
Documentação

Projeto de Software

Professor: Rohit Gheyi

Equipe 4:

Amanda Ribeiro Azevedo De Souza

Maria Cecylia Dos Santos Balbino

Bernard Dantas Odon

Mariane Maira Santos Zeitouni

Daniel Dantas Fonseca

Nathan Andrade Henriques

Felipe De Souza Siqueira

Pedro Aires Cabral

Jose Nilton Da Silva Lima Junior

Pedro Rafael Pereira Damiao

Luanna Leonel De Farias Silva

Pedro Raimundo Dos Santos Neto

Maely Silva Brandao

Raiani Idalino Pontes

Sistema Novo

1. Introdução

1.1. Motivação

O problema é...	Pessoas que precisam entrar no mercado de trabalho mas não possuem experiência.
Que afeta...	Pessoas que não possuem o acesso a oportunidades de primeiro emprego continuam à margem do mercado de trabalho.
O impacto disto é...	Impossibilidade de ascensão financeira e ciclo vicioso onde não se é contratado por não ter experiência e não ter experiência porque não é contratado.
A solução seria...	Apresentar ao mercado experiências pessoais obtidas pela pessoa através de pequenos trabalhos formais ou informais.

O problema é...	Existe uma dificuldade de oportunidade para pessoas fazerem uma renda extra com trabalhos temporários (costureira, carregador, etc...)
Que afeta...	Trabalhadores informais e qualquer um que tenha conhecimento e deseja ganhar uma renda extra trabalhando informalmente.
O impacto disto é...	Permite que pessoas complementem sua renda, melhorando sua qualidade de vida.
A solução seria...	Apresentar uma ferramenta que permite a divulgação de oportunidades de trabalho e de ofertas de serviços.

O problema é...	Existe uma dificuldade para pessoas que necessitam de algum serviço temporário e não conhecem ninguém que os possam ajudar.
Que afeta...	Pessoas que necessitam de algum tipo de ajuda ou serviço e estariam dispostas a pagar.
O impacto disto é...	Permite facilitar o dia-a-dia das pessoas por meio da terceirização de serviços.

A solução seria...

Apresentar uma ferramenta que permite a divulgação de oportunidades de trabalho e de ofertas de serviços.

1.2. Visão da Solução

Facilitar a conexão entre pessoas que buscam por trabalho e pessoas que precisam de serviços pequenos.

1.3. Visão Geral do Documento

Este documento está organizado da seguinte forma:

- Na Seção 1, apresentou-se a motivação e o problema que levou à construção desse trabalho;
- Na Seção 2 estão apresentados aos requisitos organizacionais do produto;
- Na seção 3 é apresentado o processo organizacional seguido pela equipe, como cerimônias, práticas e estratégias.
- Na seção 4 estão apresentados todos os requisitos funcionais e não funcionais necessários para o projeto.
- Na seção 5 expomos o projeto arquitetural. (não finalizada)
- Na seção 6 é apresentada a especificação formal do projeto. (não finalizada)
- Na seção 7 estão disponíveis os links referentes a implementação do projeto. (não finalizada)
- Na seção 8 são apresentados os testes. (não finalizada)
- Na seção 9 tem-se a introdução do projeto real.
- Na seção 10 estão discorridos pontos sobre a qualidade do projeto real.
- Na seção 11 testes relacionados ao projeto real. (não finalizada)
- Na seção 12 estão tópicos relacionados à evolução do projeto real. (não finalizada)
- Na seção 13 estão as conclusões do projeto inteiro. (não finalizada)

2. Planejamento

2.1. Introdução

No plano de projeto, apresentamos a descrição do Projeto Home-Service que visa oferecer uma página virtual que permite que pessoas que precisam de uma renda extra porque não conseguem encontrar um emprego formal ou porque precisam de mais renda, anunciem seus serviços e sejam localizadas por pessoas que precisam de serviços rápidos. Podemos citar como exemplos pessoas que lavam roupas, lavadores de automóveis, montadores de móveis entre outros serviços que geralmente não possuem oportunidades formais para pessoas sem experiência.

2.2. Gerência do Tempo

O cronograma feito abaixo envolve a equipe inteira; as subdivisões das seções foram divididas entre os subgrupos da equipe, são eles:

- Backend: Pedro Aires, Pedro Raimundo, Maely, Pedro Rafael, Amanda.
- Frontend: Bernard, Balbino, Raiani, Nilton.
- Documentação: Daniel, Felipe, Luanna, Mariane.
- Testes: Maely, Mariane, Balbino, Nathan.
- Projeto Real: Felipe, Mariane, Nathan, Luanna, Daniel.
- Gerentes: Bernard, Pedro Aires, Felipe e Mariane.

Como as atividades foram feitas em conjunto decidimos por fazer o cronograma desta maneira. O tempo de cada atividade está estipulado na quarta e quinta coluna do cronograma, a equipe não trabalhou nos finais de semana e os integrantes trabalharam 3 horas por dia; para as atividades posteriores essa métrica também será considerada. Essa escolha do tempo que cada integrante irá trabalhar por dia foi feita em conjunto, baseando-se no tempo de cada integrante. Se o integrante X não conseguir cumprir sua carga horária por dia, este deverá trabalhar mais horas em outros dias para que o cronograma seja cumprido.

Até o momento (segunda entrega) a equipe conseguiu cumprir o cronograma estabelecido, apesar de os integrantes terem se atrasado em uma semana em específico a qual foram realizadas atividades externas ao projeto. Porém, como foi dito anteriormente, a equipe conseguiu trabalhar por mais tempo em outros dias e continuar dentro dos prazos estipulados.

		Name	Duration	Start	Finish	Predecessors	Resources	Time Responsável
1		1.1 Motivação	1day?	24/10/2022	24/10/2022			Equipe Documentação
2		1.2 Visão da Solução	1day?	24/10/2022	24/10/2022			Equipe Documentação
3		1.3 Visão Geral do Documento	1day?	24/10/2022	24/10/2022			Equipe Documentação
4		2.1 Introdução do Planejamento	1day?	24/10/2022	24/10/2022			Equipe Documentação
5		2.4. Gerência de Riscos	2days?	21/10/2022	24/10/2022			Equipe Documentação
6		9 Projeto Real - Introdução	1day?	24/10/2022	24/10/2022			Equipe Projeto Real
7		10 Projeto Real - Qualidade	10days?	24/10/2022	04/11/2022			Equipe Projeto Real
8		2.2. Gerência do Tempo	4days?	25/10/2022	28/10/2022			Equipe Documentação
9		2.3. Gerência do Custo	3days?	25/10/2022	27/10/2022			Equipe Documentação
10		4.1. Requisitos Não-Funcionais	8days?	28/10/2022	08/11/2022			Equipe Front e Back
11		4.2. Requisitos Funcionais	8days?	28/10/2022	08/11/2022			Equipe Front e Back
12		2.5. Recursos do Projeto	2days?	31/10/2022	01/11/2022			Equipe Documentação
13		3. Processo	1day?	01/11/2022	01/11/2022			Equipe Documentação
14		4.3. Prototipação de Telas	9days?	08/11/2022	18/11/2022			Equipe Front-end
15		2.6 Comunicação	1day?	18/11/2022	18/11/2022			Equipe Documentação
16		5. Projeto Arquitetural	1day?	21/11/2022	21/11/2022			Equipe Back-end
17		6. Especificação Formal	5days?	21/11/2022	25/11/2022			Equipe Documentação e Testes
18		7. Implementação	25days?	10/11/2022	14/12/2022			Equipe Back e Front
19		8. Testes	22days?	17/11/2022	16/12/2022			Equipe de Testes
20		11. Projeto Real - Testes	10days?	21/11/2022	02/12/2022			Equipe Projeto Real
21		12. Projeto Real - Evolução	6days?	05/12/2022	12/12/2022			Equipe Projeto Real
22		13. Conclusões	1day?	18/11/2022	18/11/2022			Equipe Gerentes

Figura 1 – Cronograma do projeto inicial

2.2.1. Descrição das Estimativas

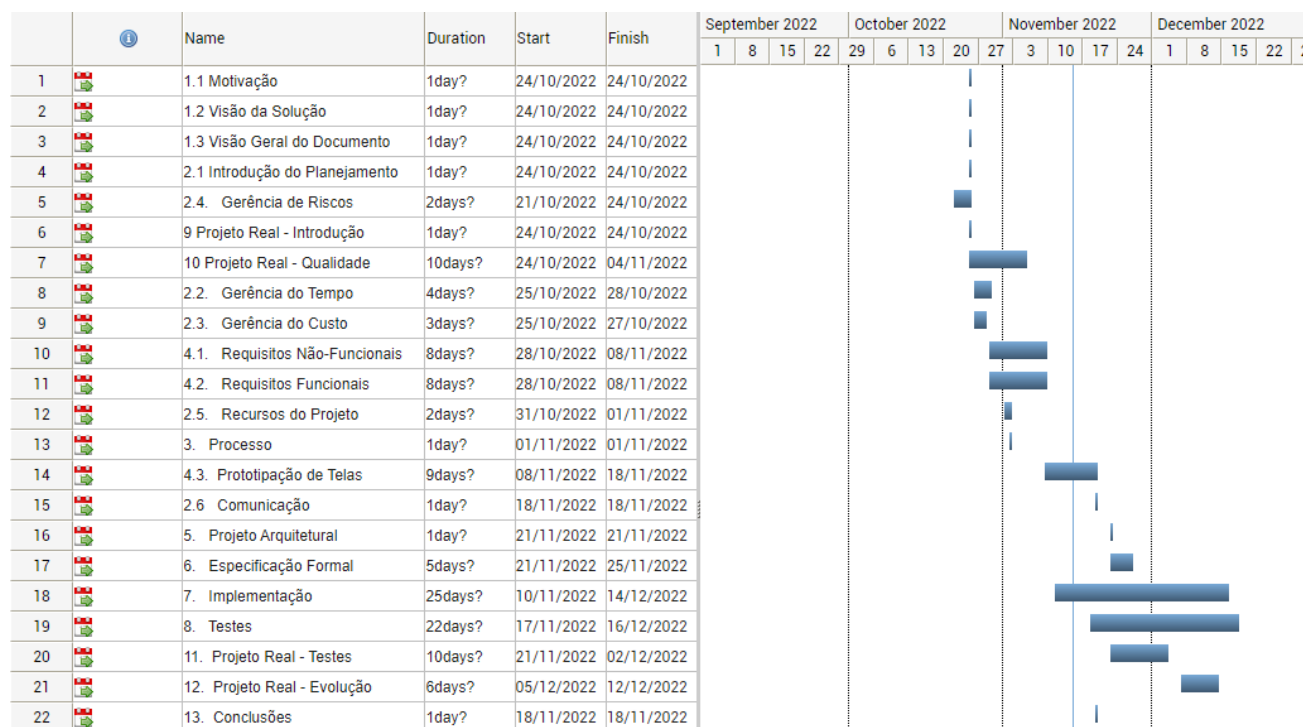


Figura 2 – Cronograma do projeto inicial
























		Name	Duration	Start	Finish	Predecessors	Resources	Time Responsável
1		1.1 Motivação	1day?	24/10/2022	24/10/2022			Equipe Documentação
2		1.2 Visão da Solução	1day?	24/10/2022	24/10/2022			Equipe Documentação
3		1.3 Visão Geral do Documento	1day?	24/10/2022	24/10/2022			Equipe Documentação
4		2.1 Introdução do Planejamento	1day?	24/10/2022	24/10/2022			Equipe Documentação
5		2.4. Gerência de Riscos	2days?	21/10/2022	24/10/2022			Equipe Documentação
6		9 Projeto Real - Introdução	1day?	24/10/2022	24/10/2022			Equipe Projeto Real
7		10 Projeto Real - Qualidade	10days?	24/10/2022	04/11/2022			Equipe Projeto Real
8		2.2. Gerência do Tempo	4days?	25/10/2022	28/10/2022			Equipe Documentação
9		2.3. Gerência do Custo	3days?	25/10/2022	27/10/2022			Equipe Documentação
10		4.1. Requisitos Não-Funcionais	8days?	28/10/2022	08/11/2022			Equipe Front e Back
11		4.2. Requisitos Funcionais	8days?	28/10/2022	08/11/2022			Equipe Front e Back
12		2.5. Recursos do Projeto	2days?	31/10/2022	01/11/2022			Equipe Documentação
13		3. Processo	1day?	01/11/2022	01/11/2022			Equipe Documentação
14		4.3. Prototipação de Telas	9days?	08/11/2022	18/11/2022			Equipe Front-end
15		2.6 Comunicação	1day?	18/11/2022	18/11/2022			Equipe Documentação
16		5. Projeto Arquitetural	1day?	21/11/2022	21/11/2022			Equipe Back-end
17		6. Especificação Formal	5days?	21/11/2022	25/11/2022			Equipe Documentação e Testes
18		7. Implementação	25days?	10/11/2022	14/12/2022			Equipe Back e Front
19		8. Testes	22days?	17/11/2022	16/12/2022			Equipe de Testes
20		11. Projeto Real - Testes	10days?	21/11/2022	02/12/2022			Equipe Projeto Real
21		12. Projeto Real - Evolução	6days?	05/12/2022	12/12/2022			Equipe Projeto Real
22		13. Conclusões	1day?	18/11/2022	18/11/2022			Equipe Gerentes

Figura 3 – Alocação de Pessoas e Tarefas

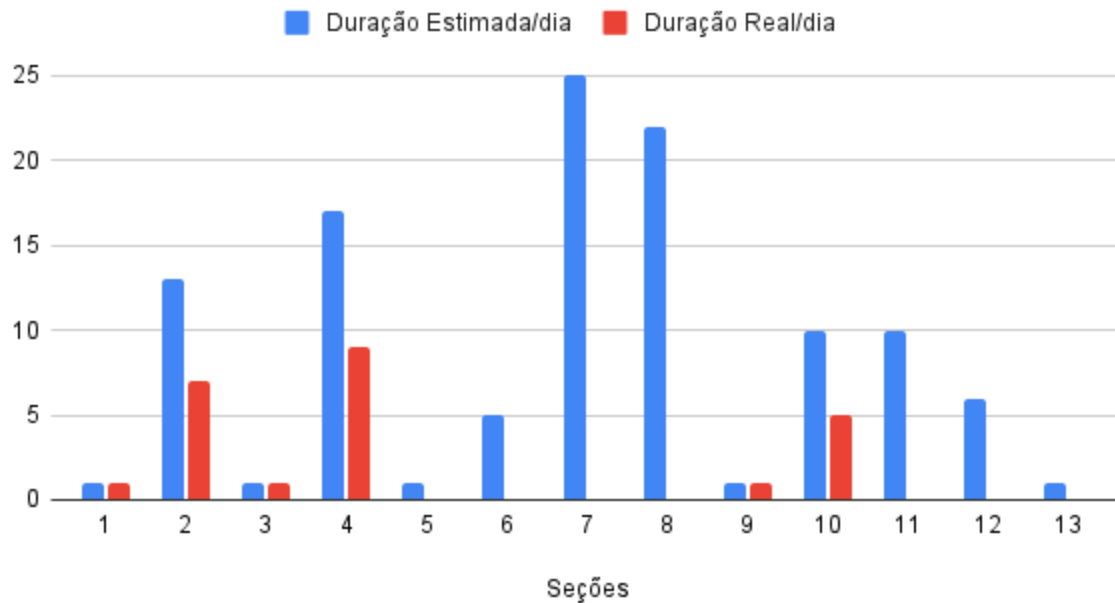
2.2.2. Análise das Estimativas

Até o momento da segunda entrega do projeto, a equipe conseguiu cumprir o cronograma estipulado inicialmente. Para não ocorrer atrasos devido a uma semana que os integrantes tiveram atividades fora do projeto, e com isso não conseguiram cumprir a carga horária diária, estes dobraram o trabalho na semana seguinte para que equipe ficasse em dia e não atrasasse em nenhuma entrega.

O gráfico abaixo mostra o tempo estimado e tempo real para cada seção deste documento. Pode-se perceber que o tempo estimado para as atividades, foram maiores que o real, isso foi devido ao tempo total que tivemos. Como era um período relativamente grande, estimamos a duração das atividades um pouco maiores do que realmente eram.

No gráfico abaixo também pode-se notar que em algumas colunas vermelhas estão faltando, isso é devido às próximas atividades que ainda não foram iniciadas. A planilha que coleciona esses dados está disponível no [link](#).

Duração Estimada/dia e Duração Real/dia



2.3. Gerência do Custo

A equipe fez uma estimativa de que o projeto levaria 560 horas no total para ser finalizado, levando em consideração que a equipe possui 14 integrantes.

Nesta [planilha](#) estão as estimativas de custo utilizando as técnicas sugeridas.

- Custo
 - Pessoal
 - $(R\$40/h \cdot 14 \text{ pessoas}) \cdot 450h = R\$18.000,00$
 - Ferramentas (software)
 - R\$1.600,00 reais
- Total Bruto = $R\$18.000 + R\$1600 = R\$19.600$
- Erro da estimativa: 15% em cima do total bruto (inflação) = 2.940,00
- Lucro 30% em cima do salário = 5.400,00
- Custo total = R\$27.940,00

2.4. Gerência de Riscos

Citem os riscos iniciais e analisem ao longo do projeto. Como cada risco inicial foi gerenciado? Quais os riscos ocorreram? Surgiu algum risco fora do planejado

inicialmente? Se sim, qual? O que vocês fizeram para lidar com isso (estratégia de gerenciamento dos riscos)? Descrevam como monitoram os riscos ao longo do projeto. Indiquem os riscos que realmente ocorreram.

Os riscos do projeto serão avaliados pelo menos uma vez em cada iteração e documentados nesta tabela. Os riscos de maior importância são listados primeiro na tabela. Como o projeto não terá muitas dificuldades de desenvolvimento, como mostra o Estudo de Viabilidade, os riscos detectados foram poucos.

Classificação do Risco	Impacto e Descrição do Risco	Estratégia de Diminuição e/ou Plano de Contingência
Alto	Integrante indisponível para realização de alguma tarefa	Procurar planejar ao máximo de antecedência para que isso não ocorra e fazer com que no mínimo dois integrantes da equipe dominem o desenvolvimento de uma determinada tarefa
Médio	Pouca experiência em Banco de Dados	Treinar durante a disciplina de Gerenciamento de Dados e Informação
Médio	Tempo de desenvolvimento não ideal	Gerenciamento de cronograma e pontualidade
Médio	Dificuldade na manutenção de consistência	Gerenciar Banco de Dados e realizar próxima tarefa quando anterior concluída
Médio	Complexidade de Interface gráfica	Definir padrão e dividir para que cada integrante implemente uma parte da interface gráfica
Médio	Indisponibilidade para reuniões	Utilizar ferramentas de comunicação assíncrona para manter a participação do integrante.
Alto	Atraso na entrega de alguma task	Readequação do prazo e movimentação de integrantes que possam ajudar o integrante atrasado.
Baixo	Falta de energia	Foco em tarefas assíncronas e offline
Médio	Quebra da máquina de algum integrante	Uso dos laboratórios da universidade.
Alto	Mudança ou adição dos requisitos funcionais da aplicação	Distribuição da task em pequenas partes para diminuição do gargalo
Baixo	Volta do modelo EAD por nova onda da pandemia	Manutenção de reuniões online e distribuição de tarefas independentes de encontros presenciais.

2.5. Recursos do Projeto

Abaixo estão listados os recursos de hardware e software a serem utilizados no projeto.

Pessoas

- 5 pessoas - Backend
- 4 pessoas - Frontend
- 4 pessoas - Documentação
- 4 pessoas - Testes

Recursos de Hardware:

- Computadores do DSC – configuração variável dependendo de cada laboratório;
- Computadores pessoais de cada integrante.

Recursos de Software:

- 1 - IntelliJ
- 2 - Eclipse
- 3 - Junit
- 4 - Lombok
- 5 - Postgres
- 6 - Swagger
- 7 - Maven
- 8 - LucidCharts
- 9 - Typescript
- 10 - Angular
- 11 - Yarn
- 12 - Angular
- 13 - Material
- 14 - Figma

Houve custo na aquisição de algum software e hardware? Comentem.

R: Nenhum custo adicional para aquisição de software foi necessário.

2.6. Comunicação

Comunicação assíncrona através do whatsapp separada em grupo geral e subgrupos.
Reuniões semanais entre gerentes e subequipes.
Encontros semanais nas aulas de ES e após as aulas.
Reuniões sob demanda.

<https://comet-gargoyle-cf5.notion.site/Engenharia-de-Software-2c5d59e049d149bfa573c380d15aff5c>

3. Processo

Para começar este projeto a equipe se reuniu duas vezes para decidir qual projeto iria se desenvolver. Após algumas discussões e votações, o projeto a ser desenvolvido foi o HomeServices, objetivo o qual foi descrito na seção 1.
Após a decisão do projeto, os subgrupos foram divididos baseando-se nas preferências dos integrantes e em qual área eles se sentiriam mais confortável em trabalhar.
A decisão por essa divisão foi feita devido ao número elevado de integrantes, então as reuniões e decisões seriam mais fáceis de serem marcadas em um número menor de pessoas. Optou-se também por fazer um grupo de gerentes, os quais seriam os representantes de cada subgrupo, isto para facilitar a comunicação entre toda a equipe e também para tomar decisões que não necessariamente precisam ser feitas por todos.

4. Requisitos

4.1. Requisitos Não-Funcionais

Alguns requisitos não-funcionais (RNFs) foram identificados para o projeto do TIG. Abaixo, são apresentadas as descrições para os três tipos de RNF (de processo, de produto e externos), destacando-se o grupo de requisitos de produto, cujo grafo SIG foi construído e comentado [6].

4.1.1. Requisitos de Processo

Ident.	Descrição
RNF/PROC-01	O sistema deve ser implementado na linguagem JAVA. Logo ele poderá ser executado em diferentes sistemas operacionais, como o Linux, Windows, etc.
RNF/PROC-02	Toda atualização nas informações tratadas pelo sistema deverá ser feita no servidor.
RNF/PROC-03	Deverá ser feita uma documentação que contenha o diagrama de classes, já que a linguagem utilizada será orientada a objetos, e informações sobre o código-fonte do projeto.
RNF/PROC-04	Deverão ser utilizadas ferramentas CASE, e é imprescindível a criação da modelagem usando a linguagem UML.

4.1.2. Requisitos de Produto

4.1.2.1. Segurança

Ident.	Descrição
RNF/SEG-05	Um CPF só pode estar vinculado a uma conta do sistema.
RNF/SEG-06	A atualização do sistema será feita apenas a partir da equipe responsável pela manutenção do sistema.
RNF/SEG-07	O sistema não deve armazenar dados de usuário senão aqueles já fornecidos por estes.
RNF/SEG-08	O CPF cadastrado não deve ser visível para nenhum usuário.
RNF/SEG-09	Caso o usuário opte por deletar sua conta, todos os seus dados devem ser excluídos da plataforma.

4.1.2.2. Performance

Ident.	Descrição
RNF/PER-01	O tempo de resposta para operações que envolvam dados pessoais não deve exceder 10 segundos. Para qualquer

	outra operação do sistema, esse não deve exceder 5 segundos.
RNF/PER-02	A página de login não deve demorar mais do que segundos para ser inteiramente carregada.
RNF/PER-03	A operação de login não deve demorar mais do que 10 segundos para ser realizada.
RNF/PER-04	Do momento do clique no anúncio de um serviço ao momento de exibição deste não devem ser contados mais do que 8 segundos.
RNF/PER-04	O usuário não deve conseguir fazer o upload de mais que 9 10 imagens devido às restrições de desempenho.

4.1.3. Requisitos Externos

4.1.3.1. Requisitos Legais

Ident.	Descrição
RNF/LEG-20	O usuário anunciante precisa assinar um termo de condições que garante a não responsabilidade jurídica pelos atos cometidos por clientes
RNF/LEG-21	Usuários que firmam direitos e/ou leis através de postagens precisam ser banidos da plataforma, sendo encontrados através de denúncias de usuários.
RNF/LEG-22	Não devem ser utilizados sistemas que precisem de uma licença paga.

4.1.3.2. Restrições Econômicas

Ident.	Descrição
RNF/ECO-22	Não devem ser utilizados softwares que precisem de pagamento.
RNF/ECO-22	O custo não deve exceder o valor de 27.940,00

4.2. Requisitos Funcionais

Cód.	Nome	Prioridade
------	------	------------

RF-01	Realizar Cadastro	Obrigatório
RF-02	Realizar Login	Obrigatório
RF-03	Esqueceu a senha	Desejável
RF-04	Alterar informações da conta	Obrigatório
RF-05	Realizar Logout	Obrigatório
RF-06	Remover conta	Obrigatório
RF-07	Ver detalhes serviço	Obrigatório
RF-08	Buscar serviço	Obrigatório
RF-09	Editar serviço	Obrigatório
RF-10	Remover serviço	Obrigatório
RF-11	Oferecer serviço	Obrigatório
RF-12	Avaliar serviço	Obrigatório
RF-13	Listar serviços oferecidos pelo usuário	Obrigatório
RF-14	Filtrar serviços	Obrigatório
RF-15	Navegar pelo menu lateral	Obrigatório
RF-16	Informar sobre o aplicativo	Obrigatório

4.2.1 Especificação dos Requisitos

RF-01	
Nome:	Realizar Cadastro
Descrição:	O usuário deve poder se cadastrar no sistema informando seu nome completo, data de nascimento, email, cpf, celular, cidade, bairro e senha.
Atores:	Usuário comum
Prioridade:	Essencial https://drive.google.com/file/d/1lbefujwwk3VH-7aJqViR30oJCkayT5Q_/view?usp=sharing
Requisitos Não Funcionais Associados:	Nenhum.
Entradas e pré-condições:	O usuário deve estar na página de cadastro.
Saídas e pós-condições:	O usuário criou uma nova conta.
Fluxos de eventos	
Fluxo principal:	1. O usuário informa os dados necessários para o cadastro:

	<ul style="list-style-type: none"> • Nome completo; • Data de nascimento; • CPF; • Celular; • E-mail; • Senha de acesso; • Cidade; • Bairro; <p>2. O sistema verifica a validade dos dados;</p> <p>3. O sistema informa que a conta foi criada com sucesso.</p>
Fluxo secundário 1:	No fluxo principal 1, caso o usuário preencha com dados inválidos, o sistema exibe um aviso com a mensagem "Dados Inválidos" e não realiza o cadastro.
Fluxo secundário 2:	Caso o usuário aperte no botão "Cancelar", o sistema retorna para a página inicial e não realiza o cadastro.

RF-02	
Nome:	Realizar <i>login</i>
Descrição:	O usuário deve poder realizar <i>login</i> no sistema.
Atores:	Usuário comum
Prioridade:	Essencial
Requisitos Não Funcionais Associados:	Nenhum.
Entradas e pré-condições:	O usuário deve possuir um cadastro no sistema, não ter feito o <i>login</i> e estar visualizando a página do "Login".
Saídas e pós-condições:	O usuário faz o <i>login</i> .
Fluxos de eventos	
Fluxo principal:	<p>1. O sistema requer do usuário:</p> <ul style="list-style-type: none"> • E-mail; • Senha; <p>2. O sistema valida o <i>login</i> do usuário.</p>

	3. O sistema exibe uma mensagem de sucesso e redireciona o usuário para a página dos serviços disponíveis.
Fluxo secundário 1:	Caso o usuário preencha com dados incorretos, o sistema exibe um aviso com a mensagem “Dados incorretos.” e não realiza o <i>login</i> .

RF-03	
Nome:	Esqueceu a senha
Descrição:	O usuário deve poder alterar senha da conta caso a tenha esquecido.
Atores:	Usuário comum
Prioridade:	Desejável
Requisitos Não Funcionais Associados:	Nenhum.
Entradas e pré-condições:	O usuário deve possuir um cadastro no sistema e estar visualizando a página do “Login”.
Saídas e pós-condições:	O usuário alterou a senha de sua conta.
Fluxos de eventos	
Interessante Fluxo principal:	<ol style="list-style-type: none"> 1. O usuário seleciona a opção “Esqueci minha senha” na página de “Login”. 2. O sistema solicita e-mail da conta para enviar pedido de redefinição de senha. 3. O sistema envia uma nova senha para o e-mail do usuário.
Fluxo secundário 1:	Caso o e-mail fornecido não seja de uma conta cadastrada, o sistema exibe um aviso com a mensagem “E-mail não cadastrado” e pede por outro e-mail.

RF-04	
Nome:	Alterar Informações da Conta
Descrição:	O usuário deve poder alterar os dados de sua conta, como nome, email, celular, cidade e bairro.
Atores:	Usuário comum

Prioridade:	Essencial
Requisitos Não Funcionais Associados:	Nenhum.
Entradas e pré-condições:	O usuário deve ter feito <i>login</i> no sistema.
Saídas e pós-condições:	O usuário alterou os dados de sua conta.
Fluxos de eventos	
Fluxo principal:	<ol style="list-style-type: none"> 1. O usuário aperta na sua foto de perfil para acessar a página de edição de conta. 2. O sistema fornece um formulário com as informações atuais do usuário. 3. O usuário altera os campos do formulário. 4. O usuário seleciona salvar. 5. Os dados são verificados e a conta é alterada.
Fluxo secundário 1:	Caso o usuário preencha com dados inválidos, o sistema exibe um aviso com a mensagem "Dados Inválidos" e não realiza a alteração.
Fluxo secundário 2:	Caso o usuário aperte no botão "Cancelar" o sistema retorna para a página inicial e não realiza alterações.

RF-05	
Nome:	Realizar <i>logout</i>
Descrição:	O usuário deve poder fazer <i>logout</i> do sistema.
Atores:	Usuário comum
Prioridade:	Essencial
Requisitos Não Funcionais Associados:	Nenhum.
Entradas e pré-condições:	O usuário deve ter feito o <i>login</i> .
Saídas e pós-condições:	O usuário faz o <i>logout</i> .
Fluxos de eventos	
Fluxo principal:	<ol style="list-style-type: none"> 1. O usuário seleciona o botão "sair". 2. O sistema exibe pergunta ao usuário se o mesmo tem certeza da ação. 3. O sistema realiza o <i>logout</i> do usuário.

RF-06	
Nome:	Remover Conta
Descrição:	O usuário deve poder apagar sua conta.
Atores:	Usuário comum
Prioridade:	Essencial
Requisitos Não Funcionais Associados:	Nenhum.
Entradas e pré-condições:	O usuário deve ter feito o <i>login</i> .
Saídas e pós-condições:	O usuário removeu sua conta.
Fluxos de eventos	
Fluxo principal:	<ol style="list-style-type: none"> 1. O usuário aperta na sua foto de perfil para acessar a página de edição de conta. 2. O usuário aperta o botão “Apagar Conta”. 3. O sistema solicita que o usuário digite “apagar a conta” para confirmar. 4. A conta é removida do sistema.
Fluxo secundário 1:	Caso o usuário digite o texto solicitado incorretamente, o sistema exibe um aviso a conta não é removida.
Fluxo secundário 2:	Caso o usuário aperte o botão “Cancelar”, a caixa de remover conta fecha e a conta não é removida.

RF-07	
Nome:	Ver detalhes serviço
Descrição:	O usuário deve poder visualizar as informações de um serviço.
Atores:	Usuário comum
Prioridade:	Essencial
Requisitos Não Funcionais Associados:	Nenhum.
Entradas e pré-condições:	Estar na página “Contrate um Serviço”.
Saídas e pós-condições:	O usuário visualiza os detalhes do serviço.

Fluxos de eventos	
Fluxo principal:	<ol style="list-style-type: none"> 1. O usuário seleciona na tela o serviço que deseja ver os detalhes. 2. O sistema exibe ao usuário as informações do serviço, disponibilizadas por quem o oferece.

RF-08	
Nome:	Buscar serviço
Descrição:	O usuário deve poder buscar serviços pelo seu título.
Atores:	Usuário comum
Prioridade:	Essencial
Requisitos Não Funcionais Associados:	Nenhum.
Entradas e pré-condições:	Usuário ter efetuado o <i>login</i> no sistema e estar visualizando a página "Contrate um serviço".
Saídas e pós-condições:	O usuário visualiza todos os serviços correspondentes à sua busca.
Fluxos de eventos	
Fluxo principal:	<ol style="list-style-type: none"> 1. O usuário insere na barra de busca o que ele deseja encontrar de correspondências nos títulos dos serviços do sistema. 2. O usuário tecla <i>Enter</i> ou seleciona o ícone da lupa de busca. 3. O sistema retorna todos os serviços que possuem títulos correspondentes com a busca informada.
Fluxo secundário 1:	Caso não sejam encontradas correspondências para a busca, o sistema exibe um aviso com a mensagem "Não foi possível encontrar serviço(s) com esse título"

RF-09	
Nome:	Editar serviço

Descrição:	O usuário deve poder editar seus serviços.
Atores:	Usuário comum
Prioridade:	Essencial
Requisitos Não Funcionais Associados:	Nenhum.
Entradas e pré-condições:	Usuário ter efetuado o <i>login</i> no sistema e ter ao menos um serviço criado.
Saídas e pós-condições:	O usuário tem o serviço anterior editado.
Fluxos de eventos	
Fluxo principal:	<ol style="list-style-type: none"> 1. O usuário visualiza seus serviços no botão “Meus Serviços”. 2. O usuário seleciona o ícone de editar do serviço que ele deseja editar. 3. O sistema exibe ao usuário as informações do serviço o qual será editado. 4. O usuário preenche os dados e aperta no botão “Salvar”. 5. O serviço é editado.
Fluxo secundário 1:	Caso o usuário selecione o botão “Cancelar”, o sistema redireciona o usuário para a página “Contrate Um Serviço” e a operação é cancelada.

RF-10	
Nome:	Remover serviço
Descrição:	O usuário deve poder remover os serviços que criou.
Atores:	Usuário comum
Prioridade:	Essencial
Requisitos Não Funcionais Associados:	Nenhum.
Entradas e pré-condições:	Usuário ter efetuado o <i>login</i> no sistema e ter um serviço criado.
Saídas e pós-condições:	O usuário tem o serviço anterior deletado.
Fluxos de eventos	

Fluxo principal:	<ol style="list-style-type: none"> 1. O usuário visualiza seus serviços no botão “Meus Serviços”. 2. O usuário seleciona o serviço ao qual será deletado. 3. O usuário seleciona o botão “apagar” e confirma a remoção do serviço. 4. O serviço é excluído da lista de serviços.
Fluxo secundário 1:	Caso o usuário selecione o botão “Cancelar”, o sistema redireciona o usuário para a página “Contrate Um Serviço” e a operação é cancelada.
Fluxo secundário 2:	No fluxo principal 1, o usuário pode selecionar o botão de excluir no serviço e confirmar a remoção deste.

RF-11	
Nome:	Oferecer serviço
Descrição:	O usuário deve poder criar um serviço.
Atores:	Usuário comum
Prioridade:	Essencial
Requisitos Não Funcionais Associados:	Nenhum.
Entradas e pré-condições:	Usuário ter efetuado o <i>login</i> no sistema.
Saídas e pós-condições:	O usuário terá criado um novo serviço.
Fluxos de eventos	
Fluxo principal:	<ol style="list-style-type: none"> 1. O usuário seleciona a página “Ofereça um serviço” 2. O usuário insere os dados do serviço. 3. O usuário seleciona “salvar”. 4. O usuário seleciona “confirmar”.
Fluxo secundário 1:	Caso o usuário selecione o botão “Cancelar”, o sistema redireciona o usuário para a página “Contrate Um Serviço” e a operação é cancelada.

RF-12	
Nome:	Avaliar Serviço
Descrição:	O usuário deve ser capaz de avaliar um serviço com 1 até 5 estrelas e um comentário.
Atores:	Usuário comum
Prioridade:	Essencial
Requisitos Não Funcionais Associados:	Nenhum.
Entradas e pré-condições:	Usuário ter efetuado o <i>login</i> no sistema e estar visualizando a página do detalhamento de um serviço.
Saídas e pós-condições:	O usuário visualiza sua avaliação no histórico de avaliações do serviço.
Fluxos de eventos	
Fluxo principal:	<ol style="list-style-type: none"> 1. O usuário escolhe uma quantidade de estrelas que deseja avaliar o serviço. 2. O sistema exibe um modal de avaliação com a quantidade de estrelas selecionadas no passo 1 e um campo para digitar um comentário. 3. O usuário digita um comentário a respeito do serviço. 4. O usuário seleciona a opção OK.
Fluxo secundário 1:	No fluxo principal 3, se o usuário não quiser fazer um comentário deve deixar em branco o campo de comentário e continuar com o passo 4.
Fluxo secundário 2:	No fluxo principal 2, se o usuário desejar mudar a quantidade de estrelas, ele pode mudar a sua escolha no modal de avaliação.

RF-13	
Nome:	Listar serviços oferecidos pelo usuário

Descrição:	O sistema deve permitir que o usuário possa listar os serviços que ele oferece.
Atores:	Usuário comum
Prioridade:	Essencial
Requisitos Não Funcionais Associados:	Nenhum.
Entradas e pré-condições:	Usuário ter efetuado o <i>login</i> no sistema.
Saídas e pós-condições:	O usuário visualiza todos os serviços que ele oferece.
Fluxos de eventos	
Fluxo principal:	<ol style="list-style-type: none"> 1. O usuário acessa a opção de Meus Serviços; 2. O usuário visualiza todos os serviços que ele oferece;

RF-14	
Nome:	Filtrar Serviços
Descrição:	O sistema deve permitir que o usuário possa filtrar serviços por características, sendo elas: Tags, Avaliação e cidade de cobertura.
Atores:	Usuário comum
Prioridade:	Essencial
Requisitos Não Funcionais Associados:	Nenhum.
Entradas e pré-condições:	Usuário ter efetuado o <i>login</i> no sistema e estar visualizando a página "Contrate um serviço".
Saídas e pós-condições:	O usuário visualiza todos os serviços que correspondem aos filtros selecionados pelo mesmo.
Fluxos de eventos	
Fluxo principal:	<ol style="list-style-type: none"> 1. O usuário seleciona a opção de adicionar filtros; 2. O usuário preenche os campos de sua preferência; 3. O usuário seleciona a opção de salvar os filtros;

	<ol style="list-style-type: none"> 4. O sistema verifica os campos que foram preenchidos; 5. O sistema busca por todos os serviços que correspondam aos filtros preenchidos; 6. O sistema exibe os serviços resultantes da busca.
--	--

RF-15	
Nome:	Navegar pelo menu lateral
Descrição:	O sistema deve permitir que o usuário possa navegar pelo menu lateral.
Atores:	Usuário comum
Prioridade:	Essencial
Requisitos Não Funcionais Associados:	Nenhum.
Entradas e pré-condições:	Usuário ter efetuado o <i>login</i> no sistema.
Saídas e pós-condições:	O usuário visualiza todas as opções do menu lateral.
Fluxos de eventos	
Fluxo principal:	<ol style="list-style-type: none"> 1. O usuário visualiza as páginas listadas no menu lateral 2. O usuário seleciona uma das páginas: <ul style="list-style-type: none"> • “Contrate Um Serviço”; • “Ofereça Um Serviço”; • “Meus Serviços”; • “Sobre”; 3. O sistema redireciona o usuário para a devida página.

RF-16	
Nome:	Informar sobre o aplicativo

Descrição:	O sistema deve permitir que o usuário visualize as informações de versão, desenvolvedores e uma breve descrição sobre o aplicativo.
Atores:	Usuário comum
Prioridade:	Essencial
Requisitos Não Funcionais Associados:	Nenhum.
Entradas e pré-condições:	O usuário deve estar acessando o sistema.
Saídas e pós-condições:	O usuário visualiza as informações do sistema.
Fluxos de eventos	
Fluxo principal:	<ol style="list-style-type: none"> 1. O usuário acessa o menu lateral. 2. O usuário seleciona o botão “Sobre”. 3. O sistema exibe as informações sobre o aplicativo.

4.3. Prototipação de Telas

Figma:

<https://www.figma.com/file/zvruMkrueM7sbBwx4zYXPM/Home-Services---Prototipagem?node-id=0%3A1&t=OIGboVppNM7Qvhcp-0>

PDF:

https://drive.google.com/file/d/1lbefujwwk3VH-7aJqViR30oJCkayT5Q_/view?usp=sharing

5. Projeto Arquitetural

1. Descrevam a arquitetura do sistema desenvolvido
 - a. Métricas mínimas: 1 ou 2 parágrafos
2. **Ferramenta**: Relatório no Word

6. Especificação Formal

Artefatos:

1. Façam uma especificação formal da estrutura do sistema usando Alloy.
Especificar o modelo de domínio usando Alloy.
 - a. Métrica Média: 20-30 Linhas

2. Especificar alguns fatos e realizar análises. Descrever as análises feitas no relatório.
3. **Ferramenta:**
 - a. Alloy Analyzer

7. Implementação

O link para o repositório é: <http://...>

O link para o screencast é: <http://...>

Artefatos:

1. Implementem o sistema usando Android, Unity, ou algum outro tipo de ambiente para desenvolvimento web/mobile/...
 - a. Métrica mínima: 1 RNF de produto por equipe (ex.: Segurança, Usabilidade)
 - i. Não é necessário implementar o requisito em todo o sistema
 - b. Métrica mínima: 1 RF por pessoa
2. Anexar imagens das telas do sistema final e suas respectivas descrições
3. Apresentar o sistema em alguma máquina que permita a avaliação, notebook, computador de algum laboratório, ou em algum servidor que permita o acesso via URL (sendo essa última a mais interessante) para o sistema
4. **Sugestões de Ferramentas**
 - a. Android: Android Studio
 - b. Unity
 - c. Escolher o ambiente de desenvolvimento de acordo com a tecnologia usada

8. Testes

Artefatos:

1. Testar o sistema novo
 - a. Criar Plano de Testes prévio (sugestão: usar o TestLink, que tem um padrão do plano de testes). Existem outros exemplos de planos teste na [internet](#).
 - b. Rodar a ferramenta Checkstyle para identificar problemas de estilo

- c. Rodar algumas ferramentas de Análise Estática (PMD, SpotBugs, dentre outras)
- d. Executar os testes e avaliar a cobertura do código
- e. Recomendação:
 - i. Descrever bem o que o teste cobre, ou seja, o que o teste implementado avalia

2. Ferramentas

- a. TestLink (para o plano de testes)
- b. Checkstyle
- c. SpotBugs
- d. PMD
- e. Cobertura dos testes

Sistema Real

9. Introdução

JUnit é um framework simples para escrever testes repetíveis. É uma instância da arquitetura xUnit para estruturas de teste de unidade.

As principais linguagens utilizadas na construção da ferramenta são Java e HTML.

10. Qualidade

- 1. Analisar a qualidade de um sistema real de acordo com:
 - a. Métricas dos métodos:

Method	CogC	ev(G)	lv(G)	v(G)
junit.runner.BaseTest	10	4	8	10
junit.samples.money	5	4	5	5
junit.samples.money	3	4	2	4
junit.samples.money	11	7	5	7
org.junit.experimental	16	8	6	8
org.junit.experimental	5	4	2	4
org.junit.experimental	6	4	3	6
org.junit.experimental	4	5	4	5
org.junit.experimental	7	5	4	5
org.junit.experimental	14	4	7	8
org.junit.internal.Core	25	5	9	16
org.junit.internal.Thre	14	4	7	8

i.

org.junit.internal.ma	6	4	4	4
org.junit.internal.run	5	4	1	5
org.junit.internal.run	5	5	5	5
org.junit.internal.run	9	4	5	6
org.junit.internal.run	11	5	7	7
org.junit.internal.run	5	5	2	5
org.junit.internal.run	7	4	5	6
org.junit.rules.Tempe	13	8	7	9
org.junit.rules.Tempe	8	5	3	5
org.junit.runner.JUnit	21	7	8	8
org.junit.runner.man	8	4	3	4
org.junit.runner.man	4	5	2	5
org.junit.runners.Par	7	5	6	6
org.junit.runners.moc	6	4	4	4

ii.

b. Metricas das classes:

Class	OCavg	OCmax	WMC
junit.framework.Asse	1,44	4	56
junit.framework.Test	1,06	3	54
junit.framework.Test	1,58	5	41
junit.runner.BaseTes	1,76	6	44
junit.samples.money	2,12	7	36
org.junit.Assert	1,44	4	102
org.junit.experiment	2,00	8	40
org.junit.experiment	2,20	4	33
org.junit.rules.Event	1,11	3	31
org.junit.rules.Tempe	2,11	8	40
org.junit.rules.Tempe	1,03	2	35
org.junit.runner.Desc	1,39	3	39
org.junit.runner.notif	3,50	4	7
org.junit.runner.notif	1,29	3	31
org.junit.runners.Bloc	1,47	2	50
org.junit.runners.Par	1,63	1	62
org.junit.runners.moc	1,57	5	33
org.junit.runners.moc	1,82	4	51
org.junit.runners.pars	2,13	3	32
org.junit.tests.assert	1,00	1	104
org.junit.tests.running	1,00	1	31

i.

- c. Analisar métricas já definidas no plugin Metrics ou qualquer outro plugin (pelo menos 15 métricas por equipe)

#	File	Line	Problem
1	home\luanna\junit4\src\main\java\junit\extensions\ActiveTestSuite.java	15	Avoid excessively long variable names like <code>FActiveTestDeathCount</code>
2	home\luanna\junit4\src\main\java\junit\extensions\ActiveTestSuite.java	15	Field comments are required
3	home\luanna\junit4\src\main\java\junit\extensions\ActiveTestSuite.java	15	Found non-transient, non-static member. Please mark as transient or provide accessors.
4	home\luanna\junit4\src\main\java\junit\extensions\ActiveTestSuite.java	15	Found non-transient, non-static member. Please mark as transient or provide accessors.
5	home\luanna\junit4\src\main\java\junit\extensions\ActiveTestSuite.java	15	Use of modifier volatile is not recommended.
6	home\luanna\junit4\src\main\java\junit\extensions\ActiveTestSuite.java	15	Use of modifier volatile is not recommended.
7	home\luanna\junit4\src\main\java\junit\extensions\ActiveTestSuite.java	17	Document empty constructor
8	home\luanna\junit4\src\main\java\junit\extensions\ActiveTestSuite.java	17	It is a good practice to call <code>super()</code> in a constructor
9	home\luanna\junit4\src\main\java\junit\extensions\ActiveTestSuite.java	17	Public method and constructor comments are required
10	home\luanna\junit4\src\main\java\junit\extensions\ActiveTestSuite.java	20	Parameter 'theClass' is not assigned and could be declared final
11	home\luanna\junit4\src\main\java\junit\extensions\ActiveTestSuite.java	20	Public method and constructor comments are required
12	home\luanna\junit4\src\main\java\junit\extensions\ActiveTestSuite.java	24	Parameter 'name' is not assigned and could be declared final
13	home\luanna\junit4\src\main\java\junit\extensions\ActiveTestSuite.java	24	Public method and constructor comments are required
14	home\luanna\junit4\src\main\java\junit\extensions\ActiveTestSuite.java	28	Parameter 'name' is not assigned and could be declared final
15	home\luanna\junit4\src\main\java\junit\extensions\ActiveTestSuite.java	28	Parameter 'theClass' is not assigned and could be declared final
16	home\luanna\junit4\src\main\java\junit\extensions\ActiveTestSuite.java	28	Public method and constructor comments are required
17	home\luanna\junit4\src\main\java\junit\extensions\ActiveTestSuite.java	33	Parameter 'result' is not assigned and could be declared final
18	home\luanna\junit4\src\main\java\junit\extensions\ActiveTestSuite.java	41	Avoid variables with short names like <code>t</code>
19	home\luanna\junit4\src\main\java\junit\extensions\ActiveTestSuite.java	41	Local variable <code>y</code> could be declared final
20	home\luanna\junit4\src\main\java\junit\extensions\ActiveTestSuite.java	41	To be compliant to J2EE, a webapp should not use any thread.
21	home\luanna\junit4\src\main\java\junit\extensions\ActiveTestSuite.java	41	To be compliant to J2EE, a webapp should not use any thread.
22	home\luanna\junit4\src\main\java\junit\extensions\ActiveTestSuite.java	41	To be compliant to J2EE, a webapp should not use any thread.
23	home\luanna\junit4\src\main\java\junit\extensions\ActiveTestSuite.java	41	To be compliant to J2EE, a webapp should not use any thread.
24	home\luanna\junit4\src\main\java\junit\extensions\ActiveTestSuite.java	49	Potential violation of Law of Demeter (method chain calls)
25	home\luanna\junit4\src\main\java\junit\extensions\ActiveTestSuite.java	49	Potential violation of Law of Demeter (method chain calls)

- d. Identificar bad smells no código usando o SpotBugs, PMD, ou qualquer outra ferramenta (pelo menos 5 bad smells por equipe)

[Class names shouldn't shadow simple name of superclass](#) High

The class name `org.junit.AssumptionViolatedException` shadows the simple name of the superclass `org.junit.internal.AssumptionViolatedException`

In file `AssumptionViolatedException.java`, lines 22 to 45
In class `org.junit.AssumptionViolatedException`
In class `org.junit.internal.AssumptionViolatedException`
At `AssumptionViolatedException.java`: [lines 22-45]

i.

[Method invokes System.exit\(...\)](#) Medium

`junit.textui.TestRunner.runFailed(String)` invokes `System.exit(...)`, which shuts down the entire virtual machine

In file `TestRunner.java` `TestRunner.java`, line 198 198
In class `junit.textui.TestRunner`
In method `junit.textui.TestRunner.runFailed(String)`
At `TestRunner.java`: [line 198]
At `TestRunner.java`: [line 198]

ii.

[Class names shouldn't shadow simple name of superclass](#) High

The class name `org.junit.internal.runners.model.MultipleFailureException` shadows the simple name of the superclass `org.junit.runners.model.MultipleFailureException`

In file `MultipleFailureException.java`, lines 10 to 11
In class `org.junit.internal.runners.model.MultipleFailureException`
In class `org.junit.runners.model.MultipleFailureException`
At `MultipleFailureException.java`: [lines 10-11]

iii.

iv.	Class is not derived from an Exception, even though it is named as such	Medium	<p>Class org.junit.internal.runners.statements.ExpectException is not derived from an Exception, even though it is named as such</p> <p>In file ExpectException.java, lines 10 to 37 In class org.junit.internal.runners.statements.ExpectException At ExpectException.java:[lines 10-37]</p>
v.	Class is not derived from an Exception, even though it is named as such	Medium	<p>Class org.junit.rules.ExpectedException is not derived from an Exception, even though it is named as such</p> <p>In file ExpectedException.java, lines 111 to 283 In class org.junit.rules.ExpectedException At ExpectedException.java:[lines 111-283]</p>

e. Rodar a ferramenta Checkstyle para identificar problemas de estilo

2505	/home/luanna/junit4/src/main/java/org/junit/internal/Checks.java	15	Parameter 'value' is not assigned and could be declared final
2506	/home/luanna/junit4/src/main/java/org/junit/internal/Checks.java	17	Avoid throwing null pointer exceptions.
2507	/home/luanna/junit4/src/main/java/org/junit/internal/Checks.java	22	Comment is too large. Too many lines
2508	/home/luanna/junit4/src/main/java/org/junit/internal/Checks.java	31	Parameter 'message' is not assigned and could be declared final
2509	/home/luanna/junit4/src/main/java/org/junit/internal/Checks.java	31	Parameter 'value' is not assigned and could be declared final
2510	/home/luanna/junit4/src/main/java/org/junit/internal/Checks.java	33	Avoid throwing null pointer exceptions.

f. Rodar algumas ferramentas de Análise Estática (PMD, SpotBugs, dentre outras)

Metrics

7971 lines of code analyzed, in 350 classes, in 32 packages.

Metric	Total	Density*
High Priority Warnings	7	0.88
Medium Priority Warnings	54	6.77
Total Warnings	61	7.65

(* Defects per Thousand lines of non-commenting source statements)

i.

2. Análise

Conforme foi pedido, foi rodado no plugin SpotBugs, e algumas métricas indicando má qualidade foram encontradas, como por exemplo, uma longa lista de parâmetros são definidos, os tornando difíceis de ler e complicando a chamada e o teste da função. Pode indicar que o propósito da função é mal concebido e que o código deve ser refatorado para que a responsabilidade seja atribuída de maneira mais clara. Esse defeito na qualidade pode ser encontrado em diversas linhas do código.

A invocação potencialmente ambígua de um método herdado ou externo, onde uma classe interna está invocando um método que pode ser resolvido para um método herdado ou um método definido em uma classe externa. Exemplos:

In file Theories.java Theories.java, line 263 263
In class org.junit.experimental.theories.Theories\$TheoryAnchor\$1
In method org.junit.experimental.theories.Theories\$TheoryAnchor\$1.createTest()

O problema não pode ser identificado visualmente, mas como ele não pode dizer a diferença entre uma invocação de um método herdado ou externo, o plugin ainda reclamará sobre uma potencial invocação ambígua.

No total foram reportados 19507 bad smells, juntamente com o plugin utilizado, porém apenas 7 foram warnings de alta prioridade, e 54 foram warnings de prioridade média. Tornando a complexidade dos warnings não tão altas comparando com a quantidade de linhas e classes que foram analisadas. (7971 linhas de código analisadas, em 350 classes)

11. Testes

Artefatos:

1. Fazer uma análise da qualidade do software real através dos testes
 - a. Gerar testes (para uma parte do código se o total não funcionar) usando alguma ferramenta (EvoSuite, Randoop), executem e tentem encontrar defeitos. Se encontrarem algum bug, confirmem e descrevam o problema.
 - i. Se as ferramentas não funcionarem para o código completo, escolham uma parte do sistema para gerar testes
 - ii. Façam uma análise da cobertura dos testes
 - iii. Relacionar as métricas coletadas durante a fase de Gerência de Qualidade do Software Real com os resultados dos testes obtidos.
 - I. Percebeu-se alguma relação entre elas?
 - II. Alguma métrica que indicava problema resultou em algum bug?
 - iv. Adicionar imagens dos resultados gerados pelas ferramentas
 - v. Indicar quantos testes foram gerados, qual a cobertura, o processo de realização dos testes.
 - b. [opcional] Realizar teste de mutação usando PIT ou alguma ferramenta similar.
 - c. [opcional] Realizar teste de stress usando Monkey ou Locust ou alguma ferramenta similar

2. Ferramentas:

1. Escolher uma ferramenta (EvoSuite, Randoop ou outra) para gerar testes automaticamente
2. JUnit para executar os testes
3. Usar alguma ferramenta para analisar a cobertura de testes
4. [opcional] Monkey (frontend Android) ou Locust (backend)
5. [opcional] Pit para realizar testes de mutação

6. [opcional] Para alguns bugs detectados, enviar o pull request com a correção.

12. Evolução

Artefatos:

1. Refatorar o sistema real para eliminar algum problema encontrado
 - a. Métricas mínimas:
 - i. Diferença entre o sistema refatorado e o inicial deve ser em torno de 100 LOC E Número de refactorings diferentes ≥ 5
 - b. Explicar a motivação do refactoring
 - i. Indicar os refactorings tradicionais apresentados no livro do Fowler que foram utilizados
 - c. Usar sintomas que o SpotBugs ou a ferramenta que coleta métricas (motivação para o refatoramento).
 - d. Executem a ferramenta que coleta métricas depois do refatoramento e avaliem se alguma métrica melhorou (qualidade).
 - e. Comparem as métricas antes e depois. As métricas melhoraram? Façam uma análise crítica.
 - f. Usar uma coleção de testes para avaliação da preservação do comportamento
2. Escrever um relatório reportando o que foi mudado, que refatoramentos foram aplicados. Qual foi o tamanho da mudança?
 - a. Para cada modificação realizada, descrever brevemente o problema em termos de código, o que foi feito para resolvê-lo e expor um diff (sugestão: usar o diff de versões do GIT)
 - b. [opcional] Envie o pull request para o projeto real com a mudança realizada.
3. **Ferramenta:**
 - a. Refactorings implementados pela sua IDE favorita

13. Conclusões

Dividir a conclusão por fases e expor as dificuldades encontradas e descrever as lições aprendidas.

- Como foi a interação com o cliente? Ele(a) ficou satisfeito(a)?
- Quais atividades/ferramentas utilizariam na execução de um novo projeto? O que deu e não deu certo?
- O que vocês pretendem mudar nos próximos projetos?