



UNIVERSIDADE FEDERAL RURAL DE PERNAMBUCO  
PRÓ-REITORIA DE PÓS-GRADUAÇÃO  
PROGRAMA DE PÓS-GRADUAÇÃO PROFISSIONAL EM ENSINO DE FÍSICA  
MESTRADO NACIONAL PROFISSIONAL EM ENSINO DE FÍSICA  
POLO 58 - UFRPE

Marcos Felipe Mendonça de Neri

## **PRODUTO EDUCACIONAL**

**GAMIFICAÇÃO NO ENSINO DA FÍSICA: O USO DOS JOGOS ELETRÔNICOS  
NO PROCESSO DE ENSINO-APRENDIZAGEM NA EDUCAÇÃO BÁSICA.**

Recife  
Janeiro de 2023

GAMIFICAÇÃO NO ENSINO DA FÍSICA: O USO DOS JOGOS ELETRÔNICOS NO  
PROCESSO DE ENSINO-APRENDIZAGEM NA EDUCAÇÃO BÁSICA.

Este produto educacional é parte integrante da dissertação: GAMIFICAÇÃO NO ENSINO DA FÍSICA: O USO DE JOGOS ELETRÔNICOS NO PROCESSO DE ENSINO-APRENDIZAGEM NA EDUCAÇÃO BÁSICA, desenvolvida no âmbito do Programa de Mestrado Nacional Profissional em Ensino de Física, polo 58 – UFRPE, como parte dos requisitos necessários à obtenção do título de Mestre em Ensino de Física.

Orientadora:  
Profa. Dra. Sara Cristina Pinto Rodrigues

Recife  
Janeiro de 2023

## AGRADECIMENTOS

Agradeço a Deus pelas pessoas que colocou no meu caminho e me permitiram alcançar este objetivo;

À minha Família, em especial a minha esposa Janice, que divide comigo os fardos advindos deste trabalho, com muita paciência, e em nome desta agradeço aos demais familiares que me apoiaram neste intento;

À Tenente Coronel Sávia, comandante do Colégio da Polícia Militar de Pernambuco (CPM), em nome de quem agradeço a toda comunidade escolar, de quem sempre tive desde o início, o apoio para a realização deste trabalho, dentre os quais destaco os colegas Nívio Bernardo, Leandro de Freitas e Isabelle França, que mais do que apoio, me deram incentivo para começar, suporte na realização e auxílio na conclusão;

À Sociedade Brasileira de Física (SBF) e a Universidade Federal Rural de Pernambuco (UFRPE) pela promoção e realização do curso através da gestão do programa Mestrado Nacional Profissional em Ensino de Física;

A todos os professores do programa, pelo conhecimento compartilhado, que muito contribuíram no meu desenvolvimento profissional. Em especial agradeço a Profa. Dra. Sara Cristina Pinto Rodrigues, pela sua parceria na inestimável orientação durante a elaboração deste trabalho;

E finalmente a todos que, de alguma forma, contribuíram para a execução deste trabalho;

O presente trabalho foi realizado com o apoio da Coordenação de Aperfeiçoamento de Pessoal de Nível Superior – Brasil (CAPES) – código de financiamento 001.

# Sumário

1. Apresentação	1
2. Objetivo	1
3. Descrição da Construção do Produto	2
3.1 - Telas do Jogo	8
3.1.1 - Tela de Inicialização	8
3.1.2 - Tela de Atualização	10
3.1.3 - Tela de Identificação de Usuário	12
3.1.4 - Tela de Ranking	15
3.1.5 - Tela de Seleção de Nível	18
3.1.6 - Tela de Questão	21
3.1.7 - Tela de Acerto	26
3.1.8 - Tela de Erro	27
3.1.9 - Tela De Vitória	29
3.1.10 - Tela de Créditos	31
4. Regras do Jogo	32
5. Descrição da Aplicação do Produto	33
6. Considerações Finais	34
7. Referências	36
Apêndice 1 - Questionário Preliminar	37

# 1. Apresentação

O produto educacional em tela neste trabalho pretende agir como facilitador de troca de significados entre estudantes e professores no que se refere aos signos relacionados aos conceitos iniciais no estudo da cinemática, e logicamente, não pretende passar por uma solução definitiva, e sim uma alternativa em meio às dificuldades encontradas pelos educadores no atual cenário da educação.

A Base Nacional Comum Curricular (BNCC) dentre os deveres elencados diz que:

“...a área de Ciências da Natureza, por meio de um olhar articulado de diversos campos do saber, precisa assegurar aos alunos do Ensino Fundamental o acesso à diversidade de conhecimentos científicos produzidos ao longo da história...”(Brasil, 2013).

Fica a cargo do educador as estratégias para atingir tal finalidade, e é neste contexto que o Produto Educacional, DinamicaMente, desenvolvido como um jogo eletrônico de perguntas e respostas mediadas por dicas contextualizadas, busca, de forma lúdica e paralela a metodologia usada em sala de aula, auxiliar na construção de um vocabulário comum entre os discentes e o docente.

O jogo deve ser aplicado às turmas em que estão sendo introduzidos os conceitos iniciais de cinemática, e sua aplicação pode anteceder, ou ser realizada concomitante às aulas desse conteúdo, podendo ser utilizado no ensino fundamental ou médio.

A seguir apresentaremos o objetivo, a descrição da construção do produto, as regras do jogo, a aplicação do produto e por fim as considerações finais deste trabalho.

## 2. Objetivo

O aplicativo DinamicaMente, aqui apresentado, é um jogo eletrônico exclusivo para dispositivos Android® desenvolvido sobre plataformas gratuitas e baseado em teorias de aprendizagem estabelecidas, que tem por objetivo facilitar o aprendizado dos conceitos iniciais de cinemática, servindo como mediador de significados entre os signos que representam estes conceitos efetivamente, e os preexistentes na mente do estudante.

### 3. Descrição da Construção do Produto

O aplicativo exclusivo para dispositivos Android®, nomeado de DinamicaMente, é um jogo eletrônico de perguntas e respostas, constituído por algumas telas que interagem permitindo um retorno ao usuário, que o guia a respostas corretas nas próximas interações. A Figura 3.1 mostra algumas telas do aplicativo.

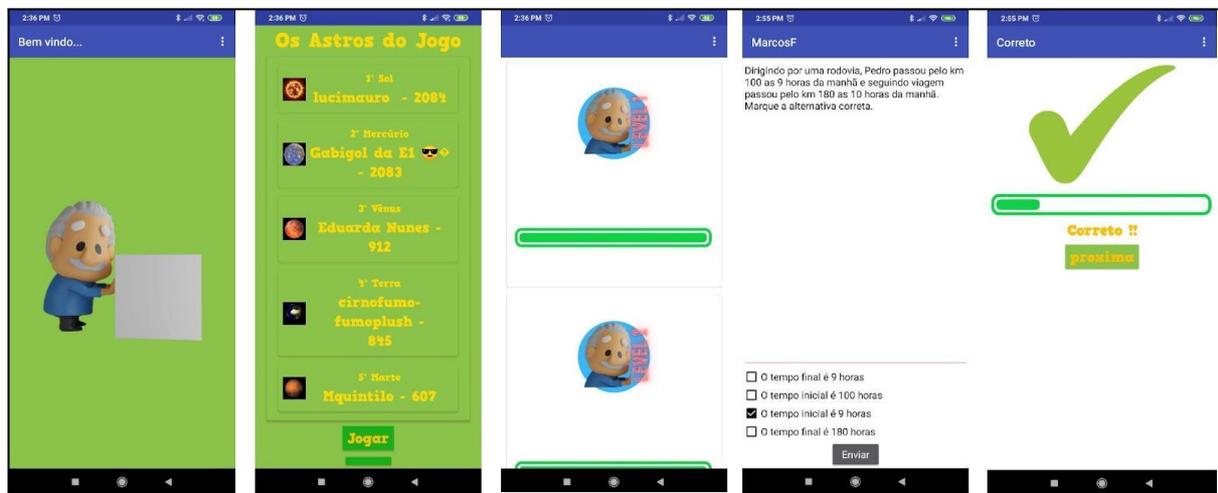


Figura 3.1- Algumas capturas de tela do Aplicativo DinamicaMente (Elaborado pelo autor, 2022).

Na construção do aplicativo DinamicaMente, foram usados *softwares* e serviços de uso gratuito, de modo que o fator financeiro não se torne um impeditivo à sua execução por parte de nenhum interessado em usá-lo. Tendo em vista esse princípio foi escolhido como plataforma para o desenvolvimento do aplicativo a ferramenta No-Code on-line KODULAR (<https://www.kodular.io/>), que permite criar as telas e definir o funcionamento de um aplicativo sem a necessidade do usuário conhecer linguagens de programação, apenas lógica. Com a finalidade de guardar os dados usamos localmente o TinyDB (disponível no ambiente do Kodular) e na nuvem o FIREBASE (<https://firebase.google.com/>), nas condições em que não são gerados custos. Na edição de Imagens foram usados o BLENDER (<https://www.blender.org/>) e o CANVA ([https://www.canva.com/pt\\_br/](https://www.canva.com/pt_br/)), também na modalidade gratuita.

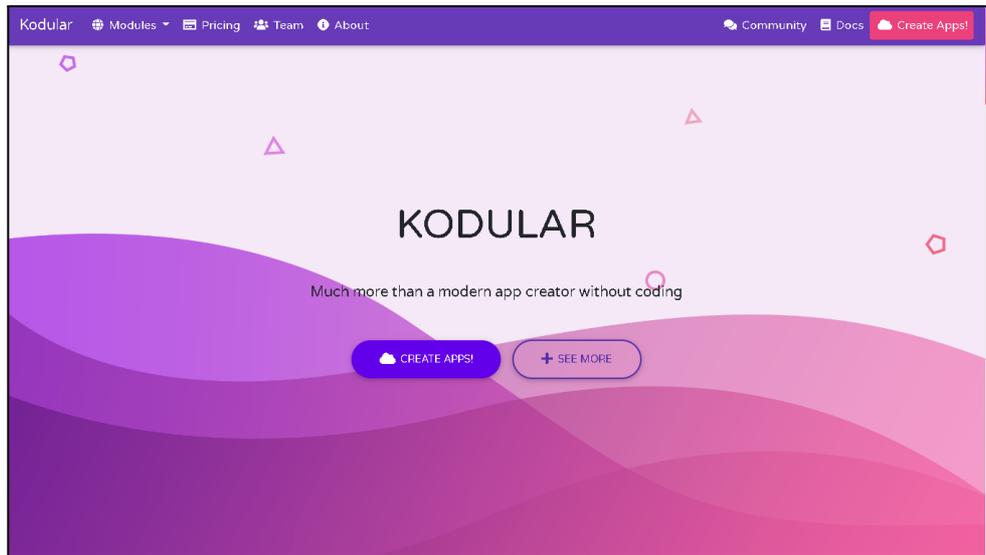


Figura 3.2 - Tela inicial do Kodular ( Captura de tela).

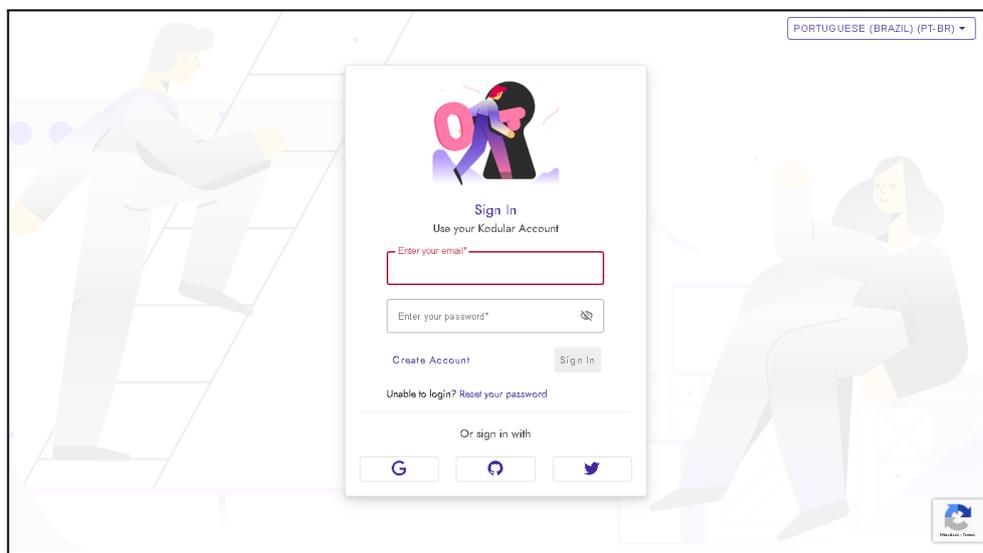


Figura 3.3 - Tela de Login do Kodular (Captura de tela).

Para iniciar usando o Kodular se faz necessário criar uma conta no serviço, acessando o site <https://www.kodular.io/> , ver Figura 3.2. Clicando no botão “create app!” será apresentada a tela observada na Figura 3.3, onde o usuário poderá criar novas credenciais de acesso ao sistema Kodular ou usar suas credenciais de outros serviços, como Google, Github ou Twitter para autenticar-se no serviço do Kodular, bastando seguir as orientações apresentadas ao clicar no botão correspondente ao serviço de autenticação desejado na parte inferior do formulário. Após credenciar seu usuário será apresentada a tela inicial da Figura 3.4, onde poderá ser iniciado um novo projeto ou abrir um antigo caso já tenha algum.

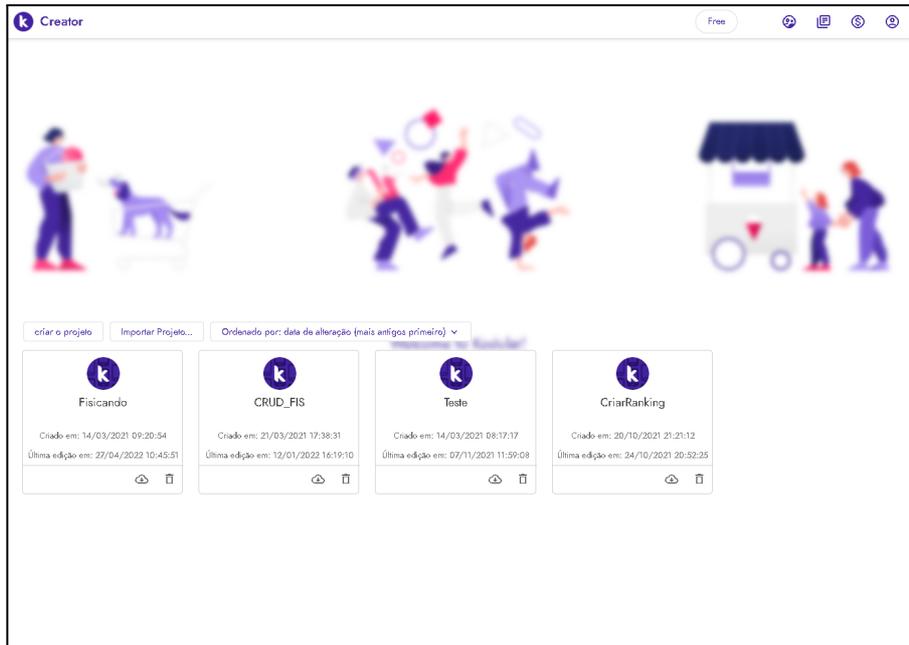


Figura 3.4 - Ambiente do Kodular após o Login (Captura de tela).

O Firebase deve ser acessado no endereço eletrônico <https://firebase.google.com/>, conforme Figura 3.5, clicando no botão “ir ao console” na parte superior direita da tela o usuário poderá logar com sua conta do Google, alcançando a tela observada na Figura 3.6, onde poderá iniciar um novo projeto.

No projeto do App DinamicaMente foi usado o Realtime Database como tipo de repositório dos dados *online* dentro do mecanismo do Firebase. As estruturas e o modo de acesso e uso desta plataforma podem ser aprendidos através dos mesmos *links* que indicaremos adiante para o aprendizado do Kodular.

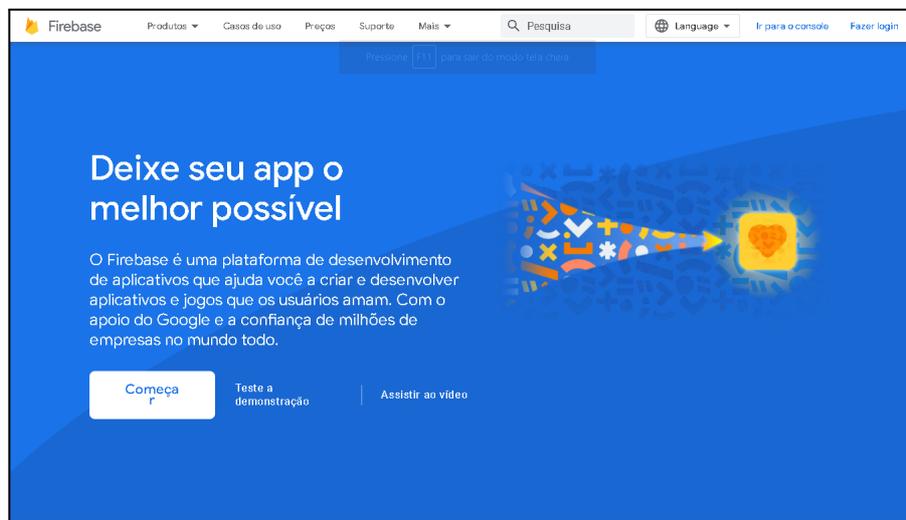


Figura 3.5 - Tela inicial do Firebase ( Captura de tela).

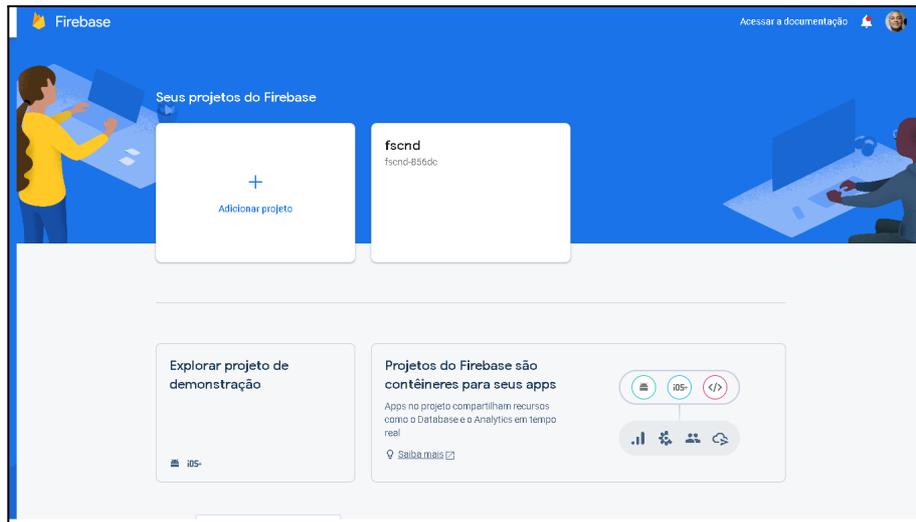


Figura 3.6 - Tela do Firebase após logado (Captura de tela).

Há de se esclarecer que não existe pretensão neste trabalho oferecer um curso de desenvolvimento de aplicações, apenas expor, para quem for habilitado e quiser recriar ou modificar o que foi realizado na confecção do aplicativo DinamicaMente, entretanto existem diversos tutoriais que podem ajudar a adquirir estas competências, como por exemplo:

- Programação em Blocos Brasil

<<https://www.youtube.com/watch?v=z64x8BxlE6o&list=PLSaHQQBC7dhrQ5WHzgDaZrJ6Js9dbcj5C>>;

- Robotizando

<<https://www.youtube.com/watch?v=LBhKuYyM6nM&list=PLe6v9G9A83kNcsNJqGlsBMTIoTDBIWnly>>;

- Deivid Rothen

<[https://www.youtube.com/watch?v=brZdGvJGsTI&list=PLgIw6\\_M7v8vrWYm-J5g2uKW12JF13tCBh](https://www.youtube.com/watch?v=brZdGvJGsTI&list=PLgIw6_M7v8vrWYm-J5g2uKW12JF13tCBh)>.

Primeiramente, analisaremos a interface de desenvolvimento do Kodular, a fim de que seja compreendido o seu funcionamento. Partindo da tela apresentada na Figura 3.4, teremos a caixa de diálogo mostrada na Figura 3.7, onde será nomeado o novo projeto. Após esse passo será apresentada a tela da Figura 3.8, para a configuração de parâmetros iniciais do aplicativo, como o nome, tema de cores e versões do Android® compatíveis. Reiterando que apenas esse sistema operacional é compatível com aplicações criadas na plataforma Kodular.

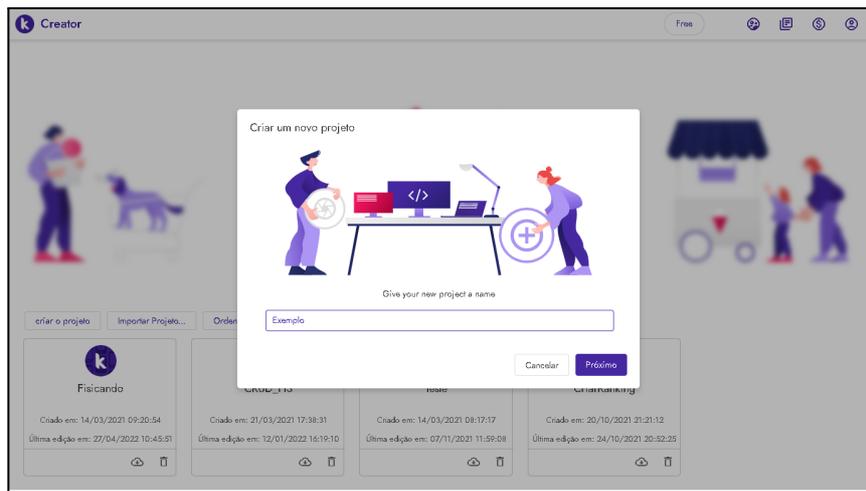


Figura 3.7 - Iniciando um novo projeto ( Captura de tela).

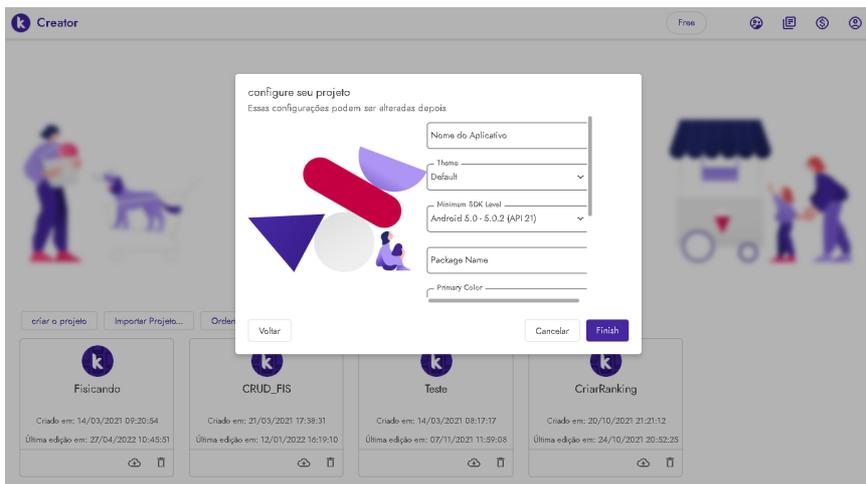


Figura 3.8 - Configurações iniciais ( Captura de tela).

Uma vez clicado no botão “Finish” , ou “Finalizar”, caso sua versão esteja configurada para a língua portuguesa, será apresentada a tela da Figura 3.9, onde destacamos algumas regiões conforme segue:

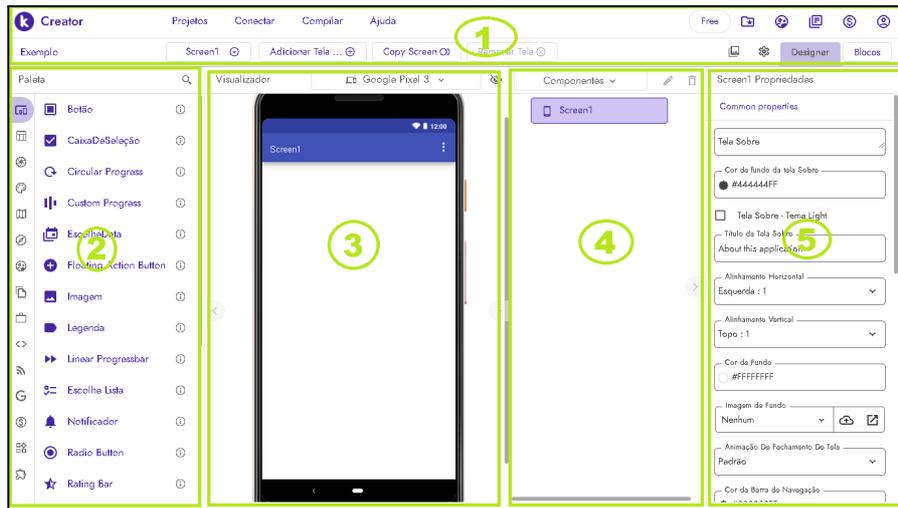


Figura 3.9 - Tela de Designer separada por regiões (Captura de tela editada pelo autor, 2022).

Região 1 - Menu superior onde podemos acessar as configurações e alternar entre os modos de “Designer” e programação com “Blocos”;

Região 2 - Paleta de componentes, onde podemos escolher os elementos que comporão a tela em construção e arrastá-los para o visualizador de componentes;

Região 3 - Visualizador de componentes, onde visualizamos a aparência final que tomará a tela do aplicativo;

Região 4 - Lista de componentes, onde são encadeados os componentes selecionados, permitindo selecionar e configurar algumas propriedades;

Região 5 - Propriedades Comuns, onde são apresentadas as propriedades dos elementos selecionados na lista de componentes.

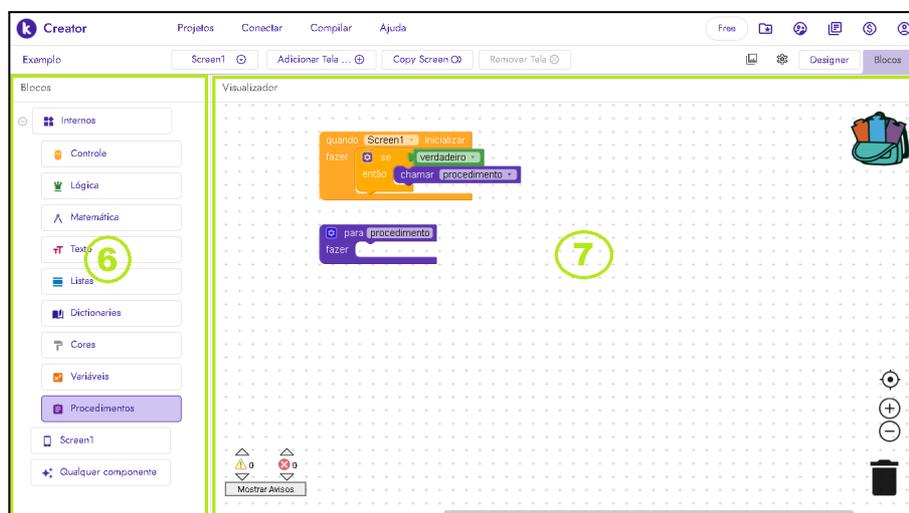


Figura 3.10 - Tela de programação com blocos com regiões identificadas (Captura de tela editada pelo autor, 2022).

Além do modo Designer, a interface do Kodular tem o modo Blocos, que é acessado na aba à direita do menu superior, de modo que a tela fica como visto na Figura 3.10, com as seguintes regiões:

Região 6 - Conjunto de Blocos, onde são exibidos os blocos de programação disponíveis para a aplicação;

Região 7 - Visualizador do algoritmo, onde são montados os blocos de codificação lógica que regem o comportamento dos elementos do aplicativo.

## **3.1 - Telas do Jogo**

O jogo é composto de um conjunto de telas que se sucedem de modo coordenado através da programação de seus componentes. Seguiremos explicando o funcionamento de cada tela com seus respectivos componentes e o diagrama de blocos associado, o que permitirá uma compreensão mais detalhada do funcionamento do aplicativo como um todo, levando em consideração que o leitor já tenha se apropriado do conhecimento necessário à esta análise.

### **3.1.1 - Tela de Inicialização**

A “Tela de Inicialização” dura poucos segundos durante a inicialização do aplicativo e é responsável por verificar se a versão do banco de questões do aplicativo está atualizada.

Na Figura 3.11 pode-se verificar os componentes utilizados na construção desta tela, conforme lista que segue:

- 1 componente da paleta Mídia - Imagem1;
- 1 componente da paleta Google - Firebase\_DB1;
- 1 componente da paleta Armazenamento - TinyDB1.

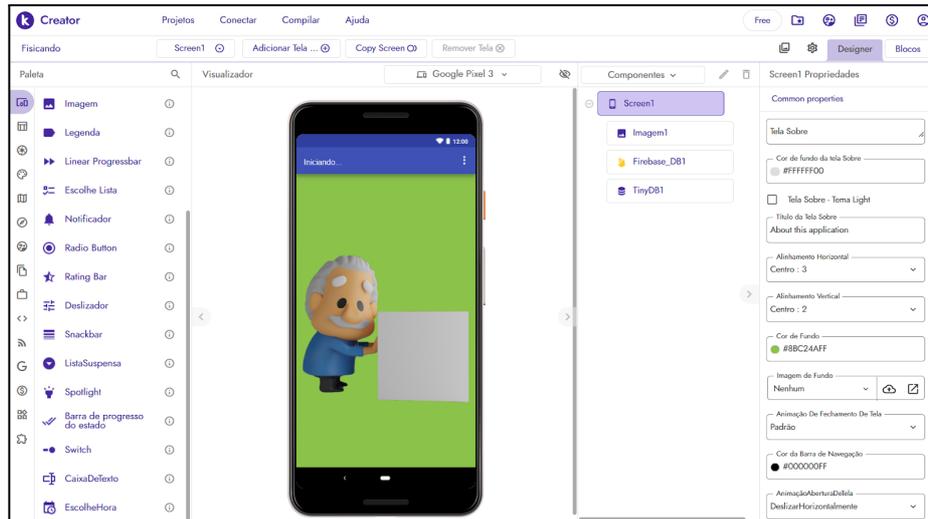


Figura 3.11 - Design da Tela de inicialização (Captura de Tela pelo autor, 2022).

O funcionamento desta tela quando ativa se resume a:

- carregar uma imagem;
- escrever uma mensagem de bem vindo na barra de título;
- verificar o nível do jogador e a versão atual do banco de questões na base online;
- verificar a versão do banco de questões da base local;
- comparar as versões dos bancos de questões;
- abrir a tela de atualização se for um usuário novo ou se as versões forem diferentes;
- abrir a tela “Ranking” caso as versões sejam iguais;
- fecha o aplicativo caso seja clicado o botão voltar do sistema durante sua execução.

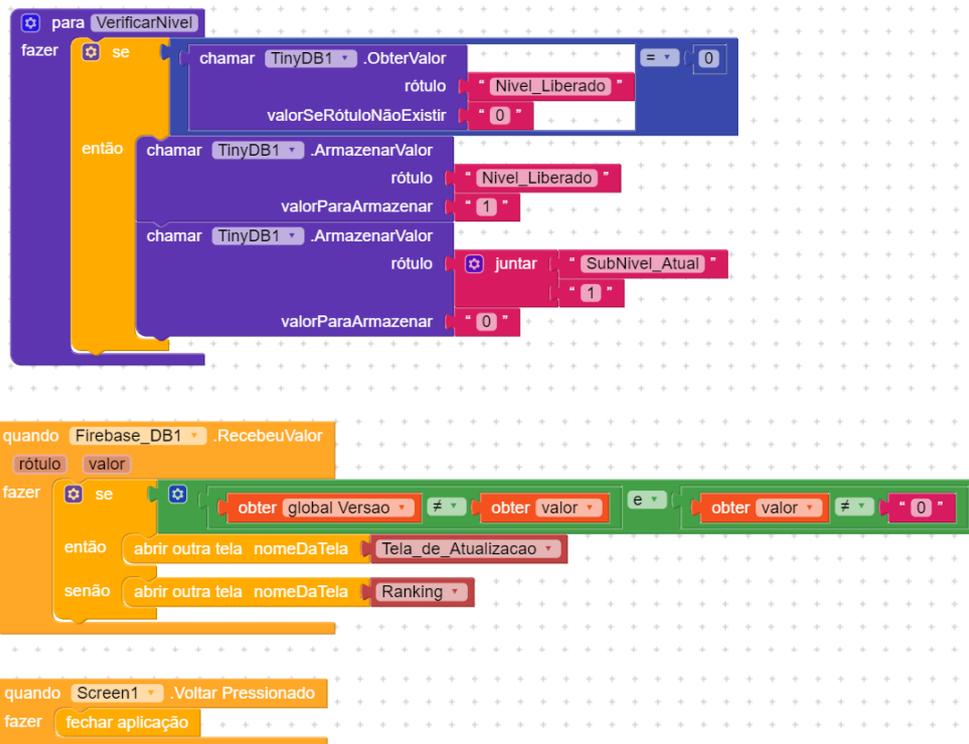
A programação em blocos correspondentes a lista descrita acima segue abaixo:

```

inicializar global Atualizar_versao para falso
inicializar global Versao para ""

quando Screen1 .Inicializar
fazer
ajustar Screen1 .Titulo para " Bem vindo..."
chamar VerificarNivel
chamar Firebase_DB1 .ObterValor
rótulo Versao
valorSeRótuloNãoExistir 0
ajustar global Versao para chamar TinyDB1 .ObterValor
rótulo Versao
valorSeRótuloNãoExistir ""

```



### 3.1.2 - Tela de Atualização

A “Tela de Atualização” ocorre apenas se for o primeiro acesso ao aplicativo ou se tiver ocorrido uma atualização no banco de questões online do aplicativo, e dura uma fração de segundos, a depender da conexão com a internet.

Na Figura 3.12 pode-se verificar os componentes utilizados na construção desta tela, conforme lista que segue:

- 1 componente da paleta Interface com o usuário - Legenda1;
- 1 componente da paleta Interface com o usuário - caixaDeTexto1;
- 1 componente da paleta Google - Firebase\_DB1;
- 1 componente da paleta Armazenamento - TinyDB1.

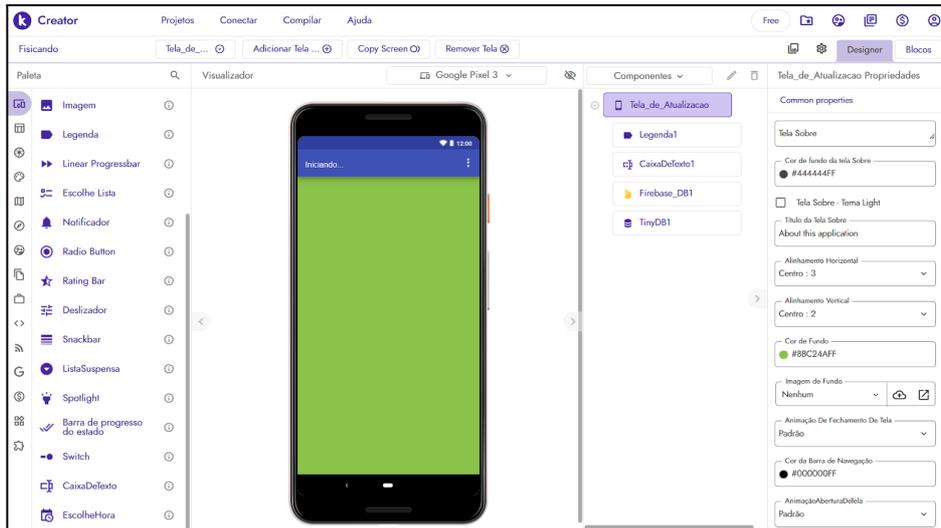


Figura 3.12 - Design da Tela de Atualização (Captura de Tela pelo autor, 2022).

O funcionamento desta tela quando ativa se resume a:

- escrever a mensagem “Atualizando...” na barra de título;
- obter a lista de tags da base de dados online;
- obter o valor de cada registro das lista de tags da base de dados online;
- atualizar cada registro da base local;
- escrever na caixa de texto o nome de cada registro atualizado;
- abrir a “Tela de Identificação de Usuário” quando terminar a atualização;
- fecha o aplicativo caso seja clicado o botão voltar do sistema durante sua execução.

A programação em blocos correspondentes a lista descrita acima segue abaixo:

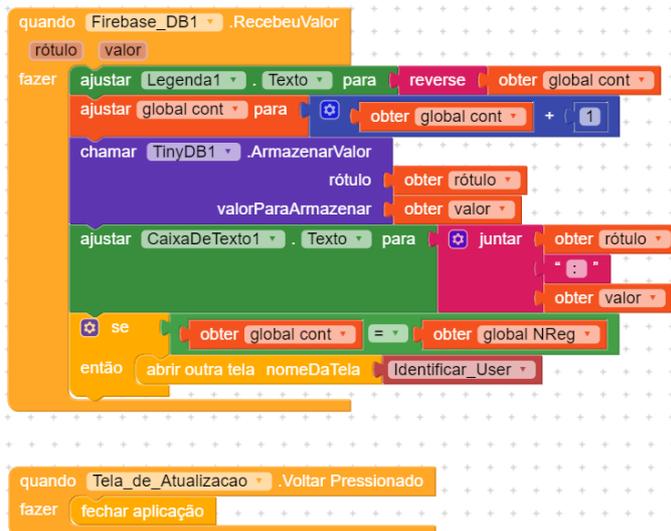
```

inicializar global NReg para 0

inicializar global cont para 0

quando Tela_de_Atualizacao .Inicializar
fazer
  ajustar Tela_de_Atualizacao . Título para "Atualizando..."
  chamar Firebase_DB1 .Obter lista de tags

quando Firebase_DB1 .Lista de Tags
  valor
  fazer
    ajustar global NReg para comprimento da lista lista obter valor
    para cada item na lista obter valor
    fazer
      chamar Firebase_DB1 .ObterValor
      rótulo obter item
      valorSeRótuloNãoExistir " "
  
```



### 3.1.3 - Tela de Identificação de Usuário

A “Tela de Identificação de Usuário” ocorre apenas se for o primeiro acesso ao aplicativo ou se tiver ocorrido uma atualização no banco de questões online do aplicativo, e verifica se o usuário está habilitado, registrando em seguida o “Nome de Jogador” escolhido por ele.

Na Figura 3.13 pode-se verificar os componentes utilizados na construção desta tela, conforme lista que segue:

- 5 componentes da paleta Layout - OrganizadorVertical1, OrganizadorHorizontal1, OrganizadorHorizontal2, Espaço1 e Espaço2;
- 1 componente da paleta Mídia - Imagem1;
- 6 componentes da paleta Interface com o usuário - Lb\_Usuario, Tb\_Usuario, CaixaDeTexto1, Bt\_Verificar, Bt\_Salvar e Notificador1;
- 1 componente da paleta Google - Firebase\_DB1;
- 1 componente da paleta Armazenamento - TinyDB1.

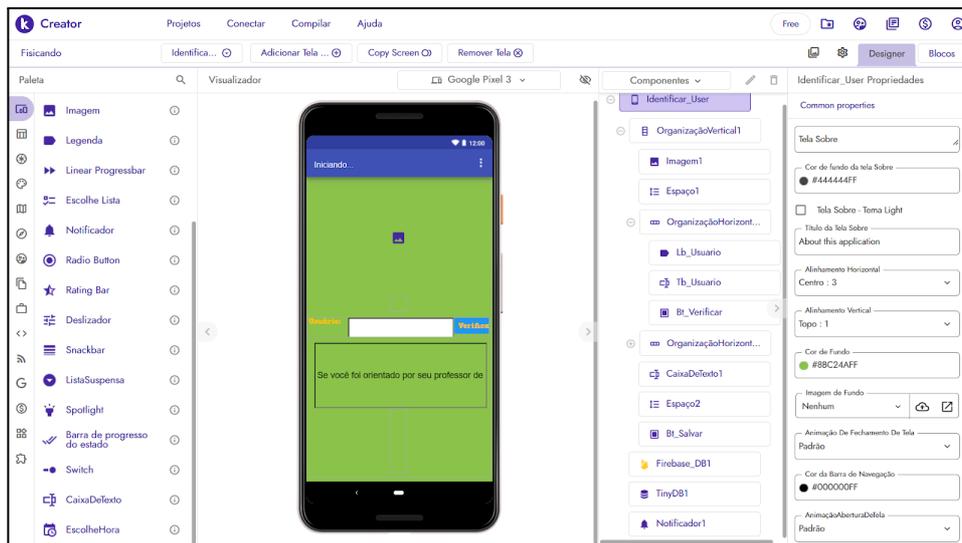


Figura 3.13 - Design da tela de identificação de usuário (Captura de Tela pelo autor, 2022).

O funcionamento desta tela é descrito a seguir:

- verifica se há usuário logado;
- abrir a tela “Ranking” caso exista usuário logado ou resgata a lista de usuários habilitados caso o usuário não esteja logado;
- verifica se o texto digitado na caixa de texto corresponde a algum usuário habilitado quando o botão verificar for pressionado;
- caso o texto não corresponda a algum usuário na lista ou a caixa de texto estiver vazia, abre notificador com a mensagem ”Usuário não encontrado”;
- caso o texto corresponda a algum usuário na lista, habilita caixa para digitar o “Nome de Jogador” e o botão “Salvar”;
- quando o botão “Salvar” é pressionado salva os dados do jogador no banco de dados local, no banco de dados online e abre a tela “Ranking” ;
- caso a caixa de texto esteja vazia ao pressionar o botão “Salvar”, será aberto um notificador com a mensagem “Informe um nome de jogador válido!”;
- fecha o aplicativo caso seja clicado o botão voltar do sistema durante sua execução.

A programação em blocos correspondentes a lista descrita acima segue abaixo:

inicializar global P\_Ranking para criar lista vazia

inicializar global User\_Status para criar lista vazia

inicializar global UsuarioLogado para criar lista vazia

inicializar global UsuarioExiste para verdadeiro

inicializar global Users\_List para criar lista vazia

```

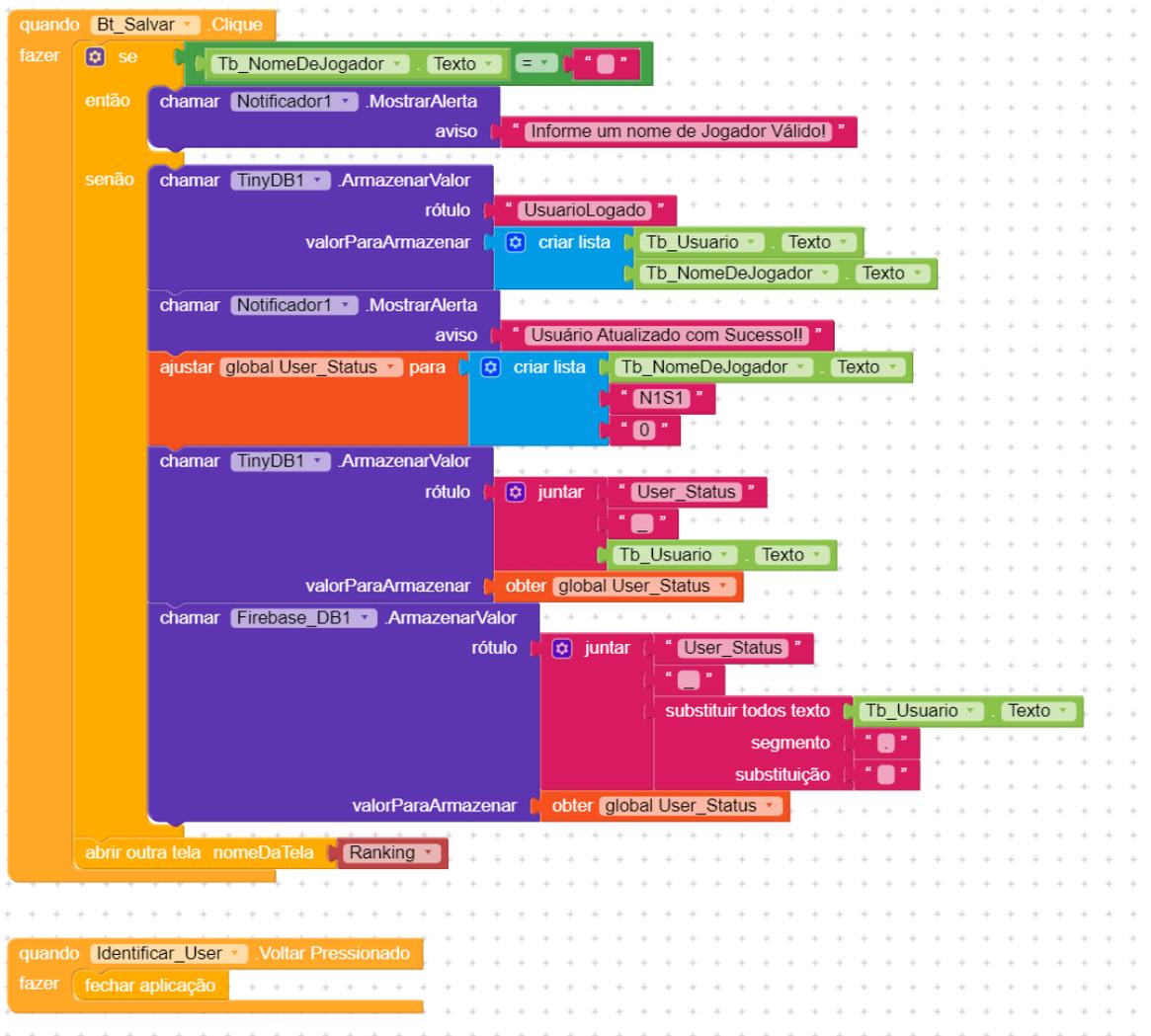
quando Identificar_User .Inicializar
fazer
  ajustar global UsuarioLogado para chamar TinyDB1 .ObterValor
  rótulo "UsuarioLogado"
  valorSeRótuloNãoExistir criar lista vazia
  se é lista vazia? lista obter global UsuarioLogado
  então
    ajustar global Users_List para chamar TinyDB1 .ObterValor
    rótulo "Users_List"
    valorSeRótuloNãoExistir criar lista vazia
  senão
    abrir outra tela nomeDaTela Ranking

```

```

quando Bt_Verificar .Clique
fazer
  se está na lista? item Tb_Usuario . Texto
  lista obter global Users_List
  então
    ajustar Bt_Verificar . Visível para falso
    ajustar Tb_Usuario . Ativado para falso
    ajustar Bt_Salvar . Visível para verdadeiro
    ajustar OrganizaçãoHorizontal1_copy . Visível para verdadeiro
  senão
    chamar Notificador1 .MostrarAlerta
    aviso "Usuário não encontrado"

```



### 3.1.4 - Tela de Ranking

A “Tela Ranking” mostra uma lista com os cinco jogadores melhor colocados no jogo, oferecendo as opções de jogar ou sair do jogo, havendo processos para a atualização do ranking em segundo plano ao iniciar, ou ao deslizar o dedo sobre a tela.

Na Figura 3.14 pode-se verificar os componentes utilizados na construção desta tela, conforme lista que segue:

- 12 componentes da paleta Layout - 1 `Swipe_Refresh_Layout`, 1 `OrganizadorVerticalRolavel`, 5 `OrganizadorVertical` e 5 `OrganizadorHorizontal`;
- 5 componentes da paleta Mídia - 5 `Imagens`;
- 13 componentes da paleta Interface com o usuário - 11 `Legendas`, 2 `botões`;
- 5 componentes dinâmicos - 5 `Card_View`;

- 1 componente da paleta Google - Firebase\_DB1;
- 1 componente da paleta Armazenamento - TinyDB1.

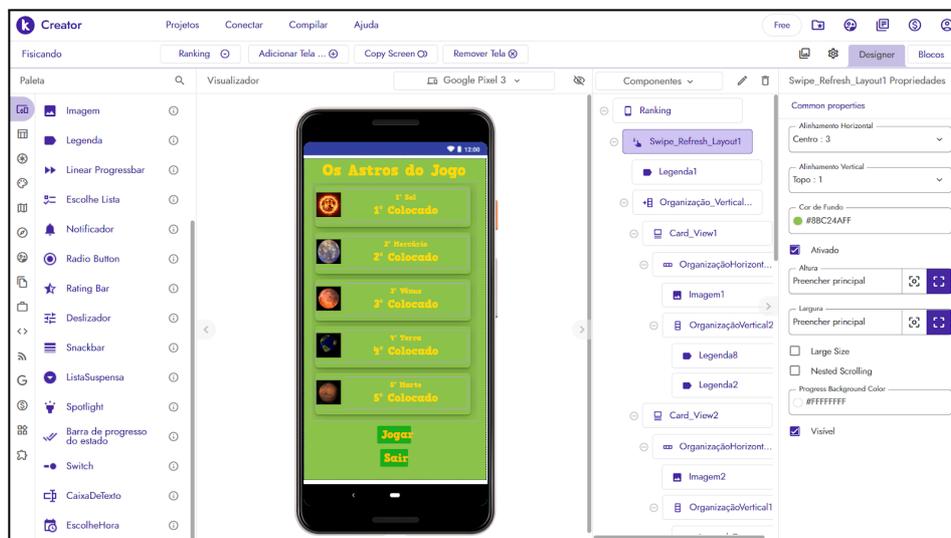


Figura 3.14 - Design da tela Ranking (Captura de Tela pelo autor, 2022).

O funcionamento desta tela é descrito a seguir:

- recupera do banco de dados local as informações do usuário e do ranking;
- recupera do banco de dados online as informações do ranking atual;
- atualiza a tela com as informações do ranking;
- verifica se o ranking é alterado pela pontuação do jogador;
- caso o ranking seja alterado pela pontuação do jogador, atualiza o ranking do banco de dados local e online;
- caso o usuário deslize o dedo para baixo, atualiza o ranking com informações do banco de dados online;
- fecha o aplicativo caso seja clicado o botão voltar do sistema durante sua execução;
- fecha o aplicativo caso seja clicado o botão “Sair”;
- abre a tela de seleção de nível caso seja clicado o botão “Jogar”.

A programação em blocos correspondentes a lista descrita acima segue abaixo:

inicializar global (aux) para criar lista vazia

inicializar global (StatusUsuario) para criar lista vazia

inicializar global (UsuarioLogado) para criar lista vazia

inicializar global (aux1) para criar lista vazia

inicializar global (List\_Ranking) para criar lista vazia

inicializar global (User\_Points) para 0

inicializar global (User\_Position) para 10

inicializar global (FezTroca) para verdadeiro

quando Botão1 - Clique  
fazer abrir outra tela nomeDaTela Select Nivel

quando Botão2 - Clique  
fazer fechar aplicação

quando Ranking - Inicializar  
fazer  
ajustar global UsuarioLogado para chamar Firebase DB1 - ObterValor  
rótulo UsuarioLogado  
valorSeRótuloNãoExistir  
ajustar global StatusUsuario para chamar Firebase DB1 - ObterValor  
rótulo User\_Status  
juntar selecionar item da lista lista obter global UsuarioLogado  
valorSeRótuloNãoExistir  
ajustar global User\_Points para selecionar item da lista lista obter global StatusUsuario  
índice 3  
ajustar global List\_Ranking para chamar Firebase DB1 - ObterValor  
rótulo Ranking  
valorSeRótuloNãoExistir  
chamar AtualizarRanking  
chamar Atualizar\_tela

para AtualizarRanking  
fazer  
chamar Firebase DB1 - ObterValor  
rótulo Ranking  
valorSeRótuloNãoExistir obter global List\_Ranking

para Atualizar\_tela  
fazer  
ajustar Legenda2 - Texto para juntar selecionar item da lista lista selecionar item da lista lista obter global List\_Ranking  
índice 1  
selecionar item da lista lista selecionar item da lista lista obter global List\_Ranking  
índice 1  
ajustar Legenda3 - Texto para juntar selecionar item da lista lista selecionar item da lista lista obter global List\_Ranking  
índice 3  
índice 1  
selecionar item da lista lista selecionar item da lista lista obter global List\_Ranking  
índice 2  
ajustar Legenda4 - Texto para juntar selecionar item da lista lista selecionar item da lista lista obter global List\_Ranking  
índice 3  
índice 1  
selecionar item da lista lista selecionar item da lista lista obter global List\_Ranking  
índice 3  
índice 3  
ajustar Legenda5 - Texto para juntar selecionar item da lista lista selecionar item da lista lista obter global List\_Ranking  
índice 1  
índice 4  
selecionar item da lista lista selecionar item da lista lista obter global List\_Ranking  
índice 4  
índice 3  
ajustar Legenda6 - Texto para juntar selecionar item da lista lista selecionar item da lista lista obter global List\_Ranking  
índice 1  
índice 5  
selecionar item da lista lista selecionar item da lista lista obter global List\_Ranking  
índice 3

```

quando Firebase_DB1 -> RecebeuValor
  rótulo (valor)
  fazer
    ajustar global List_Ranking para obter valor
    para cada número desde 2 até 5 passo 1
      fazer
        se obter global User_Points > selecionar item da lista lista selecionar item da lista lista obter global List_Ranking
          índice 3 índice obter número
        então chamar Alterar_Ranking
          break
      chamar Atualizar_tela
  
```

```

para Alterar_Ranking
  fazer
    ajustar global aux1 para obter global List_Ranking
    para cada número desde comprimento da lista lista obter global aux1 até 1 passo 1
      fazer
        se selecionar item da lista lista obter global StatusUsuario = selecionar item da lista lista selecionar item da lista lista obter global aux1
          índice 1 índice obter número
        então remover item da lista lista obter global aux1
          índice obter número
      adicionar itens à lista lista obter global aux1
      item obter global StatusUsuario
    enquanto testar obter global FezTroca
      fazer
        ajustar global FezTroca para falso
        para cada número desde comprimento da lista lista obter global aux1 até 2 passo 1
          fazer
            se selecionar item da lista lista selecionar item da lista lista obter global aux1 > selecionar item da lista lista selecionar item da lista lista obter global aux1
              índice 3 índice obter número índice 3 índice obter número
            então ajustar global aux1 para selecionar item da lista lista obter global aux1
              índice obter número - 1
            substituir item da lista lista obter global aux1
              índice obter número - 1
              substituição selecionar item da lista lista obter global aux1
                índice obter número
            substituir item da lista lista obter global aux1
              índice obter número
              substituição obter global aux
            ajustar global FezTroca para verdadeiro
          enquanto testar comprimento da lista lista obter global aux1 > 5
            fazer
              remover item da lista lista obter global aux1
                índice 6
            enquanto testar comprimento da lista lista obter global aux1 <= 5
              fazer
                adicionar itens à lista lista obter global aux1
                selecionar item da lista lista obter global aux1
                  índice comprimento da lista lista obter global aux1
            chamar Firebase_DB1 -> ArmazenarValor
              rótulo Ranking
              valorParaArmazenar obter global aux1
            chamar TinyDB1 -> ArmazenarValor
              rótulo Ranking
              valorParaArmazenar obter global aux1
            ajustar global List_Ranking para obter global aux1
  
```

```

quando Swipe_Refresh_Layout1 -> On Refresh
  fazer
    chamar AtualizarRanking
    ajustar Swipe_Refresh_Layout1 Refreshing para falso
  
```

```

quando Ranking -> Voltar Pressionado
  fazer
    fechar aplicação
  
```

### 3.1.5 - Tela de Seleção de Nível

A “Tela de Seleção de Nível” permite ao usuário escolher que nível deseja jogar, caso o nível escolhido já tenha sido habilitado para o jogador.

Na Figura 3.15 pode-se verificar os componentes utilizados na construção desta tela, conforme lista que segue:

- 1 componente da paleta Layout - 1 OrganizadorVerticalRolavel;
- 1 componente da paleta Interface com o usuário - 1 notificador;
- 4 componentes dinâmicos - 1 Dynamic\_Card\_View, 1 Dynamic\_Image, 1 Dynamic\_Space e 1 Dynamic\_Label;
- 1 componente da paleta Google - Firebase\_DB1;
- 1 componente da paleta Armazenamento - TinyDB1.

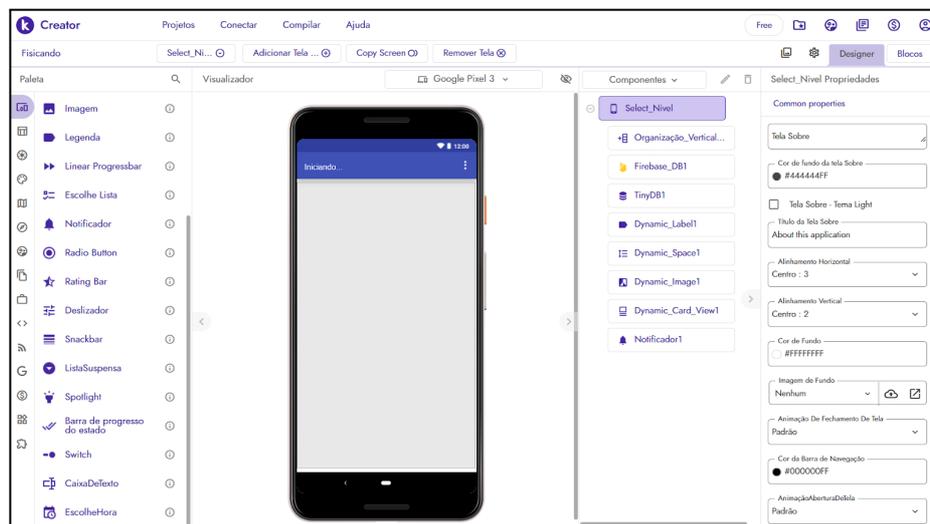


Figura 3.15 - Design da tela de seleção de nível (Captura de Tela pelo autor, 2022).

O funcionamento desta tela é descrito a seguir:

- recupera do banco de dados local as informações do nível atual do jogador;
- cria cinco Cards dinâmicos com duas imagens cada;
- caso o card seja clicado e esteja habilitado, será gerada uma lista de questões para aquele nível e será aberta a tela de questão;
- caso o card seja clicado e o nível não esteja habilitado será emitida uma mensagem “Nível ainda não liberado!”;
- abre a tela “Ranking” caso seja clicado o botão voltar do sistema.

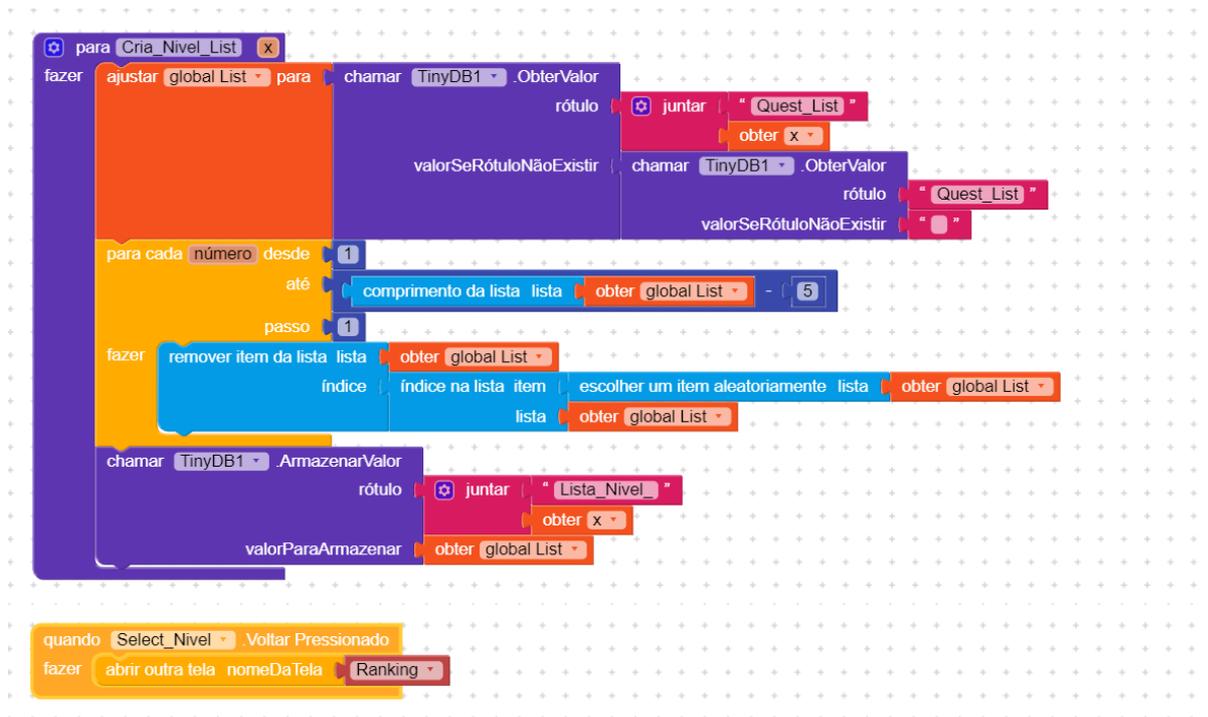
A programação em blocos correspondentes a lista descrita acima segue abaixo:

inicializar global List para criar lista vazia

inicializar global Num\_Fases para 0

```
quando Select_Nivel .Inicializar
fazer
  ajustar Select_Nivel .Titulo para " "
  ajustar global Num_Fases para 5
  para cada número desde 1
    até obter global Num_Fases
    passo 1
    fazer
      chamar Dynamic_Card_View1 .Create Card View
      id obter número
      arrangement Organização_Vertical_Rolável1
      comprimento -2
      altura 400
      chamar Dynamic_Image1 .Create Image
      id (100 + obter número)
      arrangement chamar Dynamic_Card_View1 .Get Card View By Id
      id obter número
      caminhoDoArquivo juntar "level_"
      obter número
      ".png"
      comprimento -2
      altura 200
      chamar Dynamic_Image1 .Create Image
      id (200 + obter número)
      arrangement chamar Dynamic_Card_View1 .Get Card View By Id
      id obter número
      caminhoDoArquivo juntar "SN"
      chamar TinyDB1 .ObterValor
      rótulo juntar "SubNivel_Atual"
      obter número
      valorSeRótuloNãoExistir " "
      ".png"
      comprimento -2
      altura 200
      chamar TinyDB1 .ArmazenarValor
      rótulo "SubNivel"
      valorParaArmazenar "0"
```

```
quando Dynamic_Card_View1 .Clique
id
fazer
  chamar TinyDB1 .ArmazenarValor
  rótulo "N_Pontos"
  valorParaArmazenar "0"
  chamar Cria_Nivel_List
  x obter id
  se
    chamar TinyDB1 .ObterValor
    rótulo "Nivel_Liberado"
    valorSeRótuloNãoExistir " "
    então
      abrir outra tela com valor inicial nomeDaTela Questao_1
      valorInicial obter id
    senão
      chamar Notificador1 .MostrarAlerta
      aviso "Nivel ainda não liberado!"
```



### 3.1.6 - Tela de Questão

A “Tela de Questão” apresenta os quesitos selecionados na tela de nível, conforme nível e subnível do jogador.

Na Figura 3.16 pode-se verificar os componentes utilizados na construção desta tela, conforme lista que segue:

- 2 componentes da paleta Layout - 1 OrganizadorVertical, 1 OrganizadorHorizontal;
- 1 componente da paleta Mídia - 1 Imagem;
- 6 componentes da paleta Interface com o usuário - 1 CaixaDeTexto, 4 CaixaDeSelecao e 1 Botão;
- 1 componente da paleta Google - Firebase\_DB1;
- 1 componente da paleta Armazenamento - TinyDB1.

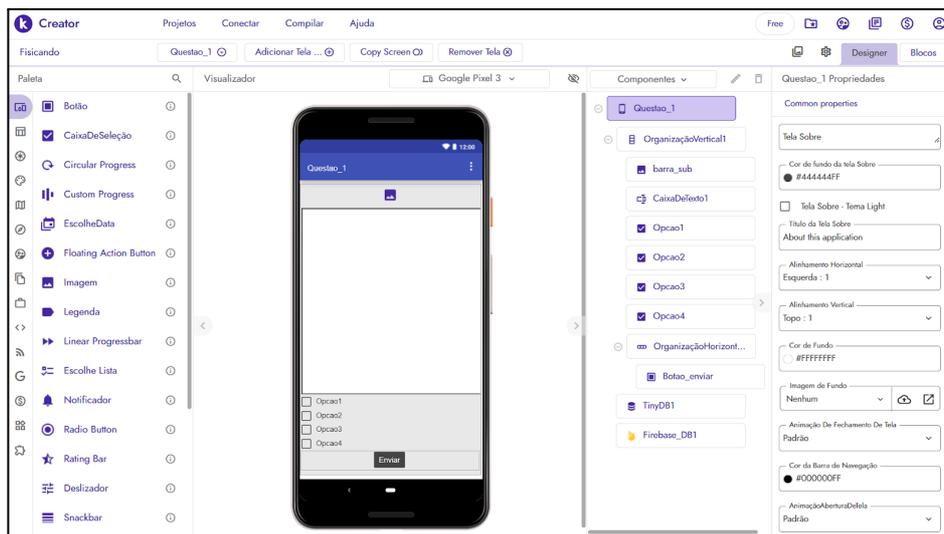


Figura 3.16 - Design da tela de questão (Captura de Tela pelo autor, 2022).

O funcionamento desta tela é descrito a seguir:

- atualiza os dados do jogador;
- atualiza a lista de questões;
- carrega a próxima questão da lista numa variável e na tela;
- caso o jogador selecione uma opção as outras são desmarcadas e é registrado internamente se ele acertou ou não;
- caso o botão “Enviar” seja pressionado, a lista de questões é atualizada, a dica associada a questão é salva no banco de dados local, se a resposta está correta abre a “Tela de Acerto” senão abre a tela “Tela de Erro”;
- caso o botão “Enviar” seja pressionado, e a lista de questões esteja finalizada será aberta a tela “Vitória”;
- abre a tela de selecionar nível caso seja clicado o botão voltar do sistema.

A programação em blocos correspondentes a lista descrita acima segue abaixo:

inicializar global Valor\_recebido para obter valor inicial

inicializar global Lista\_Q para criar lista vazia

inicializar global P\_Certo para 3

inicializar global Status\_Usuario para 0

inicializar global P\_Errado para -1

inicializar global Usuario\_Logado para ""

inicializar global Acertou para falso

inicializar global ProxQuestao para ""

inicializar global Questao para criar lista vazia

```
quando Questao_1 . Inicializar
fazer
  chamar AtualizarUsuario
  ajustar Questao_1 . Titulo para selecionar item da lista lista obter global Usuario_Logado
  indice 2
  chamar CarregarListaQ
  chamar CarregarProxQuestao
  chamar CarregarQuestao
  ajustar CaixaDeTexto1 . Read Only para verdadeiro
```

```
para AtualizarUsuario
fazer
  ajustar global Usuario_Logado para chamar TinyDB1 . ObterValor
  rótulo " UsuarioLogado "
  valorSeRótuloNãoExistir ""
```

```
para CarregarListaQ
fazer
  ajustar global Lista_Q para chamar TinyDB1 . ObterValor
  rótulo " juntar " Lista_Nivel "
  obter global Valor_recebido
  valorSeRótuloNãoExistir ""
```

```
para CarregarProxQuestao
fazer
  ajustar global ProxQuestao para selecionar item da lista lista obter global Lista_Q
  indice 1
```

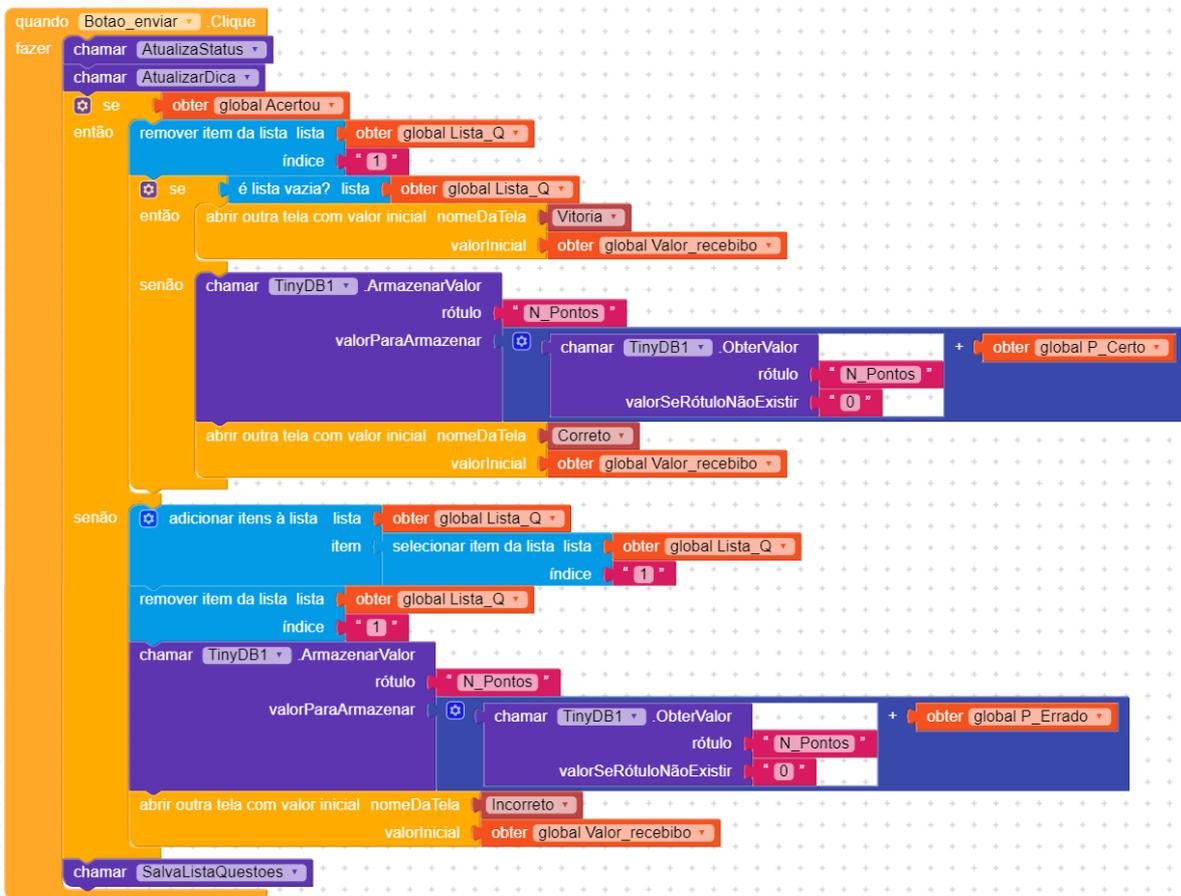
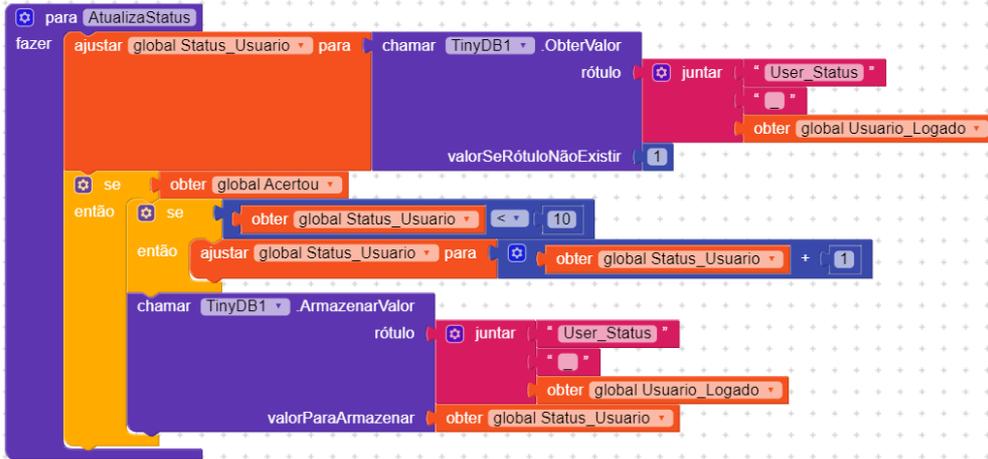
```
para CarregarQuestao
fazer
  ajustar global Questao para chamar TinyDB1 . ObterValor
  rótulo obter global ProxQuestao
  valorSeRótuloNãoExistir ""
  ajustar CaixaDeTexto1 . Texto para selecionar item da lista lista obter global Questao
  indice " 2 "
  ajustar Opcao1 . Texto para selecionar item da lista lista obter global Questao
  indice " 3 "
  ajustar Opcao2 . Texto para selecionar item da lista lista obter global Questao
  indice " 4 "
  ajustar Opcao3 . Texto para selecionar item da lista lista obter global Questao
  indice " 5 "
  ajustar Opcao4 . Texto para selecionar item da lista lista obter global Questao
  indice " 6 "
```

```
quando Opcao1 Alterado
fazer
  se Opcao1 Marcado
  então
    ajustar Opcao2 Marcado para falso
    ajustar Opcao3 Marcado para falso
    ajustar Opcao4 Marcado para falso
    se comparar textos Opcao1 Texto = selecionar item da lista lista obter global Questao
      índice " 7 "
    então
      ajustar global Acertou para verdadeiro
    senão
      ajustar global Acertou para falso
```

```
quando Opcao2 Alterado
fazer
  se Opcao2 Marcado
  então
    ajustar Opcao1 Marcado para falso
    ajustar Opcao3 Marcado para falso
    ajustar Opcao4 Marcado para falso
    se comparar textos Opcao2 Texto = selecionar item da lista lista obter global Questao
      índice " 7 "
    então
      ajustar global Acertou para verdadeiro
    senão
      ajustar global Acertou para falso
```

```
quando Opcao3 Alterado
fazer
  se Opcao3 Marcado
  então
    ajustar Opcao1 Marcado para falso
    ajustar Opcao2 Marcado para falso
    ajustar Opcao4 Marcado para falso
    se comparar textos Opcao3 Texto = selecionar item da lista lista obter global Questao
      índice " 7 "
    então
      ajustar global Acertou para verdadeiro
    senão
      ajustar global Acertou para falso
```

```
quando Opcao4 Alterado
fazer
  se Opcao4 Marcado
  então
    ajustar Opcao1 Marcado para falso
    ajustar Opcao2 Marcado para falso
    ajustar Opcao3 Marcado para falso
    se comparar textos Opcao4 Texto = selecionar item da lista lista obter global Questao
      índice " 7 "
    então
      ajustar global Acertou para verdadeiro
    senão
      ajustar global Acertou para falso
```



quando Questao\_1 Voltar Pressionado  
fazer abrir outra tela nomeDaTela Select\_Nivel

### 3.1.7 - Tela de Acerto

A “Tela de Acerto” será apresentada quando um jogador acertar um item.

Na Figura 3.17 pode-se verificar os componentes utilizados na construção desta tela, conforme lista que segue:

- 2 componentes da paleta Layout - 1 OrganizadorVertical, 1 OrganizadorHorizontal;
- 2 componente da paleta Mídia - 2 Imagens;
- 2 componentes da paleta Interface com o usuário - 1 Legenda e 1 Botão;
- 1 componente da paleta Armazenamento - TinyDB1.

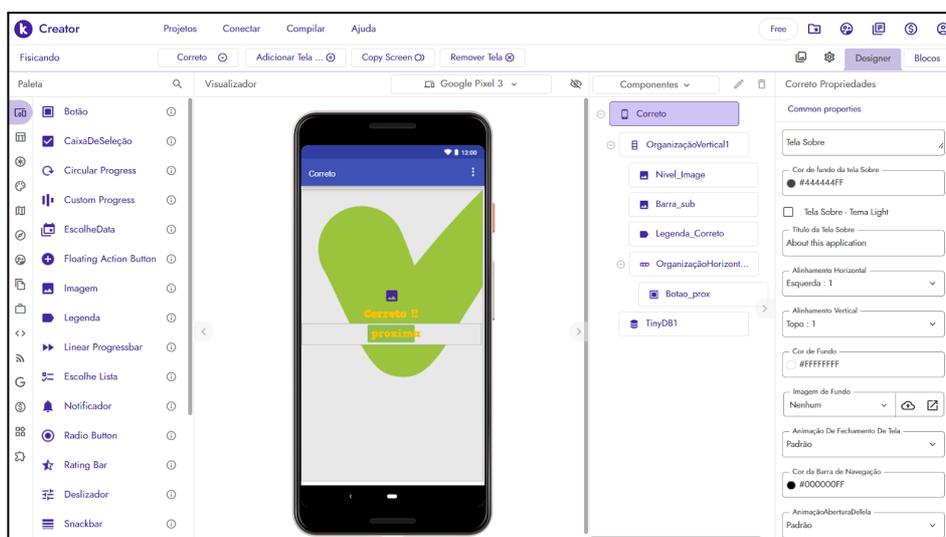
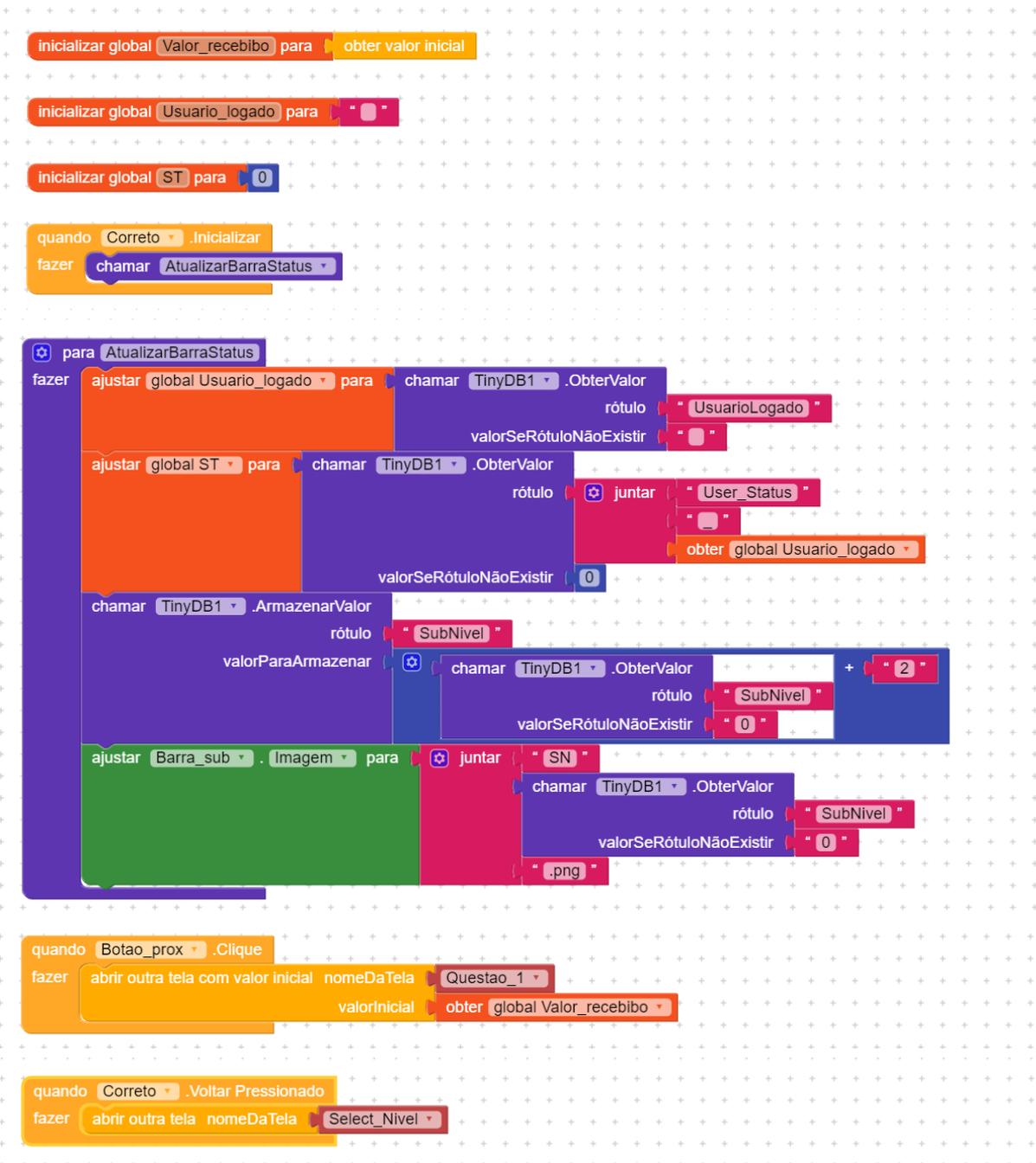


Figura 3.17 - Design da tela “Correto”(Captura de Tela pelo autor, 2022).

O funcionamento desta tela é descrito a seguir:

- atualiza a barra de status e salva a informação banco de dados local;
- caso o botão seja pressionado, abre a tela “Questão”;
- abre a tela de selecionar nível caso seja clicado o botão voltar do sistema.

A programação em blocos correspondentes a lista descrita acima segue abaixo:



### 3.1.8 -Tela de Erro

A “Tela de Erro” será apresentada quando um jogador errar um item.

Na Figura 3.18 pode-se verificar os componentes utilizados na construção desta tela, conforme lista que segue:

- 2 componentes da paleta Layout - 1 OrganizadorVertical, 1 OrganizadorHorizontal;
- 2 componente da paleta Mídia - 2 Imagens;

- 4 componentes da paleta Interface com o usuário - 1 Legenda, 2 Botões e 1 Notificador;
- 1 componente da paleta Armazenamento - TinyDB1.

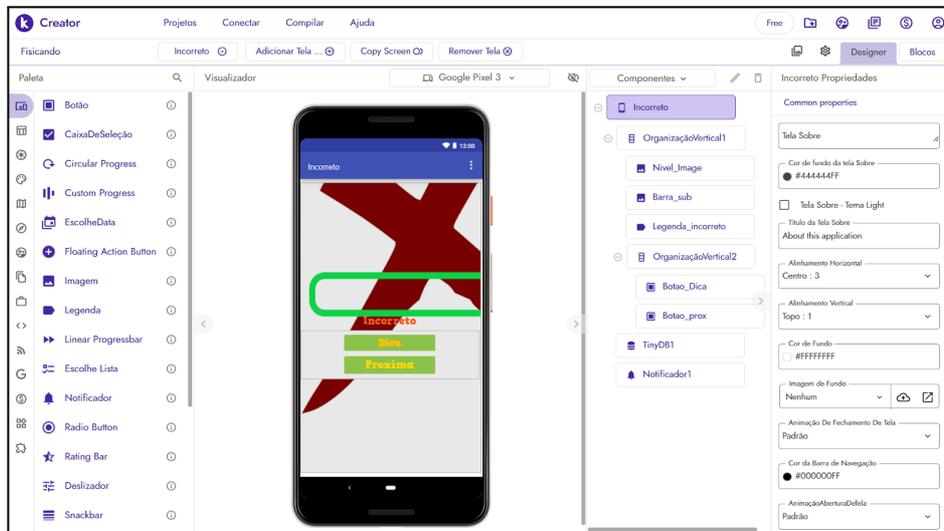
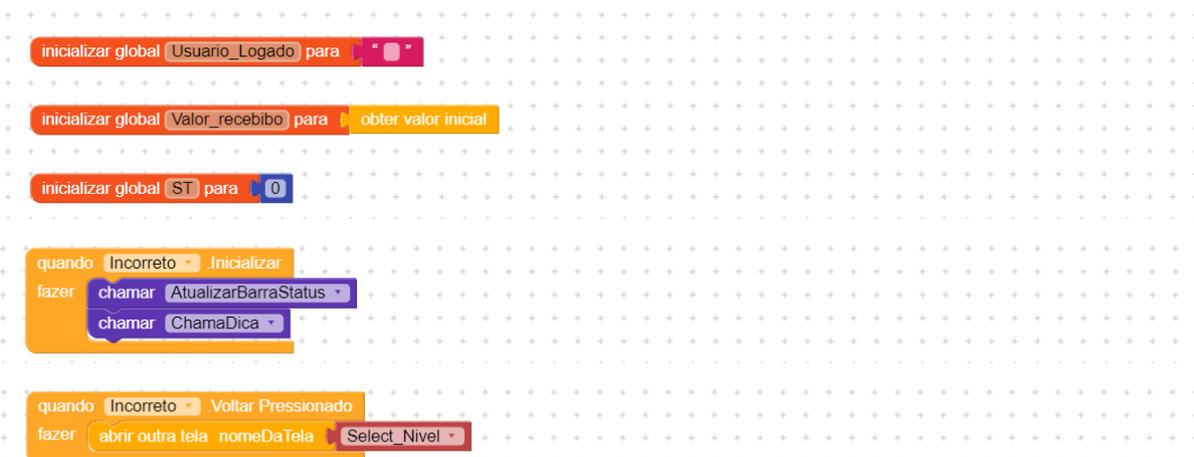


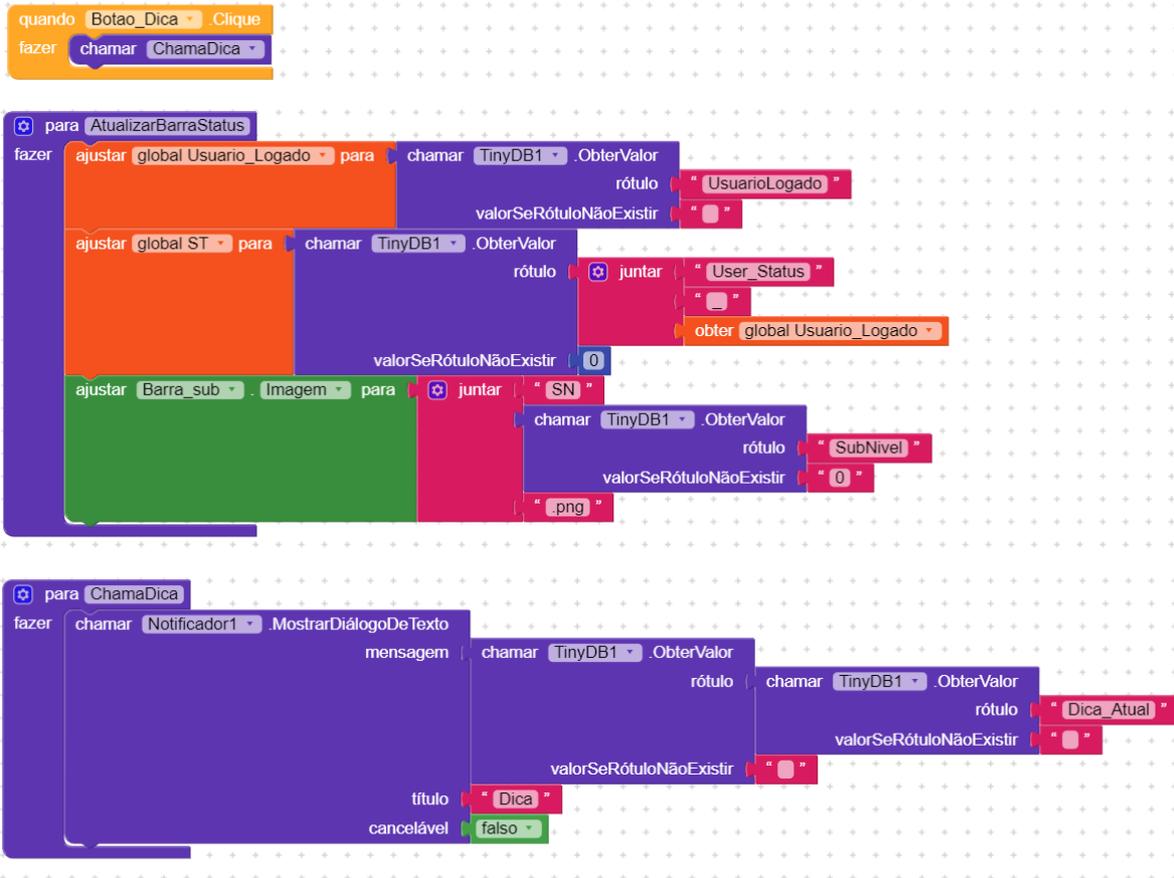
Figura 3.18 - Design da tela “Incorreto”(Captura de Tela pelo autor, 2022).

O funcionamento desta tela é descrito a seguir:

- atualiza a barra de status e salva a informação banco de dados local;
- abre uma notificação com a dica repassada pela questão anterior;
- caso o botão “Dica” seja pressionado, abre uma notificação com a dica repassada pela questão anterior;
- caso o botão “próxima” seja pressionado, abre a tela “Questão”;
- abre a tela de selecionar nível caso seja clicado o botão voltar do sistema.

A programação em blocos correspondentes a lista descrita acima segue abaixo:





### 3.1.9 - Tela De Vitória

A tela “Vitória” ocorre quando termina uma sequência de questões.

Na Figura 3.19 pode-se verificar os componentes utilizados na construção desta tela, conforme lista que segue:

- 1 componente da paleta Interface com o usuário - 1 botão;
- 1 componente da paleta Google - Firebase\_DB1;
- 1 componente da paleta Armazenamento - TinyDB1.

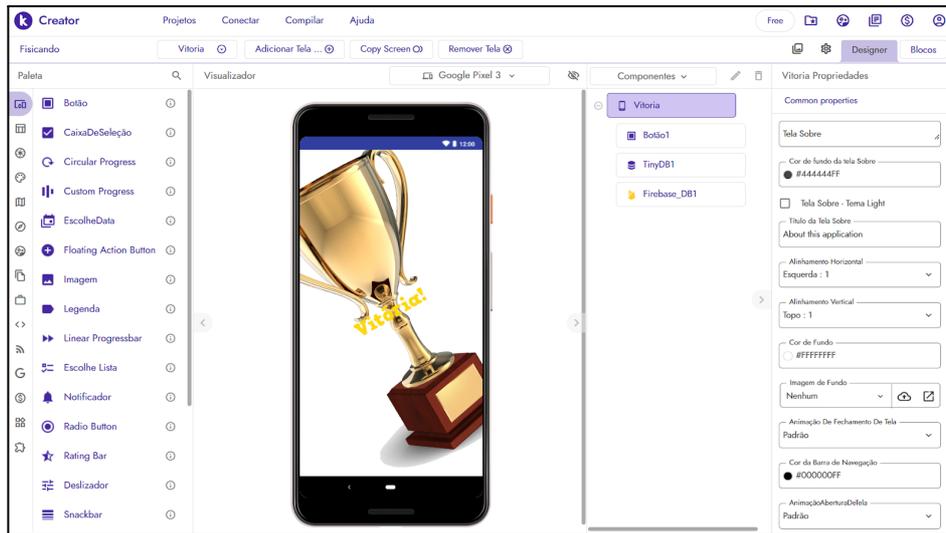
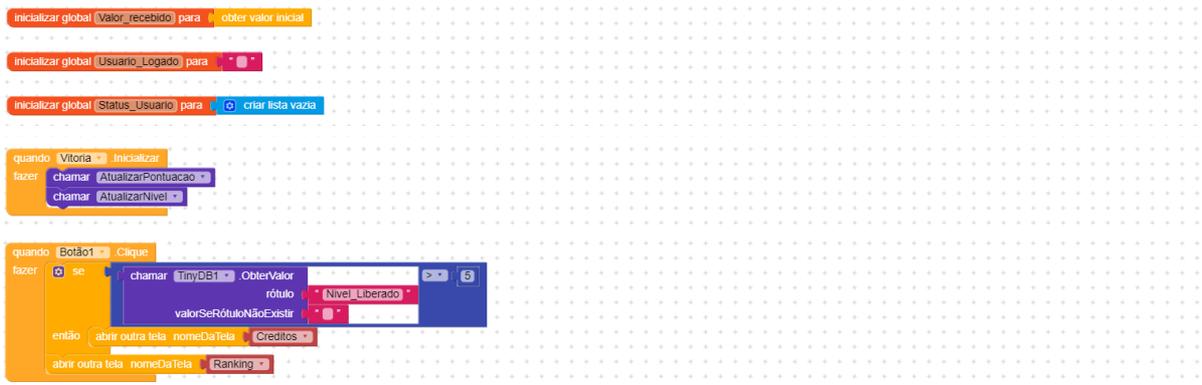


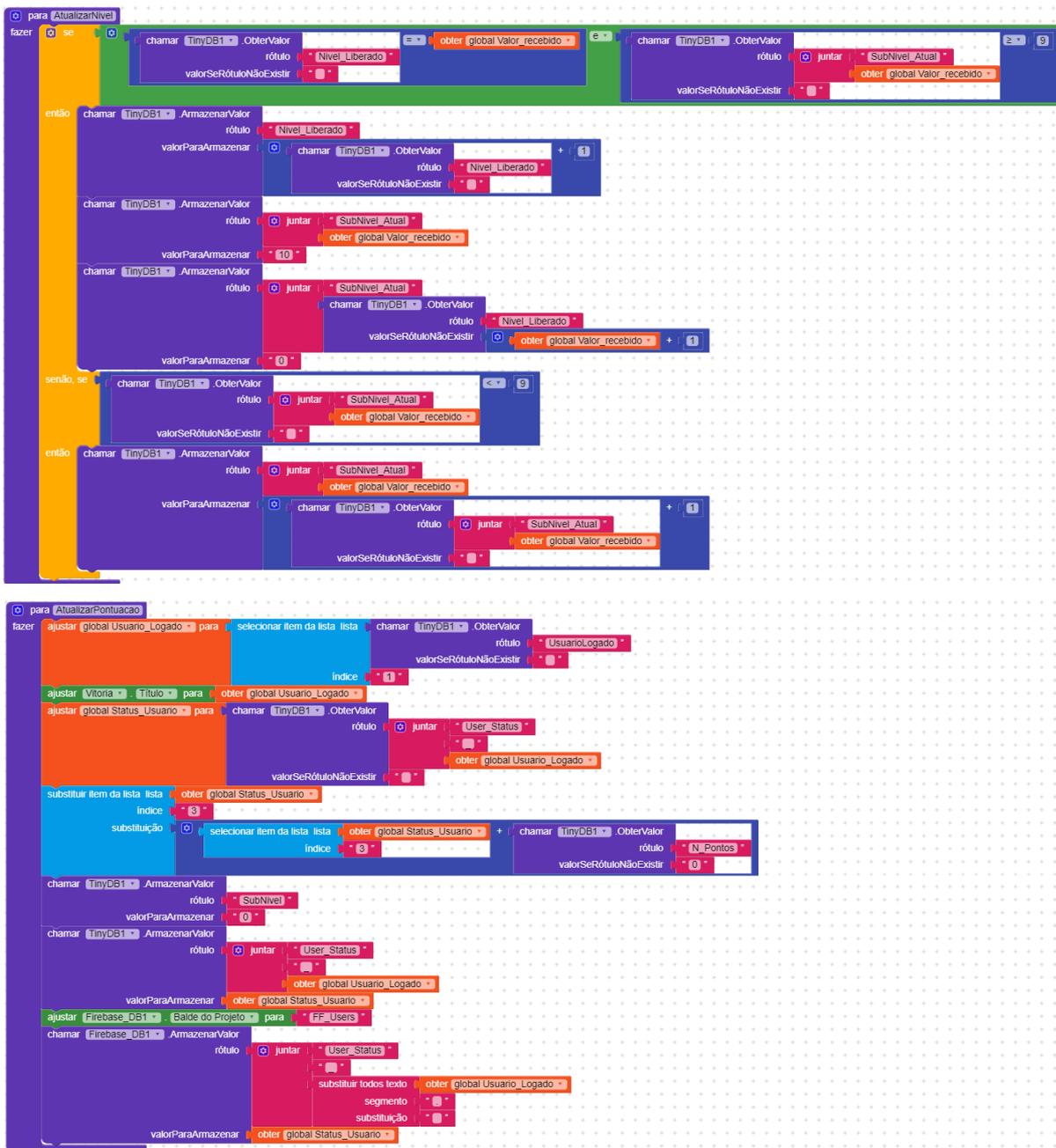
Figura 3.19 - Design da tela “Vitória” (Captura de Tela pelo autor, 2022).

O funcionamento desta tela quando ativa se resume a:

- atualiza a pontuação na base de dados local;
- atualiza o nível na base de dados local;
- salva o status do usuário na base de dados online;
- quando clicada, abre a tela “Ranking” caso tenha terminado uma lista, ou a tela créditos caso tenha terminado o jogo.

A programação em blocos correspondentes a lista descrita acima segue abaixo:





### 3.1.10 - Tela de Créditos

A “Tela de Créditos” ocorre quando termina a sequência de questões do jogo, e mostra o nome das pessoas que participaram do desenvolvimento do jogo.

Na Figura 3.20 pode-se verificar os componentes utilizados na construção desta tela, conforme lista que segue:

- 1 componente da paleta Interface com o usuário - 1 botão;

- 1 componente da paleta Google - Firebase\_DB1;
- 1 componente da paleta Armazenamento - TinyDB1.

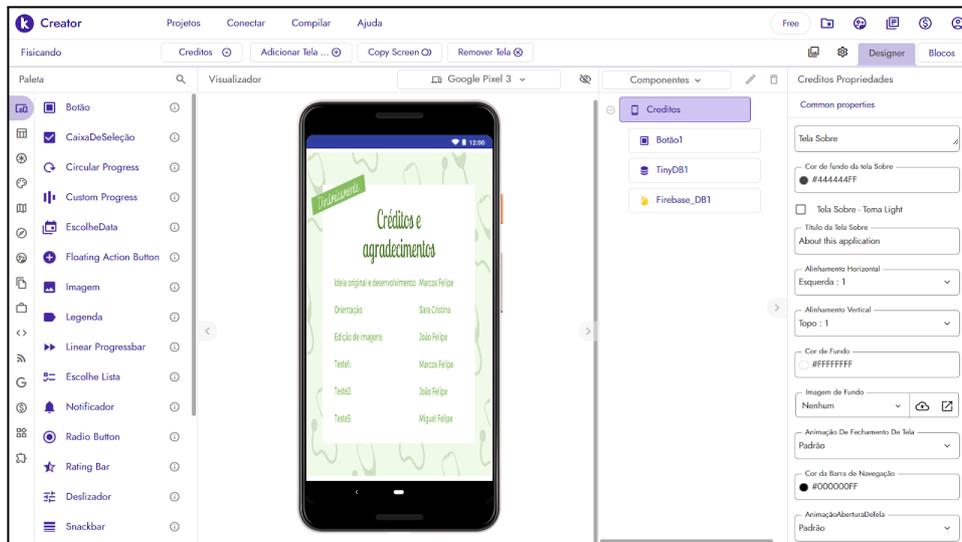


Figura 3.20 - Design da tela de créditos(Captura de Tela pelo autor, 2022).

O funcionamento desta tela quando ativa se resume a:

- abre a tela “Ranking” caso pressione o botão ;
- abre a tela “Ranking” caso pressione o botão voltar do sistema.

A programação em blocos correspondentes a lista descrita acima segue abaixo:



## 4. Regras do Jogo

O aplicativo DinamicaMente não possui regras explícitas, as ações possíveis são guiadas pelas possibilidades de cada tela, que são programadas antecipadamente, entretanto seu uso é resumido da seguinte forma:

- Ao entrar pela primeira vez no aplicativo será solicitada credenciais e Nome de Jogador, após o que será direcionado a tela de Ranking, onde pode escolher entre jogar ou sair do aplicativo;
- Ao escolher jogar, é direcionado a tela de níveis onde pode escolher jogar em algum dos níveis que estejam habilitados ou retornar à tela anterior;
- Ao escolher jogar um determinado nível, inicia-se o questionário compatível com o nível selecionado, onde poderá responder a questão ou voltar à tela anterior;
- Caso decida responder a questão, e acertar, será aberta a tela que informa do acerto e evolução dentro do nível, de onde pode-se seguir para a próxima questão ou retornar à tela de seleção de nível;
- Caso decida responder a questão, e errar, será aberta a tela que informa do erro e uma caixa de diálogo mostrando uma dica dentro do contexto da questão errada, que uma vez fechada, permite seguir para a próxima questão, rever a dica, ou retornar à tela de seleção de nível;
- Ao completar um nível, será aberto uma tela de celebração de vitória que só pode direcionar a tela de Ranking atualizada, podendo-se reiniciar a sequência em um nível mais avançado até alcançar o fim do jogo ou sair;
- Caso alcance o fim do jogo, após a tela de vitória será apresentada a tela de créditos, de onde se pode retornar ao Ranking;
- A cada questão respondida corretamente serão somados três pontos ao placar do jogador, e a cada questão errada será subtraído um ponto;
- O ranking é formado pelos cinco melhores placares do jogo.

O aplicativo é auto explicativo, implementando regras do tipo *ludos*, e é permitido jogar os níveis mais de uma vez, sem prejuízo à contagem de pontos, o que possibilita uma diferenciação na pontuação para jogadores/estudantes com estratégias diferentes, estimulando a competição.

## **5. Descrição da Aplicação do Produto**

O produto foi pensado para ser usado em paralelo à exposição do conteúdo de cinemática e de modo assíncrono, de tal modo que é aplicado um Questionário Preliminar (modelo no Apêndice 1), visando estabelecer uma base para futura comparação de sua evolução.

Para aplicar este produto em sua turma o professor deve fazer contato com o autor deste produto através do email “[dinamicamente.app@gmail.com](mailto:dinamicamente.app@gmail.com)”, onde solicitará cópia do aplicativo e credenciais de acesso, para se, e para seus estudantes.

O professor deve fornecer o aplicativo aos estudantes e ajudá-los na instalação. É necessário habilitar o aparelho para usar aplicativos baixados de fontes diversas da loja oficial, entretanto muitos alunos já possuem o conhecimento de como realizar o procedimento, e podem auxiliar o professor. Caso venham a ter dificuldades podem recorrer a tutoriais na *internet*, como o sugerido abaixo:

– Tecnoblog

<[Cada estudante deve receber sua credencial, previamente solicitada pelo professor, e pode iniciar jogando.](https://tecnoblog.net/responde/fontes-desconhecidas-android-oreo/#:~:text=Acesse%20as%20Configura%C3%A7%C3%B5es%3B,toque%20em%20Instalar%20apps%20desconhecidos.></a></p></div><div data-bbox=)

O professor decide o momento que usará o aplicativo de acordo com o seu cronograma de aulas, e após o período de aplicação do produto e regência do conteúdo, pode avaliar a evolução da turma com a reaplicação do questionário inicial, podendo ou não atribuir premiações aos melhores colocados no Ranking do jogo.

## 6. Considerações Finais

Após avaliarmos os resultados da aplicação do produto “DinamicaMente” em quatro turmas do 9º ano, foi observado que este pode trazer uma melhora no desempenho destes alunos nos conteúdos iniciais de cinemática, ajudando a desenvolver habilidades na resolução de problemas de física. E ao melhorar o intercâmbio de significados dos conceitos trocados entre o professor e os estudantes, favorece também as futuras aprendizagens.

O produto “DinamicaMente” foi construído e alimentado com questões que visam facilitar o aprendizado do conteúdo de cinemática, porém pode ser utilizado para o ensino de outros conteúdos, e inclusive outras disciplinas, bastando para isso a alimentação adequada do banco de dados do aplicativo, entretanto tais intervenções ainda requerem algum conhecimento mais avançado da estrutura do aplicativo.

De forma sintética, temos a convicção de que novas técnicas e metodologia lúdicas favorecem o aprendizado de novos conteúdos, tornando menos tensas as relações de ensino e

aprendizagem, permitindo ao estudante o desenvolvimento de um aprendizado de física consciente do mundo natural que o cerca e de sua responsabilidade na modificação deste, e acreditamos que o produto educacional DinamicaMente contribui para este objetivo.

## 7. Referências

[BRASIL, 2013] BRASIL. Ministério da Educação. Base Nacional Comum Curricular. Brasília: MEC/SEB, 2013. Disponível em:

<[http://basenacionalcomum.mec.gov.br/images/BNCC\\_EI\\_EF\\_110518\\_versaofinal\\_site.pdf](http://basenacionalcomum.mec.gov.br/images/BNCC_EI_EF_110518_versaofinal_site.pdf)> acesso em 05 de fev. de 2022.

BLENDER. Blender: the freedom to create, 2023. Página inicial. Disponível em:

<<https://www.blender.org/>> acesso em: 04 de dez. de 2022.

[Brasil, 2013] BRASIL. Ministério da Educação. Base Nacional Comum Curricular. Brasília: MEC/SEB, 2013. Disponível em:

<[http://basenacionalcomum.mec.gov.br/images/BNCC\\_EI\\_EF\\_110518\\_versaofinal\\_site.pdf](http://basenacionalcomum.mec.gov.br/images/BNCC_EI_EF_110518_versaofinal_site.pdf)> acesso em: 05 de fev. de 2022.

CANVA. Canva, 2023. Página inicial. Disponível em: <[https://www.canva.com/pt\\_br/](https://www.canva.com/pt_br/)> acesso em 04 de dez. 2022.

DEIVID ROTHEN. Youtube, 2023. Canal Deivid Rothen. Disponível em:

<[https://www.youtube.com/watch?v=brZdGvJGsTI&list=PLgIw6\\_M7v8vrWYm-J5g2uKW12JF13tCBh](https://www.youtube.com/watch?v=brZdGvJGsTI&list=PLgIw6_M7v8vrWYm-J5g2uKW12JF13tCBh)> acesso em 20 de jan. de 2023.

FIREBASE. Firebase, 2023. Página inicial. Disponível em: <<https://firebase.google.com/>> acesso em 04 de dez. de 2022.

KODULAR. Kodular: Much more than a modern app creator without coding, 2023. Disponível em: <<https://www.kodular.io/>> acesso em: 04 de dez. de 2022.

PROGRAMAÇÃO EM BLOCOS BRASIL. Youtube, 2023. Canal Programação em Blocos Brasil. Disponível em:

<<https://www.youtube.com/watch?v=z64x8BxlE6o&list=PLSaHQQBC7dhrQ5WHzgDaZrJ6Js9dbcj5C>> acesso em 20 de jan. de 2023.

ROBOTIZANDO. Youtube, 2023. Canal Robotizando. Disponível em:

<<https://www.youtube.com/watch?v=LBhKuYyM6nM&list=PLe6v9G9A83kNcsNJqGlsBMTIoTDBIWnly>> acesso em 20 de jan. de 2023.

TECNOBLOG. Tecnoblog, 2023. Como instalar apps de fontes desconhecidas no Android.

Disponível em:

<<https://tecnoblog.net/responde/fontes-desconhecidas-android-oreo/#:~:text=Acesse%20as%20Configura%C3%A7%C3%B5es%3B,toque%20em%20Instalar%20apps%20desconhecidos.>> acesso em 04 de dez. de 2022.

# Apêndice 1 - Questionário Preliminar



## Questionário Preliminar

Os dados obtidos neste formulário serão avaliados com o objetivo de subsidiar a determinação da eficácia de um produto educacional, assim como a construção de um TCC (Trabalho de Conclusão de Curso) e ajudarão no desenvolvimento de novas formas de trabalho no processo de ensino-aprendizagem.

O tratamento dos dados será realizado de modo que a identidade dos colaboradores permaneça anônima, sendo agrupados de modo a enfatizar os resultados representativos da população inteira.

Agradecemos desde já sua colaboração ao estudo!

Nome: \_\_\_\_\_

Escola: \_\_\_\_\_

Turma: \_\_\_\_\_

Data de Aplicação: \_\_\_/\_\_\_/\_\_\_\_\_ Professor que aplicou: \_\_\_\_\_

1) Partindo do instante zero, um veículo sai da posição inicial de 10 metros e chega à posição final de 60 metros após 5 segundos. Marque a alternativa correta.

- a)  A posição inicial é 60 m
- b)  A posição inicial é 10 m
- c)  A posição inicial é 0 m
- d)  A posição inicial é 5 m

2) Maria saiu de Belém às 6 horas, de um ponto da estrada onde o marco quilométrico indicava km 0. Ela chegou a Mosqueiro às 7 horas e 30 minutos, onde o marco quilométrico da estrada indicava km 60. Marque a alternativa correta.

- a)  O tempo inicial foi 60 h
- b)  O tempo inicial foi 6 h
- c)  O tempo inicial foi 7,5 h
- d)  O tempo inicial foi 0 h

3) Pedro e Maria saíram para passear de carro. Eles partiram de São Paulo às 10 h em direção à Braúna, localizada a 500 km da capital. Ao chegar no destino final, Maria olhou no relógio e viu que eram 15 h. Marque a alternativa correta.

- a)  A distância percorrida foi de 15 km
- b)  A distância percorrida foi de 10 km
- c)  A distância percorrida foi de 500 km
- d)  A distância percorrida foi de 5 km

4) Ao cobrar uma falta em um jogo de futebol, um jogador imprime à bola uma velocidade tal que a bola gasta 3 s até atingir as redes que ficam a uma distância de 36 metros. Marque a alternativa correta.

- a)  O intervalo de tempo é 3 s
- b)  O intervalo de tempo é 12 s
- c)  O intervalo de tempo é 36 s
- d)  O intervalo de tempo é 108 s

5) Um veículo partindo do repouso atinge uma velocidade final de 100 km/h após 10 segundos. Marque a alternativa correta.

- a)  A velocidade final é zero
- b)  A velocidade inicial é 100 km/h
- c)  A velocidade inicial é zero
- d)  O tempo é zero

6) Um veículo viajando a uma velocidade de 10 km/h freia e após 3 segundos atinge o repouso. Marque a alternativa correta.

- a)  A velocidade final é 10 km/h
- b)  A velocidade inicial é 10 km/h
- c)  A velocidade final é 3 km/h
- d)  O tempo é 10 segundos

7) Dirigindo por uma rodovia, Pedro passou pelo km 100 as 9 horas da manhã e seguindo viagem passou pelo km 180 as 10 horas da manhã. Marque a alternativa correta.

- a)  A distância percorrida é de 80 km
- b)  A distância percorrida é de 9 km
- c)  A distância percorrida é de 10 km
- d)  A distância percorrida é de 100 km

8) Em viagem de Recife a Caruaru, que ficam 120 km distantes uma da outra, Jeremias saiu de Recife às 7 h e deve chegar a Caruaru às 9 h. Marque a alternativa correta.

- a)  O tempo decorrido na viagem deve ser de 7 horas
- b)  O tempo decorrido na viagem deve ser de 9 horas
- c)  O tempo decorrido na viagem deve ser de 120 horas
- d)  O tempo decorrido na viagem deve ser de 2 horas

9) Um carro faz uma viagem de 180 km em 2h. Marque a alternativa correta.

- a)  A velocidade média é 120 km/h
- b)  A velocidade média é 100 km/h
- c)  A velocidade média é 90 km/h
- d)  A velocidade média é 80 km/h

10) Ao passar pelo marco “km 200” de uma rodovia, um motorista vê um anúncio com a inscrição: “ABASTECIMENTO E RESTAURANTE A 30 MINUTOS”. Considere que esse posto de serviço se encontra junto ao marco “km 245” dessa rodovia. Marque a alternativa correta.

- a)  A velocidade média considerada pelo anunciante é 120 km/h
- b)  A velocidade média considerada pelo anunciante é 100 km/h
- c)  A velocidade média considerada pelo anunciante é 90 km/h
- d)  A velocidade média considerada pelo anunciante é 80 km/h