Remember Why

Plano de ação e desenvolvimento de software

# Planejamento

## Desafio

Desenvolver um sistema em formato de CRUD (Create, Read, Update e Delete) utilizando uma dessas linguagens de programação: C, C++, Java, PHP ou Go.

Seu CRUD deve funcionar como um gerenciador de senhas, no qual o usuário vai utilizar para centralizar e armazenar todas as senhas que ele quiser. Salve essas informações em um banco de dados também de sua preferência e desenvolva uma interface para facilitar a experiência do usuário. Ao final do processo, envie seu código para o Github e compartilhe o link no desafio da plataforma.

## Ambiente de desenvolvimento integrado

A IDE usada no desenvolvimento do sistema Remember Why será a Netbeans, pois permite um desenvolvimento rápido de aplicações desktop, realçando sintática e semanticamente o código-fonte do sistema facilitando a programação das atividades.

## Linguagem de programação

A linguagem de programação utilizada no código-fonte do sistema é o JAVA SE, foi escolhida devido à familiaridade com o desenvolvedor e por fornecer recursos robustos para o desenvolvimento do sistema.

## Interface Gráfica

A API de interface gráfica escolhida é a Swing, pois já vem com um Look-and-Feel (suas cores, formatos e etc.) de forma agradável ao usuário e suportadas por outras plataformas. A GUI Builder realiza automaticamente o espaçamento e o alinhamento dos contêineres e controles permitindo um visual mais harmonioso.

## Banco de dados

O banco de dados escolhido é o Java DB ou conhecido também como Apache Derby, ele é integrado a IDE Netbeans e permite criar e gerenciar o banco utilizando interface gráfica, utiliza o driver de JDBC que chama o banco de dados Derby local.

## Criptografia

A criptografia utilizada é a Base64, ela é usada comumente para criptografar dados binários e os traduz para uma representação Base64, na aplicação ela é usada desde o momento em que o usuário cadastra sua senha, então os dados são criptografados e armazenados no banco de dados, no momento que os dados precisam ser visualizados de forma centralizada é feita a descriptografia para exibir os dados originais para o usuário.

# Execução

# Visão do Projeto

## Propósito do Projeto, Escopo e Objetivos

Desenvolvimento de um sistema de gerenciamento de senhas onde o usuário pode guardar todas as senhas que ele quiser.

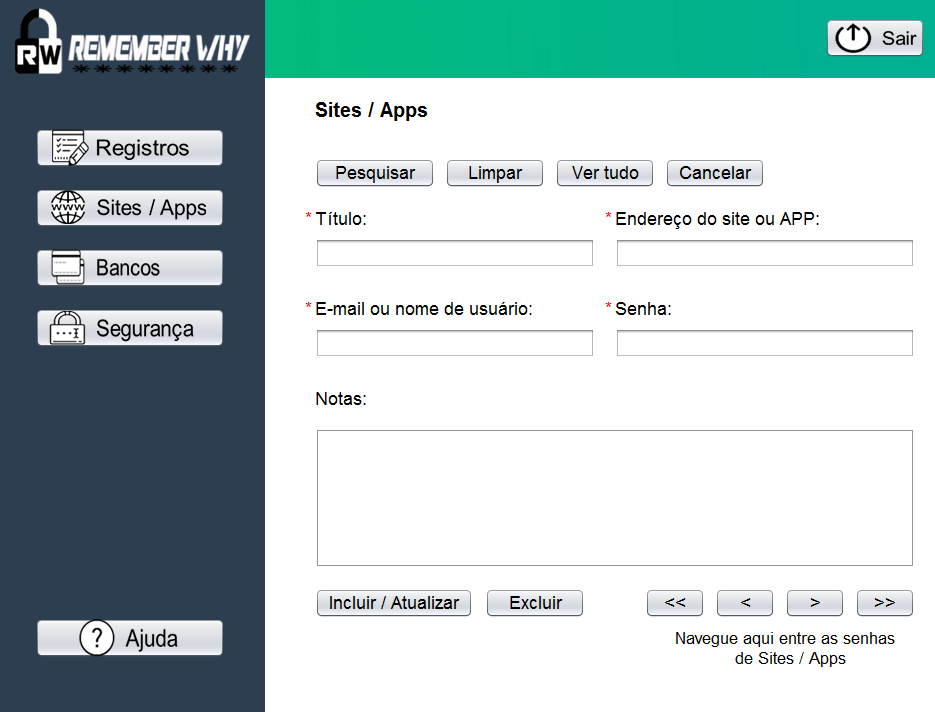
**Cadastro de usuário e Login no sistema**

É possível cadastrar um novo usuário no banco de dados e posteriormente acessar o sistema por meio da tela de Login.



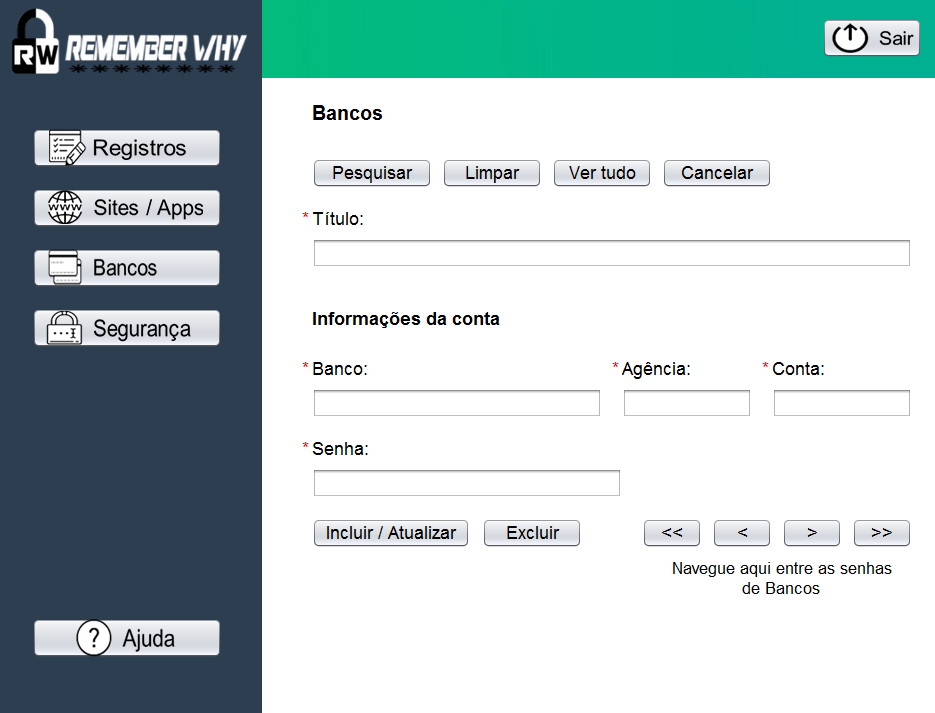
**Cadastro de senhas de Sites e Aplicativos**

O usuário pode cadastrar uma nova senha de sites e aplicativos no sistema, como o título, endereço do site ou App, E-mail ou nome de usuário, senha, notas, alterar os dados cadastrados, pesquisar um registro de senha existente, navegar entre os registros e deletar qualquer informação quando desejar.



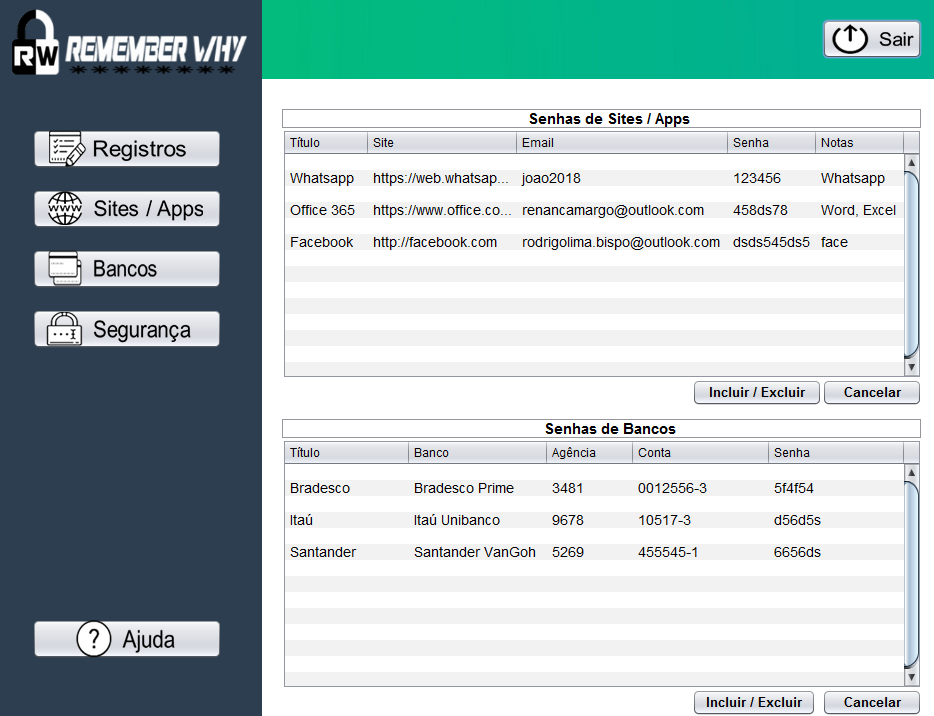
**Cadastro de senhas de Bancos**

O usuário pode cadastrar uma nova senha de bancos no sistema, como o título, banco, agência, conta, senha, alterar os dados cadastrados.



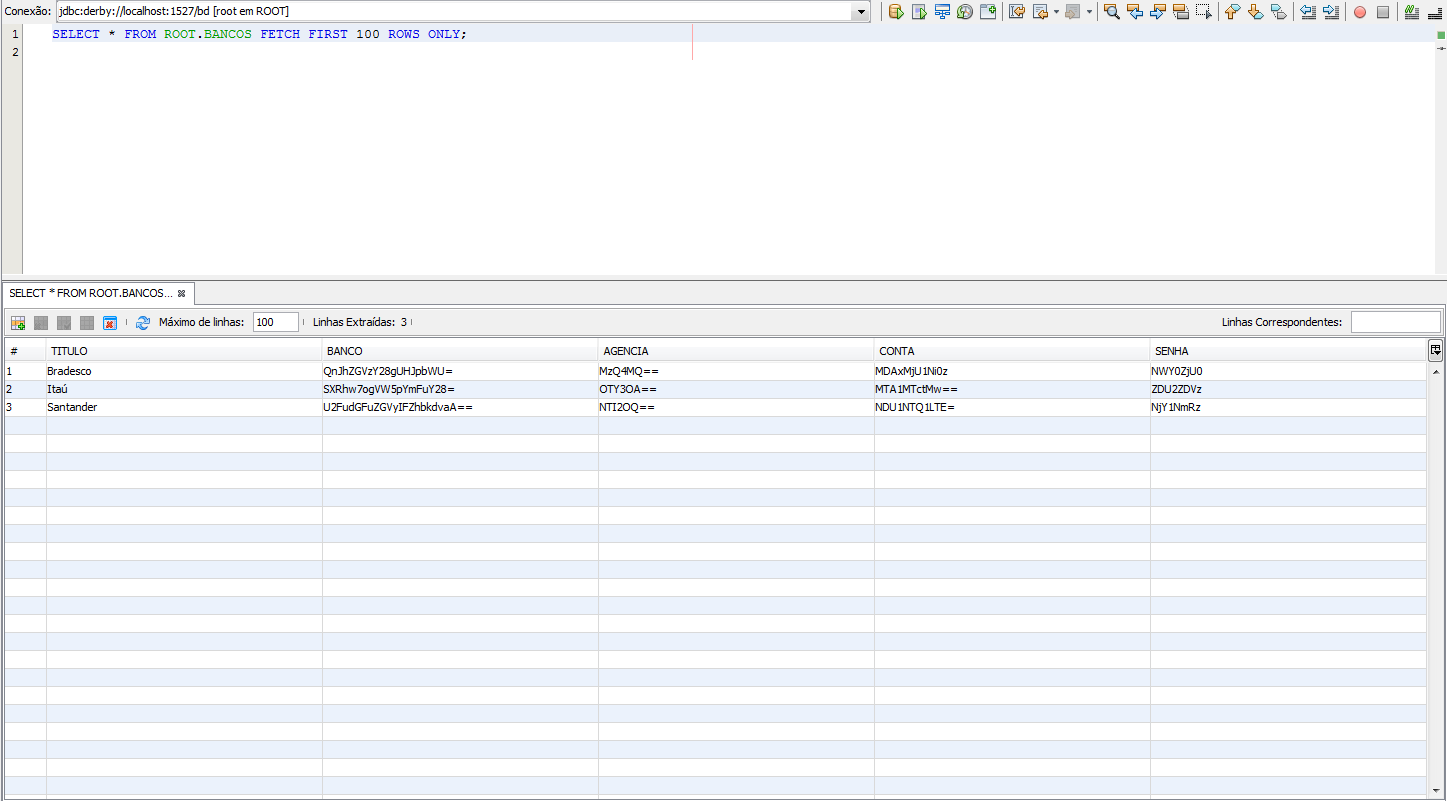
**Visualização de todas as senhas cadastradas**

Por meio da função Registros o usuário pode visualizar duas tabelas com as senhas de sites e aplicativos e outra com as senhas de bancos de forma centralizada.



**Criptografia das senhas cadastradas**

Por meio da criptografia Base64 as senhas codificadas são representadas por uma sequencia de bytes ASCII, essa combinação deixa os dados improváveis de serem modificados no processo de comunicação entre o sistema e banco de dados.



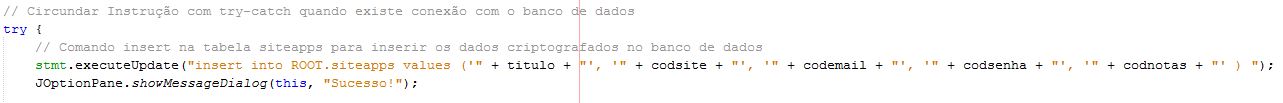
**Recurso de Ajuda**

A função Ajuda orienta o usuário sobre a utilização do sistema, facilitando no entendimento da interface gráfica e recursos.

## 

## CRUD (Inserir, alterar, consultar e deletar)

## Inserir:



A inserção do conteúdo digitado pelo usuário no banco de dados é feito o Statement com o comando executeUpdate, o comando INSERT é utilizado para armazenar o conteúdo das variáveis recebidas pelo usuário no banco de dados.

## Alterar:



Com a instrução try-catch se o conteúdo já existir na tabela ela será alterado no banco de dados pelo comando UPDATE, isso evita que sejam gravados dados duplicados na tabela.

## Consultar:



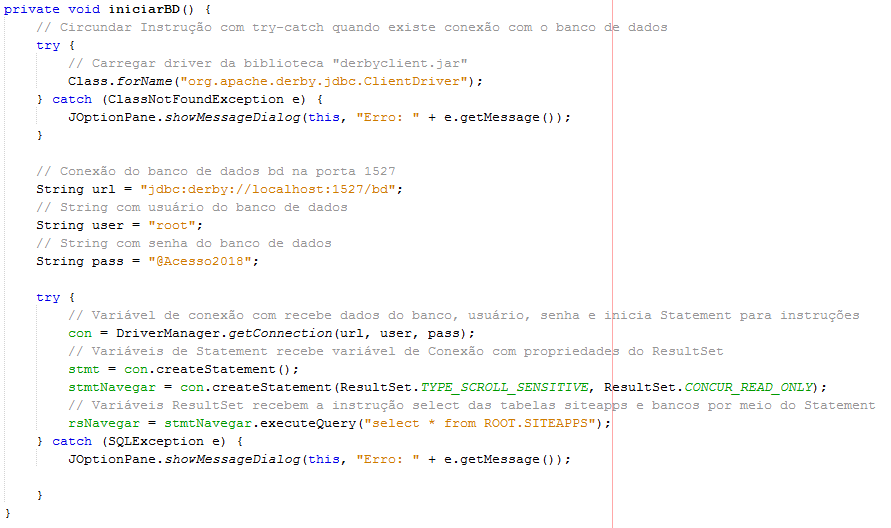
Por meio do ResultSet que armazena conteúdo do banco de dados o comando SELECT busca o título cadastrado no registro da senha no banco de dados para retornar a informação especifíca que o usuário procura.

## Deletar:

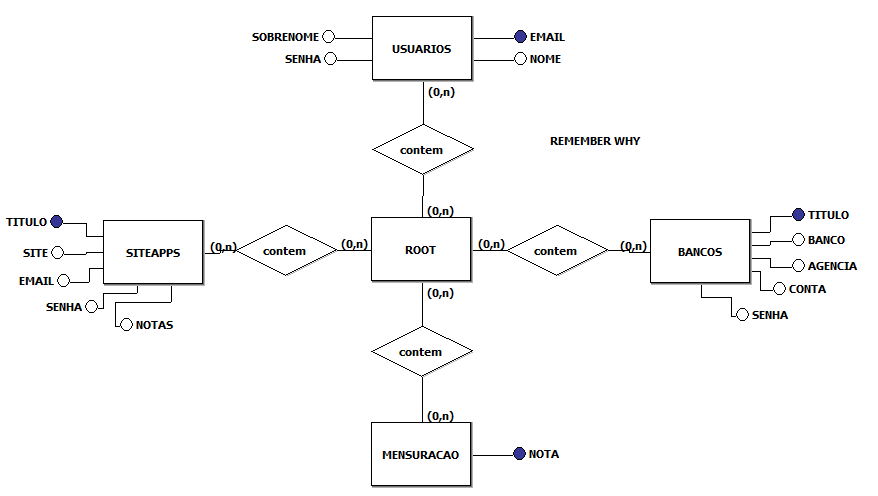


Após encontrar o título correspondente no banco de dados ele pode ser deletado por meio do comando DELETE, excluir um registro com precisão garante a integridade das outras senhas cadastradas no sistema.

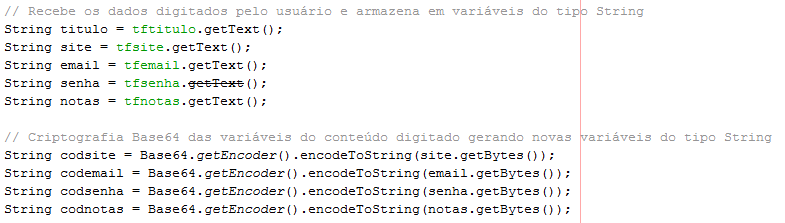
## Banco de dados Java DB / Apache Derby



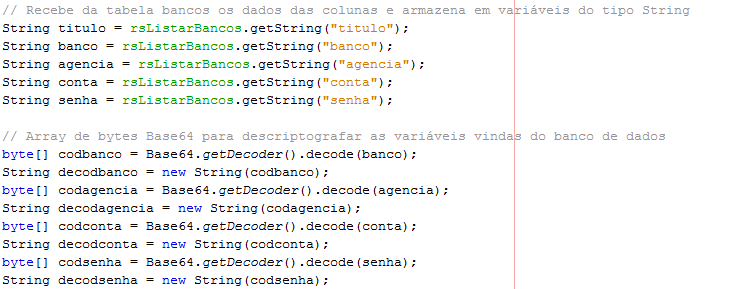
Para iniciar a conexão com o banco de dados foi necessário configurar o driver, porta de comunicação 1527, usuário e senha, além das variáveis con de conexão, o statement stmt e o ResultSet.



## Biblioteca Base64

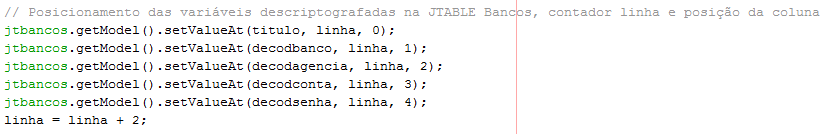


Ao receber o conteúdo digitado pelos usuários nas variaveis tipo String é utilizado o Encoder da biblioteca Base64 de criptografia, assim todas as Strings armazenadas no banco de dados passam pelo Encoder para garantir que não serão alteradas no processo.



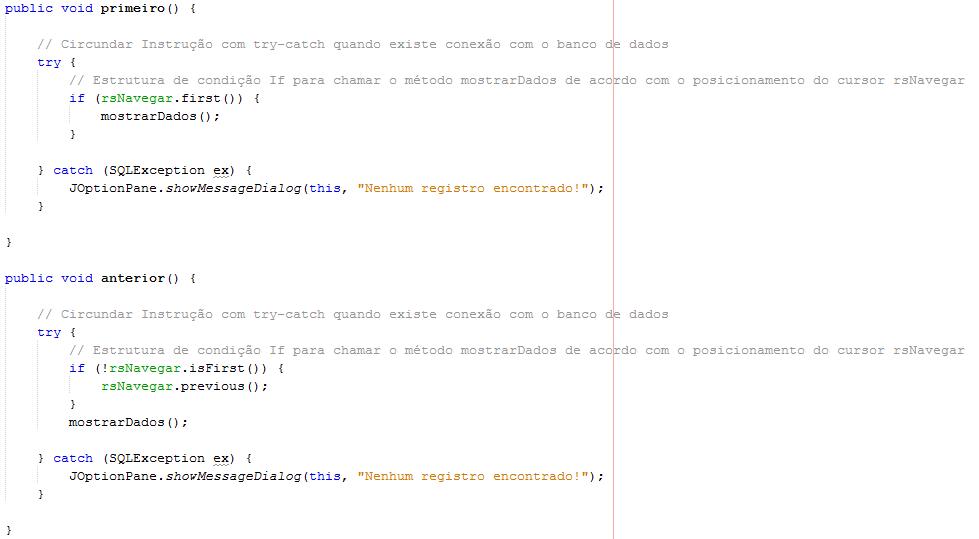
Quanto os dados são importados para as tabelas (Jtable) o conteúdo é recebido nas variáveis do tipo String e então são submetidas ao Decoder que decodifica com o Array de Bytes e guarda as informações originais em uma nova variável para ser exibida nas tabelas dos registros.

## Visualização de todas as senhas cadastradas

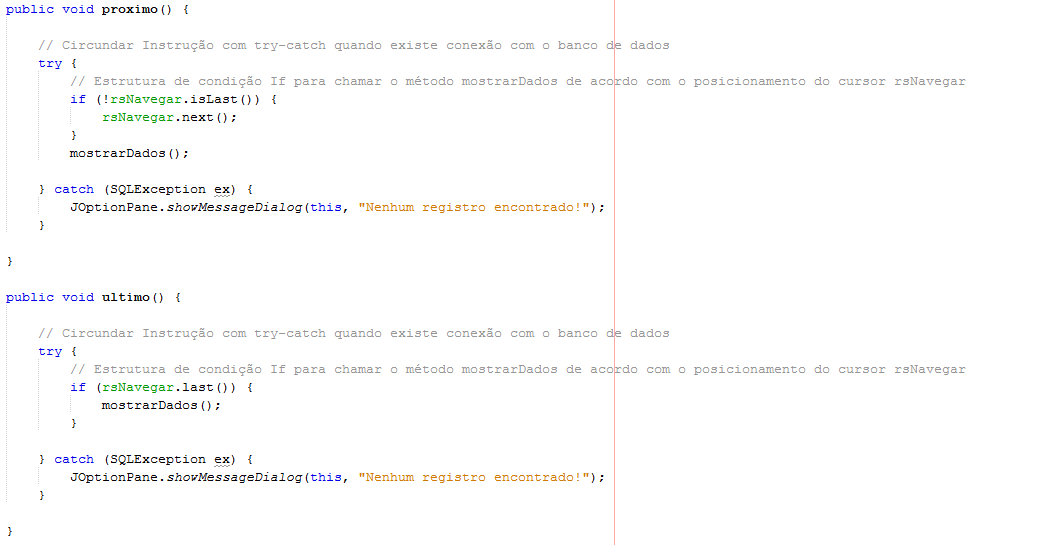


Para centralizar todas as senhas cadastradas a Jtable é criada e o conteúdo das variáveis descriptografas são alocadas, o com contador responsável por pular a linha dos registros é configurado no getModel e também o ajuste da coluna.

## Navegação entre os registros



Os botões de navegação servem para facilitar o gerenciamento das senhas cadastradas pelo usuário, o botão primeiro é definido pela condição If, ele busca o primeiro registro encontrado no ResultSet, já o botão anterior traz o registro se não for o primeiro encontrado e volta para o registro anterior.

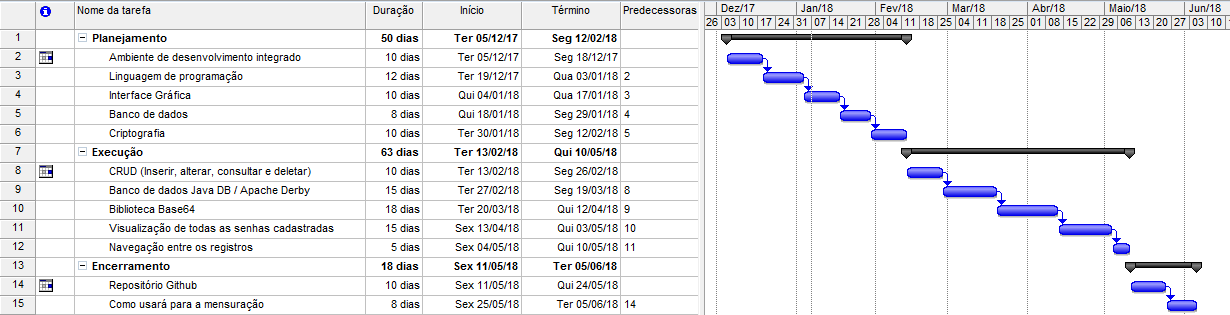


O botão próximo posiciona no próximo registro se o conteúdo encontrado no banco de dados não for o último, já o botão último tenta posicionar o cursor no último registro encontrado.

# Encerramento

## Plano de Ação 6 meses (Planejamento, Execução e Encerramento)

Gráfico de Gantt



O planejamento, execução e encerramento do projeto foram organizados em 6 meses e cada tarefa possui seu tempo de ínicio e término, de acordo com o gráfico nenhuma tarefa começa sem a anterior terminar.

## Repositório Github

O Github será utilizado para hospedar o projeto e posteriormente enviado o link para a avaliação do desafio ACESSO 2018, após limpar e construir os arquivos do projeto como o build.xml, manifest.mf e as pastas build, dist, nbproject e src serão enviadas para o repositório.

## Qual o resultado esperado para que o projeto possa ser considerado um sucesso?

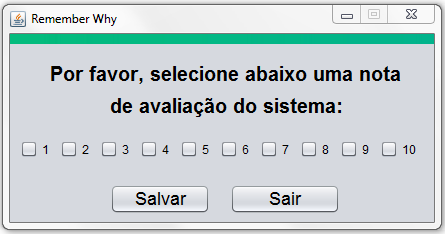
Para que o projeto possa ser considerado um sucesso é necessário que ele seja aprovado pela avaliação da ACESSO no desafio Tech, mas caso fosse disponibilizado para os usuários é esperado que o sistema passe a fazer parte da rotina das pessoas, que essa ferramenta possa ser a base para armazenar as senhas de outros sistemas, isso só é possível pela simplicidade e segurança.

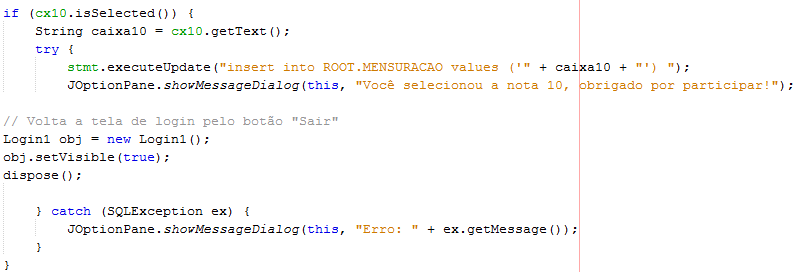
## Explique qual é a sua expectativa de impacto?

Minha expectiva é que os usuários encontrem no sistema uma forma simples e rápida de gerenciar todas as suas senhas com segurança, que a boa usabilidade possa conquistar novos usuários.

## Como usará para a mensuração?

A mensuração será feita sempre que o usuário clicar em sair do sistema, uma janela será aberta perguntando a nota de 0 a 10 para o sistema, as informações são armazenadas no banco de dados.





## Imagens do projeto

Todas as imagens externas como os ícones usados no sistema foram retiradas do site <https://icons8.com.br> que fornece icones gratuitos para download, outras imagens foram criadas com recursos próprios do photoshop.