

# Relatório - PSG2

Bruno Yoshikazu Shimada

Novembro 2017

## Resumo

Vivemos em uma época que a cada dia, novas soluções digitais são lançadas para resolver nossos problemas cotidianos de uma maneira simplificada que se incorporam ao nosso cotidiano de uma maneira que após algum tempo não conseguimos imaginar como nós conseguimos viver sem isso por tanto tempo.

Entre as soluções digitais podemos citar por exemplo o ramo econômico e bancário, com desde os básicos aplicativos de bancos para consulta a extratos, pagamentos de boletos, etc, até soluções mais sofisticadas como aplicativos para receber e processar pagamentos por cartão.

Porém um problema corriqueiro que temos no dia-a-dia é o pagamento de pequenas transações monetárias, como tomar um café na padaria, comprar um chiclete na bomboniere, emprestar uma pequena quantia de dinheiro para um conhecido, entre outras que na maioria das vezes não passam de 10R\$ e que geram uma perda de tempo de ter que realizar o pagamento com um cartão, digitando a senha pessoal em uma máquina e torcendo para o serviço da operadora do cartão estar disponível no momento.

Para resolver esse problema entram em cena os serviços de micropagamento, que buscam solucionar o problema de fazer um pagamento ou uma transação de pequeno valor entre duas pessoas, independente de ser pessoa-a-pessoa ou pessoa-a-negócio, de forma rápida, eficaz, segura e descomplicada. O propósito deste trabalho é então desenvolver um aplicativo Android, SO mais comum entre os usuários de celular no Brasil, que gerencie as transações entre usuários do aplicativo e que faça valer as características descritas anteriormente.

### Palavras Chaves

- Micropagamentos
- *Android*
- *Fintech*

## **Abstract**

We live in time that everyday new digital solutions are released to solve our daily problems in a simplified manner, that merges into our days that after some time we are unable to imagine how we lived so much time without it.

Among these digital solutions we can mention for example the economic/banking areas, that has the basic bank app for checking account information, payment of tickets, etc.. To more sophisticated apps that receive and process payments by card.

However a common problem we have in our daily lives is the payment of small monetary transactions, such as having a coffee at a bakery, buying gum from a local store, lending money to an acquaintance, among others that in most times dont surpass 10 BRL and generate a loss of time for having to make the payment with a card, typing our personal password on a machine and hoping the service of the card operator will be available at the moment.

To solve this problem the micropayments services come into play, seeking to solve the problem of making the payment or transaction of a small amount of money between two persons, regardless of being a person-to-person or person-to-business, in a quick, efficient, safe and uncomplicated way. The purpose of this work is to develop an Android app, the most common OS among mobile users in Brazil, that manage transaction between users of the app and enforce the characteristic described before.

## **Keywords**

- Micropayments
- Android
- Fintech

## Sumário

1	Introdução . . . . .	5
2	Objetivos . . . . .	7
2.1	Objetivo Geral . . . . .	7
2.2	Objetivos Específicos . . . . .	7
3	Revisão Bibliográfica . . . . .	8
4	Metodologia . . . . .	11

# 1 Introdução

A definição de micropagamento é relativamente simples, se usado uma associação das palavras que a compõem, se infere que se trata de transações cujo valor é uma quantia muito pequena. Trazendo para a realidade dele, um micropagamento é uma transação online de uma quantia muito pequena de dinheiro (INVESTOPEDIA). Quão pequeno esse valor é, varia entre as diferentes empresas no mercado, o *PayPal*, *e-commerce* que faz a transação de valores digitais entre usuários nas duas pontas, considera um micropagamento qualquer transação cujo valor seja menor do que 10USD (PAYPAL).

A evolução do mercado de micropagamentos é creditada a 3 fatores (*HERNANDEZ - VERME, BENAVIDES; 2013*) definidos com base em um relatório de *VASILJEV (2016)* e *BURELLI (2016)*, definem eles como:

- O crescimento da infraestrutura de rede e do e-commerce em geral
- O crescimento das redes sociais, jogos online e negócios de bens digitais
- O aparecimento de novas formas de serviços de pagamento online

Com base nisso podemos ver que as soluções atuais para micropagamento surgiram principalmente da necessidade de pagamento de bens para consumo digital como assinaturas de sites, compras de músicas digitais, o melhor exemplo aqui seria o *iTunes* por exemplo. Dentre as soluções que existem atualmente, a *Investopedia* dá foco em dois modelos:

- A plataforma agindo como carteira digital. Cada usuário cria sua conta na plataforma que irá gerenciar a transação entre dois usuários, a transação é efetuada e a plataforma se encarrega de armazenar esse valor, quando a carteira de qualquer uma dessas partes atinge um limite, a quantia total do dinheiro é liberada e transferida para o usuário.
- A plataforma agindo com créditos. Cada usuário cria sua conta na plataforma, cada um deles compra/recarrega a quantia desejada que deseja ter como crédito, a cada transação entre duas partes a plataforma se encarrega de atualizar os créditos de ambas as partes.

O grande problema na área de micropagamentos são as taxas cobradas pelas operadoras de cartão de crédito, meio que a maioria dos modelos existentes como demonstrado acima usa como forma preferencial de pagamento o cartão

crédito, porém é sabido que para cada transação são cobradas taxas em cima do valor pago, é um caso raro mas podem existir situações que as taxas podem, porque não, superar o valor da transação, o que para um cliente que seja dono de um negócio, torna-se algo insustentável.

## 2 Objetivos

### 2.1 Objetivo Geral

O objetivo do trabalho foi desenvolver um aplicativo *Android* que conseguisse efetuar transações monetárias entre duas pessoas usando o celular, com uma interface simples de usar.

### 2.2 Objetivos Específicos

Os objetivos específicos do trabalho foram:

- Aprendizagem em programação para *Android*
- Desenvolver a *UI* do aplicativo com base em uma plataforma de serviços e infraestrutura existente
- Aprendizagem do conceito de micropagamentos

### 3 Revisão Bibliográfica

Ao longo do desenvolvimento do aplicativo, em diversos momentos apareceram dúvidas que não conseguiam ser resolvidas por força bruta, como por exemplo, como definir a posição fixa de um botão em relação a outros no mesmo layout? Dúvidas desde as mais simples, que surgem principalmente nos momentos iniciais de aprendizado de alguma linguagem ou *framework* novo, até as mais complexas, quando já se tem uma base sólida de conhecimento sobre o tópico, são comuns no meio da tecnologia, fato que pode ser comprovado fazendo uma simples busca no *Google*, nas páginas de resultado é comum encontrar pessoas com a mesma dúvida, em variados fóruns, blogs e sites diversos. O mais interessante nesse ponto é a também a variedade de abordagens diferentes que são possíveis de encontrar para uma mesma dúvida, fazendo com que tenhamos que filtrar dentre elas a que melhor se aplica ao contexto do nosso problema.

Abaixo estão listadas as fontes consultadas ao longo do trabalho que ajudaram em diversos momentos do desenvolvimento.

Para entendimento do conceito de micropagamentos foram usadas duas fontes principais:

- A página do termo no site da *Investopedia* que deu uma visão geral sobre o assunto, enfocando primeiro em um resumo simplificado para depois se aprofundar um pouco mais nele. O interessante desse site é que a partir do termo "*micropayment*", ele busca termos relacionados buscando fazer uma trilha de informações para abranger o assunto.
- A segunda fonte foi o artigo "***Virtual currencies, micropayments and the payments systems: a challenge to fiat money and monetary policy?***" Que discorre não só sobre o conceito de micropagamentos como também aborda o tema das moedas virtuais e relaciona ambos. O artigo foca mais na parte de como os micropagamentos são mais relevantes no mundo online na compra de itens que eles classificam como microprodutos, fazem uma comparação das transações online que envolvem produtos físicos e digitais. Esse artigo considero como o principal usado já que ele aborda de uma maneira mais aprofundada o ambiente que os micropagamentos se inserem.

No começo do projeto meu conhecimento de programação para *Android*



era nulo, durante a preparação da máquina encontrei no mesmo site que disponibiliza a *IDE* para desenvolvimento, uma sub-seção intitulada *Training*, debaixo da seção *Develop*. Nela encontrei uma série de lições básicas para quem está começando a desenvolver para *Android*. Foi útil para ter uma noção básica de onde começar e os termos específicos usados.

Após a criação de um aplicativo *Hello World* básico seguindo o guia do site da *Android*, fui atrás de mais algum material para criar um aplicativo e aprofundar um pouco mais o conhecimento na plataforma. O livro utilizado nesse momento foi "***Android 5 Programming by Example***", o livro se aprofunda um pouco mais nos exercícios ajudando e servindo de guia para criar um aplicativo mais robusto do que o visto no guia do site da *Android*. O livro foi usado de guia no segundo passo do projeto onde criei um aplicativo *mock* que simulava localmente uma transação nos moldes de um micropagamento de um aparelho para um servidor, para testar os conhecimentos e novas técnicas adquiridas. Apesar do livro focar mais na versão *Lollipop*, lançada em 2015 mesmo ano da publicação do livro, grande parte dos componentes não mudou muito para a versão mais atual.

Durante a elaboração do rascunho das interfaces do aplicativo uma página que foi muita usada como fonte de consulta e guia é a *Material Design*. Nela foi possível encontrar orientações e guias de como desenhar a interface para se encaixar no padrão do *Material Design*. Dentre as ferramentas e guias disponíveis os que se destacam pelo seu uso e importância no projeto foram as seguintes:

- ***Material Icons*** - Uma espécie de repositório da *Google* com vários ícones disponíveis para uso. Todos são liberados para uso sobre a licença *Apache License Version 2.0*. Os ícones são desenhados de maneira simples e de fácil interpretação, o design deles é de uma maneira que quando se olha para o ícone é fácil deduzir qual seu significado. Por exemplo o ícone que foi usado para simbolizar a ação de pagamento:



Figura 1: Ícone usado para simbolizar a ação de pagamento

Além da página com o repósitorio dos ícones, o design e o conceito dos itens é explicado no artigo ***Style - Icons***

- ***Material Colors*** - Uma ferramenta que ajuda na escolha da paleta de cores que vão compor o aplicativo, ele oferece também um guia visual de como ficam as cores nas interfaces do sistema. Outro artigo relacionado é ***Style - Colors*** onde é explicado as boas práticas e guias para selecionar as cores dos componentes da interface do aplicativo.

## 4 Metodologia

Desenvolver a interface de um aplicativo *Android* com base numa plataforma de serviços existentes buscando fazer a mesma ser simples, no sentido de ter uma fácil utilização, buscando obedecer e seguir os guias definidos pelo *Android* como boas práticas e que atendem e se adequam a filosofia do *Material Design*.

As linguagens utilizadas no projeto foram:

- ***XML***: todas as interfaces do *Android* são escritas usando a linguagem de marcação *XML*, o *Android* usa um conjunto de elementos e atributos próprios na marcação como por exemplo o elemento `<RelativeLayout>` e o atributo `android:layout_width`. *XML* também usado para definir *Strings*, o tema base, e qualquer outro elemento visual que possa ser reaproveitado, como por exemplo um *layout* customizado para um botão. Também é usado para definir o manifesto do aplicativo, onde são declaradas configurações específicas do aplicativo, como por exemplo permissão para uso de *bluetooth*, *internet*, etc.
- ***Java***: A parte do *back-end* que controla a aplicação é toda codificada em *Java*. O *Android* usa *Java* para por exemplo codificar um *listener* que capta quando o usuário clica em algum botão por exemplo, para fazer a troca de interfaces, controle de sessão, carregar *Strings* dinâmicas na interface, tratamento de erros e exceções, etc.

Quanto a *hardware* foram usados um *notebook* próprio rodando um SO *Debian*, e dois aparelhos celulares próprios com SO *Android* para simular o comportamento de dois usuários. Devido a diferença de SO de ambos, foi interessante observar como a interface se comportava em cada um deles. A título de curiosidade seguem as especificações:

- *Notebook*
  - SO: *Ubuntu 16.04 LTS - 64-bit*
  - Memória RAM: 8GB
  - Processador: *Intel®Core™ i7-5500U CPU @ 2.40GHz x 4*
  - Processador gráfico: *Intel®HD Graphics 5500 (Broadwell GT2)*
- Celular *Motorola Moto G 3ª Geração*

- SO: *Android 5.1 Lollipop*
  - Memória RAM: 1GB
  - Processador: *Qualcomm Snapdragon 410*
- Celular *Samsung S3 Mini*
  - SO: *Android 4.1 Jellybean*
  - Memória RAM: 1GB
  - Processador: *Dual-core, 1000 MHz*