

FELIPE KOJI WAKU
FELIPE JUN OHIRA OKIDA

**INTEGRAÇÃO DE TÉCNICAS DE INOVAÇÃO E TÉCNICAS DE
ARQUITETURA DE SOFTWARE**

Trabalho de formatura apresentado à
Escola Politécnica da Universidade de
São Paulo para a obtenção do diploma de
Engenheiro de computação.

São Paulo
2016

FELIPE KOJI WAKU
FELIPE JUN OHIRA OKIDA

**INTEGRAÇÃO DE TÉCNICAS DE INOVAÇÃO E TÉCNICAS DE
ARQUITETURA DE SOFTWARE**

Trabalho de formatura apresentado à
Escola Politécnica da Universidade de
São Paulo para a obtenção do diploma de
Engenheiro de computação.

Orientador: Prof. Dr. Jorge Luis Risco
Becerra

São Paulo
2016

AGRADECIMENTOS

À Universidade de São Paulo – USP por toda a sua estrutura e oportunidades.

À Escola Politécnica da Universidade de São Paulo – EPUSP, pelas oportunidades e pela nossa formação.

Ao nosso orientador, Jorge Luis Risco Becerra, pelo suporte, pelas contribuições, correções e incentivo.

À Codus Tecnologia e à Taqtile pelas oportunidades de estágio e todos os ensinamentos.

Aos nossos amigos, pela ajuda em diversas partes do trabalho e apoio.

Aos nossos pais pelo amor, incentivo e apoio incondicional.

RESUMO

Durante anos, a engenharia de *software* estabeleceu práticas e técnicas para o desenvolvimento de *software* com qualidade e de forma organizada. Porém, no mercado, o foco em soluções pode deixar de lado o entendimento do problema no mundo real, o que gera gastos elevados com soluções desnecessárias, sendo essa a causa de que quase metade das empresas fecham. Por outro lado, recentemente houve surgimento de metodologias de inovação que cada vez mais buscam entender os problemas dos usuários de forma mais profunda e próxima, como o *Design Thinking*. Porém essas metodologias não são específicas para *software*, dessa forma algumas das soluções que idealizam nem sempre se integram bem às técnicas de engenharia

Dessa forma, esse projeto consiste em propor um processo para o desenvolvimento de produtos inovadores com foco no usuário baseado em *Design thinking*, utilizando técnicas de arquitetura de *software* e desenvolvimento e além disso, aplicar o processo com o desenvolvimento de um produto voltado para o usuário para restaurantes. O processo é composto pela fase de Inovação que tem o objetivo de entender o usuário e chegar a uma solução que atenda seus problemas ou melhore seus processos utilizando *Design Thinking*, a fase de Arquitetura em que o produto da fase de Inovação será organizado em módulos e camadas com o uso de técnica de arquitetura em três camadas e a fase de Desenvolvimento em que a arquitetura da fase anterior será construído com o uso de metodologias ágeis. O processo se mostrou bastante eficaz para entender os problemas do usuário e chegar à ideias que atendessem às suas necessidades e o grupo pode facilmente desenvolver um protótipo final em pouco tempo e foi atingida uma solução com certo grau de inovação que poderia ser melhorado com futuros incrementos.

Palavras chave: *Design thinking*; Inovação; Arquitetura de *software*

ABSTRACT

For years, the software engineering established practices and techniques to develop software with quality and organized. However, in the industry, the focus in the solution can make one forgets the understatement of the user's problems from the real world, generating elevated costs with unnecessary solutions, being one of the causes which close half of the companies. In the other side, recently some innovation methodologies appeared trying to deeply understand the user's problems like the Design Thinking. But those methodologies are not software oriented and do not always easily integrate with engineering techniques. In that way, this project consists in propose a method to develop innovative products with the focus on the user based on Design Thinking, using software engineering techniques and development and apply this method on the development of a user oriented product to restaurants. The method is composed with the Innovation phase with the objective to understand the user, get a solution that help with his problems or improve his process using Design Thinking, the Architecture phase that the product from the Innovation phase will be organized in modules and tiers using the three-tier architecture and the Development phase that the architecture from the previous phase will be build using agile methods. The method proved to be effective to understand the user's problems and to get ideas that would meet his needs and the group could easily develop a prototype in few time and the group reached a solution with a certain degree of innovation that could be improved in some future work.

Keywords: Design Thinking; Innovation; Software Architecture

LISTA DE ILUSTRAÇÕES

Pág.

Figura 1 - Motor de inovação.....	6
Figura 2 - Fases do Design Thinking.....	8
Figura 3 - Gráfico do modelo de inovação disruptiva	10
Figura 4- Exemplo de arquitetura de uma aplicação organizado em 3 camadas	13
Figura 5 - Exemplo de notação de modelagem de dados em um diagrama UML	16
Figura 6 - Diagrama que representa o sprint no Scrum	20
Figura 7 - Diagrama com o processo de Inovação e Desenvolvimento proposto.....	24
Figura 8 - Diagrama BPMN da fase de Inovação	25
Figura 9 - Diagrama BPMN da fase de Arquitetura	29
Figura 10 - Diagrama BPMN da fase de Desenvolvimento	30
Figura 11 - Agrupamento de ideias obtidos das entrevistas com donos de restaurantes	33
Figura 12 - Agrupamento de ideias obtidos das entrevistas com clientes	35
Figura 13 - Rascunho do wireframe do “É hoje”	38
Figura 14 - Rascunho do wireframe do “The Game”	39
Figura 15 - Telas do protótipo do “É Hoje”	40
Figura 16 - Telas do protótipo do “The Game”	41
Figura 17 - Protótipo do “É hoje” feito no Trello.....	43
Figura 18 - Rascunho da arquitetura de 3 camadas inicial.....	47
Figura 19 - Diagrama entidade relacionamento do sistema	48
Figura 20 - Arquitetura de 3 camadas final.....	49
Figura 21 - Product backlog organizadas no Trello	51
Figura 22 - Funcionalidades do primeiro sprint organizados no Trello	52
Figura 23 - Telas das funcionalidades Listagem de restaurante, detalhe de restaurante e listagem de pratos	53
Figura 24 - Funcionalidades: Detalhe de prato, listagem de avaliações e Criação de avaliação	54

Figura 25 - Funcionalidades do segundo sprint organizadas no Trello	55
Figura 26 - Funcionalidades: Autenticação com Facebook, Gamification e painel do usuário.....	56

LISTA DE TABELAS

	Pág.
Tabela 1 - Resultado dos testes do The Game	86
Tabela 2 - Resultados dos testes do É Hoje	87
Tabela 3 - Resultados da validação com o usuário do produto final	90

SUMÁRIO

Pág.

1. Introdução.....	1
1.1. Motivações	1
1.2. Objetivos.....	1
1.3. Justificativas	2
1.4. Metodologia de projeto de monografia	2
1.5. Estrutura da monografia	3
2. Inovação	4
2.1. Projetos de inovação	4
2.1.1. Motor de inovação.....	5
2.1.2. Design Thinking.....	8
2.1.3. Inovação disruptiva	9
2.2. Arquitetura de software	11
2.2.1. Arquitetura de múltiplas camadas	12
2.2.2. Modelagem de dados.....	14
2.3. Metodologias ágeis.....	16
2.3.1. Scrum.....	18
2.4. Startups	21
3. Desenvolvimento	22
3.1. Processo de Inovação e Desenvolvimento.....	22
3.1.1. Inovação.....	25
3.1.1.1. Empatia.....	26
3.1.1.2. Definição	27
3.1.1.3. Idealização	28

3.1.1.4.	Prototipagem.....	28
3.1.1.5.	Teste	29
3.1.2.	Arquitetura.....	29
3.1.3.	Desenvolvimento.....	30
3.1.4.	Validação e testes.....	31
3.2.	Aplicação do processo	31
3.2.1.	Fase 1 - Inovação.....	32
3.2.1.1.	Empatia e definição.....	32
3.2.1.2.	Idealização	35
3.2.1.3.	Prototipação	37
3.2.1.3.1.	Retrabalho da fase de prototipagem	41
3.2.1.4.	Testes	43
3.2.1.5.	Conclusões da fase.....	45
3.2.2.	Fase 2 - Arquitetura.....	46
3.2.3.	Fase 3 - Desenvolvimento.....	49
3.2.4.	Validação com o usuário	56
3.3.	Resultados do produto	57
3.4.	Operação do processo	58
4.	Conclusões	59
	REFERÊNCIAS	61
	APÊNDICE 1 - Modelo de entrevistas da fase de empatia	65
	APÊNDICE 2 - Resultados das entrevistas da fase de empatia na fase de Inovação	69
	APÊNDICE 3 - Resultados da fase de testes da fase de Inovação.....	85
	APÊNDICE 4 - Resultados da validação com o usuário do produto final	90

1. Introdução

1.1. Motivações

Durante anos, a engenharia de *software* estabeleceu práticas e técnicas para o desenvolvimento de *software* com qualidade e de forma organizada. Porém, no mercado, o foco em soluções pode deixar de lado o entendimento do problema no mundo real, o que gera gastos elevados com soluções desnecessárias. Segundo um estudo do SEBRAE-SP, quase metade das empresas que fecharam não sabiam os hábitos de consumo de seus clientes (SEBRAE-SP, 2016).

Por outro lado, recentemente houve surgimento de metodologias de inovação que cada vez mais, buscam entender os problemas dos usuários de forma mais profunda e próxima, como o *Design Thinking*. Porém essas metodologias não são específicas para *software*, dessa forma algumas das soluções que idealizam nem sempre se integram bem às técnicas de engenharia (Häger, 2015).

Nesse contexto, é preciso uma forma de unir esses dois domínios, do problema e da solução, no desenvolvimento de *software* para poder explorar ao máximo os benefícios e mitigar os pontos fracos de ambos os mundos.

1.2. Objetivos

Visto as motivações do grupo para a realização do trabalho, o trabalho consiste em propor um processo para desenvolvimento de produtos inovadores com foco no usuário baseado em *Design Thinking*, utilizando técnicas de arquitetura de *software* e desenvolvimento.

O trabalho consistirá na pesquisa de técnicas de inovação, arquitetura e desenvolvimento, integrar as técnicas de *Design Thinking* com arquitetura de *software* como arquitetura de múltiplas camadas e na aplicação do processo com o desenvolvimento de um produto voltado para o usuário para restaurantes.

1.3. Justificativas

De acordo com Vinnakota em *Integration of Design Thinking with Strategy and Innovation in an Enterprise Context*, a forma tradicional de levantar requisitos de *software* é uma forma de pensamento analítico. Essa forma de pensar é muito boa para resolver problemas lógico, centrada em uma estratégia, mas para se inovar é necessário um pensamento mais centrado no objetivo (Vinnakota, 2014), uma maneira que seja possível entender melhor as necessidades dos usuários como o *Design Thinking*.

Para Häger em *DT@Scrum: Integrating Design Thinking with Software Development Processes*, o *Design Thinking* oferece diversas ferramentas para entender o usuário e chegar a soluções que atendam às suas necessidades, entretanto tornar essas ideias em produtos, serviços ou *software* existem diversos desafios de integração (Häger, 2015). Dessa forma aqui será apresentada uma forma de realizar essa integração a partir de técnicas de arquitetura e desenvolvimento de *software*.

Para os executores desse projeto, o trabalho oferece uma oportunidade para estudar técnicas de inovação e realizar um experimento de integrar-las com técnicas de desenvolvimento de *software*.

Para a Escola e sociedade, o trabalho é uma contribuição para os empreendedores e inovadores que desejam revolucionar o mundo com novos produtos, oferecendo uma jornada na aplicação de técnicas de inovação e desenvolvimento e a sua integração.

Para a Engenharia de Computação, o trabalho oferece uma nova forma de realizar a análise de requisitos de *software*, que antes era voltado no uso do sistema, enquanto aqui é voltado às necessidades do usuário.

1.4. Metodologia de projeto de monografia

O projeto de monografia foi executado nos seguintes passos:

Levantamento bibliográfico das motivações e justificativas e traçado um objetivo para o projeto.

Pesquisa de técnicas de inovação como Motor de inovação e *Design Thinking*, técnicas de arquitetura de *software* e técnicas de desenvolvimento ágil.

Aplicação das técnicas de inovação e documentação de seus resultados.

Aplicação das técnicas de arquitetura de *software* e documentação de seus resultados.

Desenvolvimento do produto, seus testes e documentação dos resultados.

Análise dos resultados do projeto como um todo e documentação da conclusão.

1.5. Estrutura da monografia

O trabalho foi estruturado da seguinte maneira:

O capítulo 1 apresenta as motivações do trabalho, seus objetivos e as suas justificativas.

O capítulo 2 apresenta uma revisão dos principais conceitos utilizados no processo proposto e na aplicação. São abordados temas de inovação, arquitetura de *software*, métodos de desenvolvimento ágil e *startups*.

O capítulo 3 apresenta o processo utilizado, a sua aplicação no desenvolvimento de um produto e os aprendizados do grupo.

O capítulo 4 conclui o trabalho oferecendo reflexões do processo aplicado e observações para trabalhos futuros.

2. Inovação

Neste capítulo serão apresentados os conceitos que serão utilizados no desenvolvimento do processo e uma visão de produtos inovadores para *startups*.

Os conceitos estão separados em quatro tópicos: projetos de inovação em *software*, técnicas de arquitetura de *software*, métodos ágeis e *startups*.

O tópico sobre projetos de inovação aborda conceitos de gerais de inovação e técnicas para conhecer o usuário e chegar a um projeto inovador como *Design Thinking*.

O tópico de arquitetura de *software* aborda técnicas para organizar um sistema, será detalhada a técnica de arquitetura em múltiplas camadas e modelagem de banco de dados.

O tópico sobre metodologias ágeis aborda técnicas para o desenvolvimento de *software* e gestão de projetos, sendo detalhado o método Scrum.

O tópico sobre *startups* aborda um pouco o conceito e detalha brevemente as *fintechs*.

2.1. Projetos de inovação

Inovação é o ato ou processo de aprimorar uma pesquisa, conhecimento ou prática a partir da introdução ou aplicação de novas ideias, dispositivos ou métodos para resolver problemas ou criar novas oportunidades não que existiam antes. (Setser, 2015)

Um projeto é um empreendimento temporário para criar um produto único, serviço ou resultado. A natureza temporária indica que ele tem um fim. Esse fim pode ser atingido quando um projeto chega ao seu objetivo, o projeto foi encerrado por não ser possível alcançar o objeto ou não existe mais a necessidade do projeto existir. (Project Management Institute, 2013)

A partir das definições de inovação e projeto pode-se concluir que um projeto de inovação é um projeto com o objetivo de criar um novo produto ou serviço que

aprimore pesquisa, conhecimento ou prática a partir da introdução ou aplicação de novas ideias, dispositivos ou métodos para resolver problemas ou criar novas oportunidades.

Para se ter ideias que levam à inovação é necessário algum método ou modelo para organizar o pensamento, dessa forma será apresentado o modelo de Tina Seelig, o motor de inovação (Seelig, 2012) e o método *Design Thinking*.

Enquanto a inovação em empresas já estabelecidas são pequenas melhorias que entregam o valor de formas levemente melhoradas, as novas empresas criar novas formas de levar valor para seus clientes e quebrando a cadeia de valor tradicional, gerando a disrupção inovadora. (Christensen, 1995)

2.1.1. Motor de inovação

Motor de inovação é um modelo proposto por Tina Seelig em 2012 em seu livro *inGenius* que une os principais fatores que levam à inovação. Uma imagem do modelo pode ser visto na figura 1 – Motor de inovação.

Figura 1 - Motor de inovação



Fonte: Tina Seelig (2012)

O modelo é composto por seis partes, sendo três internas e três externas. As internas são a atitude, conhecimento e imaginação, que são de responsabilidade da pessoa que deseja inovar, enquanto as três externas, habitat, cultura e recursos se referem ao mundo externo.

O conhecimento é o domínio que a pessoa tem de uma certa área de conhecimento como música, matemática, computação, etc, esse é o combustível para a imaginação. Se uma pessoa deseja inovar em uma determinada área é indispensável que ela a conheça muito bem, por exemplo, se uma pessoa deseja criar um novo produto que revolucione os processos de uma área de tecnologia da informação é necessário que ela conheça o funcionamento atual e as necessidades para que sua nova ideia seja adequada.

A imaginação é a habilidade de criar algo novo, sem ela novas ideias não podem ser geradas. Essa habilidade inclui a capacidade de combinar ou conectar ideias, reformular problemas e desafiar premissas, essas técnicas permitem às pessoas que elas criem novas ideias.

A atitude é a faísca que inicia o processo criativo, sem ela não é possível chegar a ideias inovadoras. São a atitude e a mentalidade que determinam como a pessoa interpretará e responderá as situações.

Os recursos são o que está disponível de valor no ambiente, podendo vir de diversas formas como recursos financeiros, naturais, materiais, etc. Os recursos e o conhecimento estão lado a lado no motor de inovação pois a quantidade de recursos que podem ser mobilizados é proporcional ao conhecimento da pessoa, por exemplo, quanto mais uma pessoa conhece sobre pesca mais ela irá pescar.

O habitat é o local onde a pessoa está e sua posição no motor de inovação está localizado ao lado da imaginação pois o habitat influencia na maneira como as pessoas pensam e na habilidade criativa delas.

A cultura é a forma em que as pessoas percebem, interpretam e entendem o mundo ao redor delas. A cultura está ao lado do atitude pois mesmo um pequeno número de pessoas mudem a sua atitude então a cultura muda.

Dessa forma, podemos ver a atitude aviva a curiosidade para adquirir novos conhecimentos, este é o combustível para a imaginação que gera novas ideias. A imaginação catalisa a criação de novos habitats, o que aumenta a disponibilidade de recursos para o ambiente e esses habitats junto com a cultura influenciam a cultura de uma comunidade.

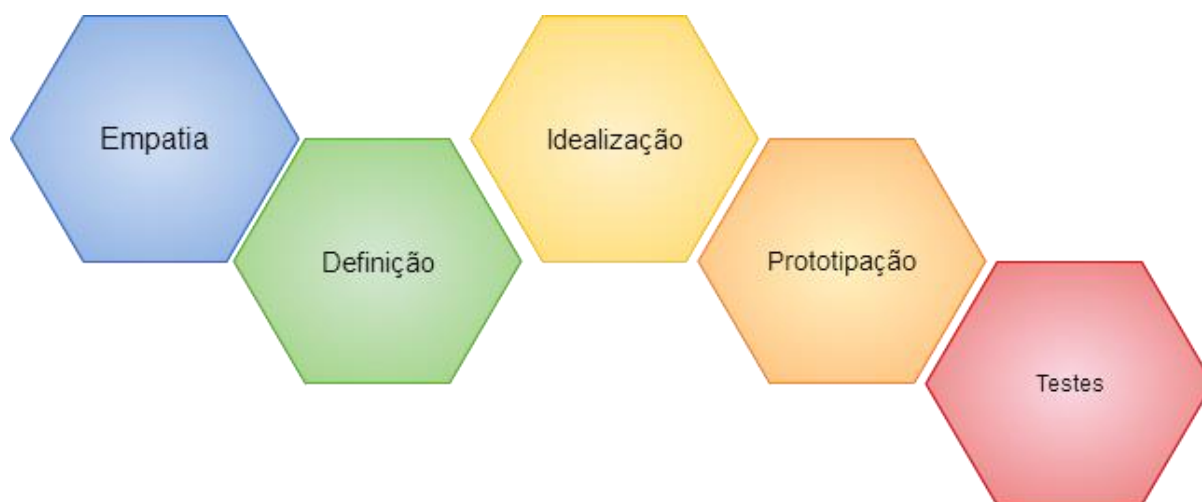
O motor de inovação é uma nova forma de ver o mundo, de forma que a pessoa que deseja inovar deve ter em mente como modelar seu contexto com o motor, ou seja, o inovador deve ter em mente o seu habitat, cultura, quantos recursos é capaz de mobilizar com seu conhecimento e com atitude ele é capaz de chegar a ideias inovadoras com sua criatividade.

2.1.2. Design Thinking

Design Thinking é um processo com um conjunto de técnicas com o objetivo de entender os problemas do usuário e chegar à uma solução. A origem do processo vem desde de 1960 e 70 do movimento chamado design cooperativo em que consistia da ideia dos usuários de *software* participarem do processo de desenvolvimento e esse movimento inspirou outros posteriores como co-design ou design participativo até que em 1980 e 90 em que aspectos cognitivos de design foram adicionados à solução dos problemas e o pesquisador Richard Buchanan popularizou em 1990 o termo Design Thinking em seu artigo chamado *Wicked Problems in Design Thinking* como uma metodologia para chegar a novas soluções em diversas áreas a partir do entendimento do usuário. (Archer, 2016)

Esse é um método usado por grandes empresas como a Google (Knapp, 2016) e foi amplamente divulgado pela organização IDEO. Suas fases podem ser vistas na imagem 2 – Fases do *Design Thinking*:

Figura 2 - Fases do *Design Thinking*



Fonte: Husso Plattner (2010)

A fase de empatia tem o objetivo de entender o contexto do usuário, seus objetivos, dores, atividades a partir de atividades como observação, entrevistas ou imersão.

A fase de definição analisa as informações obtidas na empatia e busca entender os porquês, procurar por padrões, problemas ou oportunidades.

A fase de idealização procura por soluções e ideias variadas para solucionar os problemas ou desenvolver oportunidades encontrados na fase anterior.

A fase de prototipação busca implementar de forma rápida e barata as ideias da fase anterior a fim de testá-las.

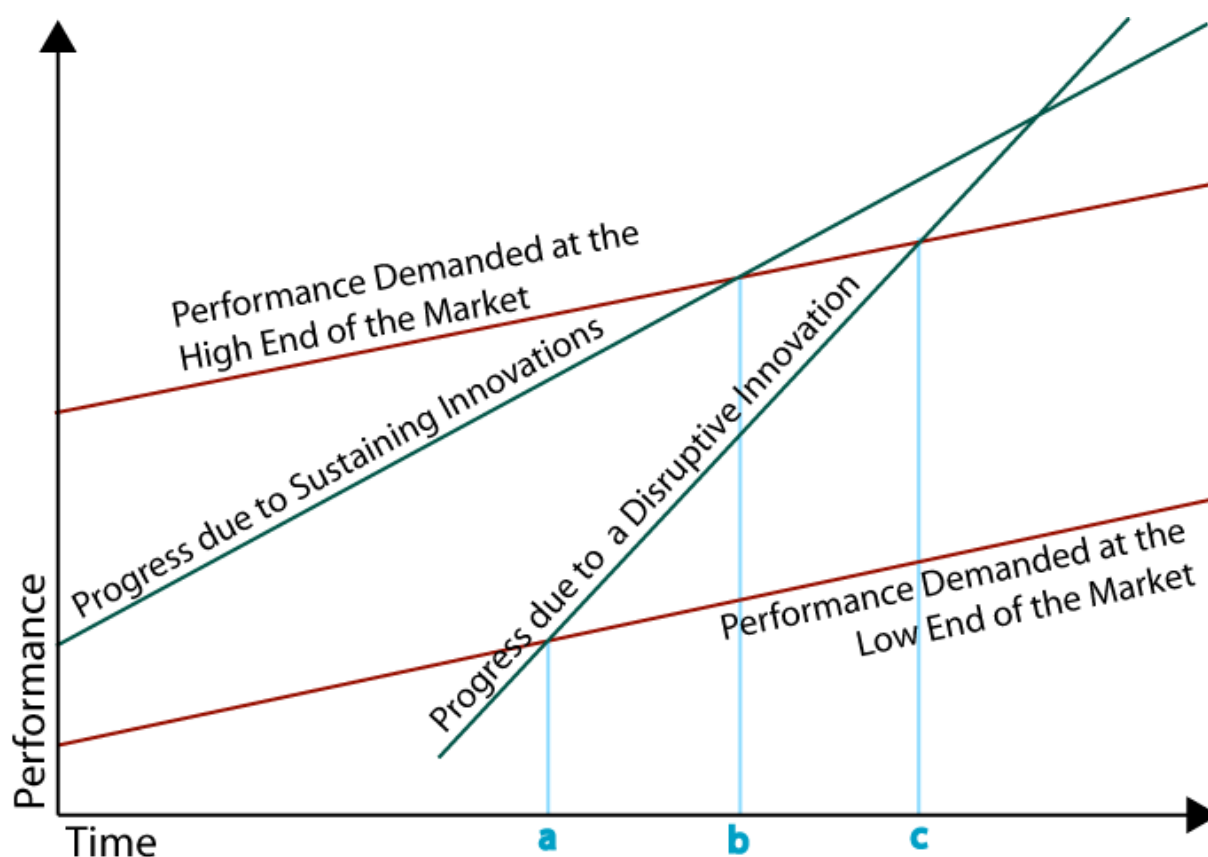
A fase de testes tem o objetivo de testar os protótipos criados na fase anterior, apresentando o protótipo e coletar o *feedback* de forma a entender como a solução impacta na rotina do usuário.

No final da última fase é concluído um ciclo do processo de *Design Thinking*, então é possível recomeçar o ciclo com os dados coletados dos testes e agora chegar à uma solução mais adequada para ele. (Plattner, 2010)

2.1.3. Inovação disruptiva

Clayton M. Christensen define disrupção como o processo em que uma pequena empresa com menos recursos consegue desafiar com sucesso uma empresa bem estabelecida. De acordo com o autor, empresas já estabelecidas possuem um produto ou serviço e geralmente fazem pequenas melhorias nele para atender à um público que traz a eles o maior lucro e geralmente ignoram outros público. Enquanto uma nova empresa geralmente oferece um novo produto que geralmente custam menos e são mais simples, mas atendem melhor à um certo público alvo. Quando os usuários começam a usar o novo produto e essa nova empresa começa a ganhar maior volume, então a disrupção ocorre (Christensen, 1995). A diferença de uma tecnologia tradicional e uma disruptiva podem ser vistas na figura 3, o modelo de inovação:

Figura 3 - Gráfico do modelo de inovação disruptiva



Fonte: Clayton M. Christensen (2015)

A figura 3 contrasta a inovação das grandes empresas (*Progress due to sustaining innovations* no gráfico), a inovação disruptiva (*progress due to a disruptive innovation* no gráfico), a demanda do mercado de ponta (*Performance demanded at the high end of the Market* no gráfico) e a demanda do mercado de baixo nível (*Performance demanded at the low end of the Market* no gráfico). Podemos ver que os produtos evoluem mais rápido que a demanda do mercado, o que implica em preços mais altos e mais do que a maioria dos consumidores deseja pagar. Então a tecnologia disruptiva, que inicialmente é menos eficiente, consegue atender a um número maior de usuários com um custo menor e logo poderá atingir a um número maior de usuários, se tornando competitivo à tecnologia líder (Christensen, 1995).

Chama-se atenção na figura 3 os pontos de intersecção das curvas: o ponto em que a demanda do mercado de ponta e da inovação de grandes empresas se

encontram é o ponto em que esses produtos começam a atender mais do que os usuários necessitam (ponto B no gráfico da figura 3), o ponto em que a performance do produto disruptivo ultrapassa a demanda do mercado de baixo nível, ou seja, começa a atender todas as necessidades básicas (ponto A no gráfico da figura 3) e o ponto em que a performance do produto disruptivo ultrapassa a demanda do mercado de ponta, em que ele atende à todas as demandas do mercado mais exigente (ponto C no gráfico da figura 3). (Christensen, 1995)

Um caso emblemático na indústria de entretenimento é a disrupção que a Netflix causou na Blockbuster na última década. A Blockbuster sempre foi líder no mercado de entretenimento com o modelo de negócio de aluguel de filmes e a Netflix surgiu em 1997 com um novo modelo de negócio com o uso de assinaturas em que os seus clientes receberiam filmes pelo correio, entretanto o correio americano não era muito eficiente e filmes demoravam dias para chegar aos clientes. Porém a Netflix pode mudar o mercado com o surgimento de novas tecnologias que permitiu que eles criassem um serviço de *streaming* de filmes que lhe garantiu sucesso e se tornarem hoje líderes do mercado. (Christensen, 2015)

2.2. Arquitetura de software

O IEEE define no livro SWEBOOK 15 áreas do conhecimento na Engenharia de *Software*. Entre essas áreas temos os requisitos de *software*, design de *software*, construção de *software*, testes de *software*, manutenção de *software*, gerenciamento de configuração de *software*, gerenciamento de engenharia de *software*, processos de engenharia de *software*, modelos e métodos de engenharia de *software*, qualidade de *software*, práticas de engenheiros de *software*, economia da engenharia de *software*, fundamentos computacionais, fundamentos matemáticos e fundamentos da engenharia. (SWEBOOK, 2014)

A área de conhecimento de design de *software* contempla o processo de definição da arquitetura, componentes, interfaces e outras características de um sistema ou componente. O resultado do processo de design de *software* descreve a arquitetura de um *software*, como este é decomposto e organizado em componentes

e a interface entre os componentes, descrevendo em detalhes que permitam a sua construção. (SWEBOK, 2014)

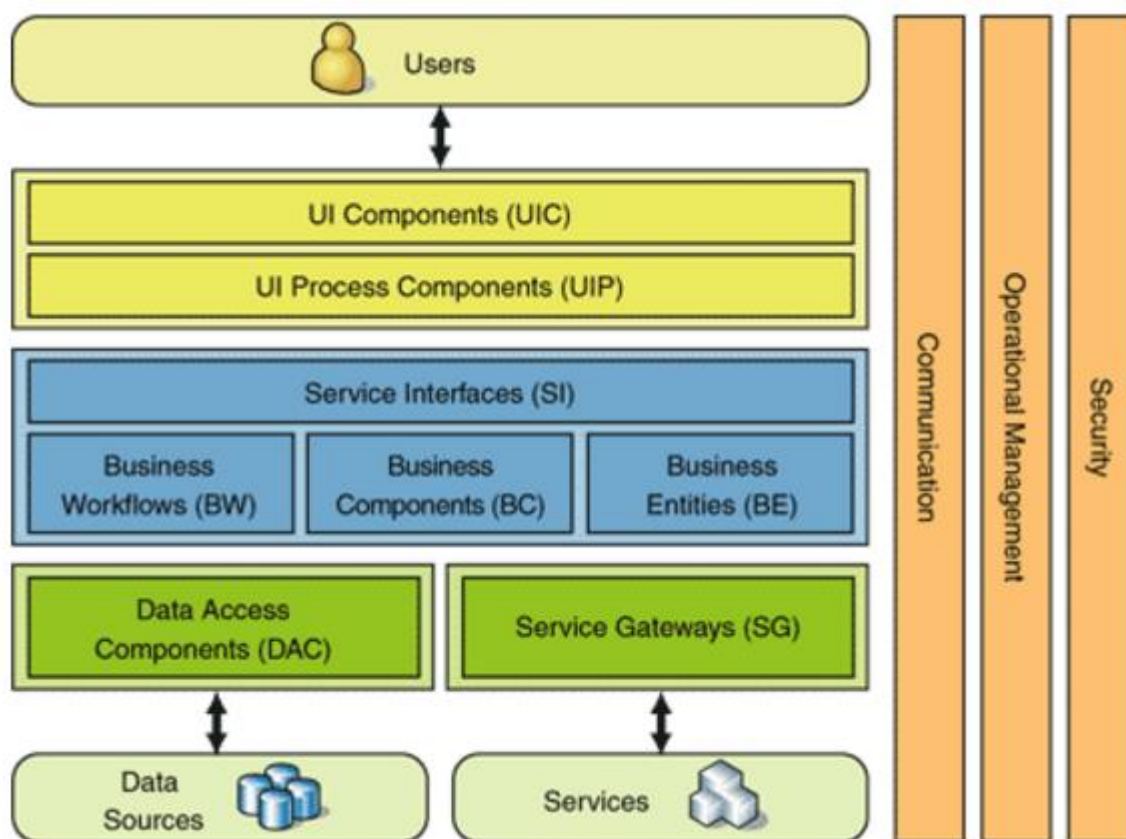
Para se obter uma arquitetura organizada é necessário o uso de técnicas de forma a organizar os módulos do sistema como a técnica de arquitetura de multi-camadas e técnicas de modelagem de dados que serão detalhadas com maior profundidade nos tópicos seguintes.

2.2.1. Arquitetura de múltiplas camadas

A técnica de arquitetura de múltiplas camadas é uma técnica usada por grandes empresas como a Microsoft que visa organizar os diversos módulos lógicos, componentes de código que realizam uma função, em camadas que oferecem um conjunto coeso de serviços e essas camadas possuem uma comunicação explícita e de baixo acoplamento, sendo geralmente mais utilizada um modelo de três camadas. (Microsoft Patterns & Practices Team, 2009 e Clements, 2010)

Em uma arquitetura de três camadas, geralmente são separadas as camadas de acesso à dados, negócio e apresentação (Microsoft Patterns & Practices Team, 2009). Um exemplo desse estilo de arquitetura pode ser visto na figura abaixo:

Figura 4- Exemplo de arquitetura de uma aplicação organizado em 3 camadas



Fonte: Microsoft Patterns & Practices Team, 2009

No exemplo da figura 4 é possível ver as camadas. A camada de apresentação está representada pelos componentes de interface para o usuário (*UI components*) e pelos componentes de interface de processos (*UI Process Components*), a camada de negócio está representada pela interface dos serviços (*Service Interfaces*) e seus componentes como fluxo de trabalho de negócio (*Business Workflow*), componentes de negócio (*Business Components*) e entidades de negócio (*Business Entities*) e a camada de acesso à dados está representada pelos componentes de acesso à dados (*Data Access Components*) e pelos gateways de serviços (*Service Gateways*).

A camada de apresentação tem a responsabilidade de ser a interface para o usuário, sendo componentes de interface como formulários e suas regras de validação. (Microsoft Patterns & Practices Team, 2009)

A camada de negócio contém as regras de negócio do sistema, contendo os processos e componentes do negócio. Nesta camada ficam as principais funcionalidade do sistema e encapsula as lógicas relevantes para o negócio. No exemplo acima essa camada apresenta os componentes de negócio que são as implementações de *software* que encapsulam as regras de negócio, os fluxos de trabalho do negócio representam os principais processos de um negócio, as entidade de negócio são containers de dados, responsáveis por encapsular dados para a camada de acesso à dados e as interfaces de serviço são as interfaces das regras de negócio para a camada de apresentação. (Microsoft Patterns & Practices Team, 2009)

A camada de acesso à dados é responsável por expor os dados armazenados para a camada de negócio. Os componentes de acesso a dados são componentes que isolam a camada de negócio das funções de acesso aos bancos de dados e os *gateways* de serviço funcionam como componentes que encapsulam a lógica para acessar serviços externos. (Microsoft Patterns & Practices Team, 2009)

Além dos componentes de *software* nas três camadas também temos os serviços de fundação como segurança, operacional e comunicação. O serviço de segurança é responsável por manter a aplicação segura, o operacional é o gerenciamento de recursos para garantir escalabilidade e disponibilidade dos serviços e a comunicação é o serviço que permite a transmissão de mensagens entre componentes. (Microsoft Patterns & Practices Team, 2009)

Esse estilo de arquitetura contém diversos benefícios como abstração dos componentes, isolamento das camadas, reusabilidade de componentes e alta testabilidade. (Microsoft Patterns & Practices Team, 2009)

2.2.2. Modelagem de dados

A modelagem dos dados é uma atividade comum no desenvolvimento de *software*. Nessa atividade é feito um modelo de dados em que a estrutura das informações estáticas são descritas em termos de entidades e relacionamentos.

Esse modelo geralmente é representado de maneira gráfica por meio de diagramas de entidade-relacionamento. (Clements, 2010)

Um elemento em um modelo de dados representa um objeto que guarda informação que precisa ser armazenada ou representada no sistema. Estes contém propriedades que podem ser nomes, atributos, chaves e regras que garantem aos usuários permissões de acesso às entidades. (Clements, 2010)

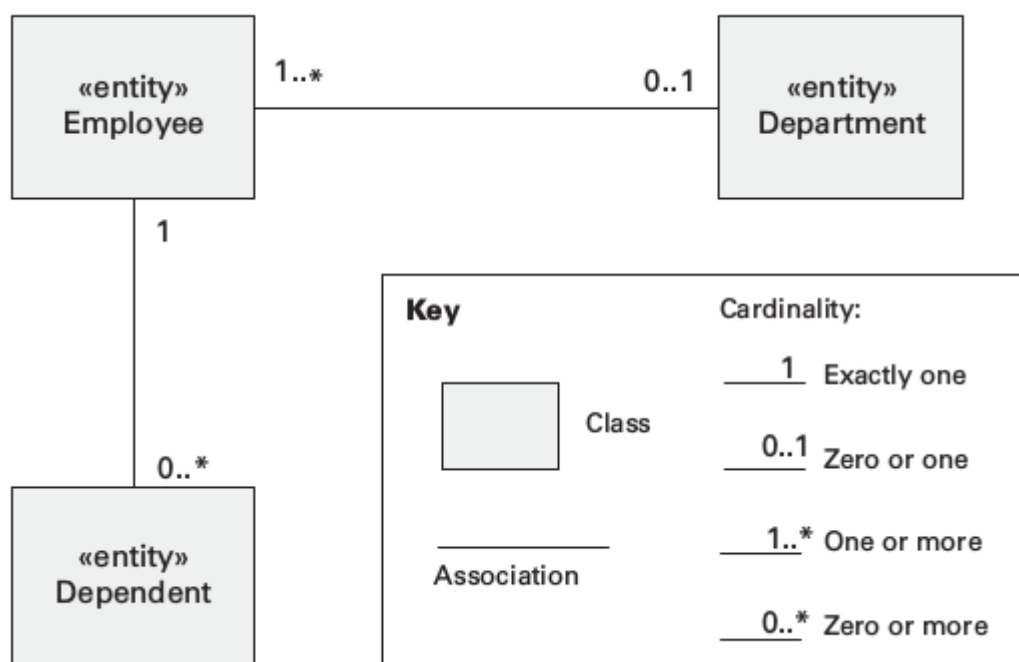
Um relacionamento representa uma associação lógica entre entidades, podendo ser uma para uma, uma para várias ou várias para várias. (Clements, 2010)

Uma boa modelagem de dados garante para a equipe de desenvolvimento uma planta para a implementação das entidades de dados em um banco de dados relacional. Além disso, uma boa modelagem também pode atingir ganhos de performance devido à arquitetura de dados escolhida. (Clements, 2010)

O arquiteto deve entender quais tipos de operações serão realizadas para acessar os dados, estas podendo ter grande impacto na performance do sistema, por exemplo, as operações de união de duas entidades são geralmente muito custosas em relação ao desempenho, portanto, deve ser feita uma modelagem de dados que evite essa operação. (Clements, 2010)

Um exemplo de notação de modelagem de dados pode ser visto abaixo:

Figura 5 - Exemplo de notação de modelagem de dados em um diagrama UML



Fonte: (Clements, 2010)

O diagrama da figura 5 é um exemplo de um modelo simplificado de um sistema de recursos humanos em que temos as entidades representadas pelos empregados, dependentes e departamentos e relacionamentos entre as entidades podendo ser um para um, zero ou um, um ou mais ou zero ou mais.

2.3. Metodologias ágeis

De acordo com Pressman, hoje as condições do mercado estão mudando muito rapidamente, as necessidades do usuário evoluindo e novos competidores podem surgir a qualquer momento. Nesse tipo de ambiente não é eficiente definir todos os requisitos do sistema antes de iniciar um projeto. Dessa forma, as pessoas devem ser ágeis o bastante para responder à essas mudanças. Entretanto o custo dessas mudanças é elevado, então os métodos ágeis surgem para reduzir esses custos. (Pressman, 2014)

Tornando-se popular em 2001, quando dezessete especialistas em processos de desenvolvimento de software se juntaram e decidiram publicar o Manifesto Ágil (SOARES, 2004), com 12 princípios, resumindo os conceitos chaves das metodologias ágeis:

1. “Nossa maior prioridade é satisfazer os clientes por meio de entregas rápidas e contínuas de valor de *software*;
2. “Aceitamos as mudanças de requisitos, até mesmo nas últimas fases do desenvolvimento. Processos ágeis aceitam a mudança para que o cliente tenha a maior vantagem competitiva;
3. Entregamos partes funcionais de *software* regularmente, de algumas semanas até meses, de preferência em períodos de tempo curtos;
4. Executivos e desenvolvedores devem trabalhar juntos diariamente durante todo o projeto;
5. Construimos projetos ao redor de pessoas motivadas. Dê a eles o ambiente e o suporte necessário e confie que o trabalho será entregue;
6. As maneiras mais eficientes e efetivas de convergir informação em um time de desenvolvimento são com conversas cara a cara;
7. *Software* funcionando é a principal métrica de progresso;
8. Processos ágeis promovem o desenvolvimento sustentável. Os patrocinadores, desenvolvedores e usuários devem ser capazes de manter um ritmo constante indefinidamente;
9. Atenção contínua à excelência técnica e bom design melhoram a agilidade;
10. Simplicidade, a arte de maximizar o trabalho não feito, é essencial;
11. A melhor arquitetura, requisitos e design surgem por meio de times auto organizados;
12. Em intervalos regulares, o time deve refletir em como ser mais eficiente, então ajustar seu comportamento de acordo.” (Beck et al., 2001)

Desses princípios surgiram diversas formas de se desenvolver *software* de maneira ágil como o Scrum.

2.3.1. Scrum

O Scrum é um *framework* desenvolvido por Ken Schwaber e Jeff Sutherland no começo da década de 90, para desenvolvimento e sustentação de produtos complexos, utilizando papéis, eventos, artefatos e regras para isso. (Schwaber, 2016)

O Scrum se baseia no processo empírico, em que o conhecimento vem da experiência e do ato de fazer as decisões com o que já se sabe. Ele também se baseia em três pilares: transparência, inspeção e adaptação. (Schwaber, 2016)

A transparência diz que o processo deve ser visível à todos os responsáveis pelo produto final, dessa forma todos têm um conhecimento comum do que pode ser visto. (Schwaber, 2016)

A inspeção é analisar como o trabalho está sendo realizado e procurar por deficiências. É importante que a inspeção nunca atrapalhe a maneira como o trabalho deve ser realizado. (Schwaber, 2016)

A adaptação é o time se ajustar o mais cedo possível para evitar futuros problemas. (Schwaber, 2016)

O Scrum é composto por um time scrum, com seus respectivos papéis, eventos, artefatos e regras. Cada um dos componentes serve à um propósito que é o uso do Scrum com sucesso. (Schwaber, 2016)

O time scrum é composto pelo *Product Owner*, time de desenvolvimento e pelo Scrum Master. (Schwaber, 2016)

O *Product Owner* é responsável pela maximização do valor do produto e o do trabalho do time de desenvolvimento. Ele também é responsável por gerenciar o *Product Backlog*, especificando todos os seus itens, ordenando eles de forma que se atinja o objetivo e missão, garantir que ele seja transparente a todos do time e o que deve ser executado em seguida e garantir que o time de desenvolvimento entenda ele no nível necessário. (Schwaber, 2016)

O *Product backlog* é um artefato do Scrum, é uma lista ordenada de tudo que pode ser necessário para o produto e a única fonte dos requisitos do produto. Nele contém uma lista de todas funcionalidades, funções, requisitos, melhorias e

correções que irão contribuir para um futuro lançamento do produto. (Schwaber, 2016)

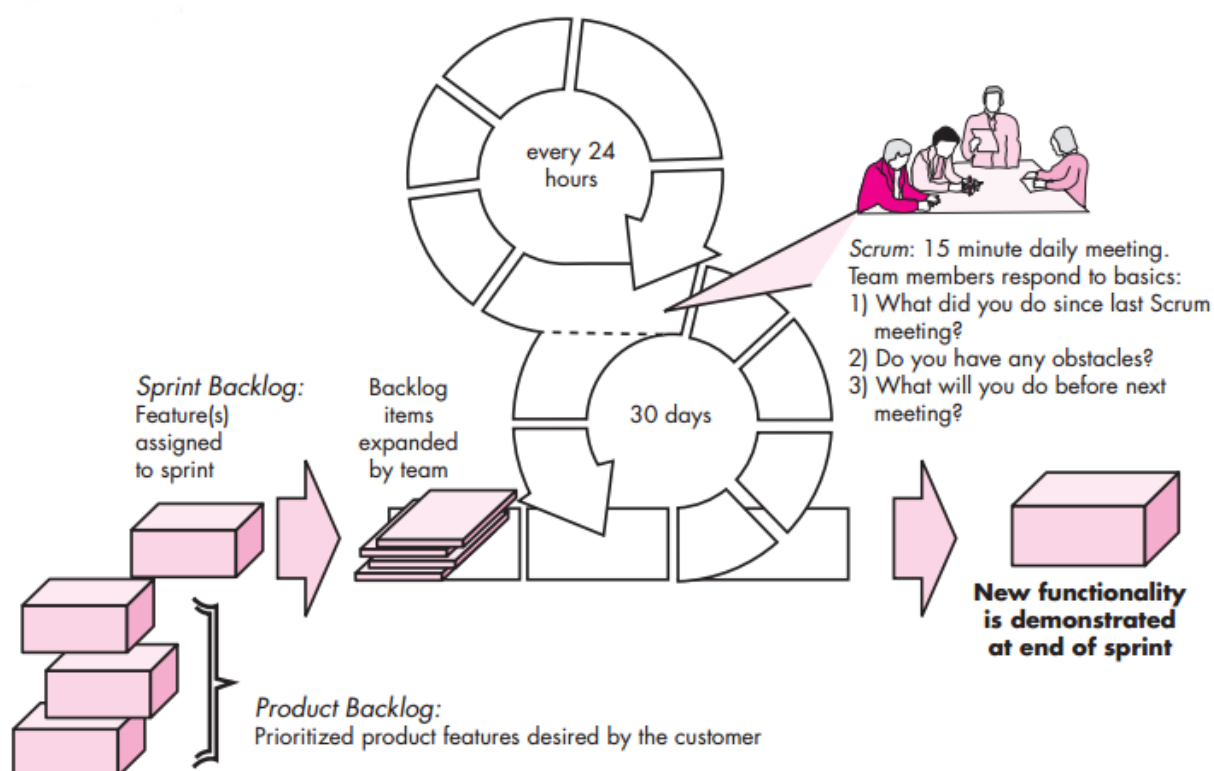
O time de desenvolvimento consiste nos profissionais que irão trabalhar na entrega de um incremento para um potencial lançamento no final de cada *sprint*. O time tem algumas características como auto organizado, multifuncional, não pode ser reorganizado em times menores e cada membro pode ter sua especialização, mas todos pertencem ao time de desenvolvimento como um todo. (Schwaber, 2016)

O *Scrum Master* é o membro que garante que todas as práticas de Scrum estão sendo seguidas corretamente. Ele serve ao time scrum e na interação das pessoas de fora do time com o time. O seu objetivo é maximizar o valor criado pelo time. (Schwaber, 2016)

O Scrum possui uma série de eventos para minimizar o número de reuniões não especificadas no Scrum. Os eventos são o *sprint*, o *sprint planning*, o *daily scrum*, o *sprint review* e a retrospectiva. (Schwaber, 2016)

Schwaber define o *sprint* como o coração do scrum. Ele é um período de tempo, geralmente um mês ou menos em que uma parte incremental, pronta, usável e potencial do produto será lançada. (Schwaber, 2016) O diagrama abaixo representa um *sprint*:

Figura 6 - Diagrama que representa o *sprint* no Scrum



Fonte: Pressman (2014)

Na figura 6, diagrama que representa o *sprint* no scrum, pode-se ver que no processo de 30 dias uma parte do *product backlog* é retirada e se transforma no *sprint backlog* que é o que será entregue no final do *sprint* e no final do *sprint* temos uma nova parte do produto entregue. Em cada um dos dias é feita uma reunião rápida em que irá ser reportado o que foi feito, os problemas e o que fazer no dia, a reunião é chamada de *daily sprint*. (Schwaber, 2016)

O *sprint planning* é uma reunião que se realiza antes do *sprint* em que será definido o que será feito no *sprint*. (Schwaber, 2016)

O *sprint review* é uma reunião que ocorre no final do *sprint* em que será analisado o produto entregue no final e são feitas adaptações no *product backlog* se necessário. (Schwaber, 2016)

A retrospectiva é uma oportunidade para o time de scrum se analisar e criar um plano para melhorar no próximo *sprint*. Essa reunião ocorre antes do próximo

sprint planning e o time analisa o que deu certo no último *sprint* como ferramentas e processos e identifica os maiores problemas que tiveram. (Schwaber, 2016)

Em todas os eventos do scrum podemos ver os pilares do Scrum, nas *dailies* e no *review* podemos ver a transparência e na retrospectiva vemos a inspeção e adaptação.

2.4. Startups

Eric Ries define no livro *The Lean Startup* que uma *startup* é uma instituição humana feita para criar um novo produto ou serviço em condições de extrema incerteza. Importante ver que nessa definição não é citado o tamanho da empresa ou do setor de atuação. (Ries, 2011)

Dada a definição é visto que uma *startup* não é sobre um produto ou uma tecnologia disruptiva ou uma ideia, ela é maior do que todos esses, ela é uma empresa humana. (Ries, 2011)

Startups usam diversas formas de inovação, criando novos produtos, novas tecnologias, novos modelos de negócio ou levar um produto ou serviço para um novo segmento de cliente, todos esses estão no centro do sucesso de uma empresa. (Ries, 2011)

Startups podem atuar em qualquer mercado, por exemplo, no mercado financeiro existem hoje as *fintechs*, empresas que unem tecnologia com mercado financeiro. Essas empresas hoje estão causando uma disrupção no mercado financeiro.

As *fintechs* fazem uso intensivo de tecnologia em seus serviços em produtos e geralmente são mais intuitivos e estão disponíveis em tempo integral. (Horn, 2016)

Exemplo de uma *fintech* brasileira hoje é a Nubank que propõe um serviço financeiro inovador com o uso de tecnologia e *data science* e atingiu o 16^a lugar como uma das 100 melhores *fintechs* do mundo. O serviço oferecido por eles é uma administradora de cartões de crédito sem taxas e burocracias. (Nubank, 2016)

3. Desenvolvimento

Este capítulo descreve o processo que será utilizado no trabalho, a aplicação deste no desenvolvimento em um produto, alguns aprendizados que o grupo teve durante a execução e o resultado do produto final.

O capítulo está dividido em seis tópicos:

O primeiro tópico apresenta as três fases do processo, descrevendo seus objetivos, técnicas que serão utilizadas e produto a ser gerado no final, como será feita a validação com o case em que o processo será aplicado;

O segundo, terceiro e quarto tópicos apresentam como foi a aplicação das técnicas, os resultados obtidos em cada fase e aprendizados;

O quinto tópico apresenta como foi feita a validação com o usuário;

O sexto tópico apresenta uma análise dos resultados do produto;

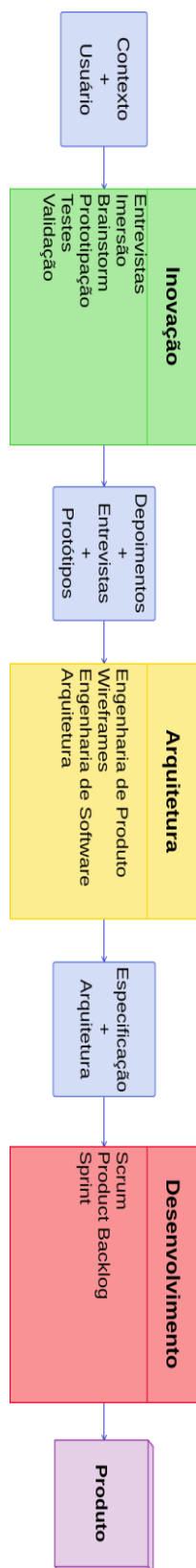
O sétimo tópico apresenta algumas reflexões para projetos futuros baseados nos aprendizados do grupo e uma breve discussão sobre o uso do processo em empresas.

3.1. Processo de Inovação e Desenvolvimento

Com o intuito de unir as técnicas de Design Thinking, necessárias para entender o usuário, com as técnicas de engenharia de *software* para desenvolvimento de *software*, o grupo se baseou nos estudos de Thomas Beyhl e Holger Giese em realizar a integração de técnicas de inovação com técnicas de engenharia. (Beyhl e Giese, 2014)

Dessa forma, dividiu-se o processo de Inovação e Desenvolvimento em 3 fases principais: Inovação, Arquitetura e Desenvolvimento.

Figura 7 - Diagrama com o processo de Inovação e Desenvolvimento proposto



Fonte: Elaborado pelo grupo

A fase de Inovação é responsável pela concepção do produto, através do entendimento dos usuários, seus processos de negócio e da interação entre o usuário e seus processos. Nessa fase também será feita a validação da ideia com os próprios usuários. As técnicas do *Design Thinking* serão utilizadas para essa fase.

A escolha do Design Thinking foi motivada pelo seu foco no usuário e suas técnicas de empatia como forma de entender o problema. Dessa forma, é possível obter soluções que irão de fato melhorar a vida do usuário. (Plattner, 2010)

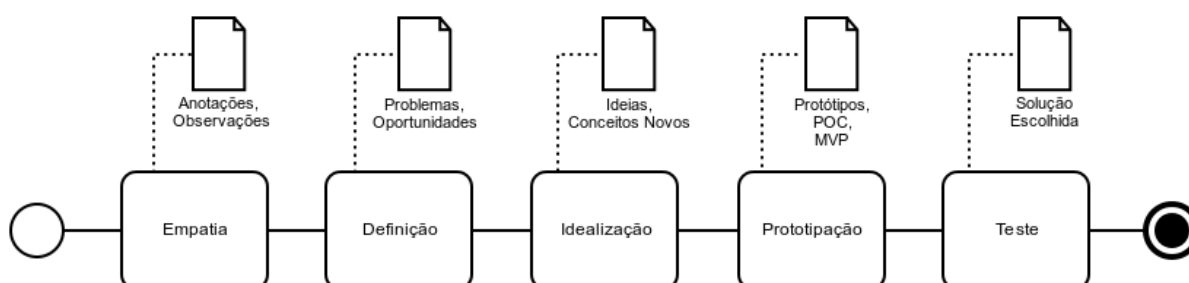
Na fase de Arquitetura, o produto imaginado pela fase de inovação será transformado para uma solução mais concreta em software, por meio da engenharia de produto. Para garantir o desenvolvimento escalável e modular do sistema, será feita a especificação e modelagem do sistema através de técnicas de engenharia de *software* baseada em visões.

Por fim, o Desenvolvimento será a fase onde será implementado de fato o produto. Como o grupo planeja um desenvolvimento rápido e flexível, de forma que seja possível acompanhar a evolução do produto de forma gradual, o grupo planeja se basear em técnicas do Scrum, que é uma metodologia ágil para gestão e planejamento de projetos de *software*.

3.1.1. Inovação

Durante essa fase o usuário será compreendido e será concebida uma ideia para um produto que resolva algum problema ou otimize algum processo. O diagrama abaixo apresenta como essa fase irá se estruturar.

Figura 8 - Diagrama BPMN da fase de Inovação



Fonte: Elaborado pelo grupo

Conforme o diagrama mostra, será feito um ciclo de *Design Thinking* com as fases de empatia, definição, idealização, prototipação e testes.

Na fase de empatia serão utilizadas técnicas para conhecer o usuário com o objetivo de entender suas dores, objetivos e seu contexto e a partir disso serão geradas anotações e observações.

Na fase de definição serão utilizados os documentos da fase anterior e a partir de técnicas de organização de ideias serão definidos alguns problemas e oportunidades.

Na fase de idealização serão apresentadas ideias e conceitos novos para atacar os problemas e oportunidades descobertos na fase anterior.

Na fase de prototipação serão criados protótipos, *Proof of concepts* (POC) ou *Minimum Viable Products* (MVP) para as soluções da fase anterior.

Na fase de testes os protótipos da fase anterior serão testados com o usuário com o objetivo de validar a ideia e então chegar a uma solução escolhida que será desenhada na próxima fase.

3.1.1.1. Empatia

Empatia é a capacidade de se colocar no lugar do usuário, podendo entender como o usuário interage com o ambiente, como ele pensa e como ele se sente, ajudando a entender do que ele precisa. (Plattner, 2010)

O objetivo dessa fase é entender o usuário, o contexto em que está inserido, seus objetivos e suas dores, para isso foram escolhidas as técnicas de observação, entrevistas e imersão.

A técnica de observação consiste em anotar os processos do usuário e tomar notas do ponto de vista do observador. É importante nessa técnica sempre procurar resposta para três perguntas: o que, como e por que, essas respostas ajudam a entender o que o usuário está fazendo, como ele faz essa tarefa e qual o seu objetivo e dessa forma buscar entender a fundo seus processos.

As entrevistas consistem em conversar com os usuários para entender melhor o ponto de vista do usuário, assim como seus sentimentos em relação ao processo estudado.

Para construir as perguntas da entrevista é importante sempre perguntar os porquês, perguntar sobre o passado ou presente e nunca sobre o futuro, encorajar que o entrevistado conte histórias que já aconteceram com ele no contexto estudado.

Durante as entrevistas é importante sempre tomar nota não só das respostas mas também da linguagem corporal e das emoções, não temer o silêncio, deixar que o entrevistado pense na resposta ou lembre de alguma história e não encorajar ou sugerir respostas, o que pode levar a pessoa a entregar respostas falsas.

A imersão consiste em entrar no mundo do usuário e interagir dentro dos processos, por exemplo, no caso de caixas de supermercados reservar um dia para trabalhar como um caixa de supermercado e entender a fundo todos os processos e sempre tomar nota de pontos de atenção.

No final desta fase, haverá diversas informações dos usuários como anotações de seus depoimentos, da observação e imersão que deverão ser formatadas e organizadas para a próxima fase.

3.1.1.2. Definição

O objetivo da fase de Definição é analisar as informações obtidas na Empatia e entender os por quês, procurando por padrões e dores do usuário. As técnicas utilizadas para atingir esse objetivo será o diagrama de afinidade.

Para essa fase é importante se questionar de todos os processos, sempre ser curioso, se questionando sobre tudo, descobrir os porquês e nunca julgar os valores dos usuários.

O diagrama de afinidade consiste em abrir todas os resultados obtidos na Empatia de forma visual como *post-its* e agrupá-los em diversas categorias de forma a encontrar os padrões. (Duarte, 2015)

No final dessa fase problemas significativo para o usuário ou oportunidades de melhoria serão encontrados.

3.1.1.3. Idealização

Na fase de Idealização serão procuradas soluções e ideias variadas para solucionar o problema encontrado na Definição. Para isso serão utilizados *Brainstorms*.

Os *Brainstorms* são discussões sobre soluções, focando mais em quantidade do que qualidade, mas sempre lembrando que apenas um deve falar por vez e não fugir do objetivo. Sem restringir nenhuma ideia previamente, sempre construindo novas ideias sobre as ideias dos outros. Tem como objetivo gerar um número grande de ideias.

Para um *brainstorm* é importante utilizar questões do tipo *How-Might-We(HMW)*, questões “como nós ...” pensando sempre no usuário e na sua necessidade, por exemplo: “Como nós como vendedores de sorvete podemos vender mais?”.

Para conseguir ideias mais variadas podem ser colocadas limitações de tecnologia, preço, etc.

Ao selecionar as melhores soluções do *brainstorm* é importante não descartar nenhuma ideia imediatamente, ela pode ter um conceito que irá ajudar depois e para a seleção podem ser utilizadas votações com as ideias favoritas, mas sempre discutindo os principais pontos das descartadas.

No final dessa fase deve existir uma grande variedade de soluções que podem ser classificadas e dentre elas podem ser escolhidas a mais relevante para o usuário.

3.1.1.4. Prototipagem

Na fase de prototipação será implementada de forma rápida e barata, mas que possa trazer resultados, a solução encontrada na fase de Idealização.

É importante na construção de um protótipo não gastar muito tempo na sua construção e idealização e construir sempre pensando no usuário, pensando em como conseguir o seu *feedback*.

Para construir o protótipo é uma boa ideia impor restrições de forma a busca incentivar a criatividade. Exemplo: Fazer um protótipo usando uma caixa de sapato. Apesar de contra intuitivo, essas restrições estimulam a criatividade dos projetistas, que devem ser criativos para encontrar soluções as restrições impostas.

O protótipo pode ser um vídeo explicando o conceito, um conjunto de telas impressas, apenas um conceito por cima de outra plataforma como utilizar o *e-mail* para um determinado serviço ou qualquer forma que possa passar ao usuário a ideia principal da solução.

No final dessa fase devemos ter um produto que possa ser testado e definirmos como iremos testá-lo.

3.1.1.5. Teste

Na fase de testes o protótipo criado na Prototipação será testado, o protótipo criado deve ser apresentado para o usuário e o *feedback* deve ser coletado e dessa forma entender mais o usuário.

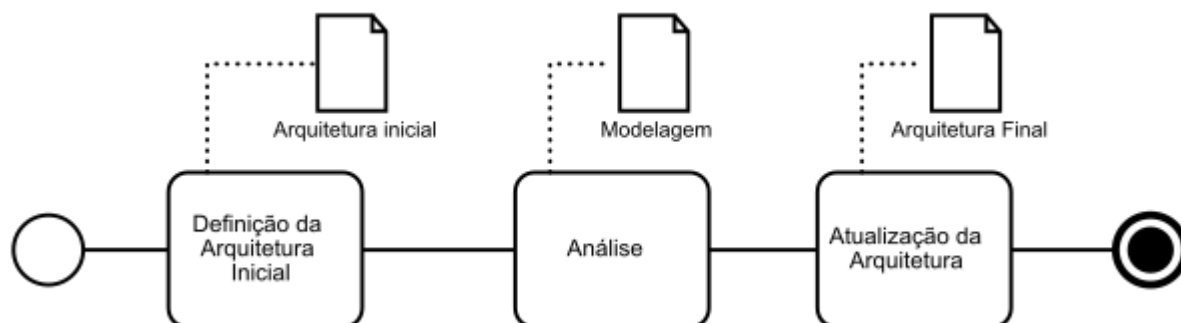
Durante os testes é importante deixar que o usuário use o protótipo, então não explicá-lo mas deixá-lo, explicando o contexto mínimo necessário para entender o que fazer e apenas observar ele interagindo, sem corrigir as ações dele. Depois do teste perguntar um pouco sobre a experiência dele e diversas perguntas para avaliar o sucesso ou fracasso da solução.

No final da fase de testes se a solução escolhida for aprovada pelos testes com os usuários devemos passar para a fase de arquitetura, caso contrário deve ser implementado outro protótipo com outra solução.

3.1.2. Arquitetura

Com base na solução que a fase anterior encontrou e validou, as técnicas arquitetura de *software* serão aplicadas para transformar essa solução em um produto de software, conforme mostra a imagem abaixo.

Figura 9 - Diagrama BPMN da fase de Arquitetura



Fonte: Elaborado pelo grupo

A partir dos documentos da fase de inovação poderão ser retiradas as funcionalidades do produto final e poderá ser feita uma fase de arquitetura inicial em que será feita uma arquitetura de 3 camadas com todas as funcionalidades do produto.

Com base na arquitetura inicial pode ser feita a modelagem do sistema, em que processos mais complexos deverão ser modelados e detalhados e ocorre a modelagem de dados do sistema caso haja a necessidade de armazenar dados de maneira estruturada.

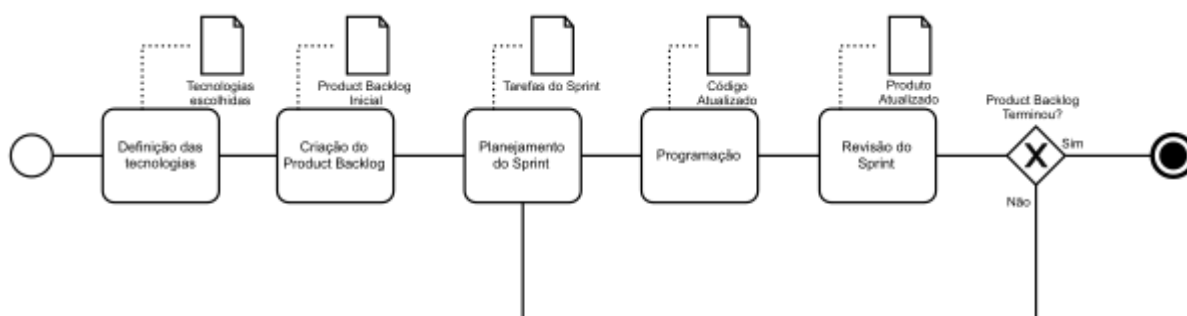
Ainda na fase de análise, pode ser feita uma análise de requisitos não funcionais e atributos de qualidade, trazendo soluções de arquitetura para estes.

Por fim, a arquitetura do sistema será atualizada com base nos documentos de especificação e modelagem.

3.1.3. Desenvolvimento

Durante essa fase, o sistema especificado na fase anterior será implementado. O diagrama do processo de Desenvolvimento é mostrado abaixo:

Figura 10 - Diagrama BPMN da fase de Desenvolvimento



Fonte: Elaborado pelo grupo

Como o diagrama mostra, primeiramente serão definidas as tecnologias a serem utilizadas no sistema como linguagem de programação, tecnologias para gerenciamento de versão, *host* do sistema, sistema de gerência de bancos de dados e outras ferramentas que irão ser utilizadas para o desenvolvimento do sistema.

Com base nos documentos de especificação e arquitetura gerados na fase anterior será gerado o *Product Backlog* com todas as funcionalidades do sistema.

Depois disso, podemos começar com os ciclos de desenvolvimento em *sprints*, que são períodos de uma a duas semanas onde ocorre o desenvolvimento das tarefas.

Cada ciclo começa com a reunião de Planejamento de *sprint*, onde as tarefas do ciclo serão selecionadas e priorizadas, com o objetivo de sempre gerar um entregável de valor ao final do *sprint*.

Durante essa reunião, qualquer tarefa que não esteja clara o bastante deve ser discutida e detalhada com a equipe. Só então deve ser passada para a etapa de programação, onde de fato o sistema será implementado.

Por fim, deve ser feita uma reunião para revisar o que foi feito no *sprint* e o que não foi, as tarefas que não foram concluídas voltam para o *Product Backlog* e serão priorizadas novamente na próxima reunião de Planejamento de Sprint.

O ciclo deve ser repetido até que as tarefas do *Product Backlog* tenham sido concluídas.

3.1.4. Validação e testes

Para realizar a validação do processo, o grupo irá aplicá-la no desenvolvimento de um produto, para isso será escolhido um tipo de usuário para que um produto que resolva algum problema ou melhore algum processo dele seja desenvolvido.

O usuário escolhido foram os restaurantes comerciais em regiões empresárias da cidade de São Paulo e seus clientes, sendo estes geralmente funcionários dos escritórios da região. Essa escolha foi feita pensando no conhecimento do grupo como clientes desses locais e na grande quantidade existente deles. Assim, o grupo irá procurar pelos responsáveis dos restaurantes e por funcionários de empresas para possa conhecer melhor seus hábitos, procurar por problemas ou deficiências e dessa forma desenvolver um produto com o processo proposto.

Assim que todas as três fases do processo forem aplicadas haverá um produto que depois será testado com os usuários para verificar se ele atende a suas expectativas.

3.2. Aplicação do processo

Neste tópico descreve como foi a aplicação do processo para os clientes dos restaurantes e seus clientes, descrevendo como as três fases foram realizadas, os problemas que ocorreram e os aprendizados.

3.2.1. Fase 1 - Inovação

3.2.1.1. Empatia e definição

A fase de empatia e definição, apesar de serem distintas são extremamente dependentes entre si. Dessa forma, foi decidido juntar ambas em uma mesma fase.

As entrevistas com os responsáveis dos restaurantes foram realizadas em 8 restaurantes na região da Av. Paulista sempre com os donos do estabelecimento.

Para descobrir dúvidas sobre o dia a dia do restaurante e a relação com o cliente, foi feita uma lista de dúvidas e dela foi montado os modelos de entrevista com ambos os tipos de usuários que podem ser vistos no apêndice 1, modelo de entrevistas da fase de empatia.

Inicialmente, o grupo não se saiu muito bem com as entrevistas com os restaurantes, pois a entrevista tinha muitas perguntas e os responsáveis pelos restaurantes estavam sempre muito ocupados.

Dessa forma o formato da entrevista foi modificado, e foram selecionadas 3 perguntas, que focavam nos principais problemas que eles enfrentavam no dia a dia.

Foi percebido que a entrevista durante o expediente deles deveria ser rápida e direta, pois de outra forma, atrapalharia o trabalho deles ou eles deixariam de responder as perguntas.

Foram levantadas diversas preocupações com relação a falta de cliente, ambiente, funcionários e até automatização de processos, sendo destes o ponto de destaque foi a falta de clientes e o atendimento dos funcionários. Os resultados completos das entrevistas com os usuários estão no apêndice 2, resultados das entrevistas da fase de empatia na fase de Inovação

Depois de conversar com os donos de restaurantes, o grupo decidiu agrupar as ideias para visualizar melhor os padrões nas respostas na figura 11, agrupamento de ideias obtidos das entrevistas com donos de restaurantes.

Figura 11 - Agrupamento de ideias obtidos das entrevistas com donos de restaurantes

Preocupação										
Qualidade da comida/produto	Comprometimento com a qualidade	Boa qualidade								
Fornecimento	Fidelidade com empresas de fornecimento									
Funcionários	Treinamento de funcionários	Bom atendimento	Bom atendimento	Atendimento ruim	Problema com funcionários	Bom atendimento	Problema com funcionários	Problema com funcionário		
Automatização	Automatização de algum processo	Investimento em infraestrutura, talvez automatização no processo de desocupação de mesas								
Ambiente	Ambiente	Investimento em infraestrutura, talvez automatização no processo de desocupação de mesas	Número de mesas							
Atrair mais clientes	Atacar a nichos	Promoções	Propaganda	Programa de fidelidade com clientes						
Falta de clientes	Falta de clientes	Clientes querem gastar menos	Falta de clientes	Crise economica - falta de clientes	Clientes querem gastar menos	Falta de clientes				

Fonte: Elaborado pelo grupo

Na figura 11 podem ser vistas diversas preocupações que os donos de restaurantes comentaram durante as entrevistas e cada uma das cores representa um comentário de um dono de restaurante.

As entrevistas com clientes foram realizadas conforme o modelo, que funcionou bem para todos os clientes, que em sua maioria foram colegas de sala e ex-colegas de trabalho do grupo. Foi levantado que os principais critérios na hora de escolher onde almoçar são preço, quantidade, qualidade e distância até o restaurante. Também foi percebido que a maioria não costuma almoçar todo dia no mesmo lugar.

Outro ponto interessante a se notar é a preocupação com o tempo de espera ou demora do restaurante. A maioria parece evitar restaurantes que demoram muito, seja no preparo do prato, na hora de pagar, ou mesmo para chegar até o local.

Figura 12 - Agrupamento de ideias obtidos das entrevistas com clientes

Gostam	Características dos clientes	Não gostam
Custo-benefício	A maioria dos cliente nunca conseguiu terminar um cartão fidelidade	Filas grandes na hora de pagar
Variedade	A maioria dos clientes tem muitas opções de restaurantes perto de onde trabalham	Espera longa para trazer a comida
Novidades	Quando um prato desagrada um cliente, ele pode voltar ao estabelecimento, mas não costuma pedir o prato de novo	Lugares muito longe
Promoções/Descontos	Para a maioria das pessoas, é mais fácil deixar de ir a um restaurante do que reclamar quando acontece algum problema	Perder muito tempo no almoço
Qualidade (Comida e atendimento)	Todos já tiveram uma experiência de vir pedido errado/esquecer algum prato	Atendimento ruim
Acompanhamentos de graça (Sobremesa, sucos, etc...)	A distancia até um restaurante ser motivo para pessoa deixar de almoçar no local	Add a card...
Receber indicações de lugares novos para comer	As pessoas levam em consideram se o estabelecimentos é muito demorado para escolherem onde almoçar	
Almoçar em grupo	Normalmente, quando as pessoas já conhecem o lugar, já tem em mente o que pedir	
Add a card...	Add a card...	

Fonte: Elaborado pelo grupo

3.2.1.2. Idealização

Na primeira reunião de brainstorm foram levantadas diversas ideias variadas descritas abaixo:

- Um sistema de simulação de restaurantes em que o dono poderia inserir dados do seu estabelecimento e do movimento e dessa forma otimizar o número de funcionários alocados em cada função. Dessa forma o

responsável poderia alocar os funcionários de forma mais eficiente, diminuindo parte da sua preocupação com atendimento e funcionários.

- Um sistema para que o cliente saiba informações do restaurante no momento como número de pessoas e comidas disponíveis e que poderia ser integrado a um sistema de geolocalização podendo estimar um tempo de almoço desde a partida até a volta. Assim os clientes poderiam programar melhor o almoço perdendo menos tempo.
- Sistemas de automação de restaurantes ou monitoramento como sistemas em que o próprio cliente realiza os pagamentos e pesagens e automatizar o processo de tomada de pedidos. Dessa forma diminuiria a dependência do gerente dos seus funcionários, além de diminuir o tempo gasto na fila ou interação com funcionários.
- E algumas ideias menores como sistema de fidelidade que ficaria no restaurante, sistema de troca de avaliações por brindes, sistema de vendas ou promoções ou brindes cruzados (como restaurantes e cafeterias), sistema de envio de notificações customizadas e sistema de delivery de marmitas customizadas. Essas ideias ajudavam nos problemas de trazer mais clientes para os restaurantes, ou eram baseadas em coisas que o clientes apreciavam.

Depois das ideias serem levantadas, o grupo começou a pensar em protótipos para validar as ideias, entretanto houveram dificuldades nessa tarefa e não foi possível visualizar protótipos ou como as ideias poderiam ser validadas. Dessa forma, o grupo voltou para o passo anterior e trocando de abordagem e focando em soluções completas que tinham potencial para se tornar produtos.

Além disso, o grupo resolveu se focar na solução para clientes para os restaurantes por ter maior empatia com esse grupo.

Nessa nova abordagem foram obtidas soluções mais detalhadas como: um aplicativo para receber notificações de pratos favoritos, aplicativo para leilão de refeição e um aplicativo social de avaliação de restaurantes com o uso de *gamification*.

Gamification é o conceito de se transformar uma tarefa em um jogo para que as pessoas se interessem mais pelo produto, dessa forma o aplicativo social seria

como um jogo em que as pessoas escreveriam avaliações dos restaurantes e ganhariam pontos de outros usuários pelas suas avaliações. Os clientes poderiam recomendar para outras pessoas novos lugares e os restaurantes receberiam novos clientes.

No aplicativo para receber notificação de pratos favoritos a pessoa poderia cadastrar seus pratos favoritos no sistema e quando um restaurante na região servisse o prato no dia a pessoa seria notificada. Essa ideia foi levantada a partir da observação que os clientes geralmente tem em mente o que pedir quando vão a um restaurante conhecido, mas nem sempre a comida está disponível no dia.

O aplicativo de leilão seria a ideia que um grupo de pessoas que saíssem para almoçar juntas poderiam enviar uma notificação para os restaurantes das redondezas e estes poderiam oferecer ofertas para o grupo de clientes. Dessa forma os clientes poderiam conhecer novos lugares e os restaurantes ganhariam novos clientes, além disso, incentivaria a prática de promoções que auxilia ambos os lados.

Depois dessas novas ideias o grupo se sentiu satisfeito com os resultados e decidiu escolher três dessas ideias para serem testadas em protótipos: o aplicativo de notificação de pratos favoritos, o sistema para saber se o restaurante está cheio ou não e o aplicativo social que utilizará *gamification*.

3.2.1.3. Prototipação

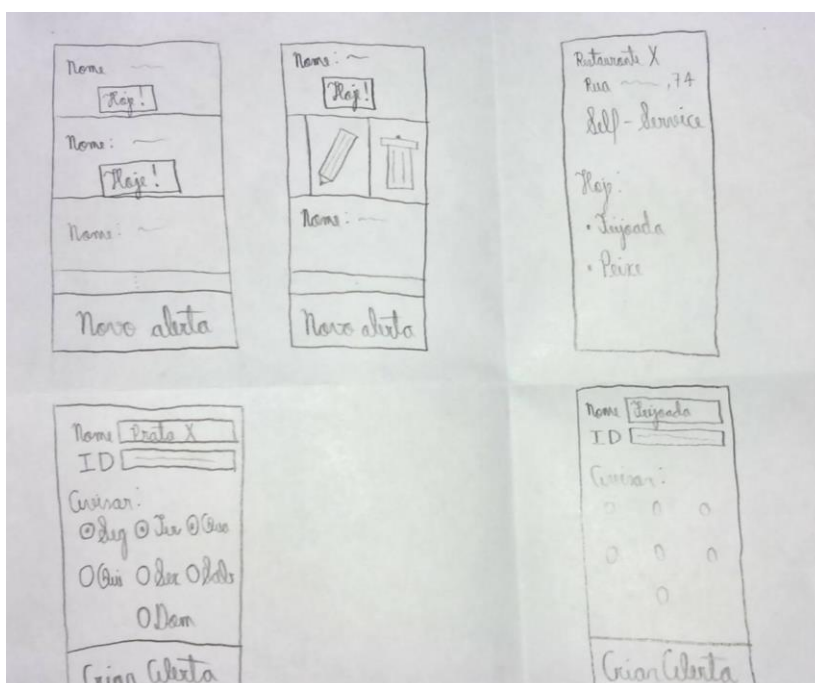
Inicialmente foram dados nomes às soluções, mesmo que temporários, para que as ideias fossem identificadas. Os nomes foram “É hoje”, “The Game” e “Não dá não”, para a ideia de notificar o usuário quando o restaurante fosse servir um prato específico, sistema com *gamification* em que o usuário poderia comentar restaurantes da região e avisar aos clientes quão cheio um restaurante está, respectivamente.

Devido a natureza dos protótipos e como as ideias foram originalmente pensadas na fase anterior, na qual as ideias foram idealizadas de forma fechadas que pudessem se tornar produtos e que o cliente poderia interagir com ele, o tipo de protótipo que mais se adequava a elas e que iriam representar de forma mais fiel às ideias seriam os *mock-ups*.

Vale lembrar que geralmente *mock-ups* são utilizados em testes de usabilidade para novos produtos, mas que nesse caso eles seriam utilizados para validação das ideias.

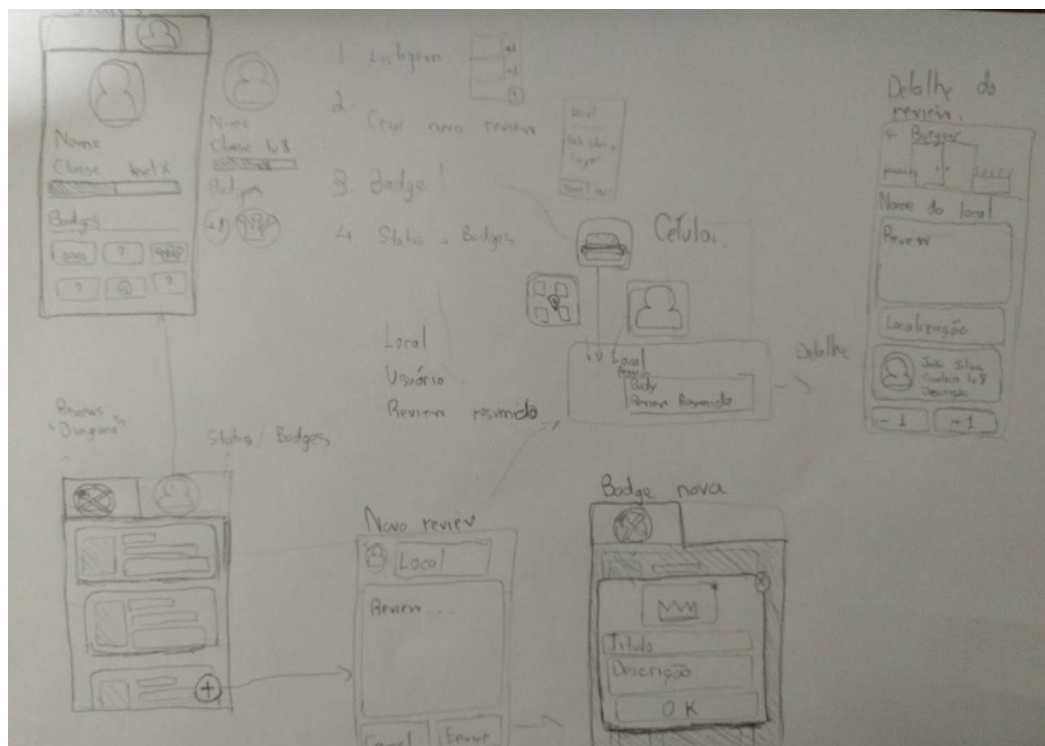
Em paralelo ao estudo da ferramenta de prototipagem, foram feitos wireframes, um protótipo de interface, das ideias para que fosse possível discutir sobre ela antes de começar a desenvolvê-la na ferramenta com o *mock-ups*, a ideia foi materializada o mais rápido possível para que pudéssemos refiná-la através de discussões em grupo.

Figura 13 - Rascunho do *wireframe* do “É hoje”



Fonte: Elaborado pelo grupo

Figura 14 - Rascunho do wireframe do “The Game”



Fonte: Elaborado pelo grupo

Devido ao tempo disponível para realizar a fase, foi decidido fazer esse protótipo com ajuda de ferramentas de prototipagem. Algumas ferramentas foram pesquisadas e 4 ferramentas foram encontradas que poderiam atender às necessidades do grupo: *Moqups*, *Invision*, *Pencil Project* e *JustInMind*.

Depois de testar e analisar cada ferramenta, foi decidido o uso do *JustInMind* (JustInMind, 2016), pois apesar da licença grátis expirar depois de 30 dias, entre as ferramentas encontradas ela demonstrou ter o maior número de recursos necessários, como testes em mobile e elementos visuais já inclusos.

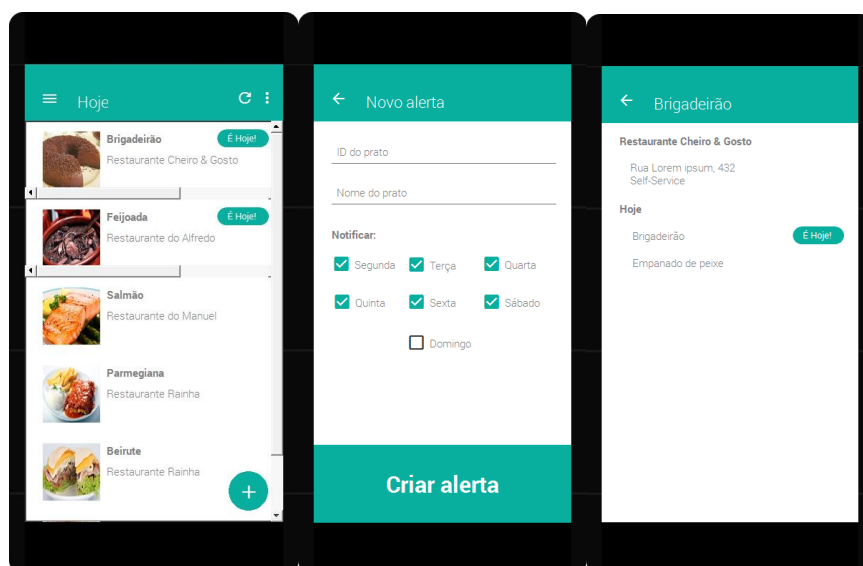
Mesmo com essa ferramenta, o uso dos rascunhos com papel se mostraram importantes para discussões do protótipo e para pensar em como testar apenas o conceito chave de cada ideia.

Foi visto que a fase de prototipagem complementa a fase de idealização, pois durante a fase anterior, foram escolhidas ideias e funcionalidades que pareciam solucionar ou auxiliar os usuários, sem a necessidade de se preocupar com a implementação de código, telas ou outros aspectos mais concretos.

Em contrapartida, essa fase se iniciou com soluções selecionadas, permitindo que o grupo se focasse mais em como mostrar aos usuários a ideia do produto, avaliando o que era importante para essa demonstração, como telas, interação física, feedbacks. Além disso, também foram discutidas formas do usuário realizar as funcionalidades previstas (telas, dados, ações necessários).

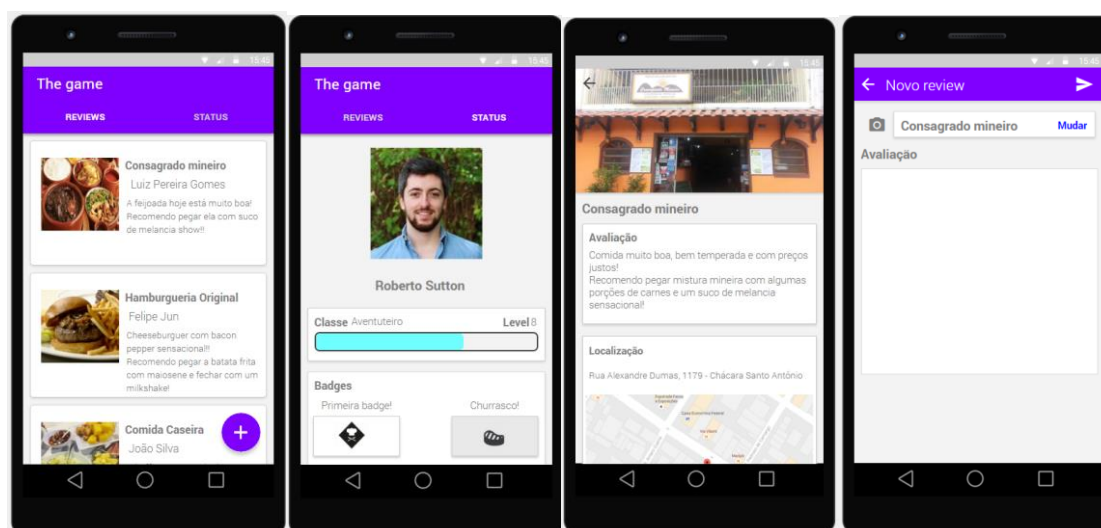
Vale ressaltar que tanto as decisões sobre as telas (visual), quanto as de funcionalidades, posteriormente poderão ser convertidos em especificação técnica. Esse resultado reafirma a importância da fase no contexto de desenvolvimento de *software* utilizando-se do *Design Thinking*.

Figura 15 - Telas do protótipo do “É Hoje”



Fonte: Elaborado pelo grupo

Figura 16 - Telas do protótipo do “The Game”



Fonte: Elaborado pelo grupo

Em decorrência do tempo disponível para essa fase, ela foi finalizada com apenas dois protótipos, das três ideias da fase anterior.

3.2.1.3.1. Retrabalho da fase de prototipagem

Antes das entrevistas começarem foi notado que os protótipos não foram feitos pensando nas perguntas que seriam feitas aos usuários. O “É Hoje” demonstrava telas muito trabalhadas e funcionalidades que iam além da ideia central do protótipo. Também haviam telas ou fluxos incompletos no *mock-up* do “The game”.

Acreditando que essas características poderiam afetar o resultado dos testes, o grupo decidiu fazer algumas melhorias em um dos protótipos e refazer o outro.

“The game” foi melhorado, finalizando seus fluxos, corrigindo problemas de resposta do *mock-up* (devido ao desconhecimento de como a ferramenta de prototipagem escolhida funcionava) e adicionando mais interação com o usuário.

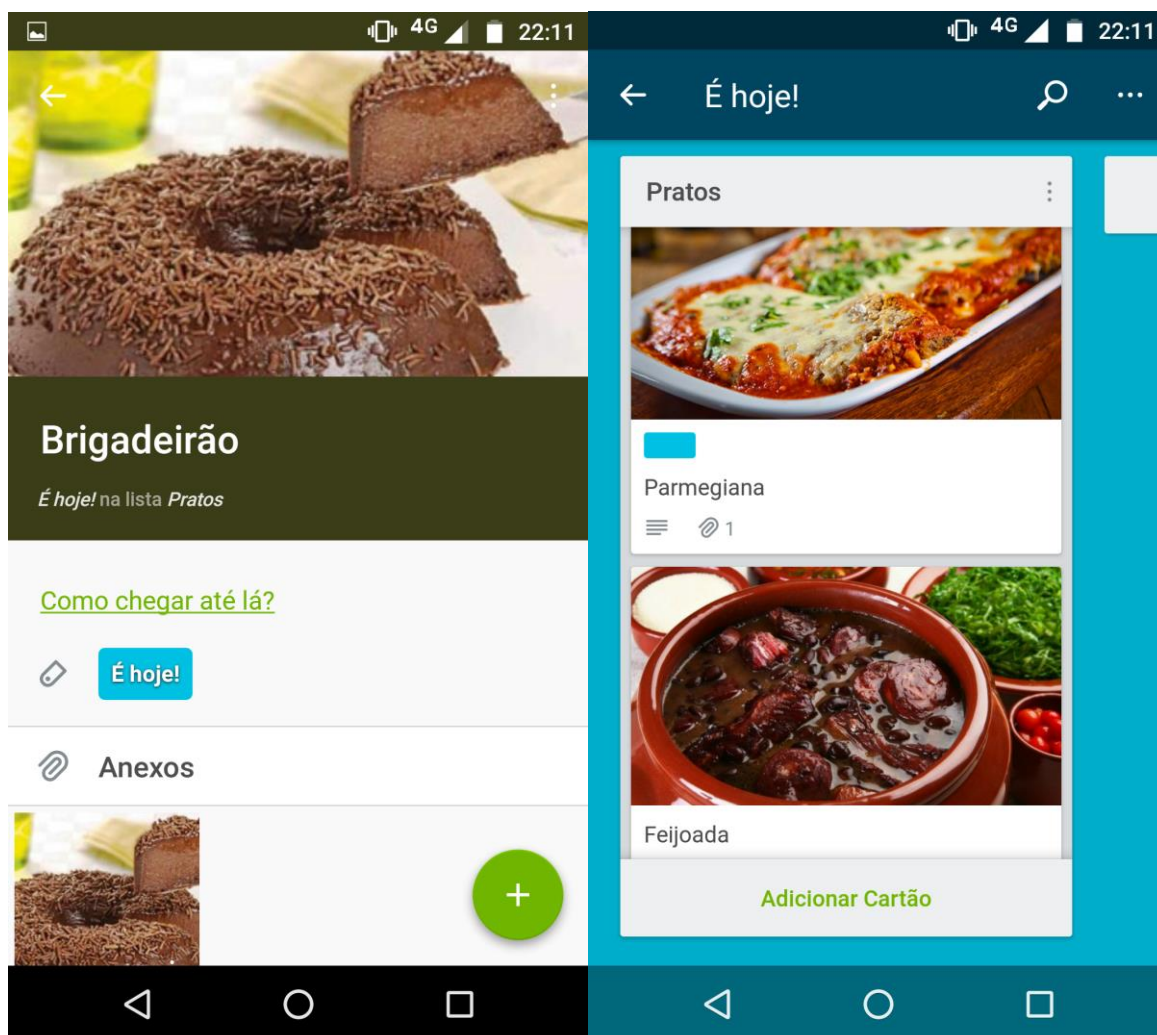
Isso foi feito acreditando que um protótipo melhor iria permitir que o usuário desse uma resposta mais focada nas funcionalidades principais do produto ao invés de criticar funcionalidades secundárias.

A decisão de refazer “É Hoje” foi feita quando o *mock-up* desenvolvido foi analisado e foi percebido que ele não deixava claro qual a solução principal ele estava validando. Com isso, o grupo teve duas alternativas, alterar o mock ups ou fazer outro protótipo que focasse mais na ideia principal da solução.

O grupo optou pela segunda opção e decidiu mudar o protótipo de um *mock-up* para um teste mais simples, focando apenas em testar a ideia principal da solução e ignorando o visual e telas finais.

O novo protótipo do “É hoje” utilizou o sistema Trello como protótipo em que os pratos ficaram em listas do sistema e é possível que o usuário veja os seus pratos favoritos pelo celular. O protótipo pode ser visto nas figuras abaixo:

Figura 17 - Protótipo do “É hoje” feito no Trello



Fonte: Aplicativo Android do Trello

3.2.1.4. Testes

Para que os protótipos construídos fossem validados, foi decidido fazer entrevistas com alguns usuários que poderiam utilizar o produto após construído, isto é, funcionários que almoçam em restaurantes perto das empresas onde trabalham.

As entrevistas consistiram da apresentação mínima do protótipo, deixando que o usuário interaja com o protótipo e posteriormente, perguntar sobre a

experiência. Paralelo a interação com protótipo, foi observado o comportamento do usuário e suas dificuldades em utilizá-lo.

Terminada as entrevistas, os pontos positivos e negativos de cada protótipo foram avaliados, focando nas questões “O propósito da solução está claro?”, “A solução imaginada atende a necessidade dos usuários?”, “Quais melhorias poderiam ser feitas?”.

O número de pessoas que foram chamadas para os testes será 5, baseado nos estudos de Jakob Nielsen que mostram que com esse número de usuários já é possível perceber 80% dos padrões (Nielsen, 2000).

Após a alterações nos protótipos, as entrevistas com os participantes foram agendadas. E foram escolhidas para fazer entrevistas pessoas do mesmo curso dos integrantes grupo, que já estagiaram em empresas e almoçavam em restaurantes. Essa escolha foi feita acreditando que eles fazem parte do público alvo como funcionários de empresas que almoçam em restaurantes perto de onde trabalham.

O grupo não se restringiu apenas às pessoas que almoçam em restaurantes, também buscou marcar entrevistas com pessoas que levam marmita para almoçar, tentando obter *feedbacks* com maior variedade possível.

Um pequeno roteiro foi preparado para as entrevistas em que primeiro o grupo seria apresentado, então seriam feitas algumas perguntas para descobrir o perfil do usuário testado, o protótipo “The Game” seria apresentado e o usuário poderia utilizar por alguns minutos enquanto seriam anotadas suas observações, dificuldades e reações, então algumas tarefas específicas do protótipo seriam dadas para o usuário executar, seria perguntada a sua experiência e por fim, o “É hoje” seria apresentado e o processo repetido.

Os *feedbacks* das entrevistas foram agrupados de forma similar as fases de definição, para facilitar a visualização dos padrões entre os usuários. Conforme mostrado no apêndice 3, resultados da fase de testes da fase de Inovação

Como pode-se observar, boa parte dos usuários gostaram dos mecanismos de gamification do “The Game”, no entanto queriam mais detalhes de como ganhar as recompensas do jogo. Todos entenderam bem o objetivo do protótipo, mostrando que a solução proposta era satisfatória.

No caso do “É Hoje”, ficou evidente que a ideia do protótipo não ficou clara. Principalmente, pela ferramenta não ter sido a melhor escolha para fazer o protótipo. Além disso, depois de explicado o objetivo do protótipo, muitos compararam suas funcionalidades com uma lista de pratos favoritos, enxergando a ideia como uma funcionalidade secundária ou invés de um produto.

3.2.1.5. Conclusões da fase

Para pessoas que estão iniciando no Design Thinking e não tem experiência como os integrantes do grupo, foi percebido que o processo como um todo se trata mais de tentativa e erro do que acertos, ou seja, o processo deve ser abordado de forma a favorecer a experimentação, não importa se aparentemente, as ideias não foram boas ou o protótipo não é o suficiente, o objetivo não é acertar de primeira.

Deve-se errar, perceber o erro e ao invés de para o processo para corrigi-lo no momento que foi percebido, é preferível evitá-lo na próxima interação do Design Thinking. O grupo acredita que isso auxilia na velocidade do projeto como um todo, além de evitar retrabalhos.

A metodologia de inovação com foco em usuário é bem diferente das abordagens que normalmente se aplicam na área de engenharia de *software*, em que o papel do engenheiro é entender os requisitos do cliente, ou projetos os problemas do usuário de forma distante, com pouca aceitação para erros.

A principal diferença é entender que o pensamento deve ser voltado a empatia com o usuário e, para isso, ele deve ser tratado como parte do processo de desenvolvimento do projeto. É através do entendimento do seu contexto, interesses, dores, vontades que o produto toma forma.

O projetista deve se limitar-se a criar suposições, de ideias ou problemas, e sempre validá-las com o usuário o mais rápido possível. Isso garante que o resultado final irá caminhar mais rapidamente para o produto ideal.

Deve-se, no entanto, tomar cuidado para não resolver os problemas específicos de um único usuário ou grupo de usuários pequenos, pois dessa forma, a solução pode se tornar tão restrita às necessidades deles, que não será bem aceita ao público como um todo. Nesse caso, o trabalho se aproxima mais de uma consultoria.

A partir dos resultados dos testes pudemos ver que os usuários se interessaram bastante por *gamification* utilizado no protótipo do “The Game” e a fácil visualização da localização com o uso de mapa e gostaram de saber do que tinha no dia nos restaurantes.

Baseado nos *feedbacks* recebidos o grupo chegou a uma nova ideia que será produzida com base no que os usuários gostaram nos protótipos e corrigindo alguns erros de usabilidade e design.

A ideia a ser implementada será uma inspiração do The Game em que será utilizado *gamification* com sistema de *badges* e teremos uma listagem de restaurantes, em cada um deles haverá uma listagem de pratos favoritos eleitos pelos usuários e será possível fazer uma avaliação simples com *likes* ou *dislikes*.

A próxima fase será de Arquitetura, em que os resultados de cada fase do *Design Thinking* em funcionalidades serão transformados em funcionalidades, requisitos e documentação do sistema.

3.2.2. Fase 2 - Arquitetura

A fase de arquitetura foi iniciada desenhando um diagrama de 3 camadas inicial do sistema para servir de arquitetura inicial. Para isso, foi utilizado os resultados da fase anterior, que no caso do projeto foi um protótipo validado com o usuário.

O grupo pode facilmente criar o diagrama de 3 camadas inicial pois tinha bom conhecimento dos processos do usuário por ter participado de todas as fases do *Design Thinking*, não precisando formalizar este em diagramas ou documentos.

Para criar a arquitetura de 3 camadas primeiro foi passado por cada fluxo a ser desenvolvido e foram listadas e telas e componentes visuais que posteriormente seriam módulos da camada de aplicação. Então foram identificadas as entidades do sistema, ou seja, dados que deveriam persistir, essas entidades devem estar relacionadas em módulos da camada de acesso à dados. E por fim, foram identificadas as regras de negócio do sistema, os módulos responsáveis pelo gerenciamento de entidades e serviços de consulta a dados externos e essas regras de negócios foram agrupadas em módulos com mesmo contexto.

Figura 18 - Rascunho da arquitetura de 3 camadas inicial



Fonte: Elaborado pelo grupo

A arquitetura do diagrama da figura 18 possui na camada mais acima a camada de apresentação, logo em baixo a camada de aplicação e por último a camada de acesso à dados.

Na camada de apresentação estão representados os módulos que irão conter as telas do sistema: a lista de restaurantes, tela de detalhe de restaurante (que irá conter um mapa e listagem de pratos), tela de criação de avaliação, painel de conquistas e um módulo de notificação de conquista e a tela de perfil de usuário.

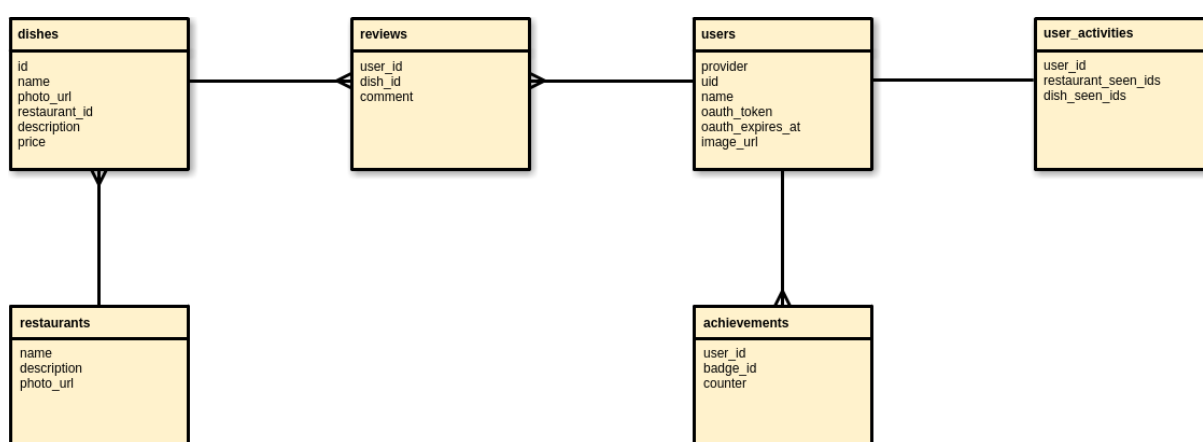
Na camada de aplicação estão os módulos responsáveis pelas regras de negócio: gerenciador de restaurante, serviço de consulta ao Google Maps, notificação de eventos, gerenciador de pratos, gerenciador de avaliação, gerenciador de conquistas, gerenciador de usuário e serviço de autenticação externo.

Na camada de acesso à dados estão os módulos responsáveis pelos dados do sistema: tabela de restaurante, tabela de avaliação, gerenciamento de pratos, regras de recompensas e tabela de usuário.

A arquitetura representa todas as funcionalidades previstas pelo produto idealizado na fase de inovação, incluindo módulos necessários para o mecanismo de gamification, não especificado totalmente na fase de inovação.

A partir da camada de acesso a dados foi possível realizar uma análise do sistema e partir disso foi feita a modelagem de dados do sistema. A modelagem de dados foi feita com um diagrama entidade relacionamento que pode ser visto abaixo:

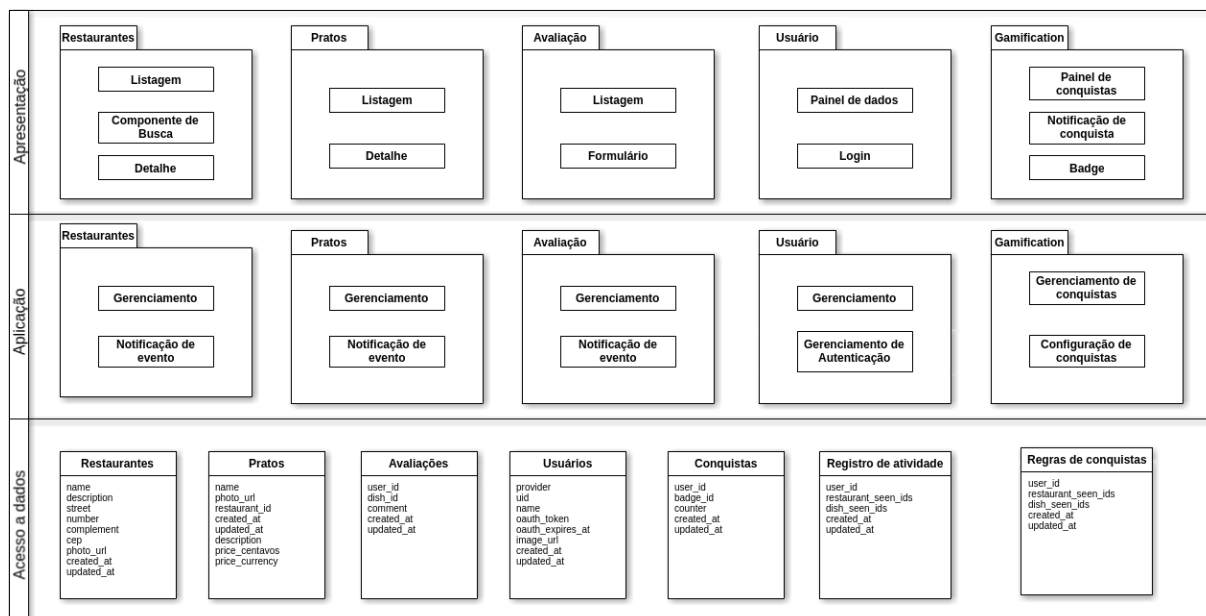
Figura 19 - Diagrama entidade relacionamento do sistema



Fonte: Elaborado pelo grupo

A partir da modelagem de dados realizada foi possível completar a arquitetura de 3 camadas e produzir a arquitetura abaixo que contempla todas as funcionalidades propostas.

Figura 20 - Arquitetura de 3 camadas final



Fonte: Elaborado pelo grupo

3.2.3. Fase 3 - Desenvolvimento

A fase de desenvolvimento iniciou com a definição das tecnologias. Foi necessário definir qual a linguagem que seria utilizada, o sistema de gerência de banco de dados, o serviço de hospedagem, ferramenta para gerenciamento de versão e serviço para armazenar o código fonte.

Foi considerado que o sistema não estaria totalmente concluído, sendo um produto inicial do ciclo de produção de software, foi escolhido tecnologias e ferramentas que permitem o rápido desenvolvimento, como forma de permitir uma iteração mais rápida que permitiria um feedback mais rápido dos usuários.

O sistema seria desenvolvido utilizando-se a linguagem Ruby, com o framework Rails, vendo que essa escolha permitiria o grupo maior velocidade de desenvolvimento, mas devido à sintaxe da linguagem ser bem aberta existem problemas de manutenção em equipes menos experientes. (Prechelt, 2011)

Com base no diagrama de entidade relacionamento, foi observado que a forma como as entidades estavam relacionadas favorecia o uso de sistemas de gerenciamento de banco de dados relacionais, pois não haviam dados não

estruturados que precisavam ser persistidos pelo sistema e requisitos não funcionais como disponibilidade e confiabilidade não era uma preocupação.

Dessa forma, o PostgreSQL foi escolhido como sistema de gerenciamento de banco de dados pela sua fácil integração com as ferramentas de desenvolvimento. Além disso, a sua escolha facilita a eventual migração da base de dados para outros sistemas SQL.

O Heroku foi escolhido como serviço de hospedagem da aplicação web. Os principais motivos foram pela facilidade de se fazer a implantação, requerendo o mínimo de configurações necessárias do desenvolvedor, pelo grande número de add-ons, plugins que adicionam novas funcionalidades e recursos ao sistema, e pelo plano que permite a hospedagem de sistemas gratuitamente com algumas limitações (Heroku, 2016).

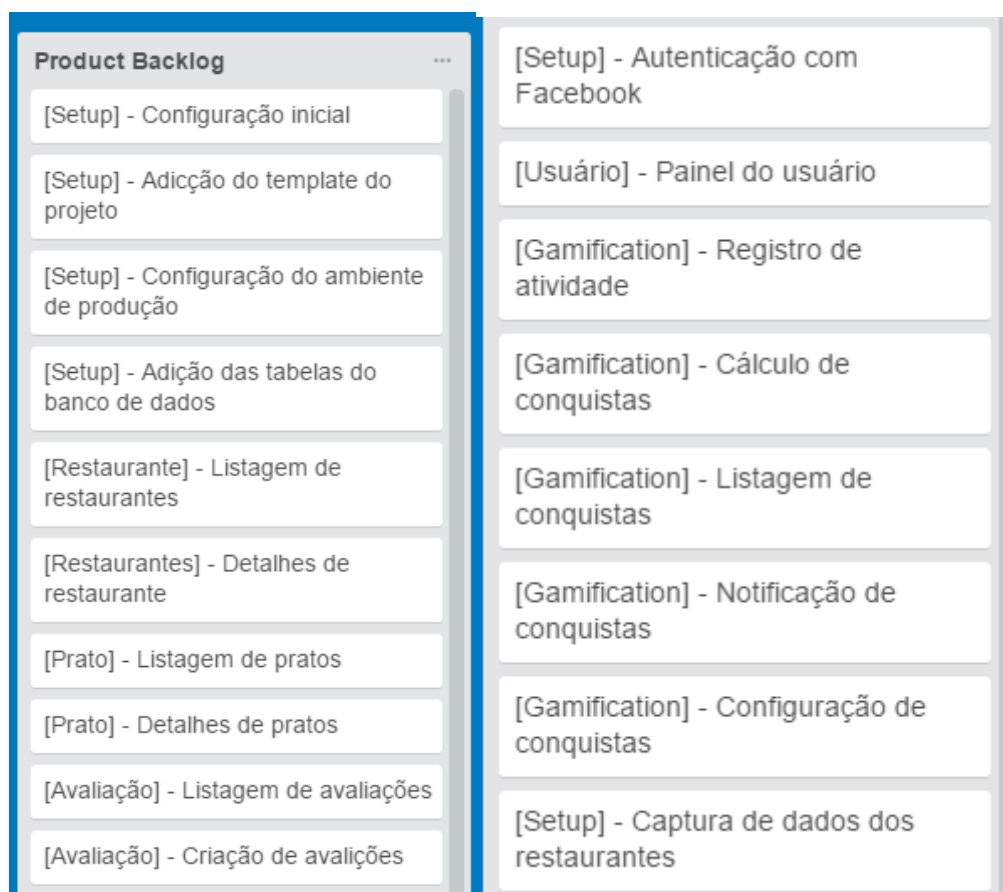
Para o versionamento do código a ser desenvolvido, a ferramenta escolhida foi o git, cujo a utilização é difundida mundialmente entre os desenvolvedores e permite o desenvolvimento distribuído.

A plataforma Bitbucket foi selecionada como hospedeiro de código pela sua integração com a ferramenta git e pelo fato do desenvolvimento privado com uma equipe pequena ser gratuito.

Depois da escolha das tecnologias foram realizadas algumas simplificações no produto de forma a agilizar o desenvolvimento e que o grupo pudesse validar rapidamente com o usuário o produto criado. Logo as simplificações foram: criação de um site *mobile*, ou seja, apenas para uso em celulares, autenticação somente pelo Facebook, filtragem de restaurantes pelo nome, comentários do usuário apenas por texto e telas redesenhadas. Dessa forma o grupo pode criar o *Product backlog*.

O *Product backlog* foi criado com as funcionalidades de acordo com os módulos da arquitetura de 3 camadas. As funcionalidades podem ser vistas na figura abaixo:

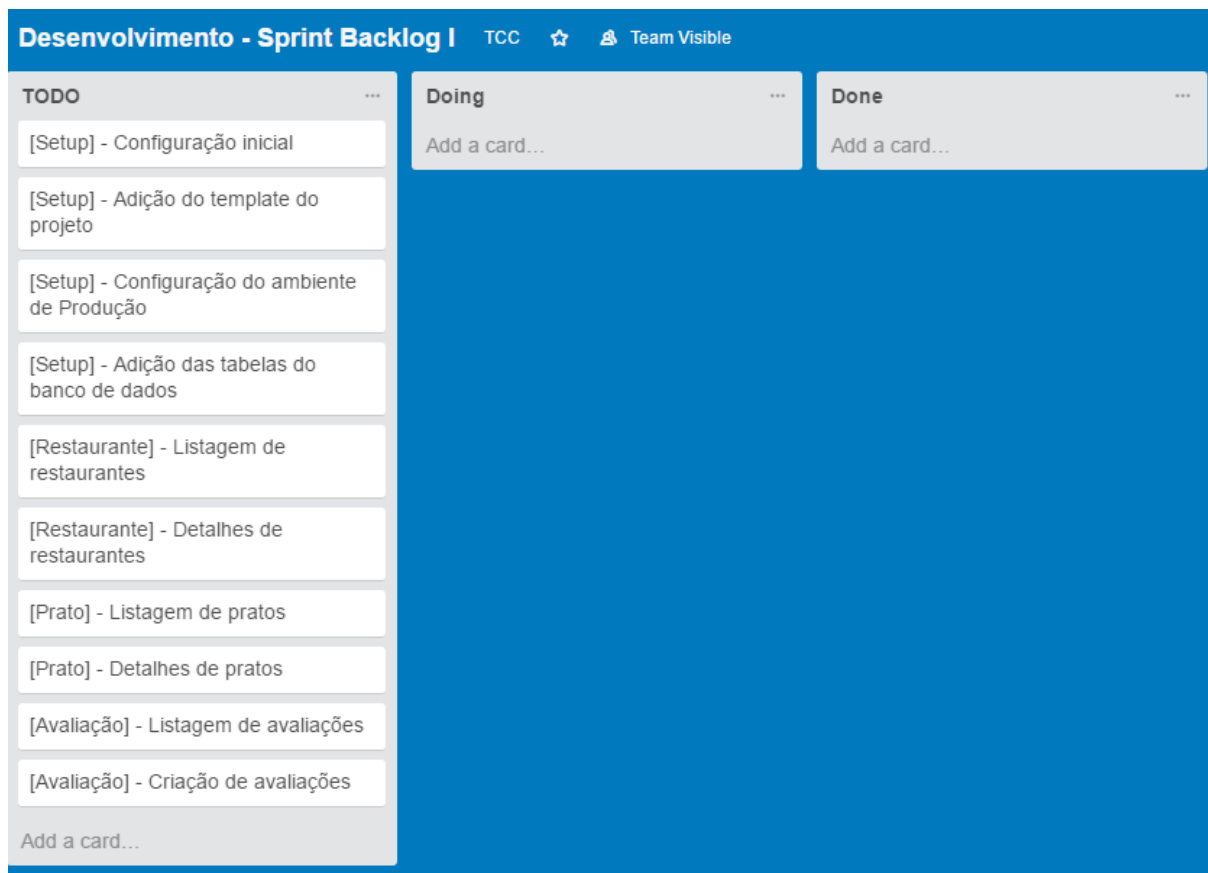
Figura 21 - *Product backlog* organizadas no Trello



Fonte: Trello

Com o *Product backlog* em mãos o grupo pode iniciar o desenvolvimento. O primeiro *sprint* de desenvolvimento foi feito priorizando-se as tarefas que entregariam maior valor para os clientes, ou seja, fazendo primeiramente as funcionalidades mais importantes do produto, em que todo o resto se baseava, a listagem de restaurantes e seus pratos. O prazo desse *sprint* foi de uma semana.

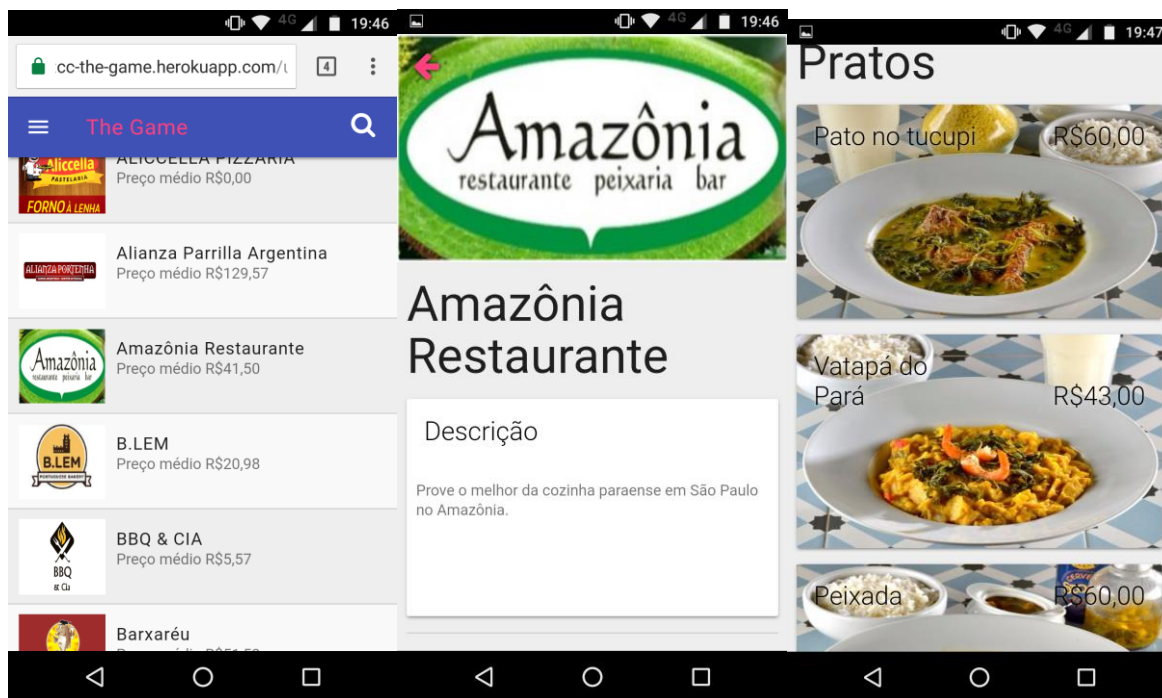
Por questão de dependência, configurações iniciais do projeto foram feitas antes do desenvolvimento, mas foram incluídas no *backlog*, para registro de atividades necessárias. Para organizar as funcionalidades a serem desenvolvidas foi utilizada a ferramenta Trello, que pode ser vista na figura abaixo:

Figura 22 - Funcionalidades do primeiro *sprint* organizados no Trello

Fonte: Trello

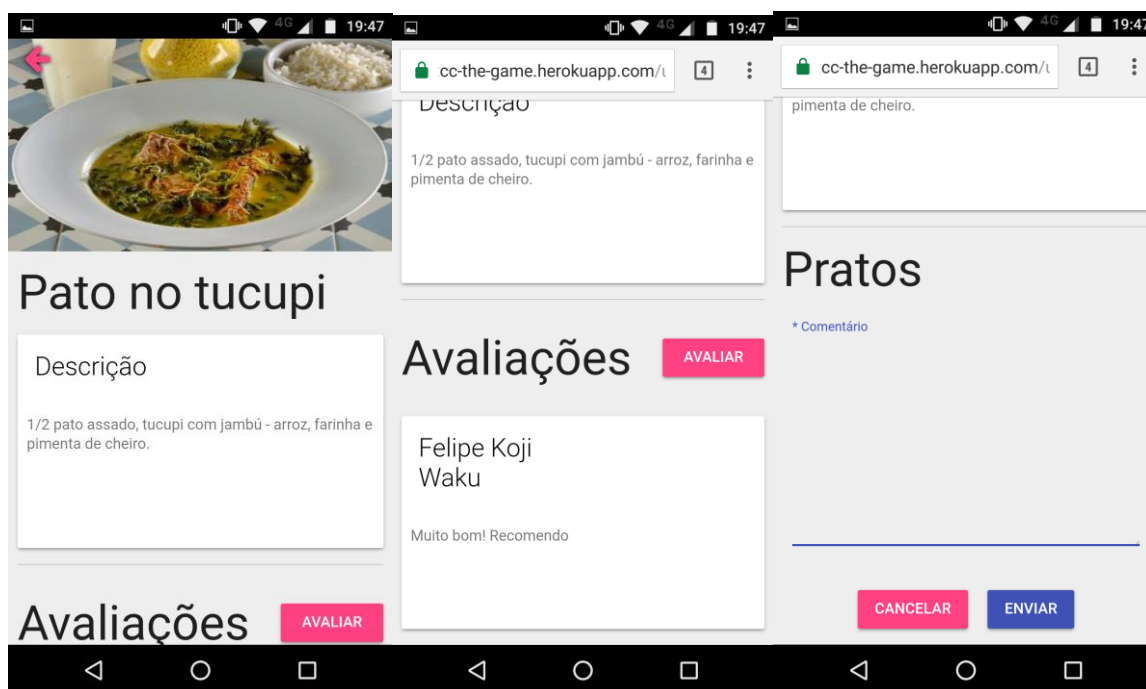
Com o *sprint* encerrado as telas das funcionalidades desenvolvidas podem ser vistas nas figuras a seguir:

Figura 23 - Telas das funcionalidades Listagem de restaurante, detalhe de restaurante e listagem de pratos



Fonte: Sistema The Game

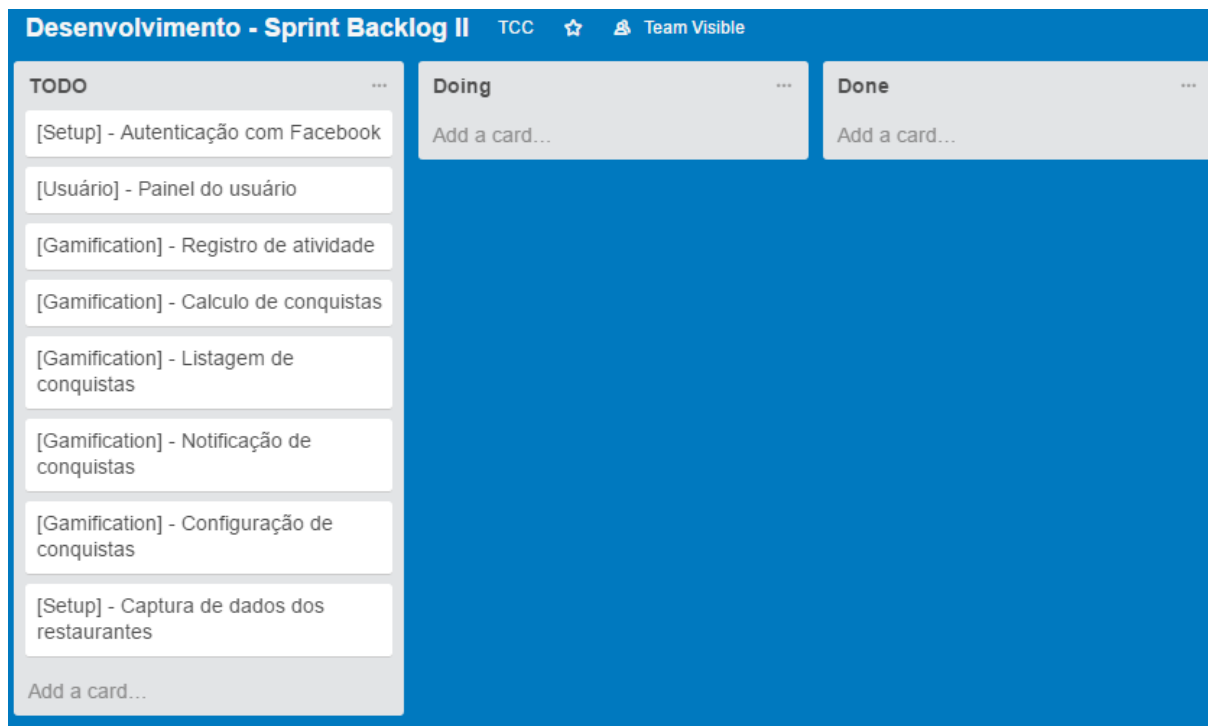
Figura 24 - Funcionalidades: Detalhe de prato, listagem de avaliações e Criação de avaliação



Fonte: Sistema The Game

Com o primeiro *sprint* terminado foi passado para o segundo que contemplava as funcionalidades restantes, ou seja, a autenticação com o Facebook, o painel do usuário, as funcionalidades de *Gamification* e um módulo para captura de dados dos restaurantes e assim como o *sprint* anterior, este também teve prazo de uma semana.

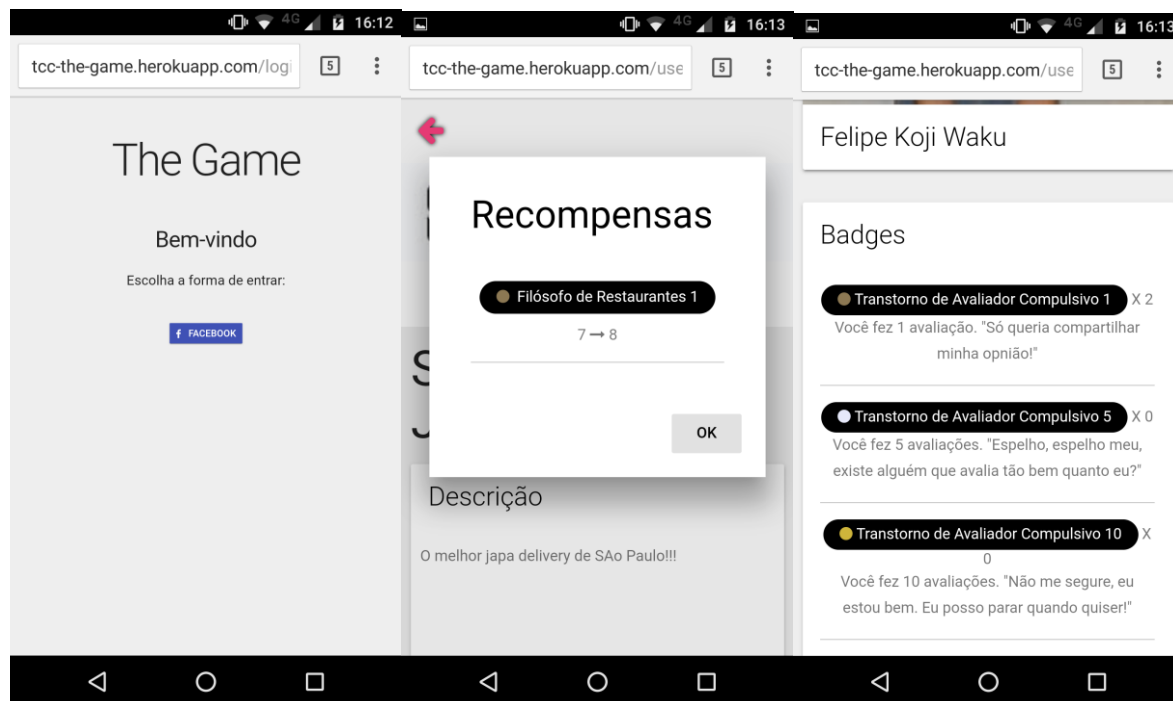
Figura 25 - Funcionalidades do segundo *sprint* organizadas no Trello



Fonte: Trello

Com o *sprint* encerrado as telas das funcionalidades desenvolvidas podem ser vistas nas figuras a seguir:

Figura 26 - Funcionalidades: Autenticação com Facebook, *Gamification* e painel do usuário



Fonte: Sistema The Game

Dessa forma todas as funcionalidades do *Product backlog* foram concluídas e o grupo pode passar para a validação com o usuário.

3.2.4. Validação com o usuário

Assim como na etapa de testes da fase de Inovação, foi feita a validação do protótipo desenvolvido com o usuário e, novamente, serão utilizados 5 usuários para os testes, baseado nas pesquisas de Jakob Nielsen. (Nielsen, 2000)

A escolha dos usuários, assim como a feita na fase de testes da etapa de inovação, foi feita com pessoas que trabalham ou já trabalharam em regiões com restaurantes comerciais, ou seja, que se encaixam no público alvo estudado.

Durante os testes foram encontrados alguns *bugs* como: alguns restaurantes não tinham pratos e seu preço médio zerava, alguns pratos não tinham descrição, alguns pratos não tinham preço e as notificações não abriam em celulares iOS.

Os resultados dos testes podem ser vistos na tabela 3 do apêndice 4, resultados da validação com o usuário do produto final.

Pela tabela 3 pode-se perceber alguns pontos interessantes que se repetiram desde a fase de design thinking como a importância da localização e do preço para a escolha de restaurantes e o uso da filtragem ou busca pelos usuários para escolher um local.

Nos resultados existem diversos pontos com relação ao design do sistema como as cores, tamanho dos nomes, textos difíceis de ler e repetição de algumas informações ou falta delas em outras telas.

Alguns usuários tiveram dificuldades de ver o propósito do sistema, achando que servia apenas para procurar quaisquer restaurantes ou era um “cardápio online”.

Os usuários pareceram gostar de ganhar recompensas ao realizar as tarefas, mas o sistema adotado no sistema em que o usuário ganha recompensas repetidas se mostrou confuso para os usuários.

3.3. Resultados do produto

O sistema apresentou diversos fatores que agradaram os usuários como o sistema de recompensas, as fotos das comidas e os preços, além de outros pontos que não agradaram muito como a falta da localização, filtragem e aspectos design.

As simplificações feitas para agilizar o desenvolvimento como a retirada do mapa da localização e a simplificação da filtragem não se mostraram muito bem aceitas pelos usuários, confirmando os aspectos da fase de inovação em que os usuários elogiaram bastante a localização dos restaurantes e pediram por um método de filtragem por preços ou localização.

O sistema The Game não se mostrou um sucesso, entretanto se fossem adicionadas mais funcionalidades que agregam valor para o usuário como mapas de como chegar aos restaurantes, uma filtragem mais elaborada, reestruturar o sistema de recompensas e correções fossem feitas de forma a melhorar seus aspectos

visuais e questões de usabilidade o sistema poderia apresentar uma ideia mais clara do seu propósito e ajudar os usuários a conhecer novos locais para almoçar.

3.4. Operação do processo

Em diversos pontos do trabalho poderiam ser feitas melhorias de forma que o projeto tivesse melhores resultados ou para que próximos projetos tenham melhores resultados.

Na fase de Inovação o grupo recomenda que a execução iterativa do processo de *Design Thinking*, executando diversas vezes todo o ciclo até que se chegue à uma solução que traga maior valor para o usuário. A repetição dos ciclos permitiria maior empatia com o usuário, entendendo melhor seus processos e problemas e chegar a uma solução mais próxima do ideal.

Ainda na fase de Inovação poderia ser utilizado outro processo como o *Lean Startup* no lugar de *Design Thinking*, que ao focar em uma produção mais rápida da ideia, permitiria mais iterações do ciclo de inovação, possivelmente permitindo que ideias inovadoras surgissem.

Com relação à fase de Arquitetura e Desenvolvimento é importante que todas as funcionalidades vistas na fase de Inovação sejam contempladas nelas, pois a fase de Inovação que traz o que o usuário deseja ou precisa, evitando realizar simplificações.

Em relação ao processo como um todo e na sua aplicação em empresas deve ser considerado que a equipe que realizaria cada uma das fases pode ser diferente, então seria necessária maior atenção aos processos de *handoff* entre as equipes, em que elas passariam os documentos para a outra, por exemplo, a equipe de inovação deve dar maior atenção aos documentos gerados para os arquitetos de *software* para que a modelagem feita contemple o que o usuário deseja.

4. Conclusões

O projeto buscava trazer um processo que realizaria a integração de técnicas de inovação com técnicas de arquitetura de *software* com o objetivo de criar produtos inovadores com foco no usuário. E além do processo deveria ser feito um case com o desenvolvimento de um produto inovador para restaurantes, podendo ser os clientes ou os gerentes.

Como pode-se ver no capítulo 3, o processo proposto se mostrou bastante eficaz para entender os problemas do usuário, chegar à ideias que atendam os problemas do usuário, validar essas ideias, realizar uma arquitetura que facilita o desenvolvimento do sistema e um desenvolvimento ágil e organizado. Tendo destaque em todo o processo as validações com os usuários que muitas vezes trouxeram muitas ideias novas e problemas não percebidos.

Foi possível executar todas as fases do processo e avaliar a importância de cada um deles. A fase de Inovação se mostrou muito eficiente em entender o usuário e chegar à soluções eficientes para seus problemas. Para chegar a uma solução mais inovadora nessa fase seria interessante repeti-la com todos os aprendizados da primeira iteração e dessa forma entender ainda mais o usuário. É importante ressaltar este processo não preza acertar na primeira vez, mas sim chegar cada vez mais próximo da solução ideal a cada vez que se realiza um ciclo de *Design Thinking*.

As integrações com as técnicas de arquitetura de software não apresentaram muita dificuldade, principalmente pelo fato do grupo ter participado dos processos de idealização da solução, facilitando o entendimento da necessidade do cliente e da solução proposta para ele.

No entanto, caso as pessoas que forem realizar a etapa de inovação para encontrar a solução para o cliente não sejam as mesmas que irão desenvolver o produto, recomenda-se que sejam feitos os documentos de engenharia de produto e descrição arquitetural, para compartilhar o entendimento dos processos do cliente e a documentação mais detalhada da solução.

A fase de desenvolvimento se mostrou bastante eficiente, desde a criação do *Product backlog* até a entrega do último *sprint*. As funcionalidades foram tiradas facilmente do protótipo da fase de Inovação e a divisão das tarefas pode ser feita facilmente devida a modularidade da arquitetura.

O feedback do produto final desenvolvido mostrou que ele não atingiu todas as expectativas do grupo por não trazer muita inovação, mas foi uma solução com diversos aspectos que agradaram os usuários como as recompensas. Isso ocorre provavelmente pela falta de empatia com o usuário em não ter repetido o processo de *Design Thinking* na fase de Inovação.

Porém, o processo proposto permitiu que os projetistas do sistema evitassem o gasto do tempo com funcionalidades ou esforços que não agradaria os clientes, evitando assim um gasto de tempo que poderia ser muito maior para um grande projeto.

O projeto trouxe para os integrantes do grupo uma grande oportunidade de conhecer técnicas de Inovação e realizar a integração delas com as técnicas de arquitetura estudadas durante o curso.

Para start-ups que desejam lançar um novo produto no mercado, esse projeto apresentou um processo se mostrou importante para poder realmente entender o usuário e criar um produto que traga valor para ele.

Para a engenharia de *software* esse projeto traz outra visão de como tratar os requisitos, mudando o foco para o usuário em vez da visão de sistema. O projeto mostrou que conversar com os usuários, entendê-los e testar ideias pode ser feito rapidamente e traz grande valor para o projeto por trazer algo que realmente resolve problemas do usuário e a integração das técnicas é feita de maneira fácil.

REFERÊNCIAS

- Archer , James, **What is “design thinking”?**. Disponível em:
<<http://jamesarcher.me/design-thinking>> Acesso em: 16 Jan 2017
- Beck, Kent et al., **Manifesto for Agile Software Development**. Disponível em:
<<http://agilemanifesto.org/>> Acesso em: 9 Nov 2016.
- Beyhl , Thomas e Giese, Holger. **Connecting Designing and Engineering Activities II**. Design Thinking Research: Part of the series Understanding Innovation pp 211-239. 2014
- Bourque P., Fairley, R.E., et al., **Guide to the Software Engineering Body of Knowledge**, 3ª Edição, IEEE Computer Society, 2014.
- Clements, Paul et al., **Documenting Software Architectures Views and Beyond**, 2ª Edição, Addison-Wesley Professional, 2010.
- Christensen, Clayton M., **The Innovator's Dilemma: When New Technologies Cause Great Firms to Fail (Management of Innovation and Change)**. 1ªedição. Harvard Business Review Press, USA. 1995.
- Christensen, Clayton M. et al. **What is disruptive innovation**. 12p. Harvard Business Review, Dezembro de 2015.
- Duarte, Jefferson, **Design Thinking – Parte 1 25 Técnicas e Ferramentas**. Disponível em: <<http://www.gp4us.com.br/design-thinking-1/>> Acesso em 23/11/2016
- Häger, Franziska et al. **DT@Scrum: Integrating Design Thinking with Software Development Processes**. 27p. Springer International Publishing Switzerland, 2015.

Heroku. **Precificação do Heroku**. Disponível em: <<https://www.heroku.com/pricing>>
Acesso em 17 Nov 2016.

Horn, Guilherme. **O que é Fintech**. Disponível em: <<http://finnovation.com.br/o-que-e-fintech/>> Acesso em 10 Nov 2016

JustInMind. **Site do JustInMind**. Disponível em: <<https://www.justinmind.com/>>
Acesso em 23 Nov 2016.

Knapp, Jake, **Sprint: How to Solve Big Problems and Test New Ideas in Just Five Days**. 1ª Edição. Simon & Schuster, 2016

Microsoft Patterns & Practices Team, **Microsoft® Application Architecture Guide**, 2ª Edição, Microsoft Press, 2009.

Nielsen, Jakob . **Why You Only Need to Test with 5 Users** . Disponível em:
<<https://www.nngroup.com/articles/why-you-only-need-to-test-with-5-users/>> Acesso em 11 Set 2016

Nubank. **Site do Nubank**. Disponível em: <www.nubank.com.br> Acesso em 10 Nov 2016.

Plattner, Husso. **Bootcamp bootleg**. 1ªedição. d.school, Instituto de Design de Stanford, 2010.

Prechelt, Lutz. **Plat_Forms: A Web Development Platform Comparison by an Exploratory Experiment Searching for Emergent Platform properties**. IEEE Transactions on Software Engineering 37(1):95-108, Jan/Fev 2011.

Pressman, Roger S. e Maxim, Bruce, **Software Engineering: A Practitioner's Approach**, 8ª Edição, McGraw-Hill Science/Engineering/Math, 2014.

Project Management Institute. **A GUIDE TO THE PROJECT MANAGEMENT BODY OF KNOWLEDGE.** 5ª edição. Project Management Institute, Inc., Pennsylvania, USA. 2013.

Ries, Eric. **The Lean Startup: How Today's Entrepreneurs Use Continuous Innovation to Create Radically Successful Businesses.** 1ª Edição. Crown Business. 2011.

SEBRAE-SP - **Serviço de Apoio às Micro e Pequenas Empresas de São Paulo. Causa Mortis: o sucesso e o fracasso das empresas nos primeiros cinco anos de vida.** Pág. 6 Disponível em:
http://www.sebraesp.com.br/arquivos_site/biblioteca/EstudosPesquisas/mortalidade/causa_mortis_2014.pdf Acesso em: 13 Abr 2016.

Schwaber, Ken e Sutherland, Jeff, **The Definitive Guide to Scrum: The Rules of the Game.** Disponível em:
 <<http://www.scrumguides.org/docs/scrumguide/v2016/2016-Scrum-Guide-US.pdf#zoom=100>> Acesso em: 10 Nov 2016.

Seelig, Tina. **inGenius A CRASH COURSE ON CREATIVITY.** 1ª edição. HarperOne, USA. 2012.

Setser, Bryan e Morris, Holly, **Building a Culture of Innovation in Higher Education: Design & Practice for Leaders.** 2015.

SOARES, Michel dos Santos. **Comparação entre Metodologias Ágeis e Tradicionais para o Desenvolvimento de Software.** INFOCOMP Journal of Computer Science, [S.l.], v. 3, n. 2, p. 8-13, nov. 2004. ISSN 1982-3363.

Vinnakota, Tirumala Rao. **Integration of Design Thinking with Strategy and Innovation in an Enterprise Context.** 318 MGPL Narayana Business Systems & Cybernetics Centre, Tata Consultancy Services Ltd, Hyderabad, India, 2014.

APÊNDICE 1 - Modelo de entrevistas da fase de empatia

Formato da entrevista com Clientes

1. Apresentar o grupo
2. Falar que estamos fazendo um trabalho sobre restaurantes
3. Perguntar sobre a pessoa
 - a. Nome
 - b. Idade
 - c. Onde você costuma almoçar?
 - d. Você tem algum critério para escolher um lugar para almoçar?
 - e. Qual foi o melhor restaurante que você já foi?
 - i. Por que?
 - ii. Pode me contar um pouco como foi?
 - f. Você já teve algum problema nesses lugares?
 - i. “Você lembra... De algum problema que você já teve em restaurantes?”
 - ii. Se não lembrar de nada tentar “provocar” com coisas do tipo “Nossa, você nunca teve problema com a comida ou com fila nesses lugares?”
 1. Dividir a conta
 2. Fila para pagar
 3. Surpresa no cardápio (Preço muito alto/não gosta das opções)
 4. Fila para entrar
 5. Falta de lugar
 6. Demora para chegar comida
 - g. Como você costuma escolher onde vai comer/almoçar?
 - h. Você sabe o que quer comer antes de sair para almoçar? Ou escolhe só no lugar?
 - i. Você costuma ter quanto tempo pro almoço? Você usa todo esse tempo para comer?
 - j. Você costuma almoçar todo dia no mesmo lugar, ou varia bastante? Por que?

- k. Perguntar de novo (e)
- 4. Agradecer

Formato da entrevista com Donos de restaurantes

1. Apresentar o grupo
2. Falar que estamos fazendo um trabalho sobre restaurantes
3. Perguntar sobre o restaurante:
 - a. Como funciona o processo no dia a dia?
 - b. Qual um problema que acontece frequentemente aqui?
 - i. Cliente louco, reclama, não sabe o que tá acontecendo
 - c. Como você calcula o faturamento?
 - d. Como você checa o estoque, fornecimento de produtos?
 - e. (self-service) Como que ele sabe quando tem que começar a fazer comida antes de esvaziar?
 - f. Você tem algum programa de fidelidade? Por que?
 - g. Você faz divulgação? Você se preocupa em atrair mais clientes? Por que?
 - h. Você monitora o atendimento ao cliente dos seus funcionários?
 - i. Já aconteceu do restaurante estiver lotado, a ponto dos clientes irem embora por causa da fila?
 - i. Como foi?
 - ii. O que você fez?
 - iii. Por que?
 - j. Quando foi a última vez que você teve algum problema no preparo dos pratos?
 - i. Muito atraso?
 - ii. Faltou comida no estoque?
 - iii. Prato queimou
 - iv. Prato caiu no chão
 - k. Já teve alguma vez que você teve dificuldade em atender os clientes?
 - i. Cliente furando fila, confusão
 - ii. Cozinha não conseguiu dar conta dos pratos?

I. Já teve alguma vez que o pedido saiu errado? Como você solucionou?

Por que?

4. Agradecer

**APÊNDICE 2 - Resultados das entrevistas da fase de empatia na
fase de Inovação**

Entrevistas com Clientes

Camila, 20 anos

Onde você costuma almoçar?

Em casa ou restaurantes

Ex: Restaurante de Temaki, com carteirinha de fidelidade

Você tem algum critério para escolher um lugar para almoçar?

Fome, se for muita fome, procura uma lanchonete, senão, um restaurante japonês ou por quilo

Custo/Benefício

Cupons (Usa o aplicativo do Groupon, que filtra os restaurantes da localidade, filtra por preço)

Exclui os pratos com ingredientes que não gosta

Qual foi o melhor restaurante que você já foi?

Restaurante faz showzinho, apresentações na hora de fazer o prato

Comida diferente e de boa qualidade

Você já teve algum problema nesses lugares?

Prato errado, reclamou e o restaurante mandou uma sobremesa como desculpa

Anotaram pedido errado

Garçom ruim, não fez nada, pq ficou com medo de reclamar

Atendimento ruim, fast-food, acontece bastante

Não fala português

Prato com comida fedida, reclamou, trocaram o prato e enviaram sobremesa como desculpas

Como você costuma escolher onde vai comer/almoçar?

Usa aplicativos para conhecer lugares/pratos novos

Four Square, procura pratos com views positivas

Procura por promoções

Pega pratos indicados por revistas, amigos

Você sabe o que quer comer antes de sair para almoçar? Ou escolhe só no lugar?

Normalmente já sabe o que quer

Em restaurantes por quilo costuma variar

Você costuma almoçar todo dia no mesmo lugar, ou varia bastante?

Costuma variar, mas tem os restaurantes que conhece e gosta de voltar

Procura por “Serendipity” - Felicidade de comer algo quando não se espera que fosse bom

Carol, 22 anos**Onde você costuma almoçar?**

Shopping, restaurantes ou em casa

Ex: Hamburger H³ - Customização de hamburger

Você tem algum critério para escolher um lugar para almoçar?

Custo/Benefício

Procura lugares que não estão muito cheios

Se estiver muito cheio, desiste

Procura por pratos que já conhece

Qual foi o melhor restaurante que você já foi?

Hideki - Comida diferente, feita na hora

Kabura - Espetinhos

Cantina - Muita comida

Você já teve algum problema nesses lugares?

Habibs(Shopping) - Cabelo no suco, reclamou e trocaram o suco

Self-service - Em uma fila muito grande, outra cliente furou fila, não reclamou para não fazer “barraco”

Outback - Fila muito grande e demorada, serviram petiscos e bebidas para os clientes não desistirem

Demora para atender

Atendente grossa, Carol não costuma reclamar, mas o pai sim

Como você costuma escolher onde vai comer/almoçar?

Procura restaurantes que já conhece

Você sabe o que quer comer antes de sair para almoçar? Ou escolhe só no lugar?

Normalmente já sabe o que vai querer

Você costuma almoçar todo dia no mesmo lugar, ou varia bastante?

Vai onde conhece

Vai por recomendações, restaurantes cheios é um bom sinal

Vitor, 20 anos

Onde você costuma almoçar?

Shopping, restaurantes ou em casa

Ex: McDonalds

Você tem algum critério para escolher um lugar para almoçar?

Custo/Benefício

Muita comida

Recomendação de outras pessoas

Qual foi o melhor restaurante que você já foi?

Cantina - Muita comida

Hong He - Muita comida

Você já teve algum problema nesses lugares?

Self-service se plaquinha, não dava pra saber o que era a comida

Pedi prato para viagem, trouxeram o normal, reclamou e corrigiram

Cobraram coisa a mais na conta, reclamou e corrigiram

Como você costuma escolher onde vai comer/almoçar?

Ouve recomendações dos amigos

Você sabe o que quer comer antes de sair para almoçar? Ou escolhe só no lugar?

Costuma pedir o que já conhece

Você costuma almoçar todo dia no mesmo lugar, ou varia bastante?

Normalmente vai onde conhece por medo dos outros lugares não serem bons

Felipe Toyoda, 23 anos

Onde você costuma almoçar?

Restaurante por Kg, dificilmente vai em PF

Almoça em grupo

Você tem algum critério para escolher um lugar para almoçar?

Variedade e preço

Custo benefício

Qualidade da comida

Qual foi o melhor restaurante que você já foi?

Mineiro - Comida barata, muita quantidade, atendentes legais, já teve sobremesa de doce de leite de graça

Você já teve algum problema nesses lugares?

Restaurante com o atendente que não sabia cortar carne, chefe da empresa reclamou, nunca mais viram esse atendente

Restaurante que servia comida fria, as pessoas frescas do grupo deixaram de ir

Restaurante com problemas de logística de como formar fila, tem que ficar esperando as pessoas pegarem a sala, mesmo que ele não quer comer salada

Como você costuma escolher onde vai comer/almoçar?

Indicações

Você sabe o que quer comer antes de sair para almoçar? Ou escolhe só no lugar?

Costuma decidir na hora

Você costuma almoçar todo dia no mesmo lugar, ou varia bastante?

Varia bastante

Observações

Almoça em região com muitos restaurantes

Acha reclamar muito trabalho, não costuma fazer isso

Tem 1h de almoço, geralmente gasta mais tempo, não tem problema, por que as pessoas recebem por hora de trabalho

Fabio Wong, 23 anos

Onde você costuma almoçar?

Restaurante por Kg, as vezes em PF

Você tem algum critério para escolher um lugar para almoçar?

Distância, preço e sabor

Qual foi o melhor restaurante que você já foi?

Churrasco - Comida boa, bem servida e ambiente agradável

Restaurante Japonês - Comemoraram o aniversário de um funcionário, comida muito boa e cara, mas o chefe pagou

Você já teve algum problema nesses lugares?

Pedido veio errado, reclamou e trouxeram o correto

Troco veio errado, reclamou e corrigiram

Problema na hora de pagar, fila muito grande

Muita gente na entrada, escolheu outro lugar

Muitos funcionários para poucos clientes

Já foi em restaurantes com um serviço ruim, que tinha inseto na comida

Como você costuma escolher onde vai comer/almoçar?

Indicação

Se é aniversário, deixa a pessoa escolher

Você sabe o que quer comer antes de sair para almoçar? Ou escolhe só no lugar?

Decide no local

Você costuma almoçar todo dia no mesmo lugar, ou varia bastante?

Varia bastante

Observações

Almoça em região com muitos restaurantes

Não conseguiu terminar nenhum cartão de fidelidade, mas usava (Levava para os restaurantes)

Almoço costuma demorar 1h~1h30, usa todo o tempo

Vinícius Oyama, 27 anos

Onde você costuma almoçar?

Restaurantes não muito caros, não muito baratos

Normalmente são restaurantes que servem pratos feitos ou restaurantes por quilo baratos

Também vai em restaurantes self-services

Cardápio varia bastante:

- Arroz-feijão
- Wrap
- Comida Japonesa
- Comida caseira

Você tem algum critério para escolher um lugar para almoçar?

Custo benefício => Quantidade de comida / preço

Qual foi o melhor restaurante que você já foi?

(Restaurante Japonês na frente do escritório)

Gostou do Salmão

Comida boa por um bom valor

Você já teve algum problema nesses lugares?

Lentidão na entrega dos pratos

Erro nos pedidos, esqueceram algo que foi pedido

No Restaurante Rainha da Brigadeiro, já chegou a esperar mais de 30 minutos pelo prato.

Não reclamou, pois considerou "Normal" (Acredita que estavam esperando todos os pratos da mesa ficarem prontos para levá-los todos de uma vez)

Em um outro restaurante, pediu Stroganoff e não gostou. Não reclamou pois não estava com muita fome, mas nunca mais pediu esse prato

Fila para pagar, acontece quando tem pessoas lentas, muita gente para pagar (A própria empresa quando vai almoçar costuma formar fila, já que todos pagam ao mesmo tempo e esperam que todos terminem de comer para começarem a pagar)

Como você costuma escolher onde vai comer/almoçar?

Ouve recomendações dos amigos

Você sabe o que quer comer antes de sair para almoçar? Ou escolhe só no lugar?

Costuma decidir enquanto está indo almoçar, quando decide o lugar, costuma já saber o que vai pedir

Você costuma almoçar todo dia no mesmo lugar, ou varia bastante?

Gosta de ir em restaurantes novos, já conheceu restaurantes bons dessa forma

Observações

Nunca reclamou com o gerente do estabelecimento

Reclama com o garçom quando demora para chegar o prato

Costuma não reclamar quando a pessoa está brava, pois acredita que ela pode estar tendo um “dia ruim”

Acha que as vezes falta “carinho” no atendimento

Não gosta de atendentes mau humorados, mas se trabalharem direito não se importa muito

Acha que seria interessante um Aplicativo para pagar a conta ou uma forma mais rápida de pagamento, como pagar passando o celular sobre um sensor que identifica a conta e cobra o valor dela

Acha mais fácil escolher comida quando tem foto dela

Entrevistas com Restaurantes

Ya Hibili- Dirce

O que você acha que faria seus clientes não voltarem mais para esse estabelecimento?

Para manter um cliente é necessário bom atendimento e cuidar de clientes mau humorados e com pressa.

Ela disse que possui clientes com mais de 20 anos de casa.

Como você costuma fazer para atrair seus clientes?

Bom atendimento, com alegria e de forma ágil e demonstrar boa qualidade dos produtos

Você já fez algum sistema de fidelidade?

Com fornecedores de alimentos e com empresas

O que poderia ser melhorado?

Automatização em um restaurante por quilo e treinamento de funcionários.

Na hora da entrevista a gerente chamou atenção duas vezes de um funcionário para ele limpar a mesa.

Ela disse que existem diversos aplicativos de comida, mas estes atacam os problemas errados.

Caldeirão da Brigadeiro - Thiago

Qual é o problema no seu dia a dia no restaurante que você considera o mais significativo?

O ambiente do restaurante não se adequa aos padrões de clientes de classes mais altas como A, B ou C como o local ter aparência de “boteco” e não possuir ar-condicionado. O seu público atual é do de classe C e D devido aos preços baixos, bom atendimento e boa qualidade, logo atende apenas um nicho.

Resaltou que atende a um público exigente.

Jonathan - Porto Café

Qual é o problema no seu dia a dia no restaurante que você considera o mais significativo?

Resaltou dois problemas, um do dia a dia e outro operacional. O problema cotidiano é a falta de movimento devido a queda da demanda que pode ser visto que muitas salas dos prédios

comerciais estão vazias e as pessoas tem um VR limitado (R\$14~15), então geralmente levam comida de casa.

O problema operacional é a impossibilidade de verificar se as mesas do segundo andar estão vazias, então seria necessário recursos para investir em câmeras ou funcionários.

Capricho

Qual é o problema no seu dia a dia no restaurante que você considera o mais significativo?

Não há problemas

Nome não encontrado - Ana Carla

Qual é o problema no seu dia a dia no restaurante que você considera o mais significativo?

Atendimento ruim dos funcionários, vindo da reclamação dos clientes. Soluciona tentando demitir.

Eugênio bar - Camila

Qual é o problema no seu dia a dia no restaurante que você considera o mais significativo?

Com a crise está tendo problemas em atrair novos clientes, para isso faz promoções.

El Malak - Munir

Qual é o problema no seu dia a dia no restaurante que você considera o mais significativo?

O espaço de cadeiras.

O que você acha que faria seus clientes não voltarem mais para esse estabelecimento?

Aumento de preço, devido a grande concorrência da região (a paulista é uma das regiões mais baratas para comer) e a qualidade da comida cair.

Você já fez algum sistema de fidelidade?

Não acha necessário. Atrai clientes com a qualidade e preço justo.

Café trianon - Alexandre

Qual é o problema no seu dia a dia no restaurante que você considera o mais significativo?

A crise econômica que fez a demanda cair. É necessário fazer mudanças como propagandas e bom atendimento.

Problema nos funcionários que atendem mal.

QSabor - Alan

Qual é o problema no seu dia a dia no restaurante que você considera o mais significativo?

A crise econômica que causa o aumento de preços dos fornecedores e não é possível repassar o valor para os consumidores devido a menor demanda (pareceu frustrado).

Problema com funcionários que atrasam ou faltam sem deixar notícias e sempre precisar ter um funcionário a mais para cobrir outro.

Bruno Bueno, 28 anos

Onde você costuma almoçar?

Restaurantes variados

Cada dia almoça em um lugar

Você tem algum critério para escolher um lugar para almoçar?

Custo benefício é muito importante

Comida boa, com qualidade

Se está chovendo, vai em algum lugar perto

Se está frio, algum lugar com comida apimentada

Se está com meninas no grupo, algum lugar que divide comida, já que elas não comem muito

Qual foi o melhor restaurante que você já foi?

Sushi Gen - Comida muito boa

Vegas - Preço bom, Proposta diferente para servir a comida, ingredientes bons e variados

Você já teve algum problema nesses lugares?

Em uma Lanchonete, já aconteceu de pedir um lanche sem vinagrete. Trouxeram com o vinagrete e cobraram por isso. Na semana seguinte, pediu de novo sem vinagrete, veio de novo com.

Todos na mesa comem e tem que ficar esperando um terminar

Chegou em um restaurante e não tinha mais comida

Estava pegando comida e tinha acabado a paqueta, reclamou e levaram pra ele depois

Mudaram o esquema de um restaurante, parou de ir nele

Restaurante não tinha lugar para todos do grupo sentarem, foram pra outro lugar (Talvez se esperassem as pessoas saírem valesse a pena, mas foram embora)

Como você costuma escolher onde vai comer/almoçar?

Indicações, panfleto com preço e comida que chamam atenção

*Japones 10 reais/Kg ninguém vai

Você sabe o que quer comer antes de sair para almoçar? Ou escolhe só no lugar?

Costuma decidir no local

Você costuma almoçar todo dia no mesmo lugar, ou varia bastante?

Gosta de ir em restaurantes novos

Observações

Preço é muito importante

Já deixou de ir em restaurantes que subiram o preço

Acha difícil conhecer novos lugares, falta forma de fazer isso

Comentou sobre um blog antigo com os restaurantes de PF

Gostaria de um app para comidas do dia-a-dia

Como tem muita opção de restaurantes na região, não costuma valer a pena ir 10 vezes no mesmo mês em um só lugar.

Não quer andar com 10 bilhetes de fidelidade, nunca conseguiu usar

Gosta de atendimento customizado, mas não faz questão, prefere: Chegar, comer e pagar

Arthur Fonseca, 28 anos

Onde você costuma almoçar?

Restaurantes perto de onde trabalha, não volta pra casa

Prefere que dê pra ir andando

Normalmente vai em restaurantes por Kg, pois é mais rápido

Prefere restaurantes de Prato Feito, pois é só pedir e o prato vem

Você tem algum critério para escolher um lugar para almoçar?

Tempo, vai no restaurante mais rápido

Qual foi o melhor restaurante que você já foi?

Mineiro - Comida farta, boa, preço razoável, mas é longe (Demora pra chegar a pé), gosta de levar gente nova lá, costumam gostar

Você já teve algum problema nesses lugares?

Já teve vezes que demorou muito, as pessoas perceberam que eles tinham esquecido algum pedido (Problema de comunicação), reclamaram com o garçom

Em um outro restaurante, Comida já veio fria, sempre erravam os sucos (O grupo apostava pra ver quantos sucos iam errar no dia, normalmente de 3~5), nunca reclamaram
Conta nunca tinha o mesmo valor (Mesmo quando duas pessoas pediam as mesmas coisas), reclamaram, depois pararam de ir.

Fila na hora de pagar, mas o restaurante só tinha 1 máquina para cobrar, não acha que é um grande problema, por que tem que esperar todos pagarem pra ir embora do restaurante

Como você costuma escolher onde vai comer/almoçar?

Escolhe antes de sair, na porta do escritório, pelo menos umas 10 pessoas tem que concordar

Você sabe o que quer comer antes de sair para almoçar? Ou escolhe só no lugar?

Escolhe antes de sair, a menos que seja restaurante por Kg.

Se sabe o lugar já sabe o que vai comer

Você costuma almoçar todo dia no mesmo lugar, ou varia bastante?

Costuma variar bastante

Descobre os lugares andando por aí, por indicações

Observações

Região com muitos restaurantes, não costuma se estressar quando algum tem problema
Acha que reclamar é estressante

Gosta de um restaurante que é bom, mas é muito caro. Depois de um certo horário, eles baixam o preço. Já viu gente enrolando na fila para pagar menos.

Jedi Burger, ficou famoso e descobriram que era do lado do escritório, descobriu por rede social

Empresa já fechou parceria com restaurante que veio até a empresa, passando os dados do restaurante, não acontece com frequência

Costuma perder cartões, então nunca conseguiu usar cartão de fidelidade

APÊNDICE 3 - Resultados da fase de testes da fase de Inovação

Tabela 1 - Resultado dos testes do The Game

Tópico	Marcela	Rafael	Felipe	André	Fábio
Nome do app na barra de título	X		X		X
Botão de adicionar review na listagem não ficou claro	X	X		X	
Filtragem de restaurantes	X	X	X		X
Indicação de preço na listagem	X		X	X	X
Gostou da badge	X	X	X		X
Gostou do mapa/localização	X		X	X	X
Gostou das comidas específicas nos reviews	X	X			
Novos lugares	X	X		X	
Foursquare++	X	X			
Não conseguiu identificar o dono do review		X		X	
Não entendeu classe no status		X	X	X	
Mais detalhes das badges		X	X	X	X
Um review por restaurante	X	X	X	X	
Listagem desorganizada		X	X		
Feedback ao avaliar um review			X		

Restaurante já preenchido na criação de review			X		
Distância dos restaurantes na listagem			X		
Faltou credibilidade no review				X	X
Achou informação de endereço insuficiente				X	
Barra de navegação pouco intuitiva				X	
Achou que o +1 -1 dos reviews eram as suas avaliações				X	
Sentiu falta do modo de funcionamento do restaurante				X	
Gostou da foto do local				X	X
Achou que o perfil do usuário era mapeado pela localização				X	
Sentiu falta de restaurantes para comemorar				X	
Gostaria de contato do restaurante				X	

Tabela 2 - Resultados dos testes do É Hoje

Tópico	Marcela	Rafael	Felipe	André	Fábio
Achou que era um app de receitas	X				
Achou com dificuldades o	X				

endereço					
O local importa mais que o prato	X				
Ideia não ficou muito clara	X	X	X	X	X
Gostou da filtragem	X				
Outras opções para almoços em grupo	X				
Lista de favoritos	X	X	X		
Sentiu falta de um tutorial	X				
Não entendeu o "É hoje"		X	X	X	
Achou que iam aparecer comidas repetidas		X			
Gostou para lugares com pratos do dia		X			
Gostou das fotos		X		X	
Gostou do acesso direto para o mapa			X		
Não entendeu muito bem a interface inicial			X		
Não achou o nome do restaurante				X	
Faltou credibilidade				X	
Usuario ia trabalhar muito				X	
Estranhou a falta de perfil				X	

Achou que o app ia indicar as melhores comidas da região				X	
Sentiu falta de preço					X
Preferia mapa					X
Achou que apresentaria pratos aleatórios					X

APÊNDICE 4 - Resultados da validação com o usuário do produto final

Tabela 3 - Resultados da validação com o usuário do produto final

Tópico	Marcela	Rafael	Felipe	Carol	Fábio
Não gostou das cores usadas	X			X	
Logos dos restaurantes com títulos ficou repetitivo	X				
Nome e preço dos pratos difíceis de ler em alguns restaurantes	X			X	X
Sentiu falta do preço médio dos restaurantes no detalhe de restaurante	X				
Nome de restaurantes muito extensos	X				
Sentiu falta da localização dos restaurantes	X	X	X	X	X
O que acontece quando não há retorno da busca?	X		X		
Achou título das recompensas estranhos	X				

Achou estranho ganhar recompensas repetidas	X	X	X	X	X
Achou que o sistema servia para ver cardápios	X	X			
Gostou do preço dos restaurantes e pratos		X			X
Gostou da lista de pratos de cada restaurante	X	X			
Gostou do detalhe de pratos e das avaliações dos pratos		X			
Gostaria de poder salvar os restaurantes que gostaria de ir		X			
Gostou das recompensas		X	X	X	X
A listagem não ficou claro se são os restaurantes próximos		X		X	
Não entendeu a necessidade de login			X		X
Gostou do nome das recompensas			X		X

Ficou confuso com as transições de tela			X		
Gostaria de avaliar o restaurante em geral			X		
Achou a busca só pelo nome muito simples			X		X
Sentiu que o sistema estava incompleto			X		
Não gostou de pressionar OK em todas as notificações			X		
Gostou das fotos das comidas		X		X	
Gostaria de mais fotos do restaurante e pratos				X	
Não viu a caixa de texto para escrever avaliações	X	X		X	
Achou que para quem não joga o sistema de recompensas não faz muito sentido					X
Gostaria de dar estrelas nas					X

avaliações					
Achou as avaliações sem credibilidade					X
Achou que o sistema servia para encontrar restaurantes em qualquer lugar					X
Achou que servia para procurar restaurantes na região				X	