

## Copilot

Para o projeto de gestão de projetos e desenvolvimento de software usando Flask, aqui estão exemplos de códigos que você poderia colocar dentro de cada arquivo .py:

### init.py (no diretório app)

Este arquivo inicializa sua aplicação Flask e reúne todos os componentes.

```
from flask import Flask
from config import config

def create_app(config_name):
    app = Flask(__name__)
    app.config.from_object(config[config_name])

    # Inicialização de extensões, como SQLAlchemy, se estiver usando
    # db.init_app(app)

    # Registro de blueprints
    from .modulos.clientes import clientes as clientes_blueprint
    app.register_blueprint(clientes_blueprint, url_prefix='/clientes')

    # Repita o registro para outros blueprints

    return app
```

### modelos.py (em cada módulo)

Define as classes de modelo para ORM, como SQLAlchemy.

```
from app import db

class Cliente(db.Model):
    __tablename__ = 'clientes'
    id = db.Column(db.Integer, primary_key=True)
    nome = db.Column(db.String(64), unique=True)
    # Outros campos...

class Projeto(db.Model):
    __tablename__ = 'projetos'
    id = db.Column(db.Integer, primary_key=True)
    nome = db.Column(db.String(64))
    cliente_id = db.Column(db.Integer, db.ForeignKey('clientes.id'))
    # Outros campos...

class Atividade(db.Model):
    __tablename__ = 'atividades'
    id = db.Column(db.Integer, primary_key=True)
    descricao = db.Column(db.Text)
    projeto_id = db.Column(db.Integer, db.ForeignKey('projetos.id'))
    # Outros campos...
```

### visoes.py (em cada módulo)

Contém as rotas e as visões.

```
from flask import render_template
from . import clientes # Importa o blueprint

@clientes.route('/')
def listar_clientes():
    # Lógica para listar clientes
    return render_template('listar_clientes.html', clientes=clientes)
```

## formularios.py (em cada módulo)

Se estiver usando WTForms, este arquivo conterá as definições dos formulários.

```
from flask_wtf import FlaskForm
from wtforms import StringField, SubmitField
from wtforms.validators import DataRequired

class ClienteForm(FlaskForm):
    nome = StringField('Nome', validators=[DataRequired()])
    submit = SubmitField('Enviar')
```

Esses são apenas exemplos básicos para começar. Você precisará adicionar mais lógica e funcionalidades conforme as necessidades do seu projeto. Lembre-se de importar e inicializar quaisquer extensões que você esteja usando, como SQLAlchemy ou WTForms, e de criar suas rotas e visões de acordo com a lógica de negócios do seu projeto. Boa sorte com o desenvolvimento! 🚀