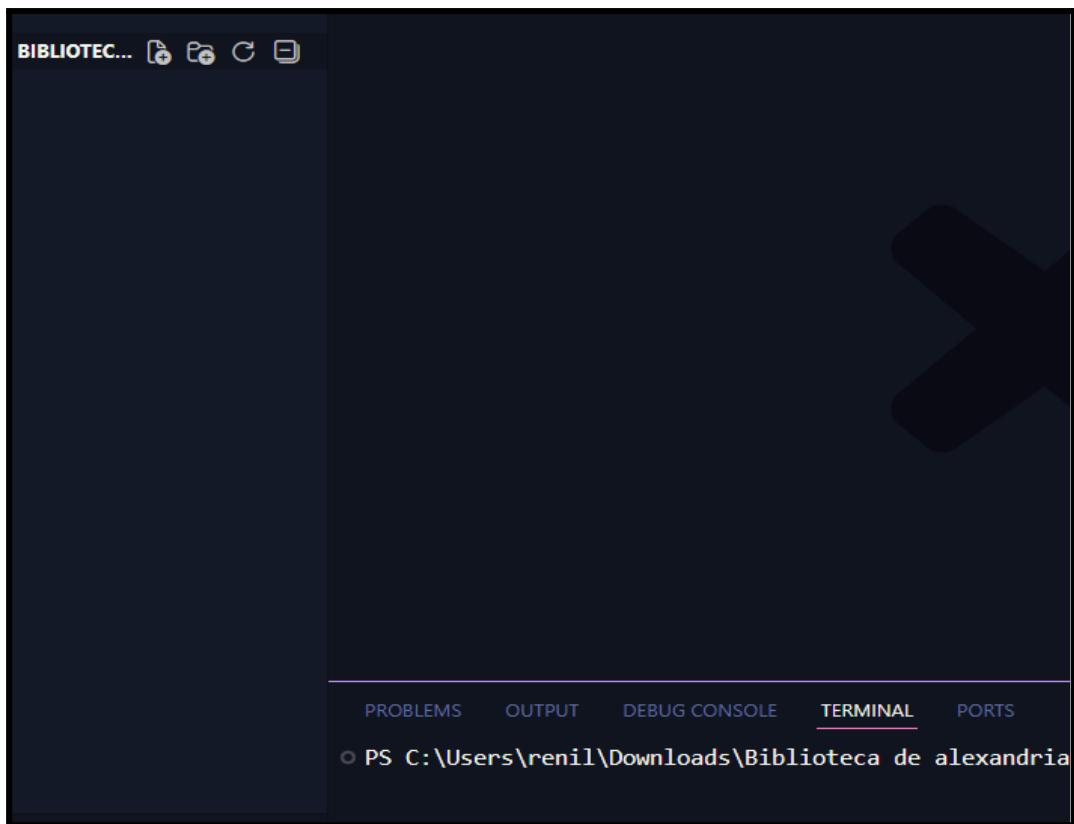


Figura 1

Comando: `python -m venv (nome desejado da pasta)`- Ativação do Ambiente Virtual

Após a criação, a estrutura de pastas do projeto ficará conforme a Figura 2. Para ativar o ambiente, use o seguinte comando:

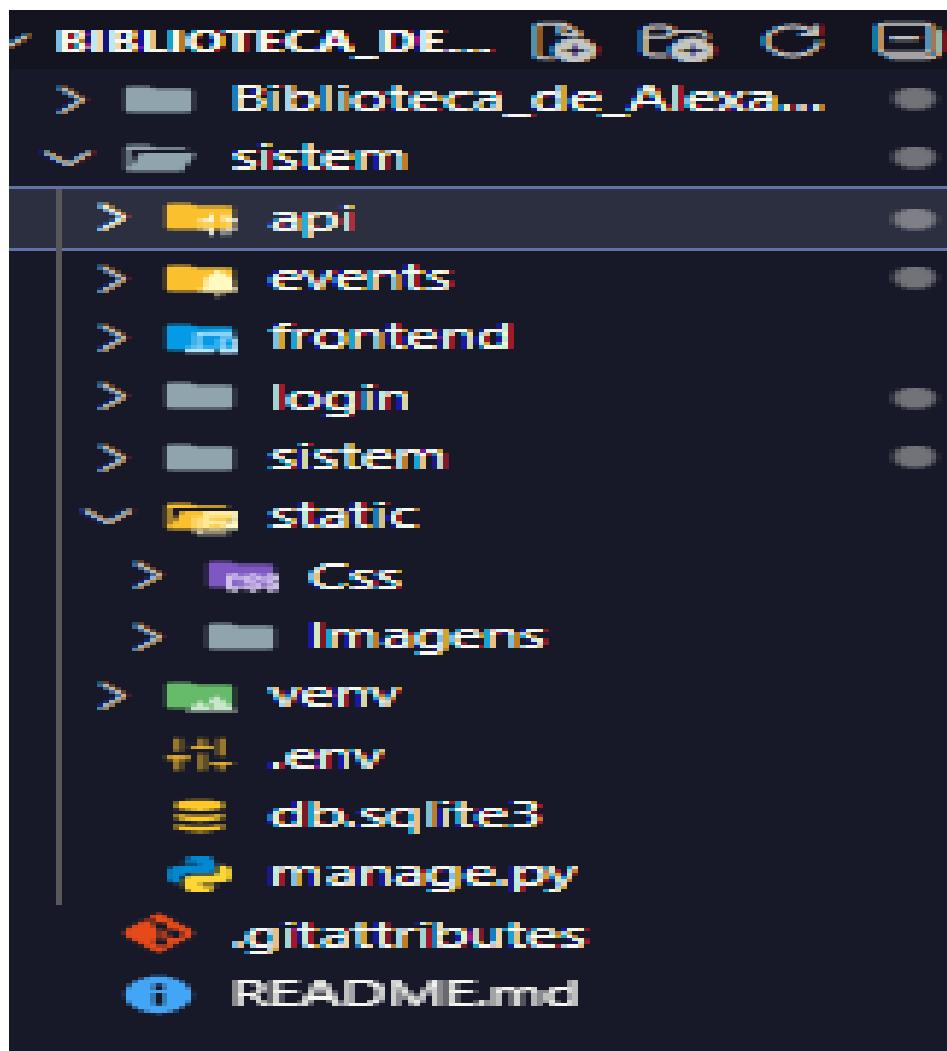


Figura 2

Comando: `(nome da pasta)\Scripts\activate`- Instalação de Dependências

Para que o projeto funcione corretamente, as seguintes bibliotecas devem ser instaladas:

- **Django (5.2.5):** `pip install django`
- **Django REST Framework (3.16.1):** `pip install djangorestframework`
- **Gerenciamento e Compartilhamento de Banco de Dados:**
 - `pip install python-dotenv psycopg2` foi usado para manter a unicidade de dados referentes ao banco de dados para assim evitar inconsistência em informações sensíveis como senhas e as chaves de API. Tendo assim um banco de dados externo armazenando os dados (**nosso trabalho usou o banco de dados estruturado no neon, por isso o sqlite3 apresentará**

- inconsistências)**
- `pip install psycopg2-binary` essa biblioteca serve para ser um adaptador do banco de dados PostgreSQL para Python por meio de uma interface que conecte ambos, permitindo comandos SQL

Inicialização da Aplicação

Após instalar todas as dependências (Seção 2.3), siga os passos para rodar o site:

1. **Baixar o Projeto:** Faça o download e extraia o arquivo do repositório https://github.com/Devrichas/Biblioteca_de_Alexandria na mesma pasta onde se encontra o ambiente virtual.
2. **Navegar até a Pasta:**
 - `cd (Nome da pasta)`
 - `cd (Biblioteca_de_Alexandria)`
 - `cd (code)`
3. **Aplicar Migrações:** `python manage.py migrate`
4. **Iniciar o Servidor:** Use o comando `python manage.py runserver`. Uma saída no terminal, semelhante à Figura 3, será gerada.

```
System check identified no issues (0 silenced).
December 07, 2025 - 19:04:03
Watching for file changes with StatReloader
Performing system checks...

System check identified no issues (0 silenced).
December 07, 2025 - 19:04:03

System check identified no issues (0 silenced).
December 07, 2025 - 19:04:03
System check identified no issues (0 silenced).
December 07, 2025 - 19:04:03
December 07, 2025 - 19:04:03
Django version 6.0, using settings 'sistem.settings'
Starting development server at http://127.0.0.1:8000/
Quit the server with CTRL-BREAK.

WARNING: This is a development server. Do not use it in a production setting. Use a production WSGI or ASGI server instead.
For more information on production servers see: https://docs.djangoproject.com/en/6.0/howto/deployment/
[07/Dec/2025 19:11:52] "GET / HTTP/1.1" 200 4856
```

Para acessar o site, use **Ctrl+clique** no link [\[http://127.0.0.1:8000/\]\(http://127.0.0.1:8000/\)](http://127.0.0.1:8000/) disponibilizado pelo terminal. Para finalizar a execução, pressione **Ctrl+c** no terminal.

Guia De Preenchimento:

Para popular o banco de dados é preciso primeiro que o usuário adicione valores válidos para usuário e evento, como:

Usuário

- nome (sem caracteres especiais ou números)
- telefone(tipo char, 15 caracteres)
- instituição de ensino é um campo não nulo
- senha sendo não nula, com limite de 8 caracteres com pelo menos 1 número e um caractere especial
- email sendo unique (só um mesmo email por usuário) e precisando ter "@gmail.com"
- token_acesso (**ORG123 é o token necessário para logar como organizador**) tendo blank e Null igual a true, pois só serão preenchidos conforme o tipo de usuário (organizador).
- tipo de usuário é definido como Char e por padrão como estudante, mas podem ser atribuídos como professores e organizadores
- o id do usuário é gerado automaticamente

Evento:

- Nome do evento não pode ser nulo
- tipoEvento tipo text
- dataIni leva em consideração uma data que será setada com dd/mm/aaaa
- dataFin leva em consideração uma data que será setada com dd/mm/aaaa
- local leva em consideração uma variável text
- quantPart usa um campo integer que tem como validação de valor mínimo 0
- Organizador que usa a chave estrangeira de usuario para pegar o token do

organizador

-vagas usa um modelo integer para não nulo com o valor minimo sendo 1

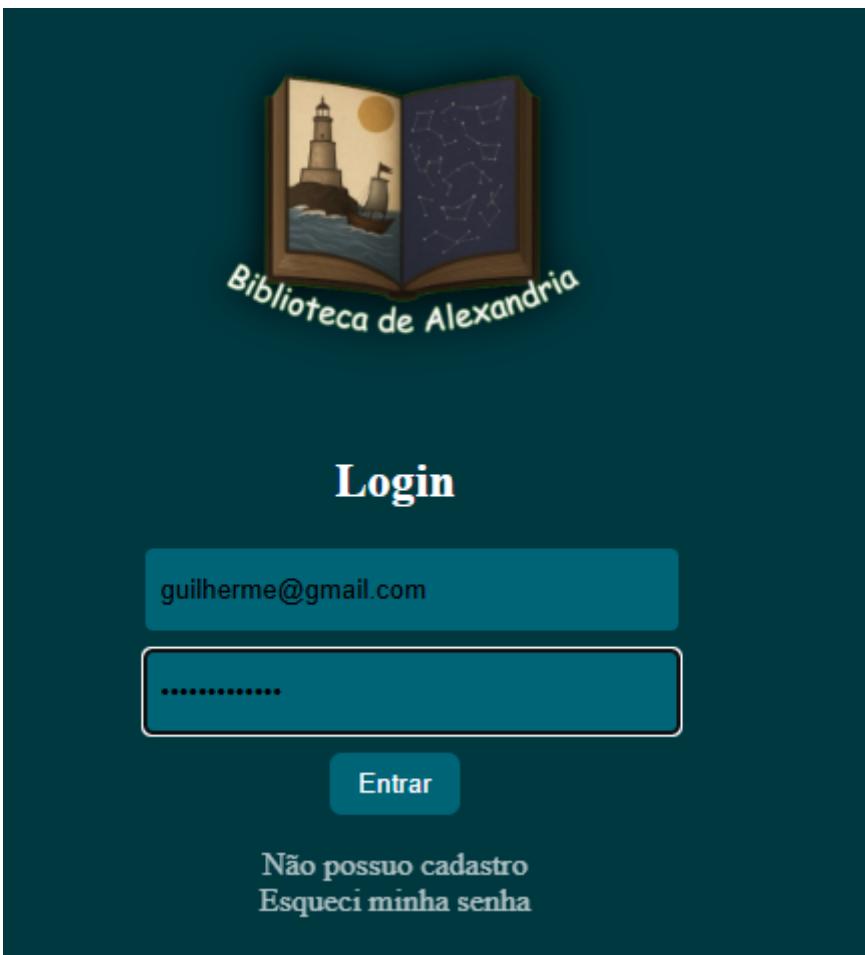
-Certificados usa um operador booleano para determinar se foi enviado ou não o certificado

Com esses dados já se pode preencher o banco de dados para receber as saídas correspondentes,

Visão Geral:

A Biblioteca de Alexandria é uma plataforma de gerenciamento de eventos. Suas funcionalidades incluem:

- Cadastro de usuários (alunos, organizador e professor).
 - Inscrição de usuários em eventos.
 - Controle das vagas disponíveis nos eventos.
 - Criação de eventos e seus respectivos tipos.
- segue alguns exemplos:



Perfil

Guilherme Santiago

(85) 56784-5124

Ceub

guilherme@gmail.com

.....

Mostrar

Salvar perfil

Cadastro de Eventos

Banner do Evento:

[Escolher ficheiro](#) [Gojoart.jpg](#)

Nome do Evento:

Java Concorrente

Tipo do Evento:

Palestra



Data de início:

09/12/2025



Data de encerramento:

12/12/2025



Duração do Evento(horas):

10

Horario de inicio:

11:00



Horario de encerramento:

15:00



Local:

Ceub

Vagas:

15



[Enviar](#)

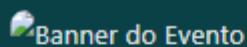
Java Concorrente Palestra 09/12 12/12 11:00 15:00 10 Ceub 0 15 Guilherme Santiago [Editar Evento](#) [Excluir](#)

Eventos em que Rafael Couraeel está inscrito

Nome	Tipo	Data de inicio	Data de encerramento	Hora de inicio	Hora de termino	Duracao	Local	Vagas
------	------	----------------	----------------------	----------------	-----------------	---------	-------	-------

Aula Web	Ensino presencial	08/12	08/12	06:13	08:13	7	Ceub	2
----------	-------------------	-------	-------	-------	-------	---	------	---

Inscrição no Evento



Programação Python -

Tipo do Evento: Ensino presencial

Local: Taguatinga

Data: 08/12 até 08/12

Horário: 06:00h às 08:00h

Vagas disponíveis: 30

[Inscrirever-se](#)



Certificados Disponíveis

Evento	Data de Encerramento	Duração (horas)	Certificado
Aula Web	08/12/2025	7	Ver

Eventos Finalizados – Emitir Certificado

Nenhum evento finalizado para emitir certificado.

Regras de Negócio:

Cadastro de eventos:

A Biblioteca de Alexandria no cadastro de ventos pede pela data de início do evento, data de fim, quantidade de horas, vagas, local, horário de início, horário de encerramento, nome do evento, tipo do evento e responsável pelo evento.

Respeitando a regra de negócio imposta, eventos não podem ser criados antes da data atual e aqui teremos demonstrações da aplicação na figura 4

Cadastro de Eventos

Banner do Evento:	<input type="button" value="Escolher ficheiro Gogoart.jpg"/>
Nome do Evento:	Java Concorrente
Tipo do Evento:	Palestra
Data de início:	09/12/2025
Data de encerramento:	12/12/2025
Duração do Evento(horas):	10
Horario de inicio:	11:00
Horario de encerramento:	15:00
Local:	Ceub
Vagas:	15
<input type="button" value="Enviar"/>	

Figura 4

Aqui segue o exemplo de um evento a ser cadastrado, tendo todas as suas informações devidamente preenchidas e a saída dos dados após a verificação na figura 5:

Java Concorrente	Palestra	09/12	12/12	11:00	15:00	10	Ceub	0	15	Guilherme Santiago	Editar Evento	Excluir
------------------	----------	-------	-------	-------	-------	----	------	---	----	--------------------	-------------------------------	-------------------------

Figura 5

Agora aqui segue a verificação feita quando a data de início do evento é menor do que a data atual na figura 6, 7 e 8

Cadastro de Eventos

Banner do Evento:	<input type="button" value="Escolher ficheiro"/> GoatTodo.jpg
Nome do Evento:	Arquitetura de Hardware
Tipo do Evento:	Ensino a distancia
Data de início:	05/12/2025 <input type="button" value=""/>
Data de encerramento:	19/12/2025 <input type="button" value=""/>
Duração do Evento(horas):	2
Horario de inicio:	21 : 01 <input type="button" value=""/>
Horario de encerramento:	22 : 15 <input type="button" value=""/>
Local:	Remoto
Vagas:	10
<input type="button" value="Enviar"/>	

Figura 6

Data de início:	05/12/2025 <input type="button" value=""/>
Data de encerramento:	19/12/2025 <input type="button" value=""/>

Figura 7

A data inicial não pode ser anterior à data atual

Figura 8

Como pode ser observado, datas anteriores a data atual sendo usadas como data de inicio do evento não são permitidas. Agora será mostrado os casos de mesma data e horários diferentes e data de fim menor que data de inicio

nas figuras 9, 10, 11 (mesma data e horários diferentes) e nas figuras 12 e 13 (data de fim menor que a data inicial)

Data de início:	08/12/2025
Data de encerramento:	08/12/2025

Figura 9

Horario de inicio:	21 : 01
Horario de encerramento:	22 : 15

Figura 10

Arquitetura de Hardware	Ensino a distancia	08/12	08/12	21:01	22:15	2	Remoto	0	10	Guilherme Santiago	Editor	Excluir
-------------------------	--------------------	-------	-------	-------	-------	---	--------	---	----	--------------------	------------------------	-------------------------

Figura 11

Como pode ser visto, o nosso sistema verifica a data e permite a criação de um evento no mesmo dia, mas com horários de início e encerramento válidos

Nome do Evento:	Inteligencia Artificial
Tipo do Evento:	Selecione um evento
Data de início:	19/12/2025
Data de encerramento:	15/12/2025

Figura 12

A data inicial não pode ser depois da data final.

Figura 13

Como demonstrado, datas de encerramento não são válidas ao serem determinadas antes da data de ínicio (**Obs: Devido ao servidor de Banco de Dados externo o fuso horário do servidor acaba permitindo datas de encerramento final antes de datas de inicio, quando a data de inicio é igual a data atual)**)

Todo evento deverá ter um professor responsável vinculado:

No nosso modelo, todo organizador é considerado um professor e todo evento precisa ter um organizador. Devido a isso, todos os eventos tem um professor cadastrado que segue a figura 14 que mostra a lista de evento e todos os organizadores/ professores registrados.

Nome	tipo	Data de inicio	Data de encerramento	Hora de inicio	Hora de termino	Duracao	Local	Participantes	Vagas	Organizador	Ações
Programação Python	Ensino presencial	08/12	08/12	06:00	08:00	30	Taguatinga	0	30	Guilherme Santiago	Editar Evento Excluir
Java Concorrente	Palestra	09/12	12/12	11:00	15:00	10	Ceub	0	15	Guilherme Santiago	Editar Evento Excluir
Aula Web	Ensino presencial	08/12	08/12	06:13	08:13	7	Ceub	0	3	Guilherme Santiago	Editar Evento Excluir
Sistema cardiovascular	Palestra	09/12	09/12	08:30	10:30	31	Unb	0	19	Gabriel Paiva	Editar Evento Excluir
Arquitetura de Hardware	Ensino a distancia	08/12	08/12	21:01	22:15	2	Remoto	0	10	Guilherme Santiago	Editar Evento Excluir
Inteligencia Artificial	Palestra	10/12	09/12	12:00	14:00	2	Ceub	0	40	Guilherme Santiago	Editar Evento Excluir

Figura 14

Um usuário não poderá se inscrever mais de uma vez no mesmo evento.

Como exemplo dessa implementação usaremos o usuário Rafael Couraeel de email(rafael@gamil.com) e senha(rafael12??) para demonstrar que ele não deve poder se inscrever em um evento que já foi realizado a inscrição do próprio. Os eventos que ele está inscrito são esses conforme a figura:

Eventos em que Rafael Couraeel está inscrito									
Nome	tipo	Data de inicio	Data de encerramento	Hora de inicio	Hora de termino	Duracao	Local	Vagas	Ações
Aula Web	Ensino presencial	08/12	08/12	06:13	08:13	7	Ceub	2	
Sistema cardiovascular	Palestra	09/12	09/12	08:30	10:30	31	Unb	18	
Ver eventos disponíveis									

Seguindo a lógica ele não deve ser capaz de se inscrever no segundo evento novamente e devido a isso não é possível interagir com a caixa de inscrição de novo, com ela contendo uma caixa “inscrito” no lugar:



Banner do Evento

Sistema cardiovascular -

Tipo do Evento: Palestra

Local: Unb

Data: 09/12 até 09/12

Horário: 08:30h às 10:30h

Já inscrito

O sistema não permitirá a inscrição de novos participantes quando o limite de vagas do evento for atingido:

Devido ao modelo de negócio pedido o sistema bloqueia inscrições depois de todas as vagas já terem sido preenchidas pelos usuários, logo para essa demonstração serão usados o usuário Rafael Courael e o usuário Ricardo Ribeiro para adentrar no seguinte evento:



Libere seu Potencial -

Tipo do Evento: Workshop

Local: Ceub

Data: 10/12 até 19/12

Horário: 10:00h às 22:00h

Vagas disponíveis: 1

[Inscrir-se](#)

O usuário Rafael faz a inscrição:

Eventos em que Rafael Couraeel está inscrito

Nome	Tipo	Data de inicio	Data de encerramento	Hora de inicio	Hora de termino	Duracao	Local	Vagas
Aula Web	Ensino presencial	08/12	08/12	06:13	08:13	7	Ceub	2
Sistema cardiovascular	Palestra	09/12	09/12	08:30	10:30	31	Unb	18
Libere seu Potencial	Workshop	10/12	19/12	10:00	22:00	50	Ceub	0

O usuário Ricardo vai até a aba de eventos para se inscrever no mesmo evento que está com 0 vagas:



Libere seu Potencial -

Tipo do Evento: Workshop

Local: Ceub

Data: 10/12 até 19/12

Horário: 10:00h às 22:00h

Vagas disponíveis: 0

[Inscrever-se](#)

Após clicar em “Inscrever-se” aparece a mensagem de erro, além da aba de “Meus eventos” continuar vazia:

Não há mais vagas disponíveis

Eventos em que Ricardo Ribeiro está inscrito

Você ainda não está inscrito em nenhum evento.

[Ver eventos disponíveis](#)

A senha do usuário deverá seguir critérios de segurança: com no mínimo 8 caracteres, contendo letras, números e caracteres especiais. O formulário de cadastro exigirá um campo de confirmação de senha.

No nosso modelo de negócio, implementamos o limite mínimo de 8 caracteres com pelo menos 1 caractere especial e um número como segue a imagem a seguir:

senha varchar(200)
guilherme12??
ricardo12??
rafael12??
jooj420:
jooj4209:

Mas, não implementamos um campo de confirmação de senha. na tela de cadastro

Cadastro de Usuarios

Nome: _____

Telefone: _____

Instituicao de Ensino: _____

Email: _____

Senha: _____

