

***The Beer Point*, aplicativo para avaliar, classificar e organizar cervejas**

David Abt, Rodrigo Souza D'ávila

Departamento de Ciência da Computação – Universidade Estadual do Centro-Oeste
(UNICENTRO)

Campus Cedeteg – 85.040-080 – Guarapuava – PR – Brasil

david_abt@hotmail.com, rsdavila@gmail.com

Abstract. *Collecting beer is a hobby for many people, especially craft and imported beers. But sometimes it's hard to keep a collection of all beers already tasted. On this basis, this paper describes the development of an application in Android to manage a collection of beers and make the activity of evaluating them simpler. For this purpose, tools were used to assist in the process, such as the integrated development environment Android Studio and the object oriented database, Parse.*

Resumo. *Colecionar cervejas é um hobby de várias pessoas, principalmente cervejas artesanais e importadas. Porém às vezes é difícil manter uma coleção com todas as cervejas já degustadas. Com base nisto, este trabalho descreve o desenvolvimento de um aplicativo, em Android, para gerenciar uma coleção de cervejas e tornar a atividade de avaliá-las mais simples. Para este fim foram utilizadas algumas ferramentas para auxiliar no processo, como o ambiente de desenvolvimento integrado Android Studio e o banco de dados orientado a objetos, Parse.*

1. Introdução

Nos últimos trinta anos o mundo cervejeiro passou por grandes mudanças, que lançaram uma luz sobre uma nova geração de cervejas e cervejarias (Tierney-Jones, 2012). O paladar do consumidor de cerveja tem se modificado com o passar dos tempos, buscando o prazer de sentir novos sabores e sensações ao degustar uma cerveja. Neste cenário entram as microcervejarias que trouxeram aos brasileiros a opção de consumir produtos exclusivos e diferenciados com vários sabores, tipos, texturas e aromas, fugindo das cervejas tradicionais que existem em grandes quantidades nos mercados (Mosher, 2009).

Assim como o mercado cervejeiro, o número de *smartphones* tem crescido consideravelmente. O Brasil é o quarto país do mundo em número de *smartphones*: 70 milhões (Guimarães, 2013). Da mesma forma como o número de *smartphones* aumentou, mudou-se também a forma de como eles são utilizados.

Os aparelhos móveis deixaram de ser simples instrumentos de comunicação e passaram a ser uma ferramenta essencial para muitas pessoas. Com o *smartphone* as pessoas estão sempre conectadas, conseguem utilizar as redes sociais, verificar *e-mails*, tirar fotos, usá-lo como mapa, além de várias outras funcionalidades que facilitam a vida do usuário. Pensando neste cenário, esta pesquisa tem como objetivo criar um aplicativo móvel para auxiliar o apreciador de cerveja a gerenciar as bebidas que ele já tomou ou está tomando. Ele conta, também, com uma funcionalidade para que o usuário

possa fazer avaliação do produto, deixar um comentário e tirar fotos para a coleção, assim, eliminando a necessidade de guardar um souvenir físico da cerveja e mantendo uma coleção digital sempre em mãos.

Este texto está organizado como segue. A Seção 2 descreve a fundamentação teórica, mostrando o uso de dispositivos móveis no Brasil e no mundo, bem como a importância deles para o dia a dia, como também apresenta um resumo sobre o mercado cervejeiro. A Seção 3 contém os materiais e métodos utilizados para o desenvolvimento deste trabalho. Na Seção 4 são apresentados os resultados obtidos e por fim, na Seção 5 são elencadas as considerações finais e trabalhos futuros.

2. Fundamentação Teórica

As cervejas industriais, em comparação com as cervejas artesanais, sempre foram o maior destaque no mercado cervejeiro, entretanto, as cervejas artesanais vêm conquistando espaço, à medida que apresentam produtos diferenciados e tem como o principal foco, a qualidade (Cervesia, 2011).

As cervejas industriais, produzidas em grande quantidade, são feitas para atingir o máximo de consumidores possíveis. Estas cervejas utilizam, em suas receitas, apenas 60% de malte e acrescentam outros cereais para diminuir os custos de produção, chegando ao consumidor com um preço relativamente baixo (Cervesia, 2011).

Por outro lado, as cervejas artesanais que são produzidas em menor escala e levam 100% de malte em sua composição, utilizam somente ingredientes nobres e selecionados focando na sua qualidade. Estes ingredientes são trazidos, muitas vezes, de países europeus, que se destacam no cultivo da matéria-prima cervejeira. O período de fermentação e maturação da cerveja artesanal acontece sem pressa, ou seja, não são adicionados produtos químicos para acelerá-lo.

Segundo os dados da Cervesia (2011), o Brasil está hoje em terceiro lugar no mundo em produção de cerveja, com produção de 12,4 bilhões de litros, atrás apenas da China (45 bilhões de litros) e Estados Unidos (35 bilhões de litros), superando a Rússia (11,6 bilhões de litros) e a Alemanha (10,8 bilhões de litros).

De acordo com dados da Associação Brasileira de Bebidas (Abrabe, 2015), considerando as classes de acordo com renda per capita, a classe C opta pelas grandes marcas, as classes A e B buscam produtos que apresentem diferenciação. Atributo fortemente encontrado nas cervejas artesanais, cujo consumo é majoritariamente masculino, com idades entre 18 e 65 anos. As mulheres que as consomem têm entre 30 e 65 anos. Isto também mostra uma crescente tendência ao consumo de quantidades pequenas de cervejas, porém de qualidade.

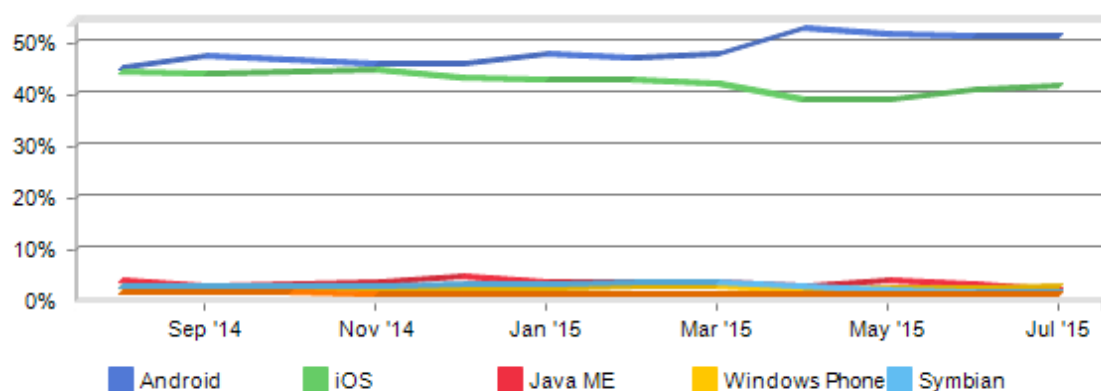
Segundo uma pesquisa realizada pelo *Google* (Google, 2013), o principal público de *smartphones*, no Brasil, são homens entre 18 e 44 anos e mulheres entre 25 e 44 anos. Com base nestes dados nota-se uma semelhança entre o público consumidor de cerveja e o público que utiliza *smartphone* diariamente. Google (2013) também definiu o *smartphone* como sendo “*um celular que oferece recursos avançados, frequentemente com funcionalidades iguais às de um computador ou a capacidade de fazer download de aplicativos*”.

Os *smartphones* estão inseridos na categoria de dispositivos móveis, que segundo Kovacs (2006) é definido como um equipamento digital que serve para realizar

tarefas diárias de trabalho, isso inclui os *Pocket PCs*, *smartphones* e relacionados. Segundo Mantovani (2010), eles são, ao mesmo tempo, uma ferramenta de gestão e difusão de conteúdos produzidos pelos usuários e um suporte para a comercialização de conteúdos padronizados. Mantovani (2010) ainda fala que tais características possibilitam aos usuários de *smartphones* a vivência do que é considerado uma cultura participativa. Esta cultura implica em novas formas de pensar, produzir e compartilhar que está fortemente vinculada à tecnologia.

No mercado de *smartphones* existem três sistemas operacionais de destaque, sendo eles o *Android* do *Google*, o *iOS* da *Apple* e o *Windows Phone* da *Microsoft* e cada um com suas vantagens. Na Figura 1 pode-se observar a porcentagem de usuários dos cinco maiores sistemas operacionais móveis no período de Agosto de 2014 à Julho de 2015. Nota-se a liderança do sistema *Android*, no mercado, em relação aos outros. Como o foco desta pesquisa é um aplicativo móvel para o sistema operacional *Android*, os outros sistemas não serão abordados neste trabalho.

Figura 1 - Porcentagem de usuários de cada sistema operacional. Fonte: Netsharemarket (2015).



O sistema operacional *Android* é baseado no núcleo Linux e foi desenvolvido pelo *Google* em conjunto com a *Open Handset Alliance*, uma aliança entre 84 empresas de hardware, software e telecomunicações, que tem como objetivo criar padrões para a indústria de telefonia móvel. Nesta aliança liderada pelo *Google* também estão nomes como a *Dell*, *Intel*, *Motorola* e *Samsung* (Lecheta, 2010).

O *Android* possui código fonte aberto e está disponível sob a licença GPL (*General Public License* - Licença Pública Geral), o que permite cada fabricante a adaptar o código de acordo com as configurações de hardware específicas.

A arquitetura do sistema operacional *Android* é dividida em camadas, onde cada parte é responsável por gerenciar os seus respectivos processos. As camadas são descritas como (Lecheta, 2010):

- Camada de Aplicações: representa as aplicações que são executadas sobre a plataforma. Nesta camada estão tanto as aplicações nativas, como aplicações criadas por terceiros;
- Camada Framework de Aplicações: ficam as APIs (*Application Programming Interface* - Interface de Programação de Aplicações) do *Android* que são utilizadas pelas aplicações que executam sobre a plataforma;

- Camada de Bibliotecas: todas as bibliotecas nativas escritas em C/C++ que fazem parte da plataforma encontram-se nesta camada. Alguns exemplos são a API do *OpenGL* para a renderização 3D e o gerenciador de banco de dados *SQLite*;
- Camada de *Runtime*: nesta camada cria-se uma instância da máquina virtual Dalvik que é uma máquina virtual criada especificamente para o *Android*, para cada aplicação;
- Camada de *Kernel Linux*: camada onde se encontra o sistema operacional da plataforma, que é baseado no Linux. Responsável pelos serviços de baixo nível, como o gerenciamento de memória e processos.

A maioria dos aplicativos *Android* é escrita na linguagem de programação Java, a qual é compilada para um *bytecode* e executada por uma máquina virtual chamada *Java Virtual Machine* (JVM), porém o *Android* não usa a JVM, mas a máquina virtual Dalvik ou *Android Runtime* (ART) (Lee, 2011). As classes Java são compiladas para um *bytecode* de formato específico que será executado pela máquina virtual Dalvik.

Um fator que leva muitas pessoas a trocarem um simples celular por um *smartphone* é a Internet. Com o *smartphone* é possível ficar conectado 24 horas, tirando proveito da principal característica deste meio de comunicação: a informação instantânea. Os *smartphones* contam, também, com uma câmera digital integrada, o que somado à conexão com Internet fazem dos *smartphones* fontes de compartilhamento de informação, pois é possível tirar uma foto, registrar eventos e enviá-los imediatamente para os seus contatos. Outro motivo pelo qual as pessoas trocam seus aparelhos celulares por *smartphones*, são os aplicativos que podem ser instalados neles. Cada usuário pode personalizar seu aparelho, instalando o que vem a ser útil no seu dia a dia.

Vários usuários de *smartphones* também são apreciadores de cervejas, especialmente cervejas artesanais e importadas, e estes usuários gostam de manter uma coleção com todas as cervejas já experimentadas, guardando a garrafa, o rótulo, a tampa ou qualquer souvenir que venha a lembrar da cerveja.

Com o surgimento de aplicativos para gerenciamento de bebidas, o usuário consegue ter uma experiência totalmente diferente no que diz respeito a colecionar cervejas, pois ficou mais fácil gerenciar as já degustadas, bastando apenas tirar uma foto da garrafa ou inserir o nome da bebida no aplicativo que a cerveja é automaticamente adicionada à sua coleção pessoal. Assim, ele consegue ter uma coleção digital sem a necessidade de guardar as garrafas.

Existem alguns aplicativos disponíveis no *Google Play* para este fim e cada um tem o seu diferencial, focando mais em uma funcionalidade do que na outra.

O aplicativo *Untappd*¹ funciona como uma espécie de rede social para apreciadores de cerveja. O usuário pode fazer *check-in* no aplicativo com a cerveja que ele está tomando. O aplicativo também utiliza o GPS (*Global Positioning System* - Sistema de Posicionamento Global) do celular do usuário para encontrar cervejas que estão perto dele. O usuário ainda pode acompanhar os amigos que também utilizam o aplicativo e ver onde e qual cerveja eles estão bebendo. A medida que o usuário faz

¹ <http://www.untappd.com>

check-in no aplicativo ele recebe emblemas como prêmio e também recomendações de cervejas com base nas já degustadas.

Outro aplicativo de gerenciamento de bebidas é o *Vivino Wine Scanner*² que ao invés de cerveja, é focado na indústria vinícola. O grande diferencial deste aplicativo está no fato do usuário poder tirar uma foto do rótulo, e esta será analisada por um algoritmo de processamento de imagens. O algoritmo irá comparar o nome do vinho com uma base de dados e retornará os detalhes do produto, caso ele esteja cadastrado. E o usuário pode optar por adicionar o vinho à sua coleção pessoal.

Apesar da existência de bons aplicativos para gerenciar bebidas, alguns deles apresentam empecilhos em relação a usuários menos experientes ou usuários que utilizam o aplicativo pela primeira vez ou até mesmo para usuários que não estão familiarizados com a língua inglesa, pois todos os aplicativos encontrados para este fim, no *Google Play*, estão em língua inglesa. Estes aplicativos podem conter excesso de informações, e assim, deixando o usuário em dúvida do que pode ou não pode ser feito nele, enquanto outros apresentam uma interface gráfica poluída visualmente e desagradável aos olhos do usuário.

A utilização destes aplicativos tem sido uma boa alternativa para usuários que não querem se preocupar em levar um souvenir da cerveja do local em que eles estão bebendo. Eles preferem uma maneira simples de guardar essa lembrança e compartilhar com os amigos. Por isso, é preciso levar em conta os usuários que irão utilizar o aplicativo. Para usuários sem muita experiência, é necessário considerar a questão da interação que ocorrerá entre ele e o aplicativo, para que interface seja adequada levando o usuário a realizar a ação desejada de uma forma fácil e tranquila, neste caso, sendo o gerenciamento de sua coleção de cervejas.

Para Barbosa e Silva (2010), a interação do usuário com o sistema tem por finalidade alcançar um objetivo e para que isso aconteça é necessário que o sistema seja adequado aos usuários. Para os autores, as características de um sistema podem promover emoções nos usuários, sendo necessário evitar que tais emoções sejam desagradáveis. Para que isso possa ocorrer é necessário conhecer quais são as necessidades dos usuários e, assim, definir quais critérios devem ser priorizados no quesito usabilidade.

Silva Filho (2010) afirma que a usabilidade da interface de um produto compreende diversos fatores, como o grau de facilidade de conhecer, entender e usar o produto, assim como o comportamento e o bem-estar do usuário ao executar suas tarefas pelo sistema. Desse modo, oferecer usabilidade, para o autor, é desenvolver um sistema cujo uso seja simples e ágil. Com uma interface intuitiva o usuário consegue realizar suas tarefas de maneira mais rápida e eficiente.

Com a ideia de que um apreciador de cerveja gosta de manter uma coleção das bebidas já degustadas e levando em consideração o uso de *smartphones* para realizar tarefas do dia a dia, construiu-se um aplicativo que leva em consideração as necessidades do usuário para que ele possa ter, em mãos, a sua coleção digital de cervejas em um sistema fácil de usar. A próxima seção apresenta os passos para a construção desse aplicativo.

² <http://www.vivino.com>

3. Materiais e Métodos

O sistema desenvolvido, denominado *The Beer Point*, é capaz de gerenciar as cervejas degustadas pelo usuário e manter uma coleção pessoal. Ele oferece funcionalidades tais como: realizar *check-in* da cerveja que o usuário está tomando, tirar uma foto da mesma e adicionar o local onde ele está no momento com base no estabelecimento já cadastrado na API do *Foursquare*. O usuário poderá classificar a cerveja com uma nota de 1 a 5, sendo 5 a melhor nota e 1 a pior e também pode adicionar um comentário sobre a cerveja.

Para a criação dos protótipos das telas do sistema (Apêndice A) utilizou-se a ferramenta *Balsamiq Mockup*³ a qual é um software de prototipação que auxilia o desenvolvedor na criação de esboços para a interface gráfica de aplicativos móveis e aplicações *web*. Para os diagramas UML (*Unified Modeling Language* - Linguagem de Modelagem Unificada) utilizou-se a ferramenta *StarUML*⁴.

Para o desenvolvimento do aplicativo *Android* utilizou-se a ferramenta *Android Studio* que é o ambiente de desenvolvimento integrado oficial oferecido pelo *Google* para o desenvolvimento de aplicativos *Android*. Nele estão inclusas funções como edição inteligente de códigos, recursos para design de interface de usuário, análise de desempenho, integração com um sistema de controle de versão, entre outras funções.

A persistência de dados foi feita utilizando um serviço denominado de *BaaS* (*Backend as a Service* - Backend como Serviço). *BaaS* é uma arquitetura de computação que conecta aplicações móveis a serviços de armazenamento em nuvem no servidor, o que possibilita o acesso assíncrono a todos os recursos lá hospedados. Este serviço auxilia o desenvolvedor na hora de criar aplicativos móveis a vinculá-lo com um banco de dados na nuvem, deixando que o próprio serviço se encarregue de gerenciar o banco de dados.

Aplicativos móveis requerem várias funcionalidades, como por exemplo, integração com redes sociais, notificações *push* (mensagem ou alerta entregue por um servidor centralizado a um dispositivo final) e armazenamento em nuvem. Cada um destes serviços tem a sua própria API, a qual precisa ser incorporada individualmente no aplicativo, sendo que tal processo pode ser complicado e demorado. Para isso existem os serviços *BaaS*, uma ferramenta para fazer a integração entre o *front-end* e o *back-end* de um aplicativo utilizando uma única API.

Neste aplicativo foi utilizado como *BaaS* o serviço *Parse*⁵. *Parse* é um banco de dados orientado a objetos na nuvem. Ele conta com uma vasta documentação e vários serviços que podem ser integrados no aplicativo, como por exemplo: armazenar dados, enviar *e-mails* e notificações, fazer a integração com redes sociais e gerenciar usuários.

Com o desenvolvimento do aplicativo, surge a necessidade de testá-lo em um dispositivo real e para tal tarefa existem os emuladores. Junto com o SDK (*Software Development Kit* - Kit de Desenvolvimento de Software)⁶ do *Android* estão inclusas várias ferramentas que auxiliam no desenvolvimento da aplicação, entre elas um

³ <http://www.balsamiq.com>

⁴ <http://www.staruml.io/>

⁵ <http://www.parse.com>

⁶ <http://www.developer.android.com/sdk>

depurador, um emulador para simular um dispositivo real, a documentação e códigos de exemplos. O emulador que acompanha o SDK mostrou-se muito limitado em questão de desempenho e velocidade e por isso optou-se pelo uso do *Genymotion*⁷. *Genymotion* é um emulador de dispositivos com sistema operacional *Android* que opera em uma máquina virtual capaz de simular vários dispositivos reais com um desempenho muito superior ao emulador nativo do *Android*. O *Genymotion* existe na versão gratuita e na versão comercial, a diferença consiste na limitação de funcionalidades, como por exemplo, o acelerômetro que está presente apenas na versão comercial.

Para este projeto utilizou-se o *plugin* do *Genymotion* para o *Android Studio*, o que tornou a execução de testes muito mais eficiente e também a possibilidade de simular vários dispositivos reais com versões diferentes do *Android*.

O aplicativo foi desenvolvido seguindo um padrão de arquitetura de software que o próprio *Android* já adota: o MVC (*Model-View-Controller* - Modelo-Visão-Controller). Segundo Gamma (2000), “o *Model* é o objeto de aplicação, a *View* é a apresentação na tela e o *Controller* é o que define a maneira como a interface do usuário reage às entradas do mesmo”.

No aplicativo, a camada *View* é responsável pelo carregamento das telas, utilizando os layouts desenvolvidos na linguagem de marcação XML (*eXtensible Markup Language* - Linguagem Extensível de Marcação Genérica). A integração do código escrito em Java e o XML é feita por uma classe gerada dinamicamente pelo compilador. A classe funciona como uma espécie de interface entre o código Java e o código XML.

As *Activities* implementam os eventos e métodos para atender as telas de uma aplicação *Android*. Uma *Activity* roda em uma *Thread* gerenciada pelo sistema operacional e ela possui um ciclo de vida de acordo com a Figura 2:

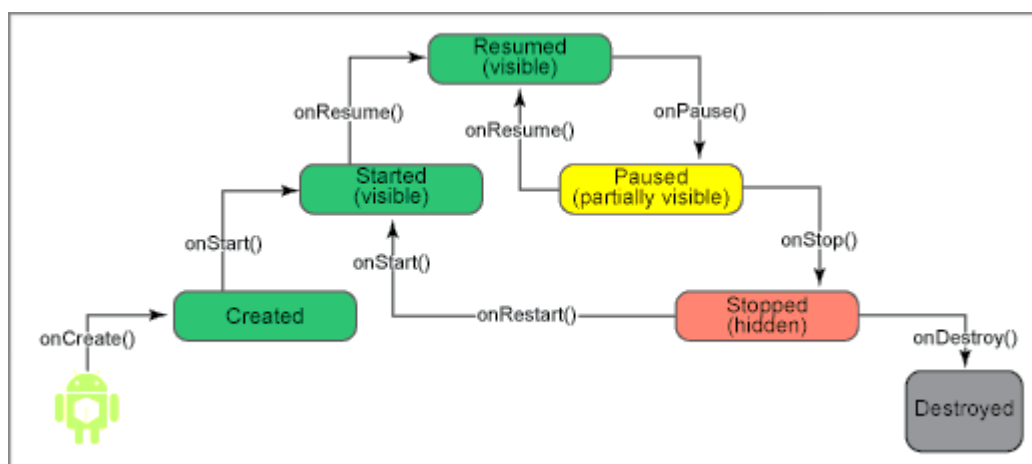


Figura 2. Ciclo de vida da *Activity* do *Android*. Fonte:
<http://www.ibm.com/developerworks/br/library/j-mobileforthemasses3/>

Os arquivos fonte de Java foram separados em quatro pacotes diferentes para melhorar a organização de classes, sendo eles: *activity*, *adapter*, *helpers* e *model*. No pacote *activity* estão as classes relacionadas ao funcionamento de uma tela do

⁷ <http://www.genymotion.com>

aplicativo. Já o pacote *adapter* contém as classes responsáveis por ler os dados armazenados no *Parse* e fornecê-los para a *View*. No pacote *helpers* estão as classes que são utilizadas para auxiliar no fornecimento de algumas funcionalidades do aplicativo. E, por fim, o pacote *model* onde estão todas as classes que modelam o banco de dados (Apêndice B) do aplicativo.

As funcionalidades do aplicativo foram definidas pelo próprio autor do projeto usando a abordagem da prototipação. A obtenção dos requisitos gerais do sistema foi por meio da análise de aplicativos semelhantes disponíveis no mercado. Exploraram-se os pontos fracos e os principais problemas que os usuários tinham com estes aplicativos.

Após ter definido os requisitos gerais do aplicativo, criou-se os protótipos das telas do sistema, seguindo as recomendações e boas práticas disponibilizadas pelo Google (2015), para o desenvolvimento de aplicações móveis, fez-se a modelagem do banco de dados usando a ferramenta *MySQL Workbench* e criou-se o diagrama de casos de uso para as funcionalidades presentes na tela inicial do aplicativo (Apêndice C).

Com os protótipos prontos, iniciou-se o desenvolvimento do aplicativo. Durante o desenvolvimento utilizou-se o *Framework JUnit*, que vem instalado com o *Android Studio*, para a realização de testes unitários. Desta forma, para cada nova função implementada no aplicativo, escrevia-se também o código para testar o funcionamento dela, garantindo, assim, que o método funcionasse como esperado.

3.1. Telas do sistema

As telas do aplicativo seguem um padrão em relação às cores, botões, tipografia e tamanho. Para seguir estas convenções, teve-se como base a nova linguagem visual criada pelo Google, o *Material design*⁸. Este novo conceito de *design*, segundo o Google (2015), tem como objetivo sintetizar os princípios clássicos de um bom *design* com a inovação, além de tentar criar uma experiência próxima das interações que existem com objetos do mundo “real”. Ainda de acordo com o Google (2015), o objetivo é desenvolver um único sistema que permite uma experiência unificada (usabilidade) em todas as plataformas e tamanhos de tela.

3.2. Funcionamento do sistema

O aplicativo inicia na tela de *login*, onde o usuário deverá entrar com seus dados para ter acesso ao conteúdo do sistema. Caso o usuário esteja acessando o sistema pela primeira vez, ele terá que realizar um cadastro informando os seus dados, inclusive a sua idade, pois o sistema não permite o acesso de usuários que não estão na idade legal para ingerir bebidas alcólicas. Após a realização do *login*, a *TBPTelaInicial* (Figura 3) é apresentada ao usuário. Nesta tela estão todas as postagens feitas pelo usuário e pelos amigos dele. São exibidas informações como, por exemplo, o nome da cerveja, o nome da cervejaria e o local onde está sendo degustada aquela cerveja. O usuário também poderá curtir a postagem clicando no ícone da caneca de cerveja.

O aplicativo possui três abas, sendo que a primeira é responsável por exibir todas as postagens feitas pelo usuário, a segunda aba, *TBPTelaListaCervejas* (Figura 4), é responsável por exibir uma lista com a coleção das cervejas que ele já degustou e fez uma avaliação.

⁸ <https://www.google.com/design/spec/material-design/introduction.html>

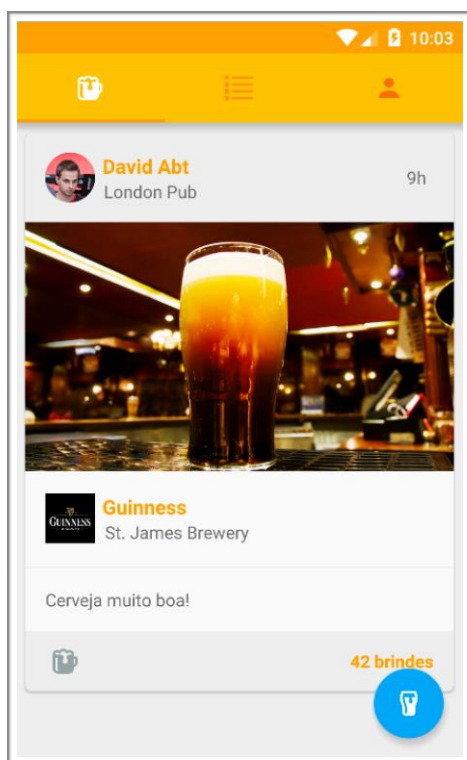


Figura 3. Exemplo da TBPTelaInicial de um usuário já cadastrado.

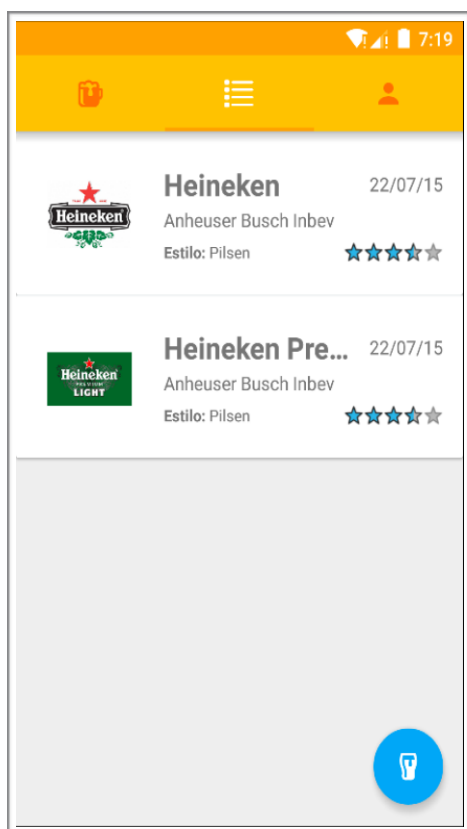


Figura 4. TBPTelaListaCervejas.

Em todas as telas está presente o botão para fazer o *check-in*. Este botão é definido pelo Google (2015) como sendo o Botão de Ação Flutuante e é usado para iniciar uma ação rapidamente.

Após clicar no botão de *check-in*, a TBPTelaCheckIn (Figura 5) será apresentada ao usuário. Nesta tela existe uma caixa de texto onde ele digita o nome da cerveja com a qual se queira fazer o *check-in*. Caso a cerveja esteja cadastrada no banco de dados do sistema, ela irá aparecer como um item nos resultados. Caso contrário, irá aparecer uma mensagem ao usuário informando-o que a cerveja não foi encontrada.

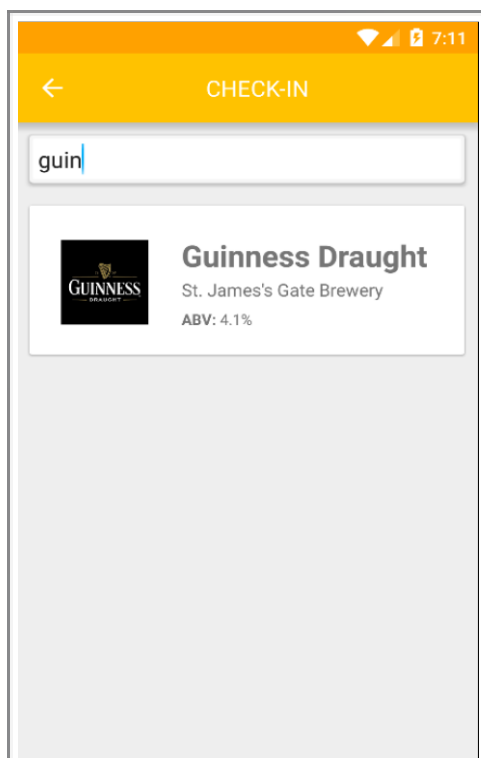


Figura 5. TBPTelaCheckIn.

Após selecionar a cerveja, o usuário será levado à TBPTelaAvaliacao (Figura 6). Nesta tela, o usuário poderá fazer a avaliação da cerveja, tendo a possibilidade de dar uma nota, inserir o local onde ele se encontra no momento, tendo como base as localidades já cadastradas no banco de dados do *Foursquare*, adicionar uma foto da cerveja e fazer um comentário sobre ela.

Por fim, quando o usuário concluir a ação de fazer o *check-in*, a lista de cervejas dele será atualizada e a postagem ficará visível na tela principal.

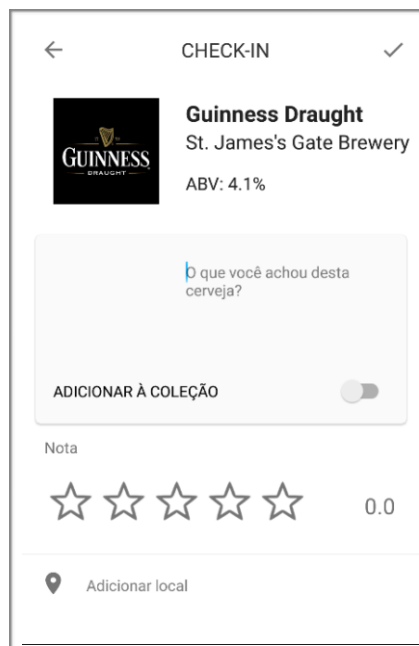


Figura 6. TBPTelaAvaliacao.

4. Resultados e discussão

Colecionar cervejas é um *hobby* de várias pessoas, principalmente cervejas artesanais e importadas. Mas muitas vezes a pessoa não pode levar junto uma garrafa para guardar de lembrança. Por isso, desenvolver um aplicativo para auxiliar estas pessoas mostrou-se oportuno, principalmente para o público e o mercado cervejeiro brasileiro.

A proposta deste trabalho foi o desenvolvimento de um aplicativo capaz de gerenciar uma coleção de cerveja de forma fácil, intuitiva e com capacidade de gerar informações confiáveis e precisas para estarem disponíveis sempre que o usuário precisar fazer uma consulta.

O aplicativo desenvolvido, denominado o *The Beer Point*, alcançou seus objetivos utilizando ferramentas de software livre para a criação de diagramas, protótipos das telas e modelagem do banco de dados. Para a criação da interface gráfica utilizou-se a documentação oficial do *Material design* oferecida pelo *Google*, a qual contém vários padrões e exemplos de como construir uma interface amigável e com boa usabilidade.

No entanto, cabe ressaltar que mesmo seguindo os padrões e as boas práticas propostos pelo *Google* (2015) para a criação de uma interface amigável, não há garantia de que o usuário venha a ter uma boa experiência ao usar o aplicativo sem antes aplicar um teste de usabilidade.

A linguagem padrão do aplicativo é a inglesa, porém optou-se por fazer a internacionalização, traduzindo todos os textos escritos, menus, erros, alertas, mensagens e opções para língua portuguesa. Nesse caso, quando o aplicativo é executado em um dispositivo com a língua padrão em português, todos os textos serão automaticamente traduzidos levando o usuário a ter uma experiência melhor e consequentemente abrangendo um público alvo maior. Fator este, em que o aplicativo *The Beer Point* se diferencia dos outros disponíveis no mercado.

Durante a realização do cadastro de novas marcas de cervejas na base de dados, encontrou-se dificuldades no fato do *Parse* não ter uma interface gráfica que permite o gerenciamento do banco de dados. Assim não é possível inserir, atualizar e excluir itens e criar relacionamento entre tabelas diretamente no *Parse*. Somente foi possível adicionar novas marcas por meio de código no próprio aplicativo.

5. Considerações finais

Essa pesquisa teve como objetivo geral desenvolver um aplicativo, voltado para o público cervejeiro, para ajudar a manter uma coleção digital das cervejas já degustadas.

Em vários estabelecimentos onde se comercializa cerveja, especialmente cerveja artesanal e importada, percebeu-se que o público que frequenta estes lugares costuma fazer uma avaliação sobre a bebida e até mesmo levar junto uma garrafa para adicionar a coleção que o degustador possa ter. Levando em conta este fato criou-se um aplicativo para ajudá-lo neste processo de avaliação.

Após analisar a opinião e os comentários dos usuários na página dos aplicativos correlatos encontrados no *Google Play*, notou-se a constante queixa de usuários brasileiros em relação à linguagem, pois todos os aplicativos encontrados eram de desenvolvedores estrangeiros e, portanto estão em língua inglesa. Considerando este fator, desenvolveu-se o aplicativo totalmente na língua portuguesa para atender este público.

Apesar de o *The Beer Point* cobrir as principais funções para quem tem o *hobby* de colecionar cervejas, como tirar foto, fazer uma avaliação e deixar um comentário sobre a bebida, futuramente pode-se complementar o aplicativo com um mapa de bebidas, ou seja, mostrar em qual estabelecimento comercial pode-se comprar quais tipos e marcas de cervejas, recomendações com base no gosto do usuário e também, onde se pode comprar uma dada cerveja.

Os objetivos desta pesquisa foram alcançados satisfatoriamente resultando em um aplicativo de fácil utilização, intuitivo e possibilitando o usuário a escolher entre a língua inglesa ou portuguesa como padrão.

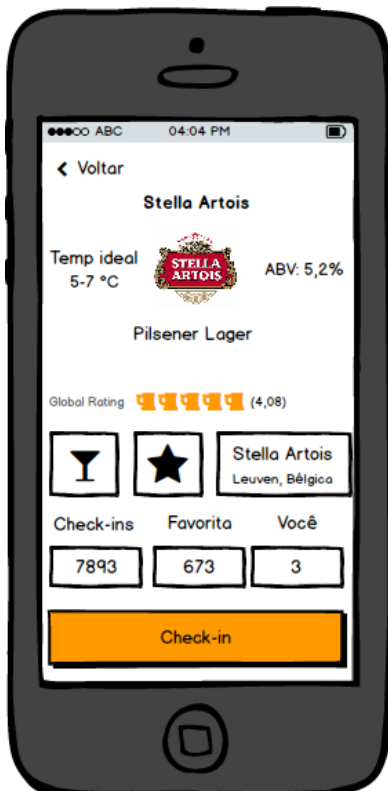
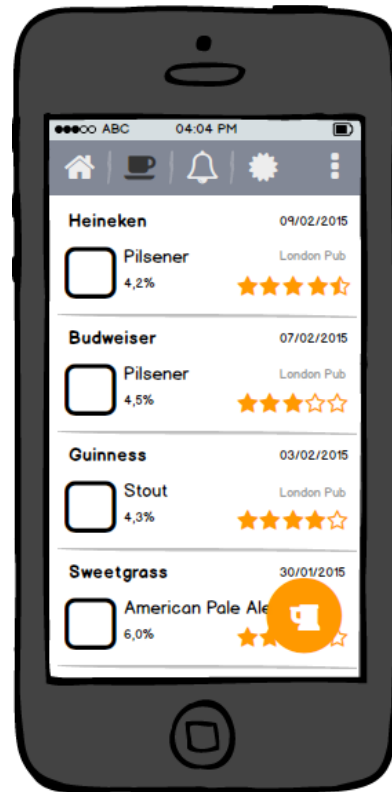
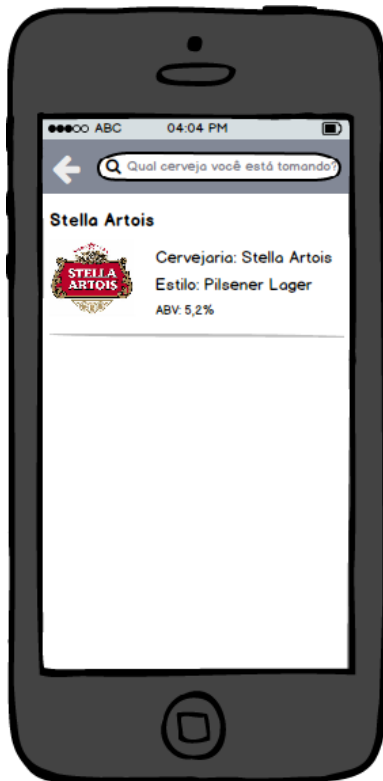
Para um projeto futuro pretende-se também portar o sistema para a *web*, fazendo uso da base de dados já existente no *Parse*, utilizando para este vínculo APIs REST. O objetivo é conquistar um público alvo maior e facilitar o gerenciamento do banco de dados de cervejas por meio de um sistema eficiente e fácil de atualizar.

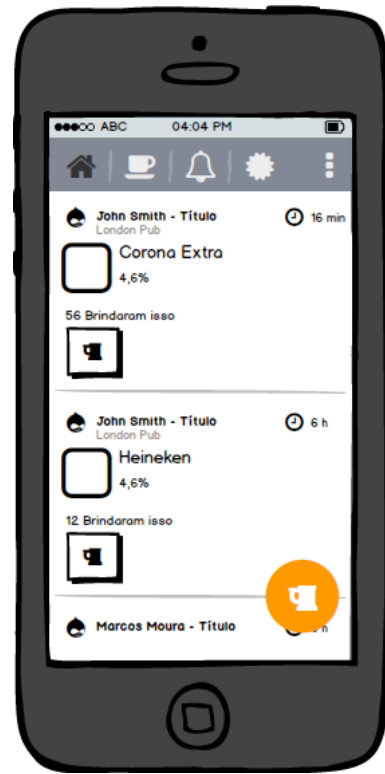
6. Referências

- Abrabe. *Associação Brasileira de Bebidas*. Disponível em: <<https://www.abrabe.org.br/categorias/>>. Acesso em: 26 jun. 2015.
- Barbosa, S. D. J.; Silva, B. S. da. *Interação Humano-Computador*. Rio de Janeiro, Editora Campus-Elsevier, 2010.
- Cervesia. *O mercado cervejeiro brasileiro atual - potencial de crescimento*. 2011. Disponível em: <<http://www.cervesia.com.br/dados-estatisticos/609-o-mercado-cervejeiro-brasileiro-atual-potencial-de-crescimento.html>>. Acesso em: 10 jul. 2015
- Gamma, Erich et al. *Padrões de Projeto: Soluções reutilizáveis de software Orientado a Objetos*. Porto Alegre: Bookman, 2000.

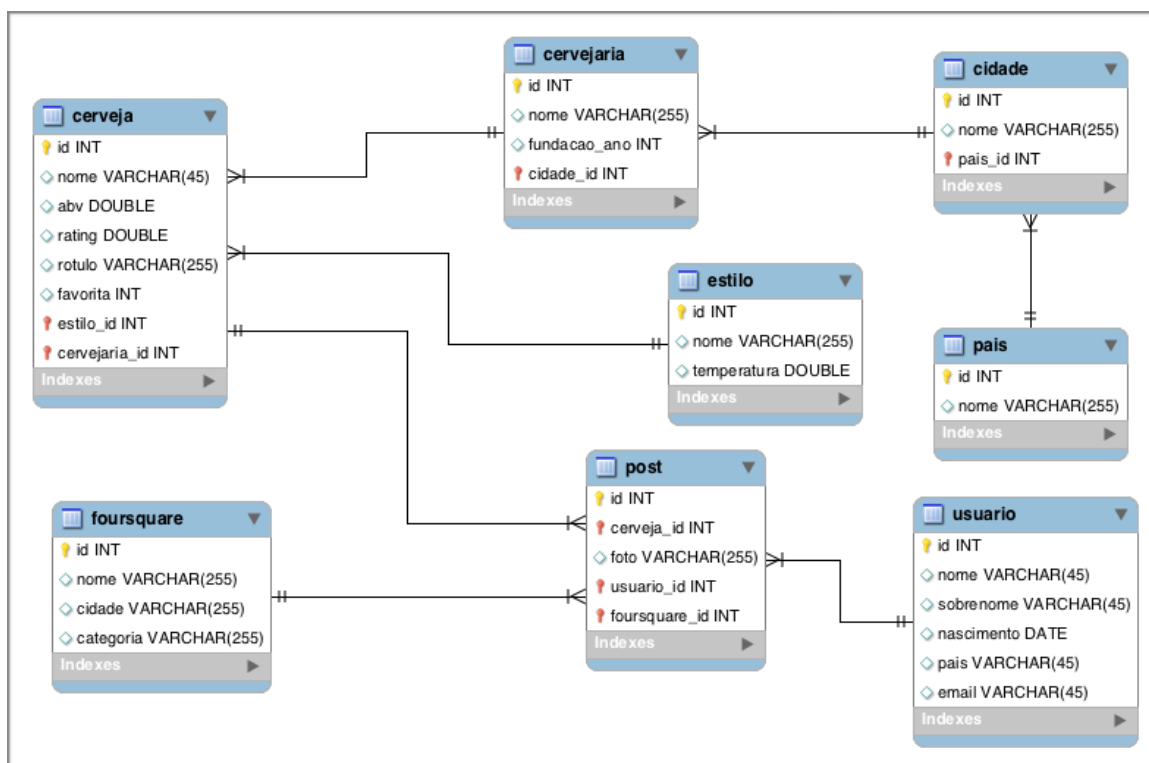
- Google. *Nosso Planeta Mobile: Brasil*. 2013. Disponível em: <<http://services.google.com/fh/files/misc/omp-2013-br-local.pdf>>. Acesso em: 4 ago. 2015.
- Google. *Material Design*. 2015. Disponível em: <<https://www.google.com/design/spec/material-design/introduction.html>>. Acesso em: 18 ago. 2015.
- Guimarães, Saulo Pereira. *Brasil é o quarto país do mundo em número de smartphones*. 2013. Disponível em: <<http://exame.abril.com.br/tecnologia/noticias/brasil-e-o-quarto-pais-do-mundo-em-numero-de-smartphones>>. Acesso em: 17 mar. 2015.
- Kovacs, B. P. U. *Um estudo prático das ameaças de segurança em dispositivos portáteis com Windows Mobile*. Monografia. Departamento de Informática – PUC-Rio. 2006.
- Lecheta, Ricardo R. *Google Android: Aprenda a criar aplicações para dispositivos móveis com o Android SDK*. 2. Ed. São Paulo: Novatec, p.23, 2010.
- Lee, Wei-Meng. *Introdução ao Desenvolvimento de Aplicativos para Android*. 1. ed. Rio de Janeiro: Editora Ciência Moderna Ltda, 2011.
- Mantovani, Camila e Dantas Geórgia. *Os fluxos informacionais nos dispositivos móveis*. UFMG. 2010.
- Mosher, Randy. *Tasting Beer: An Insider's Guide to the World's Greatest Drink*. North Adams: Storey Publishing, 2009.
- Netsharemarket. *Mobile/Tablet Top Operating System Share Trend*. 2015. Disponível em: <<http://www.netmarketshare.com/operating-system-market-share.aspx?qprid=9&qpcustomb=1&qpct=4&qpsp=187&qpnp=13&qptimeframe=M>>. Acesso em: 16 ago. 2015.
- Silva Filho, A. M. da. *Avaliação de Usabilidade: "Separando o joio do trigo"*. Revista Espaço Acadêmico, v. 10, n. 112. p. 10-14, set. 2010. ISSN 1519-6186. Disponível em: <<http://eduem.uem.br/ojs/index.php/EspacoAcademico/article/view/11077/5962>>. Acesso em: 16 ago. 2015.
- Tierney-Jones, Adrian. *1001 Cervejas Para Beber Antes de Morrer*. Rio de Janeiro: Sextante, 2012. 957 p.

APÊNDICE A – Wireframe das telas.





APÊNDICE B – Modelagem do Banco de Dados.



APÊNDICE C - Diagrama de Casos de Uso. Funcionalidades presentes na tela inicial do sistema.

