

**INSTITUTO FEDERAL DE EDUCAÇÃO CIÊNCIA E TECNOLOGIA DO RIO
GRANDE DO NORTE
DIRETORIA ACADÊMICA DE GESTÃO E TECNOLOGIA DA INFORMAÇÃO**

RELATÓRIO TÉCNICO DA PRÁTICA PROFISSIONAL

Luís Felipe Mesquita Cansanção Felipe

Orientador: Gilbert Azevedo da Silva, Doutor
(DIATINF/CNAT/IFRN)

Natal (RN), fevereiro de 2017.

**INSTITUTO FEDERAL DE EDUCAÇÃO CIÊNCIA E TECNOLOGIA DO RIO
GRANDE DO NORTE
DIRETORIA ACADÊMICA DE GESTÃO E TECNOLOGIA DA INFORMAÇÃO**

RELATÓRIO TÉCNICO DA PRÁTICA PROFISSIONAL

Luís Felipe Mesquita Cansanção Felipe

Orientador: Gilbert Azevedo da Silva, Doutor
(DIATINF/CNAT/IFRN)

Relatório técnico apresentado à DIATINF para a conclusão da Prática Profissional do Curso Técnico de Nível Médio Integrado em Informática para Internet, em cumprimento às exigências legais como requisito parcial à obtenção do título de Técnico em Informática para Internet.

Natal (RN), fevereiro de 2017.

“Ainda que eu conheça a jornada e onde ela leva... Eu a abraço. E eu saúdo cada momento dela”

- Dra. Louise Banks(Filme “A chegada”)

SUMÁRIO

LISTA DE ABREVIATURAS	5
RESUMO	6
RÉSUMÉ	7
1. INTRODUÇÃO	8
2. FUNDAMENTAÇÃO TEÓRICA	9
2.1. TECNOLOGIAS	9
3. MODELAGEM DO SISTEMA	10
3.1. REQUISITOS DO SISTEMA	10
3.1.1. REQUISITOS FUNCIONAIS	10
3.1.2. REQUISITOS NÃO-FUNCIONAIS	11
3.2. CASOS DE Uso	12
3.2.1. MÓDULOS	12
3.3. ARQUITETURA	18
3.4. DIAGRAMA DE CLASSES DO BANCO DE DADOS	19
4. RESULTADOS	20
4.1. WEB	20
4.2. SERVIÇO	20
4.3. APLICAÇÃO MÓVEL DO CLIENTE	21
4.4. APLICAÇÃO MÓVEL DO GARÇOM	22
4.4.1. GET DA MESA	23
4.4.2. DELETE DA MESA	23
4.4.3. GERADOR DE CÓDIGO ALEATÓRIO PARA ABERTURA DE MESA	24
4.4.4. EXCLUSÃO DE PEDIDOS	25
5. CONCLUSÃO	27
REFERÊNCIAS BIBLIOGRÁFICAS	28

LISTA DE ABREVIATURAS

IFRN	Instituto Federal de Educação Tecnológica do Rio Grande do Norte
DIATINF	Diretoria Acadêmica de Gestão e Tecnologia da Informação
ASP	Active Server Pages
API	Application Programming Interface
REST	Representational State Transfer

RESUMO

FELIPE, Luís. **Relatório Técnico da Prática Profissional**. Natal, Ano. MMXVII. Relatório Técnico de conclusão da Prática Profissional do Curso Técnico de Nível Médio Integrado em Informática para Internet. Diretoria Acadêmica de Gestão e Tecnologia da Informação, Instituto Federal de Educação, Ciência e Tecnologia do Rio Grande do Norte – campus Natal/Central, Natal-RN, 2017.

Produzido por uma equipe de 4 pessoas (Isaac Medeiros, Gilson Gabriel Martins, Luís Felipe Mesquita e Marcos Vinícius Bandeira) como projeto de extensão durante o período de agosto/2016 a fevereiro/2017, a produção desse aplicativo tinha por função emular uma experiência “real” num posto de trabalho. Permeado por reuniões semanais com o orientador Gilbert, pudemos vivenciar algo que acredito ser muito próximo disso, ainda que prazos e responsabilidades não fossem fidedignamente condizentes com a realidade corrida de uma agência ou de uma empresa, foi uma experimentação de extrema relevância para o nosso preparo como futuros profissionais.

Palavras-Chave: Android; App; Projeto de Extensão; Restaurante.

RÉSUMÉ

Ce article est une visage technique sur le process de développement d'une application mobile pour les portable avec Android en regardant être utilisé dans restaurants.

Devélopé par une equipe avec 4 personnes (Isaac Medeiros, Gilson Gabriel Martins, Luís Felipe Mesquita e Marcos Vinícius Bandeira) comme um projet d'extension pendant le périod de août/2016 a février/2017, le développement de cette application avait par fonction simuler une expérience "réele" dans une rôle. Avec des rendez-vous hebdomadaires avec notre conseiller Gilbert, nous avons experimenté une chose très proche d'une expérience dans une entreprise ou agence. Même que les délais de livraison ont été plus confortables, il a été une experimentation trop importante pour notre formation professionnelle.

Mots-clés: Développement; Application Mobile; Android.

1. INTRODUÇÃO

1.1. Contexto

Nos dias de hoje, em praticamente 100% das ocasiões logo no momento em que se chega no restaurante a primeira coisa a ser feita é chamar o garçom ou o responsável pelos pedidos. Salvo raras exceções de ambientes mais requintados nas quais o garçom possui um dispositivo que manda o pedido diretamente à cozinha, os atendentes utilizam-se dos instrumentos mais rústicos o possível: caneta e papel para anotar o pedido que será lido pelos cozinheiros e, assim, preparado.

Ainda que o garçom detenha essa tecnologia, ele é apenas um intermediador entre o dispositivo e o cliente, apenas comandando o que é pedido.

Com o exponencial crescimento de usuários de dispositivos móveis, vê-se que cada vez mais essa função de intermediador tende a desaparecer. Por isso, surgiu a ideia de desenvolver uma plataforma que mediasse a comunicação entre o cliente e o restaurante, propriamente dito.

1.2. Objetivo

Este projeto tem por objetivo principal desenvolver uma plataforma que agilize o processo de atendimento e pedido de produtos em restaurantes, sejam de pequena, média ou larga escala, podendo tornar-se benéfico ao restaurante pela consequente redução de custos e para o cliente com a agilização do seu pedido, através da interação cliente-plataforma e plataforma-restaurante.

1.3. Organização

Como trabalhamos numa organização com 4 integrantes foi substancial que houvesse a divisão dos trabalhos de cada integrante para que o projeto caminhasse de forma correta. É natural que num projeto desse tipo não haja uma homogeneidade nesse quesito e finde por todos se ajudarem. No entanto, se traduzíssemos isso para o relatório haveria uma enorme bagunça e traríamos apenas dor de cabeça para o nosso orientador/corretor.

Pensando nisso, adotamos o seguinte método: cada integrante dá uma pequena introdução sobre o projeto e, posteriormente no documento, na seção de resultados, mostra como foram os resultados de uma aplicação. Como são 4 aplicações e 4

integrantes, esse método encaixou perfeitamente. No caso deste documento, fiquei incumbido por dar enfoque na aplicação destinada ao Garçom.

2. FUNDAMENTAÇÃO TEÓRICA

2.1. Tecnologias

Android: Sistema Operacional (SO) para dispositivos móveis, desenvolvido pela Google e lançado oficialmente em 23 de setembro de 2008. É o SO mais utilizado no mundo presentes em cerca de 61,24% (*StatCounter GlobalStats*) desses dispositivos no mundo.

C#: Linguagem de programação desenvolvida pela Microsoft no ano de 2001. Tem sua sintaxe orientada a objetos baseada em C++, no entanto possui diversas influências de linguagens como Pascal e Java.

Microsoft Visual Studio 2015(C#): Plataforma da Microsoft voltada para o desenvolvimento de software, sobretudo para o Visual Basic.

Microsoft SQL Server: O Microsoft SQL Server é um SGBD - sistema gerenciador de Banco de dados relacional desenvolvido pela Microsoft.

JSON: Formato utilizado para intercâmbio de dados computacionais. JSON é um acrônimo para *Javascript Object Notation*.

Newtonsoft Json: *Framework JSON* popular e de alta performance utilizada junto ao .NET.

REST: Serviço Web capaz de fazer a conexão de dados entre os dispositivos utilizados no projeto.

Git: Sistema de controle de versionamento utilizado majoritariamente para o desenvolvimento de software. Foi criado em 2005 por Linus Torvalds.

GitHub: Serviço de hospedagem para projetos que utilizam o controle de versionamento Git.

ASP.NET: Framework de desenvolvimento *open source* criada pela Microsoft para suceder a tecnologia ASP. As aplicações para essa plataforma podem ser escritas em diversas linguagens, como C#, F# e Visual Basic.

Ionic Framework: Framework para desenvolvimento de aplicações móveis.

AngularJS: Framework Java Script utilizado para desenvolvimento no *Ionic*.

3. MODELAGEM DO SISTEMA

3.1. Requisitos do Sistema

3.1.1. Requisitos funcionais

Cód.	Nome	Descrição	Categoria
F01	Abrir mesa	O garçom poderá abrir a mesa.	Evidente
F02	Realizar pedido	O cliente poderá realizar o pedido.	Evidente
F03	Visualizar pedido	O cliente poderá rever seu pedido	Desejável
F04	Chamar Garçom	O cliente pode notificar o garçom.	Desejável
F05	Visualizar pedidos abertos	O gerente de pedido poderá visualizar os pedidos em abertos.	Evidente
F06	Atender pedido	O gerente de pedidos poderá definir o pedido como disponível para a entrega.	Evidente
F07	Imprimir etiqueta	Uma etiqueta será impressão quando o pedido for finalizado	Desejável
F08	Visualizar pedidos atendidos	O garçom poderá ver pedidos prontos para ser entregues.	Evidente
F09	Confirmar entrega	O garçom pode confirmar quando entregar o pedido.	Desejável
F10	CRUD Menu	O administrador do restaurante poderá gerenciar o menu.	Evidente
F11	CRUD Fila pedido	O administrador do restaurante poderá gerenciar as filas do pedido.	Evidente

F12	CRUD Restaurante	O administrador do sistema poderá gerenciar restaurantes no aplicativo.	Desejável
F13	CRUD Administrador do restaurante	O administrador do sistema poderá gerenciar administradores do restaurante.	Evidente
F14	Fechar mesa	O garçom poderá fechar a mesa do cliente.	Evidente
F15	Encerrar mesa	Quando o F14 for realizado, o caixa poderá encerrar definitivamente a mesa do cliente.	Evidente
F16	Cancelar pedido	O cliente poderá cancelar seu pedido.	Evidente
F17	Cancelar pedido	O gerente de pedidos poderá cancelar o pedido do cliente.	Evidente
F18	CRUD Garçom	O administrador do restaurante poderá gerenciar os garçons.	Evidente
F19	CRUD Gerente de Pedidos	O administrador do restaurante poderá gerenciar os gerentes de pedidos.	Evidente
F20	CRUD Caixa	O administrador do restaurante poderá gerenciar os caixas.	Evidente

3.1.2. Requisitos Não-Funcionais

Cód.	Nome	Descrição	Categoria
NF01	Prazo	O sistema deve estar pronto no final do semestre letivo de 2016.2, tendo o tempo de aproximadamente seis (06) meses para o seu desenvolvimento	Obrigatório
NF02	Robustez	O sistema deve atender ao objetivo de confiabilidade do sistema e executar sem erros recorrentes.	Desejável
NF03	Usabilidade	O sistema deve possuir uma visualização intuitiva	Desejável

3.2. Casos de Uso

3.2.1. Atores

Ator	Descrição
Cliente	Ator responsável por toda corrente de funcionalidades, a partir de sua função, que é realizar o pedido.
Gerente de pedidos	Responsável por fazer “a ponte” entre o garçom e a cozinha/bar etc.
Garçom	Responsável por liberar o acesso do cliente e fazer as entregas físicas e auxiliar no que for preciso ao mesmo.
Administrador do sistema	Responsável por gerenciar os restaurantes e os respectivos gerentes.
Administrador do restaurante	Responsável por cadastrar as filas e menus do restaurante.
Caixa	Responsável por encerrar definitivamente a conta do cliente no restaurante, e receber o pagamento.

3.2.2. Módulos

O projeto contou com diversas divisões para que obtivéssemos uma interação mais ampla por parte do usuário. Ademais, era de substancial valor que essa interação se desse com tal amplitude, uma vez que o funcionamento adequado da aplicação se dá apenas com a plena integração desses módulos.

3.2.2.1. Web – Site

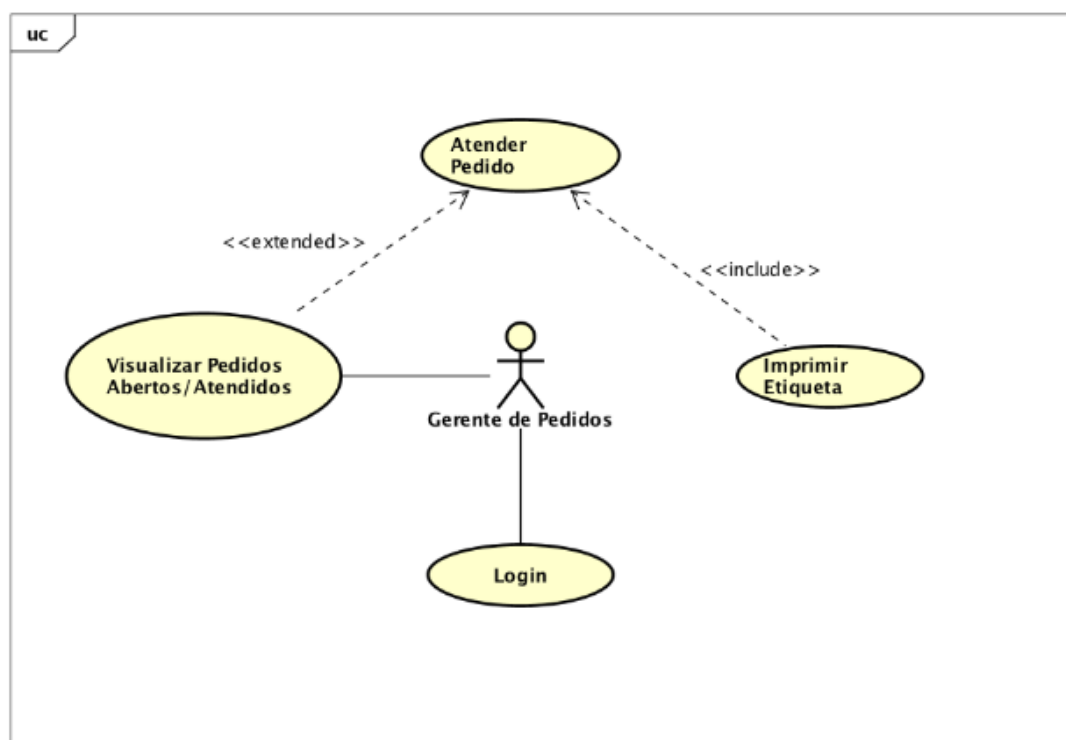
O projeto contou com a parte Web, onde há CRUDs que abrangem todo o Banco de Dados da aplicação. Lá, é possível fazer o cadastro de Produtos, qualquer tipo de Usuário (de administradores a restaurantes e respectivos funcionários), Cardápios etc.

Vale ressaltar que essa funcionalidade é intermediada por um sistema de Login e cadastro funcionais.

Cód.	Caso de Uso	Descrição	Classificação	Atores
UC01	Visualizar pedidos abertos	O gerente de pedido poderá visualizar os pedidos em abertos.	Primário	Gerente de pedidos
UC02	Atender pedido	O gerente de pedidos verá os pedidos já preparados.	Primário	Gerente de pedidos
UC03	Imprimir etiqueta	Uma etiqueta do pedido será imprimida ao ser finalizado, contendo o número da mesa correspondente.	Secundário	Gerente de pedidos
UC04	CRUD Menu	O administrador do restaurante poderá inserir, remover, atualizar ou deletar o menu do restaurante.	Primário	Admin restaurante
UC05	CRUD Fila pedido	O administrador do restaurante poderá inserir, remover, atualizar ou deletar filas de acordo com a categoria do produto no sistema.	Primário	Admin restaurante
UC06	CRUD Mesa	O administrador do restaurante poderá inserir, remover, atualizar ou deletar mesas.	Primário	Admin restaurante
UC07	CRUD Produto	O administrador do restaurante poderá inserir, remover, atualizar ou deletar produtos no sistema.	Primário	Admin restaurante
UC08	Autenticação	O admin do restaurante deverá autenticar o seu Login para ter acesso às funcionalidades descritas nos outros casos de uso.	Primário	Admin restaurante
UC09	CRUD Restaurante	O administrador do sistema poderá inserir, remover, atualizar ou deletar restaurantes na barra de procura do aplicativo	Primário	Admin sistema
UC10	CRUD Administrador do restaurante	O administrador do sistema poderá inserir, remover, atualizar ou remover	Primário	Admin sistema

		administradores do restaurante.		
UC11	Autenticação	O admin do sistema deverá autenticar o seu Login para ter acesso às funcionalidades descritas nos outros casos de uso.	Primário	Admin sistema
UC12	Cancelar pedido	O gerente de pedidos poderá cancelar um pedido do cliente caso o item esteja indisponível.	Primário	Gerente de pedidos
UC13	Encerrar mesa	Quando o UC14 for realizado, o caixa poderá encerrar definitivamente a mesa do cliente.	Primário	Caixa

Figura 1:Diagrama de Casos de Uso Gerente de Pedidos que faz uso da plataforma WEB



Fonte: Acervo do Usuário

3.2.2.2. Web – Serviço

Andando conjuntamente ao CRUD Web, há de se ter o funcionamento do serviço. Utilizando-se da tecnologia REST, todas as funcionalidades vistas no CRUD, bem como no diagrama de Casos de Uso acima, foram implementadas utilizando-se do serviço.

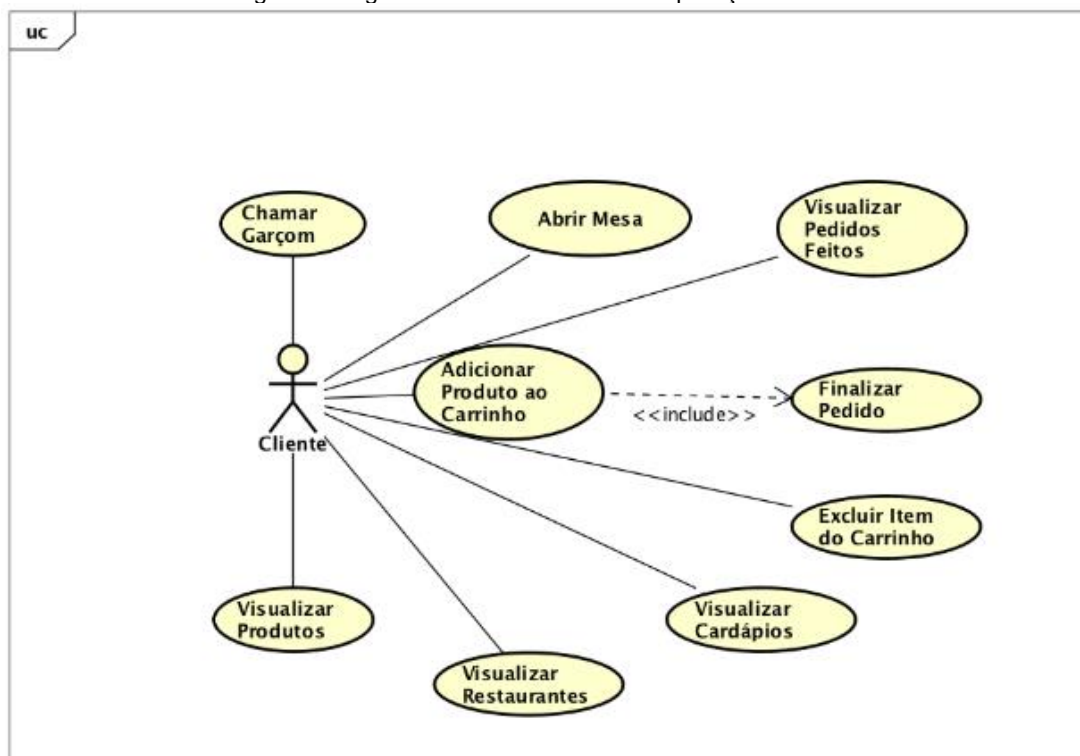
3.2.2.3. Cliente

O principal elemento da parte mobile do sistema é a aplicação do Cliente. Aqui é onde mora a principal ideia de nosso projeto e onde tivemos de trabalhar mais. A modelagem dessa seção foi imprescindível para que os problemas que buscávamos resolver fossem, de fato, resolvidos. Caso essa modelagem fosse feita erroneamente, toda a ideia do projeto seria no mínimo mal aproveitada.

Cod.	Caso de Uso	Descrição	Classificação	Atores
UC01	Abrir Mesa	O garçom deve abrir a conta para o cliente, informando o número da mesa e o código a ser utilizado.	Primário	Cliente
UC02	Visualizar Produtos	O cliente deve realizar o pedido selecionando os itens no aplicativo.	Primário	Cliente
UC03	Adicionar Produto ao carrinho	O cliente poderá visualizar e acompanhar o seu pedido.	Secundário	Cliente
UC04	Chamar Garçom	O cliente poderá chamar o garçom através do aplicativo, mandando uma notificação.	Primário	Cliente
UC05	Visualizar pedidos feitos	O gerente de pedido poderá visualizar os pedidos em abertos.	Primário	Cliente
UC06	Visualizar Restaurantes	O gerente de pedidos verá os pedidos já preparados.	Primário	Cliente
UC07	Finalizar Pedido	Uma etiqueta do pedido será imprimida ao ser finalizado, contendo o número da mesa correspondente.	Secundário	Cliente
UC08	Excluir item carrinho	O garçom verá os pedidos que já foram preparados.	Primário	Cliente

UC09	Chamar Garçom	O garçom irá confirmar quando entregar o pedido para o cliente.	Secundário	Cliente
------	---------------	---	------------	---------

Figura 2:Diagrama de Casos de Uso da aplicação do cliente



Fonte: Acervo do usuário

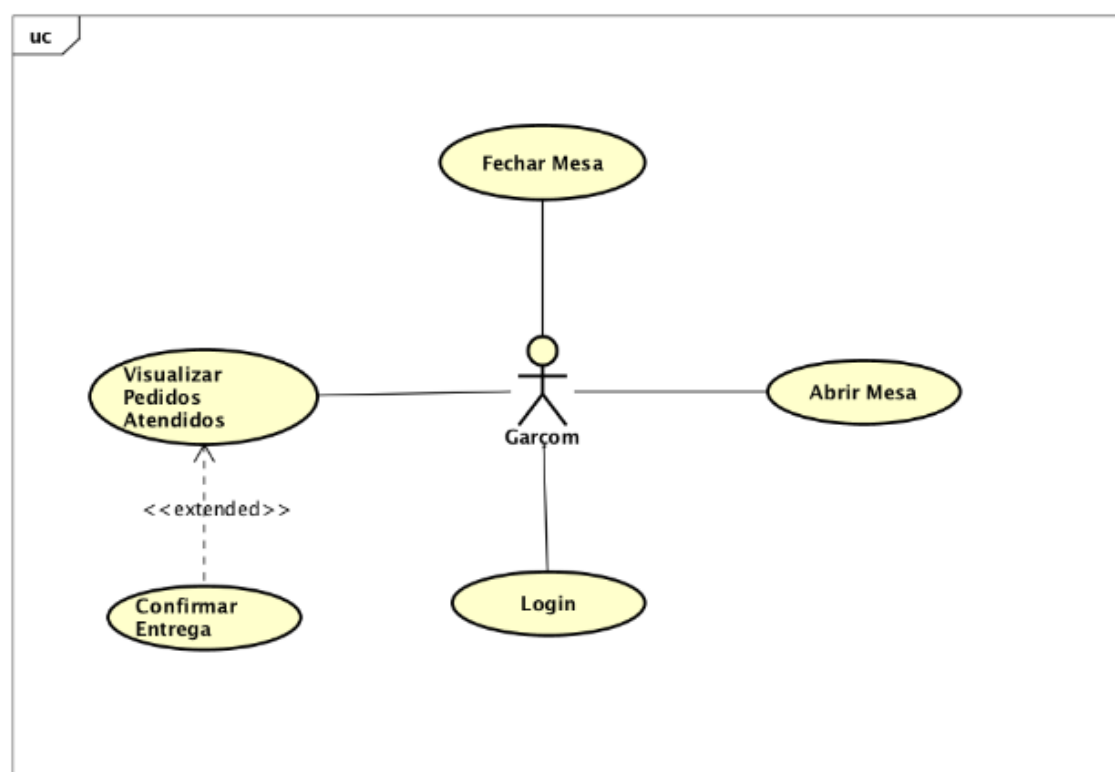
3.2.2.4. Garçom

Coexistindo com a parcela mobile do projeto está a aplicação do Garçom, na qual é possível para o Garçom “conversar” com o Cliente de maneira mais ágil e simplória. Igualmente importante ao Cliente, o Garçom é substancial para que as devidas qualidades do projeto sejam postas em prática.

Cód.	Caso de Uso	Descrição	Classificação	Atores
UC01	Abrir mesa	O garçom deve abrir a conta para o cliente, informando o número da mesa e o código a ser utilizado.	Primário	Garçom
UC02	Fechar Mesa	O garçom poderá fechar a mesa do cliente, quando o mesmo autorizar.	Primário	Garçom

UC03	Login	Permite que o ator em questão faça seu login na plataforma e tenha acesso às outras funcionalidades.	Primário	Garçom
UC04	Confirmar Entrega	O garçom irá confirmar quando entregar o pedido para o cliente	Secundário	Garçom
UC05	Visualizar pedidos atendidos	O garçom verá os pedidos que já foram preparados.	Primário	Garçom

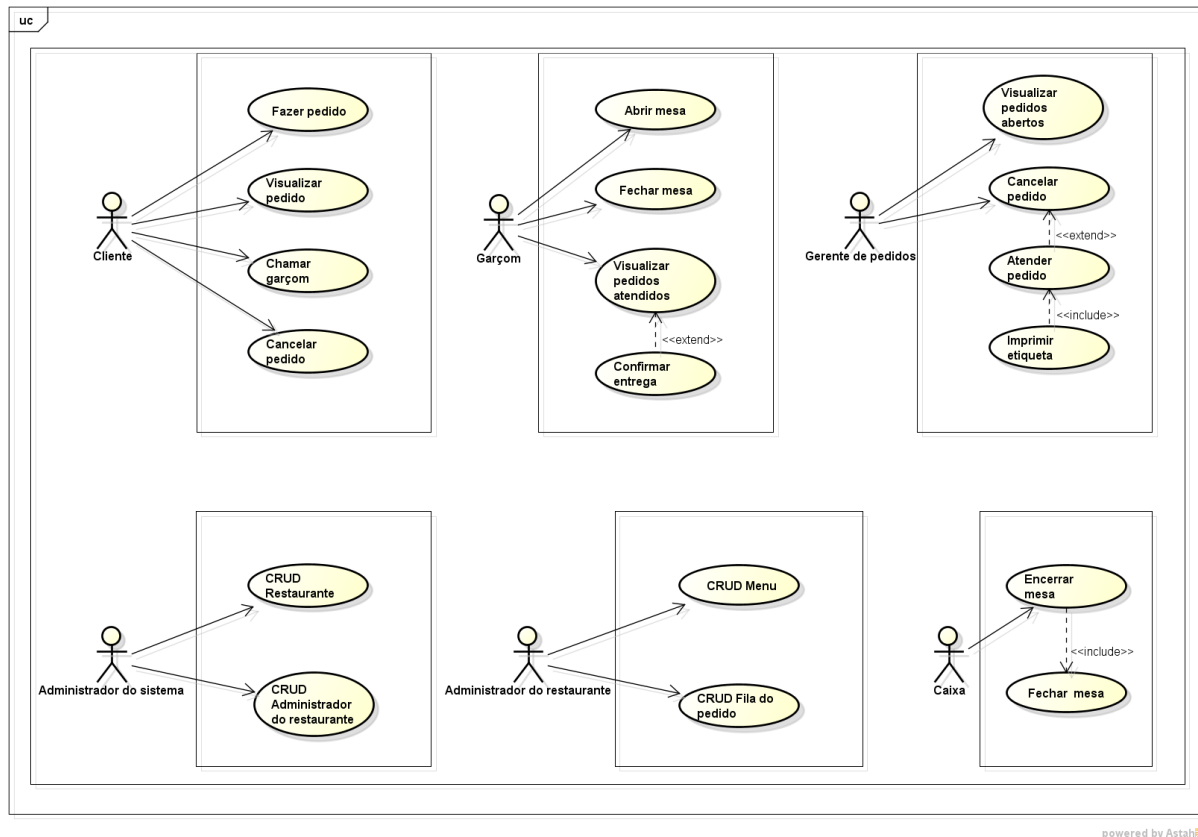
Figura 3: Diagrama dos Casos de Uso da aplicação do Garçom



Fonte: Acervo do usuário

3.3. Arquitetura

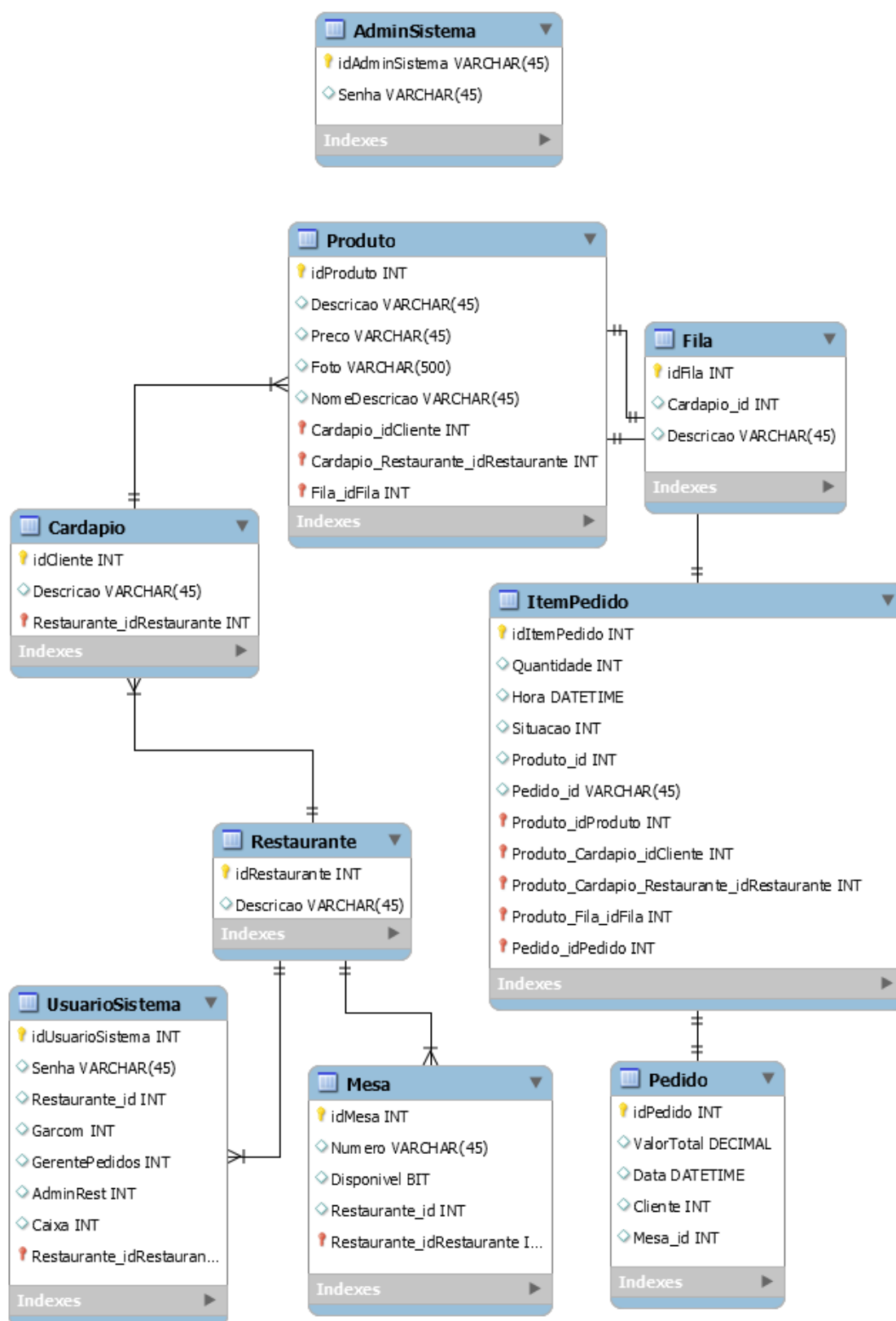
Figura 4: Diagrama de Arquitetura do projeto



Fonte: Acervo do usuário

3.4. Diagrama de Classes do Banco de Dados

Figura 5: Diagrama de Classes do Banco de Dados



Fonte: Acervo do usuário

4. RESULTADOS

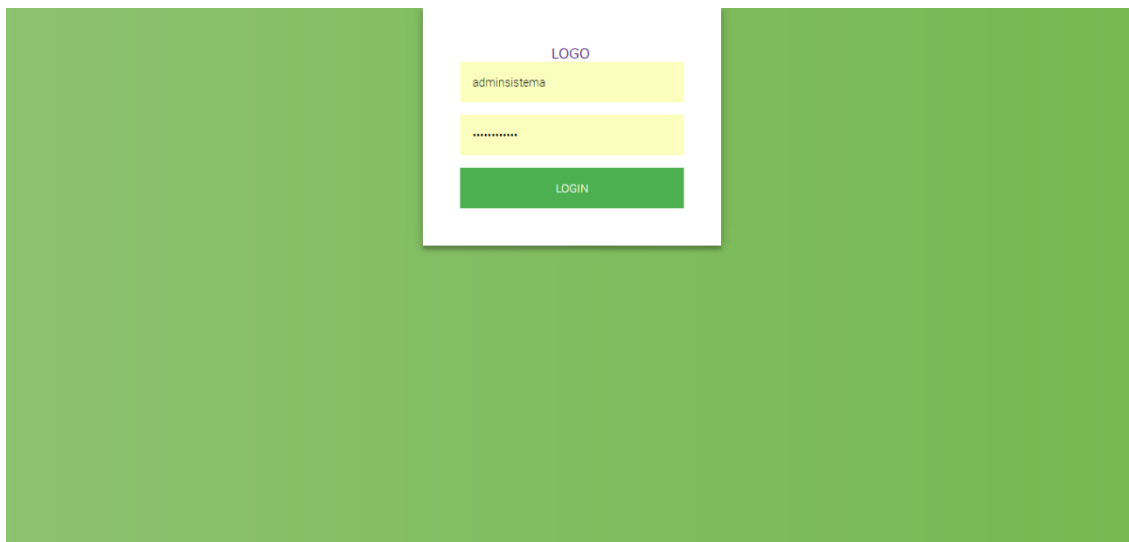
O sistema finalizado apresentou um total de 4 aplicações, sendo eles 2 para aplicativos móveis, onde ocorre a maior parte da interação do usuário, 1 serviço REST e uma aplicação WEB para o setor administrativo.

Vale salientar que utilizamo-nos do banco de dados “Valera” fornecido por nosso orientador e, também, de nossos próprios acessos para publicar o serviço. Por isso, pode-se observar nos comandos abaixo a utilização da matrícula de um dos nossos integrantes como endereço do servidor.

4.1. Web

A parte WEB consiste em alguns CRUDs que serão responsáveis pelo gerenciamento objetos principais do projeto, desde Usuários até o manejo dos Produtos de cada restaurante.

Figura 6: Página inicial da plataforma WEB



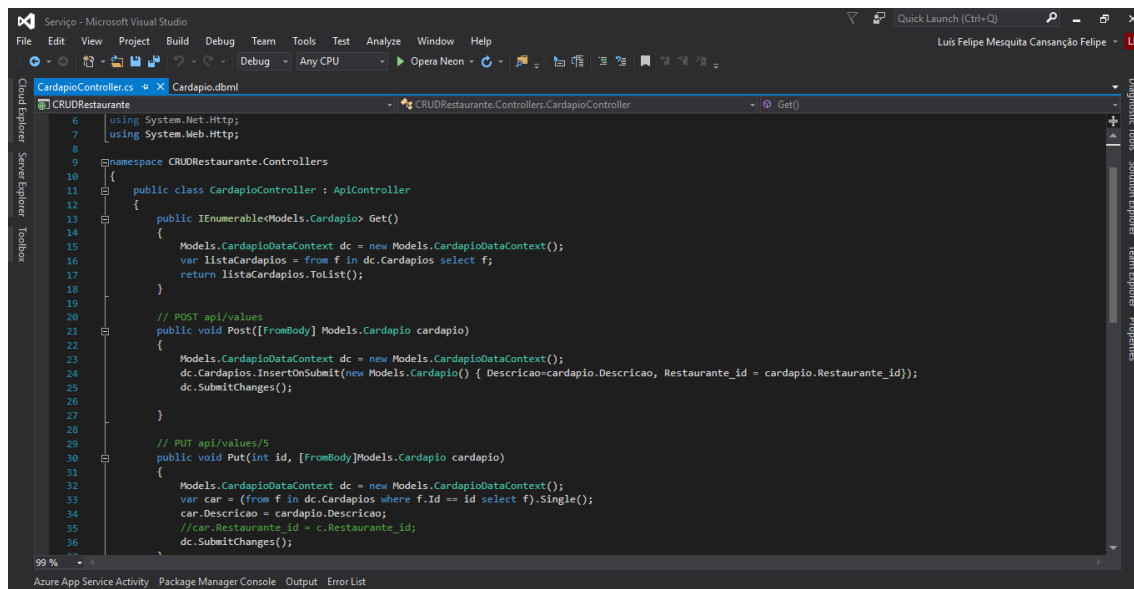
Fonte: Acervo do usuário

4.2. Serviço

No serviço foi feito basicamente um *controller* para cada tabela do banco de dados, para que assim pudessemos administrá-las por meio da plataforma WEB com exceção

de alguns como os que dão conta das contas dos usuários tem partes dedicadas a criptografia, por exemplo. No mais, cada *controller* conta com as funções de um CRUD.

Figura 7: Captura de tela de um *controller*



Fonte: Acervo do usuário

4.3. Aplicação móvel do cliente

A principal aplicação de nosso projeto contou com telas para Listagem de Restaurante, Tela Inicial do Usuário, Carrinho, Cardápios e Produtos.

Figura 8: Página Inicial da aplicação para o Cliente

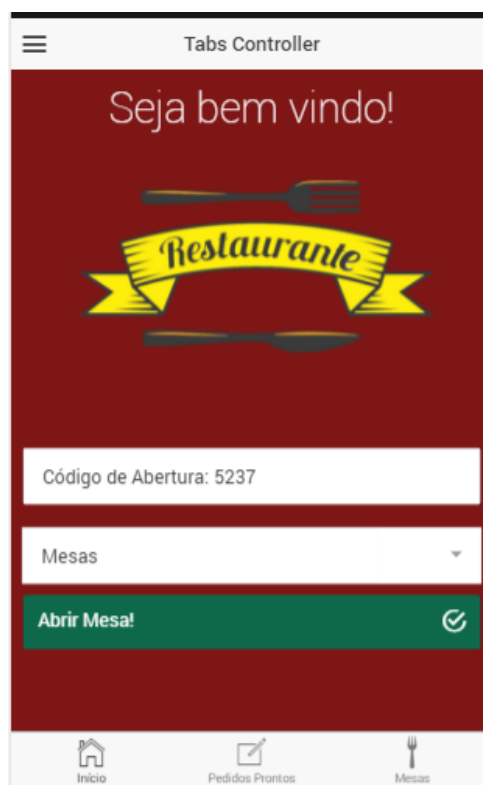


Fonte: Acervo do usuário

4.4. Aplicação móvel do garçom

O que teve a maior parte de nosso foco na aplicação do Garçom foi exatamente a parte de *controllers*. Com eles, pudemos resgatar dinamicamente as informações provindas do serviço que tinham relação com o Garçom.

Figura 9: Início da aplicação do Garçom(Atentando-se ao código de abertura)



Fonte: Acervo do Usuário

4.4.1. GET da Mesa

```
$http.get('http://10.21.0.137/20131011110061/api/mesa').  
  
  then(function(response) {  
  
    var x = response.data;  
  
    console.log(x); //Checagem do recebimento de objetos  
  
    $scope.Mesas = response.data;  
  
  });
```

4.4.2. DELETE da Mesa

```
$http.get('http://10.21.0.137/20131011110061/api/mesa').  
  
  then(function(response) {  
  
    var x = response.data;
```

```

    console.log (x); //Checagem

    $scope.mesas = response.data;

  })

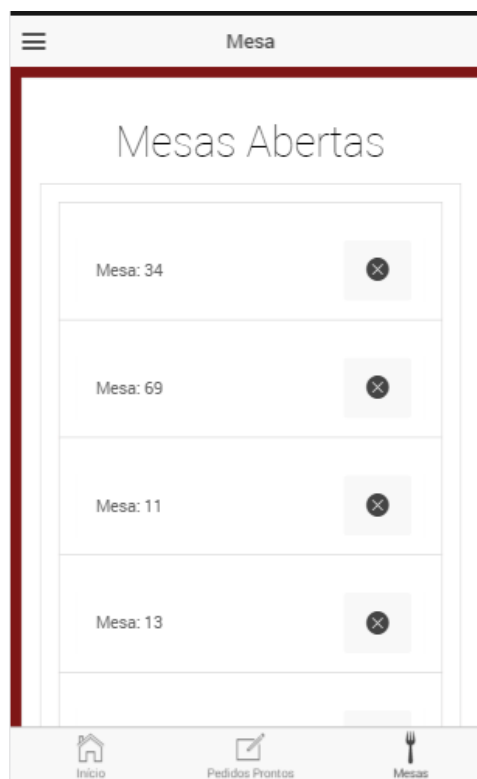
  $scope.remove = function($index){

    $scope.mesas.splice($index, 1); //Splice = remover do array

  }

```

Figura 10: Tela que mostras as mesas aberta sendo possível excluí-las



Fonte: Acervo do usuário

4.4.3. Gerador de código aleatório para abertura de mesa

```

var data = {

  Mesa: 3,

  Cliente:Math.floor(Math.random()*9000) + 1000

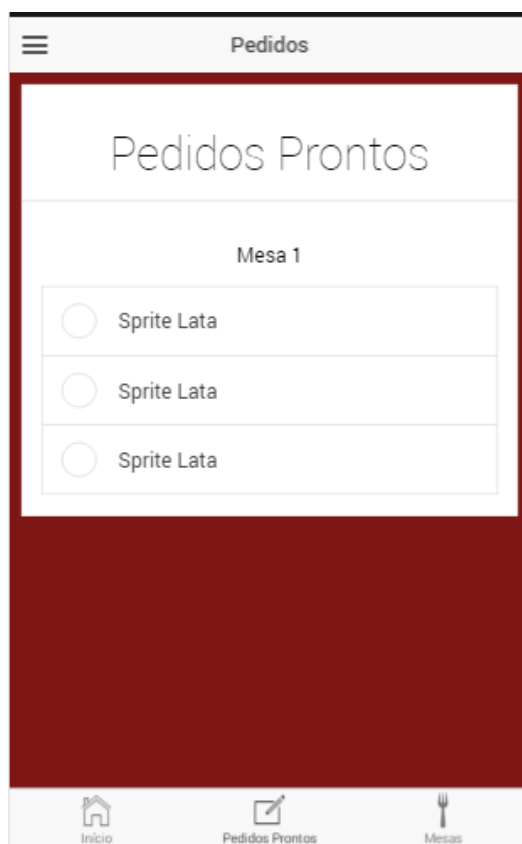
};

```


4.4.4. Exclusão de pedidos

```
$http.get('http://10.21.0.137/20131011110061/api/itempedidoproduto').  
  
  then(function(response) {  
  
    var x = response.data; //Checagem  
  
    var key.length= 0;  
  
    var y = [];  
  
    var count=0;  
  
    for(key in x) {  
  
      if(x.hasOwnProperty(key)){  
  
        length++;  
  
      }}  
  
    for(var i =0;i< length;i++){  
  
      if(x[i].Situacao== 2){  
  
        count++  
  
        y.push(x[i]);  
  
      }}  
  
    console.log(x);  
  
    $scope.itempedidoprodutos = y;})
```

Figura 11: Pedidos prontos, sendo também possível marcar se eles foram atendidos.



Fonte: Acervo do usuário

5. CONCLUSÃO

Essa experiência foi extremamente proveitosa tanto para nossos estudos quanto para a valorização de nossa profissão. Pudemos ter uma experiência mais condizente com a realidade de nosso campo de trabalho do que termos normalmente em nossas aulas e isso com certeza será de bom grado tanto para com nossas possíveis profissões, como também para ajudar-nos a definir se realmente queremos ou não seguir nessa área.

As diversas experiências, desde aprender uma nova linguagem até trabalhar com prazos foi algo substancial nessa jornada. Ademais, diferentemente de outros projetos que vimos no nosso laboratório, este projeto foi feito em equipe, o que acrescenta ainda mais no tocante a divisão laboral e trabalho em equipe.

Dessa forma, tenho plena certeza que tal prática veio só pra adicionar em nossas vidas acadêmicas e que ela com certeza foi, senão um pontapé inicial em nossas carreiras, um divisor de águas.

REFERÊNCIAS BIBLIOGRÁFICAS

- W3SCHOOLS. **AngularJS Tutorial**. Disponível em: <<https://www.w3schools.com/angular/>>. Acesso em: 17 fev. 2017.
- WIKIPÉDIA. **Microsoft Visual Studio**. Disponível em: <https://pt.wikipedia.org/wiki/Microsoft_Visual_Studio>. Acesso em: 17 fev. 2017.
- WIKIPÉDIA. **REST**. Disponível em: <<https://pt.wikipedia.org/wiki/REST>>. Acesso em: 17 fev. 2017.
- IONIC. **HomePage**. Disponível em: <<https://ionicframework.com/>>. Acesso em: 17 fev. 2017.
- WIKIPÉDIA. **JSON**. Disponível em: <<https://pt.wikipedia.org/wiki/JSON>>. Acesso em: 17 fev. 2017.
- WIKIPÉDIA. **Interface de programação de aplicações**. Disponível em: <https://pt.wikipedia.org/wiki/Interface_de_programação_de_aplicações>. Acesso em: 17 fev. 2017.
- IONIC. **Publishing your app**. Disponível em: <<https://ionicframework.com/docs/guide/publishing.html>>. Acesso em: 17 fev. 2017.
- STACK OVERFLOW. **Fórum de dúvidas**. Disponível em: <<http://pt.stackoverflow.com/questions/tagged/angularjs>>. Acesso em: 17 fev. 2017.