

# MAC0215 - Proposta

Aluno: Victor Hugo Miranda Pinto  
Supervisor: Alfredo Goldman vel Lejbman  
Co-supervisor: Renato Cordeiro Ferreira

2º Semestre de 2018

## 1 Resumo

Será escrito por último.

## 2 Introdução

Nesta sessão apresentaremos uma descrição de hackathons, dando uma definição geral sobre esse tipo de evento, e então o HackathonUSP em específico. Em seguida, introduziremos uma plataforma chamada Hacknizer, onde será integrado o código gerado nesta pesquisa.

### 2.1 Hackathons

O termo *hackathon* vem da combinação das palavras *hacking* e *marathon*. Hackathons são eventos contínuos em que os participantes se organizam em pequenos times com o objetivo de criar um protótipo de software sob um limite de tempo (24 horas em seu formato mais comum). [KPR<sup>+</sup>15]

Hackathons são realizados com muitos objetivos diferentes e de maneiras diferentes na maior parte dos casos, por empresas (normalmente de tecnologia) ou instituições de ensino. Graças à essa diversidade, não existe uma definição totalmente aceita sobre todos os detalhes de um hackathon.

Muitas empresas de tecnologia realizam hackathons internos, ou seja, apenas para seus funcionários, em geral com o objetivo de gerar inovação dentro da empresa e fazer com que os desenvolvedores trabalhem em projetos diferentes daqueles nos quais eles trabalham diariamente. Dependendo dos protótipos desenvolvidos durante o evento, a empresa pode decidir implantar a ideia. Uma empresa muito famosa por sua cultura de hackathons internos é a *Facebook*, por exemplo. É comum essas empresas também realizarem hackathons abertos ao público, em geral com o objetivo de promover um novo produto na comunidade desenvolvedora. Em geral, esses eventos também incluem algum tipo de premiação para motivar a participação, constituindo portanto uma competição.

No contexto de instituições de ensino superior são realizados hackathons *universitários*. Esse tipo de evento possui objetivos diferentes daqueles das empresas, já que costumam não ter fins lucrativos e buscam incentivar a criatividade e a inovação no ambiente universitário. Hackathons universitários são muito efetivos como ferramenta educacional para o crescimento de futuros desenvolvedores. Esses eventos fornecem um ambiente para aprendizado e experimentação com novas tecnologias de maneira muito rápida, além de incentivar os alunos a criar novas soluções para problemas muitas vezes internos [Kay17]. Além disso, incentivam a integração entre alunos de diferentes cursos, promovendo maior troca de experiências de forma complementar, dado que muitos problemas não podem ser resolvidos apenas com uma área do conhecimento.

## 2.2 HackathonUSP

O HackathonUSP é o um dos maiores hackathons universitários do Brasil, com número crescente de inscritos e participantes. Ele tem como público-alvo os alunos de graduação e pós da Universidade de São Paulo. É organizado pelo grupo de extensão USPCodeLab (UCL) em parceria com o Núcleo de Empreendedorismo da USP (NEU) e o Hardware Livre USP (HL).

O objetivo do evento é ser um hackathon acessível, o primeiro evento do tipo para alunos da Universidade, ao passo que ajude a própria comunidade USP a gerar soluções relevantes para as suas demandas.

## 2.3 Seleção de participantes

Um dos elementos mais decisivos para o sucesso do HackathonUSP é a seleção dos seus participantes. Como o evento historicamente possui mais inscritos que vagas, a etapa de seleção é importante para maximizar a participação (i.e., evitar a desistência) e a diversidade (i.e., evitar o viés de público existente na computação) do evento, possibilitando o impacto proposto na visão do evento. O algoritmo de seleção atual pode ser descrito da seguinte maneira:

Sejam  $N$  o número de inscritos no evento,  $M$  o número de participantes e  $D$  a taxa de desistência entre os selecionados para o evento. Os seguintes passos descrevem o algoritmo:

1. Selecionar os primeiros  $M$  inscritos, por ordem de inscrição.
2. Selecionar mais inscritos para obter  $M$  participantes efetivos contando com um percentual de desistência  $D$  (em geral, 150% de  $M$  para  $D$  igual a 33%), levando em consideração:
  - o completamento de times: selecionando um inscrito que participa de um time cujos demais integrantes foram selecionados nos primeiros  $M$ ;
  - a paridade de gêneros: selecionando mulheres que não tenham times para equilibrar a proporção de gênero entre os selecionados (a área da computação conta com um percentual pequeno de mulheres comparativamente ao de homens [FA90]);
  - o desempenho em edições anteriores: como último critério de completamento, selecionar times cujos membros tiveram destaque em edições anteriores mesmo que nenhum deles estivesse entre os  $M$  primeiros inscritos.

## 2.4 Hacknizer

O Hacknizer é uma plataforma para organizar e hospedar hackathons, visando prover uma solução integrada e completa para a criação desses eventos. Ele está sendo construído pelo *USPCodeLab* e seu desenvolvimento começou durante a *USPCodeLab Winter School 2018*, um evento da série de *dev.camps* (parte do ciclo *dev.journey*) organizados pelo grupo com objetivo de experimentar as mais novas tecnologias disponíveis para o desenvolvimento Web.

A arquitetura do sistema segue o padrão de microsserviços, ou seja, a estrutura de *backend* é constituída por módulos cujo desenvolvimento, implantação e replicação são semi-independentes, permitindo melhor escalabilidade do sistema [New15]. Os serviços existentes são:

1. **Hackathons:** responsável por gerenciar informações referentes a cada série de hackathons e suas respectivas edições hospedadas no sistema.

2. **Auth**: responsável por autenticação de usuários e autorização de acesso, além do armazenamento de dados básicos dos usuários (nome, e-mail, etc.).
3. **Gerador de PWAs**: responsável pela criação de *Progressive Web Apps* (PWAs) para cada hackathon, ou seja, páginas web instaláveis como aplicativos para cada evento.
4. **Participantes**: responsável pelo gerenciamento dos usuários associados aos hackathons, incluindo seus papéis numa dada edição (organizador, participante, mentor, jurado ou convidado).
5. **Grupos**: responsável pelo gerenciamento dos grupos de participantes dentro de um hackathon e pela submissão de projetos.
6. **Seleção**: responsável pela seleção manual de participantes a serem convidados para um certo hackathon a partir da lista de inscritos.
7. **API Gateway**: responsável por receber requisições dos clientes da plataforma (Web e mobile) e direcionar essas requisições para os microsserviços corretos afim de construir uma única resposta para enviar de volta ao cliente.

O projeto foi desenvolvido utilizando ferramentas e tecnologias tais como:

- **Docker**: usado para containerização dos módulos.
- **Kubernetes**: usado para gerenciamento de *clusters*.
- **GraphQL**: especificação para design de APIs.
- **Node**: linguagem e ambiente de programação para *backend*.
- **Koa**: *microframework* de *Node* para construir servidores HTTP.
- **Apollo Server**: biblioteca para construir servidores GraphQL.
- **PostgreSQL**: sistema gerenciador de bancos de dados relacionais para consulta.
- **Kafka**: transmissor de mensagens para consistência entre serviços.
- **Prisma**: mapeador do esquema de APIs GraphQL para o PostgreSQL.

### 3 Objetivos

Os objetivos dessa pesquisa são:

- **Automatizar a seleção de participantes**

Buscaremos uma maneira de automatizar o processo de seleção já existente para o HackathonUSP. Com isso, pretendemos aumentar a transparência na organização do evento, garantir a reprodutibilidade do processo e diminuir o trabalho envolvendo tal seleção dado que o evento vem crescendo a cada edição.

- **Integrar o algoritmo de seleção com o Hacknizer**

Trabalhando no serviço de **Seleção** do Hacknizer, adicionaremos um componente que recebe a lista de inscritos para um dado hackathon e é capaz de gerar uma recomendação dos participantes a serem convidados para o evento, aplicando o algoritmo desenvolvido no passo anterior.

- **Investigar possíveis melhorias no algoritmo de seleção para o HackathonUSP**

Com essa pesquisa, esperamos estabelecer um critério de seleção que minimize o grau de desistência ao passo que maximize a diversidade (gênero, ano de ingresso, curso, unidade) entre os participantes do evento.

## 4 Metodologia

- **Desenvolvimento do algoritmo**

Para o desenvolvimento do algoritmo de seleção e estudo do conjunto de dados disponíveis, vamos utilizar *Python* devido à disponibilidade da biblioteca *scikit-learn* que contém a implementação de diversos algoritmos para *aprendizagem de máquina* e *Jupyter Notebooks*, uma aplicação web de código aberto que permite a criação e compartilhamento de documentos contendo código *Python* e visualizações de dados. Em específico, vamos utilizar **florestas aleatórias**, um classificador que se baseia em múltiplas **árvores de decisão** aplicadas a um problema de classificação de forma que cada árvore classificadora é independente e contribui com o mesmo peso na classificação final [Bre01]. Variando os parâmetros considerados, vamos estudar como o algoritmo se comporta e qual seria o melhor conjunto de parâmetros para os nossos objetivos.

- **Integração com o Hacknizer**

Para integrar o algoritmo desenvolvido com o Hacknizer, vamos desenvolver um componente que fara parte do serviço de **Seleção** da plataforma. Esse componente será desenvolvido como uma API usando *Flask*, um *microframework* para desenvolvimento de APIs em *Python*, e *Graphene*, uma implementação em *Python* da especificação *GraphQL* de APIs. Essa API também será containerizada via *Docker* e sua implantação poderá ser feito junto com os demais serviços do Hacknizer. Escolhemos *GraphQL* para a API porque todas as APIs dos outros serviços do Hacknizer usam essa especificação, e *Python* com *Flask* para a parte envolvendo *aprendizagem de máquina*.

## 5 Plano de Trabalho

Os seguintes passos serão necessários para cumprir com os objetivos propostos:

- **Limpeza e organização dos dados a serem analisados (25 horas)**

Nessa primeira fase da pesquisa, com previsão de durar por volta de um mês e meio, será feita a coleta e agregação do conjunto de dados utilizado para a pesquisa, além da normalização desses dados e a geração de visualizações para primeiros *insights*. Esses dados serão coletados das edições anteriores do HackathonUSP.

- **Desenvolvimento do algoritmo (40 horas)**

Na segunda fase da pesquisa, serão definidos os critérios preliminares de decisão para o algoritmo de seleção e a comparação de diferentes modelos segundo critérios medidos sobre dados de validação.

- **Desenvolvimento do serviço (25 horas)**

Após decidir o algoritmo a ser utilizado, será desenvolvido um novo componente do serviço de seleção do Hacknizer, utilizando *Flask* para criar uma API *GraphQL* que receberá uma lista de inscritos para uma edição de um determinado hackathon e, aplicando o algoritmo desenvolvido, poderá gerar uma lista sugerindo os participantes para tal hackathon.

- **Feedback (10 horas)**

O HackathonUSP 2018.2, segunda edição da série neste ano, irá ocorrer nos dias 9 e 10 de novembro. Planejamos utilizar o serviço desenvolvido para obter uma sugestão dos participantes a serem chamados para o evento. Com isso, poderemos obter feedback do algoritmo criado, na perspectiva de organizador do evento e também dos participantes. Esse feedback seria coletado com os participantes pós-evento, enquanto a perspectiva dos organizadores virá de dentro do grupo *USPCodeLab* em uma reunião pós-evento. Com esses resultados, será elaborado o relatório final da pesquisa para este semestre.

## 6 Método de acompanhamento

Na página <https://victorhmp.github.io/mac0215-blog/> divulgarei semanalmente as discussões realizadas com meus supervisores e os avanços na pesquisa. Além da página, o código sendo desenvolvido estará no repositório do Hacknizer no GitLab: <https://gitlab.com/uspcodelab/projects/hacknizer> contento todo o código sendo desenvolvido.

Meus supervisores serão: Professor Alfredo Goldman, que pode ser contatado pelo email: [gold@ime.usp.br](mailto:gold@ime.usp.br), e o aluno de mestrado Renato Cordeiro Ferreira, que pode ser contatado pelo email: [renatocf@gmail.com](mailto:renatocf@gmail.com).

## Referências

- [Bre01] Leo Breiman. Random Forests. *Machine Learning*, 45(1):5–32, 2001. [4](#)
- [FA90] Karen A. Frenkel and Karen A. Women and computing. *Communications of the ACM*, 33(11):34–46, 11 1990. [2](#)
- [Kay17] Christina Kayastha. Enabling innovation through community and competition. In *2017 IEEE Women in Engineering (WIE) Forum USA East*, pages 1–4. IEEE, 11 2017. [1](#)
- [KPR<sup>+</sup>15] Marko Komssi, Danielle Pichlis, Mikko Raatikainen, Klas Kindstrom, and Janne Jarvinen. What are Hackathons for? *IEEE Software*, 32(5):60–67, 9 2015. [1](#)
- [New15] S. Newman. *Building Microservices: Designing Fine-Grained Systems*. O'Reilly Media, 2015. [2](#)