Plano de testes

Introdução

Propósito do sistema

O sistema EasyList é uma iniciativa com foco em auxiliar seu usuário com as suas listas de compras tornando o processo de compra menos custoso e mais objetivo.

A solução será desenvolver um aplicativo mobile que possa ser carregado com usuário até o mercado onde é possível depois de cadastrados os produtos e listas, poder fazer o check dos itens necessários.

O sistema permitirá adição de listas e produtos e também a funcionalidade de marcar os itens já comprados as quantidades e também a funcionalidade de compartilhar as listas criadas para poder encaminhar os itens necessários para outro usuário do sistema.

Requisitos

|  |  |
| --- | --- |
| **Requisitos Funcionais (RF)** | **Requisitos Não Funcionais (NF)** |
| RF001 - Cadastro de produto no banco | NF001.1 - Não pode ter produto sem categoria |
| NF001.2 - Cada produto deverá ter o seu identificador único |
| RF002 - Criar lista para a adição de produtos | NF002.1 - O nome da lista é obrigatório |
| NF002.2 - Não pode ter duas listas com o mesmo nome |
| RF003 - Editar Lista | NF003.1 - O nome da lista não pode estar vazio |
| NF003.2 - Não pode ter duas listas com o mesmo nome |
| RF004 - Excluir Lista | NF004.1 - Todos os produtos da lista devem ser excluídos |
| RF005 - Excluir produto de uma lista | NF005.1 - Excluir também a quantidade do produto |
| RF006 - Adicionar produtos na lista | NF006.1 - Quantidade mínima de produto deve ser 1 |
| NF006.2 - Nome do produto não pode estar vazio |
| NF006.3 - Não poderá haver produtos duplicados na mesma lista, porém pode ter mais de uma quantidade |
| NF006.4 - Não poderá haver produtos com quantidades negativa |
| NF006.5 - Quantidade mínima de produto deve ser 1 |
| NF006.6 - Não haverá lista sem produtos |
| NF006.7 - Salvar produtos personalizados |
| NF006.8 - Permitir a entrada de produtos não cadastrados em banco |
| RF007 - Editar produto na lista | NF007.1 - Quantidade mínima de produto deve ser 1 e máximo 99 |
| NF007.2 - Nome do produto não pode estar vazio |
| NF007.2 - Não poderá haver produtos duplicados na mesma lista, porém pode ter mais de uma quantidade |
| NF007.2 - Não poderá haver produtos com quantidades negativa |
| NF007.2 - Não haverá lista sem produtos |
| NF007.2 - Permitir a entrada de produtos não cadastrados em banco |
| RF008 - Consultar listas | NF008.1 - Deve ser possível consultar todas as listas |
| NF008.2 - Deve ser possível consultas por nome de lista |
| NF008.3 - Deve ser possível consultar listas por produtos |
| RF009 - Pesquisar produtos dentro de uma lista já criada | NF009.1 - Deve ser possível consultar por nome de produto |
| RF010 - Compartilhar Lista Criada | NF010.1 - Deve ser possível gerar o link do compartilhamento |

Detalhamento da abordagem de teste

|  |  |
| --- | --- |
| **Tipo de teste** | Funcional |
| **Subtipo de Teste** | Requisitos |
| **Objetivo do Teste** | Cadastrar um produto no sistema que não está na lista |
| **Requisito que motiva o teste** | RF001 - Cadastro de produto no banco |

|  |  |
| --- | --- |
| **Tipo de teste** | Funcional |
| **Subtipo de Teste** | Requisitos |
| **Objetivo do Teste** | Criar uma lista com nome onde será adicionado os produtos e quantidades |
| **Requisito que motiva o teste** | RF002 - Criar lista para a adição de produtos |

|  |  |
| --- | --- |
| **Tipo de teste** | Funcional |
| **Subtipo de Teste** | Requisitos |
| **Objetivo do Teste** | Editar uma lista previamente criada |
| **Requisito que motiva o teste** | RF003 - Editar Lista |

|  |  |
| --- | --- |
| **Tipo de teste** | Funcional |
| **Subtipo de Teste** | Requisitos |
| **Objetivo do Teste** | Excluir uma lista previamente criada |
| **Requisito que motiva o teste** | RF004 - Excluir Lista |

|  |  |
| --- | --- |
| **Tipo de teste** | Funcional |
| **Subtipo de Teste** | Requisitos |
| **Objetivo do Teste** | Excluir produto de uma lista previamente criada |
| **Requisito que motiva o teste** | RF005 - Excluir produto de uma lista |

|  |  |
| --- | --- |
| **Tipo de teste** | Funcional |
| **Subtipo de Teste** | Requisitos |
| **Objetivo do Teste** | Adicionar produtos em uma lista previamente criada |
| **Requisito que motiva o teste** | RF006 - Adicionar produtos na lista |

|  |  |
| --- | --- |
| **Tipo de teste** | Funcional |
| **Subtipo de Teste** | Requisitos |
| **Objetivo do Teste** | Editar os produtos de uma lista previamente criada |
| **Requisito que motiva o teste** | RF007 - Editar produto na lista |

|  |  |
| --- | --- |
| **Tipo de teste** | Funcional |
| **Subtipo de Teste** | Requisitos |
| **Objetivo do Teste** | Consultar listas previamente criadas |
| **Requisito que motiva o teste** | RF008 - Consultar listas |

|  |  |
| --- | --- |
| **Tipo de teste** | Funcional |
| **Subtipo de Teste** | Requisitos |
| **Objetivo do Teste** | Consultar produtos em listas previamente criadas |
| **Requisito que motiva o teste** | RF009 - Pesquisar produtos dentro de uma lista já criada |

|  |  |
| --- | --- |
| **Tipo de teste** | Funcional |
| **Subtipo de Teste** | Requisitos |
| **Objetivo do Teste** | Compartilhar listas previamente criadas com outros usuarios do aplicativo |
| **Requisito que motiva o teste** | RF010 - Compartilhar Lista Criada |

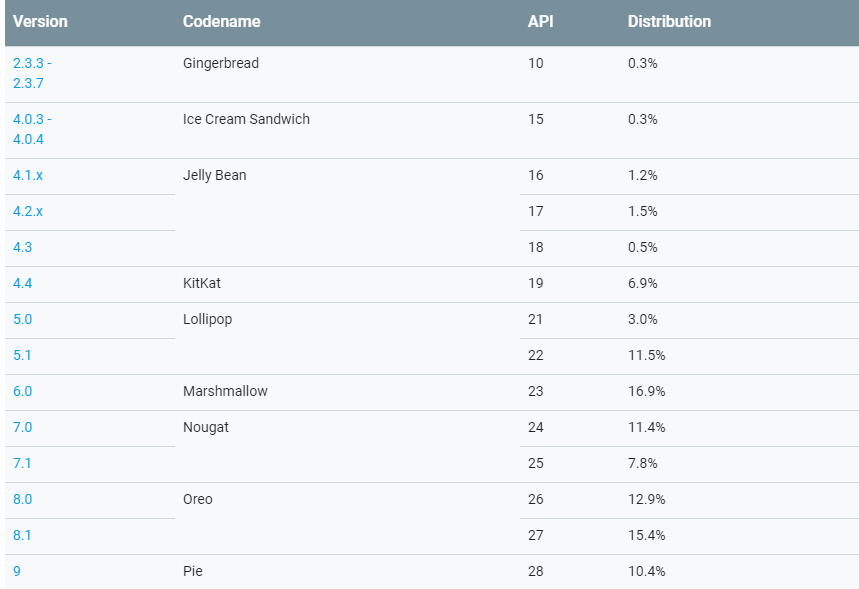
Ferramentas

|  |  |
| --- | --- |
| Gerenciamento de Defeitos |  |
| **Ferramenta** | **Fabricante** |
| Lean Testing | Crowdsourced Testing |
|  |  |
|  |  |
| Casos de Teste (Análise de Execuções) |  |
| **Ferramenta** | **Fabricante** |
| Lean Testing | Crowdsourced Testing |

Ambiente de Teste

Os testes serão realizados de forma manual devido a pequena complexidade do sistema, utilizando o LeanTesting para o gerenciamento de defeitos e também para execução dos casos de teste.

Executaremos testes no android studio com sdks a partir da versão 5.1 para garantir a compatibilidade com o maior número de usuários seguindo os dados do próprio android studio.



Fonte: <https://developer.android.com/about/dashboards?hl=pt-BR>

Casos de Teste

**Contexto: RF001 - Cadastro de produto no banco**

US001 -

**Como**: responsável pelas compras de casa

**Nina quer**: ser capaz de cadastrar produtos que não estão disponíveis

**E:** cadastrar suas categorias

**Para:** poder tornar a lista mais pessoal e compreensível para ela

TC001 -

**Dado**: que a Nina esteja na aplicação e já selecionado uma lista

**Quando:** tentar criar um produto não existente

**Então:** o app devera adicionar ao banco de dados e na lista da Nina o novo produto

**E:** solicitar uma categoria

**Contexto: RF002 – Criar Lista**

US002 -

**Como**: responsável pelas compras de casa

**Nina quer**: ser capaz de guardar todos itens que precisa comprar no mercado

**Para**: manter controle dos itens necessários e otimizar sua ida ao mercado

TC002 -

**Dado**: que a Nina esteja na aplicação

**Quando:** clicar para criar uma lista

**Então:** ela deve ser direcionada a tela de edição da lista

**E:** pode adicionar os produtos desejados

**E:** alista deve conter no mínimo 1 produto

**E:** não deve permitir itens duplicados

**E:** deve possuir nome único

**E:** deve poder salvar a lista em s

**Contexto: RF003 – Editar Lista**

US003 -

**Como**: quem faz as compras ocasionalmente pós serviço

**Bruno quer**: ser capaz de editar uma lista já criada previamente

**Para**: melhor organizar suas listas por nome

TC003 -

**Dado**: que Bruno tenha uma lista já criada

**Quando:** editar a lista

**Então:** a lista deve ser renomeada conforme desejado

**E:** que deve possuir nome único

**Contexto: RF004 – Excluir Lista**

US004 -

**Como**: quem faz as compras ocasionalmente pós serviço

**Bruno quer**: ser capaz apagara listas já criadas

**Para**: abrir espaço e melhor organizar seu app

TC004 -

**Dado**: que Bruno tenha uma lista já criada

**Quando:** excluir a lista

**Então:** a lista e seus produtos não devem mais ser visíveis

**Contexto: RF005 – Excluir Produto da Lista**

US005 -

**Como**: quem faz as compras ocasionalmente pós serviço

**Bruno quer**: ser capaz de apagar produtos de listas já criadas

**Para**: não efetuar compras desnecessárias

TC005 -

**Dado**: que Bruno tenha uma lista já criada

**Quando:** excluir um produto da lista

**Então:** este produto não deve mais ser visível

**E:** também não ser mostrada mais sua quantidade

**Contexto: RF006 - Adicionar produtos na lista**

US006 -

**Como**: responsável pelas compras de casa

**Nina quer**: ser capaz de adicionar produtos da base de dados e suas quantidades à lista

**Para**: não esquecer os itens ao ir ao mercado

TC006 -

**Dado**: que a Nina esteja na tela de uma lista

**Quando:** for adicionar produtos

**Então:** deve ser exibido uma base de produtos já criados selecionáveis

**E:** deve ser possível adicionar produtos

**E:** deve poder inserir a quantidade entre 1 e 99

**E:** não pode ser produto já existente na lista

**Contexto: RF007 - Editar produto na lista**

US007 -

**Como**: quem faz as compras ocasionalmente pós serviço

**Bruno quer**: ser capaz de editar os produtos da lista

**E:** definir a quantidade dos produtos da lista

**Para**: não efetuar compras desnecessárias

TC007 -

**Dado**: que Bruno tenha uma lista já criada com produto

**Quando:** clicar nela para editar

**Então:** deve ser possível trocar o item

**E:** também a quantidade do produto

**E:** fazer check

**Contexto: RF008 - Consultar listas**

US008 -

**Como**: responsável pelas compras de casa

**Nina quer**: ser capaz de consultar todas suas listas

**E:** pesquisa-las pelo nome

**Para**: maior agilidade ao consultar suas listas no mercado

TC008 -

**Dado**: que a Nina esteja na tela inicial

**Quando:** for visualizar suas listas

**Então:** devem ser exibidas todas já salvas

**E:** seja possível pesquisar listas por nome

**E:** seja possível pesquisar listas por produto contido na mesma

**Contexto: RF009 - Pesquisar produtos dentro de uma lista já criada**

US009 -

**Como**: quem faz as compras ocasionalmente pós serviço

**Bruno quer**: ser capaz de filtrar os produtos dentro de uma lista pelo nome

**Para**: que ao passar por um item no mercado possa verificar rapidamente se o mesmo está na lista evitando compras desnecessárias

TC009 -

**Dado**: que Bruno esteja consultando uma lista

**Quando:** filtrar pelo nome de um item

**Então:** deve ser possível visualizar somente este item

**E:** também a quantidade do mesmo

**Obs:** caso não exista o item, não deve ser exibido nenhum item

**Contexto: RF010 – Compartilhar/Importar Lista Criada**

US010 -

**Como**: quem faz as compras ocasionalmente pós serviço

**Bruno quer**: ser capaz de importar listas

**Para**: que sua esposa possa lhe mandar uma lista já pronta com o que falta na casa

TC010 -

**Dado**: que Bruno esteja na tela inicial do app

**E:** com o arquivo contendo a lista já baixado

**E:** selecionar o arquivo na tela de importar

**Quando:** selecionar a opção de importar

**Então:** deve ser possível visualizar a lista completa com itens e quantidades