

# Tamagochi

## Descrição geral

O projeto na construção de aplicativo de tamagochi (bichinho virtual), utilizando a tecnologia React Native. O aplicativo seguirá a abordagem offline first.

## Regras e entrega

- O trabalho pode ser desenvolvido em duplas ou individualmente (a avaliação será a mesma para ambos cenários)
- A entrega deverá ser feita via GitHub (apenas 1 repositório para a dupla)
  - Serão avaliados os commits das duplas, então caso algum integrante não realize o desenvolvimento, a nota do mesmo será reduzida/anulada
- A nota será com base na apresentação dos alunos, que será realizada nos dias 12 e 13 de setembro de 2024
- Alunos da modalidade EAD poderão apresentar via video-chamada
- Trabalhos onde for constatado cópia, ambos os grupos terão a nota anulada.
- Apenas um integrante da dupla deverá preencher o formulário

## Projeto

### Telas

- Tela de listagem de bichinhos
  - Listar o nome, atributos (fome, sono e diversão), status e imagem
- Tela de cadastro de bichinhos. Informar os campos para cadastro:
  - Nome
  - Imagem
- Tela de detalhes do bichinho
  - Acessada através da listagem, deve mostrar a imagem do bichinho, sua foto, atributos, status e fornecer as opções:
    - Alimentar: irá alimentar o bichinho, e por consequência aumentar o seu atributo "fome"
    - Dormir: Irá colocar o bichinho para dormir, aumentando seu atributo "sono"
    - Brincar: Irá abrir a tela de jogo
- Tela de jogos:
  - Deverão ser implementados 2 minigames distintos e em telas separadas (exemplo: jogo da velha, jogo da memória, par ou impar, pedra-papel-tesoura, etc)
    - Um dos minigames deve utilizar algum módulo nativo do dispositivo (giroscópio, câmera, localização, etc)

- Ao jogar, o atributo “diversão” do bichinho deverá ser atualizado
- Caso os alunos desejem implementar mais de 2 minigames, é permitido

## **Lógica de saúde do bichinho**

- Atributos
  - Caso os alunos queiram implementar atributos adicionais, é permitido
  - Os atributos devem ter seu valor descontado conforme o tempo passa. O valor a ser descontado fica a critério dos alunos.
    - Exemplo: a cada 1h o bichinho perde 1 ponto de fome, sono e diversão
    - Caso o bichinho tenha seus 3 atributos zerados, ele morre
  - Os atributos vão de 0 a 100
- Status
  - O status dos bichinhos é calculado através da soma dos atributos
    - 0: status “morto”
    - De 1 a 50: status “crítico”
    - De 51 a 100: status “muito triste”
    - De 101 - 150: status “triste”
    - De 151 - 200: status “ok”
    - De 201 - 250: status “bem”
    - de 252 - 300: status “muito bem”

## **Implementação**

- O app será offline, ou seja, deverá utiliza um banco de dados local para armazenar as informações. O banco a ser escolhido fica a critério do aluno (SQLite, Watermellon, Realm, etc)
- O app pode ser desenvolvido utilizando Expo ou react Native CLI
- O app deverá ser compilado, não importa se para Android, iOS ou ambos
- O app deverá ser visualmente agradável e o layout deve fazer sentido com as regras propostas