

Documentação

“Triple Thread” v.0.0.1

‘BETA’

INDEX

1. Soluções
2. Descrição
3. Recursos utilizados
4. Fatos de estudo de campo
5. Estrutura do Banco de dados
6. Engine 1.0 ou peso da fila
7. Engine 1.1 pontuação gamificação
8. Carrousel e porquê ele é incrível
9. Navegação
10. Icebox
11. Resultados aguardados
12. Entrega de Dados
13. Considerações Finais

Soluções do software

- 1 Comanda garçom X Comanda cliente
- 2 'classificador' e gestor de filas,
- 3 gamificação,
- 4 extrator de dados,
- 5 relatório real time de quem entra e consome X quem entra e não consome
- Pontuador em 5 estrelas de unidade e Staff

Descrição:

ponto do usuário 'client' solicitar um pedido direto da mesa e/ou o garçom. Um app para otimizar filas, diminuir o incidente de canhotos de papel e identificar com BI o número de clientes pagantes e não pagantes. Trazendo peso dos pedidos para uma pilha inteligente, e fornecendo um valor atributo ao usuário.

Recursos utilizados

Progressive WebApp (Mobile e WEB)

(Front-End) HTML 5, Javascript, CSS 3, Bootstrap 4

(Back-End) PHP

(DBA) Firebase

(Repositório e Hospedagem) Heroku

Estudo de campo

Outubro e Novembro

Localizado 3 perfis chave de cliente

.1 cliente "cafezal"

Cliente mais presente no período da manhã, seus pedidos são quase que exclusivamente café

.2 cliente "notebook"

Cliente do período da tarde, consome pouco e fica boa parte do tempo na loja trabalhando na maioria das vezes no seu computador

.3 cliente "fim de semana"

Consome varios tipos de produto, nunca vem à loja sem companhia.

Baixa absorção de dados profundos dos clientes como CPF, região onde habita, classe social, quem é o cliente, qual pedido mais recorrente, o quanto ele consome para o quanto ele vem a loja, quais lojas ele mais frequenta, qual forma de pagamento prefere, qual nível de escolaridade e etc....

Fim de semana garçons em horário de pico utilizam canhoto em papel.

Não há sistema de entrega de cupons tanto para cliente habituê quanto cliente que pouco consome, inexistindo com isso um pós-venda.

Não há uma oferta de produto para impulsionar as vendas de um produto novo ou algo do tipo. Não há nada para evitar quebra de produto (Dado colhido com dpto. controle de perda: Custou a todas as unidades R\$ 69.000 reais somente de Outubro até Novembro, estimado gasto anual de meio milhão com quebras, além de ser perda de mercadoria e dinheiro para as unidades, devemos considerar como tragédia qualquer comida jogada fora). Tampouco sabe-se quais produtos tem maior índice de quebra.

Não há uma estratégia para lidar com o padrão de clientes;

Estrutura da banco de dados

FIREBASE

<https://benjamin-a-padaria.firebaseio.com/>

- + /Cart/
 - + (Número atribuído a unidade)
 - + (numero do cliente)
 - + Link Hash rastreavel do carrinho (ex: [LSsef9u9LV6U6GfkzpN](#))
 - + Name (ex: "Especial" - Label do produto)
 - + Product (ex: "LSrsHNu0MoTCwGZluL1" - Hash do produto)
 - + Shop (ex: "01" - Numero associado a loja)
 - + User (ex: '12' - Número associado ao usuário)
 - + Value (ex: '10.99" - custo do produto)
 - + Total (ex: '5' - Total de priority)
- + /Product/
 - + Hash de produto (Ex: [LSrkyt9reJW7BOU17ap](#))
 - +Priority (São :1,2,3 - Peso do produto para a fila & Pontuação)
 - +Text (Ex:"Espresso" Nome do produto)
 - +Value (Ex:"10,99" -custo do produto)
- + /User/
 - + 1 (adm Super user)
 - + Email
 - + Name
 - + 2 (Cliente)
 - + 3 (Garçom)

Engine 1.0 ou Engine de “peso de fila”

Na base de dados é possível observar que existe a categoria para cada produto como “priority” Este campo representa o peso que o produto tem, e é utilizado pelo engine para otimizar o processo de entrega de produtos. Com os seguintes valores 1, 2, 3. (esses valores serão reutilizados para outra coisa, na próxima seção também)
Quanto menor o produto, mais rápido ele deve em tese ser preparado e entregue. Logo ele tem prioridade perante a fila.

Esse valor é feito no Banco de dados e calculado no “Cart” no banco de dados, a lógica funciona de maneira simples e em todos os cenários.

Cenário 1 - Produto singular

O produto que a pessoa pede já é um fator determinante ao que a pessoa deseja
O cliente que só quer um café, NÃO PODE e NÃO DEVE esperar.
Quando ele solicita um café o seu número no pedido é correspondente à “1” como visto aqui.

Schema

- + /Product/
 - + Hash de produto ([LSrkyt9reJW7BOU17ap](#))
 - +Priority (1)
 - +Text (Espresso”)
 - +Value (4,99”)

Consequentemente ele será priorizado na fila, por seu pedido ser somente um café espresso

“As vezes o cliente fala, mas eu só queria um café”

Rick

Cenário 2 - Acúmulo de produto.

Quando qualquer produto que é somado com outro tem o resultado diferenciando a negociação.

O cliente que pede mais de um produto também tem ideia de que pedir mais de uma coisa afeta o tempo de espera e consequentemente leva mais tempo ou não.

Schema

- + /Cart/
 - + (Número atribuído a unidade)
 - + (numero do cliente)
 - + Link Hash rastreavel do carrinho (ex: [-LSsef9u9LV6U6GfkzpN](#))
 - + Name ("espresso")
 - + Name ("Especial")
 - + Product ("LSrsHNu0MoTCwGZluL1")
 - + Shop ("01")
 - + User ('12')
 - + Value ('20.99')
 - + Total ('4')

Nesse contexto o cliente pediu algo da cozinha e um espresso, a fila dele sai com menor velocidade, e o cliente não está com o tempo curto como prova o pedido. Caso ele esteja pedindo para alguém e quer consumir somente o café, é possível fazer dentro do aplicativo 'rachar' a conta.

Ainda há a necessidade de saber quantas serão as filas para visualizar por parte do time da Benjamin de atendimento.

O valor atribuído de priority foi baseado no tempo de preparo dos itens

Na necessidade do cliente e na simplicidade lógica do banco de dados, tudo isso será testado quando aplicado e alterado caso os números demonstrem algum erro, ou mantido e/ou otimizado caso mostre estar certo. Lembrando que será disponibilizado Analytics para esse tipo de análise, ou exportação de JSON dos resultados do banco do Firebase.

Engine 1.1 Gamificação

Aproveitando a pontuação do peso para a fila, cada produto solicitado na loja gera o mesmo número em pontuação, algumas coisas foram pensadas, mas a estrutura já foi feita ante conversa com outras áreas para adequar e realizar testes com usuário.

A pontuação recebe no banco do usuário +1,+2,+3 proporcional ao pedido, além disso está sendo feito um “convite” via Hash, para que usuário convidem outros usuários a usar o aplicativo e conhecer a Benjamin, além de poder rachar a conta. Dando assim mais uma pontuação para o usuário que realiza o convite.

Essa pontuação possui finalidades diversas, além de usar a gamificação que é mais que provado, aumentar a atuação de clientes com a marca.

Além disso, é interessante para usar como categorizador para clientes, tendo em vista que os clientes que tem menor e maior pontuação devem ser aqueles com maior atuação por parte da Benjamin para fidelizar quem consome menos e também manter o público fiel. Sendo o público do meio o mais generalizado.

Logo no schema basta a leitura de peso dos pedidos usuários para manter o valor e adicionar toda vez que uma hash for utilizada da conta do usuário que recebeu tendo vista que a mesma é rastreando o usuário que enviou.

O resultado sai no JSON ou no proprio Analytics.

O leque de opções é literalmente infinito, e é extremamente válido a participação de outros chefes de area, funcionários para criação de mais coisas.

É Possível já replicar o mesmo modelo para os funcionários para melhorar o trabalho e criar mais empenho.

Carrousel e porquê ele é incrível!

O carrousel é uma das funções mais importantes em aplicativos de pedidos e varejo hoje, porém o da Benjamin terá uma finalidade ímpar, simples e extremamente nobre.

Além de realizar o uma forma mais dinâmica de demonstrar produtos o carrousel da Benjamin é feito em três parte

Produtos com peso 1 de prioridade, como os cafés

Produtos com peso 2 de prioridade, os da vitrine

Produtos com peso 3 de prioridade, os da cozinha ou “especiais”

E por que ele é simplesmente incrível??

O carrousel recebe a lógica mais bonita do sistema, ele vem do banco de dados e não é feita manualmente (Hard Code) usando uma simples foto, e que diferença isso faz? TODA! O produto que vem do Banco de dados recebe uma lógica muito superior que os outros, e qual a lógica? Ele virá flangeado, o que pode aplicar inteligência na hora de subir o mesmo, por isso a lógica é nobre, porque a partir do momento que usamos o reconhecimento de imagem para saber quais produtos tem menos saídas, e ainda por cima maior índice de quebra, não somente economizamos um número absurdo de gastos, como evitamos desperdício de comida dentro da empresa, algo que deveria inclusive ser futuramente louvado para os clientes. Além disso como usamos tipos diferentes de usuários, o carrousel raramente é o mesmo para mais de um, e será moldado mediante estudo de números e poderá vir a receber inteligência artificial, especialmente visto que o cliente que estuda ou trabalha no horário da empresa (cliente ‘notebook’) logo nada mais justo que ‘metralhar’ esses clientes com push notifications para chamá-lo a consumir mais e especialmente os produtos com quebra, melhor qualquer 20% de desconto que o desperdício de produto e de dinheiro.

Caso a empresa demande qualquer mudança da empresa, seja produtos novos, em destaque, sazonais como os de verão poderão ser facilmente subidos. E a lógica é sempre adaptável, futuramente poderá ser gerada a lógica pela inteligência do Watson mediante a leitura de dados de um ano inteiro, e com parâmetros medidos por diversas áreas, como Marketing, Financeiro, Controle de Perdas, Vendas, Estoque e várias outras.

O carrousel irá abranger várias lógicas para as três categorias de produto. Sempre de acordo as testes, resultados do BI e demanda do pessoal “C-Level”

Link para visualizar, funciona já no WEB e Mobile, (caso não esteja funcionando é porque está sendo alterado no momento)

<https://benjamin-a-padaria.herokuapp.com/client/cardapio/600/12>

Navegação

A lógica de navegação é simples, a combinação de PDV com E Commerce além da funcionalidades de rateio de conta e pontuações como no Uber.

Além disso o UX e UI é Material Design, o tipo mais usado no mundo popularizado pelo Google, e já de entendimento maior da população mundial, diferente da híbrida, usada pela Apple.



Segue alguns link. , (caso não esteja funcionando é porque está sendo alterado no momento)

<https://benjamin-a-padaria.herokuapp.com/client/cardapio/600/12>

<https://benjamin-a-padaria.herokuapp.com/client/carrinho/600/1>

Administrador

EXPRESSO DA VITRINE PRA VOCÊ ESPECIAIS MEUS PEDIDOS

PRODUTO	PREÇO	QUANTIDADE	TOTAL
 Especial Descrição simples do produto	R\$ 10.99	1 - +	R\$ 549 ×
 Da Vitrine Para Você Descrição simples do produto	R\$ 8.99	1 - +	R\$ 549 ×
Total		R\$2,346	PEDIR >

BENJAMIN

a padaria



Yellowstone National Park, United States



Expresso

The structured shoulders and sleek detailing ensure a sharp silhouette. Team it with a silk pocket square and leather loafers.

R\$ 3.99

PEDIR



Da Vitrine Para Você

The structured shoulders and sleek detailing ensure a sharp silhouette. Team it with a silk pocket square and leather loafers.

R\$ 8.99

PEDIR



Especial

The structured shoulders and sleek detailing ensure a sharp silhouette. Team it with a silk pocket square and leather loafers.

R\$ 10.99

PEDIR



Da Vitrine pra você



Expresso

Impeccably tailored in Italy from lightweight navy wool.

R\$ 3.99



Da Vitrine Para Você

Impeccably tailored in Italy from lightweight navy wool.

R\$ 8.99



Especial

Impeccably tailored in Italy from lightweight navy wool.

R\$ 10.99



MAIS PRODUTOS

Icebox

Como sempre muitas ideias são feitas no desenvolvimento de um aplicativo, geralmente as colocamos no Icebox. Mais itens poderão ser colocados futuramente, ou tirados etc. Tudo depende da ordem de prioridade baseada no BI da empresa e nas demandas dos diretores.

Listagem

1. conectar o ocr da prateleira para produto com menor ou maior turnover ficar no topo da listagem
2. oferecer ou dar descontos para clientes que estão no salão (Menor pontuação e maior pontuação)
3. usar uma roleta de funcionários para endereçar o atendimento
4. dentro do usuario do garçom avisar o usuário recorrente
5. rankear os garçons e estabelecimento por meio de estrelas dadas aos usuários
6. separar os atendentes em categorias de (se for mais rápido ou se atende bem)
7. Colocar a possibilidade de enviar comentários para o aplicativo para melhor atender e ouvir nossos clientes.
8. Oferecer ao cliente a possibilidade de rachar a conta informando o nome de usuário (funcionalidade Uber) (Clube do café) Sugestão Ale!
9. Enviar convite via hash no app do usuário, pontuar quando o usuário que enviou o convite no momento que o receptor utiliza a hash.
10. Oferecer ao cliente a forma de pedir no caminho e somente retirar no balcão
11. Avisos constantes via push notification de preparo de produto, novidades, promoções inaugurações ou recebimento de créditos.

12. Criar um wallet de arredondamento de troco, para que o usuário não precise demandar dinheiro da loja e que possa ser utilizado em estabelecimento de parceiros.
13. Trabalhar a integração com envio de SMS
14. Criar junto às informações do aplicativo estratégias de pós venda para a empresa.
15. Conectar junto ao OCR a primeira imagem da vitrine sendo feita para saber quando a loja começou a operar.

Resultado Esperado

Com os primeiros testes vêm os primeiros resultados. Dentre eles listaremos primeiramente o resultado esperado por parte do MVP.

1. Diminuição no tempo de espera para clientes com inteligência.
2. Maior identificação cliente loja
3. Compreendimento maior do cliente
4. Mapeamento de cliente
5. 60% de diminuição de casos de clientes que esperam demasiadamente por somente um produto 'café'
6. Possibilidade de eliminação de comanda das lojas e/ou placa de identificação de pedido.
7. Melhor "gather" de dados dos clientes mediante integração com usuários vindos da Linktel

8. Entendimento absoluto de quantas pessoas utilizam do espaço consumindo ou não da Benjamin
9. REDUÇÃO DRÁSTICA DE QUEBRA DE PRODUTOS
10. REDUÇÃO DRÁSTICA DE PERDA DE PRODUTOS (Pós Venda)
11. **Possibilidade de redução do quadro de atendentes**
12. Distribuição mais assertiva de atendimento e de pedidos.
13. Classificação em Real Time das lojas por partes dos clientes
14. Classificação em real time do staff por parte dos clientes
15. Maior agilidade
16. Mais tranquilidade para pedidos
17. Aumento de consumo por parte do Cliente “notebook” mediante envio de push notification
18. Reduzir o número de filas em lojas como a Haddock Lobo, e possibilitar de acomodar os clientes sentados ao invés de em pé e esperando o que as vezes é um pedido rápido.

Entrega de dados

A entrega de dados será no painel de Administrador. Com privilégios de usuário como superusuário, Business Intelligence, Administrador padrão e Visualizador de filas.

Superusuário - tem direito a manipulação de dados alterando estrutura de dados direto do SQL, é concedido somente a pessoa que constrói a ferramenta devido nível de risco de alteração de estrutura de Software ou perda de dados, poderá lançar somente a um usuário no máximo por período estimado de tempo uma licença total de permissão “superusuário”

Business Intelligence - Tem acesso a todo fluxo de dados e controle e manipulação somente das tables, além de ser o único usuário com Analytics, podendo conceder a permissão “business intelligence” a somente um outro usuário por período determinado.

Administrador Padrão - Administrador é a permissão padrão concedida aos gerentes e supervisores e diretoria geral;

Visualizador de filas - usuário normal, concedido aos atendentes e quem faz supervisão do pedido.

Considerações Finais

Este será o primeiro aplicativo feito para a Benjamin, começará como MVP, e será guiada sempre pelos números enviados e pelo setor de BI da Empresa, além de sempre discutida constantemente com a diretoria, e sempre testada em seus moldes iniciais para que não se perca qualquer tempo fazendo algo ou simplesmente se baseando em achismo.

Toda e qualquer funcionalidade deverá ter sua hipótese, documentação. Validação e homologação, lembrando que jamais poderá ser feita uma homologação por qualquer desenvolvedor de software visto que a finalidade mór é sempre a de limitar para quem se vê em posição de “usuário normal” ou qualquer outro tipo de usuário mapeado.

É mais que interessante para a saúde dos aplicativos reuniões com os departamentos de :

Marketing

BI

Financeiro**

Chief Executive Office**

Logística de compras

Controle de perdas e quebras

Lembrando que o Roadmap é mutável mediante ordem da diretoria, e que a versão 1.0 estará sendo lançada na segunda-feira e na terça-feira dias 10/12 11/12, tendo a homologação já marcada para dia 12/12 na quarta-feira.

Depois dessa data será acrescida na mesma documentação os resultados dos testes e possíveis alterações baseadas no mesmo e depois de discutir com a diretoria.

