



Universidade Federal do Ceará
Centro de Ciências
Departamento de Computação
Verificação, Validação e Teste de Software (CK0241)
Verificação e Validação de Software (CKP9012)

TB01 - Escopo do projeto de V&V
Food Delivery Website - Tomato

Gabriel Moreira de Andrade - 536708

Álvaro Siqueira Galvão - 537649

Fábio Agostinho Filho - 538521

Histórico de versões

Versão	Data	Responsável	Descrição
1.0	06/11/2024	Gabriel Moreira	Criação do documento
1.1	08/11/2024	Gabriel, Álvaro e Fábio	Determinação e análise dos requisitos funcionais.
1.2	08/11/2024	Álvaro	Introdução
1.3	08/11/2024	Gabriel Moreira	Aplicação
1.4	08/11/2024	Fábio	Conclusão e Referências

Sumário

1. Introdução	3
1.1. A aplicação	3
2. Requisitos Funcionais	3
3. Conclusão	4
Referências	4

1. Introdução

Para objeto de nosso Projeto para a disciplina de V&V, visando aplicar a cada etapa os procedimentos e técnicas que serão estudados durante o curso, escolhemos o software *Tomato*, de gerenciamento de pedidos de alimentos. A seguir estão detalhados o funcionamento da aplicação bem como os requisitos funcionais aos quais o software se propõe a desempenhar e que servirão de elementos para posteriores trabalhos.

1.1. A aplicação

O *Tomato* é uma aplicação web FullStack projetada para gerenciar pedidos de alimentos de forma simples e intuitiva, com foco em aprendizado e acessibilidade para desenvolvedores. A aplicação é de código aberto (*open source*). Ela foi construída com a stack **MERN** (MongoDB, Express.js, React.js e Node.js). Os códigos estão hospedados no github: <https://github.com/AthrvaNaik/Tomato-FoodOrdering?tab=readme-ov-file>.

O principal objetivo do *Tomato* é oferecer um ambiente onde clientes e administradores possam interagir com o sistema de pedidos de maneira simples:

Lado do Cliente: Os usuários podem navegar por uma lista de pratos, visualizar descrições e detalhes, e adicionar itens ao carrinho para fazer pedidos.

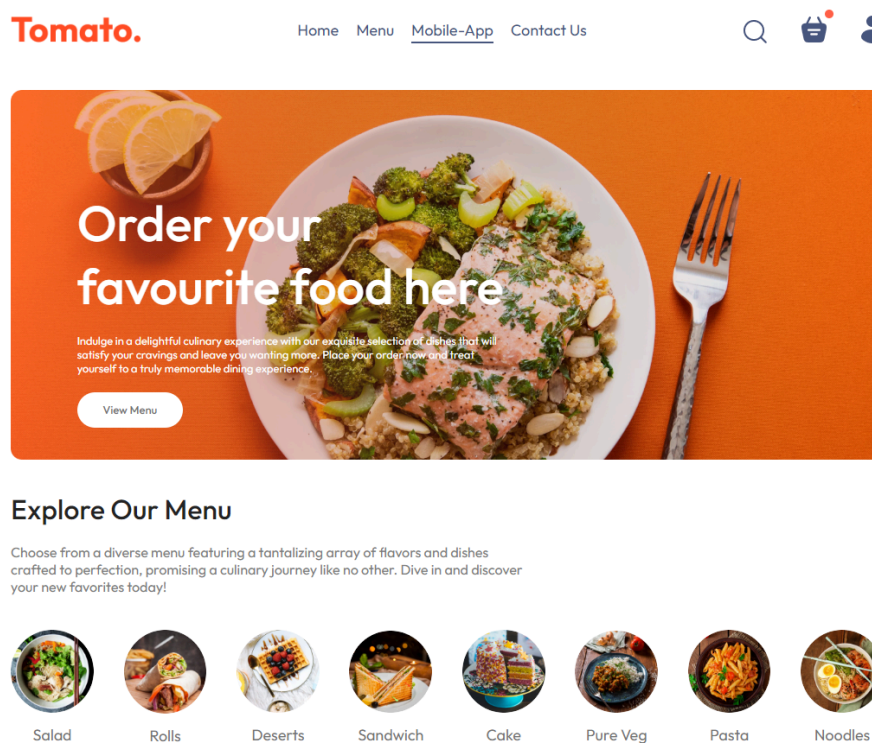


Figura 1: Tela principal da aplicação

Top dishes near you

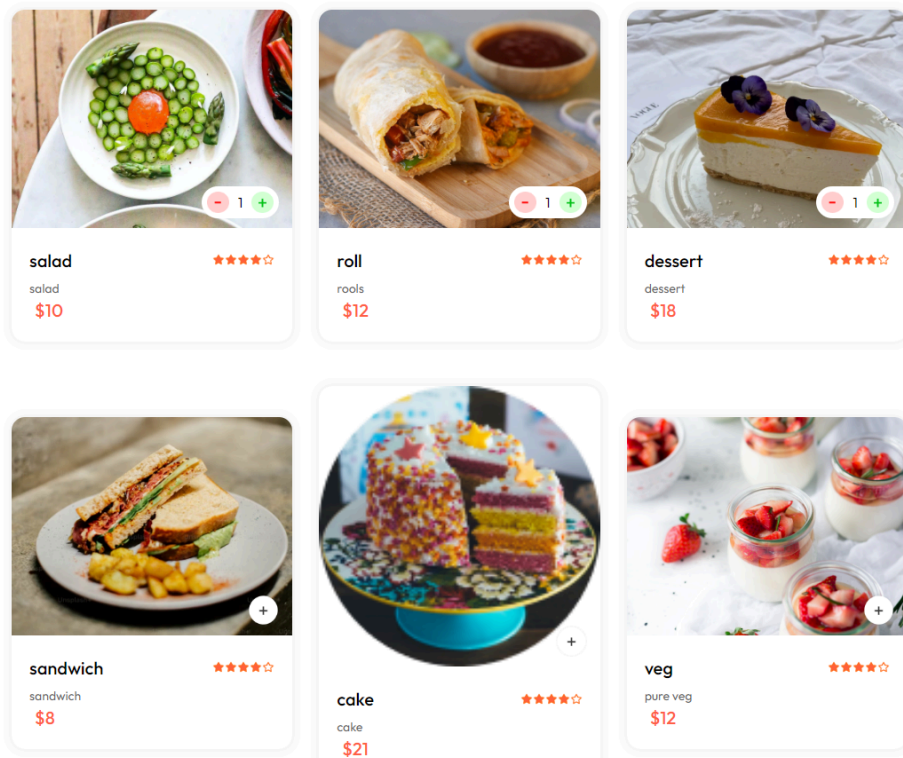





Figura 2: Tela dos pratos

Items	Title	Price	Quantity	Total	Remove
	salad	\$10	1	\$10	x
	roll	\$12	1	\$12	x
	dessert	\$18	1	\$18	x

Cart Total

Subtotal	\$40
Delivery Fee	\$2
Total	\$42

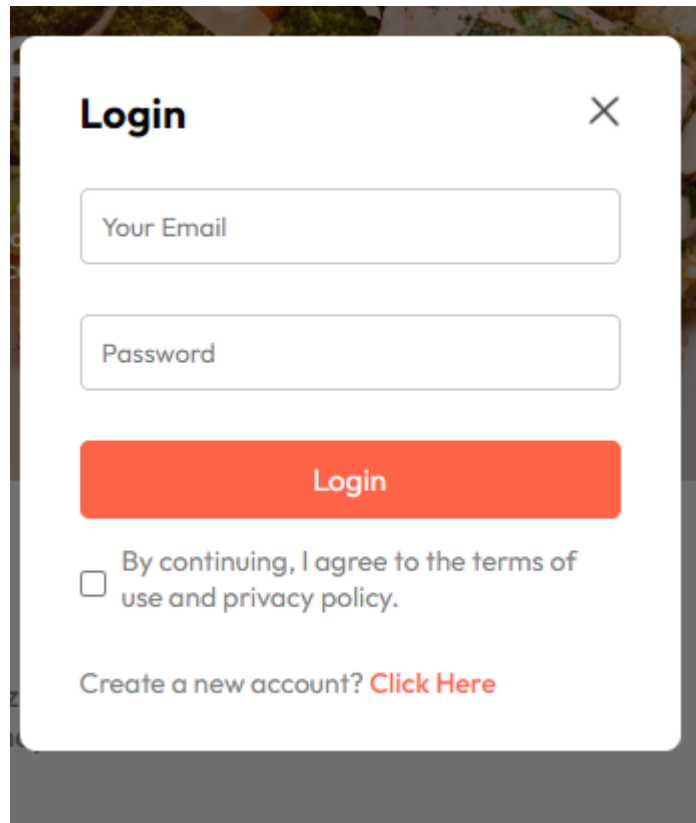
Proceed To Checkout

If you have a promo code, Enter it here

promo code

Apply

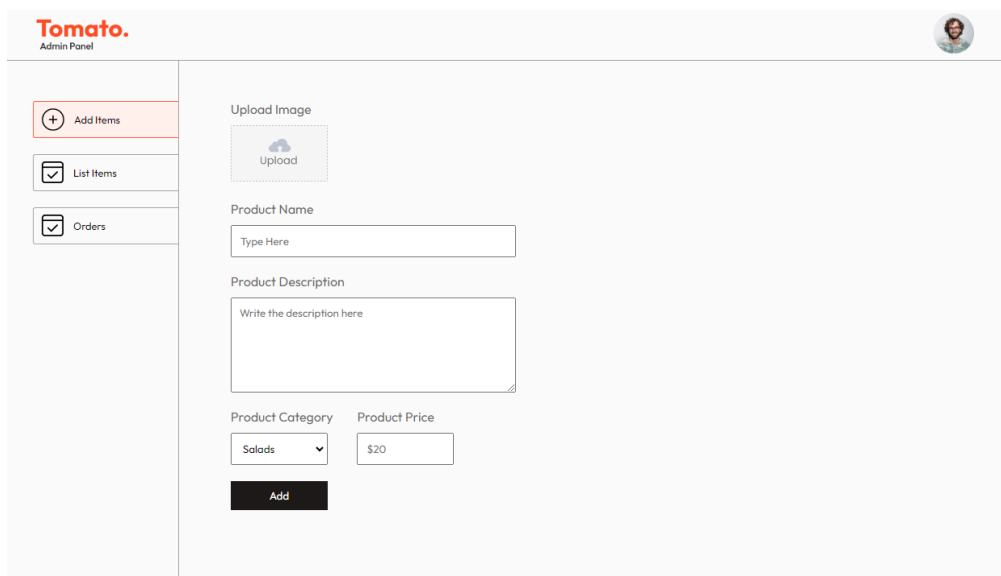
Figura 3: Tela para finalizar o pedido



A login modal window with a white background and rounded corners, set against a blurred background of tomatoes. The modal has a title 'Login' in bold black text at the top left and a close button 'X' at the top right. It contains two input fields: 'Your Email' and 'Password', both with light gray borders. Below these is a prominent orange 'Login' button. Under the button is a checkbox followed by the text 'By continuing, I agree to the terms of use and privacy policy.' At the bottom, there is a link that says 'Create a new account? Click Here'.

Figura 4: Login

Lado do Administrador: Os administradores têm acesso a uma interface que permite o controle sobre os pedidos recebidos e os pratos disponíveis. É possível adicionar, editar ou remover pratos e atualizar o status dos pedidos.



The 'Tomato. Admin Panel' interface for creating a new dish. On the left is a sidebar with three options: 'Add Items' (highlighted with a red border and a plus icon), 'List Items' (with a checkmark icon), and 'Orders' (with a checkmark icon). The main area contains the following fields: 'Upload Image' with an 'Upload' button and a dashed box; 'Product Name' with a text input field labeled 'Type Here'; 'Product Description' with a larger text area labeled 'Write the description here'; 'Product Category' with a dropdown menu currently showing 'Salads'; and 'Product Price' with a text input field showing '\$20'. At the bottom of the form is a black 'Add' button.

Figura 5: Tela criar novos pratos

<

Figura 6: Tela para ver os pratos disponíveis e apagar, caso necessário.





<div> <div>+</div> <div>Add Items</div> </div> <div> <div>✓</div> <div>List Items</div> </div> <div> <div>✓</div> <div>Orders</div> </div>	<div>Order Page</div> <div> <div> <div></div> <div> <div>food1 x 1</div> <div>Items: 1</div> <div>\$23</div> <div>Delivered ▼</div> </div> </div> <div> <div>Athrva NAIK</div> <div>Navale,Nirmal,Nallasopara West,Palghar</div> <div>401304,</div> <div>mumbai, Maharashtra, 401304, India</div> <div>07666967661</div> </div> </div> <div> <div> <div></div> <div> <div>food1 x 1, food2 x 1</div> <div>Items: 2</div> <div>\$36</div> <div>Delivered ▼</div> </div> </div> <div> <div>Athrva NAIK</div> <div>Navale,Nirmal,Nallasopara West,Palghar</div> <div>401304,</div> <div>mumbai, Maharashtra, 401304, India</div> <div>07666967661</div> </div> </div> <div> <div> <div></div> <div> <div>roll x 1, veg x 1, pasta x 1</div> <div>Items: 3</div> <div>\$39</div> <div>Food Processi ▼</div> </div> </div> <div> <div>Athrva NAIK</div> <div>Navale,Nirmal,Nallasopara West,Palghar</div> <div>401304,</div> <div>mumbai, Maharashtra, 401304, India</div> <div>07666967661</div> </div> </div> <div> <div> <div></div> <div> <div>salad x 1, dessert x 1</div> <div>Items: 2</div> <div>\$30</div> <div>Delivered ▼</div> </div> </div> </div>
--	--

Figura 7: Tela para controlar os status dos pedidos.

2. Requisitos Funcionais

Analisando o projeto e o repositório, chegamos a conclusão que esses são os requisitos funcionais

Código	Descrição do requisito
RF001	Login como cliente
RF002	Sign-up como cliente
RF003	Aumentar os produtos do carrinho como cliente
RF004	Ver histórico de compras como cliente
RF005	Logout como cliente
RF006	Diminuir os produtos do carrinho como cliente
RF007	Filtrar os produtos por categoria como cliente
RF008	Acessar ao carrinho de compras como cliente
RF009	Acessar ao status do pedido como cliente
RF010	Remover produtos como admin
RF011	Acessar aos pedidos feitos como admin
RF012	Adicionar produtos como admin
RF013	Editar o status de cada pedido como admin

3. Conclusão

Portanto, essa aplicação se mostrou como uma excelente oportunidade de utilização para testes e análises, tendo em vista a alta quantidade de requisitos funcionais, além do fato de ter uma interface gráfica e um código simples, mostrando ter conteúdo a ser testado e com uma facilidade de entender o que pode estar dando erro e o que está funcionando corretamente.

Referências

MARTIN, Robert C. *Código limpo: habilidades práticas do agile software*. Rio de Janeiro: Alta Books, 2009.

PRESSMAN, Roger S. *Engenharia de Software: conceitos e práticas*. 8. ed. São Paulo: McGraw-Hill, 2010.