**ÍNDICE**

**1** **Identificação 2**

***1.1*** ***Resumo do projeto 2***

***1.2*** ***Abreviações e glossário 2***

1.2.1 Abreviações 2

**2** **Organização 2**

***2.1*** ***Atividades e responsabilidades*** 3

**3** **Identificação de configuração 4**

***3.1*** ***Regras de identificação 4***

3.1.1 Regras de identificação 4

3.1.1.1 Número da versão de um item de configuração 4

3.1.2 Regras de identificação de documentos 4

3.1.2.1 Descrição dos identificadores do documento 5

3.1.2.2 Identificação interna 5

***3.2*** ***Identificação de referência de configuração 5***

***3.3*** ***Gerenciamento de configuração base 5***

**4** **Controle de configuração 5**

***4.1*** ***Gestão de mudança 5***

***4.2*** ***Gestão de interface 6***

***4.3*** ***Contabilidade do estado da configuração 9***

***4.4 Reviews 9***

***4.5 Manutenção do plano de gerenciamento de configuração 9***

1. **Identificação**

Este documento tem como objetivo documentar o projeto GIT Mobile, mostrando todo a estrutura que foi criada para que o projeto começasse.

* 1. ***Resumo do documento***

Este documento contém o plano de gerência de configuração do software GIT Mobile, onde em cada tópico é apresentado detalhadamente a estrutura do projeto durante seu desenvolvimento.Todo o projeto pode ser visto no GITHUB: <link do git>

* 1. ***Referências***
     1. **Referências do projeto**

|  |  |  |
| --- | --- | --- |
| # | Identificador | Título do documento |
| 1 | ID01 | GITHUB mobile |

1. **Organização**

O projeto foi divido em várias etapas, como a contextualização, a execução e finalização. Cada integrante tem uma função dentro do projeto, sendo elas: Documentação; Prototipação; Codificação; e Testes.

Primeiramente foi decidido o que o projeto iria ser, ao entramos em consenso que seria feito um git mobile levantamos os requisitos e foram feitos protótipos para que os desenvolvedores tivessem uma direção para seguir. E ao longo de todas as etapas eram feitas alterações no documento para acompanhar o desenvolvimento do projeto.

* 1. ***Atividades e responsabilidades***

Para gerenciar o projeto **Git mobile** contamos com o apoio das redes de comunicação e gerência Github, Trello, WhatsApp. Onde gerenciamos os commits e andamento pelo próprio GitHub, trocamos informações, dúvidas e saberes pelo Trello e WhatsApp.

Na divisão de tarefas e responsabilidades a equipe se dividiu em programadores /desenvolvedores, equipe de prototipação e documentação.

É primordial que em todos os projetos de software haja a Gerência de configuração de software implementada de forma clara e assertiva de modo com que traga todos os benefícios com ela atrelados. Para isso devemos ter uma equipe com boa comunicação e eficiente.

|  |  |
| --- | --- |
| **Atividades de planejamento do projeto** | **Pessoa responsável** |
| Identificar os itens de configuração | GC |
| Instalar a ferramenta de repositório de bug e configurar o banco de dados | GC |
| Instalar a ferramenta de repositório de configuração do software e configurar o banco de dados | GC |
| Gerir e estruturar o espaço de referência | GC |
| Definir os processos de configuração | GC |

|  |  |
| --- | --- |
| **Atividades durante o ciclo de vida do projeto** | **Pessoa responsável** |
| Exportar componentes para modificação, teste ou entrega | Dev |
| Definir o controle de componentes validados | GC |
| Criar versão e entregar documento de visão | GC |
| Aprovar configurações de referência | Tester |
| Verificar a versão a ser entregue e autorizar entregas | Tester |
| Espaço para backup | Dev |
| Fazer auditorias de configuração | Tester |
| Inspecionar os registros de configuração | Tester |
| Arquivo de referência da versão | GC |

|  |  |
| --- | --- |
| **Gerir atividades** | **Pessoa responsável** |
| Gerenciar versões e arquivos | GC |
| Gerenciar registros de configuração | GC |
| Produzir relatórios e estatísticas | Dev |
| Gerenciar o espaço de referência e sua lista de controle de acesso | GC |
| Gerenciar espaços de backup e mídia | GC |
| Gerenciar relatórios de qualidade | Tester |

1. **Identificação de configurações**

Controle das configurações do sistema, versões e componentes.

* 1. ***Regras de identificação***

Para um item ser identificado como um item configurado ele deverá fazer parte da estrutura do sistema de software, como componentes individuais únicos que possam ser rastreados, a fim de controlar mudanças de configuração, estabelecer mecanismos que irão ajudar a produzir software de qualidade, garantir que cada versão de software contenha elementos necessários e que tais elementos irão trabalhar corretamente juntos.

* + 1. **Regras de identificação de itens de configuração**

Para a padronização da identificação de itens configuráveis será usado o padrão ***XXX\_Y.z***, onde ***XXX*** será o nome do item ,***Y*** será o incremento da versão maior e o ***z***  incremento da versão menor. Exemplo prototipo\_1.0.

* + - 1. ***Número da versão de um item de configuração***

Considerando a regra de identificação de itens configuráveis XXX\_Y.z vamos considerar:

XXX: deverá ser um nome simples para que seja fácil identificar qual item se refere com a inicial minúscula e se nome composto letra maiúscula para o início da segunda palavra (ex: especificaçõesRequisitos).

Y: será direcionado há lterações incompatíveis com as versões anteriores da API incrementa a versão Maior, será representados por números.

z: Adições/alterações compatíveis com as versões anteriores da API incrementa a versão Menor, será representados por números.

* + 1. **Regras de identificação de documentos**
       1. ***Descrição dos identificadores do documento***

A identificação dos documentos está descrita abaixo: XXX\_ <tipo de documento> \_ <número do documento> \_ <índice de revisão>

Onde: “tipo de documento" é: Foo para documentos FOO, BAR para documentos de barra.

"número do documento" é um número incremental, com uma lista separada para cada tipo de documento, "índice de revisão" designa a iteração aprovada do documento. O índice de revisão é V1 para a primeira iteração, V2 para a segunda e assim por diante.

* + - 1. ***Identificação interna***

A identificação de uma mídia é descrita abaixo:

<identificação do item de configuração> / <media> / <volume>

Onde:

"mídia" é o número da mídia,

"volume" é um número incremental para distinguir a mídia se a entrega contiver mais de uma mídia.

* 1. ***Identificação de referência de configuração***

Cada configuração de referência é definida por:

Um identificador,

Seu conteúdo listado no documento Descrição da Entrega da Versão correspondente,

As revisões de aceitação ou validação associadas à construção da configuração de referência.

Uma configuração de referência é estabelecida para cada revisão de design e cada revisão de teste do projeto

* 1. ***Gerenciamento de configuração base***

<a ser feito>

* F01 – Login
  + Permite que o usuário use o aplicativo através de seu login da plataforma git hub e necessário possuir uma conta nessa plataforma para usar os recursos do aplicativo. No login o usuário deverá informar seu e-mail e sua senha do git hub que serão usados para obter dados para uso no aplicavo.
* F02 – Tela inicial
  + Na tela inicial do aplicativo e listado os trends do git hub que são os repositórios mais acessados e favoritados do git hub. Permitindo o usuário filtrar pelos repositórios disponíveis
* F03 – Configurações
  + Uma tela que apresenta que apresenta um pouco sobre os dados da aplicação e seria uma espécie de sobre do aplicativo, porém está incompleta.
* F04 – Menu do aplicativo
  + Disponível através da tela inicial ou outras telas mostra um menu com as funcionalidades disponíveis do sistema e que ao pressionar em algum menu o usuário será levado até ele.
* F05 – Perfil do usuário
  + Mostra os dados do perfil do usuário no git hub como o número de seguidores, de pessoas que o usuário segue, repositórios e etc.

Foi definido dois grupos de configuração: A linha de base funcional (FBL) e; a linha de base alocada (ABL).

A FBL contém a descrição das funcionalidades do sistema e suas características:

Describe what baselines are to be established. Explain when and how they will be defined and controlled.

Examples of baselines :

* functional baseline (FBL), which describes the system functional characteristics;
* allocated baseline (ABL), which describes the design of the functional and interface characteristics,
* product baseline (PBL), which consists of completed and accepted system components and documentation that identifies these products.

1. **Controle de configuração**

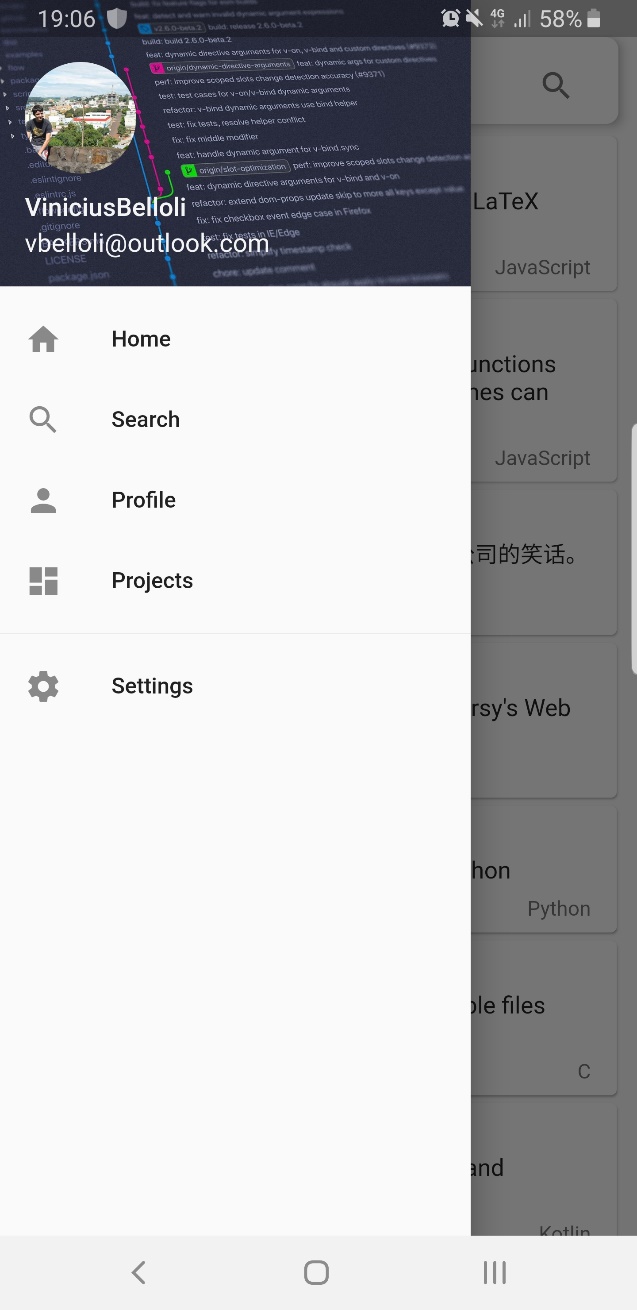
Utilizando a plataforma GITHUB, é possível gerenciar todas as versões de configurações que o projeto tem, sendo possível analisar e discutir as mudanças que ocorreram e o porquê de acontecerem.

* 1. ***Gerir mudanças***

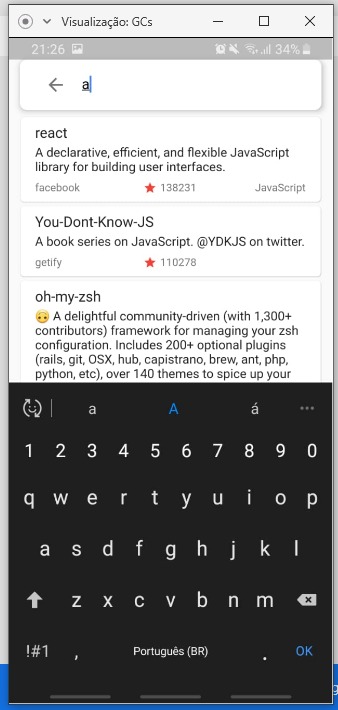
Todas as mudanças que irão ocorrer no projeto devem passar pelo *Tester* e depois pelo Gerente de configuração(GC). Após a aprovação dos dois, o projeto pode sofrer as mudanças.

* 1. ***Gestão de interface***

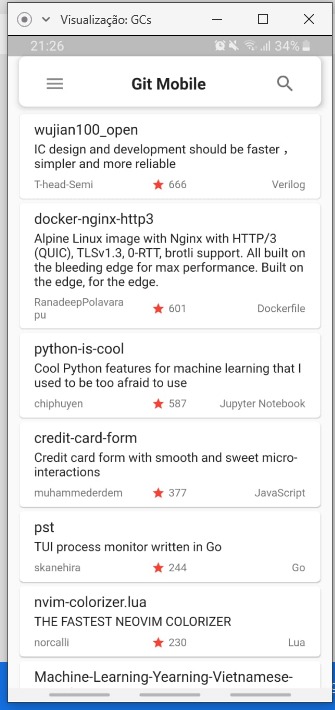
Nessa seção é documentado todos os protótipos de telas que foram feitas e os resultados finais.



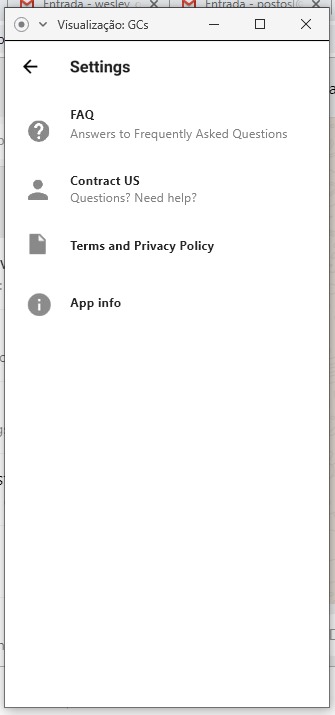
Protótipo 1: Menu inicial



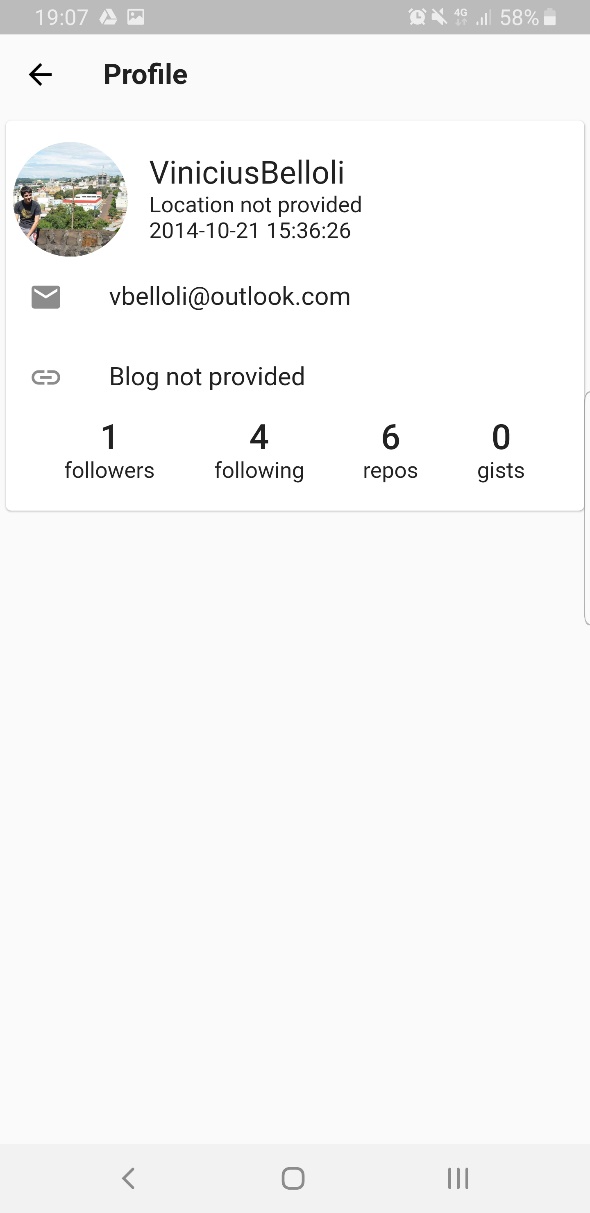
Protótipo 2: Listagem de repositórios



Protótipo 3: Listagem dos repositórios



Protótipo 4: Configurações

****

Protótipo 5: Perfil do usuário

* 1. ***Contabilidade do estado da configuração***

Configuration Status Accounting (CSA) is the process to record, store, maintain and report the status of configuration items during the software lifecycle. All software and related documentation should be tracked throughout the software life.

* 1. ***Reviews***

O gerente de configuração é o responsável pelo processo de Gerência de Configuração do projeto, disciplina que define o conjunto de atividades necessárias para a gestão de baselines, itens de configuração e requisições de mudança do projeto. No nosso caso o professor Rafael Oliveira nos orientou, com feedbacks e sedendo tempo em sala de aula para o desenvolvimento do projeto.

* 1. ***Manutenção do plano de gerenciamento de configuração***

O plano de gerenciamento de configuração sempre é atualizado conforme novos artefatos são criados, sendo primeiro discutido com a equipe as mudanças e então é passado para o documento. A cada semana, o documento passa por uma revisão, levando em conta o que pode ser melhorado e o que de novo foi criado.