
Software para análise de variantes genômicas nos
genes BRCA1 e BRCA2 a partir de dados
obtidos em NGS

Mário M. T. B. de Rezende



Software para análise de variantes genômicas nos genes BRCA1 e BRCA2 a partir de dados obtidos em NGS

Mário M. T. B. de Rezende

Supervisor: Euclides Matheucci Jr.

Empresa: QGene Ind. Com. de Equipamentos Para Laboratórios. Ltda.

Monografia de conclusão de curso apresentada ao Instituto de Ciências Matemáticas e de Computação da Universidade de São Paulo para obtenção do título de Bacharel em Ciências de Computação.

USP - São Carlos
Maio de 2014

Dedicatória

Dedico minha monografia aos meus pais, que sempre me apoiaram a perseguir meus sonhos.

Agradecimentos

Agradeço a todas as pessoas que fizeram parte de minha vida até o presente momento.

Resumo

O objetivo deste trabalho é desenvolver um sistema em forma de pipeline para analisar dados produzidos por *Next Generation Sequencing* (NGS). NGS é uma tecnologia recente para sequenciamento de DNA, que produz dados que podem ser analisados para a emissão de um prognóstico sobre o paciente. O sistema visa auxiliar os profissionais da empresa a analisar mutações e emitir o laudo. O sistema está sendo desenvolvido utilizando-se ferramentas e metodologias que permitem fácil prototipação.

Palavras-chaves: rails, TDD, bioinformática, DNA.

Sumário

1	Introdução	1
1.1	Sobre a Empresa	1
1.2	Sobre o Processo Seletivo	2
1.3	Apresentação da monografia	2
2	Planejamento do trabalho	3
2.1	Atividades planejadas para o estágio	3
2.2	Treinamentos planejados para o estágio	4
3	Desenvolvimento do trabalho	5
3.1	Atividades realizadas	5
3.1.1	Software para análise dos genes BRCA1 e BRCA2	5
3.1.2	Programação do robô de pipetação	6
3.1.3	Reimplementação de sistema legado	6
3.2	Problemas resolvidos	6
3.2.1	Software para análise dos genes BRCA1 e BRCA2	7
3.2.2	Programação do robô de pipetação	7
3.2.3	Reimplementação de sistema legado	7
3.3	Técnicas, métodos e tecnologias envolvidas	8
3.3.1	NGS	8
3.3.2	BaseSpace	8
3.3.3	Ruby on Rails	8
3.3.4	RSpec	9
3.3.4.1	Shoulda e Capybara	9
3.3.5	<i>Test Driven Development</i> (TDD)	10
3.4	Impacto	11
3.5	Problemas não resolvidos	11
4	Conclusão	13
4.1	Benefícios para o crescimento profissional	13
4.2	Considerações sobre o curso de graduação	13
4.3	Sugestões para o curso de graduação	14
4.4	Planos para o futuro	15

Lista de Listagens

3.1	Exemplo ilustrativo da sintaxe do RSpec (sem outras gemas)	9
3.2	Exemplo de Shoulda	10
3.3	Exemplo de métodos do Capybara	10

Introdução

1.1 Sobre a Empresa

A QGene Ind. Com. de Equipamentos Para Laboratórios. Ltda. foi fundada em 21 de julho de 2006 e instalada na incubadora CEDIN (Centro de Desenvolvimento de Industrias Nascentes), do sistema de incubadoras da FIESP. A partir de 2009, a empresa foi transferida para nova sede situada na Rua Santa Cruz, 969, em São Carlos, para comportar o crescimento da empresa. A QGene surgiu como uma *spin-off* da DNA Consult Genética e Biotecnologia Ltda., empresa fundada há 12 anos e especializada em análises do DNA, com forte atuação em P&D (possui em seu portfólio três projetos PIPE-FAPESP). Entre janeiro e dezembro de 2007, a QGene abrigou um projeto PIPE-FAPESP relacionado com monitoramento e diagnóstico de vírus em camarões cultivados, tendo recebido cerca de R\$ 150.000,00 da FAPESP, para compra de insumos, equipamentos e bolsas.

A QGene atua na área de genética molecular, dedicando-se ao P&D, produzindo e comercializando *kits* para análise de DNA e soluções em saúde humana, que representem inovações tecnológicas e que substituam a importação de produtos.

A QGene é uma empresa de pequeno porte que goza de parcerias com diversas empresas, como a DNA Consult e MD Genetics. Por ser uma empresa de pequeno porte, a QGene também está sempre procurando e atuando em nichos que não eram o foco inicial da empresa, além de seus produtos consolidados há anos no mercado. Atualmente, a QGene está produzindo soluções de bioinformática com foco clínico, uma vez que os softwares existentes, apesar de robustos, não apresentam facilidade de uso para médicos e muitas vezes são softwares abrangentes que não apresentam soluções clínicas.

A empresa valoriza muito seus funcionários e não mede gastos para aprimorá-los cada vez mais. A empresa entende que os funcionários, sejam os contratados como os estagiários, podem ajudar a QGene a estar sempre se reinventando como empresa e atacando novos nichos ou setores do mercado.

1.2 Sobre o Processo Seletivo

O processo seletivo por qual passei foi apenas uma entrevista. Fui entrevistado por dois sócios da empresa. Um deles possui doutorado na área de biologia molecular, e o outro sócio trabalhava no Banco do Brasil, no cargo de presidência, mas agora está aposentado. A entrevista não continha perguntas técnicas sobre computação, especialmente pois a QGene não possuía, até então, um setor de TI. O que os entrevistadores avaliaram foi algo mais subjetivo e de interesse deles: capacidade de trabalho em equipe, interesse em trabalhar com uma área interdisciplinar (no caso, bioinformática), capacidade de trabalhar de forma autônoma (ou seja, sem uma pessoa de TI me liderando), e proatividade.

1.3 Apresentação da monografia

Os próximos capítulos irão retratar de forma mais detalhada as atividades realizadas por mim durante o estágio.

No Capítulo 2, descrevo brevemente como foi o planejamento das atividades. É apresentado também como foi meu treinamento.

No Capítulo 3, apresento as atividades realizadas e ferramentas sendo utilizadas para desenvolver tais atividades.

No último capítulo, são levantadas as conclusões sobre o estágio e a relação com o curso de graduação, indicando melhorias e perspectivas para o futuro, tanto para o curso quanto para o trabalho.

Planejamento do trabalho

2.1 Atividades planejadas para o estágio

A atividade planejada é a implementação de um *pipeline* para análise de dados de NGS e gerar laudos automaticamente. Os equipamentos de NGS [6] produzem *output* em um formato padrão que pode facilmente passar por um processo de *parsing* por outras ferramentas. O objetivo deste sistema é que ele faça processamentos de tal *output* com o intuito de melhorar a confiabilidade do sequenciamento (eliminar falsos negativos, especialmente) e melhorar a visualização dos dados para ajudar o usuário a tomar decisões pertinentes sobre a emissão do laudo.

Como a bioinformática voltada para análises genéticas é uma área nova para mim, o desenvolvimento do sistema computacional não contou com um cronograma. Ao invés disso, o desenvolvimento e o estudo sobre áreas da bioinformática que não domino (áreas pertinentes a biologia e genética) são realizados de forma altamente dinâmica, à medida que surge a necessidade.

A única atividade que foi planejada foi a minha participação no curso de verão de bioinformática ministrado na Faculdade de Medicina de Ribeirão Preto (FMRP) [3]. O curso foi bem proveitoso e pude melhorar em muito minha comunicação com a equipe da empresa e meu domínio de tal área. O curso, além de abordar aspectos teóricos de genética, também continha aulas práticas onde ensinaram a utilizar a linguagem R e o *framework* Bioconductor, um *framework* com vários métodos para realizar *parsing* de arquivos de sequenciamento, analisar estatísticas e verificar mutações. Esse *framework* é muito poderoso, porém não está sendo utilizado no desenvolvimento do sistema de análise pois é mais voltado para análises de grande porte (análises de populações), o que contrasta com os requisitos de análise clínica. Esse con-

traste não justifica, até o momento, a utilização de mais uma linguagem e mais um *framework* no sistema, pois pode prejudicar o tempo de treinamento de novos funcionários e/ou a manutibilidade do sistema.

2.2 Treinamentos planejados para o estágio

O único treinamento planejado foi a participação no curso da FMRP. Demais treinamentos são realizados quando se fazem necessários.

Além de cursos, meu chefe está sempre disponível para que eu possa tirar dúvidas sobre os assuntos de genética e, inclusive, me convida a participar de eventuais palestras por ele ministradas.

Desenvolvimento do trabalho

3.1 Atividades realizadas

Nesta seção são descritas as atividades que realizei durante o estágio. Para as duas atividades de desenvolvimento de software descritas abaixo, a dinâmica de trabalho empregada foi a mesma: TDD. Essa metodologia foi escolhida pois facilita em muito a comunicação com os solicitantes dos softwares e é extremamente útil para projetos cujos requisitos são alterados constantemente.

3.1.1 Software para análise dos genes BRCA1 e BRCA2

Os genes BRCA1 e BRCA2 são responsáveis, entre outras coisas, pela capacidade da célula de corrigir possíveis erros em seu DNA. Por esse motivo, mutações patogênicas em tais genes tem um grande impacto na vida do paciente. Uma célula que não consegue se corrigir tem alta probabilidade de gerar células cancerosas ao se reproduzir.

A Agência Nacional de Saúde (ANS) emitiu recentemente uma resolução [2] que obriga todos os planos de saúde no Brasil a darem cobertura para sequenciamentos de BRCA1 e BRCA2, entre outras doenças. Tais sequenciamentos devem ser solicitados por médicos, e podem ser solicitados tanto por medida preventiva como para confirmar diagnósticos.

A atividade planejada é um sistema web com a finalidade de facilitar a requisição de tais exames e emissão de laudo preliminar. Dessa forma, o sistema ajudaria e disponibilizaria informações para que o médico escolha o melhor exame pertinente a cada doença (a princípio, relacionada ao BRCA1 e BRCA2) e solicite o exame. E, no resultado, teria informações pertinentes para construir e emitir o laudo final.

Outro requisito do sistema é que ele faça o processamento de forma automática dos dados após o sequenciamento. No NGS, utilizam-se reagentes químicos e enzimáticos para isolar e marcar as áreas do DNA que devem ser sequenciadas. Após o sequenciamento, esses reagentes, se não forem eliminados do *output* da máquina de sequenciamento, podem causar ruídos na etapa de análise dos dados. Tais ruídos são causados porque alguns dos reagentes são trechos pequenos de DNA que, por construção, são idênticos ao DNA de referência. Se o DNA sequenciado apresenta mutações especificamente nas regiões dos reagentes, a máquina acusará que em alguns ciclos do sequenciamento detectou tais mutações (no DNA original) e em outros não detectou (no reagente). Isso diminui a confiabilidade do sequenciamento e influencia a análise do DNA para a emissão do laudo.

Por fim, ao longo prazo, o sistema deve armazenar os resultados dos sequenciamentos para que a empresa possa minerar dados e possivelmente evitar que um mesmo paciente realize o mesmo sequenciamento mais de uma vez (por exemplo, mesmo gene e doenças diferentes). Dados os requisitos de tal sistema, planejei ele em forma de *pipeline* para receber solicitações de exames, fazer o processamento dos dados do sequenciamento e armazenar no banco de dados. O módulo de mineração de dados será um módulo futuro.

3.1.2 Programação do robô de pipetação

Fui solicitado a programar um robô [1] para pipetar reagentes. A empresa vende *kits*, que são misturas de compostos químicos e enzimáticos (por exemplo, para realizar testes de paternidade). Os compostos precisam ser pipetados e misturados em volumes precisos. Atualmente, a empresa conta com funcionários no laboratório que pipetam manualmente tais *kits*, mas deseja utilizar o robô pois tal tarefa é muito repetitiva.

O robô (um braço mecânico) é programado através de um software proprietário de fácil utilização. Basta indicar o tubo de origem, o tubo de destino e a quantidade de cada reagente que deve ser pipetado. A sequência de comandos pode ser salva, podendo ser repetida mais vezes.

3.1.3 Reimplementação de sistema legado

A atividade na qual estou trabalhando atualmente é a reimplementação de um sistema legado. O sistema é responsável por cadastrar clientes e animais e fazer o acompanhamento e marcha analítica dos exames de paternidade. O objetivo da reimplementação é utilizar tecnologias mais robustas para o sistema (PostgreSQL, por exemplo), disponibilizar o sistema como um sistema web e utilizar o serviço de *storage* da Amazon.

3.2 Problemas resolvidos

Nesta seção, são descritos problemas enfrentados em cada atividade e que consegui resolver.

3.2.1 Software para análise dos genes BRCA1 e BRCA2

Os dados gerados pelo NGS (padronizados segundo [4]) precisam ser processados antes de serem analisados, com a finalidade de aumentar a qualidade dos resultados (como descrito na seção anterior). Até o presente momento, o sistema ainda está em fase de prototipação. NGS é uma forma de sequenciamento recente e ainda está sendo estudada pela empresa. Dessa forma, os protótipos desenvolvidos estão ajudando a entender melhor os conceitos, os gráficos gerados pela *cloud* do fabricante, e os formatos de arquivos para representar dados de sequenciamento. Esse é um passo muito importante para que o laudo emitido pelo médico seja baseado nas informações mais fidedignas possíveis.

Um artifício utilizado pela empresa para avaliar se os protótipos estão produzindo os dados esperados foi a análise de cinco pacientes. O conjunto de pacientes que participaram foi formado por uma mãe, que já teve câncer de ovários, e suas quatro filhas, que, até então, estão saudáveis; contudo, espera-se que venham a ter risco mais elevado de desenvolverem câncer de ovários dado o histórico de sua mãe.

Os protótipos detectaram a mutação na mãe (uma inserção, que muda o quadro de leitura dos aminoácidos, o que a torna é muito patogênica). Nas filhas, nada foi detectado, apesar disso ser um resultado inesperado.

3.2.2 Programação do robô de pipetação

O problema que o robô apresenta é o erro na pipetação. A ponteira de pipeta utilizada pelo robô armazena uma quantidade muito pequena de líquido (cinquenta microlitros). Isso gerou a dificuldade de pipetar alguns reagentes que continham substâncias viscosas em sua composição, como o glicerol. Líquidos viscosos podem formar uma bolha na ponta da pipeta e/ou parte do líquido pode ficar grudada dentro da pipeta.

Para eliminar esse problema, o que fiz foi pipetar os solventes antes. Após os solventes estarem no tubo de destino, programei o robô para pipetar os reagentes viscosos e, logo em seguida, realizar uma mistura. O comando de mistura faz com que o robô pipete e solte os reagentes do tubo repetidamente. Assim, os reagentes que não eram expelidos, apesar de continuarem presos na pipeta, eventualmente eram diluídos o suficiente para serem expelidos sem perda de precisão.

3.2.3 Reimplementação de sistema legado

Essa atividade não apresentou (até então) problemas, uma vez que é apenas uma reimplementação de um sistema ao qual tenho acesso para poder realizar engenharia reversa.

3.3 Técnicas, métodos e tecnologias envolvidas

Aqui, são apresentados alguns detalhes sobre o desenvolvimento do sistema. As ferramentas usadas no desenvolvimento são bem documentadas e apresentam uma abundância de exemplos na internet. A descrição detalhada de cada ferramenta está fora do escopo deste trabalho.

3.3.1 NGS

NGS [6] é uma técnica para sequenciamento de DNA feita por leituras que utilizam um *laser* e alguns reagentes que emitem diferentes comprimentos de onda de luz. Assim sendo, após um ciclo de leitura, o equipamento tem uma imagem com cores que representam os diferentes nucleotídeos do DNA sequenciado, e aplica algoritmos de processamento de imagem para mapear tal imagem em um arquivo texto para fácil manipulação.

NGS é uma técnica um tanto quanto recente, e pode variar em alguns aspectos dependendo do fabricante do equipamento. O nome NGS é usado para representar os sequenciamentos que podem fazer leituras de trechos grandes de DNA.

Como toda técnica, essa possui certas desvantagens. O resultado é muito preciso, porém existem técnicas que produzem resultados mais fidedignos quando o trecho de DNA sequenciado é pequeno. Além disso, essa técnica, como dito antes, produz resultados viesados pois utiliza-se de reagentes que podem introduzir falsos negativos. Por isso, o processo deve ser muito bem acompanhado e os dados produzidos precisam ser tratados após a fase de sequenciamento.

3.3.2 BaseSpace

BaseSpace [5] é a *cloud* do fabricante do equipamento de NGS sendo utilizado pela empresa atualmente. Após o término do sequenciamento, o equipamento realiza o alinhamento com o DNA de referência e envia os dados brutos e dados de alinhamento para a *cloud*. Essa *cloud* possui alguns aplicativos web que podem ser utilizados para analisar os dados, mas tais análises são um tanto quanto genéricas (por exemplo, não consideram a possibilidade de ruídos devido aos reagentes).

De qualquer forma, a *cloud* ainda é utilizada pois armazena os dados e, eventualmente, a empresa cogita a possibilidade de disponibilizar o sistema que estou desenvolvendo nessa *cloud* e cobrar um valor módico por análise.

3.3.3 Ruby on Rails

O *framework* usado para a implementação do sistema é Ruby on Rails (RoR) [8]. RoR é um *framework* poderoso para construção de aplicativos web. RoR implementa de forma simples e elegante o modelo *Model-view-controller* (MVC).

RoR foi escolhido por alguns motivos. É um *framework* com alto poder de prototipação, abstrai totalmente a parte de banco de dados (SQL só é necessário para otimizar certas consultas), conta com uma comunidade muito ativa, pode ser facilmente configurado para utilizar os serviços de *storage* da Amazon, e é um *framework* que incentiva o desenvolvedor a escrever casos de teste a fim de ter maior segurança durante a codificação e refatoração do código.

3.3.4 RSpec

RSpec [9] é uma biblioteca (no contexto de RoR, também chamadas de *gemas*) projetada para a descrição e implementação de **casos de teste** (CT). A sintaxe do RSpec é muito parecida com a sintaxe da linguagem Ruby ao mesmo tempo que se parece muito com a língua inglesa, como exemplificado na Listagem 3.1. Tal listagem exemplifica também a estrutura básica dos casos de teste: **describe**, que define o sujeito do caso de teste (no caso, um objeto da classe **Pdf**), **specify**, que é um caso de teste, e, dentro do caso de teste, as expectativas (**should_not** e **expect**), que, em conjunto com os *matchers* (**be_a_new_record**, **raise_error**), verificam o comportamento do sistema. Em outras palavras, a listagem exhibe um caso de teste que testa um **pdf** que possui título. Tal **pdf** deve estar salvo no banco de dados e, quando tentarmos inseri-lo novamente no banco de dados, o sistema deve levantar uma exceção indicando que o título do **pdf** já está sendo utilizado por outra tupla. Isso faz com que os CT sejam muito descritivos e concisos, tanto para o desenvolvedor quanto para o cliente, diminuindo a dificuldade de comunicação.

É possível escrever CT para os modelos, controladores, e até mesmo para as *views* do sistema. Com a disponibilidade de uma gema tão robusta e bem projetada, tornou-se uma convenção a utilização de TDD no desenvolvimento de sistemas escritos em RoR.

Listing 3.1: Exemplo ilustrativo da sintaxe do RSpec (sem outras gemas)

```
1
2 describe Pdf do
3     specify "when title is not unique" do
4         pdf.should_not be_a_new_record
5         expect { Pdf.create!(pdf.attributes) }.to raise_error
              ActiveRecord::RecordInvalid
6     end
7 end
```

3.3.4.1 Shoulda e Capybara

Shoulda [10] e Capybara [7] são gemas que incrementam a funcionalidade do RSpec.

Shoulda possui métodos para testar controladores ou modelos com apenas uma linha de código. Por exemplo, existe um método (Listagem 3.2) que verifica se a tabela correspondente a um modelo foi criada no banco de dados, se os campos da tabela estão corretos, e se as chaves

estrangeiras estão referenciando as chaves primárias nas tabelas corretas. Essa gema não foi utilizada no início do estágio, mas quando foi adotada no projeto, o tamanho dos arquivos de CT foi reduzido consideravelmente. Isso foi muito importante, pois os protótipos não contam com um documento de requisitos, apenas com os CT, e pode-se configurar o Rspec para que ele solte um *output* em formato de documentação. Deixar a implementação dos CT concisa reduziu em muito o tamanho da documentação gerada pelo Rspec.

Listing 3.2: Exemplo de Shoulda

```
1
2 describe Post do
3     it { should belong_to :user }
4 end
```

Capybara é uma gema com métodos para se realizar testes de integração. Testes de integração são testes que simulam um usuário utilizando o sistema para determinadas tarefas. Por exemplo, alguns métodos da gema (Listagem 3.3) permitem simular um usuário que visita uma página, preenche um formulário, e pressiona o botão de submissão do formulário. Se, após tudo isso, o usuário for redirecionado para a página correta e os dados forem salvos no banco de dados, então o aplicativo passa nesse caso de teste.

Listing 3.3: Exemplo de métodos do Capybara

```
1
2 describe "signup" do
3     visit signup_path
4     fill_in "Name", with: "Example User"
5     fill_in "Email", with: "user@example.com"
6     fill_in "Password", with: "foobar"
7     fill_in "Confirmation", with: "foobar"
8
9     # expectations in plain Rspec or Shoulda ...
10 end
```

3.3.5 TDD

TDD é uma metodologia de desenvolvimento onde os CT são escritos antes da aplicação ser implementada. Tais casos de teste guiam o processo de desenvolvimento. Certos CT são escritos consultando-se o cliente para definir o comportamento do sistema, de tal forma que tais casos de teste possam ser vistos como uma forma de documento de requisitos extremamente flexível do sistema. Os demais CT são escritos para evitar/detectar *bugs* e proporcionar maior segurança e confiança ao desenvolvedor quando manutenções ou alterações forem necessárias.

O aspecto de evitar/detectar *bugs* é extremamente importante uma vez que RoR implementa certas medidas de segurança. Detectar e corrigir *bugs* nos modelos são tarefas simples, pois os modelos implementam apenas as lógicas de negócios e tabelas do banco de dados. Mas tais tarefas podem ser muito complicadas nos controladores, que são responsáveis pelo comportamento do sistema como um todo e também implementam diretivas de segurança. Uma diretiva de segurança em particular evita o problema de *mass assignment*, e caso o desenvolvedor não configure tal diretiva, pode acabar com dados espúrios no banco de dados. Tal diretiva é silenciosa, ou seja, não emite erros/exceções caso o sistema receba um ataque de *mass assignment*.

3.4 Impacto

A respeito do sistema para análise, os profissionais terão em mãos uma ferramenta valiosa para ajudá-los a analisar os dados do sequenciamento. Utilizando-se a ferramenta, espera-se que os prognósticos sejam mais precisos e possam ser usados para prevenir casos clínicos antes que estes ocorram.

Sobre as demais atividades, espera-se que os custos operacionais da empresa diminuam e tal redução possa ser repassada para os clientes.

3.5 Problemas não resolvidos

Não existe versão final do sistema de análise ainda. Apenas protótipos que estão sendo usados para explorar possíveis funcionalidades e outros aspectos do problema.

Sobre o robô, apesar de resolver o problema de reagentes viscosos, ainda estamos verificando se é viável sua utilização, uma vez que existe uma variância na quantidade de reagentes pipetados devido aos erros de medições do equipamento. Esse erro é especificado em seu manual, mas a empresa não sabe ainda se tal erro pode ser tolerado nas reações, já que as reações devem ser muito precisas. Como exemplo, a concentração de enzimas nas reações deve ser da ordem de picomols; concentrações mais grosseiras ou finas podem influenciar na cinética enzimática, proporcionando erros nas reações.

Conclusão

4.1 Benefícios para o crescimento profissional

A oportunidade de trabalhar sem ser liderado por alguém da área de TI me interessou muito, pois vi isso como uma forma de adquirir experiência em diversas áreas, desde liderança (atualmente, a empresa contratou um outro estagiário para que me ajude e seja meu subordinado), até coisas mais específicas da computação, como a parte de conversar com o cliente, fazer o levantamento dos requisitos do sistema, e tomar decisões sobre linguagens de programação e metodologia de desenvolvimento.

Outro aspecto interessante e útil de estar trabalhando na empresa é a capacidade de aumentar meu *networking*. Estou envolvido diariamente com diversas pessoas, desde economistas, biólogos, médicos, publicitários, e até mesmo, como mencionado anteriormente, com o ex-presidente do Banco do Brasil. Tal *networking* diversificado me ajuda cada vez mais a aprimorar meu perfil de empreendedor e acredito que seja uma excelente forma de treinar a habilidade de definir produtos com futuros clientes e trabalhar em áreas interdisciplinares.

4.2 Considerações sobre o curso de graduação

O curso forneceu as bases teóricas para que o estágio fosse desenvolvido. Ainda que minha experiência no mercado de trabalho seja algo recente, as bases teóricas me facilitaram bastante a aquisição de novos conhecimentos com finalidade de elevar o meu trabalho desenvolvido durante o estágio ao nível profissional.

4.3 Sugestões para o curso de graduação

As disciplinas de engenharia de software que cursei não abordavam TDD. Acredito que esta metodologia de desenvolvimento seja muito importante e interessante, e deveria ser abordada.

Criptografia é um assunto pouco abordado no curso. Estudei criptografia apenas nas disciplinas de redes. Durante o estágio, foi necessário utilizar criptografia em outras camadas da aplicação, como, por exemplo, no banco de dados. Acredito que criptografia seja um assunto que deva ser abordado também nas disciplinas de bancos de dados.

As disciplinas de matemática não foram muito proveitosas. Não usei cálculo nem geometria analítica, por exemplo. Acredito que tais disciplinas ainda sejam muito importantes, pois fornecem bases lógicas e são requisitos para outras disciplinas. Mas a carga de matemática do curso, em minha opinião, deveria incluir mais disciplinas de matemática que possam ser usadas em computação diretamente do que disciplinas que são usadas em contextos mais específicos. Considero que cálculo seja uma disciplina de contexto específico, e disciplinas como matemática discreta e probabilidade sejam mais aplicáveis em áreas diversas da computação. Tive a oportunidade de cursar outras disciplinas do curso de matemática, fornecidas como optativas para a computação. Algumas delas foram Álgebra 1, Elementos de Matemática, Tópicos de Matemática Elementar, e Tópicos de Otimização Combinatória. O conteúdo de tais disciplinas é também fornecido no curso de computação, porém de forma muito mais resumida (nas disciplinas de matemática discreta). Acredito que as disciplinas de matemática discreta não deveriam ser tão condensadas/resumidas.

Outra disciplina que deveria receber maior atenção é a de empreendedores em informática. Considero que a linha divisória entre um funcionário e um empreendedor seja muito subjetiva. Eu, como funcionário, muitas vezes recorro a conceitos vistos na disciplina de empreendedorismo. Tal disciplina deveria ser obrigatória no curso.

As disciplinas preparatórias para a maratona de programação também são muito importantes. Pude ganhar mais experiência na parte de modelar o problema, testar, implementar e otimizar a implementação para que o programa seja executado de forma rápida. Nessas disciplinas, também melhorei minha habilidade de trabalhar em equipe, pois o time para a maratona é composto por três pessoas. Muitos dos conceitos absorvidos em tais disciplinas não são aplicados extensivamente, pois a prioridade no trabalho sendo desenvolvido é manutenibilidade do código, ao invés de código extremamente eficiente. Porém, tais disciplinas me deram uma melhor capacidade de resolver os problemas que podem surgir, e acredito que foram as responsáveis por interconectar as matérias teóricas com as matérias práticas.

A única reclamação que tenho sobre o curso é sobre a duração das aulas. A maioria das aulas tem duração de duas horas. Isso já é uma duração longa, e outras aulas podem ter até quatro horas de duração. É um contraste muito grande com as aulas do Massachusetts Institute of Technology, por exemplo, que tem duração de no máximo uma hora.

Em suma, estou satisfeito com o curso. Porém, tal satisfação só foi atingida porque busquei estudar assuntos extracurriculares. Infelizmente, não pude ver isso em muitos colegas, que se formaram antes de mim e que ainda assim sentiram que o curso não foi proveitoso. Acredito que seja possível reestruturar o curso de tal forma que futuros formandos entrem mais confiantes no mercado de trabalho.

4.4 Planos para o futuro

Pretendo continuar trabalhando na empresa. Acredito que seja um lugar onde eu possa adquirir muita experiência, principalmente em como trabalhar conjuntamente com profissionais de outras áreas.

Referências

- [1] Arise Biotech. EzMate 401s. http://www.arisebio.com.tw/product_EzMate_401s.asp. Acessado: 2014-06-01.
- [2] Agência Nacional de Saúde. ANS amplia cobertura obrigatória para 29 doenças genéticas. <http://www.ans.gov.br/a-ans/sala-de-noticias-ans/consumidor/2316-ans-amplia-cobertura-obrigatoria-para-29-doencas-geneticas->. Acessado: 2014-06-01.
- [3] FMRP. 10th Summer Bioinformatics Course. <http://bioinformatics.fmrp.usp.br/curso2014/>. Acessado: 2014-05-28.
- [4] 1000 Genomes. SAM/BAM and related specifications. <https://github.com/samtools/hts-specs>. Acessado: 2014-05-28.
- [5] Illumina. BaseSpace Genomics Cloud Computing. <https://basespace.illumina.com/home/index>. Acessado: 2014-06-01.
- [6] Illumina. Next-Generation Sequencing Technology. <http://www.illumina.com/technology/next-generation-sequencing.ilmn>. Acessado: 2014-05-28.
- [7] Jonas Nicklas. Capybara GitHub page. <https://github.com/jnicklas/capybara>. Acessado: 2014-05-28.
- [8] Ruby on Rails. Ruby on Rails GitHub page. <https://github.com/rails/rails/>. Acessado: 2014-05-28.
- [9] RSpec. RSpec GitHub page. <https://github.com/rspec/rspec-rails/>. Acessado: 2014-05-28.
- [10] Thoughtbot. Shoulda GitHub page. <https://github.com/thoughtbot/shoulda/>. Acessado: 2014-05-28.