



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

TB02 - Plano de V&V

E-Commerce

Grimberg Cryzan - 362964
Beatriz Monte - 371781
José João Silva - 408847
Levir César Ribeiro - 400555

Histórico de versões

Versão	Data	Autor	Descrição
1.0	14/12/2020	José João	Criação da versão inicial do plano de V&V
2.0	14/12/2020	José João	Criação da introdução, aplicação, funcionamento, tecnologias usadas, escopo e estratégias, requisitos funcionais e não funcionais. Edição do ambiente. Criação das ferramentas, treinamento, produtos de V&V.a
3.0	15/12/2020	José João	Atualização dos produtos de V&V e criação do cronograma.
4.0	16/12/2020	José João	Atualização dos requisitos, escopo e estratégia de testes, artefatos e plano de comunicação. Criação do cronograma, riscos, conclusão e introdução.
5.0	17/12/2020	Levir César	Identificador de Plano de Teste, Bibliografia, Recursos Humanos, Glossário
6.0	17/12/2020	Beatriz Monte	Atualização do treinamento e papéis e responsabilidades.

Sumário

1. Identificador do Plano de Teste	4
2. Introdução	4
2.1. A aplicação	4
3. Escopo e Estratégia de Teste	4
3.1. Requisitos Funcionais	4
3.2. Requisitos Não Funcionais	6
4. Ambiente	6
4.1. Ferramentas	6
5. Recursos Humanos	6
5.1. Papéis e Responsabilidades	6
5.2. Treinamento	7
6. Produtos de V&V	7
7. Cronograma	7
8. Riscos	7
9. Conclusão	8
10. Referências	8
Glossário	8

1. Identificador do Plano de Teste

1.1 Título: “Verificação e Validação do Aplicativo E-commerce”.

1.2 Duração prevista: 2 semanas.

1.3 Instituição executora do Teste: Grupo da disciplina.

1.4 Entidade coordenadora: Universidade Federal do Ceará.

1.5 Local, data: Fortaleza, 17/dezembro/2020

2. Introdução

A criação do plano V&V define o planejamento do plano de testes de um projeto que vai andar por todas as etapas do desenvolvimento do produto. Com isso, é definido os requisitos a serem testados com suas respectivas estratégias de testes, definindo os artefatos a serem produzidos com seus respectivos prazos, riscos e estratégias de mitigação.

2.1. A aplicação

O E-commerce é uma aplicação open source, desenvolvida para fins educacionais, mas podendo ser estendida para uso comercial. O objetivo é que ele consiga fornecer, através de uma interface amigável, maneiras de gerenciar todos dados básicos presentes em um e-commerce, tais como funcionários, clientes, produtos, categoria de produtos, vendas e dados pessoais. Todos os códigos fonte do projeto estão disponíveis em um repositório aberto e hospedados no github. O link é: <https://github.com/jjoaosilva/ecommerce>

2.1.1. Funcionamento

Ao usar o aplicativo, o usuário terá acesso à área de login, podendo logar como cliente ou funcionário. Caso não tenha cadastro a plataforma fornece suporte para registro de nova conta.

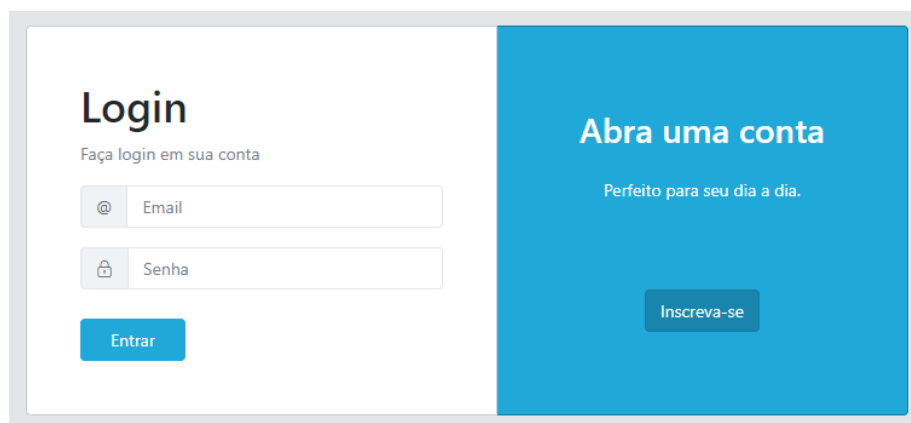


Imagem acima apresenta o login feito como cliente

Como cliente, após fazer login, é possível ver todos os produtos da loja em suas respectivas categorias, adicionar produtos ao carrinho, gerenciar o carrinho, realizar a compra de um produto, ver todo o seu histórico de compras e alterar os seus dados pessoais.

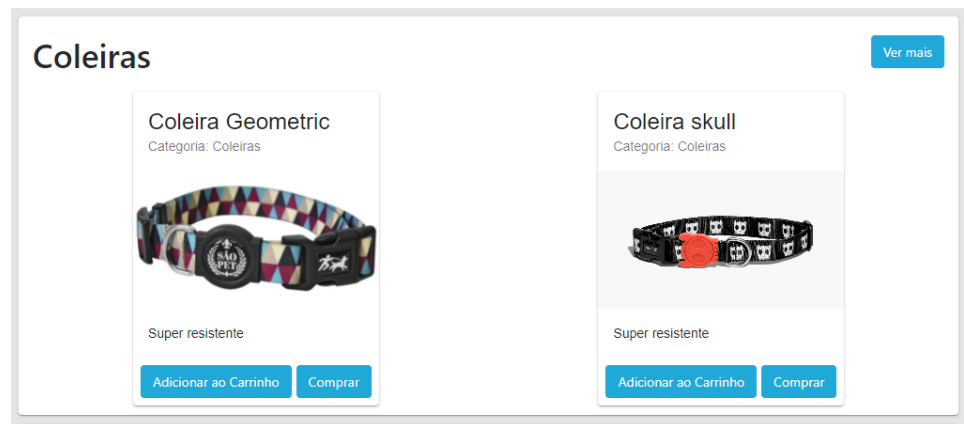
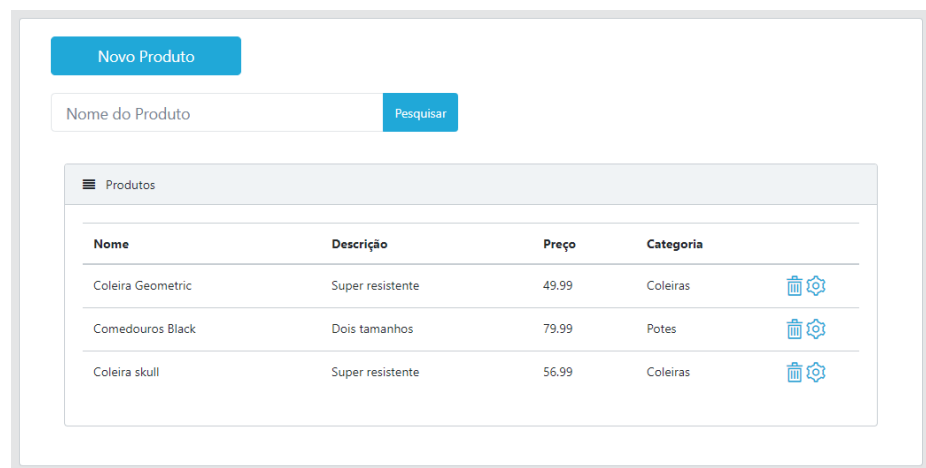


Imagem acima apresenta como o cliente visualiza os produtos de uma categoria

Como funcionário, após fazer login, é possível ver todos, cadastrar, editar e deletar: categorias, produtos, funcionários e clientes.



A imagem acima apresenta como o funcionário tem acesso aos produtos e suas respectivas ações. Esse padrão se repete para categorias, funcionários e clientes.

2.1.2. Tecnologias usadas

Todo frontend foi construído com **React JS** e é compatível com versões acima de 16.8.5 do react. o frontend se comunica com uma API (com arquitetura REST) criada com o microframework **Flask** que é codificada em **Python**, funcionando atualmente (09/12/2020) na versão 3.9.0 do Python e na versão 1.1.2 do Flask. Todos os dados são armazenados em um banco **PostgreSQL**.

3. Escopo e Estratégia de Teste

A aplicação possui no registro de categorias, produtos, funcionários e clientes as suas maiores criticidades, tendo em vista que uma vez que algum desses registros não funcione, não há dinamicidade dos dados dentro do E-commerce.

Portanto, dado as características do backend da aplicação, em que cada entidade citada possui o seu crud bem separado, o E-commerce será submetido a **testes unitários**, visando garantir que os métodos que envolvem o banco de dados do projeto funcionem de maneira correta e sem corrupção dos dados propriamente dito além de **testes de interface** com o intuito de garantir que um usuário possa navegar até uma funcionalidade e consiga concluí-la.

O E-commerce também será submetido a **análise estática de código** para verificação geral das práticas utilizadas no projeto, tendo em vista que a linguagem do backend (Python) é dependente de indentações. Serão realizados **testes de instalação** para validar o guia fornecido pelo autor do projeto e por fim **testes de carga** para analisar o comportamento do E-commerce quando submetido a múltiplas compras.

3.1. Requisitos Funcionais

A tabela abaixo indica os requisitos funcionais que serão validados e quais estratégias serão utilizadas em cada um deles.

Requisito	Descrição	Tipo de estratégia de teste
RF001	Gerenciar um produto (criar, ler, editar e deletar)	Teste unitário e teste de interface
RF002	Gerenciar uma categoria (criar, ler, editar e deletar)	Teste unitário e teste de interface
RF003	Gerenciar um funcionário (criar, ler, editar e deletar)	Teste unitário e teste de interface
RF004	Gerenciar um cliente (criar, ler, editar e deletar)	Teste unitário e teste de interface

A tabela abaixo indica os requisitos funcionais que não serão validados e os seus respectivos motivos.

Requisito	Descrição	Motivo
RF005	Logar como cliente	Apesar de todos serem requisitos importantes para o projeto, estes requisitos não ganharam prioridade neste momento do projeto de V&V pelos motivos já citados no escopo e estratégia de teste.
RF006	Logar como funcionário	
RF007	Adicionar itens ao carrinho	
RF009	Alterar quantidade de um item do carrinho	
RF010	Realizar uma compra	
RF011	Ver suas compras passadas	
RF012	Alterar dados pessoais	
RF013	Fazer logout	

3.2. Requisitos Não Funcionais

A tabela abaixo indica os requisitos funcionais que serão validados e quais estratégias serão utilizadas em cada um deles. (Os códigos dos requisitos funcionais serão os mesmos indicados no TB01 - Escopo do projeto V&V,)

Requisito não funcional	Descrição	Meta	Tipo de teste
RNF003	Portabilidade: Fornecer um tutorial de instalação.	Validar e verificar o tutorial de instalação fornecido.	Teste de instalação
RNF004	Ter todo o código backend padronizado para futuras expansões.	Ter pelo menos 70% do backend do projeto padronizado.	Análise estática de código
RNF005	Fornecer acesso rápido aos usuários independente da quantidade de acessos.	Finalização com sucesso das transações realizadas em um tempo aceitável.	Testes de carga

4. Ambiente

Do ponto de vista do hardware, será necessário uma máquina que possua as especificações mínimas a seguir: processador Intel Core i3, 4Gb de memória ram e 256Gb em HD. Do ponto de vista de software, para o backend, será necessário a instalação da linguagem de programação Python e suas dependências para o correto funcionamento da API. Para o frontend, será necessário o uso de um gerenciador de pacotes, como o Yarn, para a instalação do React e suas devidas dependências. Para o armazenamento dos dados será necessário um banco PostgreSQL.

4.1. Ferramentas

Será necessário o uso de algum editor de código, por exemplo, Visual Studio Code, para que seja realizada a construção dos testes, scripts e linters para análise estática de código. Será necessário também um SGBD PostgreSQL e alguma ferramenta que traga suporte a requisições a API's, como o Postman, por exemplo.

4.2. Plano de comunicação

A tabela abaixo descreve o plano de comunicação a ser seguido pela equipe de testes.

Artefato	Meio de comunicação	Frequência	Partes interessadas
Todos os artefatos do projeto	Síncrona: reuniões através da plataforma Google Meet Assíncrona: grupo no Telegram para envio de mensagens e arquivos.	Durante o período de cada atividade(datas descritas na seção 7) e a cada atualização necessária	João Levir Beatriz Grimberg

5. Recursos Humanos

5.1. Papéis e Responsabilidades

A tabela abaixo descreve as responsabilidades e quantidades de horas de cada membro da equipe de testes.

Papel	Responsável	Responsabilidade	Horas
Testar e Documentar	Grimberg Cryzan	Testar e documentar o requisito funcional RF001, RNF003	2
Testar e Documentar	Beatriz Monte	Testar e documentar o requisito funcional RF002, RNF005	2
Testar e Documentar	José João Silva	Testar e documentar o requisito funcional RF003	2
Testar e Documentar	Levir César Ribeiro	Testar e documentar o requisito funcional RF004, RNF004	2

5.2. Treinamento

A tabela abaixo informa os conhecimentos necessários para a realização das atividades descritas neste documento.

Tipo de Treinamento/Conhecimento	Metas	Integrantes
Uso de flask	Aprender como funciona a estrutura do micro framework.	Beatriz
SGBD	Aprender a utilizar a interface gráfica e a realizar transações básicas de CRUD.	Levir Beatriz
Pytest	Aprender a utilizar o pacote Pytest para o desenvolvimento dos testes unitários.	João Levir Beatriz
Testes de carga	Aprender como realizar testes de carga para API's Flask.	João

6. Produtos de V&I

A tabela abaixo informa os artefatos que deverão ser entregues durante o projeto de V&V.

Código	Artefato
A001	Plano de testes
A002	Relatório de análise estática
A003	Relatório de falhas
A004	Testes unitários
A005	Relatório de inspeção
A006	Especificação de testes funcionais
A007	Relatório de teste de carga
A008	Relatório de teste de instalação
A009	Relatório de teste de interface

7. Cronograma

A tabela abaixo informa o cronograma para entrega dos artefatos descritos na seção 6.

Atividades	Artefato/ Milestone	Data de início	Data de fim	Responsável
Realizar o planejamento dos testes a serem realizados para o E-commerce.	A001	09/12/2020	17/12/2020	João Levir Beatriz Grimberg
Realização da inspeção de software a fim de verificar a especificação de requisitos e arquitetura do projeto.	A005	22/12/2020	22/12/2020	João Levir Beatriz Grimberg
Realização de análise estática de código para verificação geral das práticas utilizadas no projeto.	A002	07/01/2021	20/01/2021	João Levir Beatriz Grimberg
Elaboração do relatório de testes de instalação.	A008	07/01/2021	20/01/2021	João Levir Beatriz Grimberg
Realizar os testes unitários para garantir que os métodos que envolvem o banco de dados do projeto funcionem de maneira correta e sem corrupção dos dados.	A004	27/01/2021	08/02/2021	João Levir Beatriz Grimberg
Elaboração do relatório de testes de interface.	A009	27/01/2021	08/02/2021	João Levir Beatriz Grimberg
Elaboração da especificação de teste funcional.	A006	18/02/2021	08/03/2021	João Levir Beatriz Grimberg
Elaboração do relatório de falhas.	A003	16/03/2021	30/03/2021	João Levir Beatriz Grimberg
Elaboração do relatório de testes de carga.	A007	16/03/2021	30/03/2021	João Levir Beatriz Grimberg

8. Riscos

A tabela abaixo informa os riscos inerentes ao projeto de V&V e suas respectivas estratégias de mitigação.

Categoria	Risco	Impacto*	Probabilidade**	Estratégia de Mitigação
Recursos Humanos	Equipe reduzida	Médio	Médio	Reformulação da divisão das tarefas entre os membros da equipe dando prioridade às atividades mais importantes ao contexto da disciplina.
Ambiente	Não Disponibilidade de um ambiente de teste	Alta	Baixa	Realização dos testes no ambiente de desenvolvimento.
Ambiente	Testes bloqueados por atraso no desenvolvimento	Alta	Médio	Realização de reuniões para o desenvolvimento em pair programming para a finalização das atividades atrasadas.

*Impacto no projeto: Alto, Médio e Baixo

**Probabilidade de ocorrência: Alta, Média, Baixa

9. Conclusão

Baseado nos requisitos funcionais e não funcionais do E-commerce têm-se então uma prioridade em garantir que os usuários, com enfoque, neste projeto, nos funcionários, consigam cadastrar todas entidades relacionadas ao funcionamento do projeto: categorias, produtos, funcionários e clientes. A partir dessa necessidade foram propostos as seguintes estratégias de teste: testes unitários, testes de interface, análise estática de código, testes de carga e testes de instalação. A partir desse foco, temos então a definição dos artefatos a serem entregues(seção 6), organização do cronograma(seção 7) e construção de estratégias de mitigação a partir dos riscos inerentes ao projeto.

10. Referências

João Silva. Github, 2018. E-commerce. Disponível em:
<<https://github.com/jjoaosilva/ecommerce>>. Acesso em: 08, dezembro de 2020.

Core UI. Github, 2018. Coreui-free-react-admin-template. Disponível em:
<<https://github.com/coreui/coreui-free-react-admin-template>>. Acesso em: 08, dezembro de 2020.

Glossário

Termo	Definição
SGBD	Sistema de Gerenciamento de Banco de Dados
V&V	Verificação e Validação
API	Interface de programação de aplicações