



UNIVERSIDADE FEDERAL DO CEARÁ – CAMPUS SOBRAL
CURSO DE ENGENHARIA DA COMPUTAÇÃO
DISCIPLINA: SOFTWARE EM TEMPO REAL
PROFESSOR: JERMANA LOPES DE MORAES

AZAEL FROTA VIANA GOMES - 475560
FRANK WILLIAM ARAUJO SOUZA - 473269
LUCAS JOSÉ LEMOS BRAZ - 471993
MARCOS VINICIUS RODRIGUES LIMA - 472828
OTONIEL SABINO BEZERRA DE FRAGA - 473043
RANIERY ALVES VASCONCELOS - 473532
VINÍCIUS COSTA DOS SANTOS - 473003
VITOR HUGO MUNIZ DE SOUSA SANTOS - 475767

DESCRITIVO - PLATAFORMA DE MONITORAMENTO DE SINAIS PARA PESSOAS EM LARES DE IDOSOS

SUMÁRIO

1. INTRODUÇÃO.....	3
2. MATERIAL UTILIZADO E TECNOLOGIAS UTILIZADAS.....	5
2.1 Material utilizado.....	5
2.2 Tecnologias utilizadas.....	5
3. JUSTIFICATIVA DO PROJETO COMO SOFTWARE EM TEMPO REAL.....	6
4. VISÃO GERAL.....	7
5. TAREFAS.....	8
5.1 Botão de Pânico.....	11
5.2 Oxímetro/Cardíaco (MAX30102).....	11
5.3 Sensor de Queda.....	12
5.4 Ler BPM (Pulse Sensor).....	12
5.5 Verificar conexão com Wifi.....	13
5.6 Verificar comunicação com Servidor.....	13
5.7 Comunicação com Servidor (Envio de dados).....	14
5.8 WATCHDOG.....	14
6. ESCALONAMENTO.....	15
7. FUNCIONAMENTO DOS CORES.....	15
8. IMPLEMENTAÇÃO DO HARDWARE E DO SOFTWARE.....	16
8.1 Hardware.....	16
8.2 Software.....	22
9. CONCLUSÃO.....	25
10. REFERÊNCIAS.....	27

1. INTRODUÇÃO

No cenário atual, com o envelhecimento da população em constante crescimento, surge a necessidade premente de abordar as complexidades e desafios que envolvem o cuidado e a segurança dos idosos. Com isso em mente, apresentamos um projeto de software inovador e fundamental: a Plataforma de Monitoramento de Sinais para Pessoas em Lares de Idosos.

O envelhecimento populacional é um fenômeno global que traz consigo inúmeros desafios. À medida que as populações idosas crescem, a necessidade de soluções que garantam seu bem-estar e segurança torna-se cada vez mais urgente. É com base nesse contexto que nosso projeto se destaca como uma resposta vital para melhorar a qualidade de vida e a segurança dos idosos em lares especializados.

A plataforma que desenvolvemos oferece uma gama de funcionalidades essenciais que visam proporcionar um ambiente mais seguro e acolhedor para os idosos. Entre as funcionalidades mais destacadas, encontramos o monitoramento contínuo de sinais vitais, como oximetria e batimentos cardíacos. Isso não apenas permite a detecção precoce de problemas de saúde, como também oferece uma sensação de segurança tanto para os idosos quanto para seus entes queridos, que podem acompanhar remotamente seus cuidados e bem-estar, como mostrada na figura 01.

Figura 01 - Plataforma



Fonte: Autores

Além disso, o sistema conta com um sensor de queda, que identifica quedas acidentais e pode acionar alertas automaticamente. Essa funcionalidade desempenha um papel crucial na prevenção de acidentes e no fornecimento de assistência imediata quando necessário. E para garantir que a ajuda esteja sempre ao alcance, implementamos um botão de pânico, permitindo que os idosos solicitem assistência imediata em situações de emergência, proporcionando tranquilidade a todos os envolvidos.

Essa plataforma não se limita apenas a monitorar os sinais vitais dos idosos; ela centraliza essas informações de maneira organizada e acessível. Isso significa que os profissionais de saúde e cuidadores podem tomar decisões informadas e fornecer cuidados personalizados com base nos dados coletados. Além disso, os familiares podem se sentir mais conectados e envolvidos no bem-estar de seus entes queridos, mesmo à distância.

A coleta desses sinais vitais é realizada por meio de uma luva equipada com os sensores adequados, conforme ilustrado na figura 02. Essa abordagem inovadora não apenas aprimora a precisão da monitorização, mas também proporciona uma experiência mais confortável e discreta para os idosos. Dessa forma, nossa plataforma representa não apenas um avanço tecnológico, mas um compromisso tangível com o cuidado e a segurança dos idosos em lares especializados.

Figura 02 - Luva



Fonte: Autores

2. MATERIAL UTILIZADO E TECNOLOGIAS UTILIZADAS

2.1 Material utilizado

- 1 Resistor de 220Ω ;
- 3 Resistores de $1k\Omega$;
- 2 Resistores de $10k\Omega$;
- 1 LED;
- 2 Push Button;
- 1 Buzzer ativo;
- 1 ESP-32;
- 1 ESP-8266 (NODEMCU);
- 1 PulseSensor;
- 1 MPU6050 (Acelerometro);
- 1 MAX30102.

2.2 Tecnologias utilizadas

- Linguagem C++;
- Linguagem Processing.

3. JUSTIFICATIVA DO PROJETO COMO SOFTWARE EM TEMPO REAL

O projeto da "Plataforma de Monitoramento de Sinais para Pessoas em Lares de Idosos" é categorizado como um sistema em tempo real devido a várias características distintas que o definem.

Em primeiro lugar, o sistema coleta e processa dados em tempo real, monitorando constantemente os sinais vitais, como oxigenação do sangue, batimentos cardíacos e a detecção de quedas. Essas medições ocorrem de maneira contínua e instantânea, permitindo uma avaliação constante do estado de saúde dos idosos. Isso é fundamental, uma vez que atrasos na aquisição ou processamento desses dados poderiam comprometer a segurança e o bem-estar dos idosos.

Além disso, o sistema oferece uma resposta imediata a eventos de emergência, com um botão de pânico que pode ser acionado pelos idosos em situações críticas. Quando esse botão é pressionado, o sistema reage instantaneamente, enviando um sinal de alerta para a plataforma. Esse tipo de resposta em tempo real é essencial em situações de emergência, onde cada segundo conta para a prestação de assistência adequada. O monitoramento contínuo e ativo dos sinais vitais dos idosos também é uma característica fundamental desse sistema em tempo real. Ele permite a identificação imediata de qualquer anomalia, possibilitando que a plataforma tome medidas proativas, como notificar cuidadores ou acionar dispositivos de segurança, evitando assim o agravamento de situações de risco.

Outra característica-chave é a integração de dados em tempo real. O sistema reúne e exibe os dados de vários sensores e dispositivos de maneira instantânea, proporcionando uma visão completa do estado de saúde de cada idoso. Essa visão abrangente auxilia os cuidadores na tomada de decisões informadas e na prestação de assistência oportuna.

Por fim, a confiabilidade e a disponibilidade desempenham um papel vital em sistemas em tempo real, como este. Qualquer falha ou interrupção na operação do sistema pode ter consequências sérias. Portanto, o projeto é estruturado de forma a garantir a operação contínua e a recuperação de falhas, garantindo que o monitoramento e a resposta a eventos críticos ocorram de maneira ininterrupta.

4. VISÃO GERAL

O projeto é uma solução vital para a segurança e o bem-estar dos idosos em ambientes de cuidados. Ele se destaca por sua capacidade reativa, sendo capaz de responder a eventos específicos que podem ocorrer a qualquer momento. Isso é essencial, pois permite a detecção de situações críticas, como quedas, e o acionamento imediato do botão de pânico pelos idosos.

Além disso, o projeto é caracterizado como um sistema Hard Real Time, o que significa que a resposta a eventos deve ocorrer em tempo real, com tolerâncias extremamente curtas. Qualquer atraso na resposta pode ter consequências graves para a segurança dos idosos. Essa característica é fundamental para garantir que as medidas de assistência sejam fornecidas sem demora, salvando vidas em situações de emergência.

A importância do tempo real está diretamente relacionada à sensibilidade do projeto a eventos e ao tempo. A detecção de eventos críticos, como quedas ou o acionamento do botão de pânico, é uma parte essencial do sistema, que deve responder imediatamente. Além disso, o monitoramento contínuo dos sinais vitais dos idosos também é sensível ao tempo, uma vez que qualquer atraso na detecção de alterações nesses sinais pode afetar sua segurança e saúde. Essa sensibilidade ao tempo e a eventos torna o projeto altamente eficaz em assegurar a proteção e o cuidado dos idosos.

No entanto, a natureza crítica do tempo no projeto também significa que as falhas podem ter consequências graves. Possíveis falhas no sistema incluem a falha na comunicação wi-fi, falhas na medição de sinais vitais e falhas na leitura dos sensores. Todas essas falhas são classificadas como críticas, uma vez que podem comprometer a segurança e o bem-estar dos idosos.

Para lidar com essas falhas críticas, estratégias de mitigação foram implementadas e pensadas, para a falha na medição dos sinais de batimentos cardíacos, temos dois sensores, caso haja uma falha na comunicação Wi-fi, existe um alerta emitido para o software, por fim, caso haja uma falha na leitura dos sensores, é retornado um erro e vai ocorrer um aviso ao servidor.

5. TAREFAS

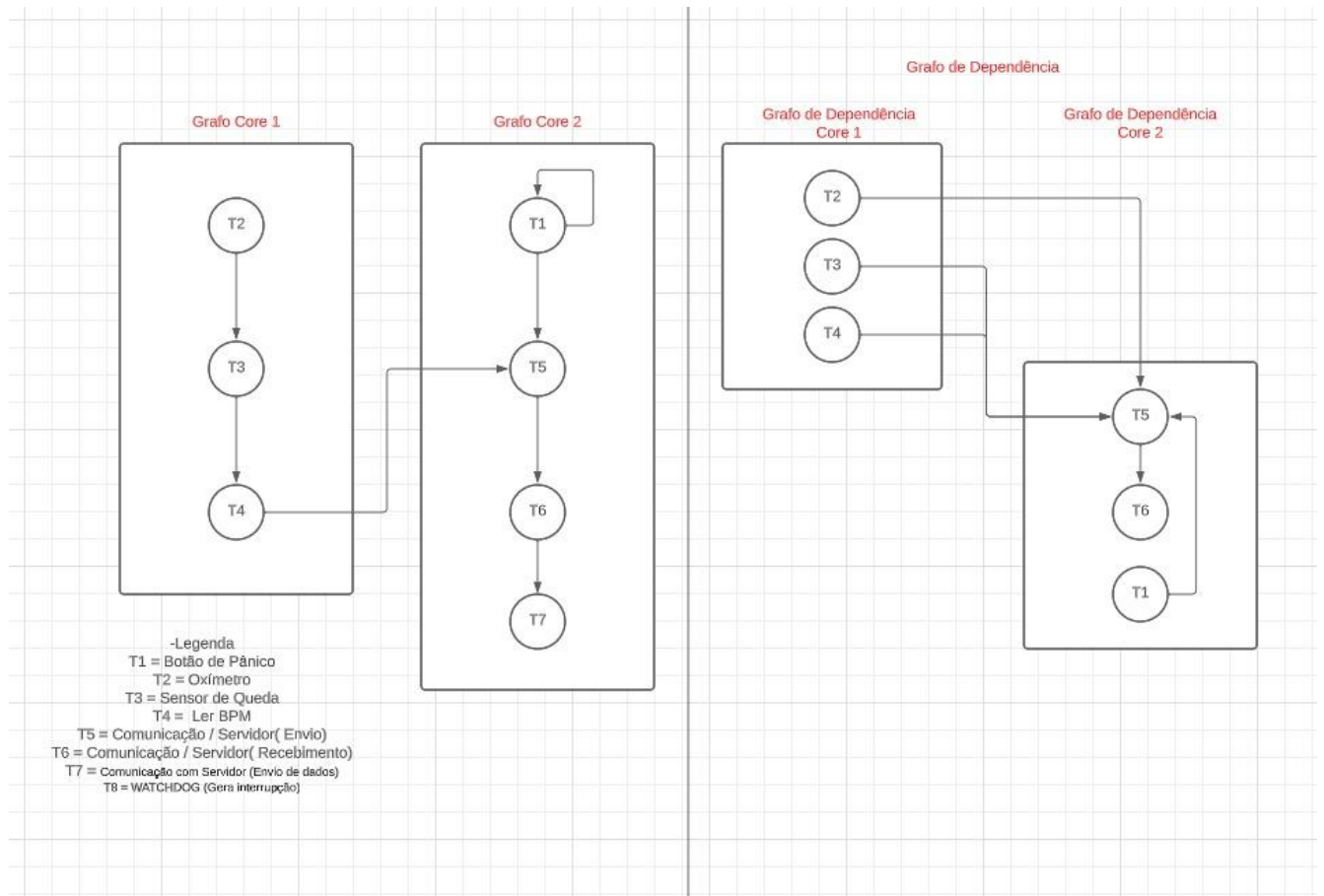
Em uma visão geral, nosso dispositivo se concentra em 8 tarefas pré-definidas: Botão de Pânico, Oxímetro/Cardíaco, Sensor de Queda, Ler batimento cardíaco por minuto, ou seja, o BPM (Pulse Sensor), Verificar conexão com Wifi, Verificar comunicação com Servidor, Comunicação com Servidor (Envio de dados) e por fim WATCHDOG. No contexto descrito, é fundamental compreender que cada uma das tarefas possuem distintas características que desempenham papéis cruciais em sua execução eficaz. Dentre essas características, destacam-se elementos como o Tipo de Aquisição de Dados, Criticidade, Prioridades, Deadlines, Periodicidade, Tempo de Computação, Tempo de Bloqueio, Jitter, BCET (Best Case Execution Time) e WCET (Worst Case Execution Time), Quantidade de Núcleos, Escalabilidade, Dependência e por fim Tipos de cargas. Considerando o exposto, pode-se definir cada uma dessas características já citadas da seguinte maneira:

- Tipo de Aquisição de Dados: Está relacionado ao método utilizado para obter informações necessárias. A escolha do método de aquisição de dados, Polling ou Interrupção, deve ser alinhada com os requisitos específicos de cada tarefa.
- Criticidade: Refere-se à importância de uma tarefa para o funcionamento global do sistema. Tarefas críticas são aquelas cuja execução é vital para o desempenho adequado do sistema como um todo.
- Prioridades: Determinam a ordem de execução quando várias tarefas estão aguardando processamento. A gestão eficaz de prioridades garante que as tarefas mais vitais sejam tratadas com a devida prioridade.
- Deadlines: São os limites de tempo estabelecidos para a conclusão de uma tarefa. São essenciais em cenários onde o cumprimento de prazos é crucial para o funcionamento adequado do sistema. Nesse contexto, pode-se destacar três categorizações principais de deadlines cada um com diferentes níveis de rigidez, visto isso, um deadlines deve ser definido por Hard (Refere-se a um prazo que é absolutamente inegociável e deve ser estritamente cumprido), Firm (Refere-se a um prazo que é importante e deve ser cumprido, mas pode haver alguma alteração em caso de circunstâncias imprevistas) ou Soft (Refere-se a um prazo que é mais flexível e pode ser estendido se necessário sem grandes impactos no projeto).
- Periodicidade: Representa a frequência com que uma tarefa é executada. Essa característica pode ser classificada em Periódica (Eventos ou fenômenos que ocorrem

em intervalos regulares e previsíveis), Esporádica (Eventos ou fenômenos que ocorrem de forma irregular e imprevisível) ou Aperiódica (Eventos ou fenômenos que ocorrem em intervalos não-lineares).

- Tempo de Computação: Refere-se à quantidade de tempo necessária para a conclusão de uma tarefa específica. É um indicador crítico da eficiência e desempenho da execução da tarefa.
- Tempo de Bloqueio (blocking time): é o intervalo de tempo durante o qual uma tarefa fica impedida de executar devido à ação de outra tarefa com prioridade mais alta.
- Jitter: é uma variação ou desvio na taxa de transmissão de dados em uma rede ou sistema de comunicação. Em outras palavras, é a flutuação no tempo de chegada de pacotes de dados que são transmitidos de um ponto a outro.
- BCET (Best Case Execution Time) e WCET (Worst Case Execution Time): elas se referem a dois tipos de estimativas de tempo de execução em um sistema computacional. Sendo assim, essas siglas podem ser definidas como Tempo de Execução Melhor Caso (BCET) e Tempo de Execução Pior Caso (WCET).
- Tipos de cargas: são termos frequentemente utilizados para descrever diferentes tipos de pressão sobre um sistema ou estrutura. Ademais, destaca-se dois perfis diferentes de pressão submetida: a Estática (Refere-se a uma carga que permanece constante e invariável ao longo do tempo) e Dinâmica (Refere-se a uma carga que varia a magnitude, direção ou frequência ao longo do tempo).
- Dependência: refere-se à relação entre duas ou mais tarefas/operações, onde uma delas precisa ser concluída antes que a outra possa começar a ser executada. No enquadramento da luva, tem-se o seguinte gráfico de dependência mostrado da figura 03.

Figura 03 - Gráfico de dependência



Fonte: Autores

É absolutamente crucial que todas essas propriedades sejam meticulosamente ajustadas para atender precisamente às necessidades e demandas específicas de cada tarefa. Essa abordagem minuciosa e altamente personalizada não apenas é fundamental, mas também é indispensável para garantir o funcionamento harmonioso e eficaz do sistema como um todo. A personalização cuidadosa dessas características não só melhora a eficiência operacional, mas também contribui significativamente para a otimização do desempenho global do sistema, assegurando que cada elemento esteja alinhado de maneira ideal com os requisitos particulares de cada tarefa. Essa atenção aos detalhes é um componente essencial para atingir níveis superiores de eficácia e eficiência em todas as operações.

5.1 Botão de Pânico

O botão de pânico é um dispositivo projetado para ser acionado pelo paciente em situações de emergência ou perigo iminente, com o objetivo de solicitar ajuda rapidamente. Nesse contexto, a tarefa de Botão de Pânico é considerada crítica e com máxima prioridade junto com o watchdog, tendo sua aquisição de dados distinta como interrupção, devido ao acionamento do botão ser um evento que não necessita de verificar ativamente em intervalos regulares e sim uma resposta a um evento. Além disso, tem uma frequência aperiódica de execução, visto que não ocorre em uma regularidade claro, ou seja, a tarefa pode acontecer a qualquer momento. Ademais, seus valores de Período, Deadline, Jitter, Tempo de Computação, tempo de bloqueio, BCET e WCET foram devidamente atribuídos de forma empírica por meio da utilização da linguagem python para a resolução dos calculados, sendo constituídos por: Período = N/A, Deadline = 1, Jitter = 0, Tempo de Computação = 0, tempo de bloqueio = N/A, BCET = (...) e WCET = (...). A tarefa de botão de pânico no dispositivo é independente, dessa forma ela poderá ser executada sem a necessidade de outra tarefa ser chamada primeiro. Por fim, identifica-se seu tipo de pressão sobre o sistema, ou seja, seu tipo de carga como estática, pois os requisitos de tempo são fixos e previsíveis.

5.2 Oxímetro/Cardíaco (MAX30102)

O oxímetro é um dispositivo médico utilizado para medir a quantidade de oxigênio no sangue, bem como a frequência cardíaca de uma pessoa. Nesse contexto, a tarefa de Oxímetro/Cardíaco é considerada não crítica e a segunda na escala de prioridade, tendo sua aquisição de dados distinta como polling devido precisar verificar regularmente o estado do oxímetro. Além disso, tem uma frequência periódica de execução, visto que ocorre em intervalos regulares e previsíveis. Ademais, seus valores de Período, Deadline, Jitter, Tempo de Computação, tempo de bloqueio foram devidamente atribuídos de forma empírica por meio da utilização da linguagem python para a resolução dos calculados, sendo constituídos por: Período = 3 s, Deadline = 3 s, Jitter = 0, Tempo de Computação = 5 ms, tempo de bloqueio = N/A. A tarefa de Oxímetro/Cardíaco no dispositivo acabou é independente, dessa forma ela poderá ser executada sem a necessidade de outra tarefa ser chamada primeiro. Por fim, identifica-se seu tipo de pressão sobre o sistema, ou seja, seu tipo de carga como estática, pois os requisitos de tempo são fixos e previsíveis.

5.3 Sensor de Queda

O sensor de queda é um dispositivo projetado para detectar uma queda ou mudança abrupta na posição de uma pessoa. Nesse contexto, a tarefa de sensor de queda é considerada crítica e a terceira na escala de prioridade, tendo sua Aquisição de dados distinta como polling devido ser necessário verificar regularmente o estado de queda e não queda do indivíduo. Além disso, tem uma frequência periódica de execução, visto que ocorre em intervalos regulares e previsíveis. Ademais, seus valores de Período, Deadline, Jitter, Tempo de Computação, tempo de bloqueio foram devidamente atribuídos de forma empírica por meio da utilização da linguagem python para a resolução dos calculados, sendo constituídos por: Período = 3.5 s, Deadline = 3.5 s, Jitter = 0, Tempo de Computação = 2 ms, tempo de bloqueio = N/A. A tarefa de sensor de queda no dispositivo é independente, dessa forma ela poderá ser executada sem a necessidade de outra tarefa ser chamada primeiro. Por fim, identifica-se seu tipo de pressão sobre o sistema, ou seja, seu tipo de carga como estática, pois os requisitos de tempo são fixos e previsíveis.

5.4 Ler BPM (Pulse Sensor)

O Pulse Sensor (sensor de pulso) é um dispositivo projetado para medir a frequência cardíaca de uma pessoa. Nesse contexto, a tarefa de Ler BPM é considerada crítica e a quarta na escala de prioridade, tendo sua aquisição de dados distinta como polling devido precisa verificar regularmente o BPM do paciente. Além disso, tem uma frequência periódica de execução, visto que ocorre em intervalos regulares e previsíveis. Ademais, seus valores de Período, Deadline, Jitter, Tempo de Computação, tempo de bloqueio foram devidamente atribuídos de forma empírica por meio da utilização da linguagem python para a resolução dos calculados, sendo constituídos por: Período = 3 s, Deadline = 3 s, Jitter = 0, Tempo de Computação = 1 ms, tempo de bloqueio = N/A. A tarefa de Ler BPM no dispositivo é independente, dessa forma ela poderá ser executada sem a necessidade de outra tarefa ser chamada primeiro. Por fim, identifica-se seu tipo de pressão sobre o sistema, ou seja, seu tipo de carga como estática, pois os requisitos de tempo são fixos e previsíveis.

5.5 Verificar conexão com Wifi

A função de verificar conexão com wifi testa se o dispositivo está conectado a uma rede sem fio e se essa rede tem acesso à internet. Nesse contexto, a tarefa de verificar conexão com Wifi é considerada não crítica e a quinta na escala de prioridade, tendo sua aquisição de dados distinta como polling devido à urgência de apurar a conexão do dispositivo com o Wifi. Além disso, tem uma frequência periódica de execução, visto que ocorre em intervalos regulares e previsíveis. Assim, seus valores de Período, Deadline, Jitter, Tempo de Computação, tempo de bloqueio foram devidamente atribuídos de forma empírica por meio da utilização da linguagem python para a resolução dos calculados, sendo constituídos por: Período = 400 ms, Deadline = 400 ms, Jitter = 0, Tempo de Computação = N/A, tempo de bloqueio = N/A. Diferente das outras, a tarefa de verificar conexão com Wifi no dispositivo acabou tendo dependência de execução, sendo assim ela somente poderá ser executada se as tarefas anteriores forem executadas. Por fim, identifica-se seu tipo de pressão sobre o sistema, ou seja, seu tipo de carga como estática, pois os requisitos de tempo são fixos e previsíveis.

5.6 Verificar comunicação com Servidor

A função de verificar comunicação com servidor testa se o dispositivo está conectado ao servidor e se esse servidor está respondendo. Nesse contexto, a tarefa de verificar comunicação com Servidor é considerada não crítica e a sexta na escala de prioridade, tendo sua aquisição de dados distinta como polling devido a urgência de apurar a conexão do dispositivo com o servidor. Além disso, tem uma frequência periódica de execução, visto que ocorre em intervalos regulares e previsíveis. Ademais, seus valores de Período, Deadline, Jitter, Tempo de Computação, tempo de bloqueio foram devidamente atribuídos de forma empírica por meio da utilização da linguagem python para a resolução dos calculados, sendo constituídos por: Período = 400 ms, Deadline = 400 ms, Jitter = 0, Tempo de Computação = 2 ms, tempo de bloqueio = N/A. A tarefa de verificar comunicação com Servidor no dispositivo acabou tendo dependência de execução, sendo assim ela somente poderá ser executada se a tarefa Verificar conexão com Wifi for chamada primeiro. Por fim, identifica-se seu tipo de pressão sobre o sistema, ou seja, seu tipo de carga como estática, pois os requisitos de tempo são fixos e previsíveis.

5.7 Comunicação com Servidor (Envio de dados)

A comunicação com o servidor para enviar dados envolve o estabelecimento de uma conexão entre o dispositivo e o servidor remoto para transmitir informações. Nesse contexto, a tarefa de Comunicação com Servidor (Envio de dados) é considerada crítica e a mínima prioridade, tendo sua aquisição de dados distinta como polling devido a obrigação de saber se todas as tarefas estão corretas para o envio dos dados. Além disso, tem uma frequência periódica de execução, visto que ocorre em intervalos regulares e previsíveis. Ademais, seus valores de Período, Deadline, Jitter, Tempo de Computação, tempo de bloqueio foram devidamente atribuídos de forma empírica por meio da utilização da linguagem python para a resolução dos calculados, sendo constituídos por: Período = 500 ms, Deadline = 500 ms, Jitter = 0, Tempo de Computação = 4 ms, tempo de bloqueio = N/A. A tarefa de Comunicação com Servidor (Envio de dados) no dispositivo não tem dependência de execução e não gera dependência de execução. Por fim, identifica-se seu tipo de pressão sobre o sistema, ou seja, seu tipo de carga como estática, pois os requisitos de tempo são fixos e previsíveis.

5.8 WATCHDOG

O "watchdog" (cão de guarda) é um mecanismo de segurança presente em muitos sistemas eletrônicos para monitorar e reiniciar automaticamente o dispositivo em caso de falhas ou travamentos. Nesse contexto, a tarefa de watchdog é considerada crítica e com máxima prioridade junto com o botão do pânico, tendo sua aquisição de dados distinta como interrupção devido ser a resposta de um evento de falha do sistema. Além disso, tem uma frequência esporádica de execução, visto que ocorre sem um padrão regular previsível, assim podendo acontecer em intervalos variáveis. Ademais, seus valores de Período, Deadline, Jitter, Tempo de Computação, tempo de bloqueio foram devidamente atribuídos de forma empírica por meio da utilização da linguagem python para a resolução dos calculados, sendo constituídos por: Período = N/A, Deadline = 2000 ms, Jitter = 0, Tempo de Computação = N/A, tempo de bloqueio = N/A. A tarefa de watchdog no dispositivo não tem dependência de execução e não gera dependência de execução. Por fim, identifica-se seu tipo de pressão sobre o sistema, ou seja, seu tipo de carga como estática, pois os requisitos de tempo são fixos e previsíveis.

6. ESCALONAMENTO

A plataforma de monitoramento de sinais para pessoas em lares de idosos é estrategicamente projetada com base no algoritmo Deadline Monotonic (DM), que desempenha um papel crucial em sistemas de tempo real. O DM é um algoritmo de escalonamento de prioridade fixa que atribui prioridades inversamente proporcionais aos prazos de execução das tarefas. Essa abordagem garante que tarefas com prazos mais curtos recebam prioridades mais elevadas, proporcionando previsibilidade e eficiência na execução das tarefas.

No contexto do DM, a plataforma adota uma abordagem de escalonamento online e estático, onde as prioridades são atribuídas no início da execução e não são alteradas dinamicamente durante o processo. A escalonabilidade dinâmica em relação aos prazos das tarefas assegura que aquelas com prazos mais curtos tenham prioridades mais altas, tornando o processo preemptivo e permitindo interrupções quando necessário.

A eficácia do DM na plataforma não se limita apenas aos benefícios técnicos, mas também simplifica a implementação e manutenção do sistema. A priorização com base em prazos críticos fortalece a confiabilidade do monitoramento de sinais vitais, garantindo a conclusão oportuna de tarefas essenciais. Esta abordagem é especialmente valiosa em um ambiente de monitoramento, onde a capacidade de resposta rápida é vital para a detecção imediata de eventos urgentes, como quedas ou acionamento de um botão de pânico.

Além disso, a plataforma destaca-se pela capacidade de adaptação dinâmica às necessidades emergentes e situações críticas, proporcionada pela aplicação eficaz do DM. O algoritmo, direcionado principalmente por tempo, contribui para a escalonabilidade de tarefas periódicas, garantindo a execução em intervalos regulares. A introdução de conceitos como deadline e períodos é fundamental para o funcionamento eficaz do DM, pois cada tarefa tem prazos e frequências específicos que precisam ser respeitados.

7. FUNCIONAMENTO DOS CORES

Sobre o funcionamento dos Cores do ESP-32, existem 2 Cores, (0 e 1), todas as tarefas estão rodando no Core 1, enquanto o Core 0 é responsável por lidar com a gestão das conexões Wi-Fi, a troca de dados com a rede e a manutenção da conectividade. Isso inclui atividades como a configuração e gerenciamento de conexões, transmissão e recepção de dados pela interface Wi-Fi. Ao atribuir essas tarefas ao Core 0, assegura-se que o

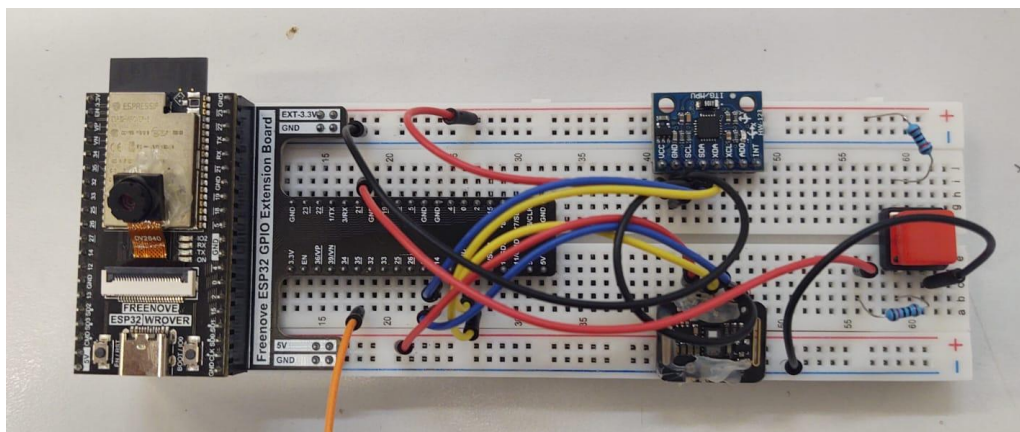
processamento associado à comunicação sem fio seja tratado de forma eficiente e sem interrupções, contribuindo para a estabilidade da conexão Wi-Fi.. Essa divisão permite que o projeto aproveite melhor os recursos do ESP-32, sem comprometer o desempenho ou a estabilidade da conexão sem fio.

8. IMPLEMENTAÇÃO DO HARDWARE E DO SOFTWARE

8.1 Hardware

A luva é um dispositivo wearable inovador, desenvolvido para monitorar de forma contínua tanto o batimento cardíaco quanto os níveis de oxigênio no sangue (oxímetro), para atingir esse propósito, ela integra dispositivos de medição: PulseSensor e MAX30102. Além dessas funcionalidades cruciais, ela incorpora um sistema de detecção de quedas ou impactos bruscos, bem como um botão manual de emergência que pode ser acionado em situações críticas, esse recurso utiliza um Push Button e um MPU6050 (acelerômetro), garantindo simplicidade e eficácia na ativação de alertas de emergência. Nesse contexto, pode-se observar a interação desses dispositivos na figura 04.

Figura 04 - Circuito da luva



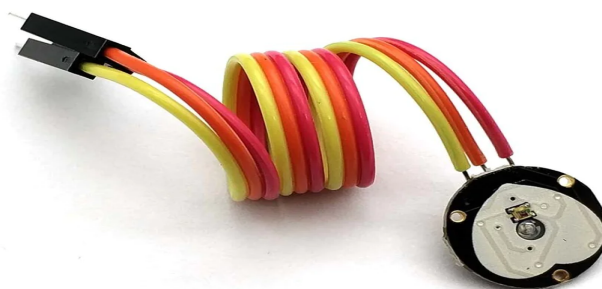
Fonte: Autores

Este dispositivo foi especialmente concebido para atender às necessidades da área da saúde, visando o monitoramento remoto de pacientes idosos. Equipada com sensores estrategicamente posicionados nos dedos e na palma da mão, a luva é capaz de medir com precisão diversos parâmetros vitais e transmitir essas informações aos cuidadores em tempo real.

Além disso, a luva é ergonômica e confortável, garantindo que os pacientes possam usá-la de maneira contínua sem interferir nas atividades diárias. Entretanto, é importante ressaltar que, em situações em que as mãos precisam ser molhadas, é recomendável retirar o dispositivo para preservar sua integridade e garantir o desempenho ideal. Seu design discreto e leve promove uma experiência de uso agradável, contribuindo para a aceitação e aderência por parte dos usuários.

De forma geral, o Pulse Sensor mostrado da figura 05, opera por meio de uma abordagem óptica, onde a luz desempenha um papel crucial na detecção das variações no volume sanguíneo decorrentes dos batimentos cardíacos. Equipado com um LED de luz vermelha, este sensor emite essa luz através da pele, enquanto um fotodiodo sensível é responsável por captar a luz transmitida ou refletida pelos tecidos adjacentes. À medida que o coração realiza sua função de bombear sangue, ocorrem flutuações no volume sanguíneo na área monitorada, influenciando diretamente a quantidade de luz absorvida pelo fotodiodo. O sinal resultante, capturado pelo fotodiodo, passa por um processamento meticuloso onde o sistema realiza uma série de operações estratégicas, cujo objetivo é extrair amostras dos dados que somente tenha informações cruciais relacionadas aos batimentos cardíacos. Esse procedimento ocorre a uma frequência notável, com sinais enviados ao Pulse Sensor a cada 2 milissegundos, para aquisição contínua de dados. Essa abordagem é fundamental para garantir a precisão e confiabilidade da leitura. Além disso, implementa-se um threshold, ou seja, um limite, para filtrar potenciais ruídos e atrasos, assim podendo assegurar que apenas dados relevantes sejam considerados, contribuindo para a eficaz eliminação de informações indesejadas da memória, além de garantir uma resposta ágil do sistema.

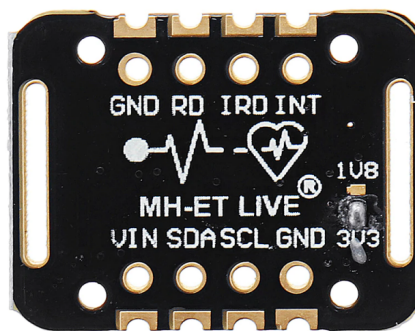
Figura 05 - Pulse Sensor



Fonte: [Heart Rate \(Pulse\) Sensor – Future Electronics Egypt \(fut-electronics.com\)](https://www.fut-electronics.com/heart-rate-pulse-sensor)

Já o MAX30102, demonstrado na figura 06, por sua vez, é um módulo sensor que expande a capacidade de monitoramento ao integrar LEDs tanto vermelho quanto infravermelho. Esses LEDs são essenciais para medir a absorção de luz pelo sangue e a quantidade de luz transmitida através do tecido. Ao utilizar dois comprimentos de onda distintos (vermelho e infravermelho), o MAX30102 consegue não apenas determinar a frequência cardíaca com precisão, mas também calcular a saturação de oxigênio no sangue. O sinal óptico resultante, capturado pelos LEDs e pelo fotodiodo, é submetido a um processo de processamento avançado, permitindo a extração eficiente dessas informações cruciais.

Figura 06 - MAX30102



Fonte: <https://www.xcluma.com/image/cache/catalog/products/BE-02-1941-800x800.png>

Ao capturar as informações do Max, que incluem o BPM (batimentos por minuto) e a saturação periférica, juntamente com os dados provenientes do Pulse Sensor é realizada uma série de procedimento como o intuito de obter a média da leitura dos dois dispositivos, posteriormente, essa média é encaminhada para o servidor, assim proporcionando uma representação mais estável e precisa das informações.

As informações coletadas pelo dispositivo podem ser acessadas e analisadas por meio de um computador, proporcionando uma visão abrangente do estado de saúde do paciente, além de garantir uma resposta imediata em casos de emergência.

Dessa forma, a luva não apenas oferece um acompanhamento contínuo e abrangente da saúde do paciente, mas também proporciona uma camada adicional de segurança por meio de suas funcionalidades de detecção de quedas e alerta de emergência. Este dispositivo representa um avanço significativo na tecnologia voltada para o cuidado de pacientes idosos, promovendo a tranquilidade tanto para os pacientes quanto para seus cuidadores.

O MPU6050 ou sensor de queda, demonstrado na figura 07, é um sensor inercial que combina as funcionalidades de um acelerômetro e um giroscópio em um único dispositivo. O acelerômetro é utilizado para medir mudanças na velocidade ou aceleração de um objeto, assim podendo ser usado para detectar movimentos, impactos e orientações em relação à gravidade. O giroscópio, por sua vez, mede a taxa de rotação angular em torno dos três eixos operando com base no princípio da conservação do momento angular, seu funcionamento envolve a detecção da mudança na orientação da microestrutura em resposta à rotação, neste processo, a microestrutura interna do giroscópio reage de forma sensível e imediata aos movimentos rotacionais, proporcionando leituras detalhadas sobre a velocidade angular experimentada pelo sensor. Essas leituras tanto do acelerômetro quanto do giroscópio são convertidas em sinais digitais e registradas para posterior processamento.

Figura 07 - MPU6050



Fonte: Autores

A conectividade sem fio integrada permite que os dados sejam enviados de forma rápida e segura para uma plataforma centralizada, acessível aos profissionais de saúde e cuidadores. Isso facilita a análise contínua das tendências de saúde do paciente, permitindo ajustes rápidos no plano de cuidados, se necessário.

A comunicação via Wi-Fi inicia uma tentativa de conexão e aguarda até que seja estabelecida para prosseguir. A configuração do Wi-Fi é realizada atualmente por meio da conexão serial, exigindo a interação física da placa com um computador para a configuração da rede. Ao entrar no modo de configuração (setup), indicado por duas piscadas da LED azul, o sistema entra em fase de configuração.

Primeiramente, realiza-se a calibração do acelerômetro, uma etapa que requer 3 segundos e demanda que a luva permaneça imóvel para assegurar uma calibração precisa. Posteriormente, a configuração do processador é executada. Caso essa etapa não seja concluída com sucesso, o sistema retorna zero, indicando uma dificuldade na configuração do processador. Isso pode resultar em falhas na execução de funções específicas ou na comunicação com outros dispositivos. Nesse contexto, é crucial revisar as configurações de rede e garantir a estabilidade da conexão Wi-Fi.

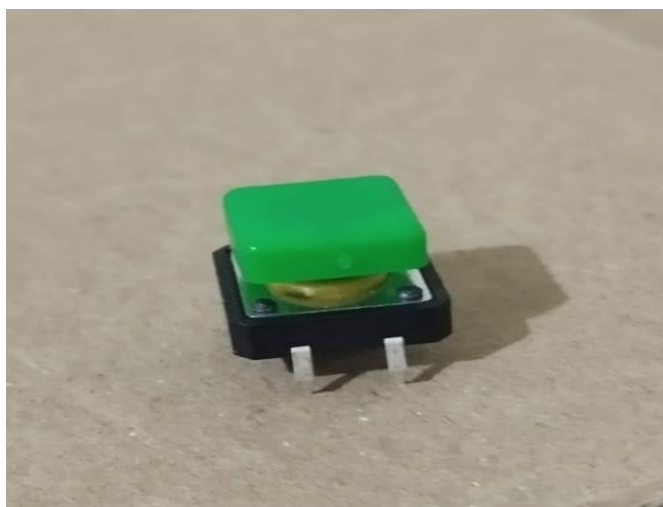
Se a configuração do processador for bem-sucedida, passa-se para a definição do botão. O botão opera em um circuito pull up, sendo ativado apenas ao detectar uma borda de descida. A ativação ocorre somente quando essa borda é identificada. Após a configuração do botão, atenção é dada à configuração do sensor de luminosidade Max30102. Durante a inicialização, é recomendável manter o dedo sobre o sensor por 4 segundos para permitir a calibração e ajuste à luminosidade ambiente.

Ao finalizar o processo de configuração e sair do setup, a placa emite três piscadas, indicando que todas as etapas foram concluídas com sucesso. Este procedimento não apenas estabelece uma comunicação eficaz via Wi-Fi, mas também configura elementos essenciais, como o botão e o sensor de luminosidade, garantindo um funcionamento adequado da luva em diferentes ambientes.

Além do monitoramento contínuo, a luva oferece a capacidade de acionar manualmente, com o botão de emergência, em caso de situações críticas. Esse recurso coloca o controle nas mãos dos usuários, permitindo uma resposta imediata em momentos de necessidade.

À vista disso, o Push Button, demonstrado na figura 08, é um dispositivo conectado a um circuito eletrônico interno à luva, assumindo o papel crucial de detectar o momento em que é pressionado. Quando o Push Button é acionado pelo paciente, o circuito eletrônico dentro da luva interpreta esse sinal como um comando significativo. Uma vez pressionado, a luva é capaz de enviar um sinal de alerta para cuidadores ou profissionais de saúde associados ao paciente. Essa comunicação rápida e eficiente possibilita uma resposta imediata às necessidades do usuário, promovendo um ambiente de cuidado mais ágil e personalizado.

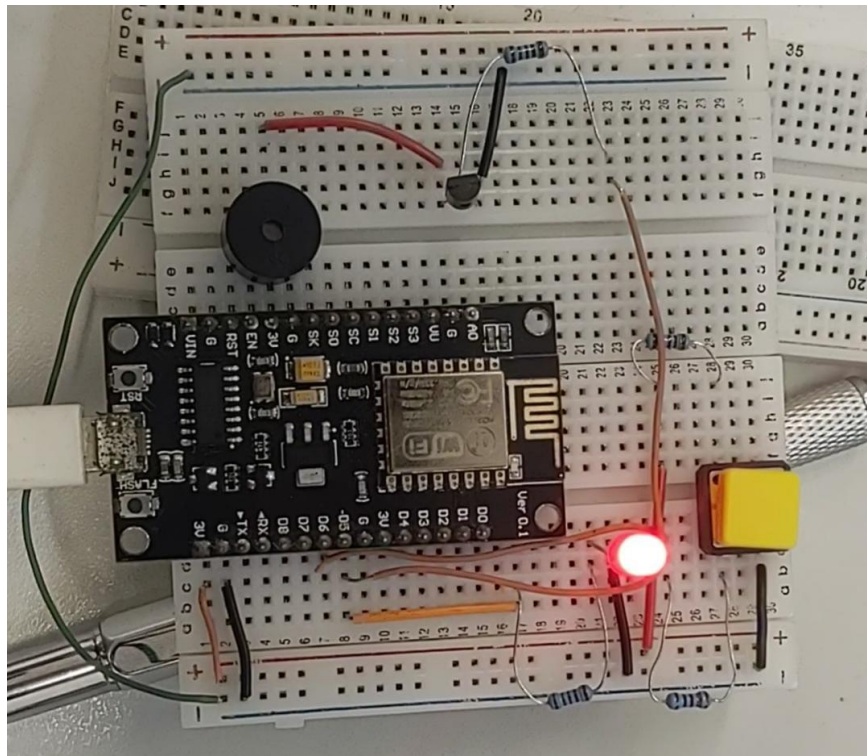
Figura 08 - Push Button



Fonte: Autores

Externo a luva, tem-se o bipper, um dispositivo simples que desempenha um papel crucial ao converter sinais elétricos provenientes da luva em ativações visuais do LED e sonora do buzzer, como essa representado na figura 09. Este sistema de sinalização, é acionado em situações de emergência como quedas, irregularidades nos batimentos cardíacos, irregularidades na oxigenação sanguínea ou pressionamento do botão de pânico, proporciona um alerta visual e sonoro imediato aos cuidadores. Esses sinais são recebidos por meio do ESP8266 um microcontrolador com capacidade de comunicação Wi-Fi integrada, essa conexão é feita com o mesmo wi-fi do servidor da luva, assim sendo necessário serem feitas todas as verificações já executadas do outro dispositivo. Nesse ponto, ao ser estabelecida a conexão dos dispositivos, o bipper espera alguma informação do servidor. Quando o servidor detecta alguma irregularidade ou situação crítica, envia um string específico, como o número 1, para o NodeMCU, ou seja, o bipper. Ao receber essa string, o NodeMCU entra em ação imediatamente, acionado o buzzer para emitir sinais sonoros e fazendo com que a LED pisque de forma intermitente. O bipper permanece nesse estado exigindo a confirmação dada por meio do acionamento do push button presente na placa, ao ser acionado o push button gera uma interrupção, que por sua vez, interrompe o led e o buzzer para de transmitir som, garantindo assim o recebimento da informação e ciência da situação do idoso.

Figura 09 - Circuito do Bipe



Fonte: Autores

Por fim, quanto ao recebimento de informações do dispositivo, em linhas gerais, são enviados dados cruciais para o servidor. Esses incluem o BPM, resultante da média calculada, a saturação periférica de oxigênio, indicador se houve ou não queda representada por (0 ou 1) e o estado do botão de emergência, também codificado como (0 ou 1). No processo de envio desses dados é necessário verificar se a comunicação com o servidor está realmente funcionando, somente após a confirmação bem-sucedida, os dados são enviados para o dispositivo do cuidador de forma que essa abordagem garante que as informações vitais relacionadas ao estado de saúde do paciente sejam comunicadas de maneira confiável, possibilitando a tomada de decisões precisas e ações rápidas em caso de emergências ou condições críticas. À vista disso, a ênfase na comunicação direta com o servidor contribui para a robustez do sistema, assegurando que as informações sejam transmitidas de maneira consistente.

8.2 Software

A implementação deste sistema destina-se a oferecer uma solução avançada e interativa para o monitoramento de idosos, combinando elementos gráficos, comunicação de rede e interatividade do usuário. O projeto de monitoramento de idosos, desenvolvido em

Processing, não apenas aborda as complexidades inerentes ao envelhecimento da população, mas também incorpora uma rica experiência visual para aprimorar a compreensão e interação. Primeiramente foi feito um primeiro esboço do que deveria ser feito, como primeiro ponto de partida, ilustrado na figura 10.

Figura 10 - Arte conceitual



Fonte: Autores

Dessa forma, foi feito o código na linguagem Processing onde teve classes, funções e métodos muito importantes para o funcionamento e a exibição da plataforma. A comunicação fluida com dispositivos externos, como a ESP-32, é assegurada pela classe Server. Essa camada integra dados externos, garantindo uma coleta contínua de informações vitais para o monitoramento dos idosos.

Sobre a classe User, existem métodos muito importantes que merecem ser citados. O método status eleva a comunicação visual ao traduzir dados complexos em representações gráficas envolventes. Utilizando códigos de cores e elementos visuais, este método oferece uma visão instantânea e compreensível do estado de saúde do idoso. Alertas visuais, como bradicardia e taquicardia, ampliam essa representação, adicionando uma camada crucial de discernimento.

A interatividade é aprimorada pelo método checkHover, que detecta a posição do cursor e promove respostas visuais dinâmicas. Ao destacar áreas específicas quando o mouse está sobre um usuário, este método contribui para uma experiência do usuário mais intuitiva.

Contudo, a estrela principal é o método `display`. Destacando-se no aspecto visual, este método oferece uma representação gráfica imersiva dos sinais vitais. Ao traduzir os batimentos cardíacos, níveis de SPO2, detecção de quedas e o estado do botão de pânico em elementos visuais, não apenas informa, mas cativa o usuário com uma visão compreensível do estado de saúde de cada idoso.

A estética visual do projeto é marcada por elementos gráficos distintivos. Cada usuário é visualmente representado por um retângulo único, criando uma identidade visual clara e facilitando a rápida identificação. Os nomes dos idosos são exibidos de maneira proeminente, contribuindo para uma identificação eficiente.

A representação gráfica dos dados vitais é enriquecida com cores dinâmicas, tornando a experiência visual não apenas informativa, mas também expressiva. Cores intensas, como verde para um estado saudável, amarelo para alertas moderados e vermelho para situações críticas, oferecem uma compreensão imediata do estado de saúde de cada usuário.

A funcionalidade `generateRandomData()` é estratégica, simulando dados realistas em ambientes de teste. Essa abordagem permite uma validação contínua do sistema, garantindo sua eficácia em diversos cenários antes da implementação em ambientes reais.

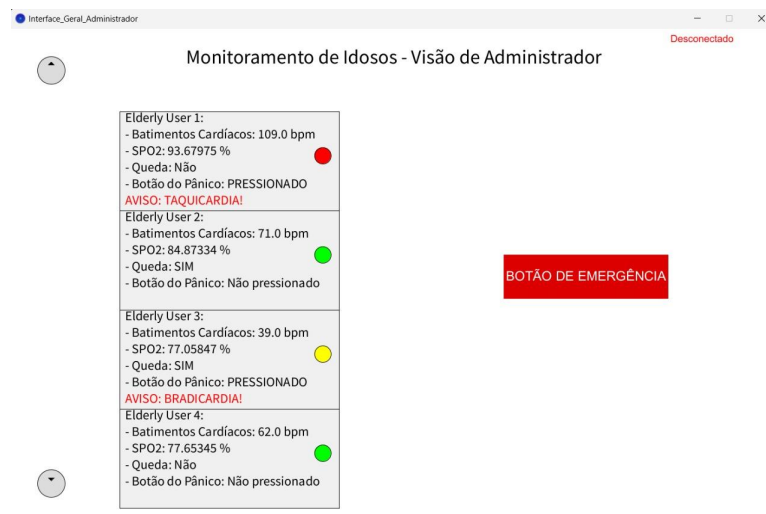
A atualização do status da conexão com a ESP-32, realizada pela função `updateEsp32ConnectionStatus`, é crucial para manter uma recepção contínua de dados externos, reforçando a confiabilidade do sistema na coleta de informações em tempo real.

O botão de emergência, representado pelo objeto `emergencyButton`, destaca-se como uma adição vital. Sua capacidade de resposta imediata a eventos críticos demonstra o comprometimento do sistema em lidar prontamente com emergências, proporcionando uma camada adicional de proteção para os idosos monitorados.

A lógica de exibição na interface gráfica reflete uma abordagem inteligente. A exibição de dados reais da ESP-32 no primeiro usuário, seguida por dados simulados nos usuários subsequentes, garante uma interface que reflete com precisão o estado dinâmico de cada idoso, oferecendo uma representação visual autêntica.

Este projeto transcende os limites acadêmicos, sendo uma iniciativa prática para enfrentar desafios reais no campo de cuidados a idosos. Ao destacar classes, métodos e funções essenciais, o código não apenas reflete inovação técnica, mas também expressa empatia, alinhando-se às crescentes demandas de uma sociedade em busca de soluções eficazes para o envelhecimento da população. Por fim, com a implementação completa foi gerada a plataforma final ilustrada na figura 11.

Figura 11 - Plataforma final



Fonte: Autores

9. CONCLUSÃO

Em conclusão, o desenvolvimento da "Plataforma de Monitoramento de Sinais para Pessoas em Lares de Idosos" representa um avanço notável na integração de tecnologia para melhorar o cuidado e a segurança dos idosos. Este projeto inovador responde de maneira proativa às demandas crescentes de uma população idosa em expansão, proporcionando uma solução abrangente e eficaz.

Através de uma combinação cuidadosa de hardware e software, a plataforma oferece monitoramento contínuo e em tempo real dos sinais vitais dos idosos, promovendo uma resposta imediata a situações críticas, além de utilizar algoritmo de escalonamento Deadline Monotonic (DM), um algoritmo de escalonamento de prioridade fixa, que atribui prioridades inversamente proporcionais aos prazos de execução das tarefas, proporcionando previsibilidade e eficiência na execução dessas tarefas cruciais. Ademais, a luva, equipada com sensores precisos e recursos de detecção de quedas, exemplifica a abordagem proativa adotada, enfatizando não apenas a precisão técnica, mas também o conforto e a usabilidade para os usuários.

A implementação estratégica de dois núcleos no ESP-32, dedicando um para o processamento de dados vitais e outro para a gestão de conexões Wi-Fi, reflete a otimização eficiente dos recursos disponíveis. Isso não apenas garante o desempenho estável da plataforma, mas também destaca a importância da confiabilidade em situações críticas.

A linguagem de programação C++ e a plataforma Processing são ferramentas úteis na concretização da visão do projeto. A representação visual dos dados, cores dinâmicas e a capacidade de simulação de dados realistas contribuem para uma experiência de usuário envolvente e informativa.

Ao abordar características críticas, como prioridades, deadlines, tipos de cargas e tempos de execução, a plataforma demonstra uma compreensão profunda das nuances do monitoramento em tempo real. A divisão eficiente de tarefas entre os núcleos do ESP-32, a simulação realista de dados e a resposta imediata a eventos críticos evidenciam uma abordagem abrangente para lidar com possíveis desafios.

Por fim, o código elaborado, as funcionalidades implementadas e a plataforma final apresentada refletem não apenas uma solução técnica, mas também um compromisso genuíno com o bem-estar dos idosos. Este projeto não apenas responde às necessidades atuais, mas também estabelece um precedente para o desenvolvimento de tecnologias dedicadas ao cuidado da população idosa em constante crescimento.

10. REFERÊNCIAS

1. Acelerômetro, disponível em: <https://mundoprojetado.com.br/acelerometro-mpu6050-arduino/> . Acesso em 18 de novembro de 2023.
2. Aspectos de um sistema de software em tempo real, disponível em: <https://www.facom.ufu.br/~ronaldooliveira/PDS-2019-1/Aula14-ProjetoTempoReal.pdf> . Acesso em 13 de novembro de 2023.
3. Escalonamento com deadline monotonic, disponível em: <https://www.ece.ufrgs.br/~fetter/eng04008/sched.pdf> . Acesso em 10 de novembro de 2023.
4. Funcionamento de cores do esp-32, disponível em: <https://curtocircuito.com.br/blog/conhecendo-esp32> . Acesso em 10 de novembro de 2023.
5. Modelos de tarefas, disponível em: https://www.cin.ufpe.br/~if728/sistemas_tempo_real/livro_farines/cap2.pdf. Acesso em 18 de novembro de 2023.
6. Polling/Interrupção, disponível em: <https://askanydifference.com/pt/difference-between-polling-and-interrupt/#:~:text=Polling%20refere-se%20ao%20processo%20de%20verifica%C3%A7%C3%A3o%20cont%C3%ADnu%C3%A7%C3%A3o,uma%20tarefa%20espec%C3%ADfica%20de%20um%20programa%20em%20execu%C3%A7%C3%A3o>. Acesso em 07 de novembro de 2023.
7. Pulse sensor, disponível em: <https://www.mundodaeletrica.com.br/o-que-sao-sensores-e-quais-as-suas-aplicacoes/>. Acesso em 16 de novembro de 2023.
8. Sensor de batimento cardíaco e oxímetro, disponível em: <https://medicalway.com.br/blog/oximetro-de-pulso-entenda-o-que-e-e-como-funciona/>. Acesso em 16 de novembro de 2023.