

**UNIVERSIDADE ESTÁCIO DE SÁ
NITERÓI I**

Detector de Queda pra Idosos

**Gabriel Guimarães Pereira Roif -- 202308267931
Lucas Diniz de Araujo Pereira – 202203897411
Paulo Vitor Silva Quintanilha – 202202780952
Nínive Fernandes de Farias Guimarães Natal – 202309146381**

Professor: André Przewodowski Filho

**2025
Niterói / Rio de Janeiro**

Sumário

| | |
|---|----|
| 1. DIAGNÓSTICO E TEORIZAÇÃO..... | 4 |
| 1.1. Identificação das partes interessadas e parceiros..... | 4 |
| 1.2. Problemática e/ou problemas identificados | 4 |
| 1.3. Justificativa | 5 |
| 1.4. Objetivos/resultados/efeitos a serem alcançados (em relação ao problema identificado e sob a perspectiva dos públicos envolvidos) | 6 |
| 1.5. Referencial teórico..... | 6 |
| 2. PLANEJAMENTO E DESENVOLVIMENTO DO PROJETO | 7 |
| 2.1. Plano de trabalho (usando ferramenta acordada com o docente)..... | 7 |
| 2.2. Descrição da forma de envolvimento do público participante na formulação do projeto, seu desenvolvimento e avaliação, bem como as estratégias pelo grupo para mobilizá-los. | 7 |
| 2.3. Grupo de trabalho (descrição da responsabilidade de cada membro) | 8 |
| 2.4. Metas, critérios ou indicadores de avaliação do projeto | 8 |
| 2.5. Recursos previstos | 8 |
| 2.6. Detalhamento técnico do projeto | 9 |
| 3. ENCERRAMENTO DO PROJETO | 21 |
| 3.1. Relato Coletivo:..... | 21 |
| 3.1.1. Avaliação de reação da parte interessada..... | 21 |
| 3.2. Relato de Experiência Individual (Pontuação específica para o relato individual) | 22 |
| 3.2.1. CONTEXTUALIZAÇÃO | 22 |
| 3.2.2. METODOLOGIA | 22 |
| 3.2.3. RESULTADOS E DISCUSSÃO: | 23 |
| 3.2.4. REFLEXÃO APROFUNDADA | 24 |
| 3.2.5. CONSIDERAÇÕES FINAIS | 25 |
| 4. BIBLIOGRAFIA..... | 25 |
| 5. ANEXOS | 26 |
| 6. REFERÊNCIAS..... | 27 |

1. DIAGNÓSTICO E TEORIZAÇÃO

1.1. Identificação das partes interessadas e parceiros

Em relação às partes interessadas, identificam-se como usuários diretos do projeto os idosos, especialmente aqueles com idade superior a 65 anos — faixa etária mais vulnerável a acidentes por queda — em foco ao sexo feminino, visto que apresentam uma maior expectativa de vida em comparação aos homens (QUEIROZ, 2023). O perfil socioeconômico do público-alvo abrange todas as classes sociais, desde que os indivíduos tenham acesso mínimo a tecnologias básicas, como telefone celular e conexão com a internet. Considera-se, ainda, o critério de alfabetização como requisito básico para a interação com a tecnologia proposta.

O senhor William Natal (de 41 anos) atua como cuidador de sua tia, uma idosa de 74 anos. Embora a referida idosa tenha optado por não participar ativamente do projeto por motivos pessoais, William se apresenta como comunicador direto da equipe e se compromete com o uso e avaliação da solução desenvolvida. A fim de oferecer melhores condições de monitoramento e segurança à sua tia, a participação de William enquanto parte interessada está registrada por meio de confirmação formal, conforme demonstrado no [Anexo 1](#).

1.2. Problemática e/ou problemas identificados

De acordo com Queiroz (2023), a queda é considerada um dos principais fatores de risco à saúde e à autonomia da população idosa. Tais acidentes frequentemente indicam fragilidade física e funcional, podendo resultar em sequelas permanentes, agravamento de doenças pré-existentes ou surgimento de doenças agudas. Segundo Muraro (2024), esse tipo de ocorrência representa uma ameaça significativa à qualidade de vida e à longevidade de pessoas idosas, principalmente quando não há assistência imediata.

Segundo Maciel (2010), é relatado que lesões acidentais figuram como a sexta principal causa de morte entre idosos, sendo as quedas responsáveis por aproximadamente 70% dos óbitos registrados na faixa etária acima de 75 anos. Os resultados indicam a gravidade da situação, exigindo estratégias eficazes de prevenção e resposta imediata.

No contexto da tia do senhor William, portadora de hérnia de disco há mais de uma década, possui uma condição de saúde que exige monitoramento constante, visto que qualquer acidente — especialmente uma queda sem atendimento imediato — pode acarretar consequências graves, inclusive risco de rompimento da coluna (ALMEIDA et al, 2014). Embora esteja em estágio avançado da doença, a paciente se recusa a permanecer em

repouso prolongado, o que eleva o risco de acidentes, demandando vigilância contínua do cuidador.

Entretanto, há limitações na obtenção de informações necessárias no momento da queda — tais como a confirmação do acidente e a localização da pessoa —. Devido a este atraso para ter acesso às informações, a eficiência do socorro e a tomada de decisão por parte dos cuidadores é comprometida.

Diante desse cenário, o projeto propõe o desenvolvimento de um dispositivo baseado em Internet das Coisas (IoT), com sensor de quedas integrado a módulo GPS, capaz de monitorar em tempo real a condição do usuário e enviar notificações automatizadas via e-mail e por meio do aplicativo Blynk. A solução visa reduzir o tempo de resposta em casos de emergência e, consequentemente, mitigar os impactos negativos das quedas sobre a saúde do idoso monitorado.

1.3. Justificativa

Como destacado no site *Governo do Estado do Paraná*, quase 1.200 idosos morreram de queda em 2023. Hoje, apesar de existirem soluções, elas acabam sendo muito caras, custando quase meio salário-mínimo. Segue uma figura de exemplo



Figura 1. Venda de pulseira para idosos. Fonte: “Loja da dryka”.

De acordo com a figura 1, dispositivos como pulseiras ou bengalas inteligentes podem não apresentar praticidade universal para a população idosa, especialmente para aqueles que não utilizam bengalas ou que possuem limitações motoras que dificultam o manuseio de equipamentos de pequeno porte.

A ausência de um sistema eficiente de detecção de quedas pode ocasionar diversas consequências negativas. Entre elas, destaca-se o atraso no atendimento de primeiros socorros, sendo o intervalo de tempo entre a queda e o recebimento de ajuda um fator crítico

para a saúde do idoso. Tal situação pode agravar-se, resultando em complicações médicas como fraturas, traumatismos e outras condições de saúde mais severas (SCIELO, 2020).

Além disso, a inexistência de uma resposta rápida às quedas pode comprometer significativamente a qualidade de vida do idoso, visto que o receio de cair novamente tende a restringir sua mobilidade e independência. Essa limitação também pode gerar sobrecarga aos cuidadores, que, na ausência de um sistema de alerta, são obrigados a manter vigilância constante sobre os idosos sob seus cuidados.

1.4. Objetivos/resultados/efeitos a serem alcançados (em relação ao problema identificado e sob a perspectiva dos públicos envolvidos)

O projeto é composto de um sistema baseado em Arduino, capaz de detectar quedas para majoritariamente pessoas idosas. A proposta consiste em identificar — por meio de sensores — uma queda, assim enviando posteriormente um alerta via *Cloud* para o celular do cuidador. O envio da notificação é realizado por meio do aplicativo *Blynk* e *Gmail*, acompanhado da localização em tempo real do dispositivo.

A iniciativa busca construir uma solução funcional que integre tecnologias de Internet das Coisas e computação em nuvem, promovendo a aplicação prática de conhecimentos adquiridos. Mesmo a partir de conceitos básicos, a proposta pretende oferecer uma solução mais acessível, que permita a transmissão quase instantânea de informações digitais, contribuindo para a segurança e a qualidade de vida do idoso.

Tem como objetivo planejar e projetar os componentes de software e hardware do sistema, garantindo a transferência adequada de dados no Arduino. Além disso, deve-se demonstrar o funcionamento do protótipo ao stakeholder, confirmando sua eficácia.

1.5. Referencial teórico

A ocorrência de quedas em idosos é amplamente reconhecida como um problema de saúde pública, uma vez que compromete diretamente a autonomia, a qualidade de vida e a segurança desse grupo etário. Segundo Perracini e Ramos (2002), as quedas representam uma das principais causas de mortalidade entre idosos, sendo responsáveis por hospitalizações frequentes, incapacidades físicas permanentes e até mesmo óbito, especialmente quando não há intervenção imediata. Tais eventos, além de físicos, podem desencadear impactos emocionais, como o medo de cair novamente, levando à redução da mobilidade e ao isolamento social.

Nesse contexto, torna-se essencial a adoção de medidas de prevenção e monitoramento, que integrem tecnologias assistivas ao cuidado cotidiano. De acordo com

Moraes et al. (2010), o uso de dispositivos inteligentes, baseados em tecnologias da informação e comunicação (TICs), tem se mostrado eficaz na promoção do envelhecimento ativo e seguro. Sistemas baseados em sensores de movimento, geolocalização e alertas automáticos possibilitam uma resposta rápida diante de situações emergenciais, como quedas inesperadas, contribuindo significativamente para a redução de complicações.

A proposta de desenvolvimento de um sensor de quedas integrado a um módulo GPS, conforme mencionado neste projeto, se encontra nas discussões de Azevedo e Bessa (2020), que defendem a inserção de tecnologias IoT (Internet das Coisas) na área da saúde como forma de ampliar o alcance dos cuidados e promover maior autonomia aos usuários. Esses autores destacam que soluções tecnológicas voltadas para o monitoramento remoto de pacientes possibilitam ganhos importantes no cuidado domiciliar, sobretudo quando se trata de indivíduos com mobilidade reduzida ou doenças crônicas.

Embora esteja em estágio avançado da doença, a paciente recusa-se a permanecer em repouso prolongado, o que eleva o risco de acidentes e demanda vigilância contínua por parte do cuidador (ALMEIDA et al, 2014).

2. PLANEJAMENTO E DESENVOLVIMENTO DO PROJETO

2.1. Plano de trabalho (usando ferramenta acordada com o docente)

Como mostrado no [anexo 2](#), cada tarefa contida no plano de trabalho tem um prazo estipulado para ser realizado, as atividades sendo em grande parte resolvidas de forma presencial e por chamadas no aplicativo de comunicação *Discord*.

2.2. Descrição da forma de envolvimento do público participante na formulação do projeto, seu desenvolvimento e avaliação, bem como as estratégias pelo grupo para mobilizá-los.

William foi considerado desde a fase inicial de formulação do projeto por meio de análises necessárias para o mesmo e suas limitações. Elas serviram como diretrizes para o desenvolvimento da solução tecnológica proposta. Embora tenha optado por não intervir diretamente no andamento das atividades até a finalização do protótipo, ele foi proporcionado informações do progresso do produto.

Dessa forma, a participação do stakeholder ocorreu de maneira estratégica, garantindo a aplicabilidade real do produto no contexto cotidiano.

2.3. Grupo de trabalho (descrição da responsabilidade de cada membro)

A equipe responsável pelo desenvolvimento do projeto de extensão foi composta por quatro alunos da Universidade Estácio de Sá (UNESA), cada um assumindo atribuições específicas no processo de execução:

Lucas Pereira estruturou e desenvolveu o código-fonte do produto. Nínive Natal atuou na montagem do Arduino e ficou encarregada na comunicação entre o stakeholder e o grupo. O discente Gabriel Roif foi responsável no teste do protótipo presencialmente, consertando diversos erros. Paulo Vitor Quintanilha cooperou na supervisão das tarefas dos outros integrantes.

Embora as tarefas tenham sido distribuídas conforme as aptidões individuais, todos os membros participaram ativamente de etapas coletivas, como a aquisição de componentes, reestrutura do código e montagem do protótipo com Arduino (que teve como colaboração outros estudantes da turma).

2.4. Metas, critérios ou indicadores de avaliação do projeto

A finalização da construção do dispositivo com Arduino deve ser entregue dia 9 de junho de 2025. Nessa etapa, será realizada a integração dos componentes GPS NEO-6M e do sensor de queda MPU-6050 ao microcontrolador ESP-32, estabelecendo a base física do protótipo. O código deve ser desenvolvido com objetivo de identificar a queda detectada e emitir alertas — por *E-mail* e *Blynk* — com a localização em tempo real do acidente.

2.5. Recursos previstos

Segue listado abaixo, todas as ferramentas, programas e peças necessárias para confecção deste projeto:

Discord – Usado para realizarmos nossas reuniões em grupo.

Microsoft Office – Usado para a confecção do roteiro de extensão.

Microsoft Excel – criação de planilhas.

Arduino IDE – Ambiente de desenvolvimento do código do arduino.

YouTube – para referências e ideias para o projeto.

Blynk – Terminal de Cloud IoT usado para armazenar as informações e a lógica de funcionamento da ESP32 com seus componentes.

ChatGPT – IA usada para consultas e sugestão de ideias.

Gmail – Usado ao receber o alerta para quando identificar a queda.

Livros disponibilizados pelo próprio plano de ensino da disciplina.

Peças utilizadas:

Módulo *Wi-Fi ESP32 Bluetooth* 30 pinos

Protoboard 830 pontos

Protoboard 400 pontos

Jumpers Macho/Macho - 80 Un

Jumpers Macho/Fêmea - 40 Un

Módulo GPS NEO 6M com Antena

Acelerômetro e Giroscópio 3 Eixos 6 DOF MPU-6050 – GY-521

Fita Dupla-Face

Placa de isopor

Valor dos gastos totais: R\$236,65

Devido ao gasto, foi decidido não pagar a versão *premium* do blynk, que dá acesso direto à localização do mapa, mostrar as coordenadas via *e-mail*, e afins.

As demais peças foram compradas no website *Eletrogate*. Disponível em: <https://eletrogate.com>

2.6. Detalhamento técnico do projeto

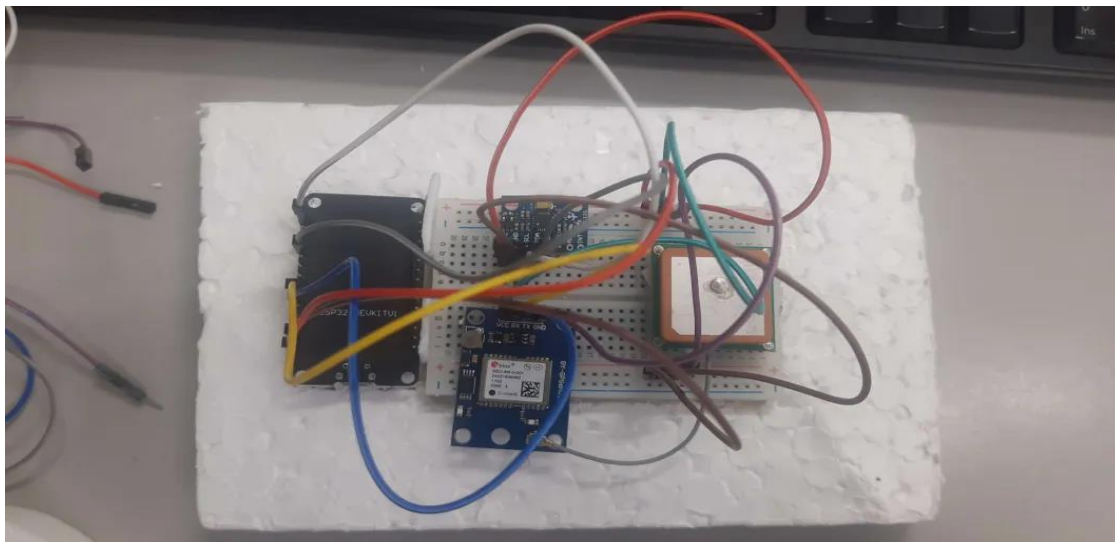


Figura 3: Esp32 montada na protoboard com os sensores instalados via jumpers.

Por meio da ferramenta *Canva* realizamos a ilustração do projeto com as peças montadas, de acordo com a figura 4:

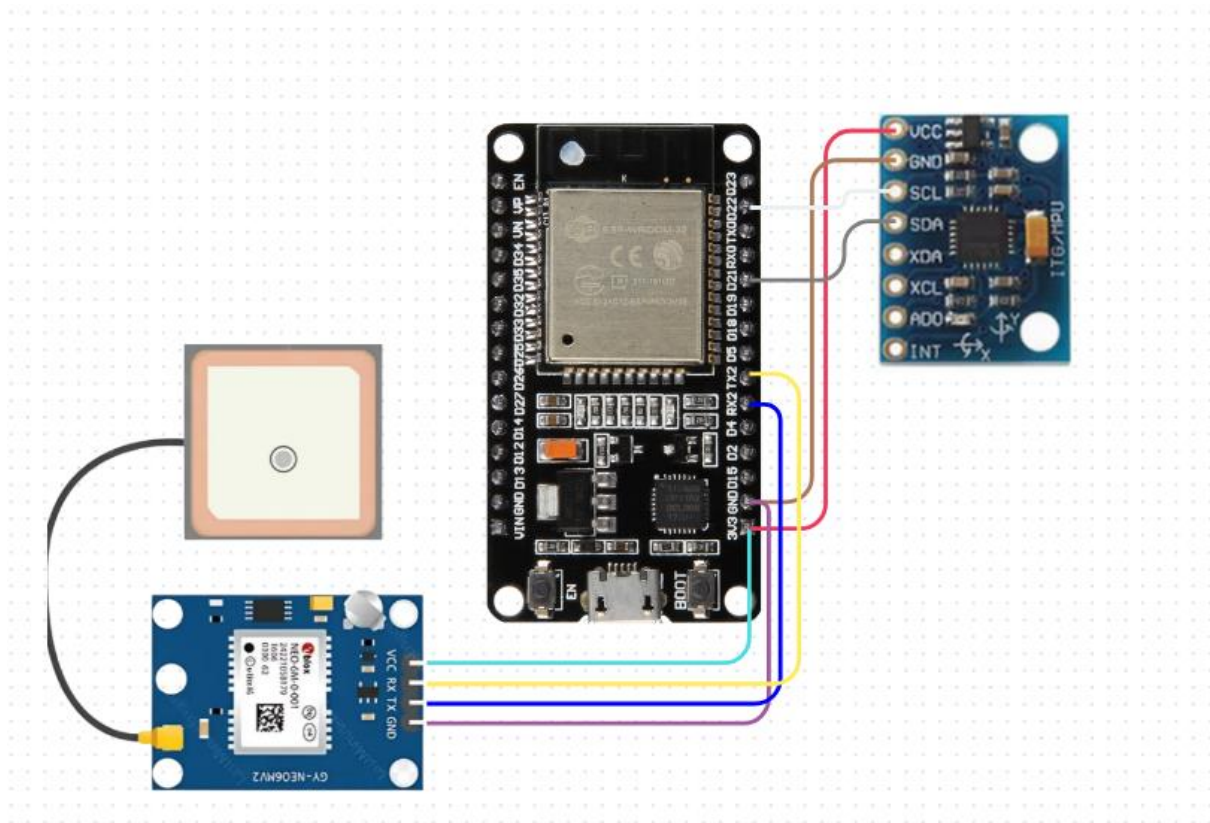


Figura 4: Representação da ESP32 feito no programa Canva.

Legenda do cabeamento: (ESP32 -> MPU6050)

- Vermelho: 3V3 -> VCC
- Marrom: GND -> GND
- Cinza: D21 -> SDA
- Branco: D22 -> SCL

Legenda do cabeamento: (ESP32 -> NEO6M)

- Verde: 3V3 -> VCC
- Roxo: GND -> GND
- Azul: RX2 -> TX
- Amarelo: TX2 -> RX

Funcionamento do Arduino ESP32:

O Arduino é um minicomputador capaz de processar informações enquanto conectado à Internet e Bluetooth. Ao receber o alerta do sensor, ele pede ao GPS sua localização, recebe-a e envia tais informações ao Blynk.

Funcionamento do sensor MPU6050:

É um dispositivo de *motion-tracking* - rastreamento de movimento - de baixo consumo de energia, possuindo os componentes de giroscópio, acelerômetro e medidor de temperatura. Porém, usamos apenas o giroscópio e o acelerômetro. Conectado ao Arduino, ele vai ter a responsabilidade de detectar as coordenadas X, Y, e Z. Usando as coordenadas X e Y – vertical e horizontal –, ele irá ativar quando o X for 6, alertando o Arduino.

Funcionamento do GPS NEO6M:

Sendo um dispositivo de localização, ele receberá um alerta do Arduino por causa do sensor de queda. Em seguida, irá enviar a latitude e longitude como *String* ao Arduino.

Funcionamento do Blynk:

O Blynk é uma plataforma IoT que serve como um terminal, interpretando as informações que vem do Arduino IDE e enviando à *Cloud*, assim permitindo o envio da notificação de queda via E-mail e celular.

Explicação do código:

Segue abaixo a explicação, linha-a-linha, do código fonte do projeto:

```
1  #define BLYNK_TEMPLATE_ID "T[REDACTED]3"
2  #define BLYNK_TEMPLATE_NAME "detector de queda"
3  #define BLYNK_AUTH_TOKEN "[REDACTED]"
4
5  #define BLYNK_PRINT Serial
6  #include <WiFi.h>
7  #include <WiFiClient.h>
8  #include <BlynkSimpleEsp32.h>
9  #include <Adafruit_MPU6050.h>
10 #include <Adafruit_Sensor.h>
11 #include <Wire.h>
12 #include <TinyGPS++.h>
13 #include <HardwareSerial.h>
14
15 char auth[] = BLYNK_AUTH_TOKEN;
16 char ssid[] = "[REDACTED]";
17 char pass[] = "[REDACTED]";
18
```

Figura 5: Início do código (linha 1 até 18).

Segundo referenciado na figura 5:

Linha 1-5: Pegando as informações configuradas no Blynk e definindo elas no código.

Linha 6-13: Importando as bibliotecas necessárias para o funcionamento da ESP32, Blynk, dos sensores e do módulo GPS.

Linha 15-17: Manda informações de conexão à internet para a ESP32, como o Token de autenticação para sincronização com Blynk, Nome do wifi e a senha.

```
19 // MPU6050
20 Adafruit_MPU6050 mpu;
21
22 // GPS
23 TinyGPSPlus gps;
24 HardwareSerial SerialGPS(2); // Usando UART2
25 #define GPS_RX 16 // RX do ESP32 (conecta no TX do GPS)
26 #define GPS_TX 17 // TX do ESP32 (conecta no RX do GPS)
27
28 BlynkTimer timer;
29 bool quedaDetectada = false;
30
31 void sendSensor()
32 {
33     // Lê dados do MPU6050
34     if (mpu.getMotionInterruptStatus()) {
35         sensors_event_t a, g, temp;
36         mpu.getEvent(&a, &g, &temp);
37
38         Serial.print("AccelX: "); Serial.print(a.acceleration.x); Serial.print(", ");
39         Serial.print("AccelY: "); Serial.print(a.acceleration.y); Serial.print(", ");
40         Serial.print("AccelZ: "); Serial.print(a.acceleration.z); Serial.print(" | ");
41         Serial.print("GyroX: "); Serial.print(g.gyro.x); Serial.print(", ");
42         Serial.print("GyroY: "); Serial.print(g.gyro.y); Serial.print(", ");
43         Serial.print("GyroZ: "); Serial.println(g.gyro.z);
```

Figura 6: Código (Da linha 19 até 43).

Segundo referenciado na figura 6:

Linha 20: criação do objeto chamado “mpu” da classe “Adafruit_MPU6050”. Esse objeto será usado para interagir com o sensor MPU6050, ler os dados de aceleração e giroscópio.

Linha 23: criação do objeto “gps” da biblioteca “TinyGPSPlus”, que serve para decodificar os dados NMEA recebidos do módulo GPS (latitude, longitude, etc.).

Linha 24: Cria um objeto “SerialGPS” da classe “HardwareSerial”, usando a UART2 do ESP32. O ESP32 tem 3 portas seriais (UART0, UART1, UART2), e aqui está sendo escolhido a UART2) para se comunicar com o módulo GPS.

Linha 25-26: Define os pinos que serão usados na UART2:

- GPS_RX (GPIO16): pino de recepção do ESP32 → ligado ao TX do GPS
- GPS_TX (GPIO17): pino de transmissão do ESP32 → ligado ao RX do GPS

Linha 28-29: Cria o objeto “timer” que será usado para executar tarefas em intervalos de tempo (como ler sensores periodicamente). Além disso, cria a variável “quedaDetectada”, que é um controle lógico (bool) para garantir que a queda só seja detectada uma vez e não repetidamente.

Linhas 31-36: Define uma função chamada “sendSensor”, que será chamada regularmente para Ler os dados do sensor MPU6050, detectar quedas e enviar as informações para o Blynk, incluindo a localização, se necessário.

“if (mpu.getMotionInterruptStatus()) {“- Verifica se houve uma interrupção por movimento, ou seja, se o sensor detectou aceleração suficiente para disparar o alerta, evitando ler o sensor o tempo todo, melhorando o desempenho.

“sensors_event_t a, g, temp”; - Cria três variáveis do tipo “sensors_event_t” para armazenar os dados:

- a: aceleração
- g: giroscópio
- temp: temperatura (não está sendo usada no projeto, mas está disponível)

“mpu.getEvent(&a, &g, &temp)”;- Lê os dados do sensor e armazena nas três variáveis acima.

Linhas 38 – 43: Essas linhas exibem no monitor serial os dados de aceleração e giroscópio:

- Mostra valores dos eixos X, Y e Z tanto da aceleração quanto da rotação.
- Útil para depuração e monitoramento em tempo real via cabo USB.

```

44 Blynk.virtualWrite(V0, a.acceleration.x);
45 Blynk.virtualWrite(V1, a.acceleration.y);
46 Blynk.virtualWrite(V2, a.acceleration.z);
47 Blynk.virtualWrite(V3, g.gyro.x);
48 Blynk.virtualWrite(V4, g.gyro.y);
49 Blynk.virtualWrite(V5, g.gyro.z);
50
51 if (a.acceleration.z <= 3.0 && a.acceleration.z > -3.0 && !quedaDetectada) {
52     quedaDetectada = true;
53     Serial.println(" ! Queda detectada!");
54
55     if (gps.location.isValid()) {
56         float lat = gps.location.lat();
57         float lng = gps.location.lng();
58
59         Serial.print("Localização: ");
60         Serial.print(lat, 6);
61         Serial.print(", ");
62         Serial.println(lng, 6);

```

Figura 7: Código (Da linha 44 até 62).

Segundo referenciado na figura 7:

Linha 44 - 49: Essas linhas enviam os dados do sensor para o app Blynk, usando as variáveis virtuais (V0 a V5):

- V0 -> Aceleração no eixo X
- V1 -> Aceleração no eixo Y
- V2 -> Aceleração no eixo Z
- V3 -> Giroscópio no Eixo X
- V4 -> Giroscópio no Eixo Y
- V5 -> Giroscópio no Eixo Z

Isso permite visualizar os valores ao vivo no aplicativo Blynk com gráficos e displays.

Linha 51-53: “if (a.acceleration.z <= 3.0 && a.acceleration.z > -3.0 && !quedaDetectada)” = Verifica se a aceleração no eixo Z (vertical) está entre -3.0 e 3.0. Isso indica uma possível queda, pois valores próximos de zero no eixo Z sugerem ausência de força contra o solo, como quando alguém está no ar ou caindo. Também verifica “!quedaDetectada” para garantir que a queda seja registrada apenas uma vez, evitando detecções repetidas.

Linha 55: “if (gps.location.isValid())” = Verifica se o módulo GPS tem um sinal válido no momento da queda. Só se essa condição for *true* (ou seja, o GPS captou a localização corretamente) o código segue para extrair latitude e longitude.

Linhas 56 – 62 (Bloco do GPS):

“float lat = gps.location.lat()”; / “float lng = gps.location.lng()”; - Obtém a latitude e longitude como float, diretamente da biblioteca “*TinyGPSPlus*”.

Bloco “Serial.print(...)” - Imprime no monitor serial a latitude e longitude com 6 casas decimais, que é o padrão de precisão GPS.

```
67
68     String url = "https://maps.google.com/?q=" + String(lat, 6) + "," + String(lng, 6);
69     Blynk.virtualWrite(V6, url);
70   } else {
71     Serial.println("✗ GPS sem sinal no momento.");
72     Blynk.virtualWrite(V6, "GPS sem sinal no momento.");
73   }
74 }
75 }
76 }
```

Figura 8: Código (da linha 67 até 76).

Segundo referenciado na figura 8:

Linha 68: Constrói uma *URL* do Google Maps com os dados de localização.

- Exemplo: “https://maps.google.com/?q=-23.564234,-46.654543”

Linha 69: “Blynk.virtualWrite(V6, url)”; - Envia essa URL com a localização para o *datastream* V6 no *Blynk*.

“else” (se não há dados de GPS): Exibe uma mensagem de erro no monitor serial e no *Blynk* caso o GPS ainda não tenha conseguido se conectar aos satélites.

```
78 void readGPS()
79 {
80   while (SerialGPS.available() > 0) {
81     gps.encode(SerialGPS.read());
82   }
83 }
```

Figura 9: Código (da Linha 78-83):

Essa função lê os dados recebidos pela UART2 (pinos 16 e 17) onde o módulo GPS está conectado. A biblioteca “*TinyGPS++*” processa os dados NMEA e os armazena no objeto gps.

```

85 void setup()
86 {
87   Serial.begin(115200);
88   delay(100);
89
90   WiFi.begin(ssid, pass);
91   Serial.print("Conectando ao Wi-Fi");
92   while (WiFi.status() != WL_CONNECTED) {
93     delay(500);
94     Serial.print(".");
95   }
96   Serial.println("\n✅ Wi-Fi conectado!");
97
98   Blynk.config(auth);
99   Blynk.connect();
100
101   // Inicializa MPU6050
102   if (!mpu.begin()) {
103     Serial.println("❌ Falha ao encontrar MPU6050");
104     while (1) { delay(10); }
105   }
106   Serial.println("✅ MPU6050 encontrado!");
107
108   mpu.setHighPassFilter(MPU6050_HIGHPASS_0_63_HZ);
109   mpu.setMotionDetectionThreshold(1);
110   mpu.setMotionDetectionDuration(20);
111   mpu.setInterruptPinLatch(true);
112   mpu.setInterruptPinPolarity(true);
113   mpu.setMotionInterrupt(true);
114
115   // Inicializa GPS na UART2
116   SerialGPS.begin(9600, SERIAL_8N1, GPS_RX, GPS_TX);
117   Serial.println("✅ GPS inicializado.");
118
119   // Temporizadores
120   timer.setInterval(500L, sendSensor); // Verifica queda a cada 0.5s
121   timer.setInterval(1000L, readGPS);   // Lê GPS a cada 1s
122 }

```

Figura 10: Código (da linha 85 – 123).

Linhas 85 – 122: Bloco “setup()” - inicializa *Wi-Fi*, *Blynk*, MPU6050, GPS e os temporizadores.

```

124 void loop()
125 {
126   Blynk.run();
127   timer.run();
128 }

```

Figura 11: Fim do código (linha 124 – 128).

Linhas 124 – 128: bloco “loop ()” - mantém o funcionamento do *Blynk* e dos timers.

Segue abaixo, a visão das informações através do terminal IoT *Blynk*:

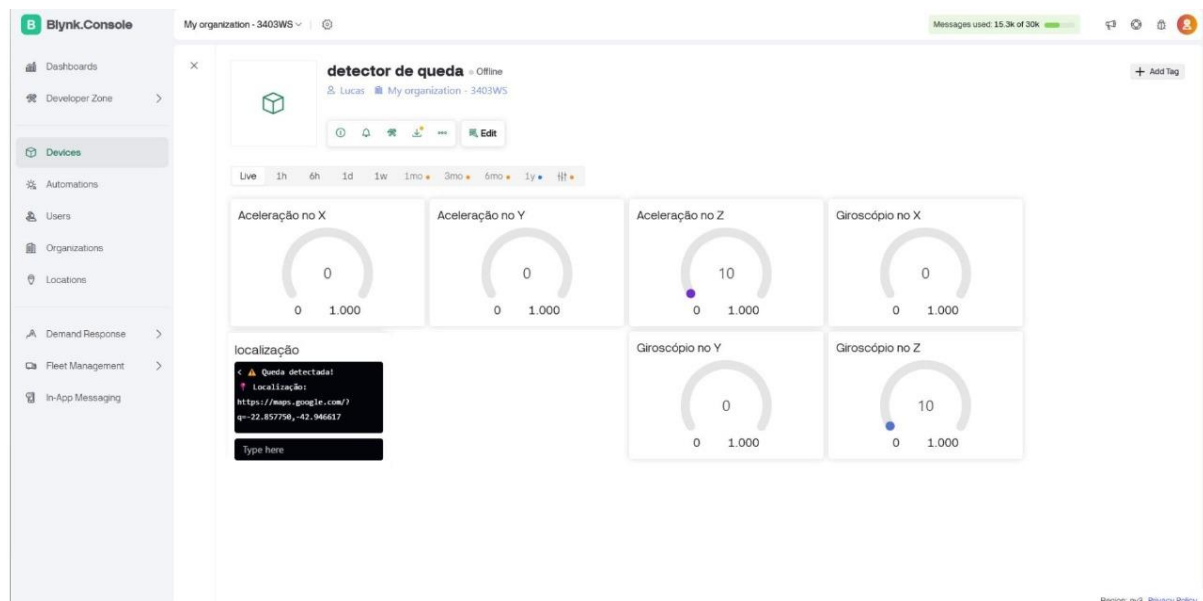


Figura 12: Visão das informações através do *blynk web*, após uma queda ter sido detectada.

Conforme mostrado na figura 12, neste painel o usuário terá acesso as informações em tempo real das coordenadas, e da localização de onde o arduíno está, em caso de queda. Também é através do *Blynk* que a lógica da detecção e notificação da queda é programada.

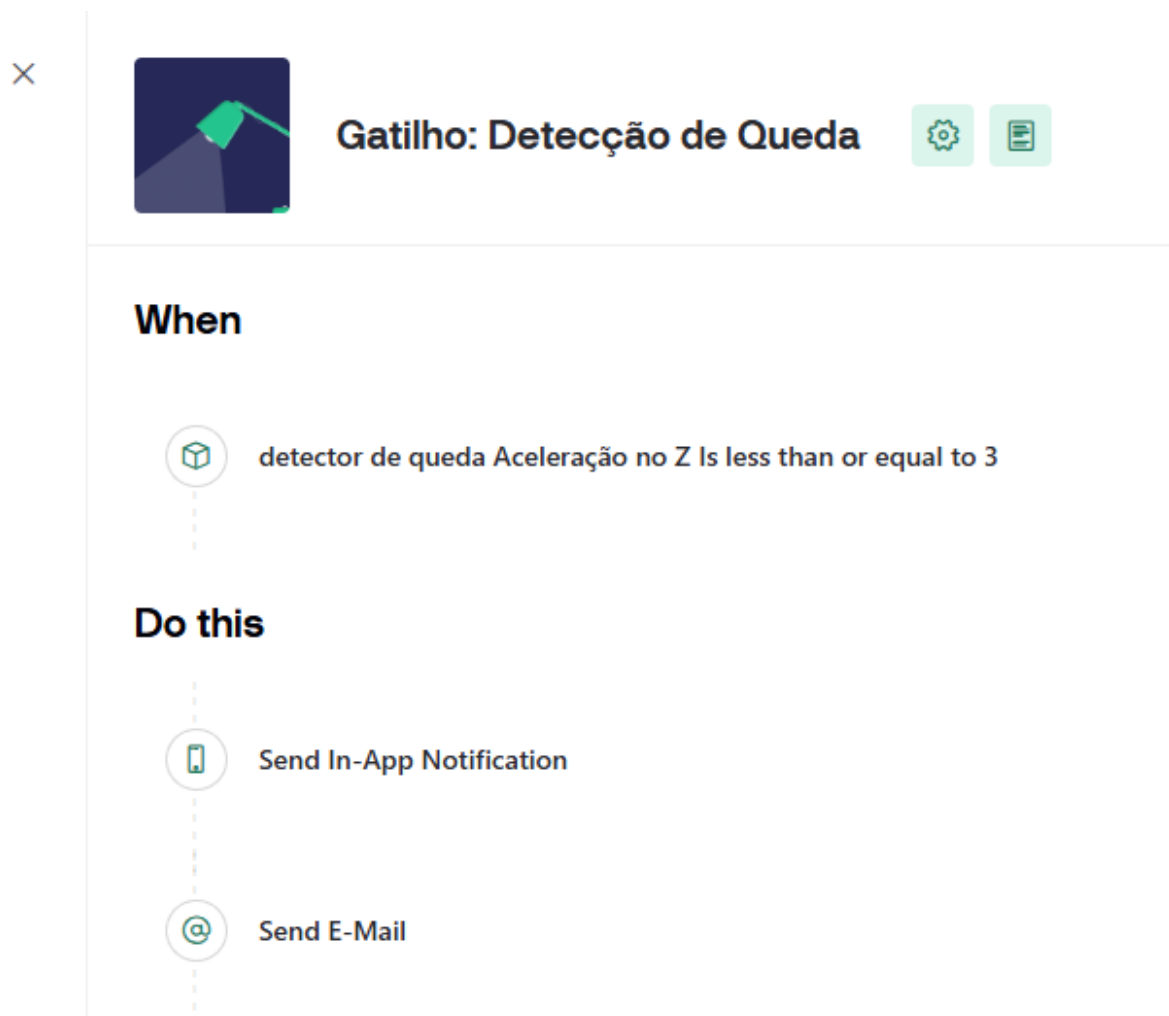


Figura 13: lógica de queda, programada através do Blynk.

De acordo com a figura 13, a lógica funciona da seguinte maneira: Quando a aceleração Z estiver menor ou igual a 3 (o que significa que um corpo se movimentou de forma brusca em direção ao chão), envia uma notificação ao aplicativo *mobile* do *Blynk*, mostrado na figura 14; E envia um *email* com a mensagem: “Uma queda foi detectada!”, ilustrado na figura 15.

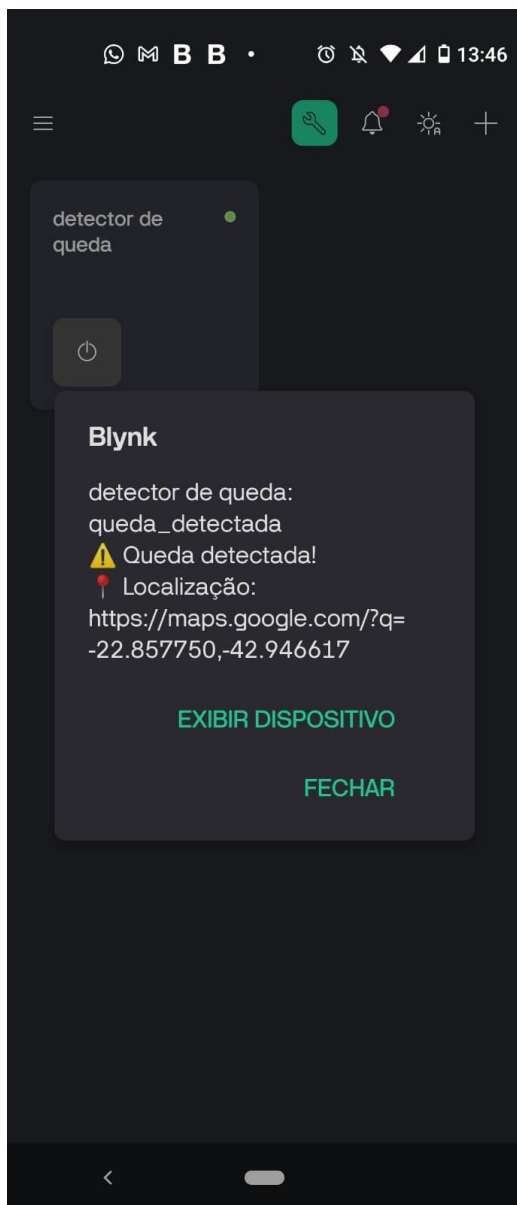


Figura 14: visão do usuário através da versão *mobile* do *Blynk*.

Caso a queda seja detectada, essa é a mensagem que aparece na tela do celular. Além disso, um alerta também será enviado para o *e-mail*, conforme a figura 15.

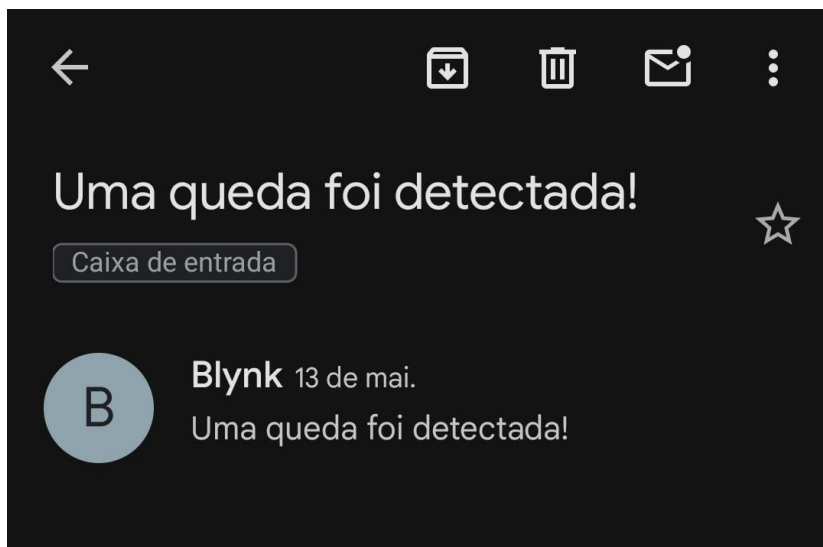


Figura 15: Visão da notificação através do aplicativo *Gmail*.

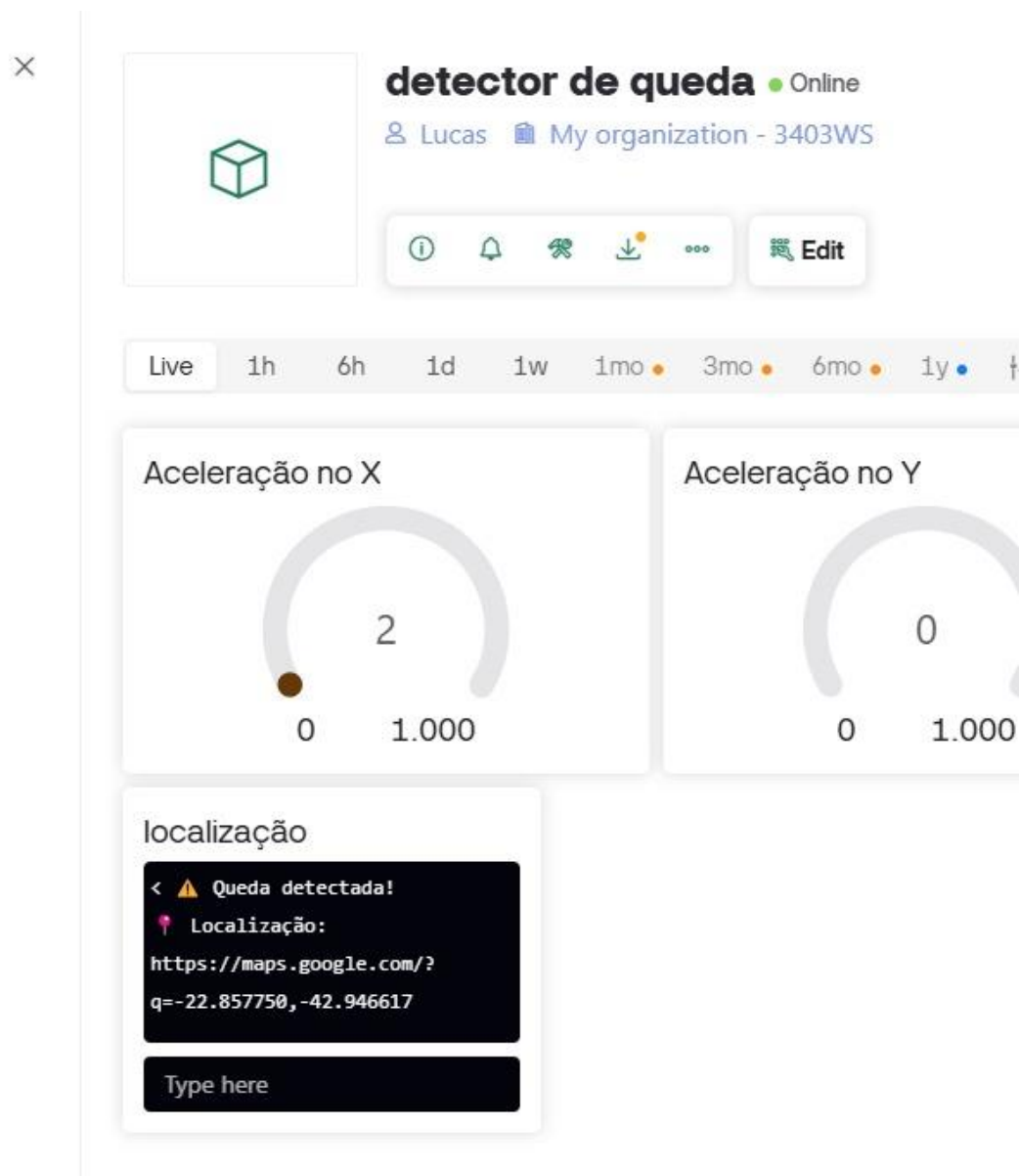


Figura 16: Visão do *Blynk web* quando uma queda é detectada.

Na figura 16, mostra que também é possível ver a localização da queda, caso a versão *web* do *blynk* esteja aberta.

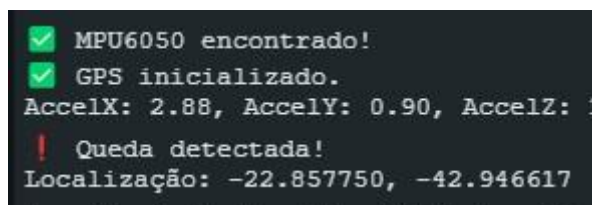


Figura 17: Visão do monitor Serial do Arduino IDE quando uma queda é detectada.

Como mostrado na figura 17, a localização aparece na saída do código em formato de latitude e longitude, no Arduino IDE.

3. ENCERRAMENTO DO PROJETO

3.1. Relato Coletivo:

De modo geral, os resultados obtidos foram considerados satisfatórios, tendo em vista que o projeto ainda se encontra na fase de prototipagem e apresenta natureza simplificada. É notável que com tempo e desenvolvimento adicional, há potencial para a realização de melhorias significativas, capazes de ampliar a usabilidade e a eficiência do dispositivo. Entre as possíveis evoluções, destaca-se a implementação do envio de mensagens por meio do aplicativo *WhatsApp*, bem como a adaptação ergonômica do dispositivo, que poderá ser acoplado ao corpo do usuário por meio de um cinto ou pulseira.

Como protótipo, o sistema demonstra potencial para ser transformado em um produto relevante para a sociedade, apresentando viabilidade tanto no âmbito social quanto econômico.

3.1.1. Avaliação de reação da parte interessada

O senhor William achou a ideia boa e funcional, embora com algumas faltas. Apesar de receber notificações no celular e por *E-mail*, ele preferiu um método mais emergencial, assim como no Anexo 3.

<https://youtube.com/shorts/FnvxrXTsjS0?feature=share>

Além de ser mencionado que o aplicativo deveria funcionar em segundo plano a todo momento, a resistência do produto não lhe foi a seu agrado ou sua aparência, pois a tia dele não gosta de acessórios e deixaria o produto de lado se lhe fosse entregue. Foi recomendado um produto mais compacto como um chip, para prender bem. Apesar de ele fazer exatamente o que deveria, o produto ainda não está a par com a maioria das expectativas do stakeholder.

3.2. Relato de Experiência Individual (Pontuação específica para o relato individual)

3.2.1. CONTEXTUALIZAÇÃO

Nínive - Eu fui a comunicação em tempo real com o stakeholder. Colaborei na construção no Arduino e nos testes. Foi interessante como as coisas quebravam, e como consertávamos o mais rápido possível, sempre visando a maior eficiência.

Lucas - Fui responsável pela construção, testes dos sensores e configuração do Arduino, fiz a escrita do código e participei ativamente em todas as áreas do projeto, com exceção da comunicação com o stakeholder, que ficou à cargo de Nínive.

Gabriel - Participei ativamente dos testes usando o meu laptop pessoal para testar e auxiliar na programação; e de diversas etapas da montagem do nosso projeto final. Considero que tanto a proposta quanto os resultados obtidos demonstraram um alto nível de qualidade e inovação.

Paulo Vitor – Fui quem escolheu e comprou as peças para a construção do projeto e ajudei a soldar algumas peças para a construção do projeto. Gostei muito de ter participado do projeto, pois foi algo inovador e que realmente pode ajudar às pessoas.

3.2.2. METODOLOGIA

Nínive - O projeto foi feito na faculdade e em várias casas. Foram recebidas opiniões de mães, tias e avós para melhor entendimento da dificuldade de senhores de idade. Este projeto durou em um total de 4 meses. Os primeiros foram para organização e planejamento, além de comprar os itens necessários. Os últimos dois meses era para refinar e reparar.

Lucas - A respeito do local, foram realizadas 10 reuniões, de forma síncrona, sendo 9 delas ocorridas na universidade Estácio de Sá, e 1 em minha residência. Tais reuniões serviram para desenvolvermos o projeto e avaliação do professor docente. 2 reuniões ocorreram de forma assíncrona para comunicação com *stakeholder*. A primeira reunião aconteceu no

início do projeto com propósito de obtermos os requisitos, cenário e contexto em que o senhor William passava para podermos desenvolver algo em cima, e a segunda reunião aconteceu no final do projeto, a fim de recebermos a opinião geral do produto. O trabalho teve início no mês de março, e fim no mês de junho.

Gabriel - O projeto foi majoritariamente feito em nossa faculdade, normalmente com a elaboração de ideias e desenvolvimento tanto do código quanto sua parte física. O código e os testes eram feitos em meu notebook, assim como a correção de alguns erros no próprio. A comunicação com o stakeholder estava inteiramente sobre a responsabilidade da integrante Nínive por sua maior proximidade com o mesmo.

Paulo Vitor – O projeto foi feito em maioria na faculdade, uma foi feita na casa do Lucas e as reuniões com o stakeholder foi feita a parte com a Nínive, e uma ou duas foi feito pelo aplicativo discord, onde cada um estava em sua casa. As reuniões no discord foram feitas para elaborarmos o projeto, já feito na faculdade foi durante as aulas e eram para as escolhas das peças, elaboração do roteiro de extensão e do projeto, já a reunião feita na casa do Lucas, ele com a Nínive fez o vídeo de testagem do projeto, pois a Nínive mora perto da residência do Lucas.

3.2.3. RESULTADOS E DISCUSSÃO:

Nínive - Sabia que teria desafios, especialmente pela falta de experiência com hardware e programação. Mesmo com o estresse causado pelas falhas do produto, ter o apoio de Lucas (que possui mais experiência prática) trouxe uma sensação de segurança. Por fim, ficou claro o quanto ainda tenho a aprender.

Lucas - Esperava que seria um projeto relativamente fácil de fazer, pois já tinha experiência anterior em montar peças através da disciplina que realizei anteriormente, “programação de microcontroladores”. O vivido foi diferente da expectativa, pois o projeto foi demonstrando ser mais desafiador do que apresentado na teoria. Em algumas etapas do projeto, fiquei bastante frustrado em relação às peças que não funcionavam e a pressão de estar trabalhando contra o tempo; pensei em abandonar o projeto algumas vezes. Porém após a ajuda de todos os membros, cada um foi dando suas ideias, dicas, e tudo começou a fluir conforme o planejado e conseguir entregar o projeto a tempo.

Gabriel - Devido já ter tido experiência com trabalhos do gênero, não estive surpreso com as dificuldades e problemas que nosso grupo tinha com ele, mas conseguimos entregar um produto até que satisfatório com base nos ideais do grupo, além de ter sido agradável trabalhar com os integrantes e uma proposta relativamente desafiadora e intrigante.

Paulo Vitor – No começo da matéria estava desanimado, pois era algo que eu nunca tinha pesquisado a fundo e até aquele momento eu não queria trabalhar, e foi bastante difícil desde a escolha do projeto até como fazer esse projeto funcionar, pois esse projeto não era algo genérico e sim algo que faça diferença para a sociedade. Ao fim do projeto fiquei bastante interessado em trabalhar na área de IOT, pois pelo que eu pesquisei é algo que vai continuar crescendo no futuro, posso ter uma boa renda trabalhando nessa área e posso ajudar a criar projetos que faça a diferença na sociedade.

3.2.4. REFLEXÃO APROFUNDADA

Nínive - Tinha expectativas de que tudo ia acontecer com facilidade, que a jornada seria tranquila até o fim. Porém, a realidade foi diferente. O Arduino não se conectava a IDE, ao sensor e nem ao GPS. Foram necessárias várias tentativas até descobrirmos os problemas: cabos mal encaixados, um laptop com mal contato e um GPS que só funciona ao ar livre. Mas nossos objetivos foram cumpridos em tempo.

Lucas - Quando começamos o projeto, acreditei ser muito mais fácil implementar as funções e montar as peças. Porém conforme passava as reuniões, vi que seria um desafio a enfrentar. Passei por diversos problemas com conexão dos sensores na ESP-32, principalmente com a MPU-6050, onde acontecia mal contato toda vez em que eu mexia a peça de lugar. Alguns cabos vieram com defeito e a calibração da detecção de queda foi um desafio de implementar no *Blynk*, porém após diversas reuniões e muita pesquisa aprofundada, conseguimos finalizar e tudo esperado no projeto foi desenvolvido com sucesso.

Gabriel - Nestes tipos de projeto, é comum que a sua montagem seja desafiadora por diversos motivos. Por isso, quando os problemas começaram a aparecer, não me foi nenhuma surpresa — como, por exemplo, o mau contato de peças que nosso produto apresentava em sua montagem pré-solda. Porém, mesmo assim, conseguimos fazer a entrega do mesmo de maneira satisfatória.

Paulo Vitor - Achei que mesmo com a dificuldade do projeto, conseguiríamos terminar o projeto com a facilidade, pois tínhamos achados tutoriais que usava os dois sensores com a ESP-32 separadamente, mas não foi isso que aconteceu tivemos que gastar há mais do que prevíamos, soldamos também as peças que foi algo que não tínhamos previsto, também houve problemas nas conexões via cabos, a protoboard também apresentou problema, além do grande problema do nosso projeto que foi o sensor do GPS, mas apesar disso tudo conseguimos obter o resultado esperado para entregarmos o projeto.

3.2.5. CONSIDERAÇÕES FINAIS

Níve - O stakeholder – meu pai – teve várias dúvidas: E se o idoso cair em cima do sensor? E se ele só largar o produto em algum canto e esquecer? E se ele cair gradualmente, se segurando até se sentar no chão? Nosso projeto é frágil, então esperamos que no futuro – caso o projeto seja continuado – que o produto seja mais resistente, com mais recursos, como alertas sonoros para facilitar sua localização além das coordenadas e que tenha mais calibrações, para detectar quando a pessoa permanecer no chão.

Lucas - O projeto já funciona de forma aceitável no estado atual, porém muitas melhorias ainda podem ser aplicadas a ele. Durante os testes, quando uma queda era detectada, caso estivesse em um ambiente fechado, o módulo de GPS NEO-6M não captava sinal, pois a peça precisa coletar dados de satélites para se localizar e enviar as informações a quem usa, tornando esse fator um problema que restringe muito o uso da função de localização. Com isso, seria interessante fazermos futuras implementações no arduíno, como a instalação de uma antena potente junto ao módulo de GPS, a fim de encontrar sinal até em ambientes com muitas obstruções de transmissão.

Gabriel - Acredito que, embora o produto esteja funcional, ainda há aspectos que precisam ser aprimorados em nosso projeto — especialmente o design, que considero ser a parte mais crucial. É fundamental que ele seja mais compacto e resiliente, facilitando o manuseio e garantindo a integridade do mesmo, especialmente para usuários que já enfrentam limitações de mobilidade.

Paulo Vitor – Apesar do nosso projeto já está adequado para um protótipo, temos muitas melhorias para fazer. Uma das melhorias seria como poderíamos implementar para colocar na pessoa idosa, minha ideia seria colocar por meio de uma pulseira, então teríamos que provavelmente fazer uma placa própria ou usar uma placa menos que a ESP 32, outra ideia seria o meio como a mensagem pode ser enviada, no nosso projeto enviamos as mensagens via email e via Blynk, mas com certeza o ideal seria enviar mensagens via whatsapp, que é o aplicativo de mensagens mais usado pela população, outra coisa que poderíamos implementar é a pessoa que seria a responsável pelo idoso saber a localização do idoso em tempo real, então poderíamos implementar um módulo de chip e criar alguma solução via código, onde a pessoa teria a localização do idoso.

4. BIBLIOGRAFIA

MACIEL, Arlindo. "Quedas em idosos: Um problema de saúde." Rev. Med. Minas Gerais 20 (2010). Disponível em: <https://encurtador.com.br/oltvO>. Acesso em: 19 mar. 2025.

QUEIROZ, Laísa. "No Brasil, prevalência de quedas entre idosos em áreas urbanas é de 25%". gov.br, 2023. Disponível em: <https://www.gov.br/saude/pt-br/assuntos/noticias/2023/junho/no-brasil-prevalencia-de-quedas-entre-idosos-em-areas-urbanas-e-de-25>. Acesso em: 26 mar. 2025.

COM 12,2 MIL ACIDENTES SOMENTE NESTE ANO, SAÚDE ALERTA PARA O RISCO DE QUEDAS DE IDOSOS. Governo do Estado Paraná, 2024. Disponível em: <https://www.saude.pr.gov.br/Noticia/Com-122-mil-acidentes-somente-neste-ano-Saude-alerta-para-o-risco-de-quedas-de-idosos>. Acesso em: 26 mar. 2025.

FALLS. World Health Organization, 2021. Disponível em: <https://www.who.int/news-room/fact-sheets/detail/falls>. Acesso em: 07 mai. 2025.

LOJA DA DRYKA. Pulseira inteligente para idosos – Help Life. Disponível em: <https://www.lojadadryka.com.br/products/pulseira-inteligente-para-idosos-help-life>. Acesso em: 1 jun. 2025.

Monk, Simon. Programação com Arduino: Começando com Sketches. 2 Ed. Porto Alegre: Bookman, 2017.

Oliveira, Cláudio Luís Vieira; Zanetti, Humberto Augusto Piovesana. Arduino Descomplicado Como Elaborar Projetos de Eletrônica. 1 Ed. São Paulo: Érica, 2015.

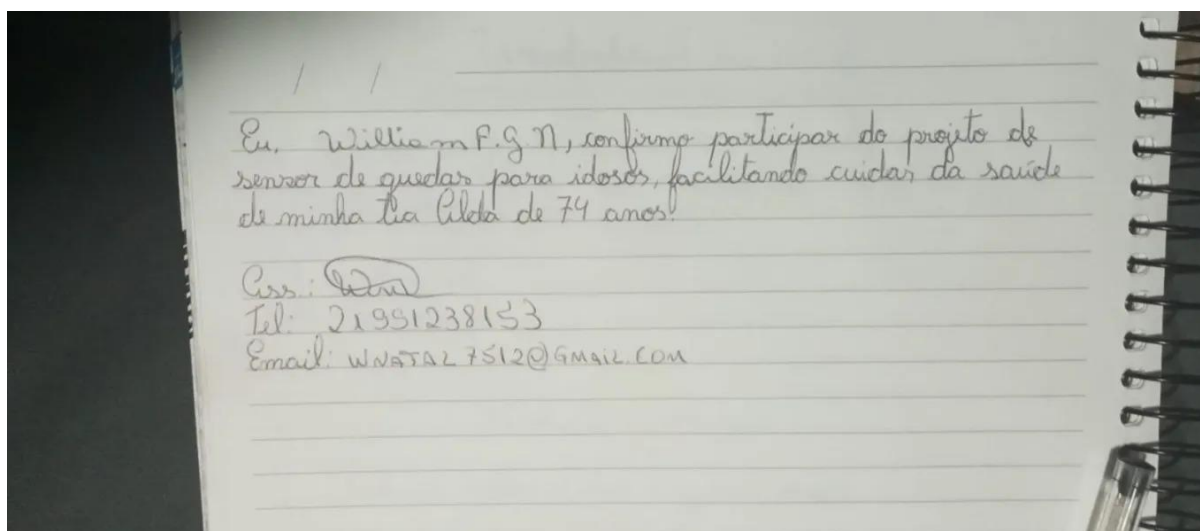
AZEVEDO, Aline Lemos de; BESSA, Marcelo Eustáquio. Tecnologias Assistivas e Internet das Coisas aplicadas à saúde do idoso. Revista Brasileira de Geriatria e Gerontologia, Rio de Janeiro, v. 23, n. 6, p. 1-10, 2020.

MORAES, Eny Fortes; APRILE, Maria Regina; MENDES, Márcia de Souza. Tecnologia e envelhecimento: propostas para o cuidado e segurança de idosos. Revista Kairós, São Paulo, v. 13, n. 2, p. 55-74, 2010.

PERRACINI, Monica R.; RAMOS, Luiz Roberto. Fatores associados a quedas em uma coorte de idosos residentes na comunidade. Revista de Saúde Pública, São Paulo, v. 36, n. 6, p. 709-716, 2002.

ALMEIDA, Taysa Rafaela Silva Houly; HENRIQUE, Maria Dantas; MOURA, Maria Eduarda Lima de; et all. Hérnia de disco lombar: riscos e prevenção. Rev. Ciênc. Saúde Nova Esperança, 2014. Disponível em: <https://www.facene.com.br/wp-content/uploads/2010/11/H--rnia-de-disco-lombar-PRONTO.pdf>. Acesso em: 4 de jun. 2025.

5. ANEXOS



Anexo 1: Contrato de cooperação.

| 1 | Data | Horário 24H | Objetivo |
|----|------------|---------------|--|
| 2 | 12/3/2025 | 15:45 - 17:30 | Criação do escopo do projeto e a parte interessada |
| 3 | 17/03/2025 | 18:00 - 20:40 | Desenvolvimento do roteiro de extensão |
| 4 | 26/03/2025 | 07:50 - 10:00 | Reparo do roteiro e compra de peças para o projeto |
| 5 | 2/4/2025 | 07:50 - 09:30 | Início da implementação do código com o Arduino |
| 6 | 9/4/2025 | 07:10 - 10:00 | Tentativa de montar o Arduino; protoboard incompatível |
| 7 | 30/04/2025 | 07:50 - 10:40 | Compra de mais peças para a construção do Arduino |
| 8 | 7/5/2025 | 07:20 - 10:50 | Tentativa de montar o Arduino; conexão falha |
| 9 | 8/5/2025 | 15:45 - 21:02 | Construção finalizada do Arduino com sensor e conexão no Blynk |
| 10 | 21/05/2025 | 07:00 - 10:00 | Soldagem das peças e regulagem do GPS |

Anexo 2: Plano de trabalho e objetivos.

6. REFERÊNCIAS

DINIZ, Lucas; FERNANDES, Nírive. *Projeto IoT - Sensor de queda para idosos*. [s.l.]; [s.n.], 2025. 1 vídeo (50 seg). Produção própria. Disponível em: <https://www.youtube.com/watch?v=l6KmqluCT1U>. Acesso em: 3 jun. 2025.

ROIFF, Gabriel; FERNANDES, Nírive. *Reação Stakeholder IoT*. [s.l.]; [s.n.], 2025. 1 vídeo (17 seg). Produção própria. Disponível em: <https://youtube.com/shorts/FnvxrXTsjS0?feature=share>. Acesso em: 3 jun. 2025.