

Python Backend con DRF

Creación rápida de APIs para prototipos

Yeison Vargas

Consultor y Desarrollador Senior



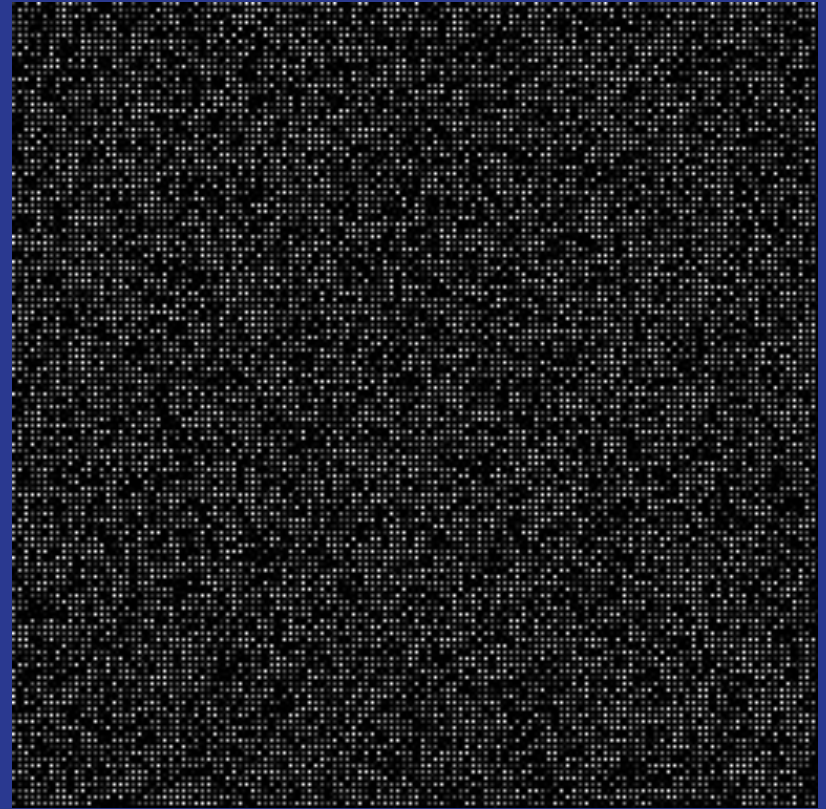
@yeisonvargasf



¿Cómo es construir un MVP?

Fail 1

Complejidad



Conway's Game of Life

Preguntas Clave y Respuestas Claras

Fail 1

¿Mis conocimientos técnicos son suficientes?

¿Quiero aprender alguna tecnología? VS construir el MVP lo antes posible

¿Soy consciente de que los cambios se necesitan muy rápido?

¿La tecnología que voy a usar es popular entre otros devs?

Preguntas Clave y Respuestas Claras

Fail 1

¿El MVP se enfoca en mi propuesta de valor?

¿Necesito un equipo técnico más grande?

¿Puedo escalar con este MVP?

¿Sé construir software de calidad?

Fail 2

No organizarse



Stupid Team discussions

Preguntas Clave y Respuestas Claras

Fail 1

¿Tengo una documentación básica y útil que explique los estándares a seguir?

¿Un desarrollador puede acoplarse rápido al equipo?

¿Estoy convencido de que por encima de mi EGO está la decisión de solucionar un problema?

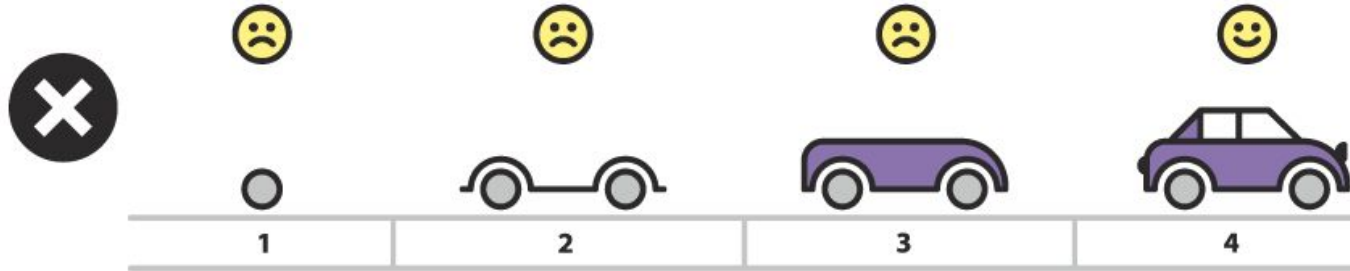
¿Tengo un mecanismo para resolver las discusiones de forma rápida?

More Fails

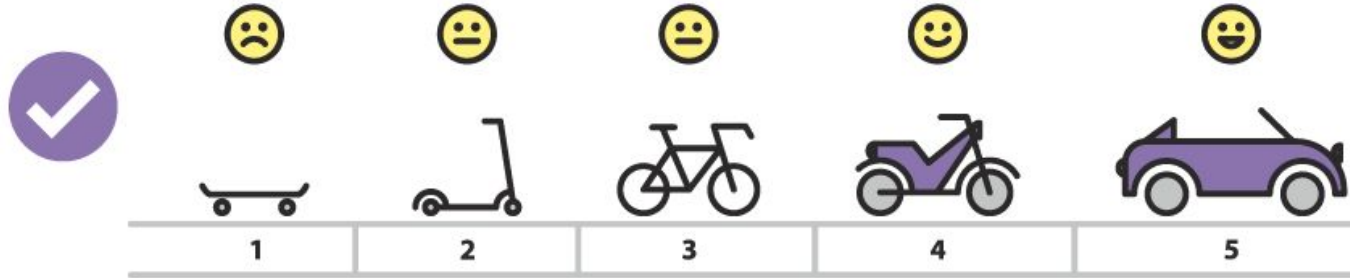


Silicon Valley (serie de televisión)

Proceso de Desarrollo



Proceso de Desarrollo





Let's hacking

Django

The Web framework for
perfectionists with deadlines

Python Framework que permite
desarrollar rápido y con calidad de
producto en mente.

DRF

Django Rest Framework

Framework en el Top de Django que permite programar API Rest de forma rápida y sencilla.

Crear Proyecto en Django

```
virtualenv -p python3 env  
source env/bin/activate  
nano requirements.txt  
Django==2.2.6  
djangorestframework==3.10.3  
pip3 install -r requirements.txt  
django-admin startproject platzi  
cd platzi/  
python manage.py startapp cursos
```

Actualizar Settings

```
nano platzi/platzi/settings.py
```

```
'rest_framework',  
'corsos',
```


Actualizar Settings

nano platzi/platzi/settings.py

```
REST_FRAMEWORK = {
    'DEFAULT_PAGINATION_CLASS': 'rest_framework.pagination.PageNumberPagination',
    'PAGE_SIZE': 10,
    'DEFAULT_AUTHENTICATION_CLASSES': (
        'rest_framework.authentication.BasicAuthentication',
        'rest_framework.authentication.SessionAuthentication',
    ),
    'DEFAULT_PERMISSION_CLASSES': (
        'rest_framework.permissions.IsAuthenticated',
    )
}
```

Agregar Modelos

```
nano platzi/cursos/models.py
```

```
python3 manage.py makemigrations
```

```
python3 manage.py migrate
```

Agregar Routers

```
nano platzi/platzi/urls.py
```

```
router = routers.DefaultRouter()
```

```
router.register(r'cursos', viewsCursoViewSet)
```

```
router.register(r'temas', viewsTemaViewSet)
```

```
urlpatterns = [
```

```
    path('admin/', admin.site.urls),
```

```
    path('', include(router.urls)),
```

```
    path('api-auth/', include('rest_framework.urls', namespace='rest_framework'))
```

```
]
```

Agregar Serializers

nano platzi/cursos/serializers.py

```
class CursoSerializer(serializers.HyperlinkedModelSerializer):  
    class Meta:  
        model = Curso  
        fields = ('id', 'nombre', 'creador', 'creado')
```

Agregar Serializers

nano platzi/cursos/serializers.py

```
class TemaSerializer(serializers.HyperlinkedModelSerializer):  
    class Meta:  
        model = Tema  
        fields = ('id', 'nombre', 'descripcion', 'actualizado')
```

Agregar ViewSets

nano platzi/cursos/views.py

```
class CursoViewSet(viewsets.ModelViewSet):  
    queryset = Curso.objects.all()  
    serializer_class = CursoSerializer  
  
    def get_queryset(self):  
        return Curso.objects.filter(creador=self.request.user).order_by('-creado')
```

Agregar ViewSets

nano platzi/cursos/views.py

```
class TemaViewSet(viewsets.ModelViewSet):  
    queryset = Tema.objects.all()  
    serializer_class = TemaSerializer  
  
    def get_queryset(self):  
        return Curso.objects.all().order_by('-actualizado')
```



Comprobemos el resultado

Gracias

@yeisonvargasf



<https://github.com/yeisonvargasf>
