**Relatório Progresso**

**de**

**Laboratório de Projeto em Engenharia Informática**

**da**

**Licenciatura em Engenharia Informática**

|  |  |
| --- | --- |
| Nome do aluno | Número do aluno |
| **Francisco Gonçalves Gouveia** | **74044** |
| **Tiago José Ferreira Fernandes** | **73701** |

|  |
| --- |
| Título do trabalho |
| **O uso de smartwatch na gestão de Sistemas de Manutenção e Avarias na Indústria 4.0** |

|  |
| --- |
| Nome do orientador |
| **Arsénio Reis** |

|  |
| --- |
| Nome do(s) coorientador(es) |
| **Tiago Pinto**  **Claudio Lima** |

|  |
| --- |
| Nome do coorientador externo à UTAD |
|  |

## Resumo

No seguimento da última fase do Projeto de Laboratório Final, foi requerido ao presente grupo, a entrega de um relatório de progresso que integrasse alguns conceitos adicionais para melhor esclarecimento das diferentes paradigmas e dimensões associadas ao assunto principal do trabalho: o desenvolvimento de uma aplicação para um *wearable.* Ora com isto em mente, o grupo, depois de uma reunião com os diferentes integrantes dos restantes projetos associados, orientadores, e coorientadores, decidiu abordar os seguintes tópicos:

1. Tabela de estudo das diferentes Tecnologias, com uma descrição da metodologia adotada;
2. Requisitos Funcionais e Não Funcionais;
3. Casos de Uso;
4. Mockups da Aplicação.

**Índice**

[Resumo 2](#_Toc131981418)

[1. Estudo das diferentes Tecnologias 3](#_Toc131981419)

[1.1. Alternativas Viáveis 4](#_Toc131981420)

[1.2. Avaliação e Decisão 5](#_Toc131981421)

[1.2.1. Desenvolvimento 5](#_Toc131981422)

[1.2.2. Conclusão 6](#_Toc131981423)

[2. Requisitos 7](#_Toc131981424)

[2.1. Funcionais 7](#_Toc131981425)

[2.2. Não Funcionais 8](#_Toc131981426)

[2.3. Casos de Uso 9](#_Toc131981427)

[3. Notas Finais 10](#_Toc131981428)

[Referências 11](#_Toc131981429)

**Índice de Tabelas**

[Tabela 1 - Comparação entre Frameworks possivelmente utilizadas para o Desenvolvimento 4](#_Toc131981409)

[Tabela 2 - Requisitos Funcionais 8](#_Toc131981410)

[Tabela 3 - Requisitos Não Funcionais 9](#_Toc131981411)

[Tabela 4 - Caso de Uso "Efetuar Login" 9](#_Toc131981412)

[Tabela 5 - Caso de Uso "Enviar Avaria" 10](#_Toc131981413)

**Índice de Figuras**

[Figura 1 - Mockup Geral 11](file:///C:\Users\franc\Desktop\Uni\LAB\Projeto\1-Relatorio-Progresso-smartwatch-FINAL.docx#_Toc132662526)

[Figura 2 - Página de Autenticação 11](file:///C:\Users\franc\Desktop\Uni\LAB\Projeto\1-Relatorio-Progresso-smartwatch-FINAL.docx#_Toc132662527)

[Figura 3 - Página de Alerta 11](file:///C:\Users\franc\Desktop\Uni\LAB\Projeto\1-Relatorio-Progresso-smartwatch-FINAL.docx#_Toc132662528)

[Figura 4 - Lista de Notificações 12](file:///C:\Users\franc\Desktop\Uni\LAB\Projeto\1-Relatorio-Progresso-smartwatch-FINAL.docx#_Toc132662529)

[Figura 5 - PopUp Notificação 12](file:///C:\Users\franc\Desktop\Uni\LAB\Projeto\1-Relatorio-Progresso-smartwatch-FINAL.docx#_Toc132662530)

[Figura 6 - Lista de Máquinas 12](file:///C:\Users\franc\Desktop\Uni\LAB\Projeto\1-Relatorio-Progresso-smartwatch-FINAL.docx#_Toc132662531)

[Figura 7 - Ecrã Inicial 12](file:///C:\Users\franc\Desktop\Uni\LAB\Projeto\1-Relatorio-Progresso-smartwatch-FINAL.docx#_Toc132662532)

[Figura 8 - Lista de Funcionários 12](file:///C:\Users\franc\Desktop\Uni\LAB\Projeto\1-Relatorio-Progresso-smartwatch-FINAL.docx#_Toc132662533)

[Figura 9 - Ecrã de Visualização 12](file:///C:\Users\franc\Desktop\Uni\LAB\Projeto\1-Relatorio-Progresso-smartwatch-FINAL.docx#_Toc132662534)

## Estudo das diferentes Tecnologias

Num desenvolvimento de uma aplicação para *wearables* (no nosso caso, um *smartwatch*) e para um *tablet,* todos os diferentes grupos experienciaram diferentes limitações e novos desafios, principalmente comparando com projetos e *softwares* desenvolvidos em Unidades Curriculares precedentes. Todos os seis membros, dos três diferentes pares, tiveram de discutir e reunir, apresentado diversas propostas com base na sua preferência pessoal, do que consideravam ser as melhores propostas para o desenvolvimento da aplicação em questão.

O grupo presente, tentando atender de melhor forma às requisições e restrições de cada grupo, tomou a diligência de procurar uma *Framework* que conseguisse a atender a três diferentes pontos, interligados e concorrentes, relacionados com a questão do projeto em si. Estas consistem em:

1. **Desenvolvimento multiplataforma:** este ponto encontra-se em primeiro lugar devido à sua relevância. Uma vez que temos de implementar soluções em diferentes *Hardwares/Dispositivos,* necessitamos de ter em conta a disponibilidade da ferramenta conseguir ter uma portabilidade fácil de efetuar, e, mais que tudo, eficiente.
2. **Design eficiente e personalizável:** devido à particularidade do problema (gestão de Sistemas de Manutenção e Avarias na Indústria 4.0), o grupo tem de dispor uma abordagem díspar, no que toca à fácil e simples utilização do software no wearable. Para além disso temos de considerar que todas as operações irão ser realizadas num ambiente diferente (devido ao tamanho e propriedades do *wearable*), e que essas mesmo deverão ser simples, rápidas de efetuar, e concisas. Isto leva-nos a um paradigma que até agora não experienciamos, e por esse mesmo motivo, a necessidade de uma ferramenta altamente personalizável impõe-se.
3. **Contemporaneidade e Comunidade Ativa:** em consequência dos últimos dois pontos, o grupo encontra-se assim numa situação ainda não experienciada. Por este mesmo motivo é imperativo a escolha de uma *Framework* ativa, com bastante documentação (para facilitar a autoaprendizagem) e que ofereça recursos, bibliotecas, etc. Para além disto, a contemporaneidade é relevante visto que, no contexto e importância de um projeto final, desejamos usar uma ferramenta que tenha utilidade para o nosso futuro profissional.

# Alternativas Viáveis

O grupo nesta subsecção pretende então, utilizando os parâmetros mencionados no ponto anterior, escolher uma *Framework* que seja viável para a execução do problema proposto, e no final do mesmo, apresentar uma escolha que seja bem fundamentada e coesa com os restantes dois subprojectos.

*A priori,* alguns nomes soantes, e devido à popularidade de utilização dos mesmos, foram logo mencionados:

1. React Native;
2. Flutter;
3. Android Native;
4. Xamarin;
5. Ionic;
6. PhoneGap;
7. MAUI.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | ANDK Kotlin | Flutter | React | Xamarin | ANDK Java |
| Linguagem | Kotlin | Dart | JavaScript | C# | Java |
| Ano Lançamento | 2011/2017 | 2017 | 2013/2015 | 2011 | 2008 |
| Nº de Utilizadores | |||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||| | ||||||||||||||||||||||||||||||||||||||||||||||||||| | ||||||||||||||||||||||||||||||||||||||||||||||||||| | ||||||| | ||||||||||||||||||||||||||||||| |
| Desempenho | |||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||| | |||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||| | ||||||||||||||||||||||||||||||| | |||||||||||||| | |||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||| |
| Ferramentas | |||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||| | ||||||||||||||||||||||||||||||| | ||||||||||||||||||||||||||||||||||||||||||||||||||| | |||||||||||||| | |||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||| |
| Customização | ||||||||||||||||||||||||||||||||||||||||||||||||||| | |||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||| | ||||||||||||||||||||||||||||||||||||||||||||||||||| | ||||||||||||||||||||||||||||||| | ||||||||||||||||||||||||||||||||||||||||||||||||||| |
| Suporte | |||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||| | ||||||||||||||||||||||||||||||||||||||||||||||||||| | ||||||||||||||||||||||||||||||||||||||||||||||||||| | ||||||| | |||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||| |
| Tamanho | Médio | Grande | Médio | Grande | Médio |

Tabela - Comparação entre Frameworks possivelmente utilizadas para o Desenvolvimento

* 1. Avaliação e Decisão
     1. Desenvolvimento

Neste momento, o grupo já possuí um leque de diferentes ferramentas que pode escolher, e por isso, passa-se assim à decisão da mais indicada, tendo em conta sempre, o problema em questão, e toda a informação disposta até este momento.

Em primeiro lugar, uns dos mais importantes fatores, é a sincronia da escolha de ambos os projetos de *Frontend.* Isto é necessário, porque assim a questão não passa só de “será que suporta ambos” ou “será que a portabilidade é um processo simples”, mas sim, “deve ser a mesma, para facilitar a implementação e a entreajuda entre os dois grupos”. Este fator, fez com que o leque fosse reduzido, especialmente no que toca ao uso de Xamarin para o desenvolvimento. As razões apresentadas pelo grupo do desenvolvimento do Tablet foram consideradas válidas, e por isso, assumindo as preocupações, especialmente relacionadas com o facto dessa *Framework* estar no processo de ser descontinuada, foi excluída. À sua sucessora MAUI, foi o oposto. Não se revelou uma opção viável, nem segura, visto que é uma *Framework* muito recente, e sem qualquer tipo de suporte, devido a isso mesmo. Passamos assim às restantes propostas.

O Kotlin, apareceu como uma linguagem promissora. Comparada com o seu precessor Java, a sua *learning curve,* é muito menor. Isto apelou inicialmente ao grupo. Para além disso, é suportada por muitos *IDEs,* e fácil de manter. Aqui não haveria problema da aprendizagem das diferentes funcionalidades de um diferente *IDE.* Porém, após alguma pesquisa e experienciação, o problema manteve-se: apesar de ser sucessora de Java*,* Kotlin continua a não ser Java e é necessária de bastante aprendizagem – o que se tornou extremamente difícil devido aos escassos recursos de aprendizagem.

O Flutter, em segundo lugar, é a linguagem que se apresentou como favorita, tanto devido à influência dos diferentes grupos. Tendo em conta, isto, será apresentada na mesma, com os seus prós e contras.

O Flutter, adquirida pela Google, ganhou significativamente, muitos utilizadores nos recentes anos. Apresenta-se na forma de um SDK (*Software Development Kit)*, que passa por uma *codebase,* e múltiplas aplicações. Um dos *features que* mais chamou à atenção no grupo, foi o *Hot-Reload,* que é auto-explicatório, mas ainda mais importante, o *Flutter App Builder,* que sai da abordagem de linguagens já anteriormente mencionadas, e passa para um *Low-Code approach.* Para além disto, o desenvolvimento de Flutter é muito intuitivo, especialmente para quem tem o seu primeiro contacto com, graças a outras funcionalidades (à exceção das já mencionadas), como o *Refactoring.*

Em terceiro lugar, mas não sem menos destaque, React. Este SDK, ou *Framework* tornou-se das principais do mundo do desenvolvimento de *Frontend,* desde a sua criação, há sensivelmente 9 anos atrás, graças à sua fácil utilização, rapidez e organização simplista. No entanto, no que toca ao desenvolvimento de aplicativos para *wearables,* o mesmo não se aplica. A linguagem ainda apresenta alguns tipos de limitações de processamento e memória, em comparação com outras linguagens nativas. O mesmo se aplica em termos de armazenamento. Contudo, nem tudo são negativos. Um dos aspetos mais atraentes de utilizar esta tecnologia advém do facto de ser uma biblioteca popular e muito bem documentada, podendo tornar o desenvolvimento mais fácil e acessível. Ainda no ponto que toca a acessibilidade e facilidade de utilização, devemos referir que através do React Native, o código pode ser compartilhado para diferentes plataformas.

* + 1. Conclusão

Chegando assim ao momento da decisão, conjuntamente com o grupo responsável pelo desenvolvimento da aplicação do Tablet, decidiu-se que a ferramenta a utilizar para o desenvolvimento do aplicativo do *wearable,* será o Flutter. Devido à sua grande comunidade e suporte, funcionalidades inovadoras, simplicidade e *minimalist approach,* destacou-se das demais. Para a virtualização do *wearable*, utilizar-se-á o Android Studio, com o respetivo *Toolchain.*

## Requisitos

Nesta secção o grupo de trabalho pretende descrever os Requisitos Funcionais e Não Funcionais. Em primeiro lugar, os funcionais, ir-se-ão referir às funcionalidades e *features* que um sistema possuí, ou vulgarmente dito como: “o que é que um sistema deve fazer”; por outro lado, os requisitos não funcionais referem-se a características gerais do sistema como um todo, como por exemplo, custos, confiabilidade, normas de proteção de dados, etc. Pode ser conhecido vulgarmente também, como: “como é que o sistema o deve fazer”.

Nota*: o grupo decidiu atribuir um atributo adicional “Prioridade” à tabela do ponto 2.1 para melhor definir a importância da implementação desse mesmo Requisito. Este tem três níveis, Baixo, médio e Alto, aqui descritos por ordem crescente de importância. No caso da tabela do ponto 2.2 o mesmo é substituído pela propriedade “Tipo”.*

# Funcionais

Na tabela seguinte, encontram-se os Requisitos Funcionais da aplicação a desenvolver:

|  |  |  |
| --- | --- | --- |
| Código | Descrição | Prioridade |
| RF [001] | A aplicação deverá permitir visualizar as linhas de montagem. | Média |
| RF [002] | A aplicação deverá permitir consultar/visualizar os funcionários da linha. | Baixa |
| RF [003] | A aplicação deverá permitir visualizar as máquinas da linha. | Média |
| RF [004] | A aplicação deverá permitir visualizar quem é o supervisor da linha. | Baixa |
| RF [005] | A aplicação deverá exibir a opção de avaria de máquina. | Alta |
| RF [006] | A aplicação deverá visualizar a receção de notificações de avarias. | Alta |
| RF [007] | A aplicação deverá exibir as notificações existentes. | Alta |
| RF [008] | A aplicação deverá exibir as avarias conforme a sua gravidade. | Alta |

Tabela - Requisitos Funcionais

# Não Funcionais

Na tabela seguinte, encontram-se os Requisitos Não Funcionais da aplicação a desenvolver:

|  |  |  |
| --- | --- | --- |
| Código | Descrição | Tipo |
| RNF [001] | A aplicação utiliza, para o desenho do *Frontend* o Flutter. | Arquitetura |
| RNF [002] | A aplicação é baseada na linguagem de programação Dart. | Arquitetura |
| RNF [003] | A aplicação é nativa no sistema operacional de WearOS. | Compatibilidade |
| RNF [004] | A aplicação está disponível na linguagem PT-PT. | *User-Interface* |
| RNF [005] | Todo o design da aplicação é baseado em Widgets. | Arquitetura |
| RNF [006] | A aplicação tem uma resolução de 450x450px. | *User-Interface* |

Tabela - Requisitos Não Funcionais

# Casos de Uso

Nas seguintes tabelas, encontram-se os Casos de Uso referentes à aplicação a desenvolver:

|  |  |
| --- | --- |
| Nome de caso de Uso | Efetuar Login |
| **ID** | UC1 |
| **Ator(es)** | Supervisor, Funcionário, Administrador |
| **Descrição** | Este caso de uso tem como objetivo fazer login do ator no sistema. |
| **Pré-Condições** | 1. O ator tem de estar registado no sistema |
| **Fluxo primário** | 1. O ator insere credenciais    1. Nome de Utilizador    2. Password   A1. O ator insere credenciais erradas.   1. O ator é autenticado com sucesso. |
| **Fluxo Secundário** | A1. O ator insere credenciais erradas.   1. Volta ao 1º passo do fluxo primário sempre que o ator introduzir as credenciais erradas. 2. O sistema apresenta uma mensagem de que a autenticação não foi realizada com sucesso. |
| **Pós-Condições** | 1. O ator fica autenticado. 2. O ator é redirecionado para ecrã principal/notificações. |

Tabela - Caso de Uso "Efetuar Login"

|  |  |
| --- | --- |
| Nome de caso de Uso | Enviar avaria |
| **ID** | UC2 |
| **Ator(es)** | Funcionário, Administrador |
| **Descrição** | Este caso de uso tem como objetivo enviar notificação de avaria. |
| **Pré-Condições** | 1. O utilizador tem de estar registado no sistema. 2. O utilizador tem de estar autenticado. 3. O utilizador tem de estar associado a uma linha. |
| **Fluxo primário** | 1. O utilizador direciona-se ao ecrã de avaria. 2. O utilizador seleciona a opção de avaria. 3. O sistema envia notificação ao Administrador da linha.    1. O Administrador recebe a notificação. |
| **Fluxo Secundário** | Não existe. |
| **Pós-Condições** | 1. O ator é redirecionado para o ecrã principal/notificações. |

Tabela - Caso de Uso "Enviar Avaria"

INSERIR AQUI OS RESTANTES CASOS DE USO

## Mockups

Nesta secção os elementos do grupo presente, apresentam assim os protótipos visuais ou mockups, de forma a se obter uma visualização do que a aplicação na semelhança deverá possuir. Cada ilustração corresponde a uma janela de visualização diferente.

*Nota: Apesar da aplicação real ser desenvolvida numa janela circular, todos os mockups apresentam-se com a forma quadrada devido a limitações impostas pelo software utilizado para a realização dos mesmos. Para além disto, estes são animados e por isso, para melhor condições de visualização recomenda-se vivamente a utilização da aplicação Figma.*

Uma imagem com Website

Descrição gerada automaticamente

Figura - Mockup Geral

Uma imagem com texto

Descrição gerada automaticamente

Uma imagem com texto, eletrónica, captura de ecrã

Descrição gerada automaticamente

Figura - Página de Alerta

Figura - Página de Autenticação

Uma imagem com texto

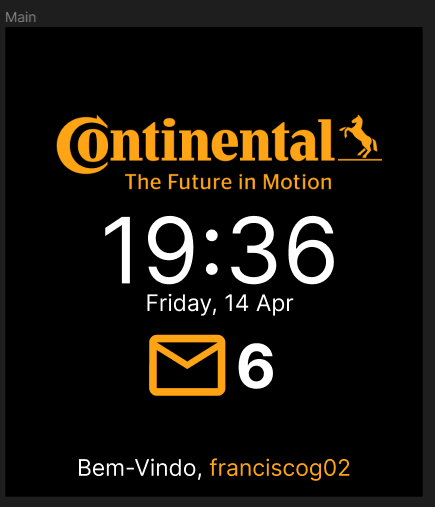
Descrição gerada automaticamente

Uma imagem com texto, marcador

Descrição gerada automaticamente

Figura - Lista de Notificações

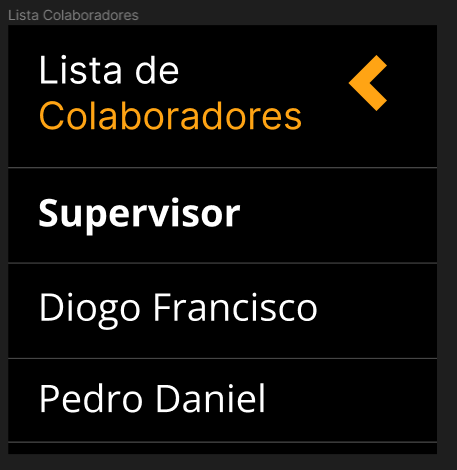
Figura - PopUp Notificação

Uma imagem com texto, marcador

Descrição gerada automaticamente

Figura - Lista de Máquinas

Figura - Ecrã Inicial

Uma imagem com texto, marcador

Descrição gerada automaticamente

Figura - Lista de Funcionários

Figura - Ecrã de Visualização

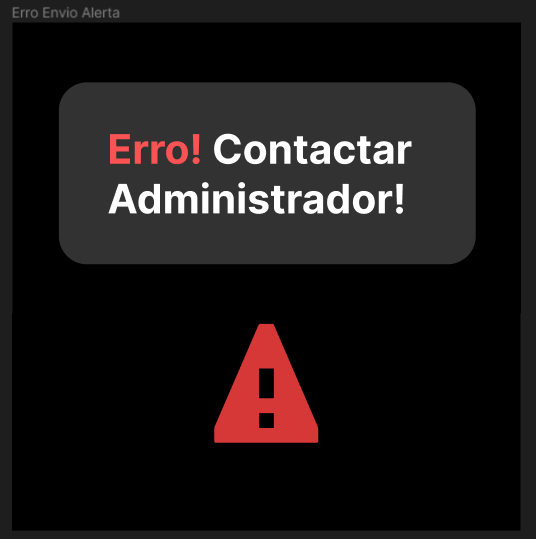


Figura - Confirmação Envio Alerta

Figura - Erro Envio Alerta

## Fase de Implementação

Uma imagem com texto, eletrónica

Descrição gerada automaticamenteUma imagem com texto

Descrição gerada automaticamenteUma imagem com texto, eletrónica

Descrição gerada automaticamenteO presente grupo então passa assim para a fase da implementação. Devido à necessidade de coordenação entre os diferentes grupos, decidiu-se assim entre os dois membros integrantes começar-se a implementar diferentes elementos na forma de *widgets* que mais tarde serão requeridos para a implementação da aplicação com os dados já a serem aplicados e recebidos. Para isto o discente Francisco Gouveia começou por criar um projeto de teste, denominado “*Flutter\_teste”,* onde tentou a implementação de um relógio mundial com data, funcionalidades de *scroll,* utilização de ícones, layouts, e por último, mas não menos importante, paginação e *routing.*

Figura - Testagem de funcionalidades num projeto de Teste

## Notas Finais

Nesta etapa os elementos do grupo consideram que atingiram os objetivos propostos pelos Coordenadores e Docentes envolvidos de forma satisfatória. Gostaríamos também de destacar a importância que esta etapa toma, no desenvolvimento do software, e boa continuação do processo geral do planeamento do mesmo.

## Referências

1. *bevkoski/react-native-android-wear-demo: Showcase of a two-way communication between React Native and Android Wear apps.* (n.d.). Retrieved March 25, 2023, from <https://github.com/bevkoski/react-native-android-wear-demo>
2. *Flutter vs React Native em 2022: comparativo - Blog UDS*. (n.d.). Retrieved March 25, 2023, from <https://uds.com.br/blog/flutter-vs-react-native/>
3. *Flutter vs React Native vs Native: comparação de desempenho aprofundada | by Rodney Neville | Medium*. (n.d.). Retrieved March 25, 2023, from <https://rodneyneville.medium.com/flutter-vs-react-native-vs-native-compara%C3%A7%C3%A3o-de-desempenho-aprofundada-156f9a6f0bd9>
4. *What is Xamarin? Xamarin vs Native App Development | AltexSoft*. (n.d.). Retrieved April 17, 2023, from <https://www.altexsoft.com/blog/mobile/pros-and-cons-of-xamarin-vs-native/>
5. *Advantages and Disadvantages of Kotlin - Tech Quintal*. (n.d.). Retrieved April 17, 2023, from <https://www.techquintal.com/advantages-and-disadvantages-of-kotlin/>
6. *Pros and Cons of Kotlin for Android App Development | by Sandeep Agarwal | Quick Code | Medium*. (n.d.). Retrieved April 17, 2023, from <https://medium.com/quick-code/pros-and-cons-of-kotlin-for-android-app-development-c4b0f95c1324>
7. *Flutter vs. React Native: Which is Better for 2023? | SaM Solutions*. (n.d.). Retrieved April 17, 2023, from <https://www.sam-solutions.com/blog/flutter-vs-react-native/>
8. *Flutter Pros & Cons - Should You Use It In Your Project? - INVO Blog*. (n.d.). Retrieved April 17, 2023, from <https://invotech.co/blog/flutter-pros-cons-should-you-use-it-in-your-project/>