

MOBILLS - (Cálculo de salário líquido)

Controle de versão / Histórico de alterações

Versão 1.0 - 10/12/2022
Definição do modelo do plano de testes
Autor: Bruna Borges

Introdução

Este documento é baseado no cronograma geral do projeto e apresenta o planejamento das atividades de testes da funcionalidade de cálculo de salário líquido do sistema Mobills, tem como objetivo: entregar ótimos resultados para o negócio, otimizando tempo e recursos disponíveis; fazer a análise e priorização dos conjuntos de testes e base para a sequência de trabalho a ser feito e consulta de ferramentas e processos a serem utilizados.

Tipos de testes

- Testes de contrato (verificação e validação, requisitos)
- Testes manuais funcionais (fluxo, smoketest e regressão)
- Testes de usabilidade e layout
- Testes automatizados
- Testes de API
- Testes de performance
- Testes de acessibilidade

Estratégia de testes

- Análise do contrato ou documentação dos critérios fornecidos pelo PO para validar se o que já foi desenvolvido está em conformidade com os requisitos da aplicação.
- Análise geral da aplicação para o levantamento das possibilidades de casos de teste
- Calcular a cobertura de testes
- Modelagem (Preparação, ambiente, massa de testes, ferramentas)
- Os requisitos a serem testados serão priorizados de acordo com as necessidades do negócio e interdependência entre cada funcionalidade
- Levantamento das possibilidades de casos de testes a serem realizados.
- Ao final de cada Sprint, deverão ser apresentados os relatórios com os casos de teste, a documentação das evidências e caso sejam encontrados bugs, será enviado um relatório ao Scrum Master para que ele direcione aos devs de forma que façam a correção.

Requisitos a testar

- CTs Apenas Manuais:
 - Validar máscara dos campos
 - Preenchimento dos campos com dados inválidos
 - Validação dos campos e botões da funcionalidade individualmente
 - Deve ser possível compartilhar o resultado do cálculo via redes sociais ou cópia do link.
 - Deve ser possível imprimir o resultado do cálculo.
 - Deve ser possível salvar o resultado do cálculo no formato .XLS
- CTs Automatizados:
 - Validar se o resultado do cálculo leva em consideração os calculos de INSS e IRPF adequadamente.
 - Validação será feita por amostragem de acordo com as regras de INSS e IRPF

Equipe e infraestrutura

- Equipe:
 - PO: William Rosendo
 - QA: Bruna Borges
 - Equipe de desenvolvimento
- Ferramentas:
 - X Mind para plano de testes e levantamento dos casos de teste
 - Excel e Word para relatórios e levantamento da massa de dados
 - IntelliJ ou VScode
 - Plugin Cucumber
 - Git e GitHub para armazenamento e versionamento da documentação e Scripts de teste
 - Navegador Google Chrome
- Documentação:
 - Contrato ou documentos com as regras de aceite da funcionalidade
 - Swagger com as informações das APIs
- Fornecedores Externos: Não há
- Equipamentos: Notebook
- Ambientes:
 - Computador pessoal
 - Ambiente de homologação

Cronograma

- Sprint1: (12/12 a 16/12):
 - 12/12:
 - Escrita dos casos de teste
 - Execução dos casos de teste
 - Relatório de Execução
 - 14/12: Automação dos casos de teste 1/2
 - 16/12: Review e Retro
- Sprint2: (19/12 a 23/12):
 - 19/12:
 - Automação dos casos de teste 2/2
 - Relatório de execução
 - 21/12: Envio da planilha com report de Bugs
 - 23/12: Review e Retro
- Backlog:
 - Testes de API
 - Validar que a arquitetura da API Rest está de acordo com as recomendações gerais do estilo de arquitetura Rest
 - Testes de performance
 - Funcionalidade cálculo de salário líquido
 - Testes de acessibilidade
 - APIs

Documentação complementar

- Documentação da funcionalidade: (A enviar pelo P.O)
- Tabela com a priorização dos requisitos: (A enviar pela equipe de Q.A)
- Casos de testes manuais: (A enviar pela equipe de Q.A)
- Execução dos casos de testes: (A enviar pela equipe de Q.A)
- Planilha com report de bugs: (A enviar pela equipe de Q.A)
- Link de acesso ao Swagger para realizar testes de API: (A enviar pela equipe de desenvolvimento)

Notas importantes:

- Testes devem ser executados nos breakpoints: 1280 e 320px
- Deverá ser utilizado Scenario Outline
- Deverá ser utilizado Cucumber
- Deverão ser realizados testes positivos e negativos
- Entregas deverão ser realizadas através do GitHub