

INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DE SÃO PAULO  
CAMPUS SÃO PAULO  
TECNOLOGIA EM ANÁLISE E DESENVOLVIMENTO DE SISTEMAS

Alkindar José Ferraz Rodrigues  
Carolina de Moraes Josephik  
Fabio Mendes Torres  
Gabriely de Jesus Santos Bicigo  
Leonardo Naoki Narita

**Lixt**

São Paulo

2021

Alkindar José Ferraz Rodrigues  
Carolina de Moraes Josephik  
Fabio Mendes Torres  
Gabriely de Jesus Santos Bicigo  
Leonardo Naoki Narita

## **Lixt**

Proposta inicial de projeto apresentada na  
disciplina de Projeto Integrado I no 1º semes-  
tre de 2021.  
Prof. Ivan Francolin Martinez  
Prof. José Braz de Araujo

Instituto Federal de Educação, Ciência e Tecnologia de São Paulo  
Campus São Paulo  
Tecnologia em Análise e Desenvolvimento de Sistemas

São Paulo  
2021

# 1 Introdução

## 1.1 Descrição do Problema

Lorem ipsum dolor sia temet ([KANAN; FOX, 2015](#), p. 3).

## 1.2 Objetivos

O objetivo deste projeto é desenvolver um aplicativo multiplataforma, focado no gerenciamento de listas de compras compartilhadas e que apresente análises quanto à variação de preço para usuários brasileiros. Para tal, é necessário que o aplicativo ofereça uma interface intuitiva e responsiva, e que os sistemas back-end sejam estáveis e rápidos.

Desta forma, uma arquitetura adequada seria uma aplicação em três camadas, com um aplicativo front-end capaz de ser acessado tanto nos dispositivos Android quanto iOS, e, eventualmente, web; que a api disponível ao aplicativo seja bem estruturada conforme os padrões REST; que os dados dos usuários sejam armazenados de maneira segura e criptografada; e que uma base de dados volumosa seja fornecida de antemão para a comodidade dos usuários quanto aos itens de mercado mais comuns no país.

## 1.3 Solução Proposta

Tendo em vista o problema exposto e as limitações que serão levantadas acerca dos concorrentes, propomos como solução o Aplicativo *Lixt*, que apresentará as seguintes funcionalidades, expostas em ordem de dependências e nível de complexidade:

- Login

O usuário deverá fazer login, de forma segura e privativa, para que possam acessar as funcionalidades que se seguem. Inicialmente, foi pensada uma solução na qual o login poderia ser opcional para as funcionalidades mais básicas, entretanto, isso impõe uma dificuldade na formulação de duas lógicas distintas no front-end.

- Construir e gerenciar listas de compras

O usuário deverá ser capaz de criar e destruir listas de compras, adicionar ou remover itens de uma lista, especificar quantos itens devem ser comprados de cada produto e marcar itens como comprados. Esta funcionalidade é considerada a mais básica e essencial para a entrega do MVP.

- Gerenciar categorias  
O usuário poderá atribuir categorias aos produtos adicionados, adicionar e deletar categorias, e filtrar itens de uma lista com base nelas.
- Compartilhamento De Listas  
O usuário terá a possibilidade de compartilhar listas com outros usuários logados, e mudanças realizadas por um usuário devem estar disponíveis para os outros usuários com acesso a lista.
- Atribuição de itens para usuários em listas compartilhadas  
O usuário poderá atribuir, em listas compartilhadas, itens a pessoas com acesso a esta lista, como forma de direcionar quem deve comprar cada item.
- Comentários em listas compartilhadas  
O usuário poderá adicionar, em uma lista que tenha acesso, comentários associados a itens específicos com informações que considerar relevantes àquele produto, e estas notas devem estar disponíveis aos demais usuários que tenham acesso aquela lista
- Gerenciamento de compras  
O usuário poderá realizar compras, vinculadas a uma lista e a um mercado, inserindo o preço pago por cada produto e, possivelmente, um desconto associado ao item. Quando o processo de compra for iniciado, o usuário verá o valor atualizado a ser pago no carrinho conforme os itens forem selecionados como “pegos”.
- Histórico de compras  
As compras realizadas deverão ser salvas e apresentadas conforme solicitadas pelo usuário, e apresentar o valor total em destaque, assim como a variação em relação as compras anteriores no mesmo mercado e com a mesma lista. Também, o preço de cada item deve apresentar a variação em relação ao valor anterior e posterior.
- Análise estatística de compras  
Será apresentado ao usuário uma análise estatística dos itens comprados e dos valores pagos, em diversos níveis de especificidade. Planeja-se para a versão final do aplicativo que o usuário possa visualizar as seguintes variações:
  - a variação de preço das diversas compras, levando em consideração ou não os mercados;
  - quantas unidades de um item foi comprado por vez;
  - a variação de preço do item a cada compra, levando os mercados em consideração;
  - a variação de da média de preços de uma categoria, considerando ou não os mercados;
  - a variação da média de preços de uma marca, considerando ou não os mercados.

## 1.4 Escopo do Projeto

## 2 Análise de Concorrentes

Auditamos soluções que existem atualmente no mercado e, ao verificar as aplicações existentes, conclui-se que há intersecções nas funções dentre os aplicativos analisados. As funções mais básicas, como gerenciamento de itens e gerenciamento de listas, estão presentes em todos, tendo em vista que são essenciais em qualquer aplicativo de lista. Outras funções básicas que deveriam ser incluídas em qualquer aplicação de lista, como gerenciamento de categorias e compartilhamento de listas, não estão presentes em todos os aplicativos analisados. Contudo, as divergências ficam claras quando analisamos o mecanismo das aplicações, entre elas destacam-se o Mealime e o Cozi Family Organizer que, apesar de serem voltados para as compras, cumprem também outras funcionalidades. O Mealime, cujo foco é o planejamento de refeições, e o Cozi Family Organizer, cujo foco é o planejamento familiar, deixam a desejar nas funções relacionadas às compras. Entre os outros aplicativos analisados, é perceptível que não possuem todas as funcionalidades propostas nesse documento, principalmente quando se trata de compartilhamento de listas, uma vez que cada software lida de modo diferente diante dessa feature. O SoftList, por exemplo, permite o compartilhamento de lista, porém não é capaz de ser gerenciada por mais de um usuário, sendo apenas importada para o usuário no qual a lista está sendo compartilhada. Ao analisar os aplicativos mais populares da categoria, constatamos que o Out Of Milk, Bring! e o OurGroceries, que são destaques na área, não se propõem a exibir análise estatística das compras do usuário e nem manter um histórico do que foi comprado.

## 2.1 Tabela de Comparação

	Cozi Family Organizer	OurGroceries	SoftList	Out of Milk	Mealime	Bring!
Login/Cadastro	X	X	X	X	X	X
Gerenciamento de Categorias		X	X	X		X
Compartilhamento de Listas	X		X	X		X
Atribuição de itens para usuários em listas compartilhadas						
Gerenciamento de Compras			X			
Histórico de Compras			X			
Análise de Compras (Estatísticas)			X			
Calculadora			X	X		X
Comentários em Listas Compartilhadas						

Figura 1 – Comparativo de Concorrentes

### 3 Tecnologias Aplicadas

Com base no escopo que foi definido inicialmente para o aplicativo Lixt optamos pela utilização de diferentes tecnologias que apresentam vantagens durante o desenvolvimento.

Para o *front-end* da aplicação escolhemos utilizar *JavaScript* com a biblioteca *React*, pois a curva de aprendizado dessa tecnologia é curta e seu código é reutilizável, podendo ser inclusive utilizado em aplicações mobile com o *React Native* (para *Android* e *iOS*).

Para o banco de dados foi definida a utilização do *MySQL* por possuir uma grande variedade de funções e configurações que facilitam o desenvolvimento.

Também no *back-end*, vamos utilizar a linguagem *Java* para o código-fonte, com o *framework Spring*, em um projeto *Spring Boot*. O *Spring Boot* pois traz mais produtividade durante o desenvolvimento, permitindo concentrar os esforços na implementação das regras de negócio do que com as configurações de um projeto *Web* (AFONSO, 2017). Ainda nesse sentido, vamos utilizar o *Hibernate*, que possibilita um desenvolvimento mais ágil nas aplicações que possuem integração com bancos de dados, como é o caso do aplicativo Lixt.



## 4 Gerenciamento do Projeto

## Referências

AFONSO, A. O que é spring boot? *Alga Works*, 2017. Citado na página [7](#).

KANAN, T.; FOX, E. A. Automated arabic text classification with p-stemmer, machine learning, and a tailored news article taxonomy. *Virginia Tech*, 2015. Citado na página [2](#).