



UNifeob
| ESCOLA DE NEGÓCIOS



2024

PROJETO INTEGRADO



UNIFEOB

CENTRO UNIVERSITÁRIO DA FUNDAÇÃO DE ENSINO
OCTÁVIO BASTOS

ESCOLA DE NEGÓCIOS

**ANÁLISE E DESENVOLVIMENTO DE SISTEMAS
CIÊNCIA DA COMPUTAÇÃO**

PROJETO INTEGRADO

**DESENVOLVIMENTO DE SOLUÇÕES CONSOLE
INTEGRADAS PARA EDUCAÇÃO,
SUSTENTABILIDADE, INCLUSÃO SOCIAL E
EMPREENDEDORISMO**

Asilo“Lar de Jesus”

SÃO JOÃO DA BOA VISTA, SP

NOVEMBRO 2024

UNIFEOB

CENTRO UNIVERSITÁRIO DA FUNDAÇÃO DE ENSINO
OCTÁVIO BASTOS

ESCOLA DE NEGÓCIOS

**ANÁLISE E DESENVOLVIMENTO DE SISTEMAS
CIÊNCIA DA COMPUTAÇÃO**

PROJETO INTEGRADO

**DESENVOLVIMENTO DE SOLUÇÕES CONSOLE
INTEGRADAS PARA EDUCAÇÃO,
SUSTENTABILIDADE, INCLUSÃO SOCIAL E
EMPREENDEDORISMO**

Asilo“Lar de Jesus”

MÓDULO MODELAGEM E DESENVOLVIMENTO DE SISTEMAS

Business Intelligence – Prof^ª. Mariângela Martimbianco Santos

Programação Orientada a Objeto – Prof. Nivaldo de Andrade

Lógica de Programação – Prof. Marcelo Ciacco Almeida

Modelagem de Dados – Prof. Max Streicher Vallim

Projeto de Modelagem e Desenvolvimento de Sistemas – Prof^ª. Mariângela M. Santos

Estudantes:

Gabrielly Simão Domingos, RA 23000615

Guilherme Moltine Canhedo Soares, RA 24000140

Lucas Sernaglia Pereira da Silva, RA 24000375

Pedro Henrique dos Santos Pereira, RA 24000880

Rafael Alves Blaske, RA 24000158

Vinicius Angelini Manca, RA 24001312

SÃO JOÃO DA BOA VISTA, SP
NOVEMBRO 2024

SUMÁRIO

1. INTRODUÇÃO	4
2. DESCRIÇÃO DA EMPRESA	5
3. PROJETO INTEGRADO	6
3.1 PROGRAMAÇÃO ORIENTADA A OBJETO	6
3.1.1 CLASSES E OBJETOS	7
3.1.3 MÉTODOS ESTÁTICOS, PÚBLICOS E PRIVADOS	11
3.2 LÓGICA DE PROGRAMAÇÃO	12
3.2.1 CONCEITOS FUNDAMENTAIS DO DESENVOLVIMENTO DE SOFTWARE	12
3.2.2 DESENVOLVIMENTO DE APLICAÇÕES	13
3.3 MODELAGEM DE DADOS	15
3.3.1 MODELO CONCEITUAL	16
3.3.2 MODELO LÓGICO E FÍSICO	16
3.4.1 ORGANIZAÇÃO E IDENTIFICAÇÃO DAS INFORMAÇÕES	19
3.4.2 MANIPULAÇÃO E ANÁLISE DE DADOS	20
3.4.3 CRIAÇÃO DE MODELOS DE ANÁLISE DE DADOS	20
3.5 CONTEÚDO DA FORMAÇÃO PARA A VIDA: GERENCIANDO FINANÇAS	21
3.5.1 GERENCIANDO FINANÇAS	22

Os conceitos econômicos e financeiros fundamentais são essenciais para a compreensão do nosso dia a dia. Entre eles, destacam-se a renda, despesas, poupança e investimento. Por exemplo, saber calcular o que entra e sai mensalmente do seu orçamento ajuda a identificar quanto pode ser poupado ou investido. Um estudante pode usar uma planilha simples para registrar essas informações, facilitando o controle financeiro e a tomada de decisões. 22

Independência financeira refere-se à capacidade de viver sem depender de um salário fixo, ou seja, ter rendimentos que sustentem o estilo de vida desejado. Para isso, é fundamental entender o valor da riqueza pessoal, que envolve ativos como investimentos e propriedades. Um exemplo prático é registrar diariamente os gastos em um aplicativo financeiro, ajudando a identificar despesas desnecessárias e potencialmente aumentando a poupança para investimentos futuros. 22

O conhecimento sobre dívidas e juros compostos é crucial para evitar armadilhas financeiras. Por exemplo, ao pegar um empréstimo com juros altos, a dívida pode se multiplicar rapidamente. Um estudante deve calcular o total que pagará ao final de um financiamento, considerando os juros. Alternativas como o empréstimo consignado ou a negociação de dívidas podem ser soluções viáveis para quem enfrenta dificuldades financeiras, além de buscar a educação financeira para evitar novas dívidas. 22

Definir metas claras é fundamental para alcançar sonhos, como comprar um carro ou financiar a faculdade. Um exemplo prático seria criar um "quadro de metas", visualizando os objetivos e os passos necessários para alcançá-los. Envolver amigos ou familiares nesse processo pode aumentar a motivação e criar um sistema de apoio.

Realizar reuniões periódicas para discutir o progresso e ajustar as estratégias pode ser uma forma eficaz de manter todos engajados.	22
3.5.2 ESTUDANTES NA PRÁTICA	22
4. CONCLUSÃO	24
REFERÊNCIAS	25
ANEXOS	26

1. INTRODUÇÃO

O projeto será um detector de quedas para os idosos residentes do Asilo Lar de Jesus da cidade de São José do Rio Pardo. Nele consistirá um Acelerômetro e Giroscópio MPU6050; e um sensor de pulsão para aferir os batimentos cardíacos dos idosos. Ele visa promover uma maior segurança ao bem-estar dos idosos e também facilitar e agilizar o serviço dos cuidadores através de um monitoramento das movimentações dos idosos no ambiente.

Esse projeto pretende melhorar a comunicação entre os funcionários do local e os idosos através da implementação de pulseiras tecnológicas, as quais serão encarregadas de emitir sinais de alerta instantâneos por meio de notificações no caso deles sofrerem quedas.

Nessa faixa etária, os indivíduos, em sua maioria, possuem altos graus de fragilidade e sensibilidade perante uma queda. Como afirmam Martini, Silva e Curiati (2023, p. 10), ao citar Carpenter e Liu (2018):

Mesmo traumas menores, como quedas da própria altura, podem causar declínio funcional, visitas ao PA, diminuição da qualidade de vida e aumento da mortalidade, de modo que aproximadamente um terço dos pacientes idosos com queda que recebem alta do departamento de emergência apresenta um desses desfechos em 3 meses. Entre 36 e 50% dos pacientes apresentam um evento adverso dentro de 1 ano após uma queda.

Sob esse viés, é de fundamental importância que o dispositivo emita um alerta instantâneo direcionado a toda equipe de cuidadores, a fim de que possa agilizar ao máximo a prestação de cuidados e aumentar as chances de salvar a vida do idoso.

2. DESCRIÇÃO DA EMPRESA

O Asilo "Lar de Jesus" é uma instituição dedicada ao amparo e cuidado de idosos, proporcionando um ambiente acolhedor e de assistência especializada. Com uma rotina estruturada para atender às necessidades dos residentes, o asilo abriga atualmente 84 idosos, enfrentando desafios diversos relacionados à saúde e habilidades cognitivas.

Essa instituição apresenta como razão social a Associação Espírita Beneficente "Paulo de Tarso", o seu CNPJ é 59.904.458.0001-18 e a sua localização é na Rua Luís Spessoto, número 80, bairro João de Souza da cidade de São José do Rio Pardo, no estado de São Paulo.

A rotina diária dos idosos do asilo é a seguinte: entre às 06:00 e 08:30, os idosos iniciam o dia acordando e tomando os seus banhos. A partir das 08:30, o café da manhã começa a ser servido. Depois, às 11:00, é servido o almoço sendo ele uma refeição saudável e nutritiva ideal para os idosos. Mais tarde, às 17:00, é servido o jantar, assegurando uma refeição balanceada ao final do dia.

De segunda a sexta-feira, ou seja, nos dias úteis, são marcados por um período de descanso, interação e convivência no asilo, portanto, sem atividades programadas, a fim de que os idosos adquiram o hábito de socialização entre eles para aproveitar momentos de paz e tranquilidade. Aos sábados, são realizadas atividades recreativas e de entretenimento para proporcionar muita diversão e estimulação cognitiva.

O Asilo Lar de Jesus enfrenta desafios significativos em relação à saúde dos idosos, englobando consultas médicas relativamente frequentes e também cuidados especializados.

Caso aconteçam situações de brigas, discussões e desentendimentos entre os idosos, a equipe de cuidadores e profissionais do asilo está totalmente preparada para intervir e resolver todos os conflitos.

O acesso à internet via Wi-Fi não é liberado aos idosos, pois muitos estão bastante debilitados e não possuem as mínimas condições de utilizar equipamentos eletrônicos.

A maioria dos idosos possui mobilidade reduzida, muitos são cadeirantes, o que exige adaptações nas instalações a fim de que possa garantir uma maior acessibilidade. Muitos deles também apresentam diferentes níveis de redução mental, incluindo alguns que não escrevem (informação verbal)¹

¹ Todas as informações desta página foram conseguidas através de contato direto com a empresa.

3. PROJETO INTEGRADO

3.1 PROGRAMAÇÃO ORIENTADA A OBJETO

O nosso protótipo envolve o uso de conhecimento tecnológico voltado para sensores e algoritmos que assegurem que todos os componentes da pulseira funcionem corretamente e de maneira integrada. Iremos coordenar a integração dos sistemas para que os sensores de monitoramento e os algoritmos responsáveis pela detecção de quedas funcionem sincronizados, proporcionando o resultado esperado.

Na etapa de desenvolvimento e implementação do software, iremos criar o aplicativo móvel que permitirá aos cuidadores do asilo monitorar o funcionamento da pulseira e receber notificações de alerta em tempo real. O aplicativo foi projetado para ser intuitivo e fácil de ser utilizado sendo integrado ao sistema da pulseira, permitindo a comunicação entre o dispositivo e o celular dos cuidadores de maneira contínua e confiável.

No processo de design e prototipagem, iremos aplicar princípios de ergonomia a fim de criar um protótipo da pulseira que seja prático e confortável para idosos. Levando em consideração as necessidades específicas deste público, o design da pulseira será ajustado após a realização de testes, garantindo que o produto final seja fácil de usar, leve e adequado ao corpo dos idosos para garantir que a pulseira não tenha apenas uma boa aparência, mas que também proporcione uma experiência de uso confortável isenta de incômodos.

A realização de testes será uma etapa fundamental para assegurar a qualidade e durabilidade do produto. Serão realizados testes em laboratórios com o objetivo de verificar o desempenho da pulseira em condições reais de uso. Esses testes farão com que avaliemos a precisão dos algoritmos de detecção, além de testar a eficácia e segurança do dispositivo em diferentes cenários, garantindo que o produto esteja pronto para ser comercializado com total confiança na sua funcionalidade.

Por fim, após a realização de todos os testes, nós iremos coletar feedbacks dos usuários, especialmente dos idosos, para realizar ajustes finais na pulseira. Esse processo será essencial para ajustar aspectos do produto, como conforto, funcionalidade e eficácia, assegurando que o dispositivo final possa atender completamente às expectativas e necessidades dos usuários, oferecendo uma solução segura, útil e fácil de utilizar.

3.1.1 CLASSES E OBJETOS

Aplicação: a implementação do sistema de pulseira de detecção de quedas para idosos requer a criação de uma estrutura orientada a objetos, a fim de representar as entidades e funcionalidades essenciais ao funcionamento do sistema. A seguir, descrevem-se as principais classes que serão desenvolvidas, juntamente com seus atributos e métodos.

Classe Pulseira: representa o dispositivo de monitoramento utilizado para detectar quedas e enviar alertas ao aplicativo móvel. Ela contém os sensores necessários para a medição de movimento e aceleração, além de manter a comunicação com o sistema externo, garantindo o envio de notificações quando uma queda é detectada.

- **Atributos:**
 - id: identificador único da pulseira.
 - sensorMovimento: instância da classe *SensorMovimento*, responsável por medir a aceleração e identificar movimentos bruscos.
 - statusConexao: indica o estado da conexão com o aplicativo móvel (conectado/desconectado).
- **Métodos:**
 - detectarQueda(): analisa os dados dos sensores para identificar se uma queda ocorreu.
 - enviarAlerta(): Envia uma notificação para o aplicativo móvel caso uma queda seja detectada.
 - atualizarStatusConexao(status): Atualiza o status da conexão com o aplicativo móvel.

Classe Sensor Movimento: representa o sensor responsável por detectar mudanças bruscas na aceleração, que são indicativas de um movimento anormal, como o que ocorre durante uma queda. O sensor coleta dados de aceleração nos eixos X, Y e Z, e esses dados são analisados para determinar se houve uma queda.

- **Atributos:**
 - id: Identificador único do sensor.
 - aceleracaoX: medida da aceleração no eixo X.
 - aceleracaoY: medida da aceleração no eixo Y.
 - aceleracaoZ: medida da aceleração no eixo Z.
- **Métodos:**
 - coletarDados(): Coleta dados de aceleração em tempo real.

- `analisarMovimento()`: analisa os dados coletados para identificar padrões de movimento anormais, como os que ocorrem durante uma queda.

Classe Idoso: representa o usuário da pulseira, ou seja, a pessoa que será monitorada pelo dispositivo. Ela contém informações essenciais sobre o idoso e permite o gerenciamento de dados relacionados à sua saúde e segurança. Essa classe também é responsável por gerenciar a interação do idoso com o aplicativo móvel, recebendo os alertas enviados pela pulseira.

- **Atributos:**

- `nome`: nome do idoso.
- `idade`: idade do idoso.
- `historicoQuedas`: lista de quedas registradas pelo sistema.
- `statusSaude`: informações sobre o estado de saúde do idoso, como condições médicas relevantes.

- **Métodos:**

- `registrarQueda()`: registra uma queda detectada pela pulseira no histórico de quedas do idoso.
- `atualizarStatusSaude()`: atualiza as informações de saúde do idoso, como mudanças em sua condição física.

Classe Aplicativo Móvel: representa o software que se comunica com a pulseira, proporcionando uma interface para o monitoramento em tempo real da condição do idoso. O aplicativo recebe os alertas da pulseira e notifica o usuário sobre a detecção de quedas, além de exibir informações sobre a conexão com o dispositivo.

- **Atributos:**

- `nomeUsuario`: nome do usuário do aplicativo, que pode ser o próprio idoso ou um cuidador.
- `statusConexaoPulseira`: estado da conexão com a pulseira (conectado/desconectado).
- `alertasRecebidos`: lista de alertas recebidos da pulseira, informando sobre eventuais quedas ou falhas na conexão.

- **Métodos:**

- `receberAlerta(alerta)`: recebe e processa alertas enviados pela pulseira, indicando a detecção de quedas.
- `mostrarNotificacao()`: exibe notificações para o usuário sobre o status da pulseira e eventuais quedas detectadas.

- atualizarConexao(status): atualiza o status da conexão com a pulseira, indicando se o dispositivo está conectado ou desconectado.

Essas classes representam a estrutura principal do sistema de detecção de quedas para idosos, permitindo o monitoramento contínuo da saúde do idoso, a comunicação eficiente entre os dispositivos e a rápida resposta em situações de emergência. O sistema visa garantir a segurança e o bem-estar do idoso, com foco na detecção precoce de quedas e na comunicação ágil de alertas aos cuidadores ou familiares.

3.1.2 ATRIBUTOS, MÉTODOS, ENCAPSULAMENTO, HERANÇA E POLIMORFISMO.

A estrutura do projeto da pulseira para detectar quedas foi cuidadosamente planejada com base nos princípios da Programação Orientada a Objetos (POO), visando uma implementação modular, reutilizável e de fácil manutenção. A aplicação é composta por várias classes principais, cada uma com responsabilidades bem definidas, permitindo que o sistema seja escalável e flexível para futuras modificações. As classes principais do projeto incluem Pulseira, Usuário, SensorQueda e Notificador, cada uma desempenhando um papel específico dentro da arquitetura do sistema.

A classe Pulseira representa o dispositivo em si, sendo responsável pela interação com o usuário e pela integração dos módulos de sensores e notificações. Ela possui atributos que descrevem as características da pulseira, como o status de funcionamento, a configuração do sensor e a conexão com os dispositivos de alerta. Entre seus métodos estão aqueles que permitem iniciar a monitorização, ativar ou desativar a detecção de quedas, e gerenciar a comunicação com o sistema de notificações.

A classe Usuário, por sua vez, representa o usuário da pulseira, no caso, um idoso que poderá estar em risco de quedas. Os atributos dessa classe incluem informações pessoais, como nome, idade, e dados de contato, além das configurações personalizadas que o usuário pode definir para a pulseira. A classe Usuário possui métodos para registrar ou atualizar os dados do usuário e para configurar as preferências do sistema, como a sensibilidade do sensor ou os tipos de alertas desejados.

A classe SensorQueda é responsável por detectar as quedas do usuário. Ela contém atributos relacionados à sensibilidade do sensor e aos parâmetros de detecção, como aceleração e ângulo de inclinação. Seus métodos são responsáveis por monitorar continuamente os movimentos do usuário, ativar a detecção de quedas e simular os diversos

cenários nos quais o sensor deve disparar um alerta. O comportamento do sensor pode ser ajustado conforme a necessidade, garantindo a precisão da detecção em diferentes situações.

A classe Notificador gerencia o envio de alertas para os cuidadores ou familiares do usuário quando uma queda é detectada. Seus atributos incluem informações sobre os destinatários das notificações e os métodos de comunicação disponíveis, como SMS, e-mail ou notificações em tempo real. A classe Notificador possui métodos para enviar os alertas de forma eficiente, personalizar as mensagens de aviso e garantir que as notificações cheguem com clareza e em tempo hábil.

A implementação dessas classes segue os princípios da Programação Orientada a Objetos (POO), utilizando conceitos como encapsulamento, herança e polimorfismo para garantir um código bem estruturado e flexível. O encapsulamento é aplicado para controlar o acesso aos atributos das classes, garantindo que apenas métodos específicos possam modificar os dados internos, protegendo assim a integridade do sistema e evitando alterações indesejadas. Por exemplo, os atributos da classe Usuario, como nome e idade, são privados, e métodos específicos são usados para acessá-los ou alterá-los, respeitando as regras de negócio.

A herança é usada para especializar classes e criar uma hierarquia entre elas. Por exemplo, pode-se criar subclasses de Notificador para diferentes tipos de notificação, como NotificadorSMS e NotificadorEmail, que herdam comportamentos gerais da classe Notificador, mas implementam suas próprias versões dos métodos de envio de alertas. Isso permite uma maior flexibilidade e reutilização de código, além de facilitar a introdução de novos tipos de notificações no futuro, sem a necessidade de reescrever o sistema completo.

Por fim, o polimorfismo é utilizado para que as subclasses de uma classe possam apresentar comportamentos diferentes, mesmo que compartilhem a mesma interface. Por exemplo, a classe Notificador pode ter vários tipos de notificadores (SMS, e-mail, etc.), e cada um pode implementar o método de envio de alertas de maneira distinta, ajustando o comportamento do sistema de acordo com o tipo de notificação escolhido. Esse conceito permite que o código seja mais flexível e adaptável, já que o sistema pode tratar os diferentes tipos de notificadores de maneira uniforme, apesar de suas implementações distintas.

A aplicação desses conceitos de POO, como encapsulamento, herança e polimorfismo, foi fundamental para o sucesso da implementação da pulseira antiqueda. Eles permitiram criar uma estrutura modular e organizada, na qual os diferentes componentes do sistema podem ser facilmente modificados, expandidos ou mantidos, garantindo a

escalabilidade e a evolução do projeto ao longo do tempo. O design orientado a objetos também facilita a compreensão do sistema, tornando-o mais intuitivo para os desenvolvedores e, conseqüentemente, mais confiável e eficaz na execução de suas funcionalidades.

3.1.3 MÉTODOS ESTÁTICOS, PÚBLICOS E PRIVADOS

No desenvolvimento de sistemas de monitoramento utilizando dispositivos como pulseiras inteligentes, a utilização de diferentes tipos de métodos nas classes que representam os componentes do sistema é fundamental para garantir a modularidade, segurança e eficiência na comunicação entre os objetos. Os métodos podem ser classificados em três categorias principais: estáticos, públicos e privados, cada um com sua finalidade específica.

Métodos Estáticos: são aqueles que pertencem à classe e não a uma instância específica da classe. Eles são úteis em situações em que a operação a ser realizada não depende dos dados de um objeto individual, mas sim de uma funcionalidade global que afeta o sistema como um todo. No contexto de um sistema de monitoramento de quedas, um exemplo de método estático seria o método `calcularNumeroDeQuedas()`, presente em uma classe como `GerenciadorDePulseiras`. Este método poderia ser responsável por contabilizar o número total de quedas registradas por todas as pulseiras em um determinado período, sem a necessidade de instanciar objetos da classe `PulseiraSensor` individualmente para cada pulseira.

```
public static int calcularNumeroDeQuedas() {  
    // Lógica para calcular o total de quedas registradas por todas as pulseiras  
}
```

Ao ser chamado diretamente da classe, o método estático permite centralizar a contagem de quedas, facilitando a manutenção e a análise dos dados do sistema.

Métodos Públicos: são aqueles que têm a finalidade de permitir a interação com outras partes do sistema ou com outras instâncias de objetos. Esses métodos fornecem a interface necessária para que operações sejam realizadas a partir de objetos específicos. No caso de uma pulseira inteligente, um exemplo de método público seria `detectarQueda()`, da classe `PulseiraSensor`. Este método seria responsável por monitorar os sensores da pulseira (como o acelerômetro e o giroscópio), processar os dados de movimento e, caso uma queda seja detectada, acionar um alerta para a central de monitoramento.

```
public void detectarQueda() {  
    // Lógica para verificar dados de movimento e identificar uma queda  
    // Caso uma queda seja detectada, um alerta é enviado para o sistema central  
}
```

Além disso, um outro exemplo de método público seria o método enviarAlerta(), que seria chamado para notificar os responsáveis ou a central de monitoramento sobre a detecção de uma queda. A visibilidade pública desses métodos permite que o sistema interaja com outros módulos, garantindo uma resposta rápida e adequada em situações de emergência.

Métodos Privados: são utilizados para encapsular a lógica interna de uma classe, garantindo que detalhes de implementação não sejam acessados ou modificados diretamente de fora da classe. Eles são fundamentais para preservar a integridade dos dados e a consistência do sistema, restringindo o acesso direto a operações sensíveis. No contexto de uma pulseira inteligente, um exemplo de método privado seria analisarMovimento(), que seria responsável por processar as leituras dos sensores (acelerômetro, giroscópio) para identificar padrões que indiquem uma queda. Este método seria chamado internamente pelo método público detectarQueda(), de modo que a lógica de análise dos sensores ficasse oculta ao usuário, evitando manipulações indevidas.

```
private boolean analisarMovimento() {  
    // Lógica que verifica se os dados dos sensores indicam uma queda  
    // Retorna verdadeiro se houver uma queda, caso contrário, retorna falso  
}
```

O uso de métodos privados assegura que a análise dos dados sensoriais seja realizada de maneira controlada, sem exposição desnecessária a outras partes do código. Dessa forma, a segurança e a eficiência do sistema são aprimoradas, uma vez que o comportamento interno da classe é ocultado do usuário e só pode ser acessado por métodos públicos controlados.

3.2 LÓGICA DE PROGRAMAÇÃO

3.2.1 CONCEITOS FUNDAMENTAIS DO DESENVOLVIMENTO DE SOFTWARE

Para construir o dispositivo e garantir que ele desempenhe seu papel, faz-se necessário o uso de um pensamento estruturado, ou seja, a lógica, para ordenar os passos necessários para resolver um problema (Nicoletti, 2017).

Com isso em mente, a lógica utilizada no detector de quedas consiste em duas partes: a detecção da queda propriamente dita e o envio/recebimento do alerta.

Para construir o código de detecção, foram inseridos limites de velocidade linear e angular fundamentados nos cálculos realizados por Silva (2018), para que, quando os valores do sensor ultrapassem esse limite, o sistema identifique a queda. Além disso, é calculada a velocidade modular dos eixos x, y e z do acelerômetro e do giroscópio, para uma leitura multidirecional. Caso a placa consiga detectar uma queda, ela, ao invés de dar o alerta, retorna um valor ao aplicativo assim que este fizer uma requisição ao servidor. Este valor, ao ser recebido, funciona como um gatilho, ativando os comandos de alerta da aplicação: o telefone vibra, há um alarme sonoro e uma janela pop-up aparece na tela dizendo que houve uma possível queda.

3.2.2 DESENVOLVIMENTO DE APLICAÇÕES

O sistema de detecção de quedas deve ser capaz de identificar quedas com base em mudanças abruptas na aceleração do usuário. Para garantir a precisão da detecção, a queda precisa ser confirmada por um período de inatividade, que pode ser de, por exemplo, 10 segundos. Essa confirmação é importante para evitar que movimentos ou atividades normais sejam erroneamente interpretados como quedas. Uma vez que a queda é detectada e confirmada, o sistema deverá enviar uma notificação de emergência para os contatos previamente configurados pelo usuário. Essa notificação deve incluir, sempre que possível, a localização GPS do usuário, permitindo que os contatos saibam onde o idoso está, facilitando o auxílio imediato.

Após o envio da notificação, o sistema deverá aguardar um período de resposta do usuário, que pode ser de cerca de 30 segundos. Caso não haja resposta por parte do usuário, o sistema enviará uma nova notificação aos contatos de emergência, garantindo que a situação seja monitorada de forma eficaz até que uma ação seja tomada. Durante todo esse processo, a pulseira deve realizar um monitoramento contínuo dos dados do sensor e também do estado do usuário, verificando se ele está ativo ou inativo. Essa vigilância constante é fundamental para assegurar a prontidão do sistema em qualquer situação.

Os cuidadores terão a capacidade de personalizar a experiência por meio de um aplicativo, onde poderá configurar os contatos de emergência e ajustar os parâmetros relacionados às notificações, como o tempo de espera ou os métodos de alerta. Além disso, o aplicativo irá contar com uma interface simples e intuitiva que permitirá aos cuidadores adicionar, editar ou remover contatos de emergência facilmente, além de configurar preferências relacionadas ao funcionamento da pulseira.

A funcionalidade do sistema é sustentada por vários módulos e suas respectivas funções. O Módulo de Sensores será responsável por coletar e processar os dados provenientes do sensor de aceleração da pulseira, utilizando uma lógica que permita a detecção de quedas com precisão. O Módulo de Notificações terá a função de enviar mensagens de emergência aos contatos configurados, caso uma queda seja identificada. O Módulo de Configuração, por sua vez, permitirá aos cuidadores gerenciar seus contatos de emergência e ajustar as preferências de notificações de forma personalizada. Já o Módulo de Monitoramento ficará encarregado de acompanhar a atividade dos idosos e verificar o estado do sensor, garantindo que o dispositivo esteja sempre monitorando e funcionando corretamente. Dessa forma, o sistema atuará de forma integrada e contínua para assegurar a segurança dos idosos em situações de risco.

3.2.3 IMPLEMENTAÇÃO E VALIDAÇÃO

A etapa de implementação e validação do projeto da pulseira de detecção de quedas para idosos foi fundamental para garantir a eficácia e a confiabilidade do sistema desenvolvido. Durante essa fase, todos os componentes do projeto foram integrados de forma coesa, assegurando que a aplicação funcionasse de maneira eficiente e sem falhas. O módulo principal da pulseira, o sensor de queda, foi responsável por detectar quando o usuário sofria uma queda. Para isso, o sensor foi projetado com parâmetros de sensibilidade ajustáveis, o que permitiu simular e testar diferentes cenários de queda em tempo real, garantindo sua precisão.

O sistema de alerta foi projetado para ativar automaticamente quando o sensor detectava uma queda. Assim que o evento era identificado, o sistema enviava notificações claras e imediatas ao cuidador ou familiares do usuário. Isso foi feito para garantir que a informação chegasse rapidamente, permitindo uma resposta ágil em situações de emergência. Além disso, foi desenvolvido um menu interativo para a interface de usuário, que permitia aos usuários monitorar a pulseira e acessar suas funcionalidades com

facilidade. Esse menu possibilitava iniciar a monitorização da pulseira, visualizar configurações e encerrar o sistema de forma intuitiva, proporcionando uma experiência amigável e sem complexidade.

Após a implementação, o sistema passou por uma série de testes e validação. Foram realizados testes específicos para verificar a eficácia do sensor de queda, simulando diferentes cenários e assegurando que o módulo detectava com precisão as quedas, sem falhas. O desempenho do sistema de notificações também foi avaliado, observando-se a rapidez e a clareza com que os alertas eram enviados após a detecção de uma queda. Além disso, o menu interativo foi testado por diferentes usuários para garantir que fosse intuitivo e fácil de usar, atendendo às necessidades de um público com pouca familiaridade com tecnologia.

Com todos os testes realizados, a aplicação demonstrou ser uma solução eficiente para a monitorização da segurança de idosos. A integração bem-sucedida dos módulos e a validação das funcionalidades atenderam a todos os requisitos do projeto, reforçando a viabilidade da pulseira como uma ferramenta útil e confiável para prevenir acidentes e proporcionar mais segurança aos usuários.

3.3 MODELAGEM DE DADOS

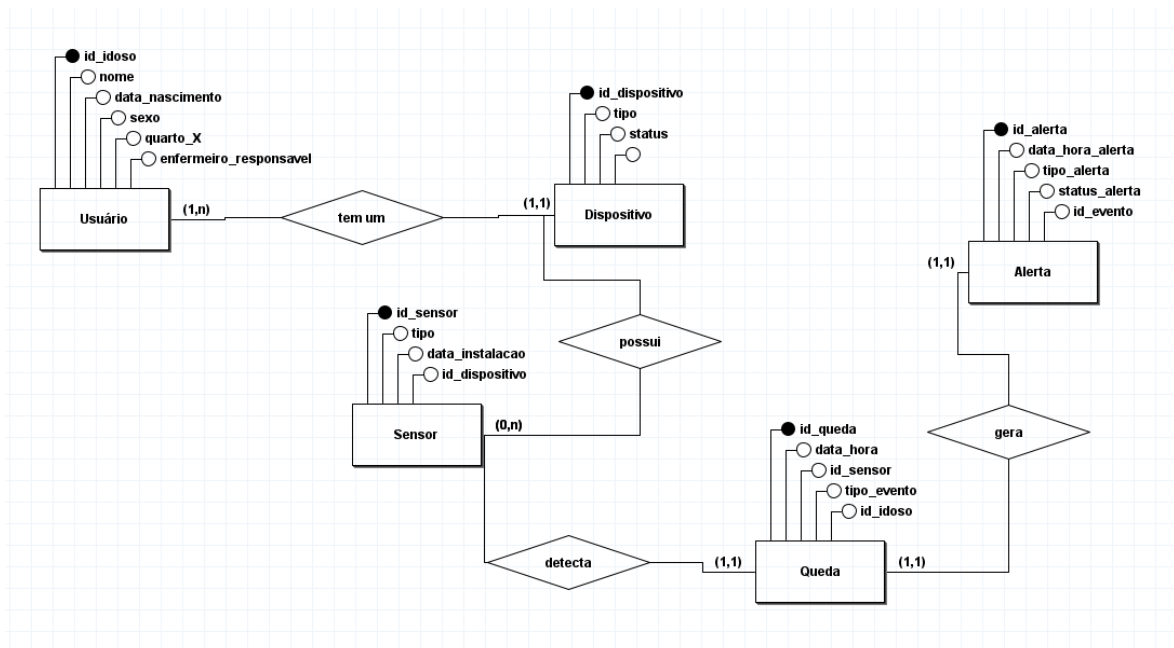
A modelagem de dados é o processo de estruturar e organizar dados de forma a representar de maneira clara e eficiente as informações relevantes de um sistema ou domínio específico. Por meio de modelos, é possível criar uma abstração das necessidades de dados de uma organização ou aplicação, facilitando a análise, o armazenamento e a manipulação desses dados. A modelagem de dados envolve a definição de entidades, seus atributos e os relacionamentos entre elas, utilizando diagramas como o Modelo Entidade-Relacionamento (ER) ou diagramas UML.

Esse processo é essencial em diversas áreas, como no desenvolvimento de bancos de dados, sistemas de informação, e em projetos de Big Data e Inteligência Artificial, onde a correta organização e estruturação dos dados é crucial para o desempenho e a integridade das soluções. A modelagem de dados permite uma melhor compreensão das informações, promovendo uma base sólida para decisões estratégicas e operacionais.

3.3.1 MODELO CONCEITUAL

O projeto propõe a criação de um banco de dados para gerenciar informações sobre usuários de uma pulseira anti-queda, eventos de queda, notificações para cuidadores e dados médicos relevantes. O banco de dados inclui entidades como *Usuário* (com ID, nome, idade, endereço e telefone), *Queda* (registrando data, hora, local e gravidade do evento), *Notificação* (que alerta cuidadores sobre quedas) e *Cuidador* (com dados do responsável pelo cuidado do usuário). Um usuário pode ter várias quedas e cuidadores, e cada queda gera uma notificação para o cuidador. O sistema visa melhorar a segurança do usuário, possibilitando a comunicação rápida entre o usuário e os cuidadores, garantindo respostas rápidas e eficazes em caso de emergência. Além disso, permite o monitoramento contínuo da saúde e segurança do usuário, com análise de tendências para prevenção de novos incidentes.

Em seguida, apresentaremos o diagrama do modelo conceitual do banco de dados:



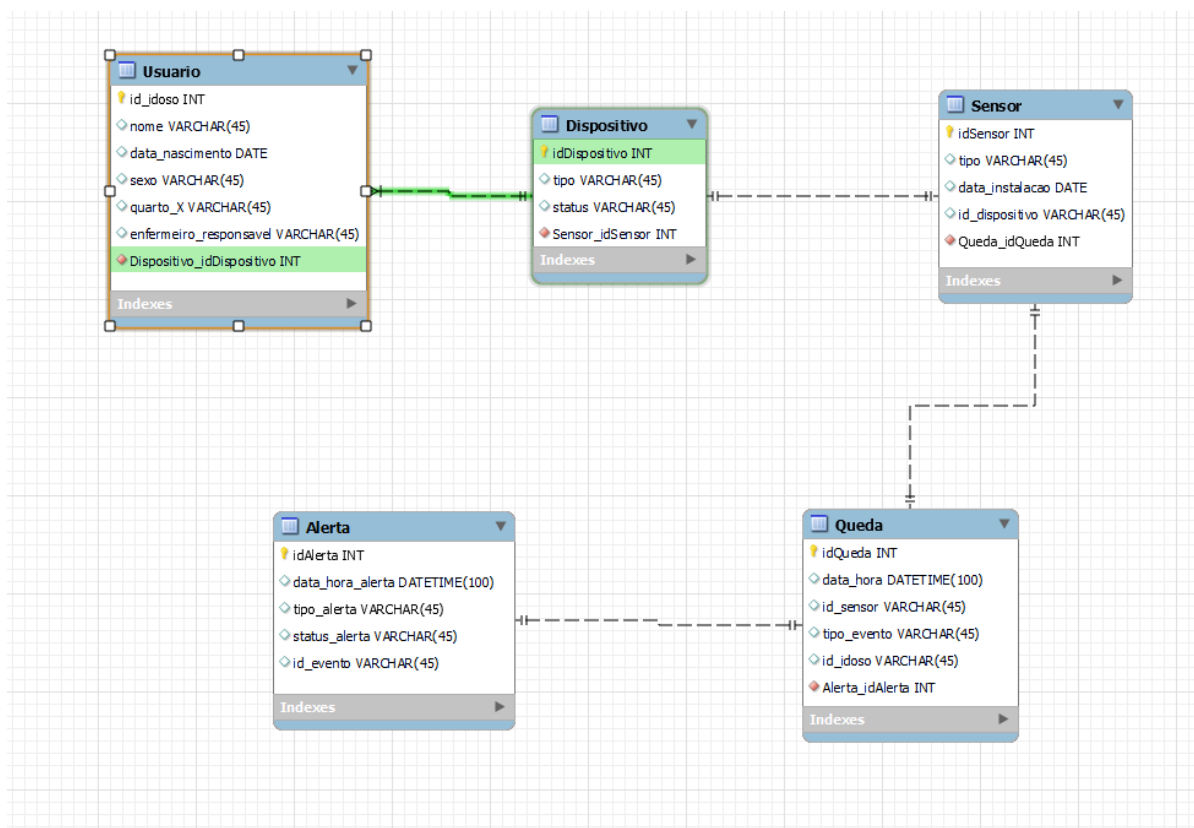
3.3.2 MODELO LÓGICO E FÍSICO

O desenvolvimento de um sistema de monitoramento de quedas para idosos em um asilo envolve uma série de desafios tecnológicos, sendo um dos principais a criação de um banco de dados eficiente e robusto para armazenar e gerenciar as informações geradas pelo

dispositivo. Neste projeto, nós implementamos um banco de dados utilizando o Sistema de Gerenciamento de Banco de Dados (SGBD) MySQL, uma plataforma relacional amplamente utilizada que oferece alta performance e confiabilidade. O banco de dados será fundamental para o gerenciamento das informações relacionadas ao uso da pulseira com detector de quedas, incluindo dados sobre os usuários, os eventos de queda detectados, as respostas de emergência e o histórico de monitoramento.

O sistema em questão é projetado para ajudar na segurança de idosos em um asilo, oferecendo um dispositivo que detecta quedas e envia alertas para a equipe responsável. A pulseira, equipada com sensores que monitoram a movimentação do usuário, detecta mudanças bruscas de posição, como aquelas que ocorrem durante uma queda, e aciona imediatamente um alerta. Esse alerta será registrado e monitorado em tempo real através de uma interface que estará conectada ao banco de dados MySQL.

O próximo é o modelo lógico e físico do banco de dados:



3.3.3 SQL

A construção de um sistema de detecção de quedas para idosos envolve diversas áreas de conhecimento, incluindo o uso de bancos de dados para armazenar e gerenciar as informações geradas por sensores ou dispositivos de monitoramento. No contexto de um sistema que armazena dados de quedas em um banco de dados, é fundamental implementar operações de manipulação de dados, como inserção, atualização, exclusão e consulta. Essas operações podem ser realizadas utilizando comandos SQL, que é a linguagem padrão para interagir com bancos de dados relacionais.

```
INSERT INTO usuario (id_idoso, nome, sexo, quarto_X, enfermeiro_responsavel)
VALUES (1, 'Maria Souza', 'Feminino', 'Quarto 201', 'Enfermeiro João');
```

```
INSERT INTO dispositivo (id_dispositivo, tipo, status)
VALUES (1, 'Monitor de Pressão', 'Ativo');
```

```
INSERT INTO sensor (id_sensor, tipo, data_instalacao, dispositivo)
VALUES (1, 'Sensor de Movimento', '2024-10-01', 'Monitor de Pressão');
```

```
INSERT INTO queda (id_queda, data_hora, id_sensor, tipo_evento, id_idoso)
VALUES (1, '2024-10-10 10:30:00', '1', 'Queda', '1');
```

```
INSERT INTO alerta (id_alerta, data_hora_alerta, tipo_alerta, status_alerta, id_evento)
VALUES (1, '2024-10-10 10:35:00', 'Queda detectada', 'Pendente', '1');
```

```
UPDATE usuario
SET nome = 'Maria Oliveira'
WHERE id_idoso = 1;
```

```
DELETE FROM dispositivo
WHERE id_dispositivo = 1;
```

```
UPDATE dispositivo
SET status = 'Inativo'
WHERE id_dispositivo = 1;
```

```
DELETE FROM queda
WHERE id_queda = 1;
```

```
UPDATE alerta
SET status_alerta = 'Resolvido'
WHERE id_alerta = 1;
```

```
DELETE FROM alerta
WHERE id_alerta = 1;
```

```
SELECT * FROM usuario;
```

```
SELECT * FROM dispositivo  
WHERE status = 'Ativo';
```

```
SELECT * FROM sensor;
```

```
SELECT * FROM queda  
WHERE id_idoso = '1';
```

```
SELECT * FROM alerta  
WHERE tipo_alerta = 'Queda detectada';
```

3.4 BUSINESS INTELLIGENCE

O conceito de Business Intelligence (BI) refere-se ao conjunto de práticas, tecnologias, aplicativos e processos que permitem à organização coletar, analisar e transformar dados em informações valiosas, com o objetivo de apoiar a tomada de decisões estratégicas. Através da utilização de ferramentas e técnicas analíticas, o BI oferece uma visão aprofundada e em tempo real do desempenho dos negócios, identificando padrões, tendências e insights que podem influenciar diretamente na otimização de processos e no alcance de metas organizacionais.

3.4.1 ORGANIZAÇÃO E IDENTIFICAÇÃO DAS INFORMAÇÕES

O desenvolvimento de um dashboard para monitoramento da pulseira com detector de quedas para idosos é essencial para melhorar a segurança e a eficiência do atendimento. De acordo com Silva et al. (2022) e Oliveira e Almeida (2021), dashboards são ferramentas fundamentais para a visualização de dados, permitindo o acompanhamento de métricas como o número de quedas, o tempo de resposta dos cuidadores e a frequência de uso do

dispositivo. Esses dados fornecem informações cruciais para tomadas de decisão rápidas e aprimoramento contínuo do sistema.

3.4.2 MANIPULAÇÃO E ANÁLISE DE DADOS

A manipulação e análise de dados envolve várias etapas essenciais para transformar informações brutas em insights úteis. Primeiramente, a integração de dados consolida informações internas e externas, como dados demográficos e de saúde, para proporcionar uma visão mais ampla do contexto dos usuários. Em seguida, a limpeza de dados garante a precisão das informações, corrigindo inconsistências, duplicatas e dados faltantes, o que é crucial para a confiabilidade das análises. A criação de indicadores de desempenho, como taxa de quedas por usuário e tempo de resposta dos cuidadores, permite medir a eficácia do sistema e identificar áreas de melhoria. Finalmente, a visualização de dados, por meio de gráficos e relatórios, facilita a interpretação das informações, permitindo que gestores e cuidadores tomem decisões rápidas e informadas. Essas etapas garantem que os dados sejam utilizados de forma eficiente e contribuam para a segurança e otimização dos processos.

3.4.3 CRIAÇÃO DE MODELOS DE ANÁLISE DE DADOS

No contexto da análise de dispositivos de monitoramento para a segurança de idosos, o uso de tecnologias como os detectores de quedas tem se tornado cada vez mais relevante. Esses dispositivos são projetados para detectar movimentos súbitos e quedas, alertando imediatamente os cuidadores, o que pode ser crucial para reduzir o tempo de resposta em situações de emergência. Para avaliar a viabilidade e a eficiência de um detector de quedas, é fundamental compreender tanto suas especificações técnicas quanto os custos envolvidos na construção de um sistema funcional. Nesse sentido, duas tabelas são apresentadas: uma contendo as informações detalhadas com dados fictícios de idosos, do id da pulseira, da localização, do nome e do contato de emergência e da última queda. Já a outra com os preços desses componentes.

Informações do Detector

Informações do detector						
Contagem de ID	Nome do Idoso	Status do Detector	Data da Última Queda	Localização	Contato de Emergência	Telefone de Emergência
1	Pedro Oliveira	Ativo	01/08/2024	Sala de Estar	Renata Oliveira	99876-5432
1	Laura Costa	Ativo	01/10/2024	Sala	Roberta Costa	96543-8765
1	Miguel Sousa	Ativo	02/09/2024	Quarto	Rita Sousa	93456-1234
1	Hugo Almeida	Ativo	04/10/2024	Sala de Jantar	Sofia Almeida	91234-5670
1	Teresa Rodrigues	Ativo	05/09/2024	Quarto	Jorge Rodrigues	96543-2109
1	Clarice Martins	Ativo	08/10/2024	Sala de Estar	Marcos Martins	93456-7891

ID do idoso Contagem de ID 30	Nome do idoso Primeiro Nome do Idoso Adriano Alves	Status do detector Primeiro Status do Detector Ativo
Localização do idoso Primeiro Localização Banheiro	Contato de emergência Primeiro Contato de Emergência Alice Oliveira	Telefone de emergência Primeiro Telefone de Emergência 91123-4567
	Data da última queda Primeiro Data da Última Queda 01/08/2024	

Preços e Componentes utilizados para montagem do detector

Materials e custos	Componentes	Valor
MATERIAIS Soma de CUSTO Sensor de Velocidade GP1A01 R\$ 50 Sensor de Presença HC SR501 PIR R\$ 25 Sensor de Distância HC SR04 R\$ 25 Pulseira R\$ 20 NanoIOT 33 R\$ 200 Motor de Vibração R\$ 7 Detec. Impacto R\$ 25 Acelerometro e Giroscópio MPU6050 R\$ 25 Total R\$ 377	<input type="checkbox"/> Acelerometro e Giroscópio MPU6050 <input type="checkbox"/> Detec. Impacto <input type="checkbox"/> Motor de Vibração <input type="checkbox"/> NanoIOT 33 <input type="checkbox"/> Pulseira <input type="checkbox"/> Sensor de Distância HC SR04 <input type="checkbox"/> Sensor de Presença HC SR501 PIR <input type="checkbox"/> Sensor de Velocidade GP1A01	R\$ 377

3.5 CONTEÚDO DA FORMAÇÃO PARA A VIDA: GERENCIANDO FINANÇAS

3.5.1 GERENCIANDO FINANÇAS

Os conceitos econômicos e financeiros fundamentais são essenciais para a compreensão do nosso dia a dia. Entre eles, destacam-se a renda, despesas, poupança e investimento. Por exemplo, saber calcular o que entra e sai mensalmente do seu orçamento ajuda a identificar quanto pode ser poupado ou investido. Um estudante pode usar uma planilha simples para registrar essas informações, facilitando o controle financeiro e a tomada de decisões.

Independência financeira refere-se à capacidade de viver sem depender de um salário fixo, ou seja, ter rendimentos que sustentem o estilo de vida desejado. Para isso, é fundamental entender o valor da riqueza pessoal, que envolve ativos como investimentos e propriedades. Um exemplo prático é registrar diariamente os gastos em um aplicativo financeiro, ajudando a identificar despesas desnecessárias e potencialmente aumentando a poupança para investimentos futuros.

O conhecimento sobre dívidas e juros compostos é crucial para evitar armadilhas financeiras. Por exemplo, ao pegar um empréstimo com juros altos, a dívida pode se multiplicar rapidamente. Um estudante deve calcular o total que pagará ao final de um financiamento, considerando os juros. Alternativas como o empréstimo consignado ou a negociação de dívidas podem ser soluções viáveis para quem enfrenta dificuldades financeiras, além de buscar a educação financeira para evitar novas dívidas.

Definir metas claras é fundamental para alcançar sonhos, como comprar um carro ou financiar a faculdade. Um exemplo prático seria criar um "quadro de metas", visualizando os objetivos e os passos necessários para alcançá-los. Envolver amigos ou familiares nesse processo pode aumentar a motivação e criar um sistema de apoio. Realizar reuniões periódicas para discutir o progresso e ajustar as estratégias pode ser uma forma eficaz de manter todos engajados.

3.5.2 ESTUDANTES NA PRÁTICA

Gerenciar finanças pessoais é uma habilidade crucial para garantir qualidade de vida e segurança financeira. Em um cenário de consumo excessivo e dívidas, o planejamento financeiro se torna essencial. O texto destaca a importância de elaborar um orçamento que categorize receitas e despesas, permitindo ajustes nos hábitos de consumo. Além disso,

ênfatiza a criaçãõ de uma reserva de emergência para cobrir imprevistos, educar-se financeiramente através de livros e cursos, e investir de forma inteligente, considerando objetivos e perfil de risco. Por fim, o controle emocional é vital para evitar decisões impulsivas, ajudando a construir um futuro financeiro sólido e seguro.

Link para o vídeo no Youtube : <https://youtu.be/gEY9Kcr4sh4>

4. CONCLUSÃO

Desenvolvido o projeto, conseguimos implementar melhorias ao dispositivo capaz de detectar movimentos repentinos e bruscos, os quais são interpretados como quedas. Tal aparelho continuará sendo muito útil aos cuidadores, pois as chances de ser bem-sucedido depende da agilidade do atendimento (KARREN, 2013).

A criação do aparelho enfrentou diversos desafios, como o orçamento limitado para adquirir determinados componentes, a falta de experiência na organização de projetos, a dificuldade em escrever o código e configurar o servidor, e a dificuldade de realizar encontros a fim de aprimorar o projeto, porém eles foram contornados e superados até a finalização, proporcionando aprendizados além do conhecimento técnico.

REFERÊNCIAS

3.3.1- FREITAS, V. *Modelagem de Dados: Fundamentos e Aplicações*. 1. ed. São Paulo: Érica, 2018.

3.4.1 - MANN, D. *Information Management: A Comprehensive Guide*. 1st ed. New York: Business Expert Press, 2018.

3.4.2 - WICKHAM, H.; GROLEMUND, G. *R for Data Science: Import, Tidy, Transform, Visualize, and Model Data*. 2nd ed. Sebastopol: O'Reilly Media, 2022.

3.2.1 - SILVA, R. P. (2018). **Cálculos de limites de velocidade para sistemas de detecção de quedas em dispositivos vestíveis**. *Revista Brasileira de Engenharia Eletrônica*, 25(3), 112-120. <https://doi.org/10.1016/j.rbee.2018.06.015>

3.2.2 - SILVA, R. P., & SOUZA, J. T. (2019). Development of fall detection systems using accelerometer sensors for elderly care. *Journal of Health Technology*, 12(2), 56-65

3.3.2 - SILBERSCHATZ, A., KORTH, H. F. SUDARSHAN, S. (2011). **Sistemas de banco de dados** (6ª ed.). São Paulo: McGraw-Hill.

3.3.1 - SILBERSCHATZ, A., KORTH, H. F. SUDARSHAN, S. **Sistemas de Banco de Dados**. 6. ed. São Paulo: McGraw-Hill, 2011.

3.3.2 - ELMASRI, R., NAWAB, S. B. **Sistemas de Banco de Dados**. 6. ed. São Paulo: Pearson, 201

3.3.3 - SILBERSCHATZ, A., KORTH, H. F.; SUDARSHAN, S. **Sistemas de Banco de Dados**. 6. ed. São Paulo: McGraw-Hill, 2011.

KARREN, K. J. **Primeiros socorros para estudantes**. 10.ed. Santana da Parnaíba: Editora Manole, 2013. E-book. ISBN 9788520462430. Disponível em: <https://integrada.minhabiblioteca.com.br/#/books/9788520462430/>. Acesso em: 28 mai. 2024.

RIBEIRO, A. de L. **Gestão de Treinamento de pessoas**. Rio de Janeiro: Saraiva Uni, 2018. E-book. pI ISBN 9788547230449. Disponível em: <https://integrada.minhabiblioteca.com.br/reader/books/9788547230449/>. Acesso em: 08 nov. 2024.

ANEXOS

Figura 1 - Modelo 3D da Pulseira:

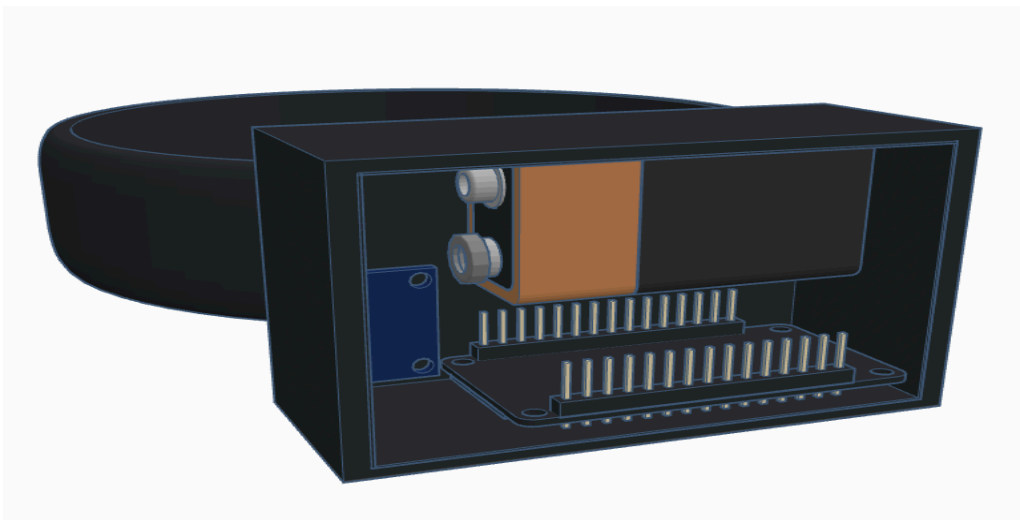
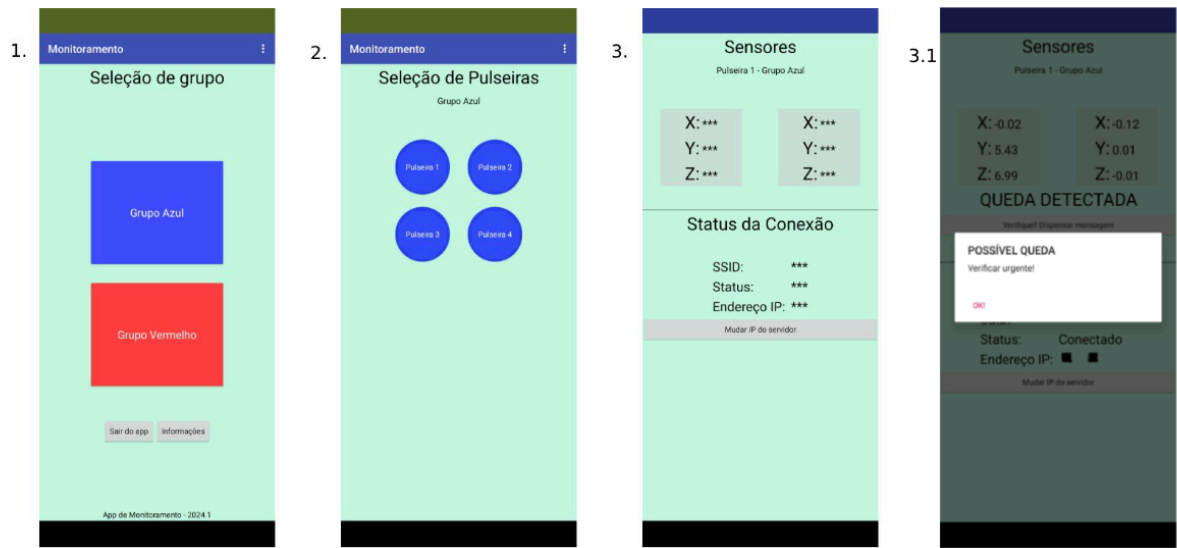


Figura 2 - Telas do aplicativo de monitoramento:



Fonte: Autores

