

Abridor de Tarefas

Ildo.Banaseski | Ricardo.Rocha | Quiu
vs 1.0

Setup

Configurar Ambiente Virtual

```
python -m pip install virtualenv
> md projeto
> cd projeto\

## Criar o ambiente virtual
projeto\> virtualenv venv

## Ativar o ambiente virtual
projeto\> venv/scripts/activate

## Instalar todas as libs
(venv)>pip install python-redmine

## Salvar o ambiente no arquivo
(venv)>pip freeze > requirements.txt
```

Note que agora criou-se o arquivo requirements.txt que contém todas as libs da nossa aplicação requirements.txt

```
python-redmine==2.3.0
```

Para reinstalar este ambiente, criando o ambiente virtual

```
python -m pip install virtualenv
> md projeto
> cd projeto\

# Copie o arquivo requirements.txt para esta pasta

## Criar o ambiente virtual
projeto\> virtualenv venv

## Ativar o ambiente virtual
projeto\> venv/scripts/activate
```

```
## Instalar todas as libs salvas no arquivo  
(venv)>pip install -r requirements.txt
```

Objetivos

1. Criar uma tarefa

Foi criado o script na versão 1.0

```
from redminelib import Redmine  
import datetime  
  
OBJETO = {  
    "status": {  
        "desenvolvimento": 36,  
        "sprint backlog": 54,  
        "pronto": 60  
    },  
    "prioridade": {  
        "crítica": 1,  
        "alta": 2,  
        "baixa": 3  
    },  
    "tipo": {  
        "atividade": 49,  
        "história": 102,  
        "análise técnica": 106,  
        "desenvolvimento": 107,  
        "teste funcional": 108,  
        "refatoração": "?",  
        "regra negócio": "?",  
        "teste unitário": "?"  
    }  
}  
  
PROJETOQUIU = 587  
AGENTEQUIU = 1834  
PRODUTOQUIU = 34446  
  
STATUS = "status"  
DESENV = "desenvolvimento"  
BACKLOG = "sprint backlog"  
  
PRIORIDADE = "prioridade"  
BAIXA = "baixa"  
ALTA = "alta"  
  
TIPO = "tipo"  
ATIVIDADE = "atividade"  
HISTORIA = "história"
```

```
ANALISETECNICA = "análise técnica"
TESTE = "teste"
REFATORACAO = "refatoração"
REGRANEGOCIO = "regra negócio"
TDD = "teste unitário"

ID = 'id'
NAME = 'name'
VALUE = 'value'
REDMINE = 'https://redmine.questor.com.br'

#region GetCustomFields
def CF_GetVersaoFinalizacao(Versao):
    return {ID: 62,
            NAME: 'Versão Finalização',
            VALUE: Versao['name']}
def CF_GetDocumentacaoIndefinida():
    return {ID: 33,
            NAME: 'Ajuda',
            VALUE: 'Documentação indefinida'}
def CF_GetIntervencao():
    return {ID: 73,
            NAME: 'Intervenção',
            VALUE: '0'}
def CF_GetAntecipado():
    return {ID: 144,
            NAME: 'Antecipado',
            VALUE: 'Não'}
def CF_GetDLLHistoria():
    return {ID: 71,
            NAME: 'DLL',
            VALUE: '0'}
def CF_GetTamanhoEstimado():
    return {ID: 127,
            NAME: 'Tamanho estimado',
            VALUE: 'P'}
#endregion

def getCustom2(Versao):
    return [
        CF_GetVersaoFinalizacao(Versao),
        CF_GetDocumentacaoIndefinida(),
        CF_GetIntervencao(),
        CF_GetAntecipado(),
        CF_GetDLLHistoria(),
        CF_GetTamanhoEstimado()
    ]

def getCustomFields(Versao, Tamanho = 'P', DLL = '0'):
    return [
        {
            'ID': 62,
            'NAME': 'Versão Finalização',
```

```

        'VALUE': Versao[NAME]
    },
    {
        'ID': 33,
        'NAME': 'Ajuda',
        'VALUE': 'Documentação indefinida'
    },
    {
        'ID': 73,
        'NAME': 'Intervenção',
        'VALUE': '0'
    },
    {
        'ID': 144,
        'NAME': 'Antecipado',
        'VALUE': 'Não'
    },
    {
        'ID': 71,
        'NAME': 'DLL',
        'VALUE': DLL
    },
    {
        'ID': 127,
        'NAME': 'Tamanho estimado',
        'VALUE': Tamanho
    }
}]

```

OBJETO[STATUS][DESENV]

```

def CriarTarefa(id, subject, description, status, priority, agent, category,
version, initialdate, enddate, tracker, custom, parent_id = null, save =
True):

```

```

    issue = redmine.issue.new()
    issue.project_id      = id
    issue.subject         = subject
    issue.description     = description
    issue.status_id       = status
    issue.priority_id     = priority
    issue.assigned_to_id  = agent
    issue.category_id     = category
    issue.fixed_version_id = version
    issue.start_date      = initialdate
    issue.due_date        = enddate
    issue.tracker_id      = tracker
    issue.custom_fields   = custom
    issue.parent_issue_id = parent_id
    if save:
        issue.save()

```

```

REDMINEKEY = "7191d4dd9cc2fa5270a8335db1945ed75c3a6f89"

```

```

REDMINEURL = 'https://redmine.questor.com.br'

```

```

redmine      = Redmine(REDMINEURL,
key           = REDMINEKEY)

```

```

VERSAO = {
    "id": 4824,
    "name": "999.999.999.999"
}

CriarTarefa(
    #parentid = '2081918',
    id = PROJETOQUIU,
    subject = 'Tarefa demonstração',
    description = 'Tarefa de demonstração criada automaticamente pelo
python\nLinha 2',
    status = OBJETO[STATUS][DESENV], # OBJETO[STATUS][DESENV] |
OBJETO[STATUS][BACKLOG]
    priority = OBJETO[PRIORIDADE][BAIXA], # OBJETO[STATUS][ALTA] |
OBJETO[STATUS][BAIXA]
    agent = AGENTEQUIU,
    category = PRODUTOQUIU,
    version = VERSAO[ID],
    initialdate = datetime.date(2022, 5, 6),
    enddate = datetime.date(2022, 5, 19),
    tracker = OBJETO[TIPO][DESENV],
    # custom = getCustomFields(VERSAO, Tamanho = "")
    custom = getCustom2(VERSAO)
)

```

Modelo de Input

Desta forma o input pode ser dividido em dois arquivos `.env` e `json`, sendo `.env` um arquivo de configuração com as constantes do projeto, e o `json` com as informações específicas da tarefa

```

id = 587 # Id projeto QUIU
status = 'Backlog' # 'Desenvolvimento' | 'Backlog'
prioridade = 'baixa' # 'baixa' | 'alta'
agente = 1834 # Agente QUIU
categoria = 34446 # Produto QUIU
inicio = '06/05/2022'
termino = '19/05/2022'

```

```

{
    "assunto": "assunto",
    "descricao": "descricao",
    "tipo": "desenvolvimento"
}

```

Exemplo passando um array de tarefas

```
[{
  "assunto": "assunto",
  "descricao": "descricao",
  "tipo": "analise técnica",
  "tarefaprincipal": "null"
},
{
  "assunto": "assunto",
  "descricao": "descricao",
  "tipo": "desenvolvimento",
  "tarefaprincipal": "null"
}]
```

Abstração

```
class Tarefa(self, assunto, descricao = '', tipo = 'desenvolvimento',
tarefaprincipal = null)
    self.assunto = assunto,
    if descricao > '':
        self.descricao = descricao
    else:
        self.descricao = assunto
    self.tipo = tipo
    self.tarefaprincipal = tarefaprincipal
```

Id

```
status_id conhecidos:
36 = Desenvolvimento
54 = Sprint Backlog
60 = Pronto

priority_id conhecidos:
3 = Baixa

tracker_id conhecidos:
49 = Atividade
102 = Historia
107 = Desenvolvimento
108 = Teste Funcional
```