

Seguem as instruções para a realização do teste para a posição de Desenvolvedor Backend na Neoprospecta Microbiome Technologies.

O deadline para entrega é dia 7 dias após recebimento deste e-mail o feedback da Neoprospecta deverá ser dado durante os próximos 15 dias.

Segue a descrição, coisa simples apenas para analisar sua familiaridade com o Django, Django Rest Framework (DRF) e práticas de programação.

Utilizar bitbucket, SQLITE, python 3.5+ e versões mais recentes do Django e Django rest framework

Criar um comando no manage.py do projeto Django para baixar, descompactar e importar as primeiras 5 mil entradas do arquivo na url  
[https://www.arb-silva.de/fileadmin/silva\\_databases/release\\_128/Exports/SILVA\\_128\\_LSURef\\_tax\\_silva.fasta.gz](https://www.arb-silva.de/fileadmin/silva_databases/release_128/Exports/SILVA_128_LSURef_tax_silva.fasta.gz)

Os dados vem no seguinte formato:

```
>GCVF01000431.1.2369
Bacteria;Proteobacteria;Gammaproteobacteria;Oceanospirillales;Alcanivoraceae;Alcanivorax;Thalassiosira rotula
CGUGCACGGUGGAUGCCUUGGCAGCCAGAGGCGAUGAAGGACGUUGUAGCCUGCGAUA
AGCUCCGGUUAGGUGGCAAACA
ACCGUUUGACCCGGAGAUUCUGGAAUGGGGCAACCCACCCGUUGUAAGGCGGGUAUCA
CCGACUGAAUCCAUAAGGUCGGU
GAGGCGAACGCGGGGAACUGAAACAUCUAAGUACCCGUAGGAACAGAAAUCAAUUGAGA
UUCCCUGAGUAGCGGCGAGCG
AACGGGGAUUAGCCCUUAAGCUGAUGACUGAUUAGGAGAACGGUCUGGGAAGGCCGAC
CAUAGUGGGUGAUAGUCCCGUA
UCCGAAAAUCUGAUUCAGUGAAAACGAGUAGGUCGGGGCACGUGUAACCUUGACUGAAC
AUGGGGGGACCAUCCUCCAAG
GCUAAAUACUCCUGGCUGACCGAUAGUGAACAGUACCGUGAGGGAAAGGCGAAAAGAA
CCCCGGAGAGGGGAGUGAAAU
AGAUCCUGAAACCGUGCACGUACAAGCAGUCGGAGCCCGCUUUGUUGGGUGACGGCGU
ACCUUUUGUAUAAUGGGUCAGC
GACUUUUUCAGUAGCGAGGUUAACCAUCUAGGGGAGCCGUAGGGAAACCGAGUCUG
AAUAGGGCGUUGAGUUGCUGGG
AAUAGACCCGAAACCGGACGAUCUACCAUGGGCAGGUUGAAGGUGCGGUAAACACGCAC
UGGAGGACCGAACCGGGAUCU
GUUGAAAAAGAUUCGGAUGACCUGUGGGUCGGAGUGAAAGGCUAUUAAGUCCGGAGA
UAGCUGGUUCUCCCCGAAAGCU
```

AUUUAGGUAGCGCCUCGUUAUAUCACCACCGGGGGUAGAGCACUGUUUCGGCUAGGGGC  
CCAUCCCGGGUUACCAAACCGA  
UGCAAACUCCGAAUACCGGUGAGUGCAGUACGGGAGACACACGGCGGGUGCUAACGUC  
CGUCGUGGAGAGGGGAAACAACC  
CAGACCGCCAGCUAAGGUCCCCAAAUUCCAGUUAAGUGGGAAACGAUGUGGGAAGGCU  
UAGACAGCUAGGAGGUUGGCUU  
AGAAGCAGCCAUCCUUUAAAGAAAGCGUAAUAGCUCACUAGUCGAGUCGGCCUGCGCGG  
AAGAUGUAACGGGGGCUCAAAC  
UGGAUACCGAAGCUGCGGCAGUGUCUAGGCAUACUGGGUAGGGGAGCGUUGUGUAA  
GUCUGUGAAGGUGUGUUGAGAA  
GCAUGCUGGAGAUUAUCACAAGUGCGAAUGCUGACGUGAGUAACGAUAAUGCGGGUGAA  
AAACCCGCACGCCGAAAGACCA  
AGGUUUCUGCGCAACGCUAAUCGGCGCAGGGUGAGUCGGCCCCUAAGGCGAGGCUGA  
AAGGCGUAGCUGAUGGGAAACG  
GGUAAUAUUCGGUACUUCUUGUAACUGCGAUGGAGAGACGGAGAAGGCUAGGCCUA  
CCGGGCGUUGGUUGUCCCGGGG  
AAAGACCGUAGGCUGGGAUCUUAAGGCAAUCCGGGAUCCUAAGGCUGAGAGUCGAGAC  
CACCGGCCCUUUGCGGCUGGGA  
AGUGGUUGAUGCCCUGCUUCCAGGAAAAUCUUCUAAGCUUCAGGUUACAAGGAACCGUA  
CCCCAAACCGACACAGGUGGU  
UGGGAUGAGUAUUCUAAGGCGCUUGAGAGAACUCGGGUGAAGGAACUAGGCAAAGUGG  
UACCGUAACUUCGGGAGAAGGU  
ACGCCACUUGGUGUGAAGGGCUUGCCCCGUAAGCACCGAGUGGUCGCAGUGAAAAGGC  
CCCUGCAACUGUUUAUUAAAAA  
CACAGCACUCUGCAAACGCGUAAGCGGACGUUAUAGGGUGUGACGCCUGCCCGGUGCCG  
GAAGGUUAAUUGAUGGGGUUAG  
CUUGCGCGAAGCUCUUGAUCGAAGCCCCGGUAAACGGCGGCCGUAACAAUAACGGUCC  
UAAGGUAGCGAAAUUCUUGUC  
GGGUAAGUUCGACCUGCACGAAUGGCGUAACGAUGGGGGCGCUGUCUCCACCCGAGA  
CUCAGUGAAAUCGAAAUCGCAG  
UGAAGAUGCUGUGUACCCGCGGCUAGACGGAAAGACCCCGUGAACCUUUACUACAGCU  
UCACAGUGGACUUUGAACCGUC  
UUGUGUAGGAUAGCUGGGAGACUUUGAAGCGGUGACGCUAGUCAUCGUGGAGUCGUCC  
UUGAAAUACCAGCCUGGCAUGU

...

No banco de dados do Django o model que irá salvar os dados deve conter o *access id* (GCVF01000431.1.2369), o *Kingdom* (Bacteria), a *Species* (Thalassiosira rotula) e a primeira linha da sequencia de DNA/RNA (CGUGCACGGUGGAUGCCUUGGCAGCCAGAGGCGAUGAAGGACGUUGUAGCCUGCGAU AAGCUCCGGUAGGUGGCAAACA)

A modelagem ficará algo mais ou menos assim:

```
class Entry
    id
    access_id (deve ser unique)
    kingdom
    specie
    sequence
```

O *kingdom* e a *specie* devem ser *foreign key*, ao adicionar uma entrada deve verificar se já existe *kingdom* e *species* e referenciá-la, se não existir deve ser criada.

```
class Kingdom
    id
    label
```

```
class Specie
    id
    label
```

Habilitar o admin do django para poder manipular entradas, *kingdoms* e *species* manualmente

Criar uma interface em Django View para exibir todas as entradas, com paginação de 1000 itens, em formato tabular.

access\_id | kingdom.label | specie.label | sequence

Criar um endpoint (serializer e viewset) com Django Rest Framework para crud do objeto Entry  
~/entry/ (Listagem)  
~/entry/\$id (Objeto)

No endpoint DRF deve ser possível fazer buscas por *kingdom*, *access id* e *species*  
ex. /entry/?kingdon\_\_label=Bacteria

O código da aplicação deve ser publicado no Github ou Bitbucket e enviado por e-mail.

O deploy da aplicação deve ser feito no Python Any Where para consulta do endpoint e da tela:  
[https://www.pythonanywhere.com/details/django\\_hosting](https://www.pythonanywhere.com/details/django_hosting)

Desejamos sucesso em seu teste e nos colocamos a disposição para esclarecer quaisquer dúvidas.