



Universidade de Brasília – UnB
Departamento de Ciência da Computação – CiC
Engenharia de Software
Profº Gustavo van Erven

Alunos:

Carlos Andre Velozo Trufini – 13/0105155
Felipe Garcia da Costa – 13/0109631
Geraldino Antônio da Silva – 13/0112267
Luciano Holanda Porto – 12/0126800
Maysa Meirelles Casella – 13/0127116
Miguel de Sousa Santos – 140077464

PROJETO DE ENGENHARIA DE SOFTWARE:

UNBSNACKS

1) Projeto:

O UnBSnacks foi desenvolvido com o objetivo de melhorar a disseminação sobre alimentação dentro do Campus Darcy Ribeiro da Universidade de Brasília. Para tal, nossos clientes solicitaram a construção de um site onde o público da universidade fosse capaz de consultar os cardápios das lanchonetes além de poderem avaliar os alimentos disponíveis em cada uma delas.

Para que este projeto fosse viável, definimos papéis entre os membros do grupo que ficaram assim divididos:

- **Programadores:** Carlos, Felipe, Geraldino e Miguel;
- **Administrador de Banco de Dados:** Luciano;
- **Testadores:** Luciano e Maysa;
- **Documentação e levantamento de requisitos:** Maysa
- **Gerenciamento do grupo:** Maysa;
- **Cliente:** membros do grupo parceiro, sendo os alunos Vitor e Vinícius os pontos focais;

Para auxiliar na viabilização do projeto, também foram utilizadas algumas práticas de negócio da XP, entre elas: cliente no time e metáforas, ou seja, mantivemos constante contato com nossos clientes, além de utilizarmos metáforas para auxiliar na explicação do sistema, como por exemplo, as suas similaridades com o aplicativo “ifood”. Entre as práticas de desenvolvimento da XP, fizemos o uso das técnicas de projeto simples e programação em pares, onde foi possível atingirmos grande parte dos objetivos propostos no início do projeto.

2) Stakeholders:

Em reuniões de levantamento dos requisitos do sistema foi possível identificar os seguintes stakeholders:

- **Público da Universidade de Brasília:**

Composta por alunos, professores, demais funcionários e comunidade em geral que frequenta o Campus, são as pessoas que se deparam diariamente com a dificuldade de saber a qualidade das lanchonetes da Universidade. Pela falta de divulgação de informações sobre as lanchonetes, os usuários acabam comprando nos locais utilizando o critério de proximidade/distância ou indicação de alguém próximo, muitas vezes tendo más experiências com o estabelecimento ou perdendo a oportunidade de comprar um lanche melhor por não ter um feedback de outro cliente.

Sendo assim, foi detectada a necessidade de um local que agregasse informações sobre os estabelecimentos, com produtos, valores e ainda uma avaliação dos usuários sobre a experiência com aquela lanchonete.

- **Lanchonetes:**

Composta por donos e funcionários das lanchonetes do Campus, são as pessoas que atendem e gerem o comércio na Universidade. Como principal problema desta classe, foi identificada a dificuldade em atrair maior público para o estabelecimento através de um serviço diferenciado, além da falta de acesso a feedbacks dos clientes.

Sendo assim, foi detectada a necessidade de um local onde fosse possível divulgar o cardápio do estabelecimento, mantendo o público informado sobre promoções e novos produtos, além de ter acesso rápido a opinião de seus clientes, podendo entrar em contato com o usuário e agradecer ou explicar algum mal

entendido que possa ter ocorrido na experiência de compra e venda.

3) Arquitetura:

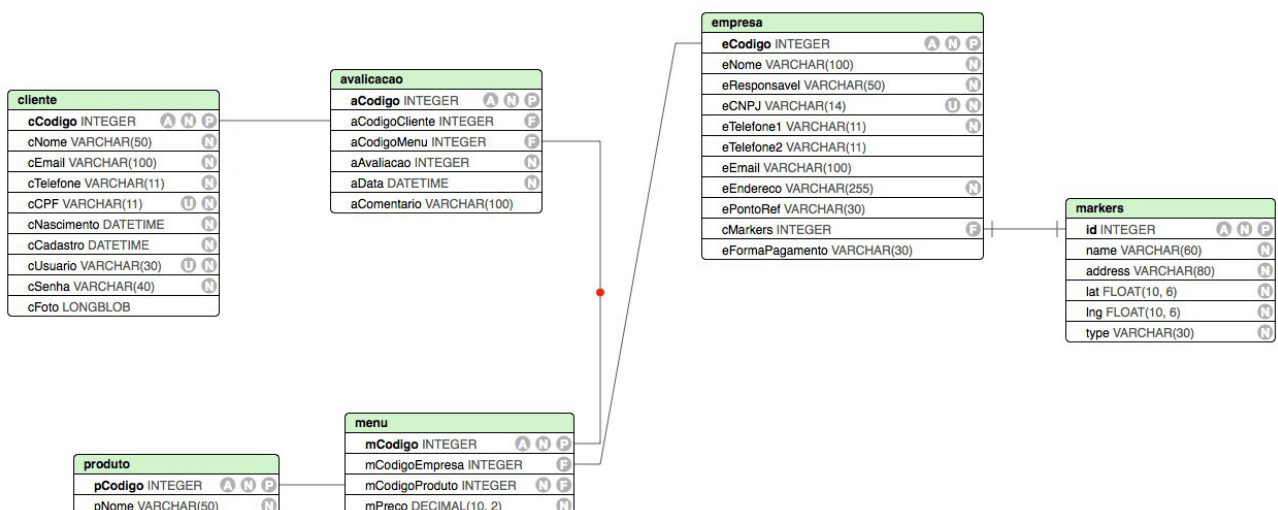
O projeto foi construído utilizando a Linguagem de Programação *Php* e framework *Laravel*, além da plataforma de infraestrutura como serviço *Heroku* e um banco de dados *PostgreSQL*.

A decisão pela utilização da Linguagem de Programação e do Banco de Dados se deu pela familiaridade e facilidade dos membros do grupo em programar com tal linguagem e infraestrutura, reduzindo substancialmente o tempo utilizado no desenvolvimento do sistema.

Quanto ao framework e a plataforma *Heroku*, a escolha se deu pela gratuidade e qualidade dos mesmos, além da facilidade para a detecção de erros encontrados, eficiência na resolução dos problemas no desenvolvimento e otimização dos recursos disponíveis.

Para auxiliar no planejamento e desenvolvimento do sistema também foi utilizada o sistema *Trello* e o repositório *GitHub*. No primeiro foram feitos cartões com cada uma das atividades que deveriam ser desenvolvidas, além de informações gerais que todos os membros do grupo deveriam ter ciência. Já o segundo foi utilizado como repositório para código e documentos.

A seguir, modelo de dados do sistema:



4) Histórias de usuário:

Para o desenvolvimento do UnBSnacks, utilizamos como base as histórias de usuário abaixo listadas, onde, além de suas definições, consta a pontuação de um a três para sua relevância e dificuldade de desenvolvimento, sendo um menos relevante e três mais relevante, além do tempo estimado para o desenvolvimento de cada uma delas, sendo cada semana composta por vinte horas distribuídas entre os sete dias da semana:

(1) Como cliente:

- **Como usuário, eu quero fazer uma busca no site para saber onde encontrar o produto.**
 - Classificação cliente (importância): 2
 - Classificação desenvolvedor (dificuldade): 1
 - Tempo estimado de desenvolvimento: 80 horas, 4 semanas.

- **Como usuário, eu quero visualizar as lanchonetes em um mapa para saber a localização do estabelecimento.**
 - Classificação cliente (importância): 1
 - Classificação desenvolvedor (dificuldade): 3
 - Tempo estimado de desenvolvimento: 50 horas, 3 semanas.

- **Como usuário, eu quero visualizar o cardápio de uma lanchonete para checar os preços, qualidades e produtos.**
 - Classificação cliente (importância): 3
 - Classificação desenvolvedor (dificuldade): 3
 - Tempo estimado de desenvolvimento: 80 horas, 4 semanas.

- **Como usuário, eu quero acessar o site para visualizar o meu perfil pessoal.**
 - Classificação cliente (importância): 2
 - Classificação desenvolvedor (dificuldade): 2
 - Tempo estimado de desenvolvimento: 50 horas, 3 semanas.

- **Como usuário, eu quero modificar meu perfil pessoal para manter meus**

dados atualizados.

- Classificação cliente (importância): 2
 - Classificação desenvolvedor (dificuldade): 2
 - Tempo estimado de desenvolvimento: 50 horas, 3 semanas.
-
- **Como usuário, eu quero avaliar os produtos que eu consumi em uma escala de 0 a 5 estrelas, para dar um feedback para a empresa e auxiliar na decisão de outros usuários.**
 - Classificação cliente (importância): 1
 - Classificação desenvolvedor (dificuldade): 2
 - Tempo estimado de desenvolvimento: 100 horas, 5 semanas.
-
- **Como usuário, eu quero visualizar o meu histórico de avaliações dos produtos, para verificar os produtos que comprei e avaliei.**
 - Classificação cliente (importância): 1
 - Classificação desenvolvedor (dificuldade): 2
 - Tempo estimado de desenvolvimento: 20 horas, 1 semana.

(2) Como Empresa:

- **Como empresa, eu quero acessar o site para visualizar o perfil da minha lanchonete.**
 - Classificação cliente (importância): 2
 - Classificação desenvolvedor (dificuldade): 2
 - Tempo estimado de desenvolvimento: 50 horas, 3 semanas.
-
- **Como empresa, eu quero cadastrar os produtos oferecidos no cardápio da minha lanchonete.**
 - Classificação cliente (importância): 3
 - Classificação desenvolvedor (dificuldade): 2
 - Tempo estimado de desenvolvimento: 100 horas, 5 semanas.
-
- **Como empresa, eu quero modificar o cardápio da minha lanchonete para manter o cadastro dos produtos atualizado.**
 - Classificação cliente (importância): 3

- Classificação desenvolvedor (dificuldade): 2
- Tempo estimado de desenvolvimento: 70 horas, 4 semanas.

5) Backlog:

Após a definição as histórias de usuário e definições de relevância e dificuldade de cada uma delas, foi possível desenvolver as funcionalidades abaixo listadas:

- Cadastro de usuários: clientes e empresas;
- Alteração de dados cadastrais;
- Empresas cadastradas em listas;
- Cardápio das empresas e restaurante universitário.

Devido a dificuldades com a programação e tempo hábil de desenvolvimento, as funcionalidades abaixo estão como backlog e poderão ser desenvolvidas no futuro:

- Empresas cadastradas em mapa com localização exata;
- Avaliação de produtos e empresas;
- Melhorias no layout do sistema;
- Exclusão de perfil já cadastrado.

6) Planejamento das iterações e resultados:

Para o desenvolvimento do sistema as iterações foram feitas sempre após a entrega de uma das funcionalidades por um dos desenvolvedores. Ou seja, através de uma programação em pares, quando um dos desenvolvedores entregava uma tarefa, outro membro do grupo conferia a funcionalidade afim de encontrar detalhes que por ventura não haviam sido desenvolvidas originalmente ou até mesmo indicar pontos de melhoria. Com essa ação, foi possível minimizar muitos problemas no desenvolvimento, além de permitir um aprimoramento dos membros do grupo que passaram a ter maior percepção de pontos do sistema.

Além da iteração entre os membros do grupo, sempre que havia a entrega de uma funcionalidade, o cliente era informado e realizava um teste rápido, onde dava o aceite ou propunha melhorias. Com isso, a primeira entrega final do sistema pôde contemplar grande parte do que foi solicitado e está de acordo com as especificações e desejos do cliente.

7) Protótipos Lo-Fi UI e Storyboards.

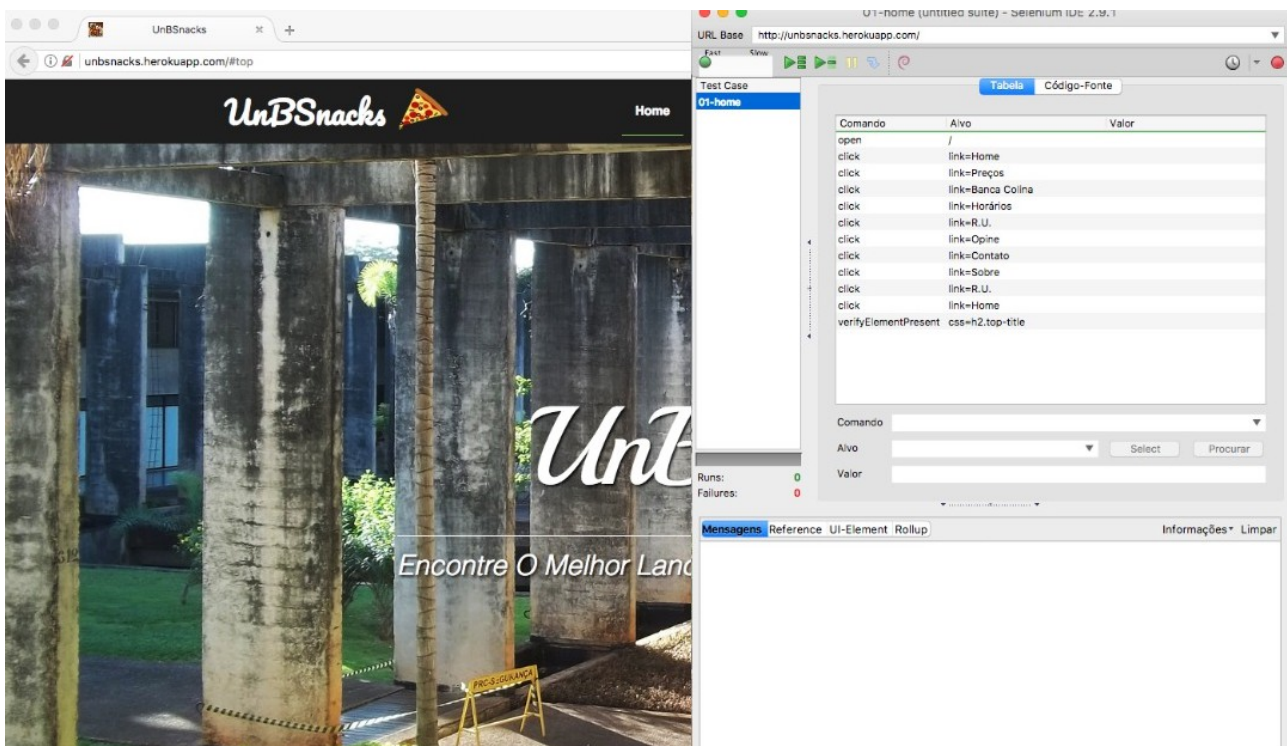
A documentação referente aos protótipos de baixa fidelidade, Lo-Fi, e os Storyboards podem ser encontrados em local específico dentro de repositório *GitHub*.

8) Testes e cobertura:

A documentação referente aos testes do sistema poderá ser encontrada em local específico dentro de repositório *GitHub*. Cabe ressaltar que foram realizados testes com a ferramenta *Selenium*, onde foram feitos *scripts* que cobrem as funcionalidades de navegação pela home; cadastro, acesso, navegação e logout dos perfis de clientes e empresas e testes de validação de campos.

Como efeito de demonstração, segue telas com resultado dos scripts de teste:

- Navegabilidade pela Home



O1-home (untitled suite) - Selenium IDE 2.9.1

URL Base: <http://unbsnacks.herokuapp.com/>

Fast Slow

Test Case: **O1-home**

Tabela Código-Fonte

Comando	Alvo	Valor
open	/	
click	link=Home	
click	link=Preços	
click	link=Banca Colina	
click	link=Horários	
click	link=R.U.	
click	link=Opine	
click	link=Contato	
click	link=Sobre	
click	link=R.U.	
click	link=Home	
verifyElementPresent	css=h2.top-title	

Comando:

Alvo: Select Procurar

Valor:

Runs: 1
Failures: 0

Mensagens Reference UI-Element Rollup Informações Limpar

[info] Executing: |click| link=Home | |
[info] Executing: |click| link=Preços | |
[info] Executing: |click| link=Banca Colina | |
[info] Executing: |click| link=Horários | |
[info] Executing: |click| link=R.U. | |
[info] Executing: |click| link=Opine | |
[info] Executing: |click| link=Contato | |
[info] Executing: |click| link=Sobre | |
[info] Executing: |click| link=R.U. | |
[info] Executing: |click| link=Home | |
[info] Executing: |verifyElementPresent| css=h2.top-title | |
[info] Test case passed

- Acesso ao cardápio do R.U.

02-AcessoRU (untitled suite) - Selenium IDE 2.9.1

URL Base: <http://unbsnacks.herokuapp.com/>

Test Case: 01-home, 02-AcessoRU

Comando	Alvo	Valor
click	link=Home	
click	link=R.U.	
click	css=span > span > span > span	

Comando: click
 Alvo: link=Home
 Valor:

Runs: 1
 Failures: 0

Mensagens Reference UI-Element Rollup

click(locator)
 Arguments:
 • locator - an element locator
 Clicks on a link, button, checkbox or radio button. If the click action causes a new page to load (like a link usually does), call `waitForPageToLoad`.

UnBSnacks - BETA

unbsnacks.herokuapp.com/admin/registrarciente

UnBSnacks

Entrar como usuário

Registro do cliente

Pag Inicial do usuário

NOVO CADASTRO

Nome

UnB Snacks Teste 04

Endereço de E-Mail

unbsnackteste04@gmail.com

Senha

Confirma Senha

Telefone

0000000000

CPF

12345678904

Data de Nascimento

01/01/1980

Data do Cadastro

02/07/2017

Nome de Usuário

unbsnacks04

Foto

-

Register

02-CadastrarUser (untitled suite) - Selenium IDE 2.9.1

URL Base: http://unbsnacks.herokuapp.com/

Test Case

01-home

02-CadastrarUser

03-SairUser

04-AcessarUser

05-NavegarUser

06-AlterarUser

Comando

Alvo

Valor

open

/

click

link=Home

clickAndWait

link=Cadastro/Acessar

clickAndWait

link=Registro de novos usuários

type

id=nome

UnB Snacks Teste 04

type

id=email

unbsnackteste04@gmail.com

type

id=password

123456

type

id=password-confirm

123456

type

id=cTelefone

0000000000

type

id=cCPF

12345678904

type

id=cNascimento

01/01/1980

type

id=cCadastro

02/07/2017

type

id=cUsuario

unbsnacks04

type

id=cFoto

-

clickAndWait

css=button.btn.btn-primary

verifyElementPresent

link=UnB Snacks Teste 04

Comando

Alvo

Valor

open

/

Runs: 0

Failures: 0

Mensagens

Reference

UI-Element

Rollup

open(url)

Arguments:

• url - the URL to open; may be relative or absolute

Opens an URL in the test frame. This accepts both relative and absolute URLs. The "open" command waits for the page to load before proceeding, i.e. the "AndWait" suffix is implicit. Note: The URL must be on the same domain as the runner HTML, due to security restrictions in the browser (Same Origin Policy). If you need to open a URL on another domain, use the Selenium Server to start a new browser session on that domain.

UnBSnacks - BETA

unbsnacks.herokuapp.com/admin

UnBSnacks

Sistema UnBSnacks - Área do usuário


Alterar usuário

Remover usuário (pendente)

Pag Inicial do usuário

Avaiar serviços das lanchonetes

Lista de lanchonetes cadastradas



03-SairUser (untitled suite) - Selenium IDE 2.9.1

URL Base: http://unbsnacks.herokuapp.com/

Test Case

01-home

02-CadastrarUser

03-SairUser

04-AcessarUser

05-NavegarUser

06-AlterarUser

Comando

Alvo

Valor

open

/admin

verifyElementPresent

link=UnB Snacks Teste 04

click

link=UnB Snacks Teste 04

clickAndWait

link=Logout

verifyElementPresent

link=Contato

Comando

Alvo

Valor

open

/admin

Runs: 1

Failures: 0

Mensagens

Reference

UI-Element

Rollup

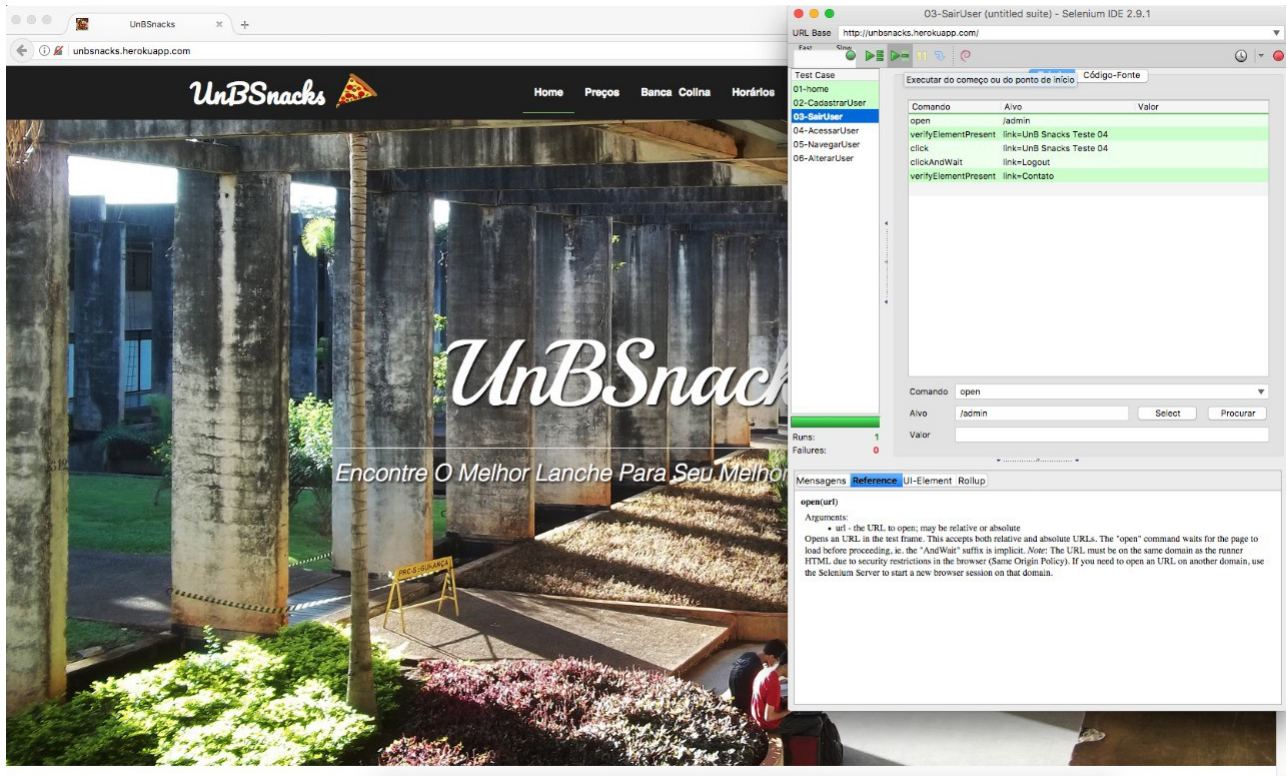
open(url)

Arguments:

• url - the URL to open; may be relative or absolute

Opens an URL in the test frame. This accepts both relative and absolute URLs. The "open" command waits for the page to load before proceeding, i.e. the "AndWait" suffix is implicit. Note: The URL must be on the same domain as the runner HTML, due to security restrictions in the browser (Same Origin Policy). If you need to open a URL on another domain, use the Selenium Server to start a new browser session on that domain.

• Logout do Sistema - Cliente

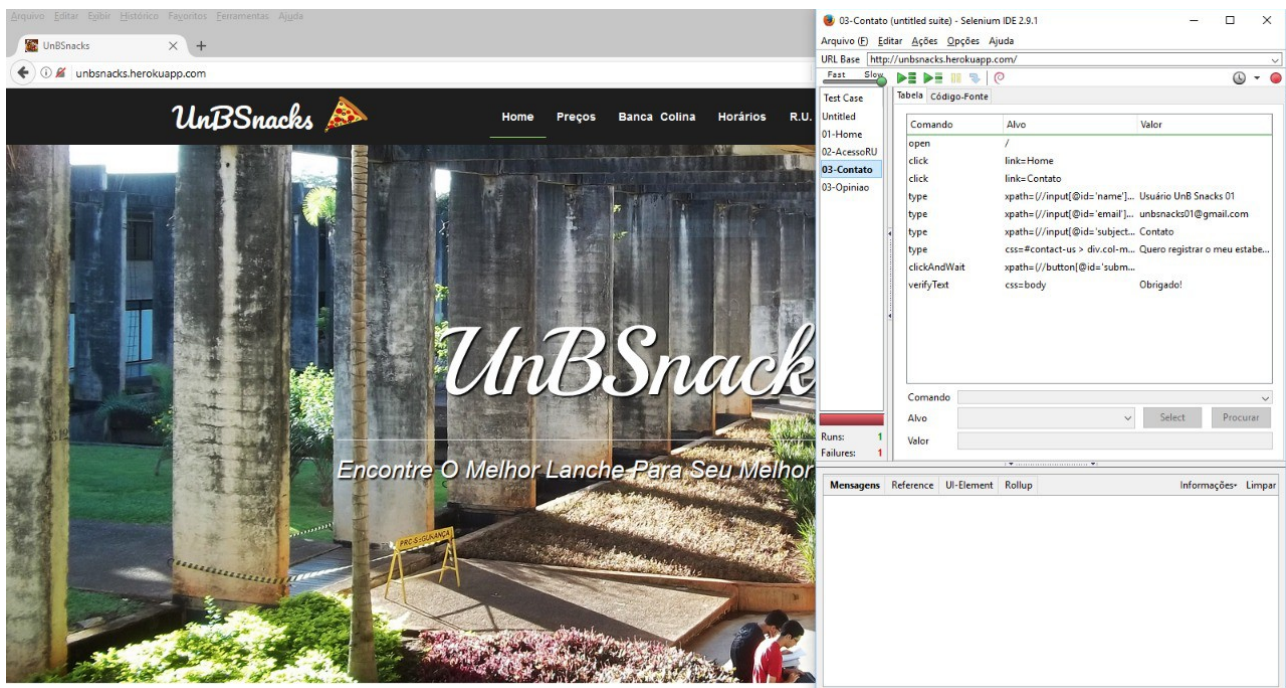


The screenshot shows a Selenium IDE window titled '03-SairUser (untitled suite) - Selenium IDE 2.9.1'. The URL Base is 'http://unbsnacks.herokuapp.com/'. The test case '03-SairUser' is selected in the Test Case list. The test steps are as follows:

Comando	Alvo	Valor
open	/admin	
verifyElementPresent	link=UnB Snacks Teste 04	
click	link=UnB Snacks Teste 04	
clickAndWait	link=Logout	
verifyElementPresent	link=Contato	

The background image shows the UnBSnacks website with the text 'UnBSnacks' and 'Encontre O Melhor Lanche Para Seu Melhor'.

• Contato



The screenshot shows a Selenium IDE window titled '03-Contato (untitled suite) - Selenium IDE 2.9.1'. The URL Base is 'http://unbsnacks.herokuapp.com/'. The test case '03-Contato' is selected in the Test Case list. The test steps are as follows:

Comando	Alvo	Valor
open	/	
click	link=Home	
click	link=Contato	
type	xpath=//input[@id='name']...	Usuário UnB Snacks 01
type	xpath=//input[@id='email']...	unbsnacks01@gmail.com
type	xpath=//input[@id='subject']...	Contato
type	css=#contact-us > div.col-m...	Quero registrar o meu estabe...
clickAndWait	xpath=//button[@id='subm...	
verifyText	css=body	Obrigado!

The background image shows the UnBSnacks website with the text 'UnBSnacks' and 'Encontre O Melhor Lanche Para Seu Melhor'.

• Opinião

The screenshot shows the Selenium IDE 2.9.1 interface. The browser window displays the UnBSnacks website with the URL <http://unbsnacks.herokuapp.com/>. The test case 'Opinião' is selected in the Test Case list. The 'Tabela Código-Fonte' (Source Code Table) shows the following commands:

Comando	Alvo	Valor
open	/	
click	link=Home	
click	link=Opine	
type	id=name	Usuário UnB Snacks 01
type	id=email	unbsnacks01@gmail.com
type	id=subject	Elogio.
type	id=message	Sistema é show!
clickAndWait	id=submit	
verifyText	css=body	Obrigado!

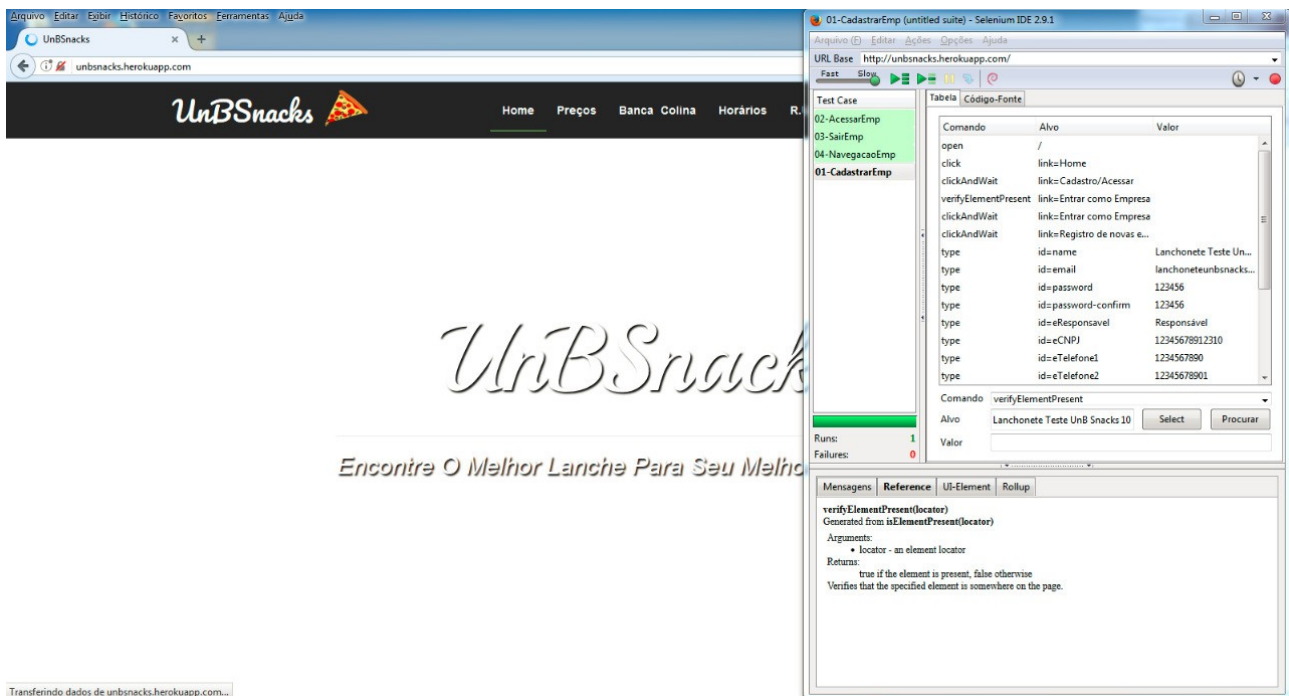
The 'Runs' counter shows 1 run and 0 failures. The 'Mensagens' (Messages) tab is active, showing the 'open(url)' command details.

The screenshot shows the Selenium IDE 2.9.1 interface after the test case 'Opinião' has been executed. The browser window displays the UnBSnacks website with the URL <http://unbsnacks.herokuapp.com/contact.php>. The test case 'Opinião' is selected in the Test Case list. The 'Tabela Código-Fonte' (Source Code Table) shows the following commands:

Comando	Alvo	Valor
open	/	
click	link=Home	
click	link=Opine	
type	id=name	Usuário UnB Snacks 01
type	id=email	unbsnacks01@gmail.com
type	id=subject	Elogio.
type	id=message	Sistema é show!
clickAndWait	id=submit	
verifyText	css=body	Obrigado!

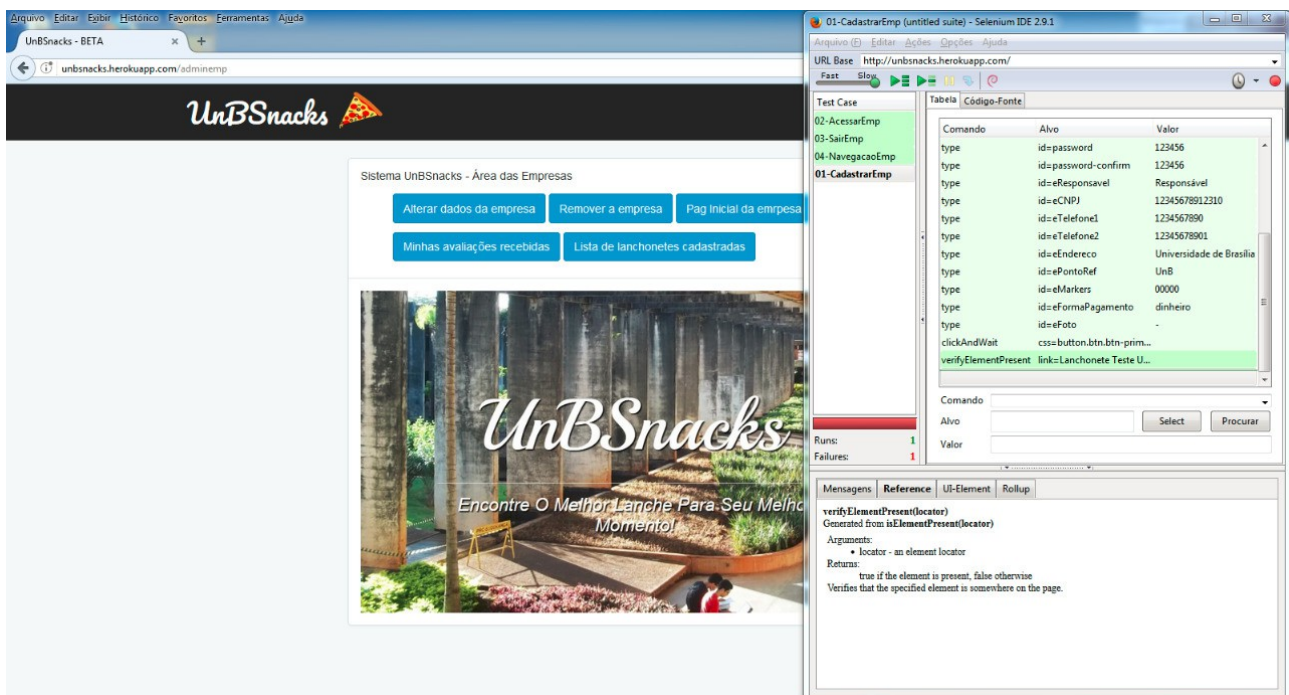
The 'Runs' counter shows 1 run and 0 failures. The 'Mensagens' (Messages) tab is active, showing the 'open(url)' command details.

• Cadastro de Empresa



Comando	Alvo	Valor
open	/	
click	link=Home	
clickAndWait	link=Cadastro/Acessar	
verifyElementPresent	link=Entrar como Empresa	
clickAndWait	link=Entrar como Empresa	
clickAndWait	link=Registro de novas e...	
type	id=name	Lanchonete Teste Un...
type	id=email	lanchoneteunbsnacks...
type	id=password	123456
type	id=password-confirm	123456
type	id=eResponsavel	Responsável
type	id=eCNPJ	12345678912310
type	id=eTelefone1	1234567890
type	id=eTelefone2	12345678901
Comando	verifyElementPresent	
Alvo	Lanchonete Teste UnBSnacks 10	Select
Valor		Procurar

• Acesso perfil da Empresa




Comando	Alvo	Valor
type	id=password	123456
type	id=password-confirm	123456
type	id=eResponsavel	Responsável
type	id=eCNPJ	12345678912310
type	id=eTelefone1	1234567890
type	id=eTelefone2	12345678901
type	id=eEndereco	Universidade de Brasília
type	id=ePontoRef	UnB
type	id=eMarkers	00000
type	id=eFormaPagamento	dinheiro
type	id=eFoto	-
clickAndWait	css=button.btn.btn-prim...	
verifyElementPresent	link=Lanchonete Teste U...	
Comando		
Alvo		Select
Valor		Procurar

Arquivo Editar Exibir Histórico Favoritos Ferramentas Ajuda

UnBSnacks - BETA

unbsnacks.herokuapp.com/adminemp

UnBSnacks 

Sistema UnBSnacks - Área das Empresas


Alterar dados da empresa

Remover a empresa

Pag Inicial da empresa

Minhas avaliações recebidas

Lista de lanchonetes cadastradas



UnBSnacks

Encontre O Melhor Lanche Para Seu Melhor Momento!

02-AcessarEmp (untitled suite) - Selenium IDE 2.9.1

Arquivo Editar Ações Opções Ajuda

URL Base http://unbsnacks.herokuapp.com/

Fast Slow

Test Case

02-AcessarEmp

03-SairEmp

04-NavegacaoEmp

01-CadastrarEmp

Comando	Alvo	Valor
click	link=Home	
clickAndWait	link=Cadastro/Acessar	
clickAndWait	link=Entrar como Empresa	
verifyElementPresent	link=Registro de novas e...	
type	id=email	lanchoneteunbsnacks02...
type	id=password	123456
click	name=remember	
clickAndWait	css=button.btn.btn-prim...	
verifyElementPresent	link=Lanchonete Teste U...	

Comando click

Alvo link=Home

Valor

Select

Procurar

Runs: 1

Failures: 0

Mensagens

Reference

UI-Element

Rollup

click(locator)

Arguments:

- locator - an element locator


Clicks on a link, button, checkbox or radio button. If the click action causes a new page to load (like a link usually does), call waitForPageToLoad.

• Logout do Sistema - Empresa

Arquivo Editar Exibir Histórico Favoritos Ferramentas Ajuda

UnBSnacks - BETA

unbsnacks.herokuapp.com/adminemp

UnBSnacks 

Sistema UnBSnacks - Área das Empresas


Alterar dados da empresa

Remover a empresa

Pag Inicial da empresa

Minhas avaliações recebidas

Lista de lanchonetes cadastradas



UnBSnacks

Encontre O Melhor Lanche Para Seu Melhor Momento!

03-SairEmp (untitled suite) - Selenium IDE 2.9.1

Arquivo Editar Ações Opções Ajuda

URL Base http://unbsnacks.herokuapp.com/

Fast Slow

Test Case

02-AcessarEmp

03-SairEmp

04-NavegacaoEmp

01-CadastrarEmp

Comando	Alvo	Valor
open	/adminemp	
verifyElementPresent	link=Lanchonete Teste U...	
clickAndWait	link=Pag Inicial da emp...	
click	link=Lanchonete Teste U...	
clickAndWait	link=Logout	
verifyElementPresent	link=Contato	

Comando open

Alvo /adminemp

Valor

Select

Procurar

Runs: 1

Failures: 0

Mensagens

Reference

UI-Element

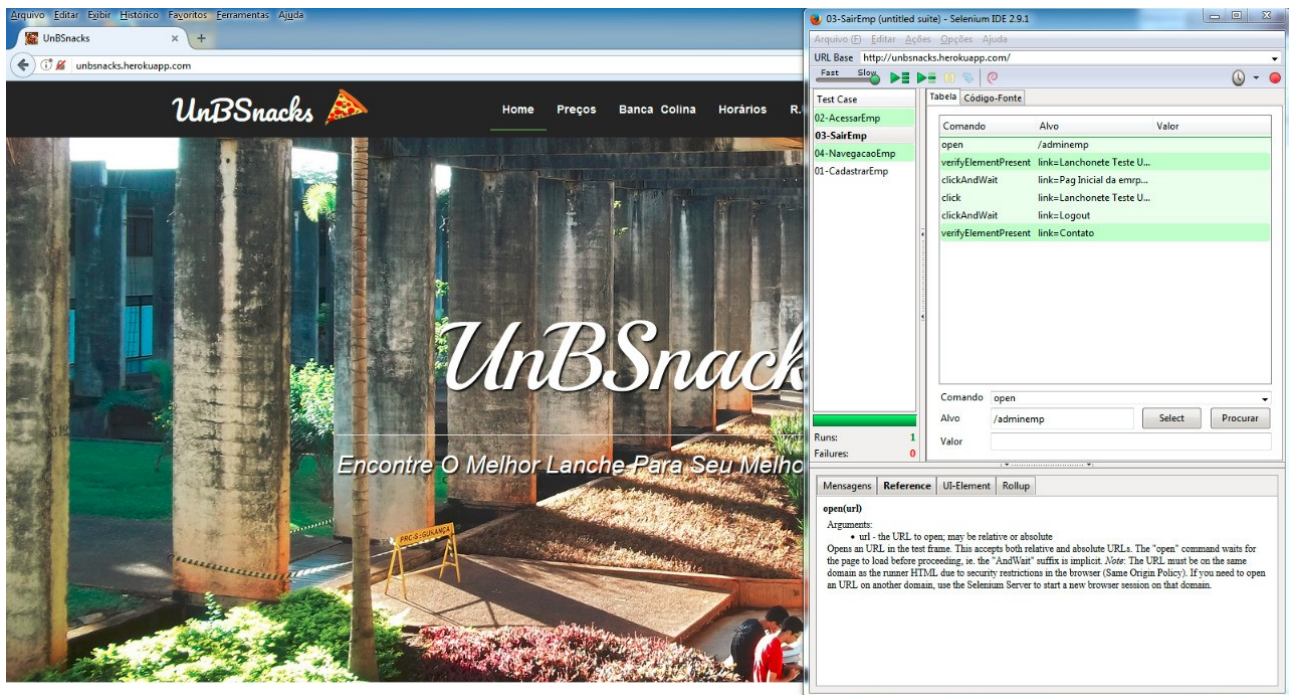
Rollup

open(url)

Arguments:

- url - the URL to open; may be relative or absolute

Opens an URL in the test frame. This accepts both relative and absolute URLs. The "open" command waits for the page to load before proceeding, i.e. the "AndWait" suffix is implicit. *Note:* The URL must be on the same domain as the runner HTML due to security restrictions in the browser (Same Origin Policy). If you need to open an URL on another domain, use the Selenium Server to start a new browser session on that domain.



9) Decisões de projeto e spikes

Dentre todas as decisões de projeto cabe ressaltar as mais importantes, como a definição da arquitetura que foi utilizada e os critérios para priorização de histórias de usuário, além da distribuição das tarefas entre os membros do grupo. Por não termos elegidos nenhum dos membros para acompanhar o processo como um todo, todas as decisões foram tomadas em comum acordo entre todos os membros do grupo, onde o voto da maioria definia a forma como trabalharíamos.

Para nos auxiliar a atingir os objetivos propostos, dividimos as tarefas em quatro spikes, conforme descrito abaixo em detalhes:

- **1º Spike:**
 - Levantamento dos requisitos do sistema junto ao cliente;
 - Definição da arquitetura a ser utilizada;
 - Aprender a utilizar as ferramentas de arquitetura (linguagem de programação, framework, etc);
 - Divisão das tarefas entre os membros do grupo.
 - Tempo: 15 dias
- **2º Spike:**

- Desenvolvimento das histórias de usuário classificadas como prioritárias;
- Elaboração de protótipos Lo-Fi e Storyboards.
 - Tempo: 30 dias
- **3º Spike:**
 - Continuação do desenvolvimento das histórias de usuário classificadas como prioritárias;
 - Desenvolvimento das histórias de usuário classificadas como intermediárias;
 - Elaboração da documentação de sistema.
 - Tempo: 30 dias
- **4º Spike:**
 - Finalização do desenvolvimento das histórias de usuário classificadas como prioritárias e intermediárias;
 - Elaboração e execução de scripts de testes;
 - Aprimoramento do sistema para primeira entrega final.
 - Tempo: 30 dias

Ao longo do processo foi necessário abrimos mão do desenvolvimento das histórias de usuário não prioritárias, uma vez que sub estimamos o tempo que seria necessário para desenvolver algumas partes do sistema, com isso, não houve tempo hábil para entregarmos todas as funcionalidades que estavam previstas nas histórias de usuário iniciais, nos atendo ao MVP.

10) Lições aprendidas:

Ao final do processo de desenvolvimento do sistema, podemos destacar como lições aprendidas:

- Não houve a escolha de um componente do grupo para direcionar e acompanhar o que e como estava sendo desenvolvido o projeto, com isso, encontramos muitos problemas no tempo de execução de algumas tarefas, uma vez que, mesmo delimitando o tempo de entrega das spikes, os prazos não foram cumpridos muito bem.
- A demora na decisão do grupo de escolher se o desenvolvimento do sistema

seria com ou sem a utilização de um framework fez com que algumas funcionalidades ficassem comprometidas por utilizarmos tempo de desenvolvimento em uma plataforma que não foi utilizada ao final do trabalho.

- A linguagem de programação php atendeu muito bem as necessidades do sistema e se mostrou como uma boa decisão de projeto.
- Scripts de testes desenvolvidos com o *Selenium* auxiliam muito no processo de testes de repetição, uma vez que pode ser testado o sucesso ou fracasso de uma funcionalidade através de várias entradas para um mesmo campo.

11) Aceite do cliente para o MVP:

Apesar das dificuldades encontradas durante o processo de desenvolvimento do UnBSnacks, o cliente se mostrou satisfeito com a primeira entrega final do sistema, aprovando o MVP e concordando com a dificuldade na elaboração de algumas funcionalidades, como a que previa um mapa contendo a localização de todas as lanchonetes do Campus. Ao final, o cliente também propôs melhorias em algumas funcionalidades entregues, como o layout dos perfis dos clientes e das empresas que poderia apresentar maiores detalhes.