Fundação Indaiatubana de Educação e Cultura

Fillipe Martins

Lucas de Almeida

Renan Pereira

**Projeto Cantina**

Indaiatuba

Junho de 2017

Fillipe Martins

Lucas de Almeida

Renan Pereira

**Projeto Cantina**

Documentação do trabalho prático desenvolvido na disciplina Linguagem de Programação II sob orientação do professor Fillipe Silva.

Indaiatuba

Junho de 2017

**Sumário**

Introdução .................................................................................................................................. 3

Atores do sistema ....................................................................................................................... 4

Funcionário cantina ........................................................................................................ 4

Cliente comum ............................................................................................................... 5

Cliente funcionário ......................................................................................................... 6

Janelas da aplicação ................................................................................................................... 7

Janela principal .............................................................................................................. 7

Janelas secundárias ........................................................................................................ 8

Controle de débito .............................................................................................. 8

Inserção de produtos ........................................................................................... 9

Relatório de receita do dia ................................................................................ 10

Banco de dados da aplicação .................................................................................................... 11

Tabelas do banco de dados ........................................................................................... 11

Produto ............................................................................................................. 11

Cliente .............................................................................................................. 12

Dívida ............................................................................................................... 13

Compra ............................................................................................................. 14

Conclusão ................................................................................................................................ 15

Referências .............................................................................................................................. 16

**Introdução**

O projeto idealizado na linguagem de programação C# visa informatizar o atual sistema da cantina, uma vez que este não tem um controle informatizado para o seu controle geral, portanto, se torna útil para uma suposta evolução em seu funcionamento.

**Atores do Sistema**

O Sistema possui três atores, isto é, pessoas que interferem com este, sendo eles:

**I. Funcionário Cantina**

Este ator é responsável por operar o sistema, sendo que suas ações se resumem a iniciar compra, inserir pedido, solicitar pagamento, devolver troco e encerrar venda.

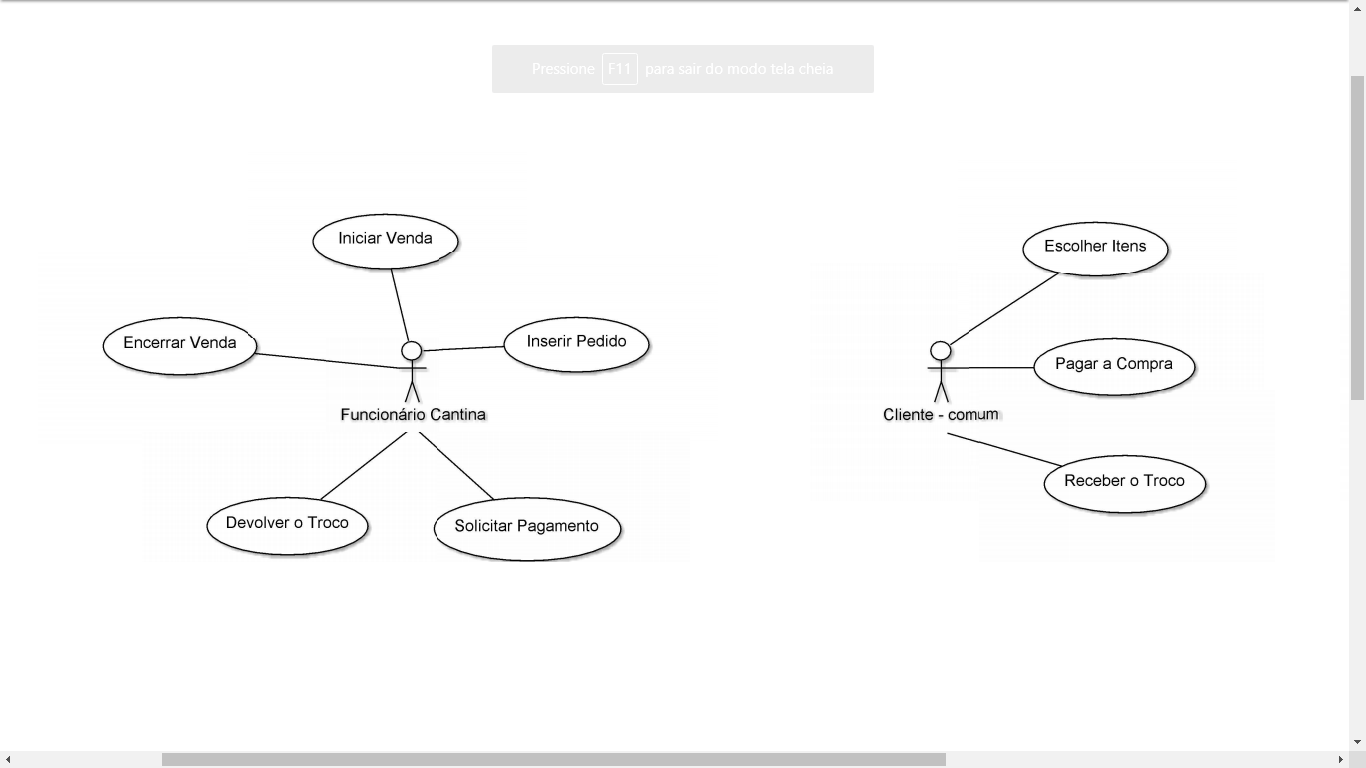


Imagem 1 - Casos de uso do ator funcionário da cantina

Fonte: Própria

**II. Cliente Comum**

Este ator é quem dá o lucro para a “empresa”, para o sistema, ele é responsável por: escolher itens, pagar a conta e receber o troco.

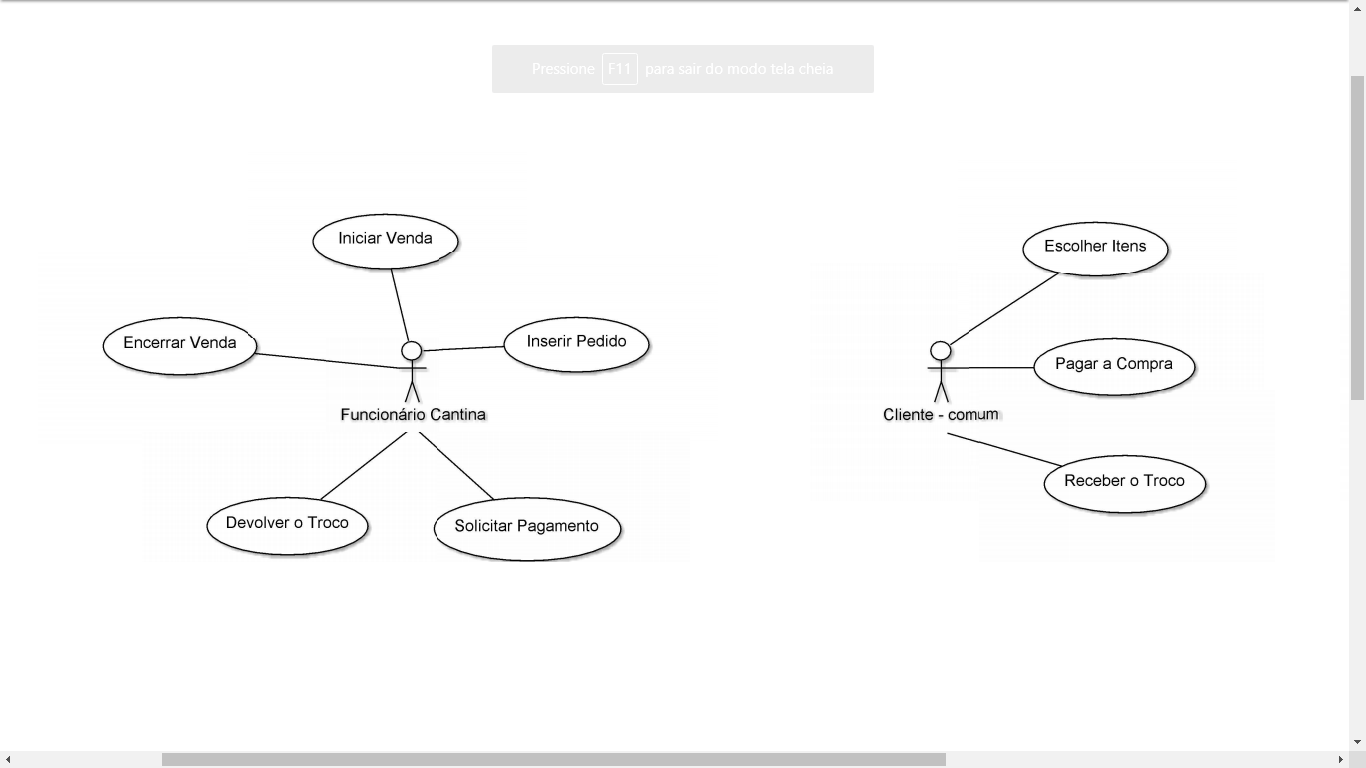


Imagem 2 - Casos de uso do ator cliente comum

Fonte: própria

**III. Cliente Funcionário**

Este ator é semelhante ao Cliente Comum, porém, o fato de ser funcionário da empresa lhe dá o privilégio de poder pagar sua dívida no fim do mês. Suas ações são: escolher itens, pagar a conta, receber o troco e marcar na conta.

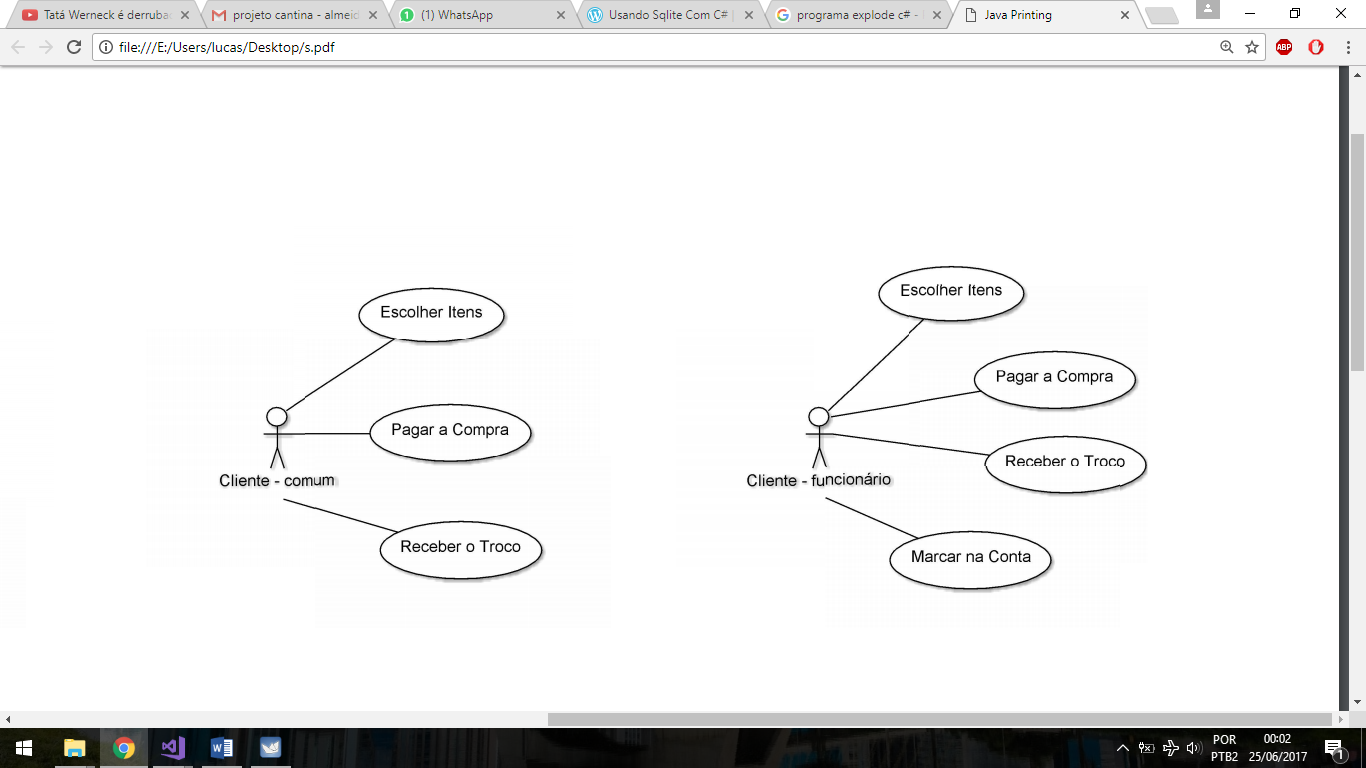


Imagem 3 - Casos de uso do ator cliente funcionário

Fonte: própria

**Janelas da aplicação**

O Sistema é composto por quatro janelas, sendo a principal usada para o controle de venda, e as secundárias para a consulta do débito pendente do cliente funcionário ou cadastramento do mesmo caso este ainda não o seja, adicionar novos produtos ao banco de dados e consultar a receita diária.

**Janela principal**

É composta por campos para inserir código do produto, sua quantidade e para mostrar uma prévia da nota fiscal; *labels* para definir os rótulos e receber nome e preço dos produtos; e, botões para confirmar produto, inserir novos produtos ao banco de dados, mostrar relatório diário de venda, marcar a conta e finalizar compra.

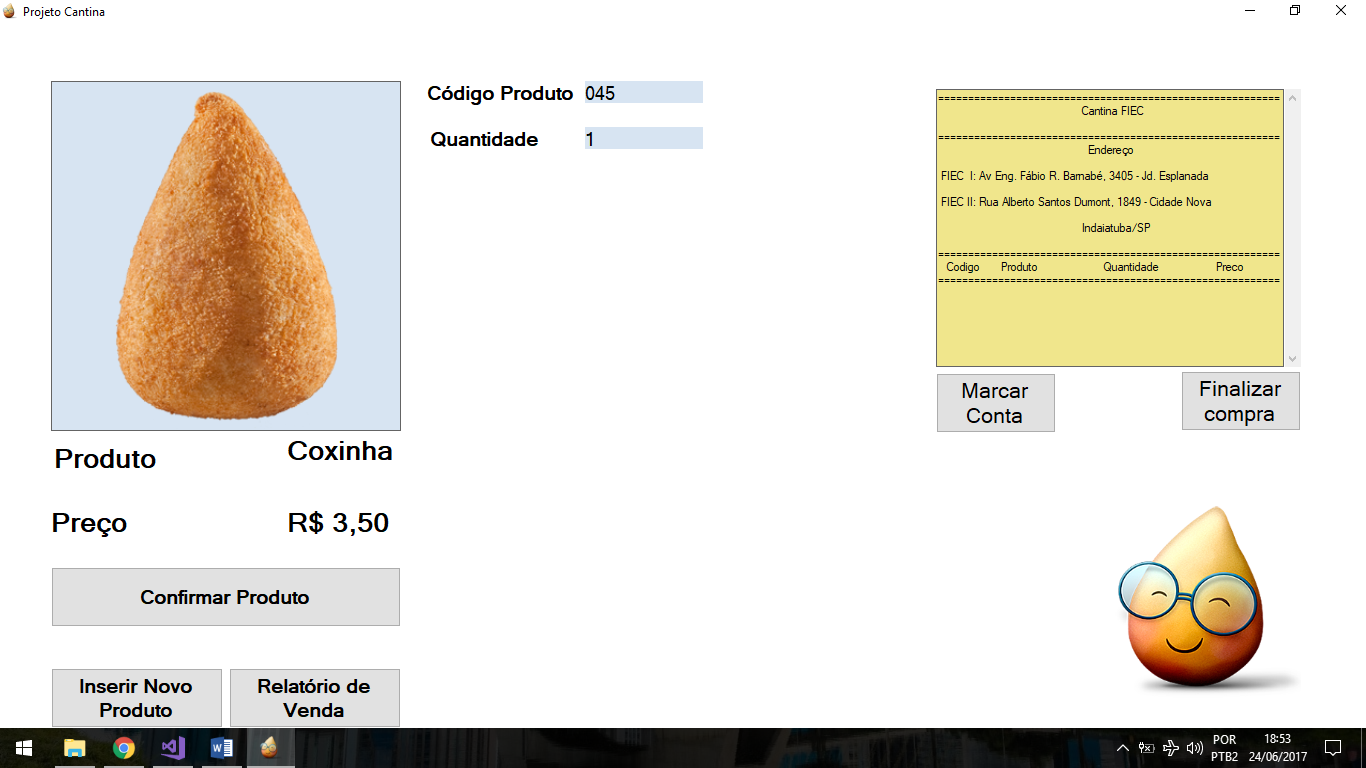


Imagem 4 - Interface da janela principal

Fonte: própria

**Janelas secundárias**

**I. Controle de débito (Conta do Usuário)**

É composta por *label* para rotulá-la, logo da empresa, campo para inserir o código do cliente e botões para buscar o funcionário, pagar a dívida, encerrar o resumo e sair (fecha a janela).

Nela o operador insere o número do cliente, caso este não possua uma conta ela é criada na hora e já é feita uma inserção no banco de dados a dívida do usuário com relação a primeira compra, caso este já possua uma conta a dívida anterior é somada com o valor da última compra e mostrada em tela.

Caso o cliente pague a conta, o operador fará isso no sistema que atualizará o banco de dados, definindo o campo “dívida Pendente” igual a 0.

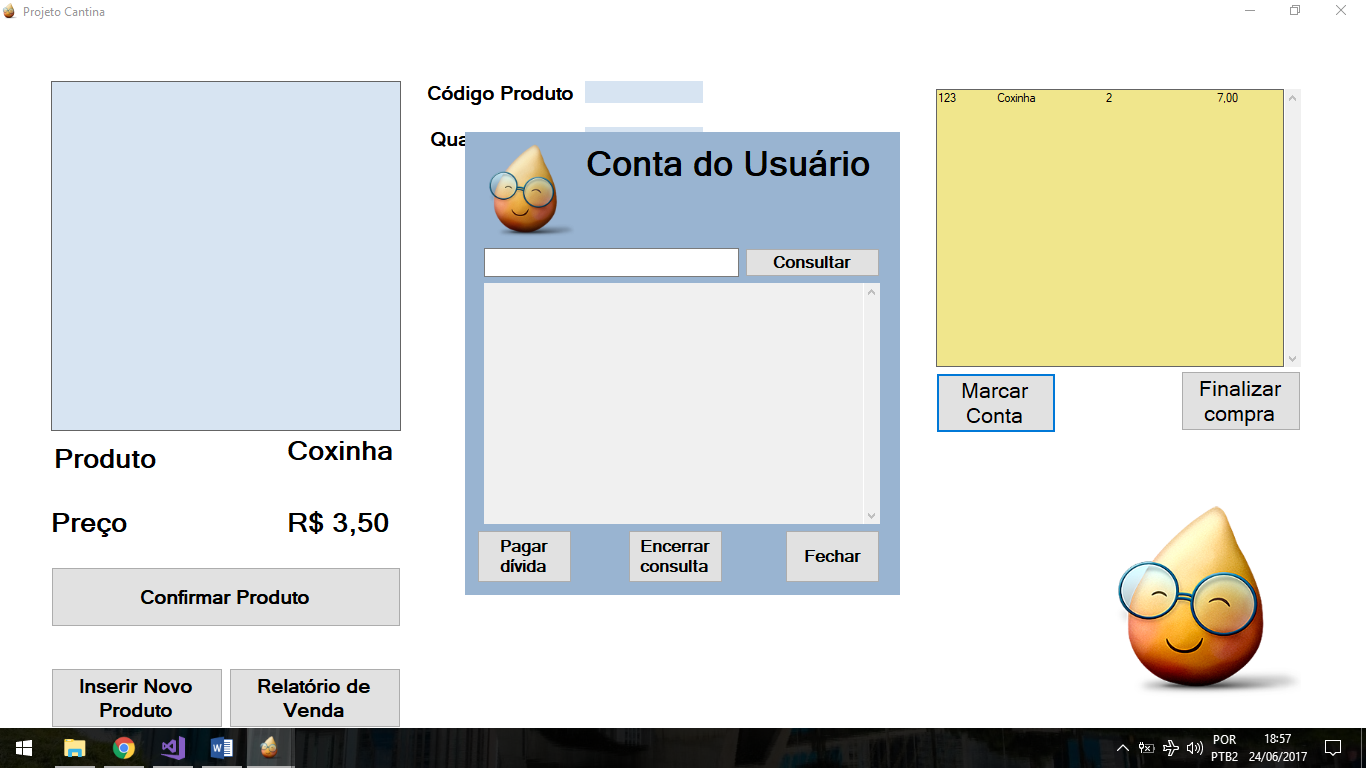


Imagem 5 - Interface da janela de contabilização da dívida do usuário

Fonte: própria

**II. Inserção de produtos**

É composta por *labels* que identificam a janela e os campos a serem preenchidos, estes por sua vez recebem os valores que serão adicionados ao banco de dados ao ser clicado o botão “confirmar produto”. O botão “cancelar limpa os campos preenchidos” e o botão sair (X) fecha o *form.*

Caso algum dos campos esteja com a sintaxe de preenchimento errada, uma mensagem de erro é mostrada, sendo assim o programa continua rodando sem explodir.

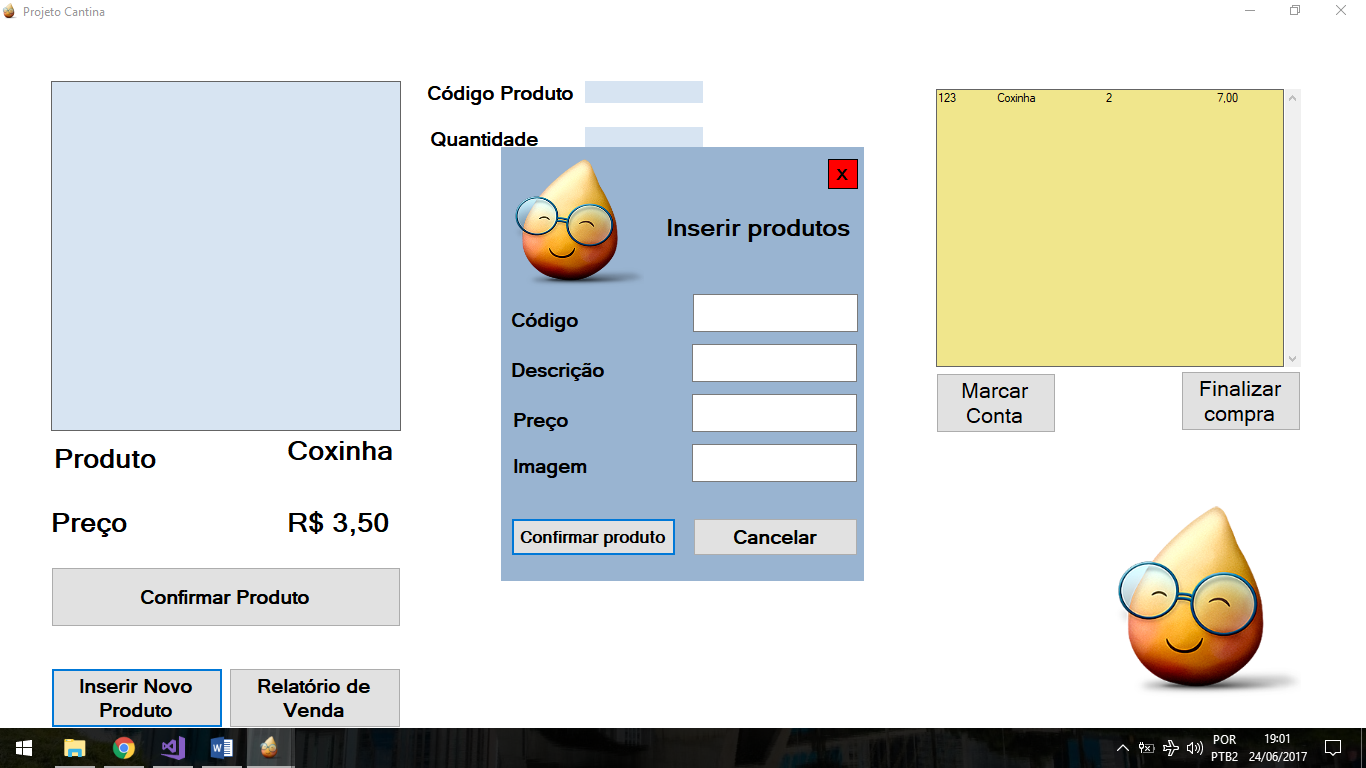


Imagem 6 – interface da anela de inserção de produtos

Fonte: própria

**III. Relatório de receita do dia**

Simples porém útil, este pequeno *form* é responsável por fazer uma consulta no banco de dados e retornar o lucro diário até o presente momento e possui um botão sair (X) para encerrá-lo após a consulta.

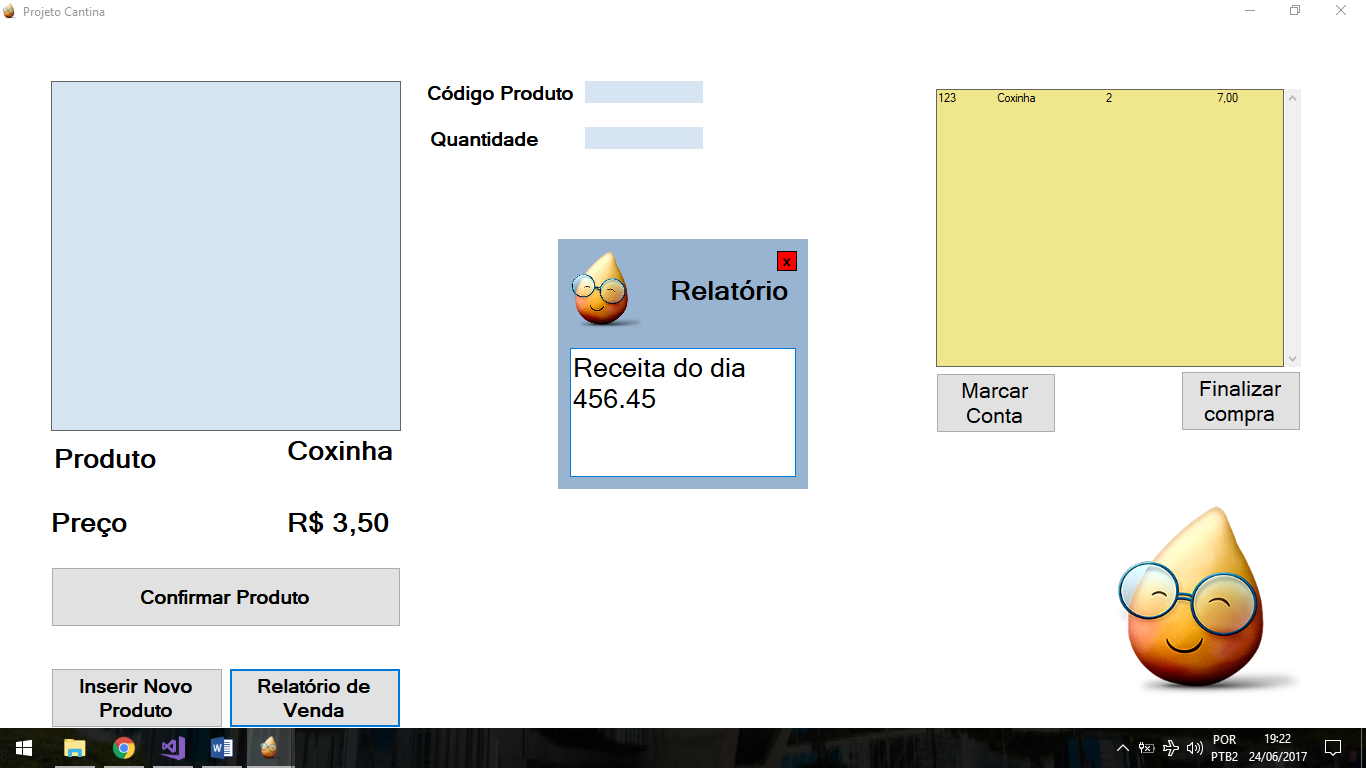


Imagem 7 - Interface da janela do relatório da receita diária

Fonte: própria

**Banco de dados da aplicação**

O banco de dados é uma parte oculta (isto é, não visível ao operador de caixa ou ao cliente) da aplicação, porém, sem este o software seria muito mais obsoleto, uma vez que os produtos disponíveis deveriam ser inseridos em arquivos de texto, que são mais suscetíveis à erros.

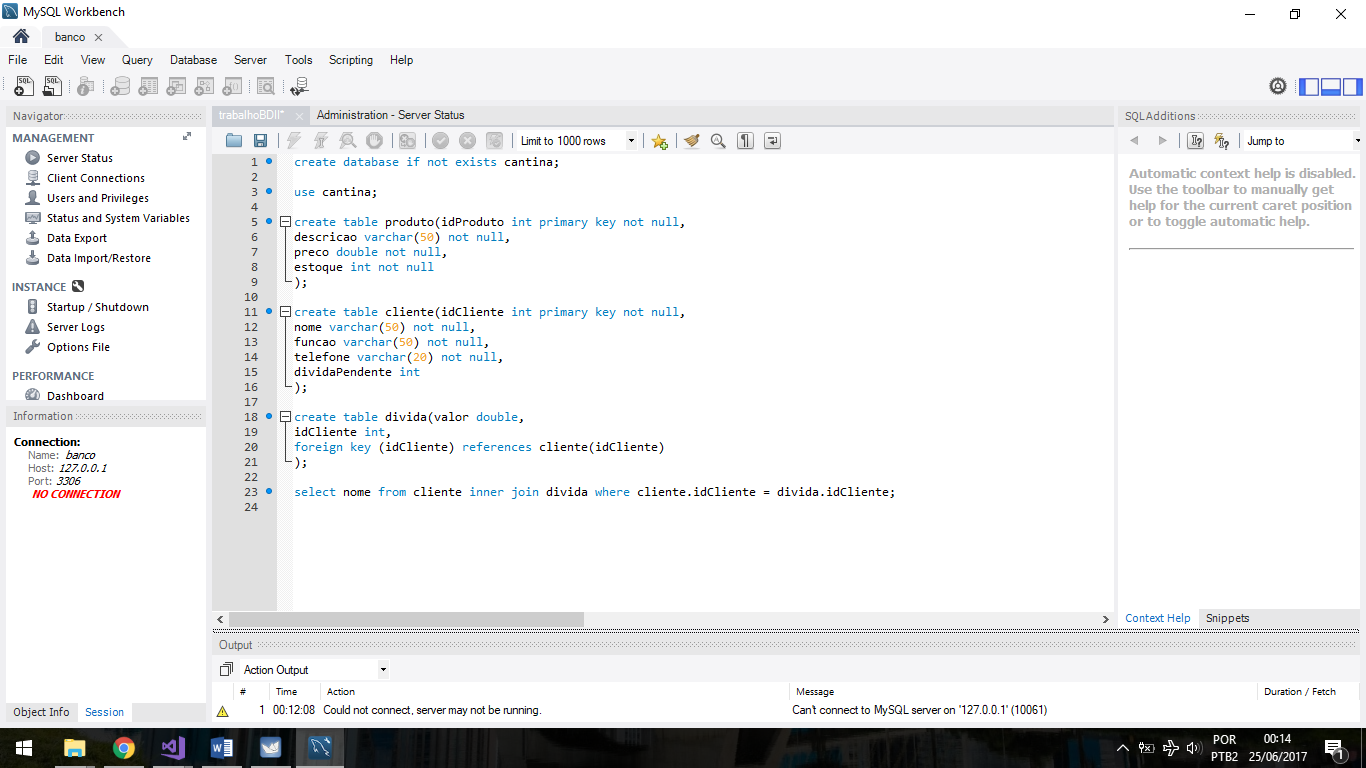


Imagem 8 - Sintaxe básica para a criação do banco a ser usado e selecioná-lo para a criação de tabelas e consultas posteriores

Fonte: própria

**Tabelas do Banco**

São as estruturas dentro do banco que armazenam os dados relativos às suas respectivas funções, isto é, a tabela de produtos, por exemplo, vai armazenar o código, descrição, preço e imagem deste.

**I. Produto**

Já dito anteriormente, esta tabela armazena os dados relacionado ao produto.

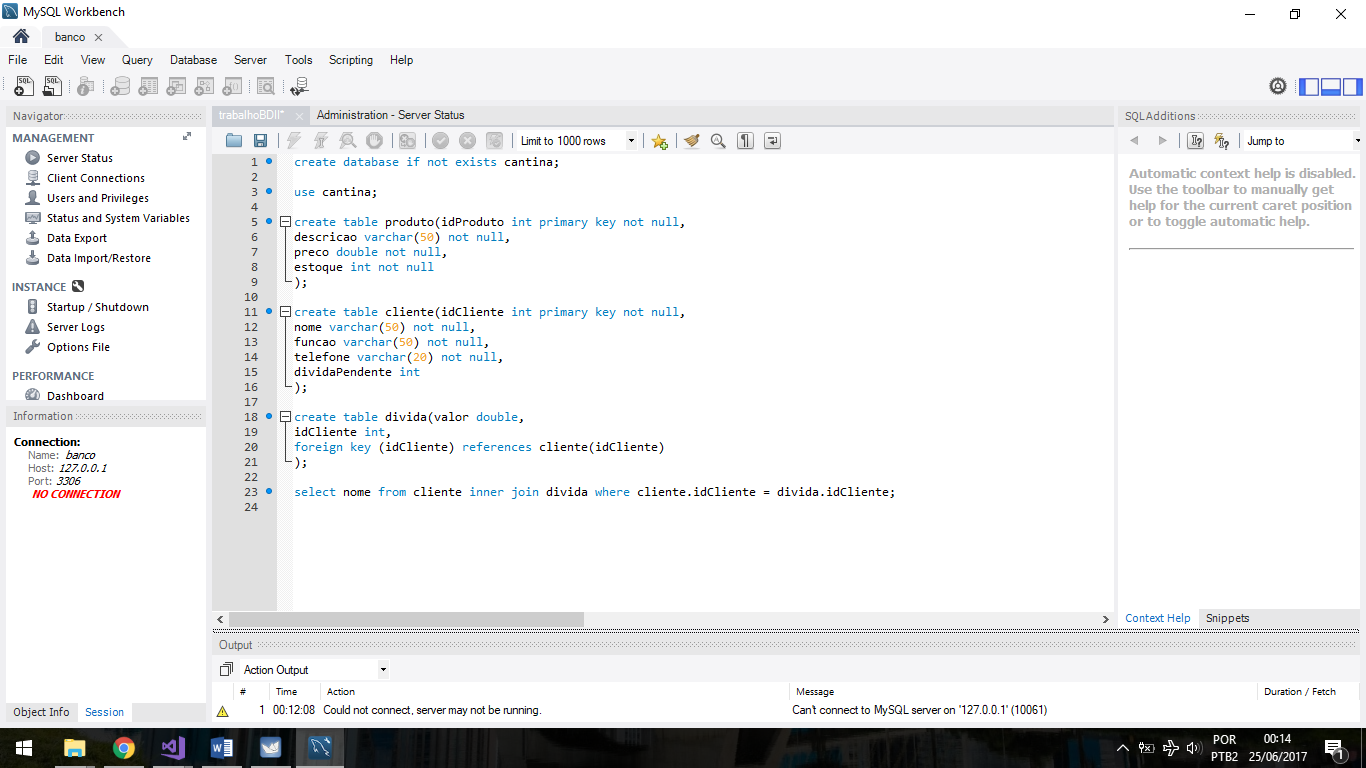


Imagem 9 - Sintaxe para a criação da tabela produto

Fonte: própria

**II. Cliente**

Armazena os atributos primordiais (código, nome, cargo, telefone e dívida pendente) para a criação da conta do cliente e um possível contato, caso necessário.

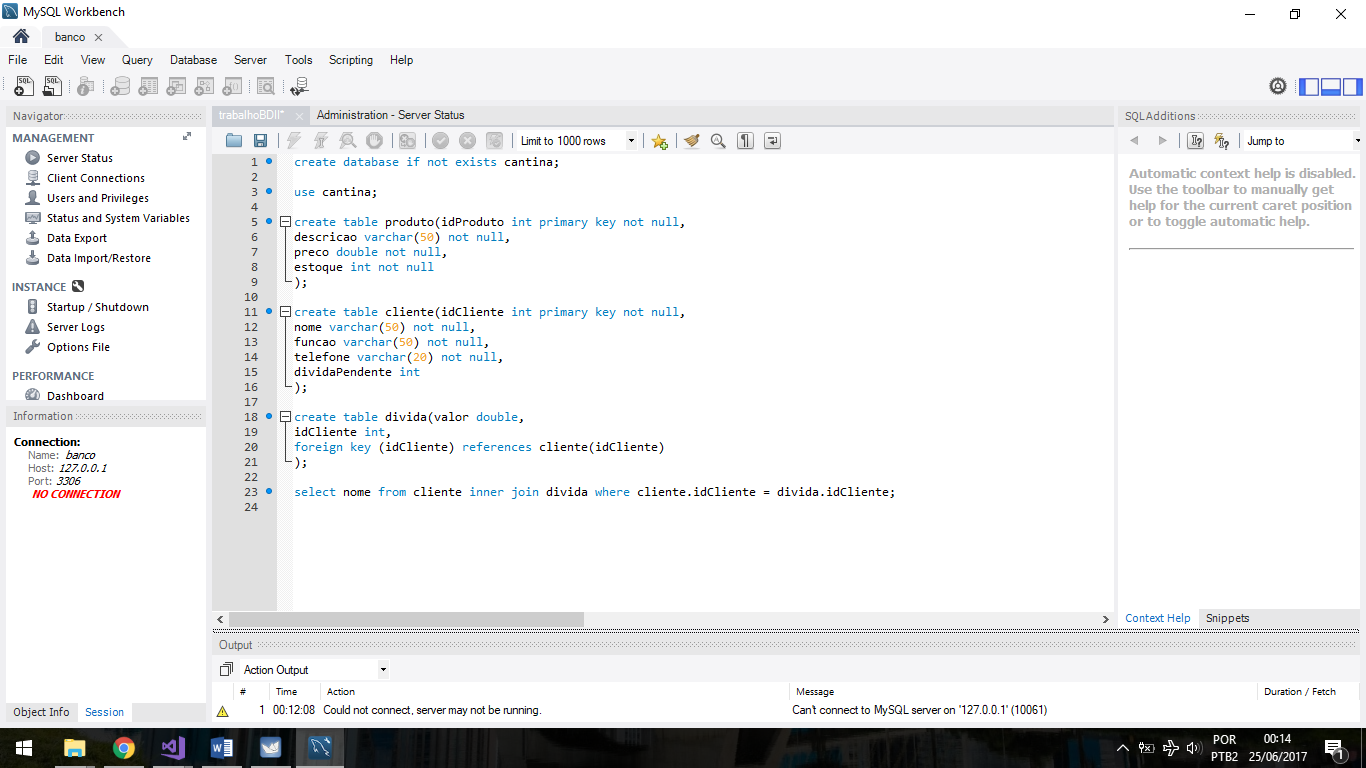


Imagem 10 - Sintaxe para a criação da tabela produto

Fonte: própria

**III. Dívida**

Tabela que “amarra” o cliente à sua dívida pendente.

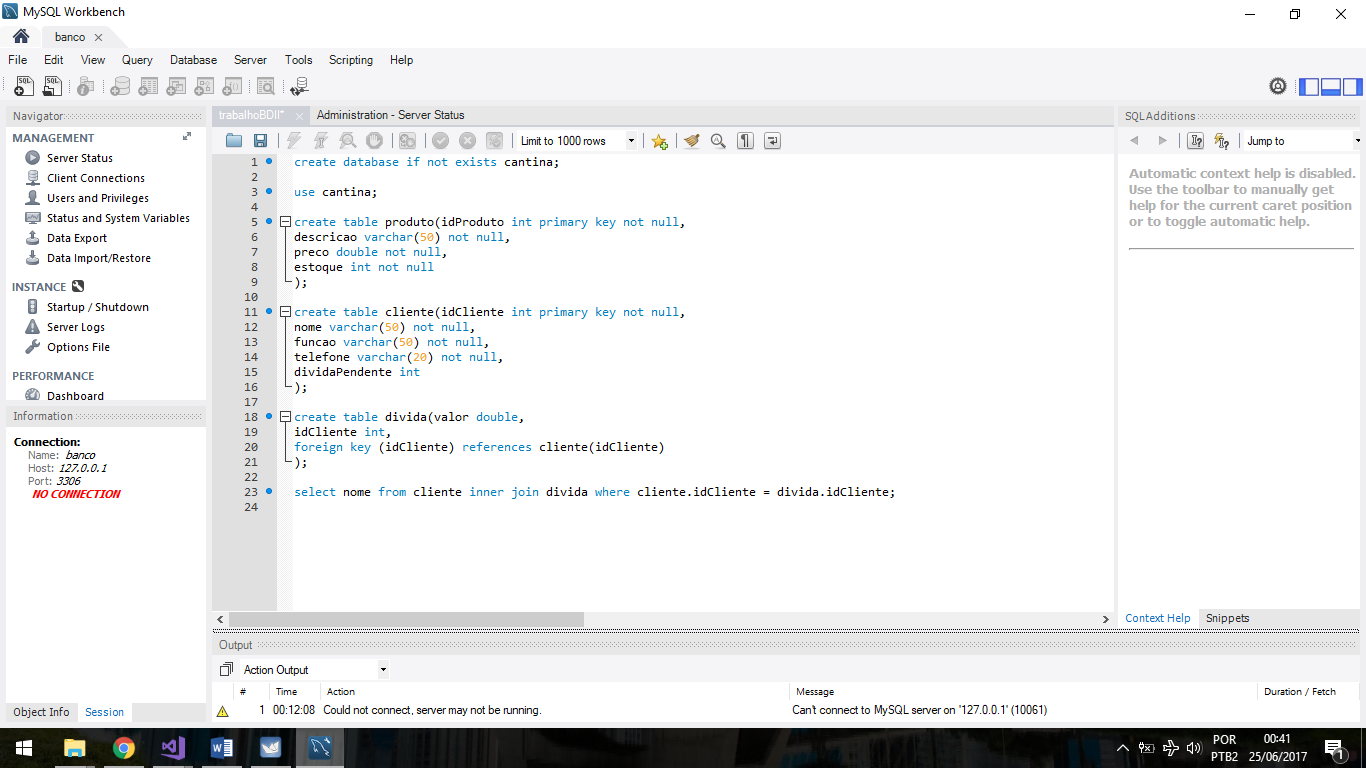


Imagem 11 - Sintaxe para a criação da tabela dívida

Fonte: própria

**IV. Compra**

Armazena dados das compras realizadas, isto é, o código, a data e o valor total.

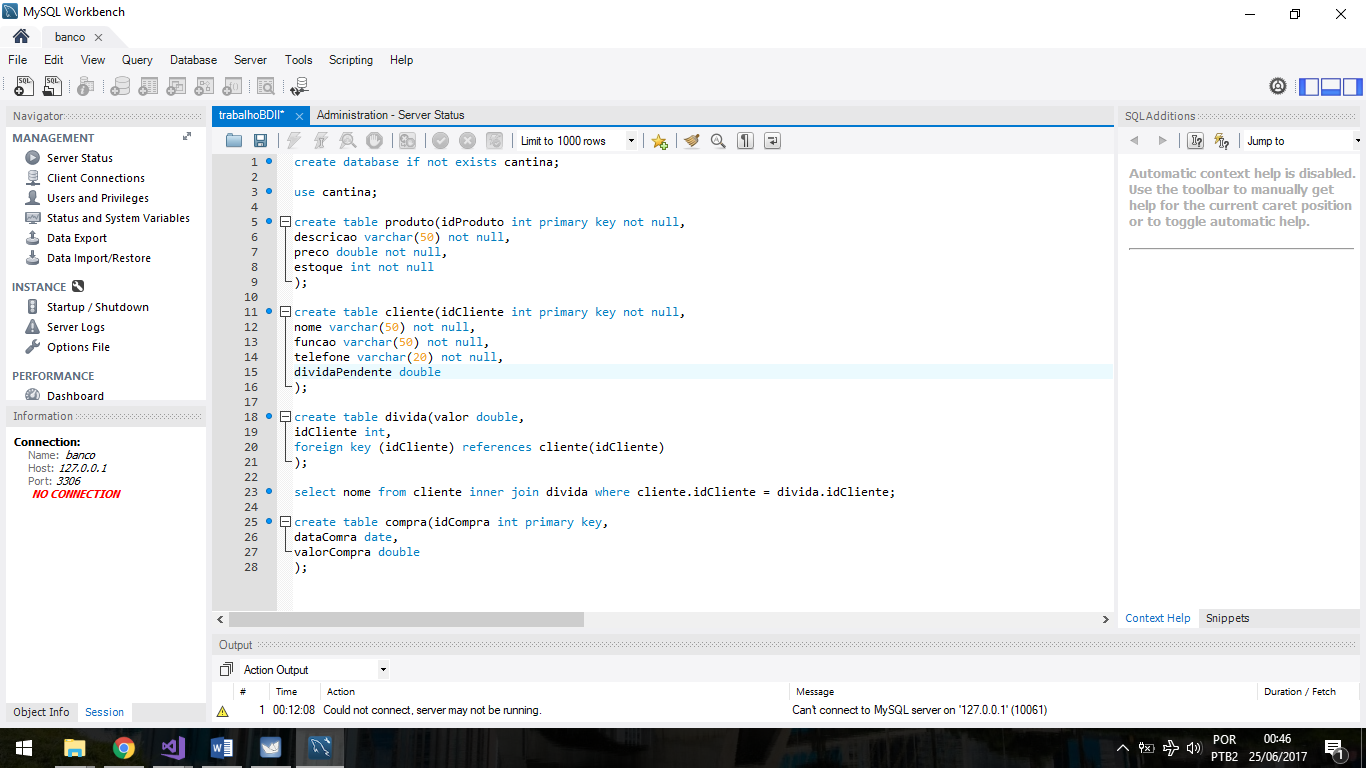


Imagem 12 - Sintaxe para a criação da tabela dívida

Fonte: própria

**Conclusão**

Ao fim desse projeto espera-se solucionar o problema apresentado na introdução que se dava pelo fato de o sistema da cantina não ser informatizado. Agora, se usado o *software* existe um controle organizado e seguro para o andamento do negócio.

**Referências**

MACORATTI. **C# - crud básico no sqlite com vs 2015**. Disponível em: <http://www.macoratti.net/16/10/c\_csqlite1.htm>. Acesso em: 19 jun. 2017.

MICROSOFT DEVELOPER NETWORK. **Como distinguir entre cliques e cliques duplos**. Disponível em: <https://msdn.microsoft.com/pt-br/library/ms171543(v=vs.110).aspx>. Acesso em: 22 jun. 2017.

MICROSOFT DEVELOPER NETWORK. **Joptionpane do java como e usado no c#?** Disponível em: <https://social.msdn.microsoft.com/forums/pt-br/cc4dd15a-5c1a-4b11-b533-c20b838b47fb/joptionpane-do-java-como-e-usado-no-c?forum=vscsharppt>. Acesso em: 24 jun. 2017.

SQLITE. **Sql features that sqlite does not implement**. Disponível em: <http://www.sqlite.org/omitted.html>. Acesso em: 23 jun. 2017.

WORDPRESS - RAFAEL RODRIGUES. **Usando sqlite com c#**. Disponível em: <https://rafaelrgi.wordpress.com/2008/01/30/usando-sqlite-com-c/>. Acesso em: 23 jun. 2017.