

Relatorio da primeira entrega do codigo “organizador de atividades extras na UFRJ”

Luiz Henrique Gonçalves Miranda¹

¹Universidade Federal do Rio de Janeiro (UFRJ)
– Rio de Janeiro – RJ – Brazil

luizhenrique0706@poli.ufrj.br

***Resumo.** O trabalho se trata de uma entrega intermediaria da matéria linguagem de programação do segundo semestre do ano de 2018 O mesmo se trata de resolver um problema quaisquer encontrado na UFRJ, onde tentei fazer uma previa de um controlador de atividades extra.*

1. Informações Gerais

Se trata de uma prévia do trabalho, que deve ser melhorado, possui apenas o nucleo foi implementado para conceito, mas tem o objetivo de ser generico e na sessão final, terá uma lista de povaveis proximos passos.

2. Problema

Com a nova obrigatoriedade de mais horas de atividades extras para se conseguir o diploma na UFRJ, se torna mais obvio e gritante a necessidade de uma plataforma que facilite as ofertas e divulgação de vagas para os mesmo, além de reunir documentos e links necessarios para tais atividades serem apreciados pelas multiplas secretarias de cada instituição principalmente, quando se faz entre instituições diferentes (ex: departamento de Historia, no campus Praia Vermelha recebendo documentos de uma extensão realizada na Politecnica), já que por mais que os documentos do siga sejam iguais, as coordenações que dão parecer, podem ser desconhecidas dos alunos.

3. Principio da modelagem

A modelagem inicial, para teste de conceito, alunos que desejam entrar em projetos e coordenadores de projetos, são iguai e indiferentes em relação ao centro, ou seja, a entidade que possui as informações de centro foi desconsiderada (nela se daria informações do centro, documentos especificos completando os documentos genericos da UFRJ que é dado no options-postalém disso ela seria ligada por user, gerando a diferença de centro de cada aluno e com isso realizando as filtragens necessarias).

Além disso, a extrapolação do user feita pelo options-user, geraria uma chave conjunta para uma criar os parametros basicos para os provaveis, e inseridos campos citados:

3.1. Adminitrador

Administrador é um servidor que terá credenciais de servidor adminitrativo, ele assim como feito, tem o poder de cancelar contas e confirmar a criação de 'criadores de oportunidade', além da habilidade (não incrementada) de deletar oportunidades, obs: ele teria

a oportunidade de deletar, mas não de alterar, como foi criado duas funções separadas, já prevendo essa necessidade. Além disso seria cargo dele a criação de novos tipos de vagas (Iniciação, Extensão ou Equipe) e centros (Poli, CCMN, IM) (incrementado) e alterações do mesmos, por ultimo reset de senhas dos usuarios (não incrementado).

3.2. Criadores de oportunidades

O responsavel por gerar as postagens, eles teriam na chave conjunta a chave junto a instituição, como codigo do projeto que é atrelado e professor responsavel, eles criam as oportunidades, preenchendo todos os campos necessarios e links especiais, além de conseguir alterar as suas oportunidades abertas, podem ver as outras oportunidades disponiveis, mas não podem se submeter a elas.

3.3. Pedidores de oportunidades

Seriam os alunos que desejam se inscrever, eles conseguem ver perfil geral de todas as oportunidades, no futuro, teriam como filtrar por centro e área da postagem e além disso podem ver todos os detalhes de uma oportunidade especifica e se inscrever nela (foi feito uma simples html dizendo que foi disparado um e-mail com id do usuario + id do criador da oportunidade para prova de conceito, mas seria utilizado algo com os contatos dados no perfil do usuario)

4. Visual

A visualização é simples, utilizando o mesmo template simples do blog exemplo do Flask

4.1. Pedidores de oportunidades e suas paginas

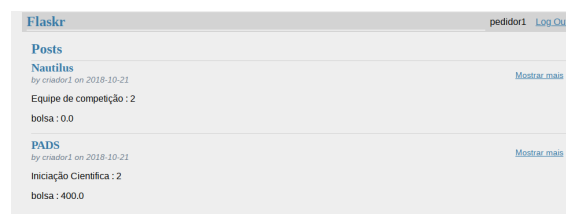


Figura 1. Exemplo de quem pede vagas

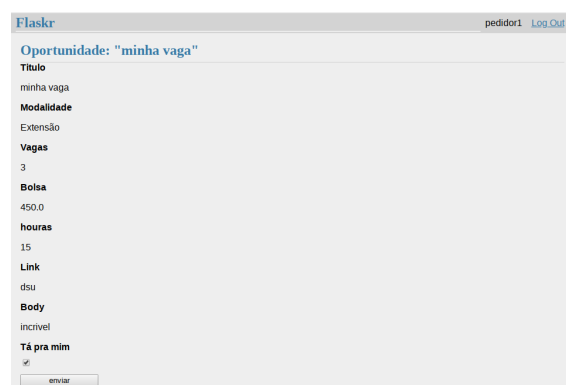


Figura 2. Exemplo de quem pede vagas pedindo vagas

4.2. Criador de oportunidades e suas paginas

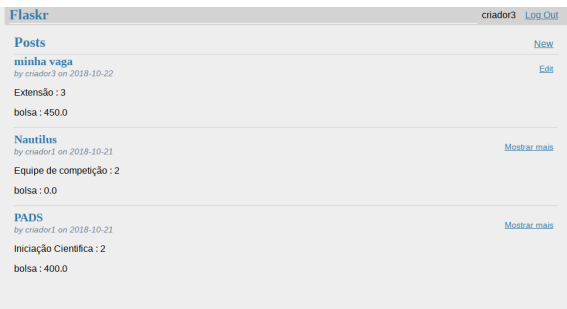


Figura 3. Geral do criador de vagas



Figura 4. Criador visualizando outra oportunidade que não é sua

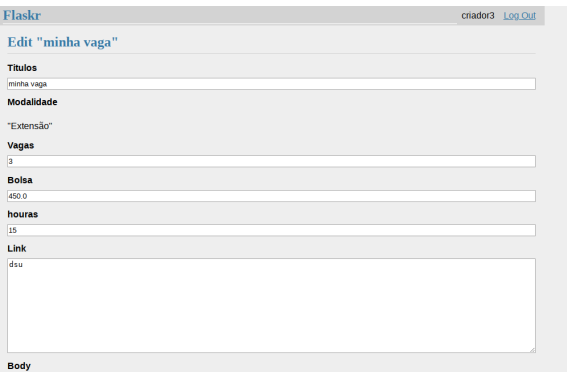


Figura 5. Criador visualizando sua oportunidade e conseguindo alterar

4.3. Root e suas pagina

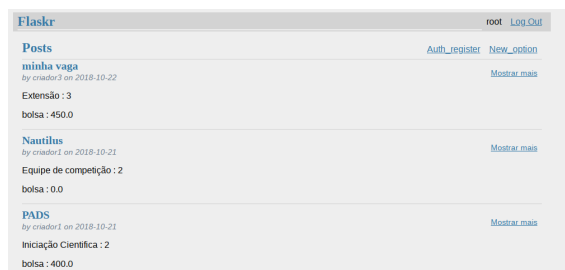


Figura 6. Opções do root

5. Considerações

Com um boa garimpagem, e maior dedicação no css (só foi utilizado dois diferentes e a pagina base é a mesma do curso exemplo do proprio site do Flask, além dos acréscimos citados durante, o app poderia ser muito utilizado e até mesmo ser adicionado a paginas da UFRJ, pois melhora muito, tanto o modo de divulgação, quanto o proprio modo de controle e por ultimo ainda facilitaria a parte burocratica que os alunos tem ao conseguir horas e certificados, já que poucos centros, possuem essas documentações bem organizadas e muitas secretarias pedem comprovantes diferentes (principalmente, quando são licenciatura e saúde)

6. References

Curso de Flask