



CENTRO UNIVERSITÁRIO LUTERANO DE PALMAS

Recredenciado pela Portaria Ministerial nº 1.162, de 13/10/16, D.O.U. nº 198, de 14/10/2016
AELBRA EDUCAÇÃO SUPERIOR - GRADUAÇÃO E PÓS-GRADUAÇÃO S.A.

Athos Vinicius Correia Soares

APLICATIVO MOBILE PARA CÁLCULO DE ANÁLISES FÍSICO-QUÍMICAS DE ALIMENTOS

Palmas – TO

2021

Athos Vinicius Correia Soares

**APLICATIVO MOBILE PARA CÁLCULO DE ANÁLISES FÍSICO-QUÍMICAS DE
ALIMENTOS**

Projeto de Pesquisa elaborado e apresentado como requisito parcial para aprovação na disciplina de Trabalho de Conclusão de Curso I (TCC I) do curso de bacharel em Sistemas de Informação pelo Centro Universitário Luterano de Palmas (CEULP/ULBRA).

Orientador: Prof. Esp. Fábio Castro Araújo.

Palmas – TO

2021

Athos Vinicius Correia Soares

**APLICATIVO MOBILE PARA CÁLCULO DE ANÁLISES FÍSICO-QUÍMICAS DE
ALIMENTOS**

Projeto de Pesquisa elaborado e apresentado como requisito parcial para aprovação na disciplina de Trabalho de Conclusão de Curso I (TCC I) do curso de bacharel em Sistemas de Informação pelo Centro Universitário Luterano de Palmas (CEULP/ULBRA).

Orientador: Prof. Esp. Fábio Castro Araújo.

Aprovado em: ____/____/____

BANCA EXAMINADORA

Prof. Esp. Fábio Castro Araújo.

Orientador

Centro Universitário Luterano de Palmas – CEULP

Prof M.e Jackson Gomes de Souza.

Centro Universitário Luterano de Palmas – CEULP

Prof. Esp. Fernanda Pereira Gomes.

Centro Universitário Luterano de Palmas – CEULP

Palmas – TO

2021

Aos meus pais, primos, professores, amigos e colegas de curso e a todos aqueles que de certa forma contribuíram para a realização deste trabalho.

AGRADECIMENTOS

A Deus, pelo amor e pela misericórdia derramada sobre a minha vida, bem como por iluminar a minha mente nos momentos difíceis, dando-me forças e coragem para seguir.

Agradeço aos meus pais, Euclides e Marlúcia, que, com humildade e honestidade, fizeram-me melhor. A vocês, todo o meu amor e a minha gratidão.

Ao meu orientador, professor Esp. Fábio Castro, que me auxiliou e esteve presente sempre que necessitei, contribuindo com o desenvolvimento do trabalho e ajudando-me a acreditar na minha ideia.

Ao meu primo, Lucas Meira, pelas oportunidades de aprendizagem, ensinamentos e troca de experiências.

Aos meus tios, Antônio e Norbelina, que me deram inicialmente a oportunidade de estudar e morar em sua casa.

A minha tia, Irenilde, que me ofereceu inicialmente a oportunidade de estudar sendo minha fiadora.

Aos meus tios de consideração, Aclessivaldo e Claudineia, que me acolheram e deram-me a oportunidade de continuar estudando pelo financiamento da faculdade.

Agradeço também, a todos os professores pelos ensinamentos e valores passados a mim durante o curso.

A todos os meus amigos, com quem divido todas as minhas alegrias e angústias, especialmente ao Cleiton, Cryslei, Otávio, Leandro, Marcos Muniz, Kamilla e Fabiano, a qual me ajudaram durante toda essa trajetória universitária.

Enfim, a todos que fizeram parte desta grande etapa da minha vida.

RESUMO

SOARES, Athos V. C. **Aplicativo mobile para cálculo de análises físico-químicas de alimentos**. 2021. Trabalho de Conclusão de Curso (Graduação) – Curso de Sistemas de informação, Centro Universitário Luterano de Palmas, Palmas/TO, 2021.

Nos cálculos de análises físico-químicas de alimentos é que são descobertas as propriedades dos alimentos. As propriedades são utilizadas em um processo de cálculo que é feito de forma manual, sendo possível a análise posterior dos resultados físico-químicos do alimento. O presente trabalho propõe uma aplicação *mobile* para realizar cálculos de análises físico-químicas de alimentos de forma sistematizada. Assim, foi desenvolvida uma aplicação *mobile* que permite a inserção de dados das amostras, de forma a facilitar posteriormente que os profissionais possam consultar o cálculo já realizado. Sendo possível aos profissionais que atuam nesse segmento acompanhar os resultados que foram gerados de forma sistematizada. Para a elaboração do presente trabalho serão utilizadas as seguintes ferramentas: *Firebase* para o desenvolvimento do banco de dados, *Dart* que é a linguagem de desenvolvimento e *Flutter framework* para o desenvolvimento do *software*. Como resultado, a implementação de um aplicativo *mobile* com a automatização de cálculos para análises físico-químicas de perda por dessecação, resíduo por incineração e lipídios ou extrato etéreo.

Palavras chave: Análise, Físico-Químico, Firebase, Flutter, Dart.

LISTA DE FIGURAS

Figura 1 - Estufa convencional para secagem	14
Figura 2 - Forno de aquecimento mufla	15
Figura 3 - Arquitetura clean architecture	18
Figura 4 - Fluxo do trabalho	22
Figura 5 - Estrutura clean architecture	23
Figura 6 - Estrutura da camada de <i>presentation</i>	23
Figura 7 - Estrutura da camada de <i>domain</i>	24
Figura 8 - Estrutura da camada de <i>data</i>	24
Figura 9 - Estrutura das pastas <i>datasources</i> e <i>models</i>	25
Figura 10 - Importar biblioteca do Firebase	26
Figura 11 - Funcionalidade para armazenar um cálculo	27
Figura 12 - Representação do cálculo de perda por dessecação	28
Figura 13 - Funções que realizam o cálculo de perda por dessecação	29
Figura 14 - Cálculo de resíduo por incineração	30
Figura 15 - Funções que realizam o cálculo de resíduo por incineração	31
Figura 16 - Cálculo de lipídios ou extrato etéreo	32
Figura 17 - Funções que realizam o cálculo de lipídios ou extrato etéreo	33
Figura 18 - Tela de login e homepage	34
Figura 19 - Tela de cálculos de incineração e lipídios	35
Figura 20 - Tela de cálculo de dessecação e tela de menu	36
Figura 21 - Telas de histórico de cálculo e editar cálculos	37

LISTA DE EQUAÇÕES

Equação 1 - Fórmula do cálculo de perda por dessecação (umidade)	13
Equação 2 - Fórmula do cálculo de resíduo por incineração	15
Equação 3 - Fórmula do cálculo de lipídios ou extrato etéreo	16

LISTA DE ABREVIATURAS E SIGLAS

API – Application Programming Interface

JIT – Just in Time

FPS – Frames per Second

SDK – Software Development Kit

SUMÁRIO

1 INTRODUÇÃO	11
2 REFERENCIAL TEÓRICO	13
2.1 ANÁLISE FÍSICO-QUÍMICA	13
2.2 PERDA POR DESSECAÇÃO	13
2.3 ANÁLISE DE RESÍDUO POR INCINERAÇÃO	15
2.4 ANÁLISE DE LIPÍDIOS OU EXTRATO ETÉREO	16
2.5 DESENVOLVIMENTO PARA DISPOSITIVOS MÓVEIS	17
3 METODOLOGIA	20
3.1 MATERIAIS	20
3.1.1 Firebase	20
3.1.2 Dart	20
3.1.3 Flutter	21
3.2 MÉTODOS	22
4 RESULTADOS E DISCUSSÃO	23
4.1 ESTRUTURA DO PROJETO	23
4.2 IMPLEMENTAÇÃO DO FIREBASE	25
4.3 IMPLEMENTAÇÃO DOS CÁLCULOS	27
4.4 IMPLEMENTAÇÃO DO CÁLCULO DE DESSECAÇÃO	28
4.5 IMPLEMENTAÇÃO DO CÁLCULO DE RESÍDUO POR INCINERAÇÃO	30
4.6 IMPLEMENTAÇÃO DO CÁLCULO DE LIPÍDIOS OU EXTRATO ETÉREO	32
4.7 PROTÓTIPO	34
5 CONSIDERAÇÕES FINAIS	39
REFERÊNCIAS	41

1 INTRODUÇÃO

Segundo Silva (2020, p. 4), “com a evolução tecnológica e a popularização dos dispositivos móveis, surgem frequentemente novas aplicações para diferentes finalidades, que interferem diretamente na cultura e nos hábitos da sociedade”. Assim, ocorre a inovação tecnológica e o surgimento de novas tecnologias, ou novas aplicações, para atender às necessidades do novo ambiente que tem se tornado frequente (ALBERTIN, 2001, p. 43).

Assim, os dispositivos móveis vêm proporcionando uma revolução na forma de comunicação e interação na sociedade, uma vez que estes aparelhos ganharam destaque no cotidiano de uma boa parcela da população (ROSARIO, 2015, p. 5). Nesse contexto, o desenvolvimento de aplicações (APPs) para tais dispositivos se torna altamente visado. Dessa forma, novas tecnologias e abordagens surgem constantemente no mercado para apoiar seus desenvolvimentos (JUNIOR, 2019, p. 1).

As análises físico-químicas visam verificar e quantificar suas características e os nutrientes presentes no alimento, ela também é possível verificar e contribuir na análise da elaboração da tabela nutricional (HERNANDES, 2018). Assim, se faz necessário o estudo dessas propriedades para ajudar na aplicação de diversos métodos de conservação que têm por objetivo evitar as alterações indesejáveis, sejam elas de origem física ou química (SEDUC, 2012, p. 9). Nas análises físico-químicas em alimentos existem cálculos que são realizados em amostras e uma delas é a perda por dessecação (umidade) – Secagem direta em estufa a 105°C, é um método utilizado para remover toda umidade do alimento. Segundo o Instituto Adolfo Lutz (2008, p. 95) “a umidade corresponde à perda de peso sofrida pelo produto quando aquecido em condições nas quais a água é removida”. Assim, com o processo de dessecação realizado, pode-se prosseguir com as amostras e realizar o cálculo.

Uma outra análise bastante utilizada é a de resíduo por incineração, onde segundo o Instituto Adolfo Lutz (2008, p. 102), é a análise onde se obtém resíduos por aquecimento de um produto em temperatura próxima a 550-570 °C. Através desse método os resíduos das cinzas são utilizados pelos profissionais onde eles conseguem retirar a quantidade de amostra necessária para realizar o cálculo. Existem também as análises realizadas nos lipídios ou extrato etéreo – Extração direta em Soxhlet, os lipídios são substâncias que não se podem desidratar ou desfazer em águas. Segundo o Instituto Adolfo Lutz (2008, p. 113), eles “são compostos orgânicos altamente energéticos, contêm ácidos graxos essenciais ao organismo e atuam como transportadores das vitaminas lipossolúveis”.

Neste contexto, nota-se que o processo de realização de uma análise físico-química envolve muitas etapas, desde a preparação dos materiais, a coleta dos parâmetros de resultado do processo e a realização dos cálculos para cada tipo de análise (CAVALCANTI, 2018). Assim, para chegar até o cálculo já é um processo demorado e consequentemente realizar esse cálculo manualmente custaria um tempo que poderia ser otimizado em outra atividade. Nesse sentido, auxiliar na execução dos cálculos de análises físico-químicas ajudaria os profissionais que atuam na área possibilitando uma diminuição no tempo das análises.

Portanto, é justificável o desenvolvimento de um aplicativo mobile que auxilie os profissionais que atuam nesse segmento, sem a necessidade de estar realizando os cálculos de forma manual, trazendo uma praticidade aos mesmos. Uma vez que o software poderá ajudar os profissionais mantendo o histórico de cálculos realizados anteriormente.

Dessa forma, o objetivo principal deste trabalho foi desenvolver um aplicativo mobile que auxilie na execução de cálculos físico-químicos aos profissionais que trabalham com análises físico-químicas em alimentos. Para isso, foram utilizados os conceitos dos cálculos através do livro de edição digital do Instituto Adolfo Lutz que se chama "Métodos físico-químicos para análise de alimentos" (INSTITUTO ADOLFO LUTZ, 1985).

Para atingir esse objetivo foi implementado os cálculos físico-químicos na linguagem de programação *Dart*, utilizado o *Firebase* para armazenar os cálculos realizados por cada profissional e para o desenvolvimento das interfaces foi utilizado o *framework Flutter*.

Este documento está estruturado em cinco capítulos. O segundo capítulo apresenta uma fundamentação teórica sobre a análise físico-química, perda por dessecação, análise de resíduo por incineração, análise de lipídios ou extrato etéreo e desenvolvimento para dispositivos móveis, que auxiliou na base para o desenvolvimento deste trabalho. O terceiro capítulo descreve os materiais utilizados no desenvolvimento do aplicativo mobile, além da metodologia adotada. O quarto capítulo expõe os resultados obtidos e discussões sobre esses resultados. Por fim, o quinto capítulo apresenta as considerações finais e trabalhos futuros.

2 REFERENCIAL TEÓRICO

Nesta seção serão abordados os conceitos relacionados a Análise físico-química na seção 2.1, Perda por dessecação na seção 2.2, Resíduo por incineração na seção 2.3, Lipídios ou extrato etéreo na seção 2.4 e Desenvolvimento para dispositivos móveis na seção 2.5.

2.1 ANÁLISE FÍSICO-QUÍMICA

As análises físico-químicas servem para avaliar as características físicas e químicas da matéria prima ou produto (SEDUC, 2012, p. 9). Pois através delas é possível classificar eles perante a legislação bem como identificar qual método de conservação empregar. Por exemplo, a análise de umidade da farinha é utilizada para avaliar se ela se encontra dentro dos padrões abaixo de 15%, pois acima desse valor o produto pode sofrer alterações indesejáveis de origem química e microbiológica (COSTA et al., 2008).

De acordo com Silva et al. (2016, p. 2), a “análise físico-químicas de frutos de juazeiro ocorrem durante todo o processo de maturação, afetando constituintes como ácidos, amido, açúcares, vitamina C, e umidade, entre outros”.

Assim, com os estudos em busca de características de um alimento, como por exemplo o pequi, apresenta-se um valor nutricional, que é um teor de lipídios (33,4%), compostos fenólicos (209,0 mg.100g⁻¹) e carotenoides (7,2 mg.100g⁻¹) (PAZ et al., 2014, p. 3). Trazendo para o parâmetros físico-químicos, o pequi é caracterizado pelo pH neutro e teores consideráveis de sólidos solúveis (10°Brix) (OLIVEIRA et al., 2010), obtendo um valor nutricional do mesmo.

2.2 PERDA POR DESSECAÇÃO

A perda por dessecação (ou umidade) consiste basicamente na remoção de água em alimentos através da secagem direta em estufas, que pode chegar a uma temperatura de até 105°C. Esta análise pode sofrer algumas variações metodológicas para alimentos mais sensíveis termicamente. Segundo o Instituto Adolfo Lutz (2008, p. 95), as “amostras de alimentos que se decompõem ou iniciam transformações a esta temperatura, devem ser aquecidas em estufas a vácuo, onde se reduz a pressão e se mantém a temperatura de 70°C”.

A secagem segundo Celestino (2010) é “a operação por meio da qual a água ou qualquer outro líquido é removido de um material”. Os alimentos devem ser reduzidos no seu tamanho, em pequenas partículas, para auxiliar na evaporação da água.

A Figura 1 é ilustrada uma estufa convencional para laboratório onde é posto um alimento para realizar secagem.

Figura 1 - Estufa convencional para secagem



Fonte: 7LAB (2021)

A Figura 1 apresenta um exemplo de uma estufa onde é empregado na elaboração de produtos secos para análise. Segundo o procedimento utilizado no Instituto Adolfo Lutz (2008, p. 95), para realizar uma dessecação são utilizadas amostras que podem pesar de 2 a 10g e levadas para uma estufa onde serão aquecidas durante 3 horas. As amostras são resfriadas em um dessecador, até atingir temperatura ambiente, e pesadas novamente. Essa operação se repete até que obtenha peso adequado para realização do cálculo.

Com todo processo realizado, é feito o cálculo para expressar o resultado em porcentagem. A fórmula do cálculo será ilustrada na Equação 1 (IAL 2008).

Equação 1 - Fórmula do cálculo de perda por dessecação (umidade)

$$\frac{100 \times N}{P} = \text{Umidade ou substâncias voláteis a } 105^{\circ}\text{C por cento m/m}$$

Na Equação 1 é apresentada a fórmula usada para o cálculo de perda por dessecação. Onde N é o peso das gramas de umidade e o P é o número de gramas da amostra que foi adicionado. Assim, basta calcular o peso das gramas de umidade, multiplicar por 100 e dividir

pelo número de gramas da amostra, trazendo como resultado a porcentagem da perda por dessecação.

2.3 ANÁLISE DE RESÍDUO POR INCINERAÇÃO

Segundo o Instituto Adolfo Lutz (2008), o “resíduo por incineração ou cinzas é o nome dado ao processo que gera um resíduo obtido por aquecimento de um produto em temperatura próxima a (550-570) °C”. Aquecimento que é feito em um forno específico que se chama Mufla, como apresentado na Figura 2.

Figura 2 - Forno de aquecimento mufla



Fonte: PROLAB (2014)

O forno apresentado na Figura 2 é utilizado para aquecimento rápido e segundo Prolab (2014) é “bastante similar a uma estufa, e é utilizado, principalmente, em laboratórios de química, quando há necessidade de temperaturas muito elevadas na calcinação de substâncias”.

Como o forno trabalha em altas temperaturas, os alimentos postos dentro dele acabam se tornando em resíduo. Os resíduos de acordo com o Instituto Adolfo Lutz (2008, p. 102), “são denominados “cinzas sulfatizadas”. Muitas vezes, é vantajoso combinar a determinação direta de umidade e a determinação de cinzas, incinerando o resíduo obtido na determinação de umidade”.

As amostras ao passar pelo forno em mufla, tem como resultado cinzas que são compostas por resíduo mineral de sódio, potássio, magnésio, cálcio, ferro, fósforo, cobre,

cloreto, alumínio, zinco, manganês e outros compostos minerais (ZAMBLAZI, 2010). As mesmas são mantidas no dessecador até atingirem uma temperatura ambiente. Essa operação poderá ocorrer até que obtenha o peso constante, dependendo do tipo de amostra analisada.

Para realizarem o procedimento de avaliação das cinzas citadas no parágrafo anterior, a análise pode ser realizada e calculada em base seca. (KRUMREICH et al., 2013, p. 2), como a Equação 2 (IAL 2008).

Equação 2 - Fórmula do cálculo de resíduo por incineração

$$\frac{100 \times N}{P} = \text{Cinzas por cento m/m}$$

Na Equação 2 é apresentada a fórmula usada para o cálculo de resíduo por incineração. Onde N é o peso das cinzas e o P é o número de gramas da amostra que foi adicionado. Assim, basta calcular o peso das cinzas, multiplicar por 100 e dividir pelo número de gramas das amostras, trazendo como resultado a porcentagem de resíduo por incineração.

2.4 ANÁLISE DE LIPÍDIOS OU EXTRATO ETÉREO

Lipídios são gorduras e óleos e segundo Cecchi (2003, p. 86) como “componentes do alimento que são insolúveis em água e solúveis em solventes orgânicos, tais como éter etílico, éter de petróleo, acetona, clorofórmio, benzeno e álcoois”.

Um dos tipos de análise empregados para determinação de lipídeos é a extração com uma metodologia de análise de extração com solvente com tipo de equipamento Soxhlet.

O procedimento para a extração de lipídios utilizando a metodologia Soxhlet é feito por etapas. As amostras são pesadas de 2 a 5 gramas de amostra e colocadas em cartucho de Soxhlet ou papel filtro. Em seguida, as amostras são transferidas para o aparelho extrator tipo Soxhlet. Na próxima etapa elas são acopladas ao extrator do balão de fundo chato previamente tarado a 105°C. Finalmente é adicionado éter em quantidade suficiente para um Soxhlet e meio (INSTITUTO ADOLFO LUTZ 2008, p. 114). A mesma possui características de extrato que utiliza refluxo de solvente e podem ser utilizados somente com amostras sólidas e o processo de extração é intermitente (CECCHI 2003, p. 88).

Após a etapa, o éter destilado é transferido para o balão com o resíduo extraído para uma estufa a 105°C, mantendo por cerca de uma hora. As amostras são resfriadas em um

dessecador até a temperatura ambiente. Em seguida são pesadas e a operação é repetida por 30 minutos na estufa e resfriamento até peso constante (no máximo 2 h). (INSTITUTO ADOLFO LUTZ 2008, p. 115). Logo após o processamento é realizado o cálculo da análise de lipídios, como é ilustrado na Equação 3 (IAL 2008).

Equação 3 - Fórmula do cálculo de lipídios ou extrato etéreo

$$\frac{100 \times N}{P} = \text{Lipídios ou extrato etéreo por cento m/m}$$

Na Equação 3 é apresentada a fórmula usada para o cálculo de lipídios ou extrato etéreo. Onde N é o peso das gramas de lipídios após o processo e o P é o número de gramas da amostra que foi adicionado (de 2 a 5 gramas). Assim, basta calcular o peso das gramas de lipídios, multiplicar por 100 e dividir pelo número de gramas da amostra, trazendo como resultado a porcentagem de lipídios ou extrato etéreo.

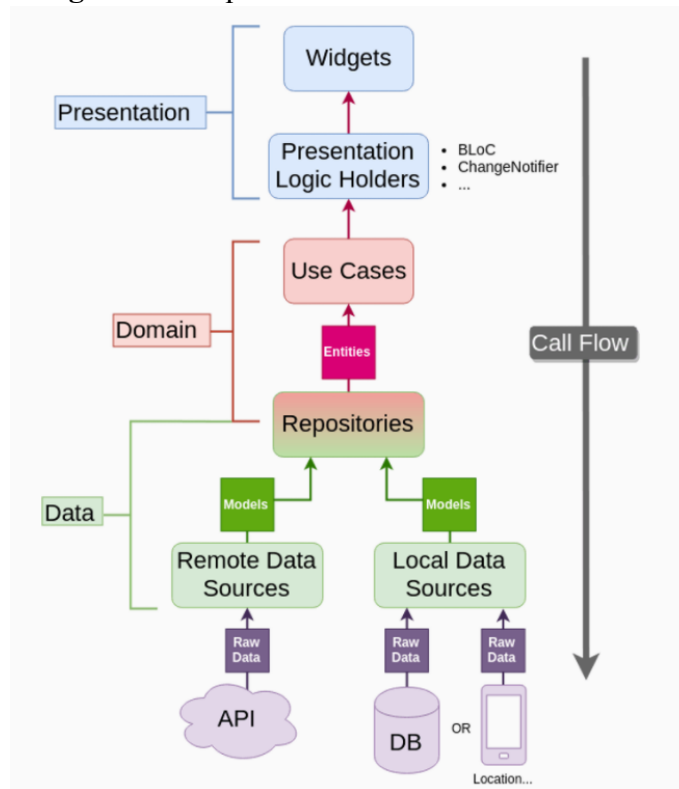
2.5 DESENVOLVIMENTO PARA DISPOSITIVOS MÓVEIS

O desenvolvimento de dispositivos móveis teve um crescimento enorme desde os anos 90 (FIGUEIREDO, 2003, P. 1). A constante evolução das tecnologias de comunicação, em especial dos telefones celulares que estão cada vez mais rápidos, baratos e funcionais, vem provocando uma mudança na forma de se adquirir novos conhecimentos (MATOSKI et al., 2020, p. 2)

Segundo Matoski et al. (2020, p. 2) “Observa-se um aumento da demanda desses equipamentos, tornando-se comum possuir celulares com processadores cada vez mais poderosos”. Dessa forma, explica-se o crescimento de tantos aplicativos móveis, inclusive nos processos de ensino e aprendizagem por conta da inserção das novas tecnologias baseadas na utilização de dispositivos móveis. (MACHADO, 2016, p. 2).

Assim, para desenvolver um aplicativo *mobile* é importante entender os processos de arquitetura. A arquitetura que é apresentada na Figura 3 é a *Clean Architecture* que vem sendo utilizada em média e grande complexidade como uma das melhores soluções, de modo a criar uma arquitetura que seja facilmente escalável, testável e manutenível (MICHUURA, 2020).

Figura 3 - Arquitetura clean architecture



Fonte: MICHIURA (2020)

Na camada de *Presentation*, apresentada na Figura 3, são representadas pelas *Views* e *ViewModels*. *View* é construída e utilizada no *Flutter* como *Widgets* e são representadas como interface visual. Ao passo que, *ViewModels* tem como objetivo de possuir regras de desenvolvimento com objetivo de armazenar e gerenciar os dados, sendo eles recebidos por meio de requisições de API e banco de dados (MICHIURA, 2020).

Na camada de *Domain* são representadas pelos *UseCases* e *Repositories*. *UseCases* é responsável por ordenar e direcionar as regras de negócios e fluxo de dados. Embora o *Repositories* ele verifica e decide onde estão os dados para executar a lógica de acesso aos mesmos.

Por conseguinte, a camada de *Data* é responsável por encontrar e acessar os dados. O mesmo é representado pelo *Data Sources* que pode ser tanto *Remote* ou *Local* dependendo de como pode ser realizada a busca. A busca pelo *Remote* é através de requisições na API e *Local* é através de requisições diretamente no banco de dados ou por celulares.

Importante ressaltar que o aplicativo que será desenvolvido no presente trabalho utilizará a tecnologia *Offline First*. Quando se trata de aplicativo *mobile* é importante atentar-se que haverá lugares em que não haverá conexão. Assim, a solução funcionará com a *SDK* do *Firebase* persistindo os dados no cache do *smartphone* quando não houver rede e sincronizando os dados com o banco de dados *Firestore* quando a rede for restabelecida (DEV, 2021). Dessa forma, o aplicativo funcionará estando conectado à internet ou não.

3 METODOLOGIA

Nessa seção foram apresentados os materiais e os métodos utilizados no processo de desenvolvimento do aplicativo *mobile*.

3.1 MATERIAIS

Os materiais que foram utilizados no desenvolvimento do trabalho proposto são *Firebase*, *Dart* e *Flutter*, descritos a seguir.

3.1.1 Firebase

Firebase é uma plataforma da Google que fornece ferramentas para o desenvolvimento de aplicativos de alta qualidade (FIREBASE, 2021). Segundo Silva (2021) “com ele, é possível oferecer experiências de aplicativos mais ricas para o usuário, otimizando a performance e a experiência da plataforma”.

Dessa mesma forma, ele tem um serviço de banco de dados em tempo real que permite o armazenamento e sincronismo dos dados entre usuários e dispositivos em tempo real com um banco de dados *NoSQL* hospedado na nuvem (NÓBREGA, 2019, p. 14).

No presente trabalho o *Firebase* foi utilizado para armazenamento de dados sendo eles: dados do cadastro de novos usuários e cálculos. Ambos dados foram armazenados no *Firebase*, ao passo que cada usuário terá seus cálculos realizados em uma lista para posteriormente poder utilizá-los de forma rápida e prática.

3.1.2 Dart

Dart é uma linguagem de programação flexível e possui uma integração com *Frameworks* e serviços tais como: *Flutter* e *Firebase* (FRANKLIN; SAMUEL FILHO, 2020, p. 3). O *Dart* foi desenvolvido pela Google em um projeto criado pelos desenvolvedores Lars Bark e Kasper Lund, que foi anunciado oficialmente na conferência GOTO em outubro de 2011 em Aarhus, na Dinamarca (DART, 2021).

O *Dart* é uma linguagem *open source*, fortemente tipada, apesar das declarações de tipo ser opcionais, segue o paradigma de orientação a objetos, além disso sua sintaxe é baseada na linguagem C (SILVA, 2020). Dessa forma o *Dart* oferece um compilador *just-in-time* (JIT) com recompilação incremental (habilitando o *hot reload*), coleções de métricas ao vivo (alimentando *DevTools*) e suporte avançado para depuração (DART, 2021).

Dart é uma plataforma de desenvolvimento de aplicações multiplataforma e pode ser desenvolvido em scripts para desenvolvimento web, aplicações *desktop* e *mobile* utilizando o *framework Flutter* (SILVA, 2020).

Dessa forma, no presente trabalho o *Dart* foi utilizado como código fonte para desenvolvimento das funcionalidades e cálculos juntamente com o *framework Flutter* que fez toda a estrutura lógica e de *layout* do aplicativo. Portanto, o trabalho em conjunto dos mesmos é de grande importância para o desenvolvimento do presente projeto.

3.1.3 Flutter

O *Flutter* é um *framework open source* criado pelo Google, para o desenvolvimento de aplicativos *Android*, *iOS*, *Web* e *Desktop* (SILVA, 2020). Embora seu propósito seja permitir que desenvolvedores criem aplicativos de alta performance e fluídos com experiência nativa em todas as plataformas (FLUTTER, 2021).

O *Flutter* possui um fluxo orientado ao *design* e *widgets*, nos *widget* possui elementos estruturais como botões e menus e elementos de estilo como fontes, cores e formato de texto (BRAGA, 2018). Dessa mesma forma, os *widgets* são estruturados em uma árvore de *widgets* em uma hierarquia pai e filho, onde toda árvore é o que forma o *layout* que é renderizado na tela (SILVA, 2020).

A renderização do *Flutter* executa a 60 *Frames per Second* (FPS) e pode chegar a 120 FPS em dispositivos que suportam essa frequência. *Flutter* usa o motor de renderização chamado *Skia 2D*. Este pode trabalhar com diferentes tipos de hardware e software.

Assim, neste presente projeto o *Flutter* foi utilizado para desenvolver a interface *front-end* do aplicativo, pois o mesmo permite otimizar a interação com cliente criando interfaces em poucas linhas de código e trazendo bons resultados.

3.2 MÉTODOS

Na Figura 4 é demonstrado como seguiu o fluxo de trabalho para o desenvolvimento deste projeto:



A primeira etapa consistiu em reuniões com a especialista do domínio, Professora Msc. Francine Oliveira Batista. Nessas reuniões foram abordados conceitos necessários para compreensão das análises físico-químicas. O estudo do domínio de engenharia de alimentos, assunto específico que foge da área de estudo do autor deste projeto, foi feito mediante reuniões com a especialista por formação nesse campo. O especialista no domínio de Arquitetura de Software, Prof. Esp. Fábio Castro Araújo orientou os conceitos e métodos dentro do escopo da computação e desenvolvimento de software.

Com a etapa de reuniões concluída, foi desenvolvido um protótipo do aplicativo a fim de ser validado com a especialista de domínio para dar continuidade nas próximas etapas do projeto. O protótipo foi de alta fidelidade onde foram apresentadas versões para a especialista de domínio realizar as validações e apresentar correções necessárias para o aplicativo proposto.

Na etapa de validação com a especialista, foi apresentado o protótipo, com o intuito da mesma validar as informações que foram apresentadas no aplicativo. Nesta etapa, a especialista deu o *feedback* sobre a utilização do protótipo. Por fim, após a validação deu-se início à fase de desenvolvimento, onde foram implementadas as funcionalidades solicitadas. Posteriormente, foi realizada uma outra validação com a especialista para verificar se o aplicativo desenvolvido está em conformidade com o que foi decidido na prototipação.

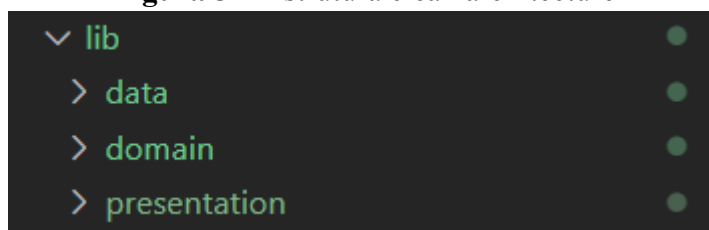
4 RESULTADOS E DISCUSSÃO

Nesta seção é descrito o processo de execução de cada etapa da solução proposta. Nesse contexto, ressalta-se que foram estudados os cálculos de análise físico-químicas de alimentos e das tecnologias: *Dart*, *framework Flutter* e *Firebase* para corroborar no desenvolvimento do presente trabalho. Assim, é descrito nesta seção os resultados obtidos durante a execução do projeto.

4.1 ESTRUTURA DO PROJETO

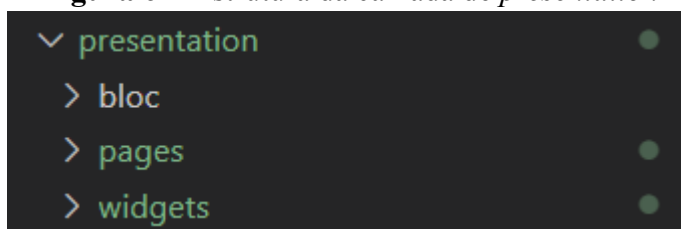
Nesta seção é descrito como foi organizada a estrutura do aplicativo. Assim, foi seguido a arquitetura referenciada no tópico 2.5 (Figura 3), aplicando as regras de cada camada. A *Clean Architecture*, conhecida como arquitetura limpa e organizada pelas camadas de *Presentation*, *Domain* e *Data*, como mostra na Figura 5 a seguir.

Figura 5 - Estrutura clean architecture



Presentation é a camada de apresentação, ou seja, ela é composta por *Pages* e *Widget* que são responsáveis por exibir algo na tela. Nesse sentido, elas disparam eventos para buscar e levar dados para as camadas de *data* e *domain*. A Figura 6 a seguir, mostra como é estruturada a camada de *presentation*.

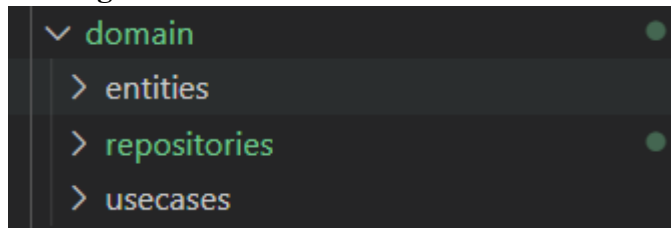
Figura 6 - Estrutura da camada de *presentation*



Na Figura 6, é possível observar que na camada de *presentation* é dividida em subpastas onde é possível visualizar toda a estrutura que gerencia a parte visual do aplicativo. Nesse

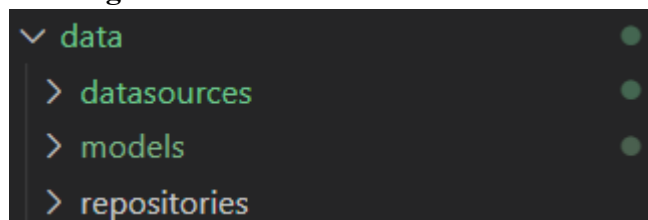
sentido, a pasta mais utilizada foi a pasta *pages*, onde a mesma recebeu os arquivos que gerenciam as telas do aplicativo, ao passo que a pasta de *doc* não recebeu nenhum arquivo e a pasta de *widget* recebeu a tela que gerencia a parte visual do *login* pela *google*. A Figura 7 a seguir, apresenta como é estruturada a camada de *domain*.

Figura 7 - Estrutura da camada de *domain*



A figura 7, apresenta a estrutura de *domain* que é a camada de domínio, ou seja, ela é uma camada que possui lógica de negócios que são os casos de uso e objetos de negócios que são as entidades. Nesse sentido, no presente trabalho não foram utilizados os casos de usos ou entidades diretamente, mas foi utilizado o repositório de gradiente de colorização para realizar o *background* das páginas do aplicativo.

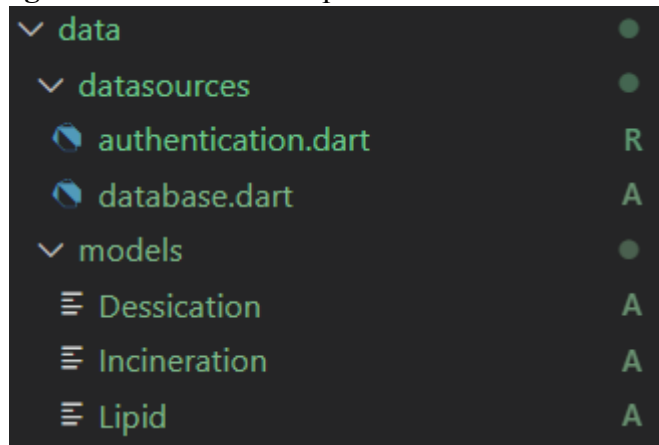
Figura 8 - Estrutura da camada de *data*



A figura 8, apresenta a estrutura do *data* que é a camada de dados, sendo uma das principais camadas, pois através dela é possível realizar as requisições, conexão e armazenamento de dados com ferramentas externas, *API*. Nesse sentido, no presente trabalho foi utilizado como ferramenta externa de armazenamento o *Firebase*, a qual é possível ler, editar, salvar e excluir os dados.

Dessa maneira, a pasta *datasources* é a responsável por receber os arquivos que fazem as requisições de dados e autenticação, ao passo que a pasta *models* é a responsável por realizar o armazenamento de dados. A figura 9, mostra como está sendo estruturadas as pastas *datasources* e *models*.

Figura 9 - Estrutura das pastas *datasources* e *models*




É possível observar que a pasta *datasources* recebe os arquivos de *authentication.dart* e *database.dart*. O arquivo *authentication.dart* é o responsável por autenticar o usuário na plataforma como também verificar se o mesmo está logado na plataforma toda vez que o usuário fecha e abre o aplicativo. O arquivo *database.dart* é o arquivo que faz as requisições de leitura, editar, adicionar e remover dados do *Firebase*.

Logo, a arquitetura do presente trabalho está utilizando também o *Offline First*, a qual é a funcionalidade responsável por ler, editar, adicionar e remover dados no aplicativo localmente mesmo estando sem conexão com a rede. Após a conexão, a tecnologia permite sincronizar o que foi feito localmente com a base de dados remota. Diante disso, o *Firebase* oferece essa funcionalidade de *Offline First* que é integrada ao *Firestore*. Assim, o *Firestore* usa um banco de dados local que sincroniza automaticamente com o servidor assim que o aplicativo estabelecer conexão com a internet.

4.2 IMPLEMENTAÇÃO DO FIREBASE

Para a implementação do *firebase* dentro do presente trabalho, foi necessário importar a sua biblioteca via código dentro do arquivo *pubspec.yaml* que se encontra na raiz de projeto do *flutter*, como mostrar a figura 10.

Figura 10 - Importar biblioteca do Firebase



```
1 dependencies:
2   flutter:
3     sdk: flutter
4
5
6   cupertino_icons: ^1.0.2
7   firebase_core: ^1.3.0
8   cloud_firestore: ^2.2.2
9   firebase_auth: ^1.0.1
10  google_sign_in: ^5.0.0
11  modal_bottom_sheet: ^2.0.0
```

Na figura 10 é apresentado na linha 7 e 8 o *firebase_core* e *cloud_firestore* com suas respectivas versões. O *firebase_core* é responsável por conectar o aplicativo ao projeto *firebase* (FIREBASE, 2022). O *cloud_firestore* é um banco de dados em nuvem *NoSQL* flexível e escalável para armazenar e sincronizar dados (FIREBASE, 2022). Portanto, essa parte do código é usada para importar a biblioteca do *firebase* para desfrutar das funcionalidades de armazenamento que a ferramenta disponibiliza para os seus usuários.

Com o *firebase* importado, foi necessário criar e configurar o projeto do aplicativo junto com a plataforma do *firebase*. Assim, foi possível construir as funcionalidades responsáveis por criar, editar e deletar os cálculos propostos do presente trabalho. A figura 11, mostra uma funcionalidade implementada para armazenar um novo cálculo utilizando o *firebase*.

Figura 11 - Funcionalidade para armazenar um cálculo

```
1 Future<void> create(data) async {
2   try {
3     CollectionReference collectionReference =
4     FirebaseFirestore.instance.collection('calc');
5     await collectionReference.add(data);
6   } catch (e) {
7     print(e);
8   }
9 }
10
```

A figura 11, apresenta a funcionalidade *create*. Ela é responsável por fazer as inserções dos cálculos realizados no aplicativo. Na linha 3, é possível observar que o código acessa a coleção que se chama *calc*, onde está sendo armazenado os dados. Na linha 5, ela aguarda o *firebase* responder que foi encontrado a coleção *calc*, se ela for encontrada, a funcionalidade faz uma *add* que armazena o cálculo que veio na variável *data*, caso não encontre, a funcionalidade retorna um erro que é apresentado na linha 7.

4.3 IMPLEMENTAÇÃO DOS CÁLCULOS

Para o desenvolvimento dos cálculos de análise físico-químicas, foi utilizado o livro de edição digital do Instituto Adolfo Lutz que se chama "Métodos físico-químicos para análise de alimentos" (INSTITUTO ADOLFO LUTZ, 1985). Assim, foi escolhido e estudado apenas três tipos de cálculos: os cálculos de Perda por dessecação, Resíduo por incineração e Lipídios ou extrato etéreo.

De acordo com as equações 1, 2 e 3 apresentadas nas seções 2.2, 2.3 e 2.4 vistas anteriormente no presente trabalho, foi possível desenvolver os cálculos com o apoio das tecnologias Dart e Flutter como mostra as seções 4.4, 4.5, 4.6.

4.4 IMPLEMENTAÇÃO DO CÁLCULO DE DESSECAÇÃO

De acordo com a equação 1 apresentada na seção 2.2, foi possível desenvolver a implementação do cálculo de dessecação como mostra a figura 10.

Figura 12 - Representação do cálculo de dessecação

A screenshot of a code editor window with a dark background and three colored window control buttons (red, yellow, green) in the top-left corner. The code is written in a light-colored font and is as follows:

```
1 Text(  
2   calculatorDesiccationNumber(  
3     dessication.content, dessication.sample),  
4   style: TextStyle(fontSize: 18)),
```

Na figura 10 é apresentado na linha 1 o código onde é usada a *Widget Text* que é responsável por apresentar de forma textual o cálculo da análise de Perda por dessecação. Assim, é possível visualizar na linha 2 que a *Widget* chama a função *calculatorDesiccationNumber*. Essa função recebe dois parâmetros: *dessication.content* e *dessication.sample*, ambos parâmetros receberam os valores que foi digitado pelo usuário e armazenado no *model Dessication*. Nesse sentido, é possível pegar esses valores para passar no parâmetro da função e assim realizar o cálculo. A figura 11 é a responsável por ilustrar como é realizado e retornado o cálculo de dessecação.

Figura 13 - Funções que realizam o cálculo de perda por dessecação

```
1 calculatorDesiccationNumber(content, sample) {
2   return calculatorDesiccationFmt(
3     content, sample, calcululatorDesiccation(content, sample));
4 }
5
6 calcululatorDesiccation(content, sample) {
7   return content.length > 0 && sample.length > 0
8     ? (100.0 * double.parse(content)) / double.parse(sample)
9     : '0';
10 }
11
12 calculatorDesiccationFmt(content, sample, calc) {
13   return '(100X${content})/${sample} = $calc';
14 }
```

A função *calculatorDesiccationNumber* recebe dois parâmetros: *content* que é o parâmetro responsável por receber o peso das gramas dos conteúdos das análises e o *sample* que é o parâmetro responsável por receber o número de gramas das amostras de cada tipo de análise físico-química. Nesse sentido, nas linhas 2 e 3 é chamada a função *calculatorDessicationFmt* que passa como parâmetro os valores das amostras e a função *calculatorDessication* que é a responsável por retornar o valor do cálculo.


É importante ressaltar que a função *calculatorDessication* que está na linha 6 retorna o cálculo da seguinte forma: primeiro ela realiza uma validação se *content* e *sample* é maior que 0, pois os valores que estão dentro de *content* e *sample* são do tipo *string*. Assim, o *Dart* realiza a conversão de *string* para *double* para manipulação dos valores. Nesse sentido, o *Dart* obriga a realizar essa validação antes que seja feita a conversão. Na sequência, a função multiplica por cem (100) o conteúdo de *content* e após isso divide pelo valor de *sample* que gerará o valor resultante do cálculo.

Na linha 12 a função *calculatorDesiccationFmt* recebe o resultado do cálculo que foi realizado pela função *calculatorDesiccation* e os parâmetros das amostras. A função na linha 13 retorna para a Figura 10 a *string* que possui o valor do cálculo junto com a fórmula usada, sendo possível do usuário identificar o resultado do cálculo junto com a fórmula usada pelo aplicativo.

4.5 IMPLEMENTAÇÃO DO CÁLCULO DE RESÍDUO POR INCINERAÇÃO

De acordo com a equação 2 apresentada na seção 2.3, foi possível desenvolver a implementação do cálculo de resíduo por incineração como mostra a figura 12.

Figura 14 - Cálculo de resíduo por incineração

A screenshot of a code editor window with a black background and light-colored text. The code is written in Dart and defines a Text widget. It consists of four lines of code, numbered 1 to 4. Line 1 starts with 'Text(' and line 4 ends with a closing parenthesis and a comma. The code is color-coded: 'Text' is blue, 'calculatorIncinerationNumber' is green, 'incineration.content' and 'incineration.sample' are green, and 'fontSize: 18' is purple. The editor has three colored window control buttons (red, yellow, green) in the top-left corner.

```
1 Text(  
2   calculatorIncinerationNumber(  
3     incineration.content, incineration.sample),  
4   style: TextStyle(fontSize: 18)),
```

Na figura 12 apresenta na linha 1 a *Widget Text* que é responsável por apresentar de forma textual o cálculo da análise de Resíduo por incineração. Assim, na linha 2 é possível visualizar a *Widget* que chama a função *calculatorIncinerationNumber*. A função recebe dois parâmetros: *incineration.content* e *incineration.sample*, ambos parâmetros receberam os valores que foi digitado pelo usuário e armazenado no *model Incineration*. A figura 13 é a responsável por ilustrar como é realizada e retornado o cálculo de incineração.

Figura 15 - Funções que realizam o cálculo de resíduo por incineração

```
1 calculatorIncinerationNumber(content, sample) {
2   return calculatorIncinerationFmt(
3     content, sample, calcululatorIncineration(content, sample));
4 }
5
6 calcululatorIncineration(content, sample) {
7   return content.length > 0 && sample.length > 0
8     ? (100.0 * double.parse(content)) / double.parse(sample)
9     : '0';
10 }
11
12 calculatorIncinerationFmt(content, sample, calc) {
13   return '(100X${content})/${sample} = $calc';
14 }
```

A função *calculatorIncinerationNumber* recebe dois parâmetros: *content* que é o parâmetro responsável por receber o peso das gramas dos conteúdos das análises e o *sample* que é o parâmetro responsável por receber o número de gramas das amostras de cada tipo de análise físico-química. Assim, é possível observar que nas linhas 2 e 3 é chamada a função *calculatorIncinerationFmt* que passa como parâmetro os valores das amostras e a função *calculatorIncineration* que é a responsável por retornar o valor do cálculo.

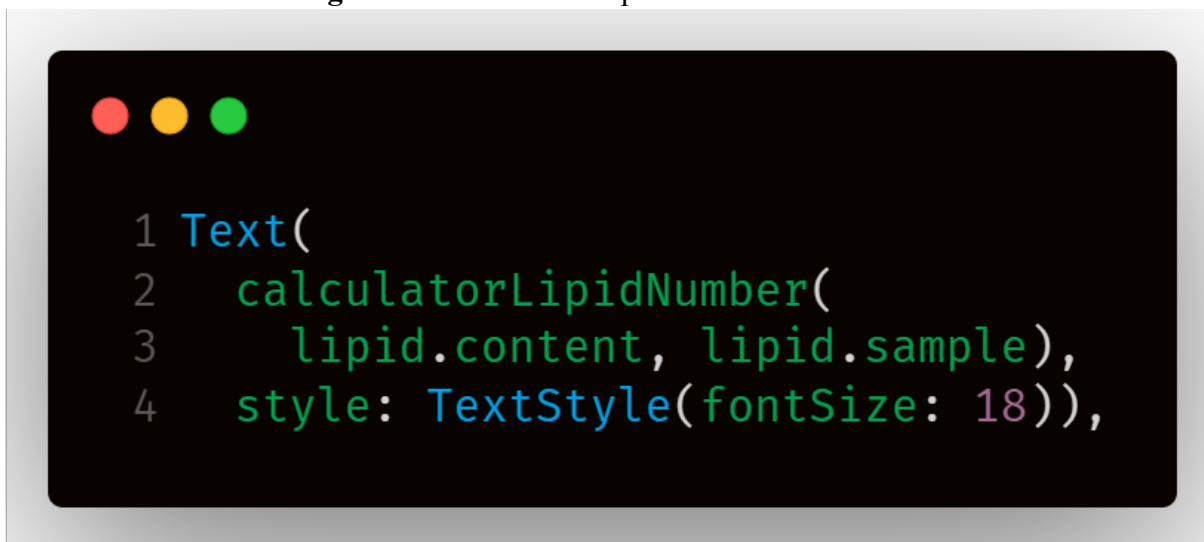
É importante ressaltar que a função *calculatorIncineration* que está na linha 6 retorna o cálculo da seguinte forma: primeiro é realizado uma validação se *content* e *sample* possui o valor maior que 0, pois os valores que estão dentro de *content* e *sample* são do tipo *string*. Assim, o *Dart* realiza a conversão de *string* para *double* para manipulação dos valores. Nesse sentido, o *Dart* obriga a realizar essa validação dos valores antes que seja feita a conversão. Na sequência, a função multiplica por cem (100) o conteúdo de *content* e após isso divide pelo valor de *sample* que gerará o valor resultante do cálculo.

Na linha 12 a função *calculatorIncinerationFmt* recebe o resultado do cálculo que foi realizado pela função *calculatorIncineration* e os parâmetros das amostras. Assim, a função na linha 13 retorna para a Figura 12 a *string* que apresentará o valor do cálculo junto com a fórmula utilizada pelo aplicativo.

4.6 IMPLEMENTAÇÃO DO CÁLCULO DE LIPÍDIOS OU EXTRATO ETÉREO

De acordo com a equação 3 apresentada nas seções 2.4, foi possível desenvolver a implementação do cálculo de lipídios ou extrato etéreo como mostra a figura 14.

Figura 16 - Cálculo de lipídios ou extrato etéreo

A screenshot of a code editor window with a black background and three colored window control buttons (red, yellow, green) in the top-left corner. The code is written in a light blue font and consists of four lines. Line 1: `1 Text(`. Line 2: `2 calculatorLipidNumber(`. Line 3: `3 lipid.content, lipid.sample),`. Line 4: `4 style: TextStyle(fontSize: 18)),`.

```
1 Text(  
2   calculatorLipidNumber(  
3     lipid.content, lipid.sample),  
4   style: TextStyle(fontSize: 18)),
```

Na figura 14 é apresentado a *Widget Text* na linha 1, a qual é a responsável por apresentar de forma textual o cálculo de Lipídios ou extrato etéreo. Assim, é possível visualizar na linha 2 que é chamada a função *calculatorLipidNumber*. Essa função recebe dois parâmetros: *lipid.content* e *lipid.sample*, ambos parâmetros receberam os valores que foi digitado pelo usuário e armazenado no *model Lipid*. Nesse sentido, é possível passar esses valores como parâmetro e assim realizar o cálculo. A figura 15 é a responsável por ilustrar como é realizada e retornado o cálculo de lipídios.

Figura 17 - Funções que realizam o cálculo de lipídios ou extrato etéreo

```
1 calculatorLipidNumber(content, sample) {
2   return calculatorLipidFormt(
3     content, sample, calcululatorLipid(content, sample));
4 }
5
6 calcululatorLipid(content, sample) {
7   return content.length > 0 && sample.length > 0
8     ? (100.0 * double.parse(content)) / double.parse(sample)
9     : '0';
10 }
11
12 calculatorLipidFormt(content, sample, calc) {
13   return '(100X${content})/${sample} = $calc';
14 }
```

A função *calculatorLipidNumber* recebe dois parâmetros: *content* que é o parâmetro responsável por receber o peso das gramas dos conteúdos das análises e o *sample* que é o parâmetro responsável por receber o número de gramas das amostras de cada tipo de análise físico-química. Nesse sentido, nas linhas 2 e 3 é possível observar que é chamada a função *calculatorLipidFormt* que passa como parâmetro os valores das amostras e a função *calculatorLipid* que é a responsável por retornar o valor do cálculo.

É importante ressaltar que a função *calculatorLipid* na linha 6 retorna o cálculo da seguinte forma: primeiro ela realiza uma validação se os valores de *content* e *sample* é maior que 0, pois os valores que estão dentro de *content* e *sample* são do tipo *string*. Assim, o *Dart* realiza a conversão de *string* para *double* para manipulação dos valores. Nesse sentido, o *Dart* obriga a realizar essa validação antes que seja feita a conversão para assim a função multiplicar por cem (100) o conteúdo de *content* e após isso divide pelo valor de *sample* que gerará o valor resultante do cálculo.

Na linha 12 a função *calculatorLipidFormt* recebe o resultado do cálculo que foi realizado pela função *calculatorLipid* e os parâmetros das amostras. Assim, é possível a função na linha 13 retornar para a Figura 14 a *string* que possui o valor do cálculo junto com a fórmula usada pelo aplicativo.

4.7 PROTÓTIPO

Após o estudo sobre os cálculos, foi criado um protótipo do aplicativo a ser desenvolvido. O protótipo é de alta fidelidade onde foram apresentadas versões para a especialista de domínio realizar as validações e indicar correções necessárias. Nesse viés, foi utilizado para a criação do protótipo o *Figma*, uma ferramenta para prototipagem de projetos de *design* de interface.

Nas figuras a seguir, será apresentado as interfaces do protótipo juntamente com a explicação de cada tela.

Figura 18 - Tela de login e homepage



A Figura 16 apresenta as principais telas do aplicativo *mobile*, a tela 1 é a tela inicial do aplicativo onde apresenta o nome do aplicativo a qual se chama %NutriFood, que por sua vez foi escolhido o nome em inglês por se tornar um nome mais curto e objetivo. Nesse sentido, a tela 1 também é a responsável pela autenticação do usuário, uma vez que os

usuários autenticados no aplicativo, terão acesso para realizar os cálculos de Incineração, Lipídios e Dessecação como está sendo apresentado na Figura 16 e tela 2.

A tela 2 permite que os usuários selecionem o cálculo de análise físico-química de alimentos que necessitam. É possível também acessar o menu lateral clicando no ícone do lado esquerdo superior para buscar históricos de cálculos já realizados no aplicativo, como também sair da sessão do mesmo.

Figura 19 - Telas de cálculos de incineração e lipídios



A tela 3 é a tela que permite ao usuário realizar o cálculo de Incineração. Nesse sentido, o mesmo poderá adicionar os valores obtidos através do estudo da temática, adicionando a quantidade recolhida de Cinzas e Amostras. Assim, ao preencher os campos o aplicativo automaticamente realizará o cálculo para o usuário que será mostrado logo abaixo com a fórmula usada junto com o valor resultante. Importante ressaltar que o usuário poderá

salvar esse cálculo através do botão Salvar no canto inferior da tela, como também, sair da tela sem salvar o cálculo, clicando no botão superior da esquerda.

A tela 4 permite ao usuário realizar o cálculo de Lipídios. Assim, ela possui a mesma estrutura lógica que foi apresentada na tela 3. Diante disso, o usuário poderá adicionar os valores obtidos durante o estudo sobre a temática nos campos do formulário para assim o aplicativo devolver o resultado do cálculo. É válido pontuar, que o usuário poderá salvar o cálculo clicando no botão Salvar no canto inferior da tela, quanto sair da tela clicando no lado esquerdo superior.

Figura 20 - Telas de cálculo de dessecação e tela de menu



A tela 5 permite ao usuário realizar o cálculo de Dessecação. Assim, ela possui a mesma estrutura lógica que foi apresentada na tela 3 e 4. Diante disso, o usuário poderá adicionar os valores obtidos durante o estudo sobre a temática nos campos do formulário para

assim o aplicativo devolver o resultado do cálculo. É válido pontuar, que o usuário poderá salvar o cálculo clicando no botão Salvar no canto inferior da tela, quanto sair da tela clicando no lado esquerdo superior.

Destarte, é possível observar que nas telas 3, 4 e 5 são de cores diferentes. Diante disso, através das validações das telas com a especialista de domínio a mesma ressaltou que seria interessante aproximar o aplicativo do usuário, ou seja, colocar as telas de cálculos com a coloração referente a especialidade do cálculo.

Dessa forma, a tela 6 é a tela que é aberta ao clicar no menu no canto superior à esquerda, referenciada na tela 2. Assim, é possível observar que na tela é apresentado o Email, Nome e Foto que está sendo usado pelo usuário na sua conta de Gmail. Nesse sentido, o usuário também poderá visualizar o histórico de cálculos realizado por ele no aplicativo como também sair do aplicativo clicando em Sair.

Figura 21 - Telas de histórico de cálculo e editar cálculos



A tela 7 é a tela responsável por listar o histórico de cálculos salvos pelo usuário. Assim o usuário consegue visualizar todos os cálculos que ele já realizou no aplicativo com sua conta. Nesse sentido, é apresentado o histórico por itens, mostrando o nome do tipo de cálculo realizado junto com a equação e o valor do resultado do cálculo.

A tela 8 é a tela que é possível acessar quando o usuário clica no cálculo que foi listado na tela 7. Diante disso, é permitido nessa tela que o usuário edite os valores que foram passados no formulário e salvar o cálculo com um novo valor ou até mesmo deletar esse cálculo salvo clicando no botão no lado direito superior. Logo, o usuário retornará para tela 7 com um cálculo a menos no histórico.

5 CONSIDERAÇÕES FINAIS

Este trabalho veio com o propósito de desenvolver um aplicativo que pudesse auxiliar na execução de cálculos físico-químicos aos profissionais que trabalham com análises físico-químicas em alimentos. Para essa finalidade, foram executadas etapas que envolveram diversos objetivos, desde o estudo sobre análise físico-química de alimentos, cálculos que foram realizados até estudos das linguagens de desenvolvimento *mobile*.

Os objetivos específicos do trabalho, que eram organizar os cálculos de análises físico-químicas. Assim, foram escolhidos os cálculos de perda por dessecação, resíduo por incineração, lipídios ou extrato etéreo, a criação do protótipo do aplicativo e desenvolver o aplicativo que realizasse os cálculos. Ambos objetivos foram atingidos.

O objetivo de organizar os cálculos de análises físico-químicas para implementação foi alcançado através do estudo do livro de edição digital do Instituto Adolfo Lutz que se chama "Métodos físico-químicos para análise de alimentos" (INSTITUTO ADOLFO LUTZ, 1985). Sendo a etapa que mais demandou atenção, pois foi realizado um estudo específico de uma área de atuação diversa da tecnologia, então necessitou de uma atenção maior para que não houvesse informações incompatíveis sobre o tema abordado.

O objetivo de criar um protótipo foi alcançado utilizando a plataforma *Figma*, plataforma que é possível desenvolver diversos protótipos de interfaces de baixa fidelidade como também de interfaces de alta fidelidade. Nesse sentido, no presente trabalho foi utilizado protótipo de alta fidelidade para que houvesse a validação das telas pela especialista de domínio.

O objetivo de desenvolver aplicativo que realiza os cálculos foi alcançado utilizando o *Flutter* e a linguagem de programação *Dart*. Assim, utilizando essas ferramentas foi possível desenvolver o aplicativo com todos os cálculos propostos. Além disso, o código-fonte que foi utilizado no desenvolvimento é centralizado, ou seja, não possui dependência de terceiros como *plugins*, assim sendo possível de realizar futuras manutenção e adição de novas funcionalidades na aplicação.

Em vista disso, e através dos resultados apresentados na seção anterior, o objetivo principal deste trabalho foi atingido através do desenvolvimento do aplicativo *mobile* inédito. O aplicativo permitirá que os profissionais realizem os cálculos de forma automatizada, além de permitir uma otimização no tempo das análises de cada estudo.

Para trabalhos futuros, é possível realizar melhorias no aplicativo como por exemplo: inserção de mais cálculos de análise físico-químico que torna o aplicativo mais amplo em diversas análises. Realizar autenticação por mais redes sociais ou cadastro interno pelo aplicativo, ou seja, trazer formas diferentes para que os profissionais loguem e acessem o aplicativo de diversos modos. A inserção de um novo menu de relatórios de levantamento de cálculos, ou seja, possibilitando uma visão de quais cálculos mais realizados por aquele usuário no aplicativo.

REFERÊNCIAS

ALBERTIN, Alberto Luiz. Valor estratégico dos projetos de tecnologia de informação. **Revista de Administração de Empresas**, [S.L.], v. 41, n. 3, p. 42-50, set. 2001. FapUNIFESP (SciELO). <<http://dx.doi.org/10.1590/s0034-75902001000300005>>.

BRAGA, Vinicius. **Flutter Framework: O que é e como funciona**. 2018. Disponível em: <https://micreirhttps://micreiros.com/flutter-framework-o-que-e-e-como-funciona/os.com/flutte-r-framework-o-que-e-e-como-funciona/>. Acesso em: 19 jun. 2021.

CAVALCANTI, Dra. Elisa D'avila Costa. **Preparo de Amostra para Análise de Alimentos**. Rio de Janeiro: En/Unirio, 2018. 59 slides, color.

CECCHI, Heloisa Máscia. **Fundamentos teóricos e práticos em análise de alimentos**. 2. ed. Campinas, Sp: Unicamp, 2003. 208 p.

CELESTINO, S. M. C. **Princípios de secagem de alimentos**. Planaltina, DF: Embrapa Cerrados, 2010.

COSTA, Maria das Graças da *et al.* Qualidade tecnológica de grãos e farinhas de trigo nacionais e importados. **Ciência e Tecnologia de Alimentos**, [S.L.], v. 28, n. 1, p. 220-225, mar. 2008. FapUNIFESP (SciELO). <http://dx.doi.org/10.1590/s0101-20612008000100031>

DART. Dart programming language: **Dart documentation**, 2021. Disponível em: . Acesso em: 21 abr. 2021.

DEV, Pub. **Brick Offline First**. 2021. Disponível em: https://pub.dev/packages/brick_offline_first. Acesso em: 13 jul. 2021.

FIGUEIREDO, Carlos MS; NAKAMURA, Eduardo. **Computação móvel: Novas oportunidades e novos desafios**. T&C Amazônia, v. 1, n. 2, p. 21, 2003.

FIREBASE. **Documentation.** 2021. Disponível em: <<https://firebase.google.com/docs?hl=pt-BR>>. Acesso em: 21 abr. 2021.

FLUTTER. Flutter: Beautiful native apps in record time, Flutter documentation, 2021. Disponível em: . Acesso em: 21 abr. 2021.

FIREBASE. **Core.** 2022. Disponível em: <https://firebase.flutter.dev/docs/core/usage/>. Acesso em: 26 jan. 2022.

FIREBASE. **Cloud Firestore.** 2022. Disponível em: <https://firebase.flutter.dev/docs/firestore/overview/>. Acesso em: 26 jan. 2022.

FRANKLIN, Matheus Maião; SAMUEL FILHO, Ronaldo Aparecido. **Desenvolvimento de um Sistema de Gestão Escolar com o uso da Linguagem Dart com Framework Flutter.** Revista E-F@Tec, Garça, v. 10, n. 1, p. 1-10, 1 out. 2020.

Geraldo R. Mateus; Antonio A. F. Loureiro. **Introdução a Computação Móvel.** 11a Escola de Computação, Rio de Janeiro, RJ, 1998.

HERNANDES, Pedro. **Análise físico-química dos alimentos: entenda como funciona.** Entenda como funciona. 2018. Disponível em: <https://www.alsglobal.com/%2Fpt-br%2Fnews%2Fartigos%2F2018%2F12%2Fanlise-fsicoquimica-dos-alimentos-entenda-como-funciona>. Acesso em: 16 jun. 2021.

JUNIOR, Heleno de Souza Campos. Desenvolvimento de aplicações para dispositivos móveis. **ANALECTA-Centro Universitário Academia**, v. 4, n. 4, 2019.

INSTITUTO ADOLFO LUTZ. **Normas Analíticas do Instituto Adolfo Lutz. v. 1: Métodos químicos e físicos para análise de alimentos**, 1. ed. São Paulo: IMESP, 1985. p. 96-116.

KRUMREICH, Fernanda Doring et al. Teor de cinzas em acessos de abóboras (cucurbita máxima l.) do Rio Grande do Sul. In: **Embrapa Clima Temperado-Artigo em anais de**

congresso (ALICE). In: SIMPÓSIO DE ALIMENTOS PARA A REGIÃO SUL, 8., 2013, Passo Fundo: anais. Passo Fundo: Universidade de Passo Fundo, 2013., 2013.

MACHADO, Leonardo Davi Pereira; BERCKENBROCK, Carla Diacui Medeiros; SIPLE, Ivanete Zuchi. Desenvolvimento de aplicativos para Aprendizagem Colaborativa apoiada por Dispositivos Móveis: uma análise dos requisitos. **Anais do Computer on the Beach**, p. 001-010, 2016.

MATOSKI, Adalberto et al. **Uso de dispositivos móveis como ferramenta de aprendizado: riscos e oportunidades.** Brazilian Journal of Development, v. 6, n. 1, p. 4673-4687, 2020.

MICHIURA, Fabio. **A importância da Clean Architecture no Flutter.** 2020. Disponível em: <https://www.objective.com.br/insights/a-importancia-da-clean-architecture-no-flutter/>. Acesso em: 13 jul. 2021.

MICROSOFT. **Funções genéricas (C++/CLI).** 2021. Disponível em: <https://docs.microsoft.com/pt-br/cpp/extensions/generic-functions-cpp-cli?view=msvc-160>. Acesso em: 13 out. 2021.

NÓBREGA, Claison Marques da. **Implementação de um sistema mobile de apoio ao Profissional de Educação Física para a realização da avaliação física, periodização de treino e prescrição dos exercícios.** 2019. 22 f. TCC (Graduação) - Curso de Sistemas de Informação, Centro Universitário Luterano de Palmas, Palmas, 2019.

OLIVEIRA, M.E.B.; GUERRA, N.B.; MAIA, A.H.N.; ALVES, R.E.; MATOS, N.M.S.; SAMPAIO, F.G.M.; LOPES, M.M.T. **Características químicas e físico-químicas de pequis da chapada do Araripe, Ceará.** Revista Brasileira de Fruticultura. v.32, n.1, p.114-125, 2010.

PAZ, Josiane Gonçalves da *et al.* **ANÁLISE DA COMPOSIÇÃO NUTRICIONAL E DE PARÂMETROS FÍSICO-QUÍMICOS DO PEQUI (CARYOCAR BRASILIENSE CAMB) IN NATURA.** Revista Científica Linkania Master, [s. l], v. 1, n. 5, p. 1-14, jan. 2014.

PROLAB. **Saiba o que é e como funciona um forno mufla.** 2014. Disponível em: <https://www.prolab.com.br/blog/equipamentos-aplicacoes/saiba-o-que-e-e-como-funciona-um-forno-mufla/#:~:text=Forno%20Mufla%20ou%20Forno%20de,elevadas%20na%20calcina%C3%A7%C3%A3o%20de%20subst%C3%A2ncias..> Acesso em: 24 abr. 2021.

SEDUC – EEEP (**Escola Estadual de Educação Profissional. Curso Técnico em Nutrição e Dietética** – Microbiologia de Alimentos. Ceara, 2012). Disponível em: https://www.seduc.ce.gov.br/wpcontent/uploads/sites/37/2011/10/nutricao_e_dietetica_microbiologia_de_alimentos.pdf

ROSARIO, C. FÁBIO, 2015. **Developing mobile applications multi-platform.** 39 f. Trabalho de Diplomação (Curso Superior de Tecnologia em Análise e Desenvolvimento de Sistemas). Universidade Tecnológica Federal do Paraná – UTFPR. Medianeira. 2015

SILVA, Eduardo. **Firestore: o que é e quando usar no desenvolvimento mobile?** 2021. Disponível em: <https://blog.geekhunter.com.br/firebase-o-que-e-e-quando-usar-no-desenvolvimento-mobile/>. Acesso em: 19 jun. 2021.

SILVA, Jéssica Leite da; COSTA, Franciscleudo Bezerra da; NASCIMENTO, Ana Marinho do; COSTA, Rafaela Rodrigues do Vale; SANTIAGO, Mahyara de Melo. **Avaliação física e físico-química de frutos de juazeiro (Zizyphus joazeiro Mart.) em diferentes estádios de maturação.** Revista Verde de Agroecologia e Desenvolvimento Sustentável, [S.L.], v. 11, n. 5, p. 177, 9 dez. 2016. Grupo Verde de Agroecologia e Abelhas. <http://dx.doi.org/10.18378/rvads.v11i5.4743>.

SILVA, Lucas Costa. **Desenvolvimento de um aplicativo mobile para auxílio à localização no ambiente interno do CEULP/ULBRA.** 2020. 25 f. Trabalho de Conclusão de Curso (Graduação) – Curso de Sistemas de Informação, Centro Universitário Luterano de Palmas, Palmas/TO, 2020.

WILBERG, Felix Emílio Prado Cornejo; Regina Isabel Nogueira; Viktor Christian. **Secagem e desidratação.** Disponível em: https://www.agencia.cnptia.embrapa.br/gestor/tecnologia_de_alimentos/arvore/CONT000fid5sgie02wyiv80z4s473tokdiw5.html. Acesso em: 24 abr. 2021.

ZAMBIAZI, R.C. **Análise Físico Química de Alimentos.** Pelotas: Editora Universitária/UFPEL, 202p. 2010. SAS Institute. System for Information, versão 8.0. Cary, 2007. 1 CD Rw.

7LAB. **Estufa de secagem e esterilização 7Lab Analógica INOX - 50 a 250°C - 85 L (Bivolt).** 2021. Disponível em: https://www.7lab.com.br/equipamentos-para-laboratorio/estufa-de-secagem-e-esterilizacao/estufa-de-secagem-e-esterilizacao-7lab-analogica-inox-50-a-250oc-85-l-bivolt?parceiro=5060&campaignid=10948323807&adgroupid=107551945356&keyword=&network=u&utm_medium=cpc&gclid=CjwKCAjwz_WGBhA1EiwAUAXIcVw7nS2SAKOk3suohpZaWMvYyRjMtzzfobfUTIIDzQdaT1p6MUMOqhoCKEYQAvD_BwE. Acesso em: 02 jul. 2021.