

UNIVERSIDADE FEDERAL RURAL DE PERNAMBUCO

EDGER DIJKSTRA

Um algoritmo eficiente para a solução do problema da
tela azul

GARANHUNS

2016

Edger Dijkstra

Um algoritmo eficiente para a solução do problema da tela azul

Monografia apresentada como requisito parcial para
obtenção do grau de Bacharel em Ciência da Compu-
tação da Unidade Acadêmica de Garanhuns da Uni-
versidade Federal Rural de Pernambuco.

Orientadores:

Alan Turing

John von Neumann

Garanhuns, 7 de julho de 2016

Monografia apresentada como requisito parcial para obtenção do grau de Bacharel em
Ciência da Computação da Unidade Acadêmica de Garanhuns da Universidade Federal
Rural de Pernambuco, aprovada pela comissão examinadora que abaixo assina.

Alan Turing - Orientador
UAG
UFRPE

John von Neumann - Orientador
UAG
UFRPE

John Hopcroft - Examinador
UAG
UFRPE

Richard Karp - Examinador
UAG
UFRPE

Stephen Cook - Examinador
UAG
UFRPE

John Backus - Examinador
UAG
UFRPE

Garanhuns, 7 de julho de 2016

Resumo

Neste relatório é apresentado o SUPER-System (nome provisório), um sistema de automação de comanda para uma loja de conveniência. Com o SUPER-System, o cliente realiza um autoatendimento através de um cartão comanda, com o qual ele insere os produtos consumidos sem o auxílio de um funcionário. Ao fim do processo, o cliente solicita ao operador de caixa que feche sua comanda para que o pagamento possa ser realizado. Dentre os benefícios do sistema, destacam-se: praticidade, agilidade e liberdade para o cliente. A aplicação foi implementada na linguagem Java juntamente com JavaFX e PostgreSQL. Além disso, técnicas de Engenharia de *Software* e de Banco de Dados e uma ferramenta de controle de versão foram utilizadas para um melhor desenvolvimento do projeto. Durante o processo de modelagem, reuniões com clientes e visitas ao estabelecimento possibilitaram a criação de requisitos, atendendo às necessidades dos clientes. Foram realizados teste unitários que indicaram um bom funcionamento do sistema, juntamente com uma confiabilidade no quesito de segurança. Por fim, todos os requisitos propostos foram atendidos e o sistema encontra-se em funcionamento, passando por teste de usabilidade.

Palavras-chave: Automação. Agilidade. Auto atendimento

Abstract

Edsger Wybe Dijkstra (11 May 1930 – 6 August 2002) was a Dutch computer scientist. A theoretical physicist by training, he worked as a programmer at the Mathematisch Centrum (Amsterdam) from 1952 to 1962. He was a professor of mathematics at the Eindhoven University of Technology (1962–1984) and a research fellow at the Burroughs Corporation (1973–1984). He held the Schlumberger Centennial Chair in Computer Sciences at the University of Texas at Austin from 1984 until 1999, and retired as Professor Emeritus in 1999.

One of the most influential members of computing science's founding generation, Dijkstra helped shape the new discipline from both an engineering and a theoretical perspective. Many of his papers are the source of new research areas. Several concepts and problems that are now standard in computer science were first identified by Dijkstra and/or bear names coined by him.

Keywords: keyword1, keyword2, keyword3.

Sumário

Lista de Figuras	iv
Lista de Tabelas	v
1 Introdução	1
1.1 Objetivo	1
1.2 Justificativa	1
1.3 Organização do relatório	2
2 Fundamentação teórica	3
2.1 Análise de requisitos	3
2.2 Modelagem de dados	3
2.3 Desenvolvimento	4
3 Atividades Desenvolvidas	5
3.1 Análise de requisitos	5
3.2 Modelagem do sistema	6
4 O Sistema	8
5 Conclusões	9
A Apêndice	10
Bibliografia	11

Lista de Figuras

3.1	Diagrama de caso de uso do cliente do estabelecimento.	6
3.2	Diagrama de caso de uso do operador administrador.	6
3.3	Diagrama de classes.	7

Lista de Tabelas

3.1	Requisitos do Funcionais do sistema	5
3.2	Requisitos Não-Funcionais do sistema	6

Capítulo 1

Introdução

Parte indispensável na formação profissional, o Estágio Supervisionado Obrigatório (ESO), nos permite vivenciar momentos que nos fazem ter uma visão real do dia a dia de uma empresa de tecnologia, tendo contato com rotinas reais da fabricação de *software*, desde a análise de requisitos, passando pela escolha da tecnologia que será utilizada e melhor se encaixa com as necessidades do cliente, até a concepção prática do *software*, vendo em loco as dificuldades reais enfrentadas no decorrer do projeto. Tudo isto visando um preparo além da teoria da sala de aula, preparando melhor o aluno para o mercado de trabalho.

1.1 Objetivo

O objetivo primordial deste estágio foi desenvolver um sistema para automatizar o sistema.

1.2 Justificativa

A execução deste estágio foi de grande valia para agregar uma experiência de mercado de trabalho aos conhecimentos já adquiridos pelo estagiário, dando-lhe um *know-how* para enfrentar dificuldades e desafios inerentes ao processo de concepção de um software, dando-lhe também a experiência de estar inserido em uma equipe de profissionais, vivenciando o trabalho em equipe por um mesmo objetivo final.

Do ponto de vista da empresa, o estágio foi útil ao suprir uma demanda recém chegada, que foi o desenvolvimento deste sistema, já que os demais profissionais da empresa estavam envolvidos em outros projetos e este projeto poderia ter seu tempo de concepção alongada, trazendo risco aos prazos previamente definidos.

1.3 Organização do relatório

A organização deste relatório segue da seguinte forma: o Capítulo 2 apresenta a fundamentação teórica dos conceitos utilizados nas atividades desenvolvidas; o Capítulo 3 apresenta o detalhamento das atividades desenvolvidas; o Capítulo 4 descreve o sistema **AutoCom** e, por fim, o Capítulo 5 mostra a conclusão do trabalho realizado e os resultados obtidos.

Capítulo 2

Fundamentação teórica

Nesta seção são apresentados conceitos utilizados no desenvolvimento deste projeto e suas respectivas justificativas para o uso dos mesmos.

2.1 Análise de requisitos

Dentre as muitas partes componente da Engenharia de Software, uma que conhecemos e traz grande contribuição para o projeto do *software* é a Engenharia de Requisitos que fornece o mecanismo apropriado para entender o que o cliente deseja, analisando as necessidades, avaliando a exequibilidade, negociando uma condição razoável, especificando a solução de modo não ambíguo, validando a especificação e gerindo os requisitos [3].

Dentre os requisitos podemos dividi-los em Requisitos Funcionais (RF) que são aqueles que definem recursos específicos do sistema, as funções de um sistema e seus componentes, podendo ser cálculos, comportamentos, manipulação de dados dentre outras. Já os Requisitos Não-Funcionais (RNF) são aqueles que dizem respeito ao uso do sistema em termos de desempenho, usabilidade, confiabilidade, segurança dentre outros, podendo estes, se não observados da forma correta, causar o mau funcionamento do sistema ou até sua inutilização [4].

2.2 Modelagem de dados

Tanto quanto outras partes do sistema, o Banco de Dados é algo que também precisa ser bem pensado e modelado para que sejam evitados erros e ambiguidade de informações. De acordo com Elmasri & Navathe [2], modelo de dados é uma coleção de conceitos que podem ser usados para descrever a estrutura de um banco de dados.

Um modelo que traz uma ideia de alto nível é o modelo Entidade Relacionamento (ER). Juntamente com suas variações, o modelo ER é normalmente empregado para projeto conceitual de aplicações de um banco de dados [2]. Neste modelo as entidades

podem ser um objeto com uma existência física, como por exemplo uma pessoa, uma empresa ou um produto.

2.3 Desenvolvimento

Por ser Java uma linguagem portátil e facilmente integrável, foi escolhida para este projeto. Segundo Deitel & Deitel [1], ao contrário da linguagem de máquina, que é dependente do *hardware* específico, os *bytecodes* são independentes de plataforma, eles não dependem de uma plataforma de *hardware* particular. Portanto, os *bytecodes* do Java são portáteis sem recompilar o código-fonte.

Algumas bibliotecas foram utilizadas para auxiliar algumas partes do desenvolvimento do projeto como:

- PostgreSQL para a comunicação do sistema com o banco de dados;
- iText para a geração e formatação dos relatórios em pdf.

Capítulo 3

Atividades Desenvolvidas

As atividades que seguem descritas envolvem tudo o que foi desenvolvido na MBA Sistemas, com vistas a desenvolver uma ideia vinda da própria empresa.

3.1 Análise de requisitos

A princípio a análise de requisitos do sistema foi, em parte, feita antes do início do estágio, sendo primordialmente apresentada logo que foi tido o conhecimento do trabalho a ser desenvolvido. Houve a disponibilidade de ir a uma loja de conveniência, local para o qual estaria sendo desenvolvido o sistema, com objetivo de uma conversa informal, na qual foram captados alguns requisitos que não haviam sido contemplados na primeira análise. Após a visita, foram feitas reuniões nas quais deram origem aos requisitos que ajudaram na concepção do sistema.

Os requisitos funcionais (RF) e requisitos não-funcionais (RNF) seguem como na Tabela 3.1 e Tabela 3.2 (respectivamente).

Tabela 3.1: Requisitos do Funcionais do sistema

ID	Nome	Prioridade
RF01	Id e Senha de usuário	Essencial
RF02	GUI do Sistema	Essencial
RF03	Validação do usuário	Essencial
RF04	Verificação da comanda utilizada	Essencial
RF05	Verificação do produto	Essencial
RF06	Verificação do formato do CPF/CNPJ	Essencial

No decorrer do projeto surgiram outras histórias de usuário, já que existe a necessidade de outras funcionalidades, como por exemplo, a necessidades de controle de acesso a certas áreas do sistema, que não seriam disponíveis a todos os operadores.

Tabela 3.2: Requisitos Não-Funcionais do sistema

ID	Nome	Prioridade
RNF01	Mascarada de campo numérico	Desejável
RNF02	Mascara de CPF/CNPJ	Desejável
RNF03	Mascara de campos monetários	Desejável
RNF04	Uso de leitor de código de barras	Essencial
RNF05	Uso de impressora não fiscal	Essencial

3.2 Modelagem do sistema

Após a busca dos requisitos e a definição das histórias de usuário, foi feita a criação de alguns documentos do sistema com o objetivo de nortear sua construção. Dentre esses documentos, foram gerados os diagramas de caso de uso, para um melhor entendimento do que cada usuário pode fazer no sistema. Dois dos diagramas de caso de uso gerados foram os que são mostrados nas Figuras 3.1 e 3.2.

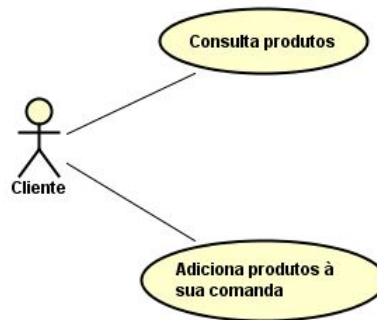


Figura 3.1: Diagrama de caso de uso do cliente do estabelecimento.

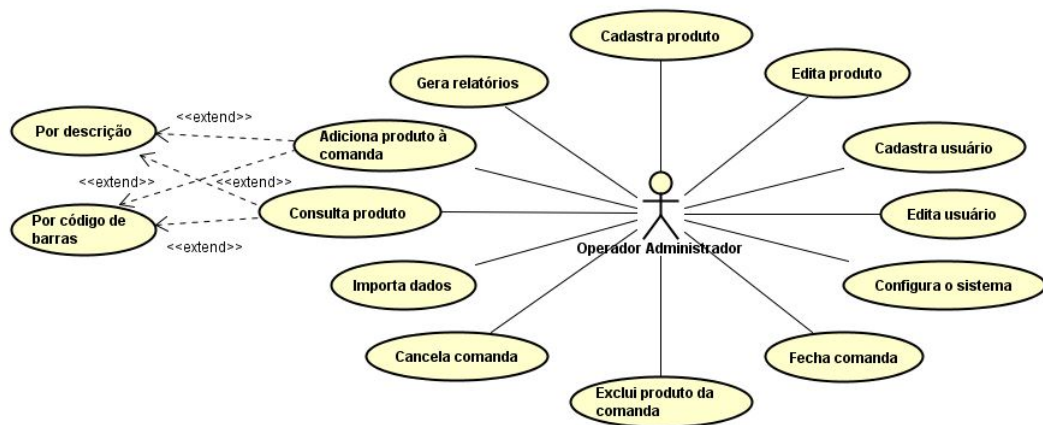


Figura 3.2: Diagrama de caso de uso do operador administrador.

Outro diagrama gerado foi o diagrama de classes, para que se tivesse uma visão geral das classes que seriam geradas e das dependências entre elas. A Figura 3.3 mostra um

diagrama com uma visão geral das classes, não especificando os atributos e funções das classes.

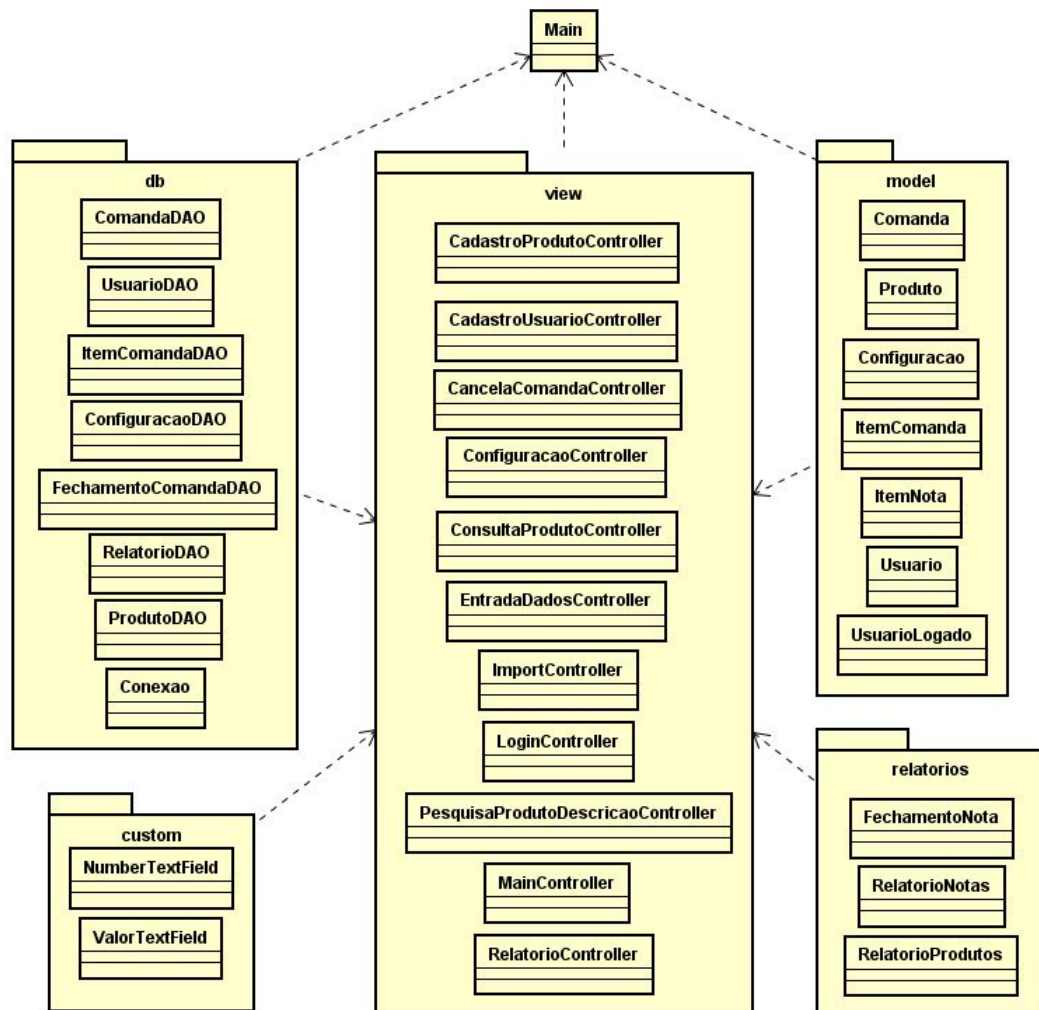


Figura 3.3: Diagrama de classes com uma visão mais geral, sem especificação de atributos e funções das classes.

Capítulo 4

O Sistema

O sistema NomeSistema (nome provisório) é um sistema que busca agilizar o atendimento de clientes em uma loja.

O sistema vem com a proposta de automatizar este processo de “anotação” das comandas, através de um sistema fácil de operar e que precisa do mínimo de intervenção do operador durante sua utilização.

Capítulo 5

Conclusões

Ao final do período do estágio, concluindo todas as atividades propostas, mesmo vivenciando todo o dia a dia dentro de uma empresa de *software* e percebendo as dificuldades que surgem durante o percurso, cumprimos os prazos estabelecidos.

O conhecimento e experiência que o estágio pôde me proporcionar serão de grande utilidade na minha vida profissional, pois me deram a real noção do que é enfrentar um projeto real, cumprir prazos, atender a requisitos reais do cliente, lidar com os riscos do projeto, buscar conhecimento para atender a uma demanda do projeto, utilizar tudo aquilo que foi vivenciado na teoria, buscando os melhores caminhos e os menores custos computacionais, discutindo junto com o restante da equipe a melhor forma de atender os requisitos demandados pelo cliente.

Apêndice A

Apêndice

Este é o Apêndice.

Bibliografia

- [1] DEITEL, P. ; DEITEL, H. *Java como programar*. 8th. Pearson, 2009.
- [2] ELMASRI, R. ; NAVATHE, S. B. *Sistemas de banco de dados*. 6th. Pearson, 2011.
- [3] PRESSMAN, R. S. *Engenharia de software: uma abordagem profissional*. 6th. Porto Alegre : Bookman, 2009.
- [4] SOMMERVILLE, I. *Engenharia de software*. 8th. São Paulo : Pearson, 2007.