**CURSO TECNOLOGIA EM ANÁLISE E DESENVOLVIMENTO DE SISTEMAS**

**TURNO DIURNO 1º SEMESTRE 2017**

**ENGENHARIA DE SOFTWARE II v1**

**GRUPO 4 – CONTAS A PAGAR**

BRUNO HENRIQUE M. FRANÇA 0030481611039

FELIPE DA SILVA LIBERAL 0030481521010

GIOVANE DE LUCAS HADDAD 0030481611018

VANESSA OLIVEIRA VIEIRA 0030481511039

VICTOR COSTA 0030481611034

VINICIUS EIDI HONDA 0030481611036

**SUMÁRIO**

[1 DEFINIÇÃO DE TELAS 2](#_Toc483797354)

[1.1 Login 2](#_Toc483797355)

[1.2 Confirmar senha de administrador 5](#_Toc483797356)

[1.3 Cadastrar usuário 7](#_Toc483797357)

[1.4 Editar usuário 10](#_Toc483797358)

[1.5 Cadastrar contas bancárias 13](#_Toc483797359)

[1.6 Editar contas bancárias 16](#_Toc483797360)

[1.7 Efetuar cadastro rápido de fornecedores 18](#_Toc483797361)

[1.8 Lançar contas a pagar 20](#_Toc483797362)

[1.9 Editar contas a pagar 24](#_Toc483797363)

[1.10 Pagar conta 28](#_Toc483797364)

[1.11 Pagar contas em lote 32](#_Toc483797365)

[1.12 Dar baixa pelo arquivo retorno 36](#_Toc483797366)

[1.13 Dar baixa pela tesouraria 39](#_Toc483797367)

[1.14 Simular contas a pagar 42](#_Toc483797368)

[1.15 Importar extrato bancário 45](#_Toc483797369)

[1.16 Carregar extrato bancário 47](#_Toc483797370)

[1.17 Conciliar extrato bancário 49](#_Toc483797371)

# DEFINIÇÃO DE TELAS

## Login

(Parte feita por: Bruno)

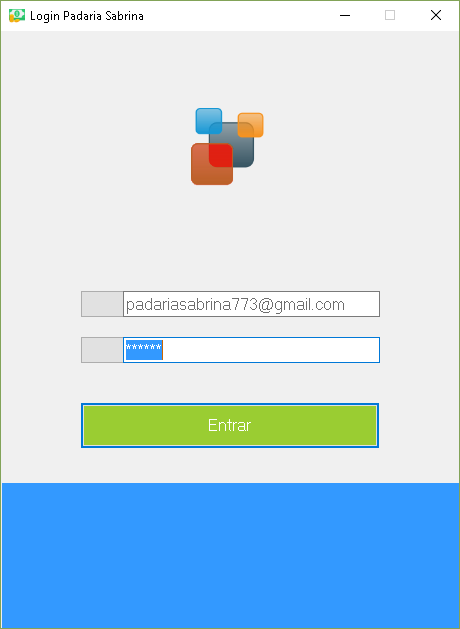
**Objetivo:** Este caso de uso permite fazer login no sistema.

**Enunciado:**

**Descrição da tela:** O botão da tela terá um evento associado. No centro terá dois textbox, um para digitar o e-mail e o outro para digitar a senha para fazer login no sistema, e em seguida terá o botão para entrar no sistema.

**Eventos:** No clique do botão Entrar execute o método ObterUsuario da classe Usuario, confira se o e-mail e a senha digitado confere com algum usuário que tenha no sistema. Caso confira ver o nível de acesso e permitir o login. Se digitado errado dar um aviso de e-mail ou senha incorreta.

**Layout**



**Classes**



## Confirmar senha de administrador

(Parte feita por: Felipe)

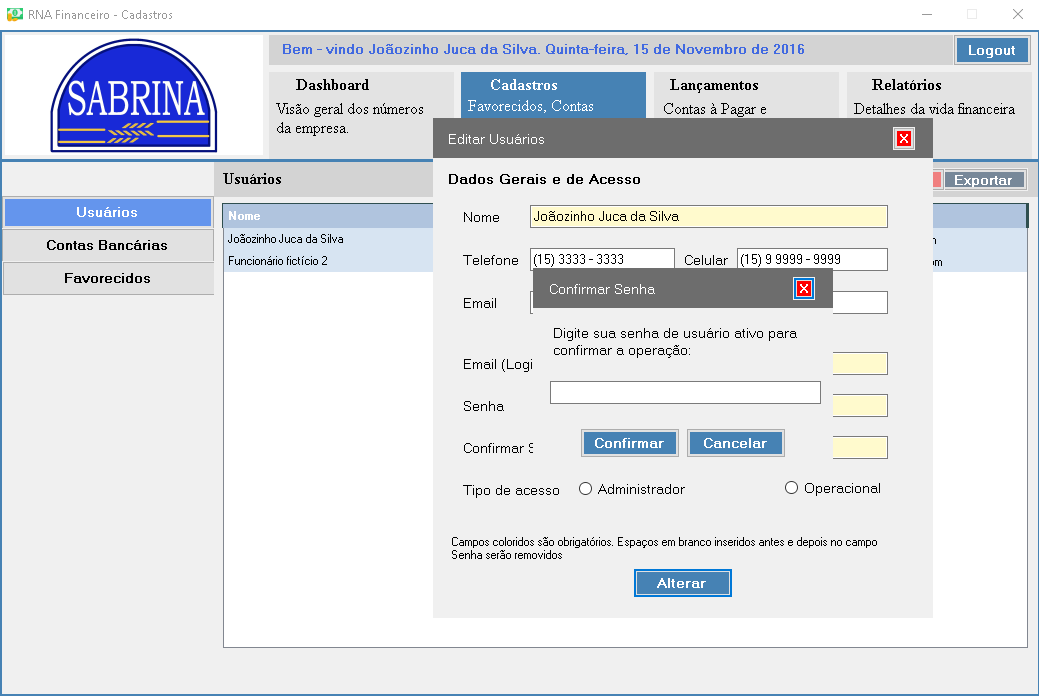
**Objetivo:** Este caso de uso permite confirmar senha no sistema.

**Enunciado:**

**Descrição da tela:** Cada um dos dois botões da parte inferior terá um evento associado. Há um único campo textbox password para confirmar senha, já cadastrada, para alteração.

**Eventos:** Compare o campo Senha da classe Usuario, com o que foi digitado no campo obrigatório de máscara password, se este tem no mínimo 8 caracteres. No clique do botão Confirmar alteração, execute o método EditarUsuario da classe Usuario, confirme alteração e acesse Editar Usuários. No clique do botão Cancelar, cancele alteração.

**Layout**

****

**Classes**



CadastrarUsuario: recebe o código do usuário, e verifica se o mesmo já está cadastrado.

EditarUsuario: recebe o código do usuário cadastrado, e exibe seus dados para serem editados.

RemoverUsuario: recebe o código do usuário cadastrado, e dá a opção de exclusão do mesmo.

ObterUsuarios: recebe o código do usuário cadastrado, e os lista na tela.

## Cadastrar usuário

(Parte feita por: Felipe)

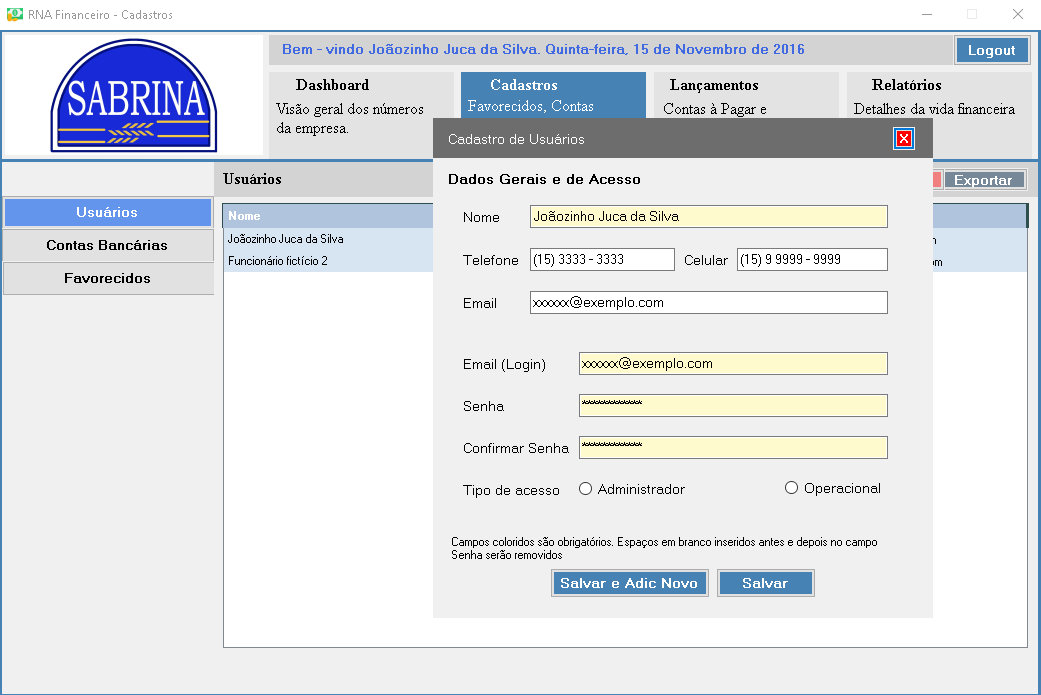
**Objetivo:** Este caso de uso permite cadastrar usuários no sistema.

**Enunciado:**

**Descrição da tela:** Cada um dos dois botões da parte inferior terá um evento associado. Do lado esquerdo, o usuário terá os textbox nome, telefone, celular, email e os radio buttons do tipo de acesso (administrador ou operacional) cadastrados. Do lado direito, o usuário terá o textbox login, textboxes password senha e confirmar senha, para serem preenchidos e cadastrados. O preenchimento é obrigatório para todos os campos coloridos.

**Eventos:** No clique do botão Salvar e Adic., execute o método CadastrarUsuario da classe Usuario e cadastre os dados dos campos Nome (compare o campo Nome da classe Usuario, com o que foi digitado no campo obrigatório de nome), Telefone (compare o campo Telefone da classe Usuario, com o que foi digitado no campo de máscara numérica Telefone, se há 10 números, contando com o DDD e se já existe), Celular (compare o campo Celular da classe Usuario, com o que foi digitado no campo de máscara numérica Celular, se há 11 números, contando com o DDD e se já existe), Email (compare o campo Email da classe Usuario, com o que foi digitado no campo de E-mail, verifica se antes e depois do arroba existem caracteres digitados, e se já existe), Login (compare o campo Email da classe Usuario, com o que foi digitado no campo obrigatório de Email Login, verifique se antes e depois do arroba existem caracteres digitados, e se já existe), Senha (compare o campo Senha da classe Usuario, com o que foi digitado no campo obrigatório de máscara password Senha, se está digitado com o mínimo 8 caracteres), Confirmar Senha (compare o campo Senha da classe Usuario, com o que foi digitado no campo obrigatório de máscara password Confirmar Senha, se está digitado com o mínimo 8 caracteres e igual ao campo Senha) e o tipo de acesso (Administrador ou Operacional, compare o campo Acesso da classe Usuario, com o que foi selecionado nos campos radio buttons); habilite o botão Salvar.

**Layout**

****

**Classes**



CadastrarUsuario: recebe o código do usuário, e verifica se o mesmo já está cadastrado.

EditarUsuario: recebe o código do usuário cadastrado, e exibe seus dados para serem editados.

RemoverUsuario: recebe o código do usuário cadastrado, e dá a opção de exclusão do mesmo.

ObterUsuarios: recebe o código do usuário cadastrado, e os lista na tela.

## Editar usuário

(Parte feita por: Felipe)

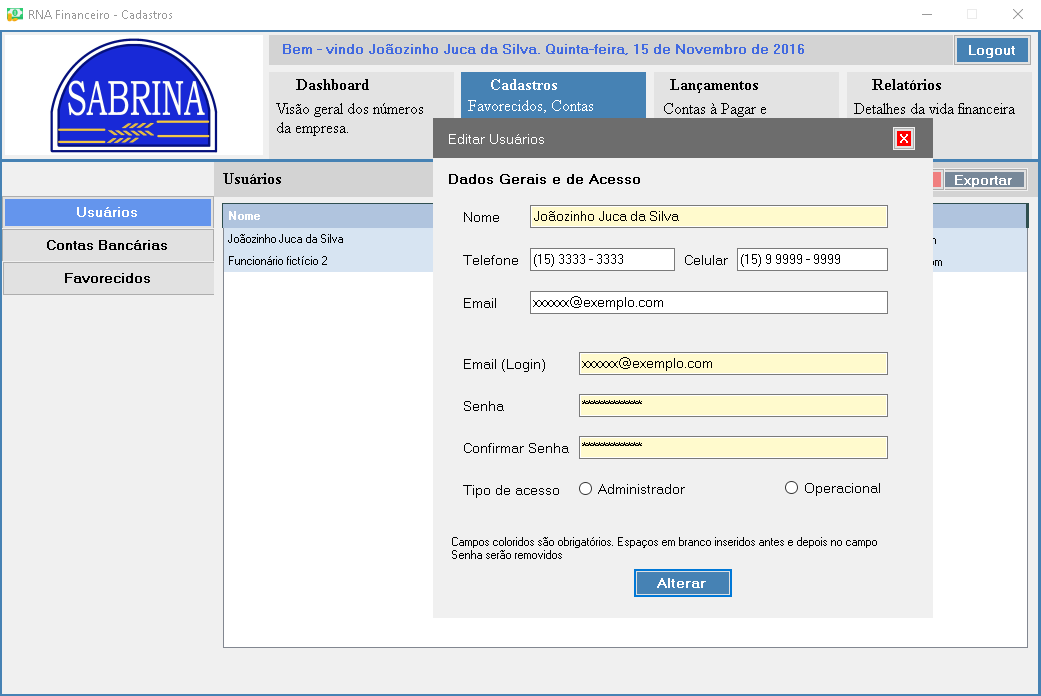
**Objetivo:** Este caso de uso permite editar usuários no sistema.

**Enunciado:**

**Descrição da tela:** Cada um dos dois botões da parte inferior terá um evento associado. Do lado esquerdo, o usuário terá os textbox nome, telefone, celular, email e os radio buttons do tipo de acesso (administrador ou operacional) cadastrados. Do lado direito, o usuário terá o textbox login, textboxes password senha e confirmar senha, para serem editados. São idênticos ao Cadastro de Usuários. O preenchimento é obrigatório para todos os campos coloridos.

**Eventos:** No clique do botão Alterar., execute o método EditarUsuario da classe Usuario e edite os dados dos campos Nome (compare o campo Nome da classe Usuario, com o que foi digitado no campo obrigatório de nome), Telefone (compare o campo Telefone da classe Usuario, com o que foi digitado no campo de máscara numérica Telefone, se há 10 números, contando com o DDD e se já existe), Celular (compare o campo Celular da classe Usuario, com o que foi digitado no campo de máscara numérica Celular, se há 11 números, contando com o DDD e se já existe), Email (compare o campo Email da classe Usuario, com o que foi digitado no campo de E-mail, verifica se antes e depois do arroba existem caracteres digitados, e se já existe), Login (compare o campo Email da classe Usuario, com o que foi digitado no campo obrigatório de Email Login, verifique se antes e depois do arroba existem caracteres digitados, e se já existe), Senha (compare o campo Senha da classe Usuario, com o que foi digitado no campo obrigatório de máscara password Senha, se está digitado com o mínimo 8 caracteres), Confirmar Senha (compare o campo Senha da classe Usuario, com o que foi digitado no campo obrigatório de máscara password Confirmar Senha, se está digitado com o mínimo 8 caracteres e igual ao campo Senha) e o tipo de acesso (Administrador ou Operacional, compare o campo Acesso da classe Usuario, com o que foi selecionado nos campos radio buttons); receba os dados já preenchidos e habilite o botão Alterar.

**Layout**

****

**Classes**



CadastrarUsuario: recebe o código do usuário, e verifica se o mesmo já está cadastrado.

EditarUsuario: recebe o código do usuário cadastrado, e exibe seus dados para serem editados.

RemoverUsuario: recebe o código do usuário cadastrado, e dá a opção de exclusão do mesmo.

ObterUsuarios: recebe o código do usuário cadastrado, e os lista na tela.

## Cadastrar contas bancárias

(Parte feita por: Vinícius)

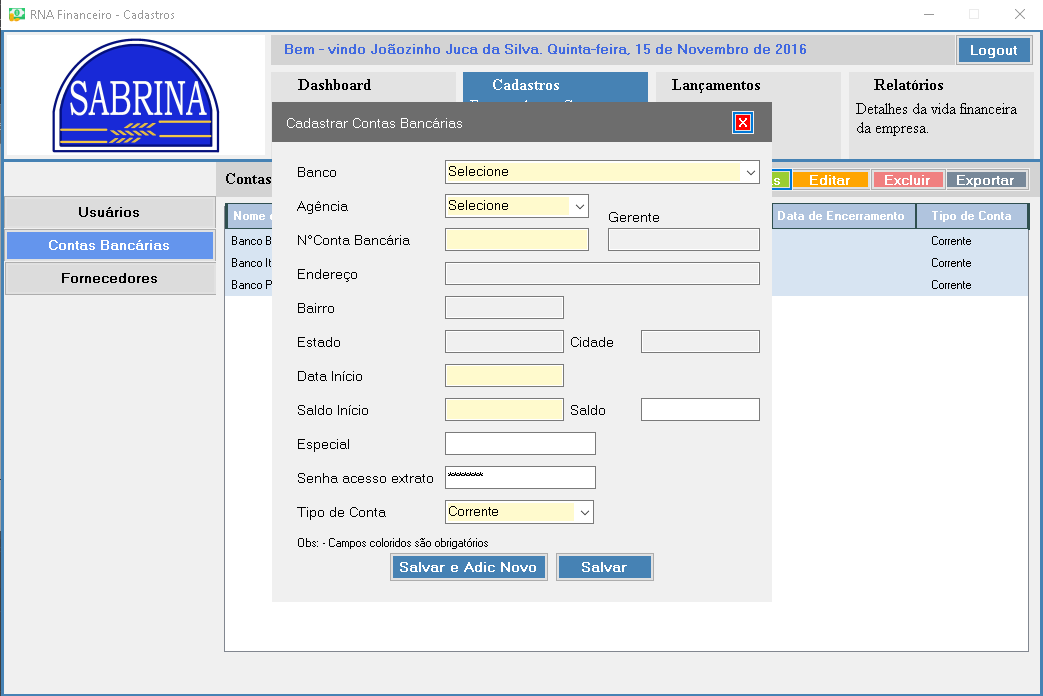
**Objetivo:** Este caso de uso permite realizar o cadastro de Contas Bancárias.

**Enunciado:**

**Descrição de tela:** Cada um dos dois botões terá um evento associado. O usuário digita os dados nos campos obrigatórios, os ComboBox (Banco, Agência, e Tipo de conta), os TextBox (Número, Data Início, Saldo Início). O usuário digita opcionalmente os TextBox (Endereço, Bairro, Gerente, Especial, Senha acesso extrato), e os ComboBox (Estado, e Cidade).

**Eventos:** No evento clique do botão Salvar, verifique se os campos obrigatórios estão preenchido, se ComboBox (Banco, Agência, Tipo de Conta) foram selecionados, verifique se o TexBox Número possui mais que 8 caracteres, se o TextBox Data Início é uma data válida, o TextBox Saldo Início é maior ou igual a zero, e o TextBox, Saldo é maior ou igual a zero, e execute o método CadastrarContaBancária da classe Conta Bancária, feche a tela de cadastro. No evento clique do botão Salvar e Adic novo, verifique se os campos obrigatórios estão preenchidos, se os ComboBox (Banco, Agência, Tipo de Conta) foram selecionados, se o TexBox Número da conta possui mais de 8 caracteres, se o TextBox Data Início é uma data válida, o TextBox Saldo Início é maior ou igual a zero, e o TextBox Saldo é maior ou igual a zero, e execute o método CadastrarContaBancária da classe Conta Bancária, retorna a tela de cadastro para um novo cadastro. As classes associadas estão definidas abaixo.

**Layout:**



**Classes**

CadastrarContaBancaria: Receba os campos obrigatórios, Banco, Agência, Número, Data início, Saldo início, tipo de conta, senha de acesso extrato, verifica se os campos obrigatórios foram preenchidos (Banco, Agência, Número, Data Início, Saldo Início, Saldo, Tipo de Conta), recebe opcionalmente os campos (Endereço, Bairro, Gerente, Especial, Senha acesso extrato, Estado, e Cidade) e execute o método AdicionarContaBancária.

AdicionarContaBancária: Armazena todos os campos digitados no método CadastrarContaBancaria (Banco, Agência, Número, Data Início, Saldo início, Sado, Tipo de Conta, Endereço, Bairro, Gerente, Especial, Senha acesso extrato, Estado, e Cidade).

## Editar contas bancárias

(Parte feita por: Vinícius)

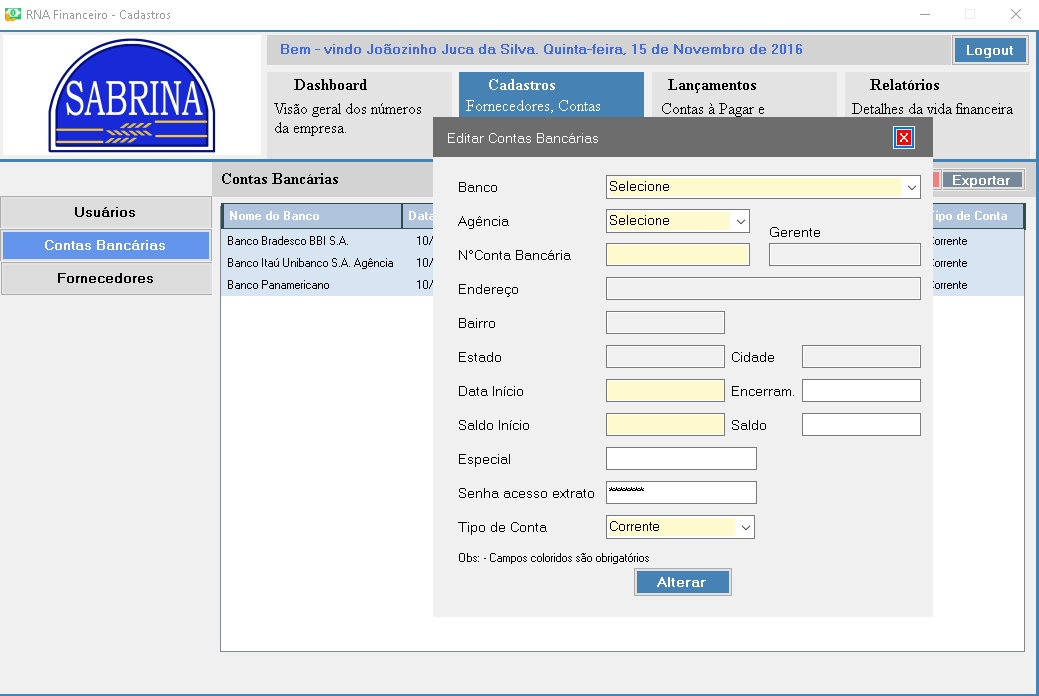
**Objetivo:** Esse caso de uso permite realizar a edição de uma conta bancária.

**Enunciado:**

**Descrição de tela:** O botão terá um evento associado. Os dados da conta selecionada são exibidos, campos obrigatórios os ComboBox (Banco, Agência, e Tipo de conta), os TextBox (Número, Data Início, Saldo Início). E se houverem os campos opcionais TextBox (Endereço, Bairro, Gerente, Especial, Senha acesso extrato), e os ComboBox (Estado, e Cidade).

**Eventos:** No evento clique do botão salvar verifique se os campos ComboBox (Banco, Agência, Tipo de Conta) foram selecionados, se o TexBox Número possui mais que 8 caracteres, se o TextBox Data Início é uma data válida, o TextBox Saldo Início é maior ou igual a zero, e o TextBox, Saldo é maior ou igual a zero, e execute o método EditarContaBancaria da classe Conta Bancaria.

**Layout:**



**Classes**

EditarContaBancaria: Receba os campos da conta selecionada, Banco, Agência, Número, Endereço, Bairro, Gerente, Estado, Cidade, Data início, Saldo início, especial, tipo de conta, senha de acesso extrato, verifica se os campos obrigatórios estão preenchidos (Banco, Agência, Número, Bairro, Cidade, Data Início, Saldo Início, Tipo de Conta) e execute o método AtualizarContaBancaria.

## Efetuar cadastro rápido de fornecedores

(Parte feita por: Vinícius)

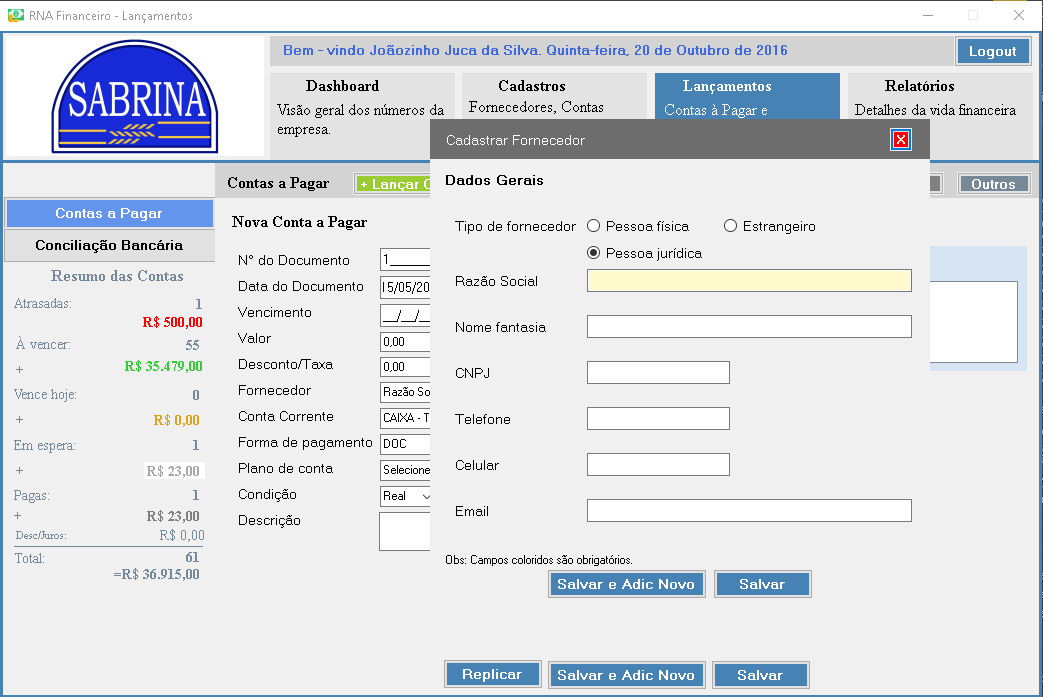
**Objetivo:** Esse caso de uso permite realizar o cadastro de fornecedores.

**Enunciado:**

**Descrição de tela:** Cada um dos dois botões terá um evento associado. O usuário digita os dados no campo obrigatório Textbox (Razão Social), e opcionalmente nos campos TextBox (Nome fantasia, Telefone, Celular e e-mail).

**Eventos:** No evento clique do botão Salvar, verifique se o campo obrigatório TextBox (Razão Social) possui um comprimento maior que zero caracteres, se o campo não obrigatório TextBox (Email), foi digitado, verifique se possui um caracter “@” e pelo menos 1 caractere antes e 1 depois do “@”, se o campo não obrigatório TextBox (Telefone) foi digitado, se os os caracteres digitados são números e execute o método CadastrarFornecedor da classe Fornecedor, feche a tela de cadastro de fornecedores. No evento clique do botão Salvar e Adic Novo, verifique se o campo obrigatório TextBox (Razão Social) possui um comprimento maior que zero caracteres, se o campo não obrigatório TextBox (Email), foi digitado, verifique se possui um caracter “@” e pelo menos 1 caracter antes e 1 depois do “@”, e execute o método CadastrarFornecedor da classe Fornecedor, retorne à tela de cadastro para realizar outro cadastro. As classes associadas estão definidas abaixo.

**Layout**



**Classes**



CadastrarFornecedor: Recebe o campo obrigatório textbox (razão social), opcionalmente os textbox (Nome fantasia, Telefone, Celular, Email), verifica se o campo obrigatório foi preenchido, e grava o fornecedor.

## Lançar contas a pagar

(Parte feita por: Vitor)

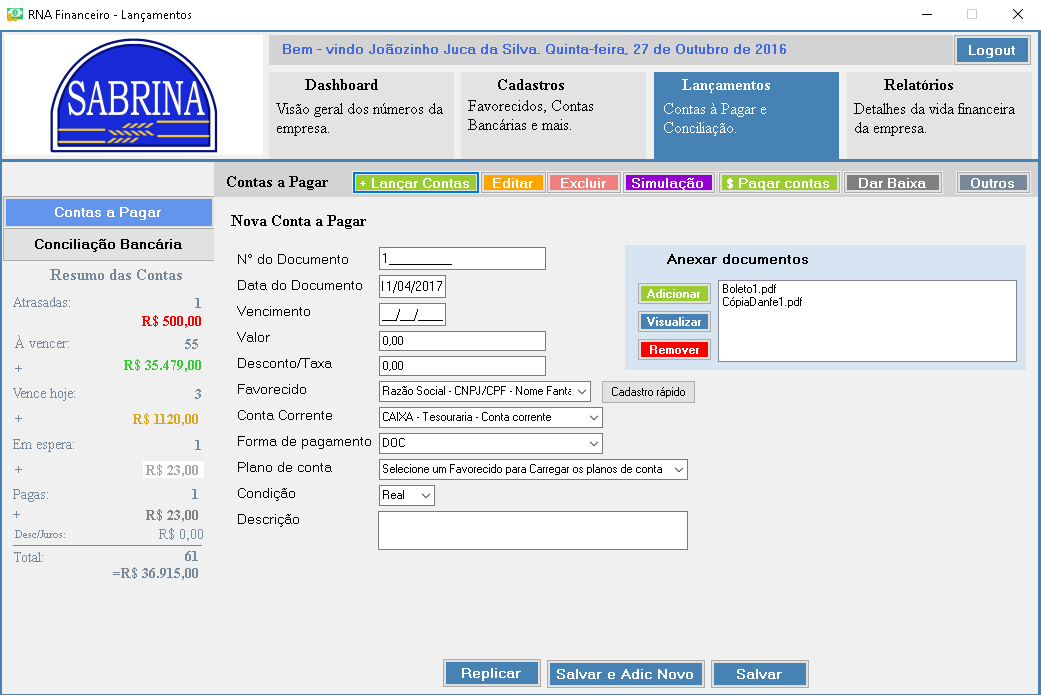
**Objetivo:** Este caso de uso permite o lançamento de novas conta no sistema.

**Enunciado:**

**Descrição da tela:** Cada um dos três botões da parte inferior, assim como os quatro botões do lado direto, terão um evento associado. Na parte superior da tela, há o menu de abas. No centro da tela, o usuário terá os textbox Numero do documento, Data do documento, Vencimento, Valor, Desconto/taxas, Descrição. No centro também se terá dos seguintes combobox, fornecedor, conta corrente, Forma de pagamento, Plano de conta e Condição, e ao lado da combobox do Fornecedor se terá um botão Cadastro Rápido. Na parte esquerda da tela sem tem o resumo das contas. Na parte superior direita da tela há a parte de anexo de documento da conta, onde se possui 3 botões, sendo eles, Adicionar, Visualizar, e o Remover. Na parte direita da tela, o usuário terá a parte de Rateio por Centro de Custa/Receita, onde há as textbox de percentual de Custo padrão e Total, e ao lado delas, as textbox de Valor Sugerido de Custo padrão e total. Na parte inferior direta da tela, há a parte de Rateio de Conta de Resultado, onde há os textbox de Percentual de Investimentos e Total, e ao lado direito delas, os textbox de Valor sugerido de Investimentos e Total. Na parte inferior da tela há a parte de replicação de contas, onde há o textbox de número de parcelas, e as combobox de Frequência de Vencto e de Frequência de Emissão. Logo abaixo há três botões Salvar, Salvar e Adic Novo e Remover Replica

**Eventos:** No evento clique do botão Salvar, verifique se o campo Numero do documento possui apenas números, o campo Data do documento possui somente números e é uma data válida, campo Data vencimento possui apenas números e é uma data válida, campo Desconto/taxas possui valor numérico (moeda), o campo Fornecedor foi selecionado alguma opção, o campo Conta corrente foi selecionado alguma opção, o campo Forma de pagamento foi selecionado alguma opção, o campo Condição foi selecionado alguma opção, o campo Plano de conta foi selecionado alguma opção, os campos Percentual e Valor sugerido (Custo padrão, Total de Rateio de custo e Investimentos e Total de Rateio de Conta de Resultado) possuem valores numéricos, e então execute o método LançarConta da classe Conta a pagar. No evento clique do botão Salvar e Adic Novo, verifique se o campo Numero do documento possui apenas números, o campo Data do documento possui somente números e é uma data válida, campo Data vencimento possui apenas números e é uma data válida, campo Desconto/taxas possui valor numérico (moeda), o campo Fornecedor foi selecionado alguma opção, o campo Conta corrente foi selecionado alguma opção, o campo Forma de pagamento foi selecionado alguma opção, o campo Condição foi selecionado alguma opção, o campo Plano de conta foi selecionado alguma opção, os campos Percentual e Valor sugerido ( Custo padrão, Total de Rateio de custo e Investimentos e Total de Rateio de Conta de Resultado) possuem valores numéricos, e então execute o método LançarConta da classe Conta a pagar e em seguida limpe os campos. Caso seja colocado algum número de parcela da conta, no evento clique do botão em Salvar ou Salvar Adic Novo, verifique se o campo Numero do documento possui apenas números, o campo Data do documento possui somente números e é uma data válida, campo Data vencimento possui apenas números e é uma data válida, campo Desconto/taxas possui valor numérico (moeda), o campo Fornecedor foi selecionado alguma opção, o campo Conta corrente foi selecionado alguma opção, o campo Forma de pagamento foi selecionado alguma opção, o campo Condição foi selecionado alguma opção, o campo Plano de conta foi selecionado alguma opção, os campos Percentual e Valor sugerido ( Custo padrão, Total de Rateio de custo e Investimentos e Total de Rateio de Conta de Resultado) possuem valores numéricos, o campo Frequência de Vencto foi selecionado alguma opção, o campo Frequência de emissão foi selecionado alguma opção e o campo Nº de Parcelas possui somente números, e então execute o método ReplicarConta junto com o método LançarConta da classe Conta a pagar. No evento clique do botão Remover Replica execute o método RemoverReplica da classe Conta a pagar. No evento clique do botão Cadastro Rápido, execute o método CadastrarFornecedor da classe Fornecedor. No evento clique do botão Adicionar, na parte de Anexo da tela, execute o método IncluirAnexo da classe Conta a paga, Anexo e exiba o nome do anexo na tela ao lado. No evento clique do botão visualizar, na parte de Anexo da tela, execute o método VisualizarAnexno da classe Conta a pagar, Anexo. No evento clique do botão Remover, na parte de Anexo da tela, executo o método RemoverAnexo da classe Conta a pagar, Anexo.

**Layout:**



**Classes:**



CadastrarFornecedor: Recebe a o campo razão social, nome fantasia, tipo, e-mail, celular, telefone e grava o fornecedor.

IncluirAnexo: Recebe o arquivo seleciona pelo usuário, e grava associado com a conta.

VisualizarAnexo: Exibe o anexo da conta.

RemoverAnexo: Remove o anexo que está associado com aquela conta.

LançarConta: Recebe os dados do campo Número da conta, Data do documento, Vencimento, Valor, Desconto/Taxa, Fornecedor, Conta corrente, Forma de pagamento, Plano de conta, Condição e Descrição, e grava.

ReplicarConta: Recebe a Frequência de Vencto, Frequência de emissão e Número de parcelas e grava associado a conta.

RemoverReplca: remove a replica associada com a conta.

## Editar contas a pagar

(Parte feita por: Vitor)

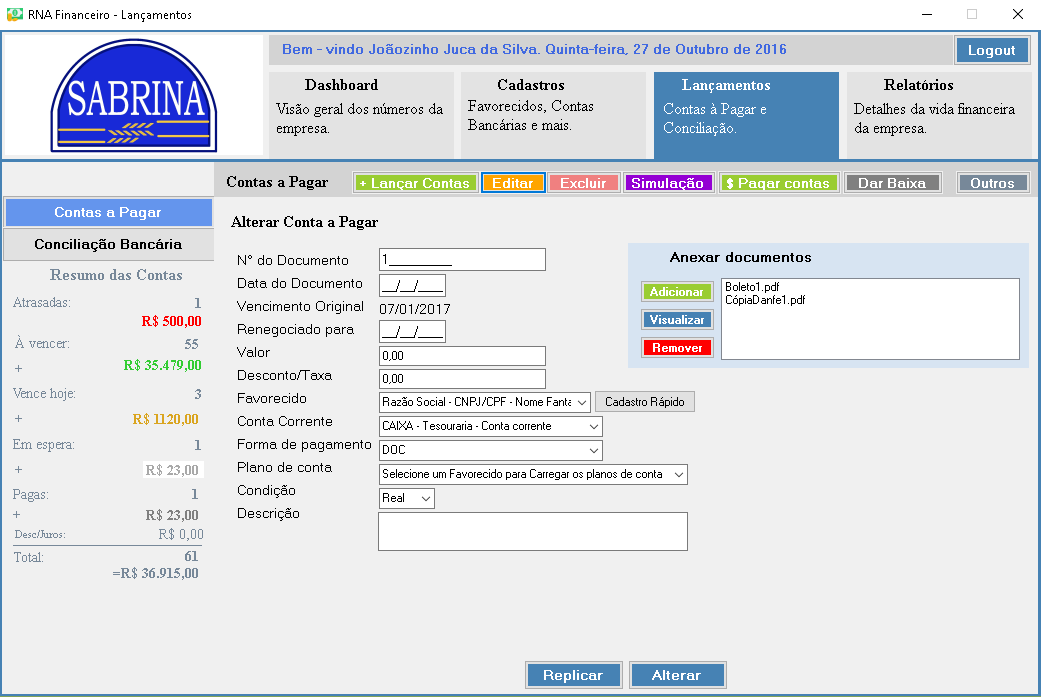
**Objetivo:** Este caso de uso permite a alteração de dados das contas já lançadas no sistema.

**Enunciado:**

**Descrição da tela**: Cada um dos dois botões da parte inferior, assim como os quatro botões do lado direto, terão um evento associado. Na parte superior da tela, há o menu de abas. No centro da tela, será exibido a data de vencimento original da conta e terá os textbox Numero do documento, Data do documento, Renegociado para, Valor, Desconto/taxas, Descrição. No centro também se terá dos seguintes combobox, Fornecedor, Conta corrente, Forma de pagamento, Plano de conta e Condição, e ao lado da combobox do Fornecedor se terá um botão Cadastro Rápido. Na parte esquerda da tela sem tem o resumo das contas. Na parte superior direita da tela há a parte de anexo de documento da conta, onde se possui 3 botões, sendo eles, Adicionar, Visualizar, e o Remove. Na parte direita da tela, o usuário terá a parte de Rateio por Centro de Custa/Receita, onde há as textbox de percentual de Custo padrão e Total, e ao lado delas, as textbox de Valor Sugerido de Custo padrão e Total. Na parte inferior direta da tela, há a parte de Rateio de Conta de Resultado, onde há os textbox de Percentual de Investimentos e Total, e ao lado direito delas, os textbox de Valor sugerido de Investimentos e Total. Na parte inferior da tela há a parte de replicação de contas, onde há o textbox de número de parcelas, e as combobox de Frequência de Vencto e de Frequência de Emissão. Logo abaixo há dois botões Alterar e Remover Replica.

**Eventos:** No evento clique do botão Alterar execute o método EditarConta da classe Conta a pagar. Caso seja colocado algum número de parcela da conta, ao clicar em Salvar ou Salvar Adic Novo, execute o método ReplicarConta junto com o método EditarConta da classe Conta a pagar. No evento clique do botão Remover Replica execute o método RemoverReplica da classe Conta a pagar. No evento clique do botão Cadastro Rápido, execute o método CadastrarFornecedor da classe Fornecedor. No clique do botão Adicionar, na parte de Anexo da tela, execute o método IncluirAnexo da classe Conta a paga, Anexo e exiba o nome do anexo na tela ao lado. No evento clique do botão visualizar, na parte de Anexo da tela, execute o método VisualizarAnexno da classe Conta a pagar, Anexo. No evento clique do botão Remover, na parte de Anexo da tela, executo o método RemoverAnexo da classe Conta a pagar, Anexo.

**Layout:**



**Classes**:



CadastrarFornecedor: Recebe a o campo razão social, nome fantasia, tipo, email, celular, telefone e grava o fornecedor.

IncluirAnexo: Recebe o arquivo seleciona pelo usuário, e grava associado com a conta.

VisualizarAnexo: Exibe o anexo da conta.

RemoverAnexo: Remove o anexo que está associado com aquela conta.

EditarConta: Recebe os campos que foram alterados e grava.

ReplicarConta: Recebe a Frequência de Vencto, Frequência de emissão e Número de parcelas e grava associado a conta.

RemoverReplca: remove a replica associada com a conta.

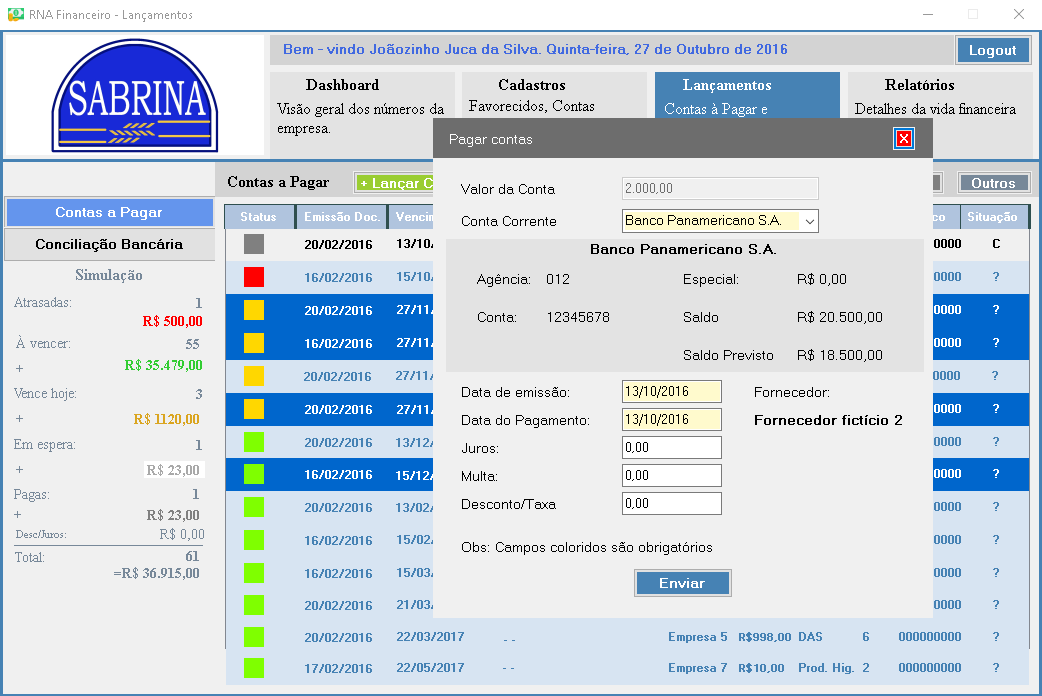
## Pagar conta

(Parte feita por: Vanessa)

**Objetivo:** Este caso de uso permite pagar uma ou várias contas através de integração com banco.

**Enunciado  
Descrição da tela:** O valor da conta selecionada no programaé exibido para o usuário e isto não pode ser alterado. A conta corrente é exibida conforme foi registrada no lançamento, o campo é indicado por uma cor diferenciada. As informações da conta corrente selecionada são pré carregadas e exibidas em labels contendo o número da agência, o número da conta, especial no formato monetário, o saldo atual no formato monetário e o saldo previsto no formato monetário. O fornecedor corresponde a quem forneceu a matéria prima ou serviços, já vem pré carregada no formato texto, é indicado por uma cor.A data da emissão corresponde a data que o documento foi emitido, já ve pre carregada pelo sistema no formato dd/mm/aaaa, é indicada por uma cor. A data de pagamento corresponde ao dia que será paga a conta, a data já vem pré carregada no formato dd/mm/aaaa, é indicado por uma cor.O campo numérico juros poderá ser preenchido pelo usuário no formato monetário. O campo numérico multa poderá ser preenchido pelo usuário no formato monetário. O campo numérico desconto/taxa poderá ser preenchido pelo usuário no formato monetário. Na parte inferior do formulário aparece uma observação de que os campos coloridos obrigatórios. O botão enviar tem um evento associado. **Eventos:** No evento select do combobox conta corrente execute o método obter conta bancária da classe Conta bancária e habilite o evento load para as informações referentes a esta conta corrente. No evento clique no campo da data de emissão exibe um calendário .No evento clique na data de pagamento exibe um calendário. No evento clique do botão enviar verifique se os dados foram preenchidos de forma correta (a conta corrente deve estar preenchida com um número de conta bancária já cadastrada,a data da emissão deve conter uma data válida não pode estar em branco e a data da emissão deve ser menor ou igual a data do pagamento, a data de pagamento deve ser uma data válida e não pode estar em branco). Execute o método gerar arquivo remessa da subclasse Remessa e após executar o método enviar arquivo remessa em lote da subclasse Remessa. Depois executar o método atualizar situação da conta da classe Pagar conta. As classes associadas estão definidas abaixo.

**Layout**



**Classes**





ObterContasBancárias: Carregar os dados das contas bancárias(número da agência, e da conta, nome do banco) que foram anteriormente cadastradas e indicadas nas contas selecionadas.

ObterFornecedores: Carregar os dados do fornecedor que forama anteriormente cadastradas.

GerarArqRemessa: Os dados da conta e os dados da conta bancária estabelecidos no layout do banco são recebidos e é gerado um arquivo remessa em formato texto específico do layout do banco selecionado.

EnviarArquivoRemessa: O arquivo remessa gerado é enviado ao banco selecionado.

AtualizarSituacaoDasContas: A situação da conta é atualizada, podendo ser paga, a vencer, atrasada, aguardando arquivo retorno

ObterTodasAsContas: Todas as contas registradas são exibidas.

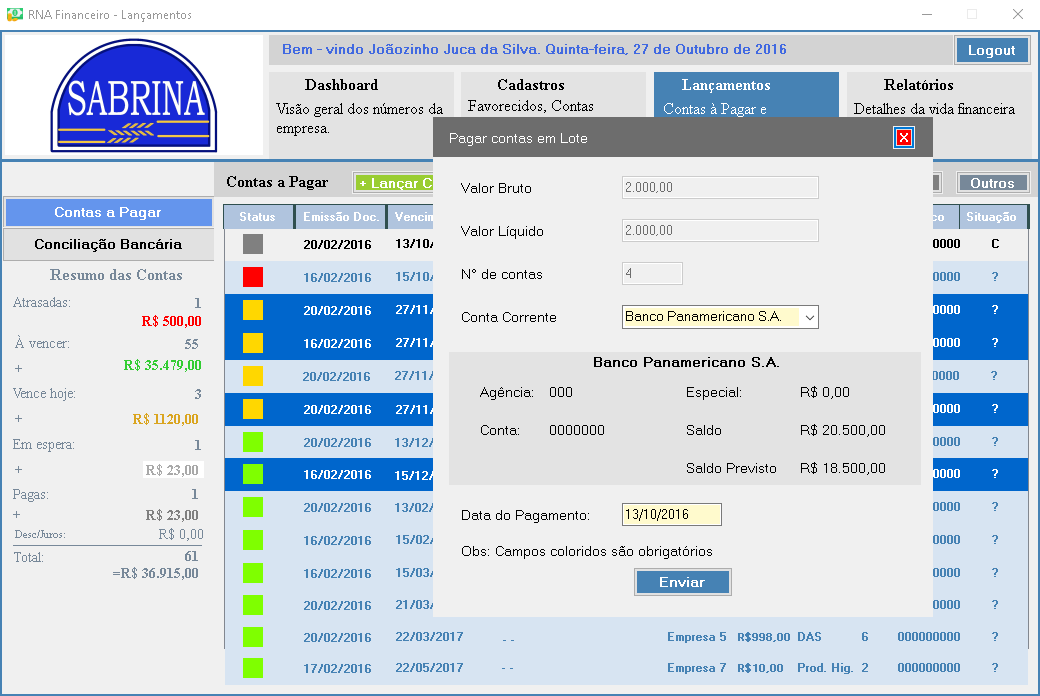
## Pagar contas em lote

(Parte feita por: Vanessa)

**Objetivo:** Pagar várias contas que foram selecionadas através de um único arquivo remessa.

**Enunciado:  
Descrição da tela:** O valor bruto corresponde ao valor total de todas as contas selecionadas, é um campo numérico que já vem preenchido e não pode ser alterado, o campo valor líquido é um campo numérico que já vem preenchido e não pode ser alterado. O campo número de contas corresponde ao número total de contas selecionadas, é um campo numérico que já vem preenchido e não pode ser alterado, esses três campos tem como fundo a cor azul . A conta corrente é exibida , o campo é indicado por uma cor diferenciada. As informações da conta corrente selecionada são pré carregadas e exibidas em labels contendo o número da agência, o número da conta, especial no formato monetário, o saldo atual no formato monetário e o saldo previsto no formato monetário. A data da emissão corresponde a data que o documento foi emitido, já vem pre carregada pelo sistema no formato dd/mm/aaaa, é indicada por uma cor. O fornecedor corresponde a quem forneceu a matéria prima ou serviços, já vem pré carregada no formato texto, é indicado por uma cor. Na parte inferior do formulário aparece uma observação de que os campos coloridos obrigatórios. O botão enviar tem um evento associado. **Eventos:** No evento select do campo conta corrente execute o método obter conta bancária da classe Conta bancária e habilite o evento load para as informações referentes a esta conta corrente. O evento clique na data de pagamento exibire um calendário. No evento clique do botão enviar verifique se os dados foram preenchidos de forma correta (a conta corrente deve estar preenchida com um número de conta bancária já cadastrada,a data da emissão deve conter uma data válida não pode estar em branco e a data da emissão deve ser menor ou igual a data do pagamento, a data de pagamento deve ser uma data válida e não pode estar em branco). Execute o método gerar arquivo remessa da subclasse Remessa e após executar o método enviar arquivo remessa em lote da subclasse Remessa. Depois executar o método atualizar situação da conta da classe Pagar contas. As classes associadas estão definidas abaixo.

**Layout**



**Classes**





ObterContasBancárias: Carregar os dados das contas bancárias(número da agência, e da conta, nome do banco) que foram anteriormente cadastradas.

ObterFornecedor: Carregar os dados do fornecedor que foram anteriormente cadastrados.

GerarArqRemessa: Os dados da conta e os dados da conta bancária estabelecidos no layout do banco são recebidos e é gerado um arquivo remessa em formato texto específico do layout do banco selecionado.

EnviarArqRemessa: O arquivo remessa gerado é enviado ao banco selecionado.

AtualizarSituacaoDasContas: A situação da conta é atualizada, podendo ser paga, a vencer, atrasada, aguardando arquivo retorno

ObterTodasAsContas: Todas as contas registradas são exibidas

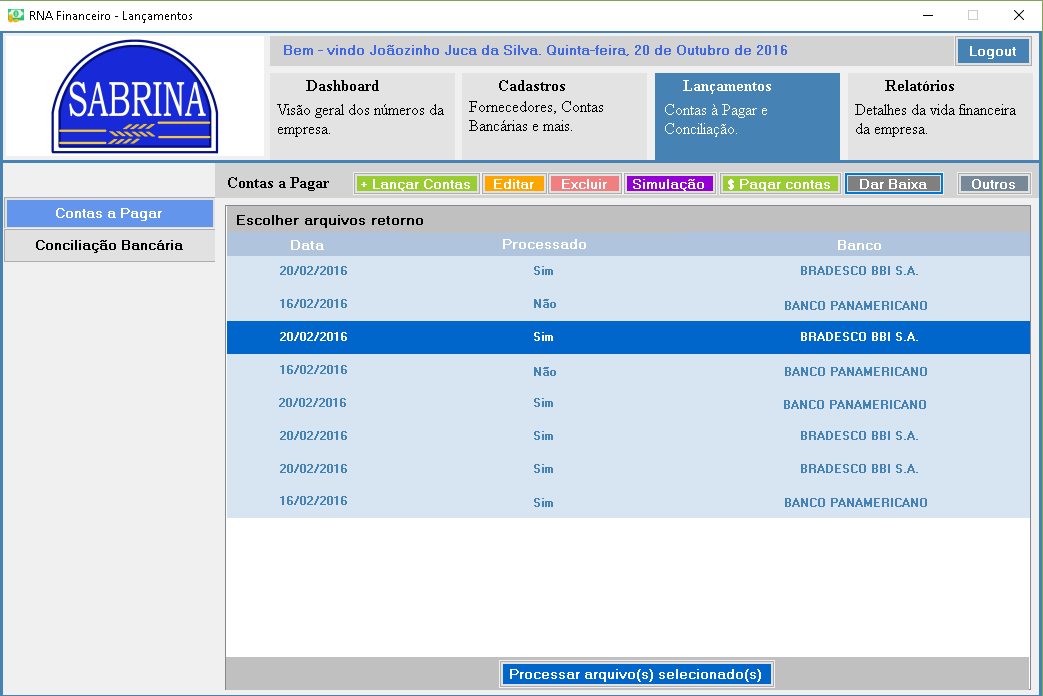
## Dar baixa pelo arquivo retorno

(Parte feita por: Vanessa)

**Objetivo**: Confirmar o pagamento das contas, utilizando o arquivo retorno disponibilizado pelo banco.

**Enunciado:  
Descrição da tela:** No menu de contas a pagar fica o botão de dar baixa, quando selecionado este exibe uma tabela contendo todos os arquivos retorno recebidos do banco. Nesta tabela consta um textbox informando para escolher arquivos retorno, abaixo na primeira coluna tem a data do recebimento do arquivo, em sequência o tipo do arquivo, a operação em que será usado esse arquivo, e o banco provedor do arquivo. Na parte inferior no centro, encontra-se o botão processar arquivo(s) selecionado(s).  
**Eventos:** No evento cliquedo botãodar baixa o evento load mostre a tela com os todos os arquivos retorno recebidos do banco que ainda não foram processados e que estão no banco de dados. Após selecionar os arquivos no evento clique do botão processar arquivo(s) selecionado(s) execute o método Baixar arquivo retorno da Classe Retorno.

**Layout**



**Classes**





BaixarArquivoRetorno: Este método utiliza os dados do arquivo retorno selecionado, que foi obtido do banco de dados, para confirmar o pagamento das contas. Depois da confirmação o método baixar arquivo retorno chama os métodos Atualizar situação das contas da classe Pagar contas e o método Obter todas as contas da classe Pagar contas. Esse método leva para a tela com todas as contas e se a conta não foi paga a linha da tabela desta ficará toda vermelha e quando selecionada executa o método Exibir motivo de recusa da classe Parcela, Retorno.

AtualizarSituacaoDasContas: A situação da conta é atualizada, podendo ser paga, à vencer, atrasada, aguardando arquivo retorno

ObterTodasAsContas: Todas as contas registradas são exibidas

ExibirMotivoDeRecusa: neste método é exibido uma mensagem informando o motivo da recusa.

MostrarMsgIndisponibilidade: Neste método o sistema busca o arquivo retorno no banco de dados e se não é encontrado nenhum arquivo é exibido uma mensagem informando que não há arquivo disponível.

## Dar baixa pela tesouraria

(Parte feita por: Vanessa)

**Objetivo:** Confirmar o pagamento de uma conta que foi lançada pela tesouraria e não foi paga pela integração com o banco.

**Enunciado:  
Descrição:** O valor da conta selecionadaé exibido para o usuário e este não pode ser alterado. A conta corrente é exibida como tesouraria, o campo é indicado por uma cor diferenciada. As informações da conta corrente selecionada são pré carregadas e exibidas em labels contendo o número da agência, o número da conta, especial no formato monetário, o saldo atual no formato monetário e o saldo previsto no formato monetário. A data da emissão corresponde a data que o documento foi emitido, já vem pre carregada pelo sistema no formato dd/mm/aaaa, é indicada por uma cor. O fornecedor corresponde a quem forneceu a matéria prima ou serviços, já vem pré carregada no formato texto, é indicado por uma cor. A data de pagamento corresponde ao dia que será paga a conta, a data já vem pré carregada no formato dd/mm/aaaa, é indicado por uma cor.O campo numérico juros poderá ser preenchido pelo usuário no formato monetário. O campo numérico multa poderá ser preenchido pelo usuário no formato monetário. O campo numérico desconto/taxa poderá ser preenchido pelo usuário no formato monetário. Na parte inferior do formulário aparece uma observação de que os campos coloridos obrigatórios. O botão Dar baixa tem um evento associado.  
**Evento:**  No evento select do campo data de emissão exiba um calendário. No evento select do campo data de pagamento exiba um calendário. No evento clique do botão enviar verifique se os dados foram preenchidos de forma correta (os camos obrigatórios devem ser preenchidos, as datas devem ser válidas, a data de emissão deve ser menor que a data de pagamento). Execute o método DarBaixaEmContaPelaTesouraria da classe Contas a pagar. Em sequência execute o método AtualizarSituacaoDasContas . As classes associadas estão definidas abaixo.

**Layout**



**Classes**



DarBaixaPelaTesouraria: Este método confirma que a conta já foi paga.

AtualizarSituacaoDasContas: A situação da conta é atualizada, podendo ser paga, à vencer, atrasada, aguardando arquivo retorno

ObterTodasAsContas: Todas as contas registradas são exibidas.

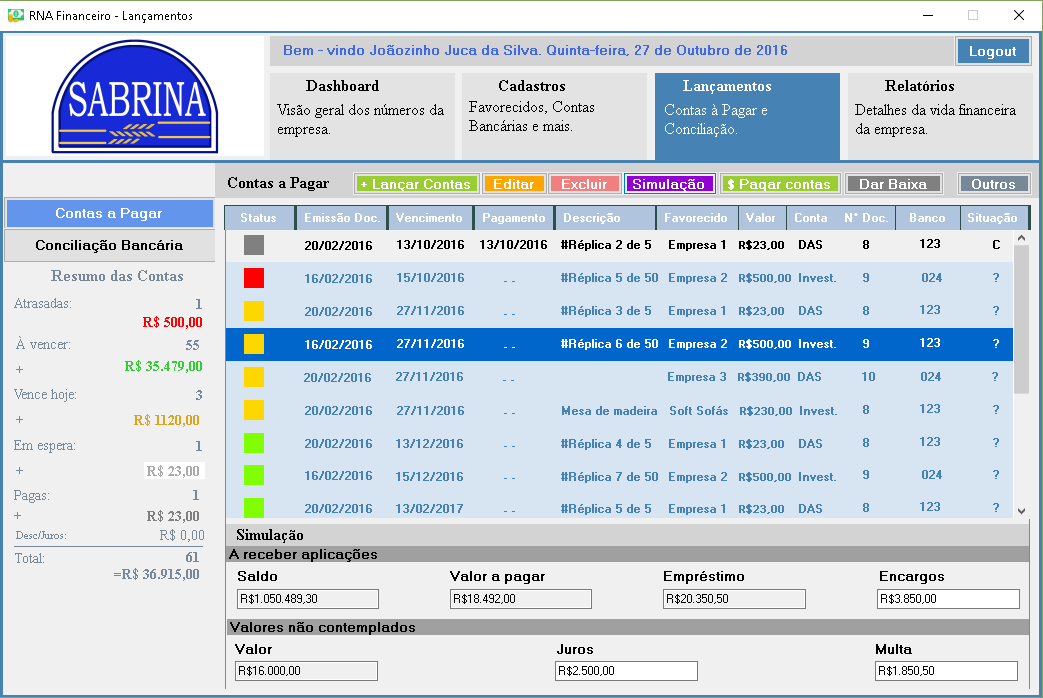
## Simular contas a pagar

(Parte feita por: Vanessa)

**Objetivo:** Simular o valor total das contas para decidir quais contas serão pagas

**Enunciado:**  
**Descrição da tela**: No menu de contas a pagar consta um botão simulação, que quando selecionado exibe uma tela,esta exibe uma tabela com todas as contas registradas**,** onde o usuário poderá selecionar as contas para simular, nessa tabela consta o status da conta indicado por cores, a emissão do documento no formato data, a data do vencimento no formato data, data do pagamento no formato data , descrição da conta, o fornecedor que é quem receberá o valor da conta, o valor da conta no formato monetário, conta que corresponde ao plano de contas que a conta foi registrada, o nº do documento que é a nota fiscal ou algum nº de documento correspondente, o nome do banco, a condição que é se a conta é real ou prevista para contas parceladas. Na parte inferior tem uma textbox informando da simulação e abaixo ideal na parte inferior esquerda tem um painel, com resultados das contas selecionadas, com um campo saldo em formato monetário exibindo o saldo total, um campo com o valor a pagar em formato numérico para exibir os valores calculados automaticamente ao selecionar as contas, um campo empréstimo em formato monetário informando o valor que corresponde a um possível empréstimo necessário para quitar as contas , e um campo encargos em formato monetário exibindo o valor dos encargos financeiros gerados a partir do empréstimo. No canto inferior direito há um painel com os valores que não foram pagos na simulação, nesse painel consta um campo no formato monetário com o valor das contas que não foram pagas, um campo para os juros no formato monetário, para exibir o valor total dos juros, e um campo multa em formato monetário para exibir o valor total das multas.  
**Eventos:** No evento click do botão simulação o evento load exibe a tela com as contas através do metodo ObterTodasAsContas. No evento select das contas o sistema executa o método simular pagamento de contas da classe contas a pagar.A tela é dividida em duas, na parte superior ficam as contas e na parte inferior e éxibido um painel com os valores selecionados, neste painel consta o saldo da conta, o valor total das contas que é acumulado a cada evento select das contas, o valor do empréstimo que é o valor do saldo menos o valor do valor total das contas e deste valor final, se negativo é exibido no textbox empréstimos esse valor e no textbox encargos é calculado e exibido o valor dos juros e multas decorrentes do empréstimo. No painel valores não contemplados é exibido o textbox valor que corresponde ao resultado do valor total das contas selecionadas menos o valor do saldo, e no textbox juros o valor total dos juros acumulados de cada conta selecionada é informado, e no textbox multa o valor total das multas acumuladas de cada conta selecionada é informado.

**Layout**



**Classes**

GerarSimulacao: Neste método o sistema obtém o saldo do banco de dados, a data do vencimento, o valor da conta, o valor dos juros, o valor das multas, desconto/ taxas, calcula os valores correspondentes, se o valor das contas ultrapassar o valor do saldo, deve ser calculado o valor de um empréstimo e o valor dos encargos decorrentes. Os valores devem ser exibidos como informado na descrição da tela.

## Importar extrato bancário

(Parte feita por: Giovane)

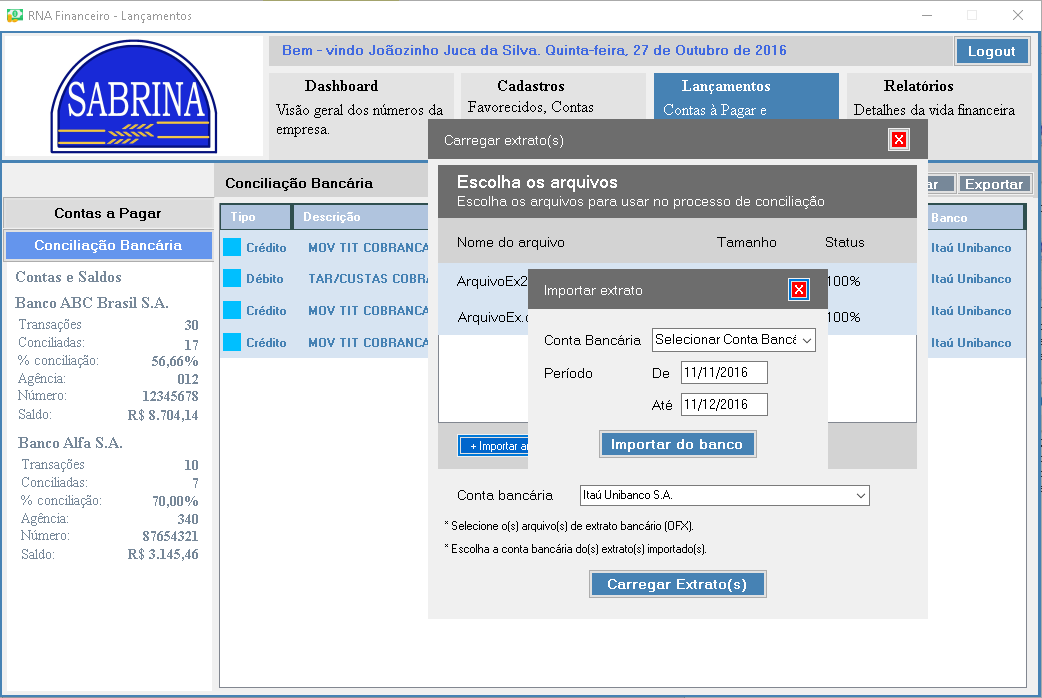
**Objetivo:** Realizar a importação de um extrato do banco para o servidor do sistema online.

**Enunciado:**

**Descrição da tela:** O botão da parte inferior da tela terá um evento associado. No combobox será escolhida uma conta bancária dentre as cadastradas no sistema, a listagem será feita com o método ObterContasBancárias. A maskedTextbox “De” iniciará vazia, e a “Até” iniciará com a data atual. Ao se selecionar uma conta bancária a maskedTextbox “De” será iniciada com a data final do último extrato importado da conta bancária referida.

**Eventos:** No evento SelectedIndexChanged da combobox “Conta bancária” execute o método DatarUltimoExtrato, coloque a data recebida no maskedTextbox “De” e habilite o botão “Importar do banco”. No clique do botão “Importar do banco” execute o método ImportarExtrato, retornando do banco o extrato num arquivo texto em formato ofx (com a Data, o Valor, a Descrição e o Tipo da transação) relativo ao período escolhido e armazenando-o junto aos outros extratos do sistema e feche a tela de importar extrato. Todos esses métodos estão na classe “Extrato, Arquivo”.

**Layout:**



**Classes:**



ObterContasBancárias: Lista todas as contas bancárias cadastradas mostrando o nome do banco, o tipo e número da conta e o saldo da conta.

DatarUltimoExtrato: Recebe a conta bancária e retorna a data de “Até” do último “extrato, arquivo” registrado.

ImportarExtrato: Recebe a conta bancária, as datas “De” e “Até” e envia ao banco a solicitação do extrato referente ao período informado. Quando recebe o extrato salva-o no armazenamento de dados do servidor do sistema.

## Carregar extrato bancário

(Parte feita por: Giovane)

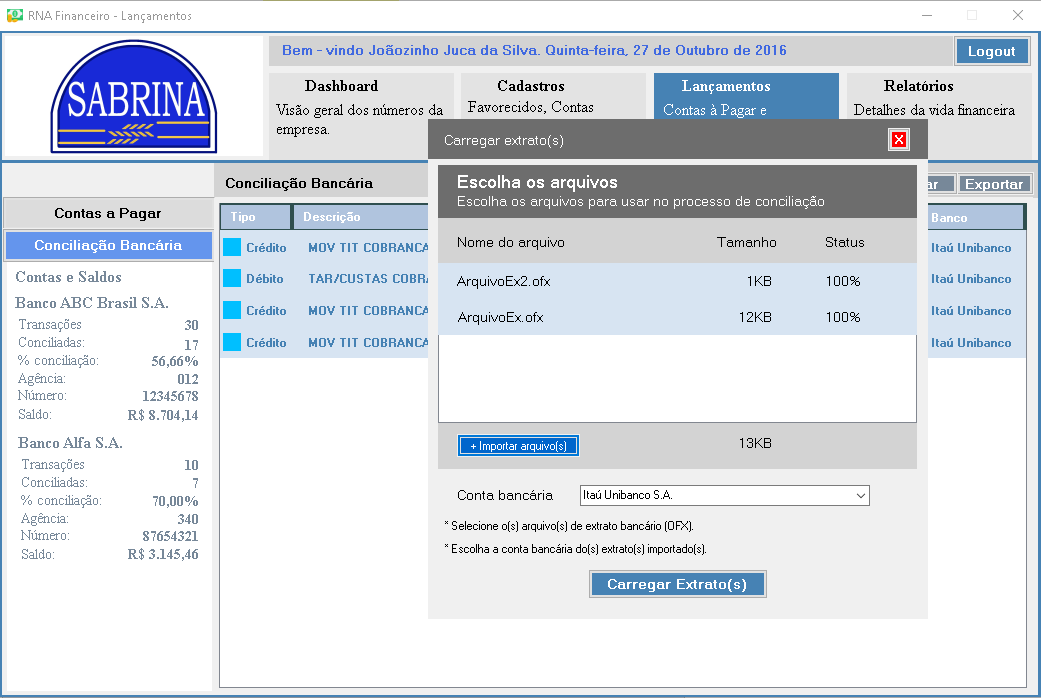
**Objetivo:** Escolher extratos armazenados no servidor online para serem carregados no sistema para posterior conciliação.

**Enunciado:**

**Descrição da tela:** O botão da parte inferior da tela terá um evento associado. A Listbox do quadro central será inicializada com os dados de todos os extratos já importados pelo sistema (através do método ObterExtratos) e, se o extrato já foi conciliado, o campo status informará a porcentagem das transações contidas no extrato que já foram conciliadas com o sistema. O botão Importar arquivo(s) chamará a tela de “Importar extrato bancário”. No combobox será escolhida uma conta bancária dentre as cadastradas no sistema, a listagem será feita com o método ObterContasBancárias. O botão Carregar Extrato(s) chamará a tela “Conciliar extrato bancário”.

**Eventos:** No evento load do formulário liste os extratos disponíveis na Listbox através do método ObterExtratos e carregue as contas bancárias cadastradas no sistema na combobox “Conta Bancária” através do método ObterContasBancarias. No evento clique do botão Importar extrato bancário abra a tela de “Importar extrato bancário”. No evento clique do botão “CarregarExtrato(s)” execute o método CarregarExtrato da classe “Extrato, Arquivo” e abra a tela de “Conciliar extrato bancário”.

**Layout:**



**Classes:**



ObterExtratos: Lista todos os extratos importados para o armazenamento de dados do servidor do sistema em ordem decrescente (da data mais recente à mais longínqua).

ObterContasBancárias: Lista todas as contas bancárias cadastradas mostrando o nome do banco, o tipo e número da conta e o saldo da conta.

CarregarExtrato: O sistema lê os dados Data, Valor, Descrição e Tipo de transação de cada conta (que ainda não foi conciliada) contida no arquivo de texto do extrato conforme o layout de extrato para sistemas informacionais armazenando-os na memória principal com o número da quantidade total de contas, essas informações serão organizadas na tela de “Conciliação bancária” que será mostrada para posterior conciliação.

## Conciliar extrato bancário

(Parte feita por: Giovane)

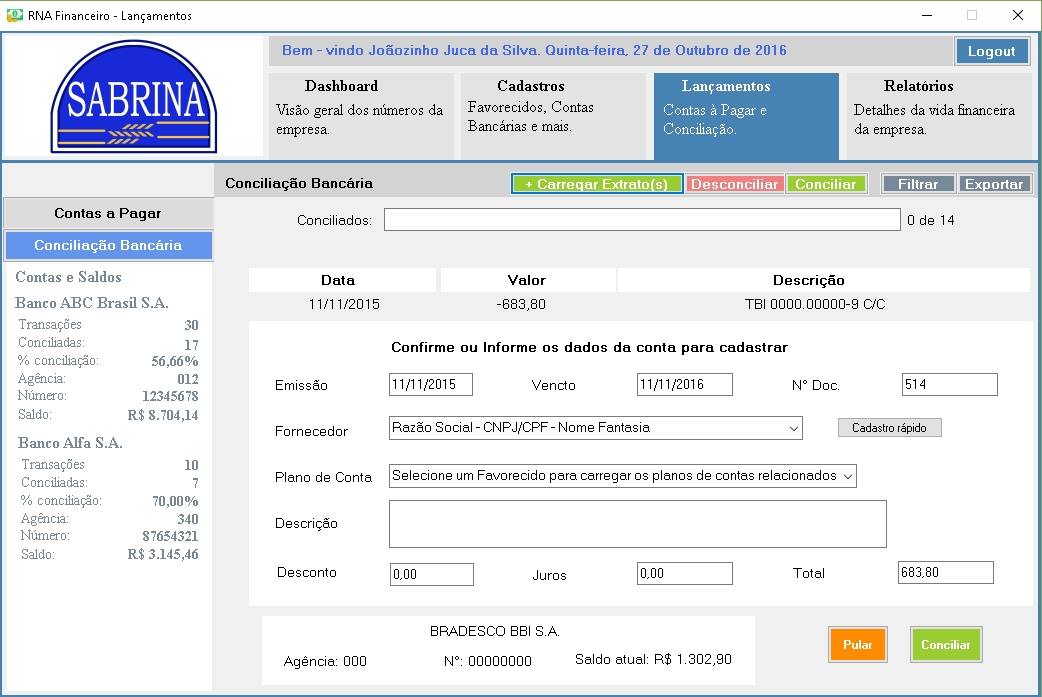
**Objetivo:** Confrontar a movimentação bancária com o controle interno do sistema

**Enunciado:**

**Descrição da tela:** Os botões da parte inferior da tela e o da parte central terão eventos associados. A barra de progresso na parte superior do quadro de conciliação iniciará na posição 0 de N, onde N indica a quantidade de contas identificadas no carregamento do extrato, aparecendo nessa tela as informações de data, valor e descrição da conta 1 entre N do extrato. No quadro inferior a essa parte o sistema, através do método CompararContas da classe Conta a Pagar, compara os dados da conta do extrato com as contas do sistema e se encontra dados iguais já mostra preenchidos os campos Data de emissão, Data de Vencimento, N° de Documento, Fornecedor, Plano de Conta, Descrição, Desconto, Juros e Total da conta encontrada. Na parte inferior aparecem os dados da conta bancária escolhida para a conciliação, bem como seu saldo atual do controle interno do sistema. O botão Conciliar será acionado e, com isso, o saldo atual da conta bancária será atualizado, a barra de progresso da parte superior da tela será incrementada em 1 e os dados da conta do extrato seguinte serão mostrados. Se o usuário substituir os dados da conta sugeridos ou o sistema não encontrar semelhança entre as contas e o usuário pressionar o botão Conciliar, a conta será cadastrada como movimentação realizada por fora do sistema, alterando apenas o saldo do banco, cadastrando a conta como conta extra através do método CadastrarContaExtra da classe Conciliações e incrementando a barra de progresso em 1. Também será possível cadastrar um novo fornecedor no momento do cadastro de nova conta através do botão Cadastro Rápido. Se o usuário escolher o botão Pular, nenhuma conta é cadastrada nem conciliada, e passa-se aos dados da próxima conta. Repete-se todo o processo até que se chegue à conta N encontrada no extrato, então quando se acionar novamente o botão Conciliar, executará sua função e chamará a tela de listagem de conciliações (no caso do botão Pular apenas levará direto à mesma tela).

**Eventos:** No evento load da tela, execute o método CompararContas e preencha os campos Data, Valor e Descrição da conta referentes à primeira conta ainda não conciliada no extrato, execute o método ObterContasBancarias e preencha os dados Nome do banco, Agência, Número e Saldo atual da conta bancária a que o extrato se refere. No evento clique do botão “Cadastro Rápido” mostre a tela de cadastro rápido de fornecedor. No evento clique do botão Pular execute o método PularConciliacao. No evento clique do botão Conciliar verifique se os dados Emissão, Vencto, N° Doc, Descrição, Desconto, Juros, Total ou das combobox Fornecedor e Plano de Conta foram alterados: se sim, execute os métodos CadastrarContaExtra, AtualizarSituacaoDasContas (indique a situação da conta como “Conciliada”) e AtualizarSaldo (preencha os novos dados nos campos referentes à conta bancária). Senão apenas execute os métodos AtualizarSituacaoDasContas (indique a situação da conta como “Conciliada”) e AtualizarSaldo (preencha os novos dados nos campos referentes à conta bancária).

**Layout:**



**Classes:**

CompararContas: Compara os dados das contas lançadas no sistema com os dados das contas encontradas no extrato e, encontrando semelhanças, sugere a conta na conciliação

CadastrarContaExtra: Cadastra transações efetuadas na conta bancária fora do controle interno do sistema

PularConciliacao: Calcula porcentagem do extrato que equivale a conta que está sendo pulada, subtrai do status do arquivo retorno referente e mostra dados da próxima conta ou volta à listagem de conciliações.

AtualizarSituacaoDasContas: Muda situação das contas para “Conciliada”.

AtualizarSaldo: Altera o saldo da conta bancária de acordo com a movimentação requerida.

ObterContaBancaria: Lista os dados da conta bancária cadastrada solicitada mostrando o nome do banco, a agência, o número e o saldo da conta.