**RESTAURANTE.NET**

**André Luiz Magalhães, Edson Oliveira dos Santos, Thiago Roberto da Silva Menezes, Rangel Soares de Souza, Lucas Roberto dos Santos, Ivan Paulovich**

**Antônio da Mota Moura Junior** (Orientador)

Centro Universitário de Belo Horizonte, Belo Horizonte, MG

*Resumo: A crescente informatização das mais diversas atividades transforma a tecnologia da informação, em uma área cada vez mais relevante economicamente. A expansão levou à especialização e, atualmente, é possível encontrar várias subáreas de TI dedicadas a tarefas específicas – e que demandam profissionais com conhecimentos igualmente aprofundados. Tendo em vista o grande crescimento no uso de tecnologias interligadas entre si, e com o e-commerce se tornando um fator determinante na internet, este artigo propõe um sistema integrado de compra de produtos alimentícios por meio do uso da internet. Um sistema no qual tem o fim de facilitar a venda deste tipo de produto, sem que haja a necessidade que o comprador busque a mercadoria pessoalmente, mas sim, que acompanhe o pedido e a receba em sua residência, de foram rápida, sem maiores problemas.*

*ABSTRACT: The increasing computerization of various activities transforms information technology in an increasingly economically relevant area. The expansion led to specialization and currently one can find various subfields of IT dedicated to specific tasks - and also require professionals with in-depth knowledge. Given the increasing use of interconnected technologies and e-commerce becoming a factor in the internet, this paper proposes an integrated system for buying food through the use of internet products. A system in which it has to facilitate the sale of this type of product, without the need to search for the buyer the goods personally, but yes, accompanying the order and receive in your residence, were quick, with no major problems.*

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**1 Introdução**

Vamos falar um pouco de Revolução Digital, o que significa Revolução? Revoluções implicam, por sua própria essência, em transformações sensíveis que reestruturam paradigmas das mais diversas naturezas. A evolução humana é formada por revoluções e a história nos mostra que os indivíduos e organizações que progridem têm sucesso e evoluem são aqueles que mais rapidamente se adaptam às mudanças. [Gabriel Martha, educar – 1 ed., 2013].

Se isso é fato para qualquer tipo de revolução, ele torna-se ainda mais acentuado quando analisamos a Revolução Digital. Nunca, em nenhuma outra época da história da humanidade, a velocidade de mudança foi tão rápida. Estamos vivendo um “momento de vertigem” inédito que requer um grau de adaptabilidade inédito também. A plasticidade do nosso cérebro permitiu a nossa evolução. A nossa adaptabilidade digital como indivíduos permitirá nosso sucesso na era digital. [Gabriel Martha, educar – 1 ed., 2013].

Devido à grande necessidade em que vivemos encontramos no mercado uma grande dificuldade em comprar alimentos on-line e em tempo mais preciso. Assim então reunimos a equipe e decidimos que desenvolver um produto que facilite essa compra on-line, deixando a compra de uma forma bem mais simples e eficaz para o cliente.

O produto consiste em um sistema on-line em que o usuário por um meio de login, consegue consultar produtos cadastrados, comparar preços entre produtos similares e até comentários de outros clientes que adquiriram os produtos que esteja interesse de comprar.

Os produtos cadastrados no sistema em que poderá encontrar inicialmente será:

● Sanduiche;

● Pizza;

● Doces: Açaí / Sorvete;

Após a compra do produto o cliente poderá acompanhar o status de criação e entrega do seu produto. Em seguida o cliente irá fazer uma análise da compra dando uma nota para alguns itens de compra e entrega do produto, e até mesmo se o produto é bom ou ruim.

**2 Fundamentos teóricos / Referencial Teórico**

Após estudado com a equipe, analisamos algumas plataforma única para desenvolvimento e execução de sistemas e aplicações e concluímos que .NET seria mais viável para nosso projeto. Assim falaremos um pouco sobre essa ferramenta.

A plataforma .NET é executada sobre uma Common Language Runtime - CLR (Ambiente de Execução Independente de Linguagem) interagindo com um Conjunto de Bibliotecas Unificadas (framework). Esta CLR é capaz de executar, atualmente, mais de 33 diferentes linguagens de programação, interagindo entre si como se fossem uma única linguagem. Mas nós utilizaremos “C#.

Esta plataforma permite a execução, construção e desenvolvimento de Web Services (Aplicações Web) de forma integrada e unificada.

A plataforma .NET baseia-se em um dos principios utilizados na tecnologia Java (Just In Time Compiler - JIT), os programas desenvolvidos para ela são duplo-compilados (compilados duas vezes), uma na distribuição (gerando um código que é conhecido como "bytecodes") e outra na execução.

Um programa é escrito em qualquer das mais de trinta e três linguagens de programação disponíveis para a plataforma, o código fonte gerado pelo programador é então compilado pela linguagem escolhida gerando um código intermediário em uma linguagem chamada MSIL (Microsoft Intermediate Language).

Este novo código fonte gera um arquivo na linguagem de baixo nível Assembly, de acordo com o tipo de projeto:

· EXE - Arquivos Executáveis, Programas

· DLL - Biblioteca de Funções

· ASPX - Página Web

· ASMX - Web Service

No momento da execução do programa ele é novamente compilado, desta vez pelo compilador JIT, de acordo com a utilização do programa, por exemplo: Temos um Web Site desenvolvido em ASP.NET, ao entrar pela primeira vez em uma página o JIT irá compila-la, nas outras vezes que algum outro usuário acessar esta página, ele usará esta compilação.

Também é possível, através de ferramentas específicas, "pré-compilar" o código para que não se tenha o custo da compilação JIT durante a execução.

Para gerenciar o sistema usaremos o Windows Server 2008 R2 que é um servidor multifuncional desenvolvido para aumentar a confiabilidade e a flexibilidade da sua infraestrutura de servidor ou nuvem privada, ajudando você a poupar tempo e reduzir custos. Ele oferece ferramentas avançadas para reagir às necessidades da empresa com mais rapidez do que nunca e maior controle e segurança.

A versão mais recente do Service Pack 1 (SP1) para o Windows Server 2008 R2 fornece muitos aprimoramentos para você implementar um plano robusto de virtualização de datacenters e desktops. Se você quiser consolidar servidores, criar uma nuvem privada ou oferecer VDI (Virtual Desktop Infrastructure), os recursos do Windows Server 2008 R2 permitirão que você eleve sua estratégia de datacenter e desktop a outro patamar.

Para gerenciar nosso banco de dados usaremos o MS SQL Server é um SGBD - sistema gerenciador de Banco de dados relacional desenvolvido pela Microsoft.

Como um Banco de dados, é um produto de software cuja principal função é a de armazenar e recuperar dados solicitados por outras aplicações de software, seja aqueles no mesmo computador ou aqueles em execução em outro computador através de uma rede (incluindo a Internet). Há pelo menos uma dúzia de diferentes edições do Microsoft SQL Server destinadas a públicos diferentes e para diferentes cargas de trabalho (variando de pequenas aplicações que armazenam e recuperam dados no mesmo computador, a milhões de usuários e computadores que acessam grandes quantidades de dados a partir da Internet ao mesmo tempo). Suas linguagens de consulta primárias são T-SQL e ANSI SQL.

Com a cloud computing, muitos aplicativos, assim como arquivos e outros dados relacionados, não precisam mais estar instalados ou armazenados no computador do usuário ou em um servidor próximo.

Este conteúdo passa a ficar disponível nas nuvens, isto é, na internet. Ao fornecedor da aplicação cabe todas as tarefas de desenvolvimento, armazenamento, manutenção, atualização, backup, escalonamento, etc. O usuário não precisa se preocupar com nenhum destes aspectos, apenas com acessar e utilizar.

Geocodificação é o processo de conversão de endereços (por exemplo, "1600 Amphitheatre Parkway, Mountain View, CA") em coordenadas geográficas (por exemplo, latitude 37.423021 e longitude -122.083739), que podem ser usadas para colocar marcadores ou posicionar o mapa. A Google Geocoding API fornece um forma direta de acessar um geocodificador por meio de uma solicitação HTTP. Além disso, o serviço permite realizar a operação inversa (transformando coordenadas em endereços). Esse processo é conhecido como "geocodificação reversa".

O processo de converter um endereço em um par de latitude/longitude é chamado endereço geocodificação. Inúmeros serviços on-line (incluindo MapPoint, Virtual Earth e Google Terra) fornecendo geocodificação como um serviço. Para converter um endereço nos EUA para um ponto, você pode encapsular uma chamada para a geocodificação MapPoint Web Services em uma função do SQLCLR.

Mas onde você deve armazenar a latitude/longitude dentro de seu banco de dados do SQL Server?

SQL Server 2008 vem com os dois tipos de dados necessários para armazenar dados espaciais: geometria e Geografia. Os dois tipos de dados são implementados usando a arquitetura .NET para tipos definidos pelo usuário, que significa que eles podem ter propriedades e métodos. Vamos continuar com o tipo de dados de Geografia agora porque isso mapeia mais de perto para o problema. Isso possibilita com que consiga ordenar uma lista de pontos por distancia crescente a partir de um referencial geográfico.

**3 Objetivos**

**3.1 Objetivo Geral**

Criar um sistema online para venda de produtos alimentícios via internet.

**3.2 Objetivos Específicos**

* Identificar para o cliente, o local mais perto de sua residência para a compra do seu produto através de pontos geográficos.
* Analisar os dados das compras realizadas e apresentar a porcentagem de cada produto vendido.
* Monitorar o tempo de Fabricação / Entrega do produto.

**4 Justificativa**

No Brasil, o comércio eletrônico começou a surgir a partir de 1995. Desde então inúmeras empresas adotaram o e-commerce como uma forma de atender um público amplo, que tem um tipo de necessidade. Percebe-se que as grandes empresas do ramo estão aumentando a gama de produtos que há em seu portfólio, justamente para atrair mais clientes. Mas ainda existe certas áreas de consumo que ainda não foram devidamente exploradas.

Com isso, nosso interesse é atender uma parcela de pessoas, oferecendo um tipo de produto, no caso os alimentos, que ainda não é devidamente comercializado neste meio.

**5 Metodologia**

Dede o início do projeto cientifico tivemos esse objetivo em fazer um sistema que atenderia nossas necessidades em compras alimentícias, poupar tempo e comodidade em sair de nossa residência, ou até mesmo, do local de trabalho para realizar uma compra de um alimento, tipo “fast food”.

Assim veio a ideia de criar esse sistema chamado RESTAURANTE.NET. Um sistema WEB onde o cliente acessa e escolhe seu pedido via internet, pelo seu computador ou até mesmo seu celular, monta seu carrinho de compras e efetua o pagamento.

A partir deste momento ele acompanha o processo do seu pedido por STATUS do seu pedido, tais como: PAGAMENTO EFETUADO; EM PRODUÇÃO; SAIU PARA ENTREGA. E isso em tempo real.

No momento em que o cliente acessa o SITE, o mesmo informa seu CEP e a partir deste o sistema indica os restaurantes mais próximos da sua residência.

Neste momento ele seleciona o(s) restaurante(s) onde deseja realizar a compra, faz o pedido e efetua a compra. E acompanha o processo de produção.

O sistema tem muitas melhorias a ser implementada, devido ao tempo de entrega do projeto tivemos que limitar um pouco as funcionalidades do sistema.

No sistema atual tem implementado algumas partes significativas:

1. Cloud Computing;
2. API da Google Geográfica;
3. Dados Geográficos do SQL;

Em seguida segue algumas telas do sistema em funcionamento:

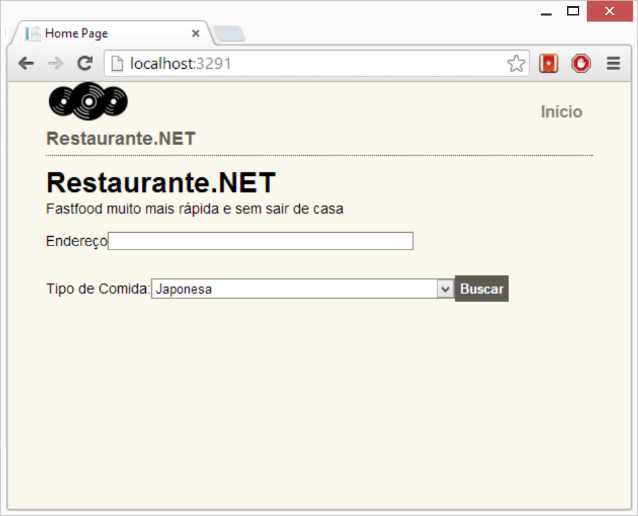


Figura 1 – Menu principal do programa, onde o usuário entra com os dados que foram solicitados. Disponível em: <[http://tigcloud.cloudapp.net](http://tigcloud.cloudapp.net/)> ou por meio do QRcode abaixo:



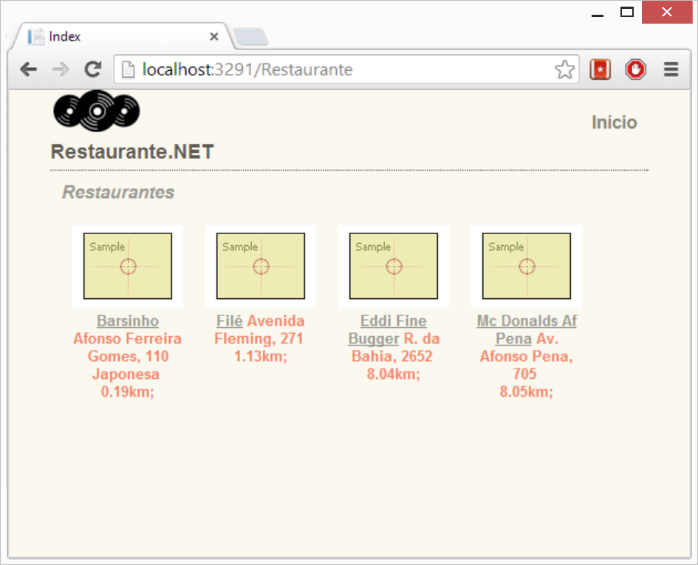


Figura 2 – Resposta do programa, mostrando quais os restaurantes que tem o tipo específico de comida solicitada, mais próximo do cliente.

Figura 3 – Dados do restaurante que foi solicitado, com opção de compra dos seus produtos.

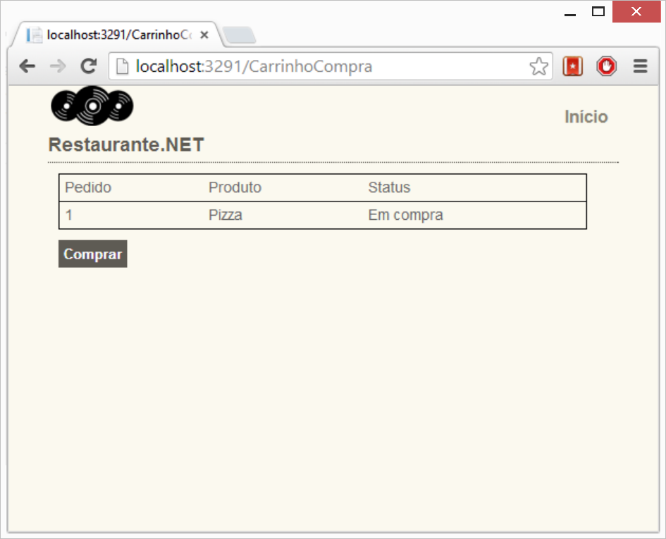


Figura 4 – Confirmação da compra pelo cliente.

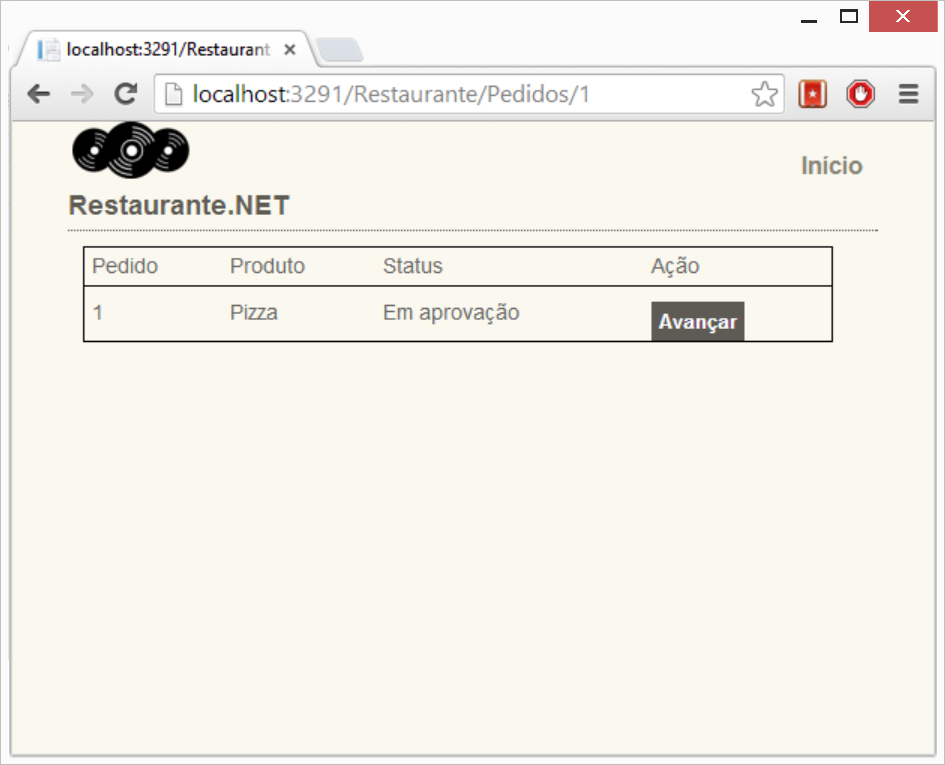


Figura 5 – Controle de pedido por parte da empresa, logo que o cliente realizou o pedido.

**6 Cronograma**

Tabela 1

Cronograma do Projeto

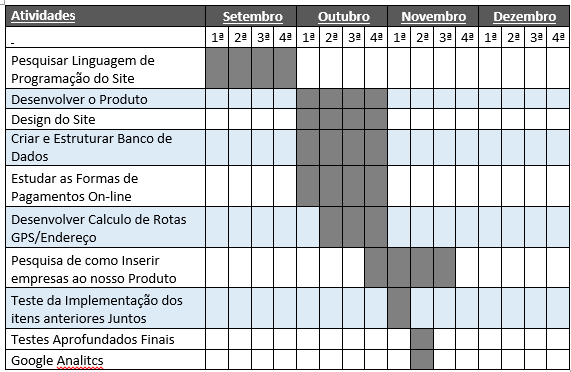
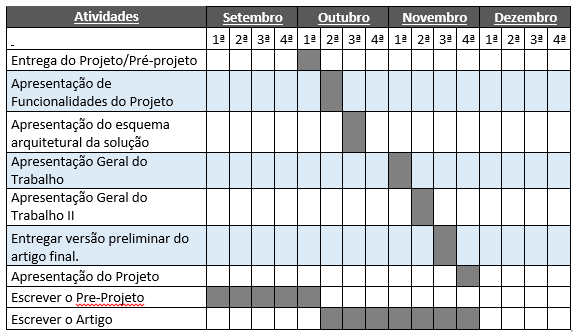


Tabela 2

Cronograma Estudantil



**7 Ferramentas Utilizadas**

**7.1 Ambiente de Desenvolvimento**

Foi utilizado o PowerPoint Storyboard para a concepção das páginas do site, o intuito dessa ferramenta é permitir visualizar em alto nível os principais elementos das páginas.

A IDE utilizada é o Visual Studio 2012 que possui recursos para desenvolvimento de aplicações para a Web e Mobile. A linguagem de programação escolhida foi o C# que é orientada a objetos, robusta e bem conceituada no mercado. O Visual Studio possui integração com bancos de dados SQL Azure e com o serviços de hospedagem em nuvem chamado Windows Azure.

**7.2 Ambiente de Servidor**

O ambiente servidor é hospedado na nuvem da Microsoft, o Windows Azure. Neste ambiente podemos criar sites e bancos de dados SQL Azure. Esta estrutura permite alterar o processador da máquina ou mesmo acrescentar novos servidores dinamicamente. Este recurso é útil em períodos de demanda sazonal.

**8 Problemas Encontrados**

O aplicativo possui uma série de funcionalidades que podem ser exploradas mais profundamente como:

* Redes sociais;
* Cálculos para recomendar os melhores restaurante e produtos;
* O uso de dispositivos móveis para uso tanto por empresas quanto por consumidores físicos.

Em vista da riqueza de possibilidades de investimento, o foco do desenvolvimento pode se perder. Para prevenir este problema o escopo da aplicação foi delimitado pelo protótipo de telas.

**9 interdisciplinaridade**

**9.1 Programação Orientada à Objetos**

A Programação Orientada à Objetos (POO), metodologia de programação que permite criar programas mais concisos com código reutilizáveis e de fácil manutenção. Neste paradigma as unidades do vistos como objetos que se inter-relacionam. Cada categoria de objetos, por exemplo a categoria Lanche ou cliente, é modelado em ponto comum denominada classe, que encapsula o comportamento comum de todos os objetos daquela classe. A Analise e o Projetos do Nosso Software também foram feitos utilizando metodologias OO.

Arquitetura Cliente Servidor - O software foi dividido em duas partes, o aplicativo cliente que rodará no browser do usuário. E o Aplicativo do servidor Que será responsável por receber e processar e processar as requisições feitas através do aplicativo cliente.

A parte do cliente foi modelando usando o Padrão Arquitetural Model-View-Controller (MVC). Com este padrão o código do é separado em três partes o que facilita o reuso e a manutenção. O Model contém a representação dos dados de parte da lógica do Domínio. O View contém a interface de usuário, são as telas do software. O Controller contém o código que faz a intermediação do Model com as Views e responde aos eventos das Views.

A Parte do Servidor está sendo desenvolvida usando Domain-Drive-Design (DDD) Projeto Orientado a Domínio em tradução Livre. Este modelo de desenvolvimento é fortemente embasado na linguagem utilizada pelos especialistas do Domínio do problema. A definição das classes, métodos, propriedades e demais artefatos do sistema são criados observando o vocabulário utilizado pelas pessoas que lidam com o problema que está sendo modelado. O DDD divide o software em cinco Principais categorias de artefatos. As Entidades, os Objetos de Valor, os Repositórios, As Fábricas e os Serviços, que são padrões do desenvolvimento Orientado à Objetos.

**9.2 Arquitetura e Organização de Computadores**

O ambiente em que um software será executado é parte integrante do seu projeto. Por ambiente entende-se tanto a parte do software -Sistema Operacional, Plataforma (.Net, JAVA, etc...), quanto a parte de Hardware - Memoria Necessário, Espaço em Disco, Processadores, Latência e Banda de Rede Etc. A parte do Hardware é a que nos interessa especificamente em relação a AOC. Uma configuração de hardware otimizada deve levar em conta fatores como o número de usuários que acessarão o sistema, a quantidade de dados que serão armazenadas os tipos e quantidades de operação de entradas e saídas que serão feitas por requisição. Ainda estamos na fase de implementação do nosso software após o qual passaremos teremos mais subsídios para tomar decisões a respeito da configuração de Hardware necessária para atender as suas demandas de uso.

**9.3 Estatística**

Usaremos a Estatística para nos ajudar a tomar importantes decisões em três pontos do nosso projeto: na Gerencia de Produto, Gerências da Infraestrutura e para dar informações relevantes aos usuários.

Para Gerência do produtos mediremos a forma como o consumidor chega ao nosso site, quanto tempo fica, em que ponto mais demora, em que ponto mais há desistência da compra. Com objetivo de identificar de onde vem os nossos usuários, quais os pontos da interface não fáceis e quais são mais difíceis, e onde há ponto de gargalo. Para podermos melhorar a experiência do usuários e consequentemente o retorno do produto.

Para Gerência de infraestrutura mediremos a quantidade de acessos, o tempo de respostas de várias partes no sistemas, Acesso ao Banco de Dados, Latência da rede interna e externa, períodos de Pico de Acesso, para com estes resultados tomarmos decisões de configuração de hardware e software no servidor e mudanças em algoritmos no software.

Para o usuário forneceremos métricas de produtos mais consumidos, mais baratos, mais rápidos para serem entregues.

**9.4 Cálculo de Várias Variáveis**

Para Cálculo, usaremos referências de coordenadas via GPS. O sistema funciona de certa forma em que o receptor (o GPS) capita sinais de 4 satelites para determinar as suas próprias coordenadas, e ainda o tempo. Então, o receptor calcula a distância a cada um dos satélites pelo intervalo de tempo entre o instante local e o instante em que os sinais foram enviados (esta distância é chamada pseudodistância). Decodificando as localizações dos satélites a partir dos sinais de micro-ondas (tipo de onda eletromagnética) e de uma base de dados interna, e sabendo a velocidade de propagação do sinal, o receptor, pode situar-se na intersecção de quatro calotes, uma para cada satélite.

Outra forma de utilizarmos cálculo, é pelas equações que criamos para analisar as medias de tempos gastas nas rotas de entregas.

**10 Cálculo de Várias Variáveis**

Foi construído um sistema online, para venda de produtos alimentícios via internet chamado restaurante.net, analisando os dados das compras realizadas, identificar e mostrar online para o cliente, o status real de seu pedido, e apresentar a porcentagem de cada produto vendido. Identificar para o cliente o local mais perto de sua residência para a compra de acordo a variedade de sabores ou produtos escolhidos pelo próprio cliente através de um filtro de conteúdo altamente explicativo e objetivo.

**11 Referência Bibliográfica**

* Model-View-Controller GotDotNet community, Pesquisa sobre arquitetura MVC

Disponível em: http://msdn.microsoft.com/en-us/library/ff649643.aspx. Acesso em 30 de setembro de 2013.

* An Introduction To Domain-Driven Design

David Laribee, Pesquisa sobre DDD, Disponível em: http://msdn.microsoft.com/en-us/magazine/dd419654.aspx. Acesso em 30 de setembro de 2013.

* Gabriel, M., educar a (r)evolução digital na educação – 1ª ed., 2013.