

CENTRO UNIVERSITÁRIO DO INSTITUTO MAUÁ DE TECNOLOGIA

Escola de Engenharia Mauá

Engenharia Elétrica

BRUNO CALDEIRA FERNANDES

BRUNO ROSCHEL CARRER

LUCAS ROMANO AMARANTE

RENATO CESAR DE SOUZA FILHO

Baby IoT – monitoramento do bem-estar de recém-nascidos

São Caetano do Sul

2019

BRUNO CALDEIRA FERNANDES

BRUNO ROSCHEL CARRER

LUCAS ROMANO AMARANTE

RENATO CESAR DE SOUZA FILHO

***Baby IoT*– monitoramento do bem-estar de recém-nascidos**

Trabalho de Conclusão de Curso apresentado à Escola de Engenharia Mauá do Centro Universitário do Instituto Mauá de Tecnologia como requisito parcial para a obtenção do título de Engenheiro Elétrico.

Orientador: Fernando de Almeida Martins

Área de concentração: Engenharia Elétrica

São Caetano do Sul

2019

Baby IoT: monitoramento do bem-estar de recém-nascidos. / Bruno Caldeira Fernandes [et al.] — São Caetano do Sul: CEUN-IMT, 2019.
82 p.

Trabalho de Conclusão de Curso – Escola de Engenharia Mauá do Centro Universitário do Instituto Mauá de Tecnologia, São Caetano do Sul, SP, 2019.

Orientador: Fernando de Almeida Martins

1. Bebê. 2. Bem-estar. 3. Internet das Coisas. 4. Saúde. 5. LoRaWAN. I. Fernandes, Bruno Caldeira. II. Carrer, Bruno Roschel. III. Amarante, Lucas Romano. IV. Souza, Renato Cesar V. Instituto Mauá de Tecnologia. Escola Engenharia. VI. Título.

BRUNO CALDEIRA FERNANDES

BRUNO ROSCHEL CARRER

LUCAS ROMANO AMARANTE

RENATO CESAR DE SOUZA FILHO

Baby IoT – monitoramento do bem-estar de recém-nascidos

Trabalho de Conclusão de Curso aprovado pela Escola de Engenharia Mauá do Centro Universitário do Instituto Mauá de Tecnologia como requisito parcial para a obtenção do título de Engenheiro Elétrico.

Banca avaliadora:

Prof. Me. Fernando De Almeida Martins.
Orientador

Prof. Dr. Edval Delbone
Avaliador

Prof. Dr. Wanderson Assis
Avaliador

São Caetano do Sul, 09 de dezembro de 2019.

Dedicamos este trabalho a Deus. Sem ele nada seria possível.

AGRADECIMENTOS

Agradecemos primeiramente nossos familiares pela dedicação durante os anos de graduação.

Ao professor Me. Fernando Martins pela orientação e dedicação na realização do trabalho.

À professora Dr^a Agda Carvalho pelas informações de *design* e *marketing* do projeto.

À equipe do setor do FabLab pela disponibilidade em aplicar o desenvolvimento do protótipo do trabalho.

Ao Instituto Mauá de Tecnologia pela infraestrutura e contribuição no aprendizado e possibilitar o desenvolvimento do Trabalho de Conclusão de Curso.

Procure ser um homem de valor, em vez de ser um homem de sucesso.

Albert Einstein

RESUMO

O trabalho tem como objetivo melhorar a qualidade do bem-estar de recém-nascidos. Utilizando o conceito de Internet das Coisas em conjunto com a tecnologia LoRaWAN, o projeto propõe a introdução de um dispositivo eletrônico na vestimenta do bebê e em seu quarto, captando dados e tratando-os. O monitoramento é apresentado por meio de um aplicativo, disponibilizando informações para os pais, pediatras e profissionais da área. A instalação pode ser feita tanto em berçários de hospitais como nas próprias residências, visando a integração da tecnologia na área da saúde.

Palavras-chave: Bebê. Bem-estar. Internet das Coisas. Saúde. LoRaWAN.

ABSTRACT

The work aims to improve the quality of well-being of newborns. Using the IoT concept in conjunction with LoRaWAN technology, the project proposes the introduction of an electronic device into the baby's clothes and bedroom, obtaining data and treating them. Monitoring is presented through an app, providing information for parents and pediatricians of the newborn. The installation can be done in hospital nurseries and their own homes, aiming at the integration of technology in health care.

Keywords: Baby. Well-being. Internet of Things. Health. LoRaWAN.

LISTA DE FIGURAS

Figura 1 - Tripé da Inovação	27
Figura 2 - Taxa de mortalidade de acordo com a idade	31
Figura 3 – Esquemático do projeto.	36
Figura 4 - Sensor MAX30100	37
Figura 5 - Gráfico da taxa de absorção da luz pelo espectro	38
Figura 6 - Diagrama do fluxo de dados do MAX30100 até o RN2903.....	40
Figura 7 - Esquema Elétrico do módulo MAX30100.....	42
Figura 8 - Sensor MAX30100 conectado ao Arduino UNO.....	43
Figura 9 - Pinagem do Sensor MAX30100	44
Figura 10 - Representação da montagem do Sensor MAX 30100 conectado ao Arduino.....	44
Figura 11 - Esquemático eletrônico utilizando o MAX30100, PIC24FJ128GA306 e RN2903A.	45
Figura 12 - Pinagem MPU6050	47
Figura 13 - Esquema de Montagem do Sensor MPU6050 no Arduino UNO.....	47
Figura 14 - Sensor MPU6050 ligado ao Arduino UNO.....	48
Figura 15 - Esquema Elétrico do MPU6050	48
Figura 16 – Arquitetura do sistema	49
Figura 17 - Exemplo de medição salva no banco de dados.....	51
Figura 18 - Rotas da API.....	52
Figura 19 - <i>Dashboard Baby IoT</i>	54
Figura 20 - Tela de <i>Login</i>	55
Figura 21 - Tela Inicial	56
Figura 22 - Menu do aplicativo.....	56
Figura 23 - Tela de alarmes.....	57

Figura 24 - Tela diagnósticos.....	58
Figura 25 - Tela dos contatos dos pediatras	59
Figura 26 - Localização de hospitais próximos.....	59
Figura 27 - Aplicativos de rotas vinculados com o <i>Baby IoT</i>	60
Figura 28 - Publicação do aplicativo	60
Figura 29 - Desenho realizado no programa SolidWorks®	61
Figura 30 - Processo de pintura	62
Figura 31 - Botões de fixação.....	63
Figura 32 – Peças 3D finalizadas	64
Figura 33 - Caixa para moldar o boneco	64
Figura 34 - Molde de silicone	65
Figura 35 – Peça desenvolvida com silicone	66
Figura 36 - Roupas adaptadas para o protótipo	66
Figura 37 - Molde para furação.....	67
Figura 38 - Equipamento de monitoramento	69

LISTA DE TABELAS

Tabela 1 - Porcentagem das causas de óbitos evitáveis - Fonte SIM	31
Tabela 2 - Características específicas da idade dos Movimentos Gerais normais	33
Tabela 3 - Frequência cardíaca website VeryWellFamily	34
Tabela 4 - Frequência cardíaca Artigo ComplexChild	34
Tabela 5 - Oximetria durante o descanso Artigo ComplexChild	35
Tabela 6 - Pinagem do Sensor MAX30100	41

LISTA DE ABREVIATURAS E SIGLAS

ABNT – Associação Brasileira de Normas Técnicas

API – *Application Programming Interface*

bpm – Batimentos por minuto

EC2 – *Amazon Elastic Compute Cloud*

I2C - *Inter-Integrated Circuit*

IoT – *Internet of Things*

LED – *Light Emitting Diode*

LoRa – *Low Power Long Range*

LoRaWAN – *Low Power Long Range Wide Area Network*

MQTT – *Message Queuing Telemetry Transport*

SCL – *Serial Clock Line*

SDA – *Serial Data Line*

SMS – *Short Message Service*

SUMÁRIO

1	INTRODUÇÃO.....	27
1.1	OBJETIVOS	28
1.2	METODOLOGIA DO TRABALHO	28
1.3	PROPÓSITO DA PESQUISA	28
1.4	INSTRUMENTO DE ANÁLISE DE DADOS	28
1.5	PROCEDIMENTOS	29
2	AMBIENTE DE ESTUDO	30
2.1	DESCRIÇÃO DA UTILIZAÇÃO DE SENSORES	32
2.1.1	ESTUDO DE SENSORES DO AMBIENTE	32
2.1.2	ESTUDO DE SENSORES PARA O BEBÊ.....	32
3	METODOLOGIA	36
3.1	DESENVOLVIMENTO DO HARDWARE.....	36
3.1.1	SENSOR DE BATIMENTO CARDÍACO E OXIMETRIA	36
3.1.2	SUBSISTEMAS.....	37
3.1.3	CAPTAÇÃO DE DADOS	38
3.1.4	COMUNICAÇÃO I2C.....	39
3.1.5	LEITURA DE DADOS E COMUNICAÇÃO COM MICROCONTROLADOR PIC	40
3.1.6	MAX30100: ESQUEMA ELÉTRICO E PINAGEM	40
3.1.7	SMART CAMPUS MAUÁ.....	42
3.1.8	PROGRAMAÇÃO EM ARDUINO	42
3.1.9	SENSOR ACCELERÔMETRO	46
3.1.9.1	Funcionamento	46
3.1.9.2	Programação em Arduino	46
3.2	SOFTWARE	49
3.2.1	ARQUITETURA DO SISTEMA	49
3.2.2	BANCO DE DADOS	50
3.2.3	API	51
3.2.4	SERVIÇOS	52
3.2.4.1	<i>Upload</i>	52
3.2.4.2	<i>E-mail e SMS</i>	52

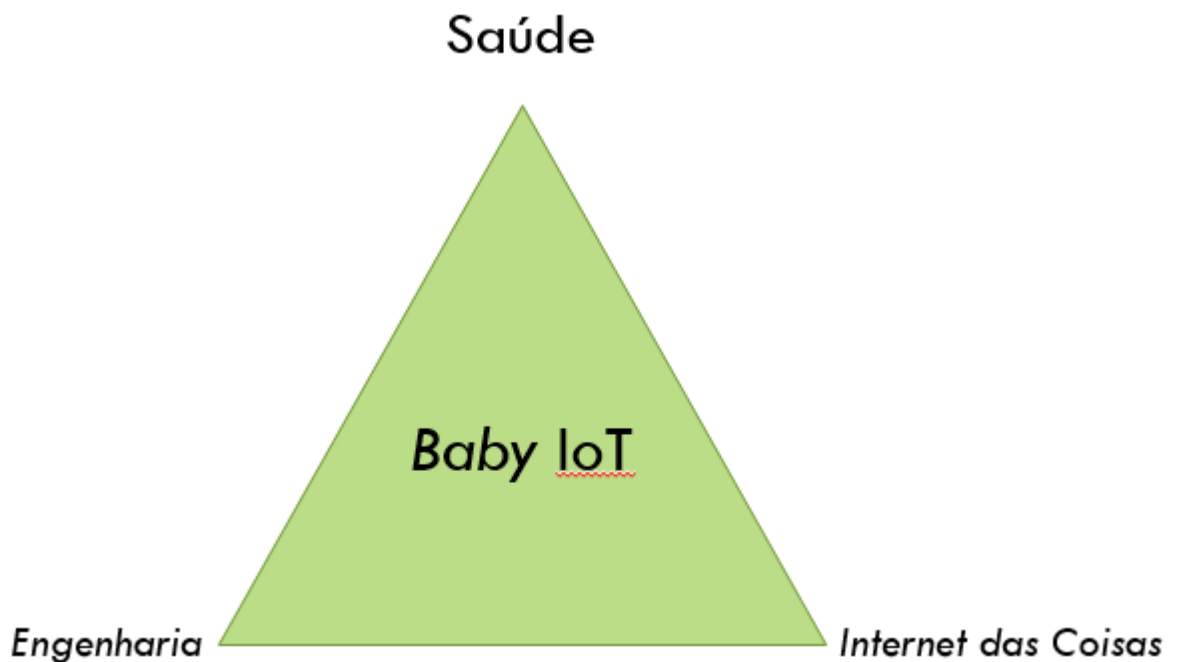
3.2.4.3	Tecnologia Utilizada	53
3.2.5	APLICATIVO MOBILE	53
3.2.5.1	<i>Dashboard</i>	54
3.2.5.2	Funcionalidades	55
4	RESULTADOS DO PROJETO	61
4.1	MONTAGEM DE APLICAÇÃO	61
4.1.1	APLICAÇÃO DO PRODUTO	61
4.2	APLICAÇÕES ASSOCIADAS AO PROJETO	67
4.2.1	PEDIATRAS	68
4.2.2	PAIS	68
4.2.3	HOSPITAIS	68
4.3	CUSTOS E MERCADOS.....	69
5	CONCLUSÃO	71
5.1	CONSIDERAÇÕES FINAIS	71
5.2	TRABALHOS FUTUROS	72
	REFERÊNCIAS	73
	ANEXO A – CÓDIGO DO MAX30100 EM ARDUINO	75
	ANEXO B – CÓDIGO DO MPU6050 EM ARDUINO	76
	ANEXO C – ESQUEMA ELÉTRICO PIC	78
	ANEXO D – ESQUEMA ELÉTRICO LORA	79
	ANEXO E – PROGRAMAÇÃO DO APLICATIVO	80
	ANEXO F – FLUXOGRAMA DO APLICATIVO	82

1 INTRODUÇÃO

Este trabalho aborda um caso particular de desenvolvimento de um dispositivo eletrônico para a captação de informações relacionadas ao bem-estar de recém-nascidos. O dispositivo objeto deste projeto foi microprocessado e teve a capacidade de transmissão de informações que foram obtidas do recém-nascido como temperatura, oximetria e aceleração.

As informações foram disponibilizadas aos usuários por meio de aplicativo para acompanhamento remoto. Houve uma integração da Internet das Coisas e engenharia aplicadas na área da saúde como um tripé, representado pela Figura 1:

Figura 1 - Tripé da Inovação



Fonte: Os Autores, 2019.

Esse conceito visou oferecer um maior conforto aos pais do recém-nascido e uma maior qualidade de informações que podem ser utilizadas pelo médico pediatra para uma melhor avaliação do bem-estar do bebê.

1.1 OBJETIVOS

O trabalho teve por objetivo o desenvolvimento de um sistema para o monitoramento do bem-estar de recém-nascidos. Utilizou dispositivos de baixo consumo e integrados ao conceito de Internet das Coisas, contando com um leitor de batimento cardíaco, um sensor de taxa de oxigênio, um acelerômetro, além do sensoriamento do ambiente em que o recém-nascido se encontra, que monitora a temperatura ambiente, a umidade e a luminosidade do ambiente.

1.2 METODOLOGIA DO TRABALHO

O sistema de sensoriamento utilizou um servidor de rede, uma antena LoRaWAN conectada à Internet e um dispositivo sensor com um módulo de rádio frequência (RF) LoRaWAN, possibilitando a transmissão de dados à um aplicativo móvel e fornecendo aos responsáveis pelo bebê todas as informações possíveis.

1.3 PROPÓSITO DA PESQUISA

A ideia foi criar um sistema de monitoramento em uma roupa usual, uma vez que o uso da vestimenta nesses indivíduos é uma prática comum.

A aplicação dos sensores na vestimenta do bebê e no ambiente em que ele se encontra tem como objetivo realizar medições constantes como do batimento cardíaco e taxa de oxigenação, visando uma posterior análise dos dados para a percepção de anomalias. Por meio de um estudo adequado desses dados pode possibilitar medir a qualidade de sono, prevenir doenças e alertar perigos imediatos.

1.4 INSTRUMENTO DE ANÁLISE DE DADOS

Foi utilizado um sensor de frequência cardíaca e oximetria de pulso. Por meio deste sensor foram emitidos dois comprimentos de ondas através de dois LEDs, medindo a absorção do sangue por meio de um sensor fotodetector de luz. O sensor tem baixo consumo energético

e com precisão. Serão utilizados no ambiente sensores de temperatura, umidade e luminosidade ambiente, sendo fundamental para a aplicação proposta.

1.5 PROCEDIMENTOS

Os sensores foram implementados no macacão do bebê, com o posicionamento ajustado ao melhor ponto de captação de dados, melhor conforto ao recém-nascido e menor risco de impacto ao dispositivo. Também foi estudado o material necessário para revestir esses componentes, para minimizar os possíveis impactos que estes podem receber. O monitoramento é realizado através de um dispositivo equipado com o rádio LoRaWAN.

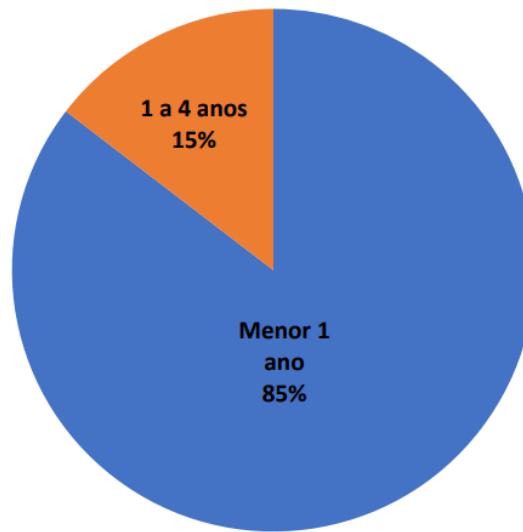
2 AMBIENTE DE ESTUDO

Com a evolução da tecnologia e o aprimoramento da comunicação entre os dispositivos, o termo IoT está mais presente no cotidiano sendo aplicado, por exemplo, em sistemas que visam o benefício à sociedade. É interessante abordar o tema Internet das Coisas na medicina e na área médica proporcionando a possibilidade de monitoramento em tempo real de sinais vitais com o registro dos dados coletados e ainda com a comodidade de realizar essas medições sem a necessidade da presença de um profissional da saúde.

Nesse caso específico, o projeto proporcionou a criação de um dispositivo no qual foram aplicados conceitos de *wearables*, ou seja, tecnologias vestíveis destinadas para o uso em recém-nascidos.

Buscou-se atuar no bem-estar e contribuir para a diminuição da mortalidade de bebês em casos evitáveis. Segundo o Portal de Arquivos da Saúde (NOGALES, 2018), em 2016, dentre 17432 óbitos de crianças entre 28 dias a 4 anos, 8564 seriam casos evitáveis mesmo havendo alguns fatores relacionados ao neonatal não considerados. As causas evitáveis em porcentagem foram relatadas na Tabela 1 (SIM, 2018). A Figura 2 (SIM,2018) apresenta dados percentuais referentes à taxa de mortalidade na infância.

Figura 2 - Taxa de mortalidade de acordo com a idade
Mortalidade na Infância por idade



Fonte: SIM, 2018.

Tabela 1 - Porcentagem das causas de óbitos evitáveis - Fonte SIM

Causas evitáveis por agrupamento	2010	2011	2012	2013	2014	2015	2016	2017
Reduzível pelas ações de imunização	0,4	0,6	1	1	1,6	0,6	0,3	0,5
Reduzíveis atenção à mulher na gestação	6,9	6,3	7	7,5	7,8	8,2	7,7	8
Reduzíveis por adequada atenção à mulher no parto	2,7	2,7	2,3	2,4	2,6	2,4	2,7	2,6
Reduzíveis adequada atenção ao recém-nascido	8,3	9,1	8,7	8,2	8,8	9,6	9,3	10,9
Reduzíveis por ações no diagnóstico e tratamento adequado	41,5	43,7	43	42,4	41,7	40,7	40,9	40
Reduzíveis com ações promoção à saúde	40,3	37,6	38	38,5	37,6	38,4	39,1	38,1

Fonte: SIM, 2018.

Conforme os dados acima mostrados, foi possível verificar que a taxa de mortalidade esteve concentrada em crianças menores que um ano, mostrando que com um monitoramento

adequado, principalmente de recém-nascidos, seria possível reduzir a quantidade de óbitos evitáveis.

2.1 DESCRIÇÃO DA UTILIZAÇÃO DE SENSORES

A seguir foram apresentados textos sobre o tema do trabalho, incluindo estudo sobre o melhor ambiente do bebê e suas características biológicas como características do bem-estar relacionado aos sensores mencionados durante o objetivo.

2.1.1 ESTUDO DE SENSORES DO AMBIENTE

As principais medições foram referentes à temperatura e a umidade do local. Condições que poderiam ocasionar riscos à saúde do bebê. Visando também a importância do controle da quantidade de luz presente no ambiente no qual o bebê dorme, sendo necessário definir a quantidade de luz para cada momento de sono do bebê ao longo do dia, evitando o risco de o bebê ter um descanso prejudicado caso a luminosidade não seja adequada.

2.1.2 ESTUDO DE SENSORES PARA O BEBÊ

Conforme o artigo, sobre o sistema de monitoramento de movimentos de recém-nascidos (CHEN et al., 2016), foi possível determinar os problemas que um sensor capaz de identificar a aceleração pode identificar em um recém-nascido, pois indica o desenvolvimento neuromotor cerebral, avaliando possíveis riscos de deficiências neurológicas durante o seu desenvolvimento. Os problemas que podem ser observados foram convulsões com movimentos anormais de membros e sobre a possibilidade de ocorrerem distúrbios durante uma avaliação do funcionamento motor, podendo também identificar um sono seguro e riscos de quedas dos bebês.

Ainda, no mesmo artigo, foi apresentada a classificação de Movimentos Gerais, de acordo com a idade, como mostrado na Tabela 2:

Tabela 2 - Características específicas da idade dos Movimentos Gerais normais

Tipo Movimentos Gerais	Idade dos Bebês em semanas	Descrição
Movimentos prematuros	De \pm 28 semanas a 36 – 38 semanas	Grande variação ao longo do tempo, mais proximal do que nos dias anteriores e caracterizada de amplitude baixa a moderada e de velocidade lenta a moderada
Movimentos contorcidos	De 36 a 38 semanas a 46 – 52 semanas	Parece ser um pouco mais lento e mostrar menos participação da pélvis e do tronco
Movimentos inquietos	De 46 a 52 semanas a 54 – 58 semanas	Consiste em um fluxo contínuo de movimentos pequenos e elegantes. Ocorre de forma irregular em todo o corpo, cabeça, tronco e membros participam de forma semelhante.

Fonte: NCBI, 2016.

Com os dados acima, foi gerado um padrão de diagnóstico para a avaliação do sistema motor do recém-nascido.

Um exemplo de aplicação de acelerômetro foi apresentado no artigo *Baby by the Numbers* (LEE, 2015). Nesse artigo, o acelerômetro foi instalado diretamente no tornozelo, monitorando a sua posição e movimentação, enviando alertas aos pais caso seja identificada uma queda ou um rolamento do bebê.

Nos estudos do sensor de frequência cardíaca foram apresentadas doenças que poderiam ser identificadas caso essa frequência estivesse elevada. De acordo com o website *Verywellfamily*, a elevação da frequência cardíaca pode estar associada à dor, febre ou desidratação. Também foi possível classificar o tipo de pulsação, sendo rápida, taquicardia, ou lenta, bradicardia.

As tabelas 3 e 4, apresentadas no website *Verywellfamily* (IANNELLI, 2019) e no artigo *ComplexChild* (AGRAWAL, 2009) indicaram a faixa correta da frequência cardíaca de acordo com a idade, e se está em repouso ou não. Esses dados foram utilizados para comparar e definir a faixa de frequência cardíaca normal de um recém-nascido.

Tabela 3 - Frequência cardíaca website VeryWellFamily

Frequência cardíaca WebSite Idade	Frequência cardíaca em repouso	Intervalo normal
0 a 3 meses	143 batimentos / minuto	107 a 181 batimentos / minuto
3 a 6 meses	140 batimentos / minuto	104 a 175 batimentos / minuto
6 a 9 meses	134 batimentos / minuto	98 a 168 batimentos / minuto
9 a 12 meses	128 batimentos / minuto	93 a 161 batimentos / minuto
12 a 18 meses	116 batimentos / minuto	88 a 156 batimentos / minuto
18 meses a 24 meses	116 batimentos / minuto	82 a 149 batimentos / minuto
2 a 3 anos	110 batimentos / minuto	76 a 142 batimentos / minuto

Fonte: VeryWellFamily, 2019.

Tabela 4 - Frequência cardíaca Artigo ComplexChild

Idade	Intervalo Normal (Descansando) em batimentos por minuto
Prematuro	120 – 170
0 – 3 meses	100 – 150
3 – 6 meses	90 – 120
6 – 12 meses	80 – 120
1 – 3 anos	70 – 110
3 – 6 anos	65 – 110
6 – 12 anos	60 – 95
Maior que 12 anos	55 – 85

Fonte: ComplexChild, 2019.

Por fim, o sensoriamento da oxigenação foi necessário para verificar se o bebê chorou e identificar possíveis problemas de respiração durante o sono. A Tabela 5 apresenta os valores típicos de oximetria para crianças durante o período de sono.

Tabela 5 - Oximetria durante o descanso Artigo ComplexChild

Idade	Intervalo Normal (Descansando) em respiração por minuto
Prematuro	40 – 70
0 – 3 meses	35 – 55
3 – 6 meses	30 – 45
6 – 12 meses	25 – 40
1 – 3 meses	20 – 30
3 – 6 meses	20 – 25
6 – 12 meses	14 – 22
Maior que 12 anos	12 – 18

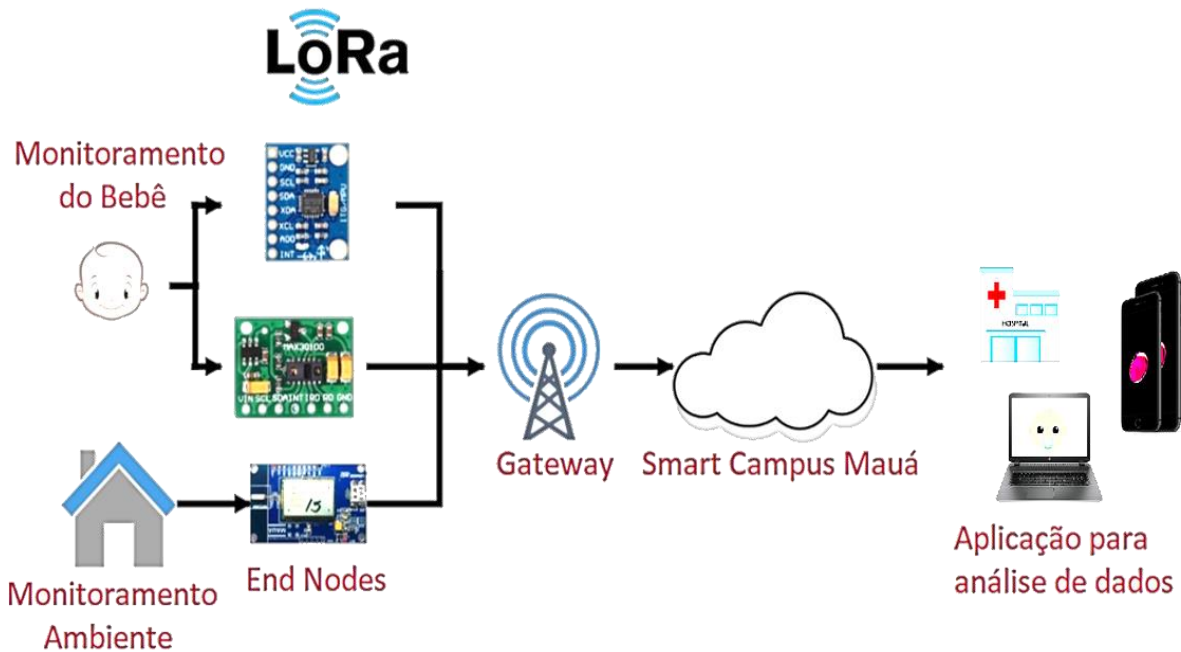
Fonte: ComplexChild, 2019.

3 METODOLOGIA

3.1 DESENVOLVIMENTO DO HARDWARE

Nesta seção será descrito o projeto do hardware utilizado no projeto.

Figura 3 – Esquemático do projeto.



Fonte: Os Autores, 2019.

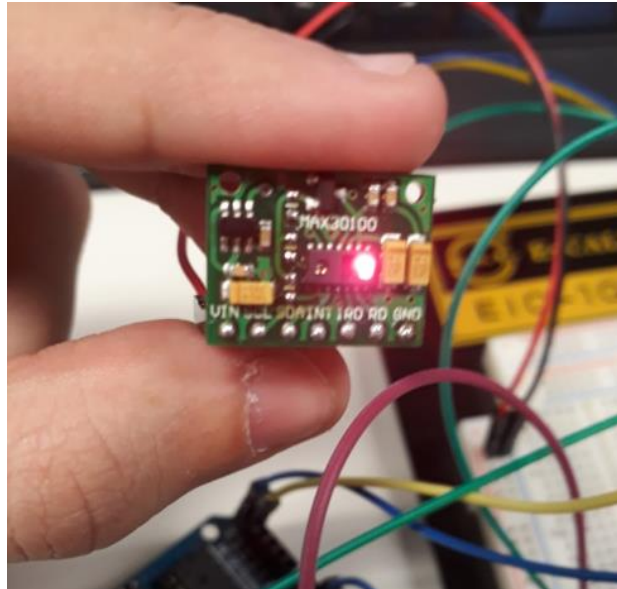
3.1.1 SENSOR DE BATIMENTO CARDÍACO E OXIMETRIA

O módulo MAX30100 foi projetado para atender aos requisitos de um dispositivo *wearable* por ser um dispositivo pequeno e com característica de baixo consumo de energia, operando na faixa de tensão entre 1.8 e 3.3 VDC e diminuindo seu consumo de energia em modo de espera, permitindo que seja alimentado à bateria por longos períodos.

O MAX30100, representado na Figura 4, é um dispositivo de comunicação serial I2C (protocolo serial de comunicação) e a configuração foi realizada através de registros de *software* (MAXIM INTEGRATED PRODUCTS, 2014). Esse dispositivo é composto por 2 LEDs, um vermelho outro infravermelho, um fotodetector, um processamento de sinal de baixo ruído e apresentou

funcionalidade avançada quanto ao seu desempenho de medição devido ao sistema de cancelamento de luz ambiente, alta capacidade de taxa de amostragem e rápida capacidade de saída de dados.

Figura 4 - Sensor MAX30100



Fonte: Os Autores, 2019.

3.1.2 SUBSISTEMAS

O subsistema é composto por ALC (*Ambient Light Cancellation*), ADC (Conversor Analógico-Digital) sigma-delta de 16-bits (programado de 50Hz a 1kHz) e um filtro próprio de tempo discreto, o qual pode rejeitar interferência de 50Hz/60Hz e ruído residual de baixa frequência.

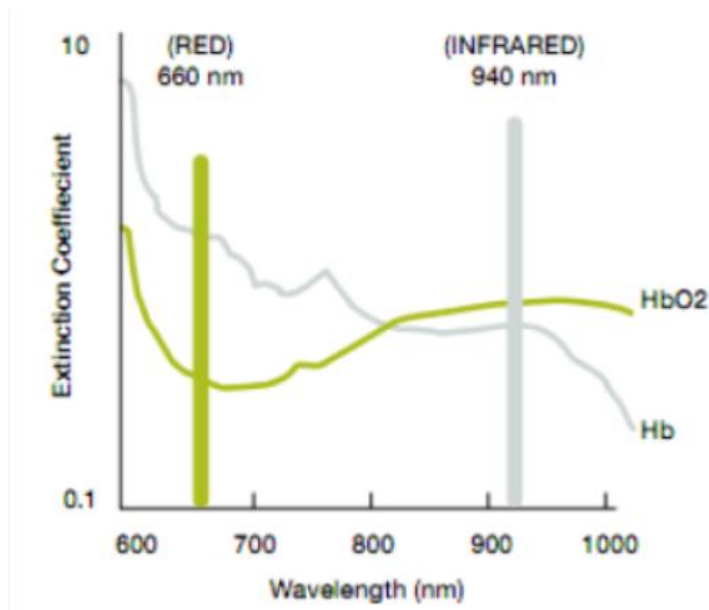
O módulo também foi composto de um sensor de temperatura interna que auxilia o subsistema SpO2 para a coleta de dados de oximetria de pulso. Um problema que ocorreu na medição foi a mudança de temperatura ambiente que afeta a precisão da leitura do comprimento de onda do LED vermelho. Utilizando os dados do sensor de temperatura interna foi possível compensar esse erro de SpO2. Os *drivers* de LEDs foram encontrados no MAX30100 e geraram pulsos de LED. A largura de pulso do LED entre uma de faixa de 200 μ s à 1.6 ms, foi programada para otimizar a precisão da medição e o consumo de energia com base nos casos de uso.

3.1.3 CAPTAÇÃO DE DADOS

Primeiramente, observa-se que os LEDs vermelho (R) e infravermelho (IR) emitem luz em diferentes comprimentos de onda da luz: vermelho (650 nm) e infravermelho (950 nm). A partir do espectro, podem ser distinguidos o LED vermelho do infravermelho.

Para os dados de batimento cardíaco, o módulo fez com que o LED IR emitisse luz no corpo do bebê. Medindo a luz refletida, foi estimado o batimento cardíaco. O fotodiodo, como já dito anteriormente, é um componente do módulo MAX30100 e tem como função medir a absorção ou reflexão de luz que ocorre em certa parte do corpo, representado na seguinte Figura 5 (STROGONOV, 2017).

Figura 5 - Gráfico da taxa de absorção da luz pelo espectro



Fonte: Raivis Strogonovs, 2017.

A leitura do batimento cardíaco foi realizada a partir da intensidade da luz IR refletida no sangue. Quando o coração contrai, maior é a absorção de luz nas hemoglobinas e menor é a reflexão de luz. Quando o coração realiza o movimento de relaxamento, menor é a absorção de luz no sangue e, portanto, maior a reflexão de luz. Logo, o sensor detecta uma maior intensidade de luz.

A partir desse princípio e alguns ajustes de onda que o sensor detecta (como remover o sinal DC, filtro mediano médio e filtro passa-baixa), foi possível determinar o intervalo entre um batimento e o outro, viabilizando encontrar o parâmetro bpm (batimentos por minuto).

A oximetria de pulso é um dispositivo que possibilita realizar a medição de quantidade de oxigênio no sangue em percentual. Ao emitir uma luz IR e R no corpo de um bebê, uma parte da luz é absorvida pelas hemoglobinas e pelas oxihemoglobinas. Para diferentes comprimentos de onda de luz, as absorções de luz em hemoglobinas são diferentes. O mesmo ocorre para oxihemoglobinas. No caso de haver mais hemoglobinas presentes na artéria, uma maior quantidade de luz vermelha tende a ser absorvida. Para o caso de ter mais oxihemoglobinas presentes, uma maior quantidade de luz infravermelha é absorvida. A partir dessa relação de luz R e IR absorvida no sangue foi calculado o nível de oxigênio.

3.1.4 COMUNICAÇÃO I2C

O módulo MAX30100 utiliza o protocolo de comunicação serial I2C. Esse protocolo conecta o dispositivo mestre (PIC24FJ128GA306) ao escravo (MAX30100) através de duas vias seriais: a via de *clock* (SCL) e a via de transmissão de dados (SDA). O funcionamento do I2C é síncrono e a saída de bits é sincronizada com a amostragem de bits por um sinal de *clock* na via SCL. A via SDA é responsável pelo envio e recebimento de dados em função do *clock* da SCL.

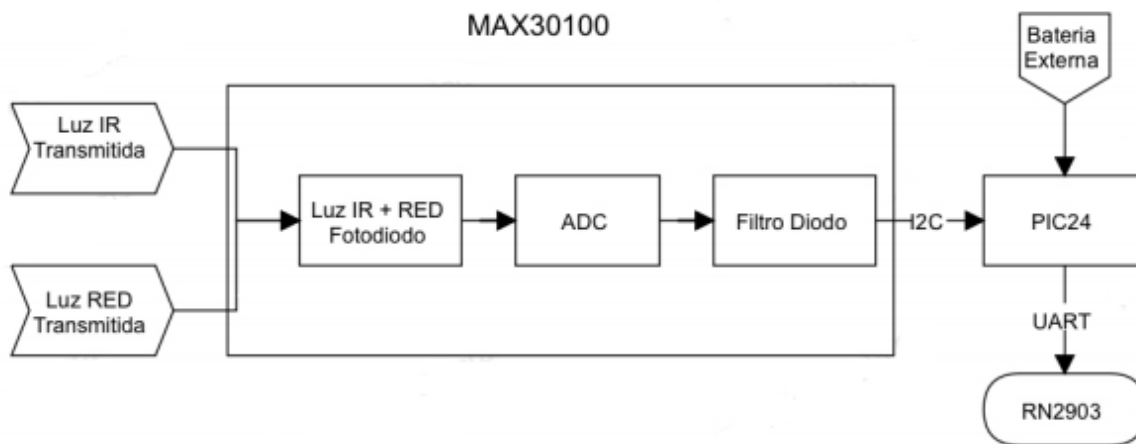
O sinal é transferido em sequências de 8 *bits* sendo necessária uma condição de início do comando e outra de para encerramento. A condição de início é dada pela mudança de estado do valor digital da linha SDA de nível alto para nível baixo quando o SCL tem valor digital de alta. A condição de encerramento, por sua vez, é dada quando o estado de SDA muda de baixa para alta com SCL em alta. Após realizar a condição de início, são enviados 8 *bits* que correspondem ao ID do *escravo* (os 7 primeiros *bits* são o endereço e o último *bit* é comando de escrita ou leitura, Se $R/\overline{W} = 0$, a condição é de escrita; se $R/\overline{W} = 1$, a condição é de leitura). Depois de enviar os 8 *bits* tem o *bit* de *acknowledge* (nono *bit*, ACK) que é usado pelo dispositivo escravo para indicar se recebeu com êxito a sequência anterior de *bits*. Em

sequência, o endereço de registro, que é o local da memória que se quer acessar, e enviado e é seguido de outro ACK.

3.1.5 LEITURA DE DADOS E COMUNICAÇÃO COM MICROCONTROLADOR PIC

As luzes emitidas pelos LEDs atravessam a corrente sanguínea, tendo parte absorvida ou refletida, foram medidas pelo fotodiodo presente. As informações coletadas pelo fotodiodo são analógicas e, portanto, passaram por um filtro analógico/digital para converter os dados em valores digitais. Em seguida os dados passaram por um filtro digital com a finalidade de diminuir as interferências presentes no sinal. Após esse processo, as informações foram enviadas do escravo para o mestre através da comunicação serial I2C, em que o mestre é o PIC24FJ128GA306 e o escravo é MAX30100. O PIC24FJ128GA306 foi alimentado por uma bateria externa. O diagrama do sistema foi representado, pela Figura 6 abaixo.

Figura 6 - Diagrama do fluxo de dados do MAX30100 até o RN2903.



Fonte: Os Autores, 2019.

3.1.6 MAX30100: ESQUEMA ELÉTRICO E PINAGEM

Para a programação do *hardware*, foi necessário o entendimento do esquema elétrico do módulo e como deve ser ligado à placa controladora (mestre). As funções de pinagem do sensor MAX30100 foram apresentadas na Tabela 6 (ALI, 2019).

Tabela 6 - Pinagem do Sensor MAX30100

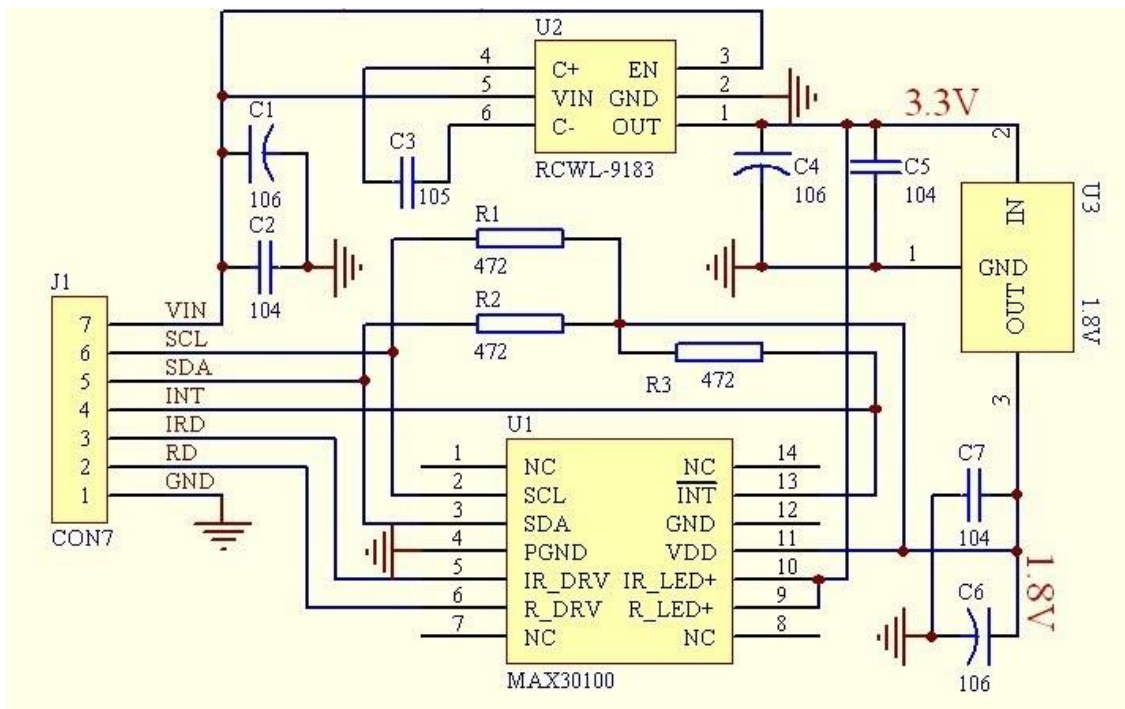
PIN	Tipo	Parâmetros
1, 7, 8, 14	NC	Terminais sem utilidade, esses terminais ficam em aberto.
2	SCL	Terminal usado para o <i>clock</i> I2C para entrada.
3	SDA	Terminal de envio de dados bidirecional I2C.
4	PGND	Terminal usado para energia dos condutores de diodos emissores de luz.
5	IR_DRV	Terminal é usado para o cátodo de LED infravermelho para se conectar com o ponto de drivers de LED.
6	R_DRV	Terminal é para o cátodo do LE vermelho se conectar com o seu ponto de acionamento.
9	R_LED +	Terminal é para o LED vermelho conectar com a tensão de alimentação.
10	IR_LED +	Terminal ajuda a conectar o ânodo de LED infravermelho à tensão de entrada.
11	VDD	Terminal da fonte de entrada para este módulo.
12	GND	Terminal de aterramento analógico.
13	INT	Terminal de interrupção <i>Active-Low</i> .

Fonte: Zahid Ali, 2019.

Os terminais SDA, SCL e INT são saídas de *open-drain*, o que significa que são necessários resistores de 4,7 k Ω para realizar o devido funcionamento.

O sensor foi ligado ao módulo pelo esquema elétrico conforme a Figura 7 (Mikro Electron, 2018):

Figura 7 - Esquema Elétrico do módulo MAX30100



Fonte: Mikro Electron, 2018.

3.1.7 SMART CAMPUS MAUÁ

O *Baby IoT* utilizou a plataforma Smart Campus Mauá para a aquisição de dados que foram enviados pelo sensor. Esse sistema possui uma estrutura de uma rede LoRaWAN com *gateway LoRaWAN*, *Network Server LoRaWAN* e uma plataforma de captura e armazenamento de dados.

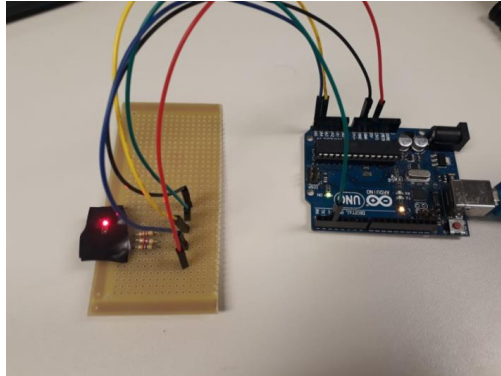
O PIC24FJ128GA306 obtém os dados através dos sensores e os envia para o módulo de rádio frequência RN2903A através de comunicação serial UART (Universal Asynchronous Receiver/Transmitter). Esse módulo envia o pacote de dados, criptografados, para o gateway LoRaWAN via rádio frequência do tipo LoRa segundo o protocolo LoRaWAN. O *gateway* empacota estes dados e envia o novo pacote ao Network Server LoRaWAN, o qual disponibiliza os dados à aplicação a ele conectada.

3.1.8 PROGRAMAÇÃO EM ARDUINO

Primeiramente, o módulo MAX30100, foi ligado a uma placa Arduino UNO a fim de se observar os conceitos da programação do *hardware* para coleta e transmissão de dados até a

comunicação com o Smart Campus Mauá. Em seguida foi aplicado esses conceitos aprendidos em um microcontrolador PIC24FJ128GA306.

Figura 8 - Sensor MAX30100 conectado ao Arduino UNO

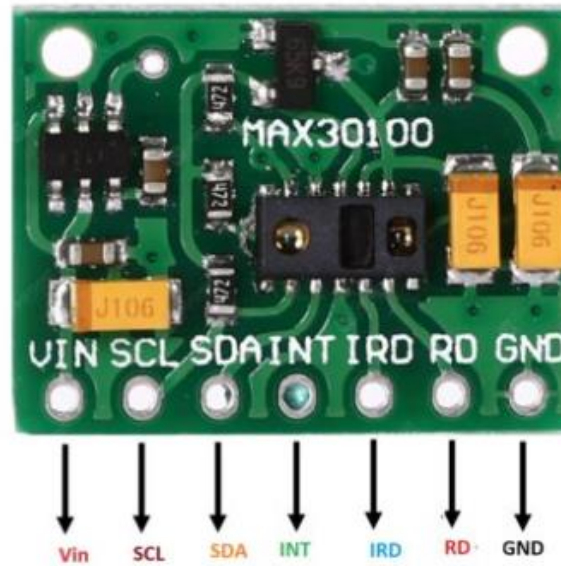


Fonte: Os Autores, 2019.

Segundo as especificações técnicas do módulo MAX30100, a tensão elétrica para o seu correto funcionamento é de 1,8 V. O Arduino, entretanto, pode ser alimentado com tensões de 3,3 V a 5,0 V. Segundo a Figura 8, os terminais SCL e SDA foram ligados com de resistores de 4,7 k Ω , pois sem estarem conectados não houve funcionamento do sensor diante de microcontroladores de nível lógico alto (Arduino). A solução para este problema foi remover os resistores da placa e adicionar resistores de 4,7 k Ω externos. Esses novos resistores foram ligados na saída VIN do módulo, sendo um ligado ao SCL, outro ao SDA e outro ao INT.

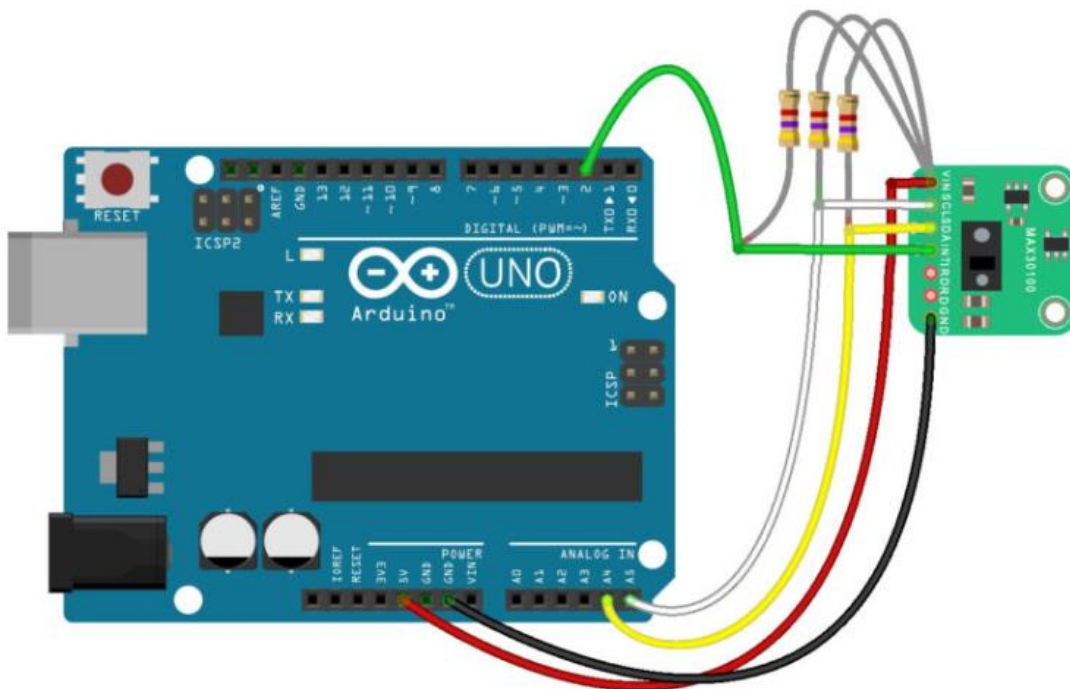
Resolvendo a condição de contorno, foi ligado o módulo MAX30100 (escravo) ao Arduino (mestre), utilizando apenas as seguintes saídas do módulo: VIN, GND, SCL, SDA e INT, Figura 9. A ligação realizada para comunicação entre o mestre e escravo foi dada da seguinte maneira: a saída VIN é conectada ao 5V do Arduino, o GND do módulo foi ligado ao GND do Arduino, o SDA e o SCL foram conectados às portas de entrada analógica do Arduino (A4 e A5, respectivamente) e o INT foi ligado a uma porta de entrada digital do Arduino (porta 2).

Figura 9 - Pinagem do Sensor MAX30100



Fonte: Zahid Ali, 2019.

Figura 10 - Representação da montagem do Sensor MAX 30100 conectado ao Arduino



Fonte: Teach Me Micro, 2018.

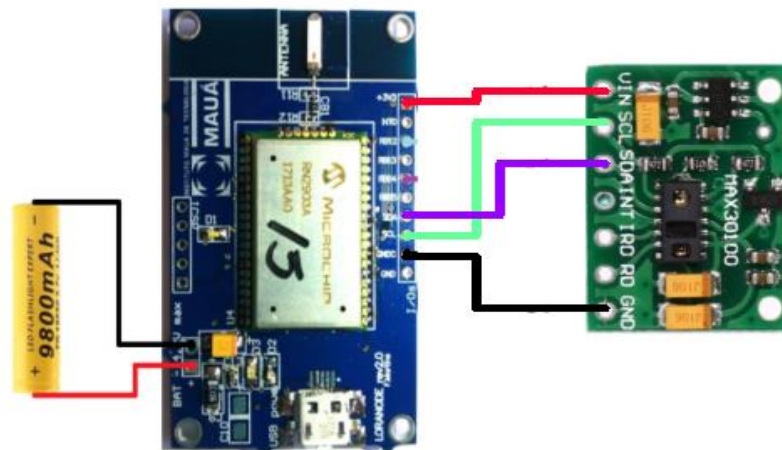
O teste com Arduino da Figura 10 (Teach Me Micro, 2018), seguindo o esquemático do Teach Me Micro (2019), resultou no funcionamento do sensor, porém, com uma alta variação da leitura de batimento cardíaco. A variação na leitura foi aceitável devido aos movimentos de

contração e relaxamento que afetam o fluxo de hemoglobinas que fazem com que medições possuam uma boa variação de uma leitura de uma para outra.

No Arduino, foi programado um sistema que realizou a média aritmética do batimento cardíaco e da oximetria utilizando 10 valores de leitura do sensor para realizar a média. No programa também foi programado uma restrição de leitura, para que valores acima 220 bpm não fossem contabilizados no cálculo de média, diminuindo o erro obtido por leitura de valores que estão acima do que o sensor consegue observar. Outro fator importante a ser percebido é a relação da configuração da biblioteca com a programação. Para uma melhor leitura do MAX30100 foi necessário alterar o limite de pulso mínimo na biblioteca para 20 bpm e colocar na programação uma frequência de quase 37Hz para chamar o *update* da leitura.

Após os testes utilizando o Arduino, foi montado o seguinte esquema eletrônico utilizando o PIC24FJ128GA306 representado na figura 11.

Figura 11 - Esquemático eletrônico utilizando o MAX30100, PIC24FJ128GA306 e RN2903A.



Fonte: Os Autores, 2019.

Este esquemático é caracterizado pelas seguintes ligações no sensor MAX30100:

- a) Pino VIN do MAX30100 conectado com o pino +3,3 V do módulo LoRaWAN.
- b) Pinos SCL, SDA e GND conectados entre os módulos.

- c) Positivo da bateria conectado com o BAT pino 1 e negativo com o BAT pino 2 no módulo LoRaWAN.

3.1.9 SENSOR ACCELERÔMETRO

O MPU6050 é um módulo que possui um sensor de 6 eixos (6 valores de saída), sendo 3 valores do acelerômetro e 3 do giroscópio. O sensor tem dimensões 19x14x3 mm compatíveis para aplicações em *wearables*, e possui comunicação através de protocolo I2C para comunicação serial.

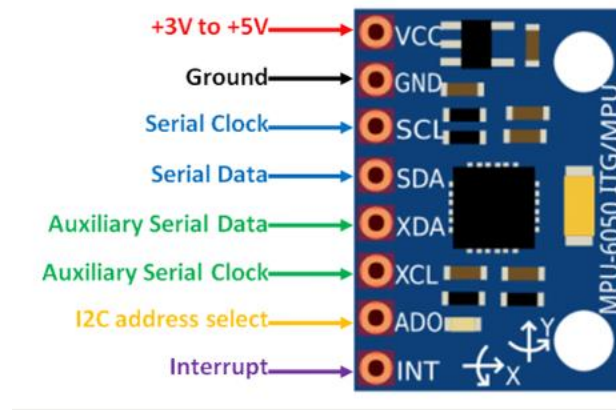
3.1.9.1 Funcionamento

O acelerômetro funciona pelo princípio de uma caixa com uma bola pequena dentro. As paredes dessa caixa são feitas com cristais piezoelétricos. Esses cristais geram tensão elétrica quando há uma pressão mecânica sobre eles. Portanto, quando a bola por efeito aceleração incidir sobre uma ou mais paredes será gerada tensão elétrica sobre elas. Cada par de paredes do cubo (caixa) representa um eixo no espaço tridimensional (x, y, z).

3.1.9.2 Programação em Arduino

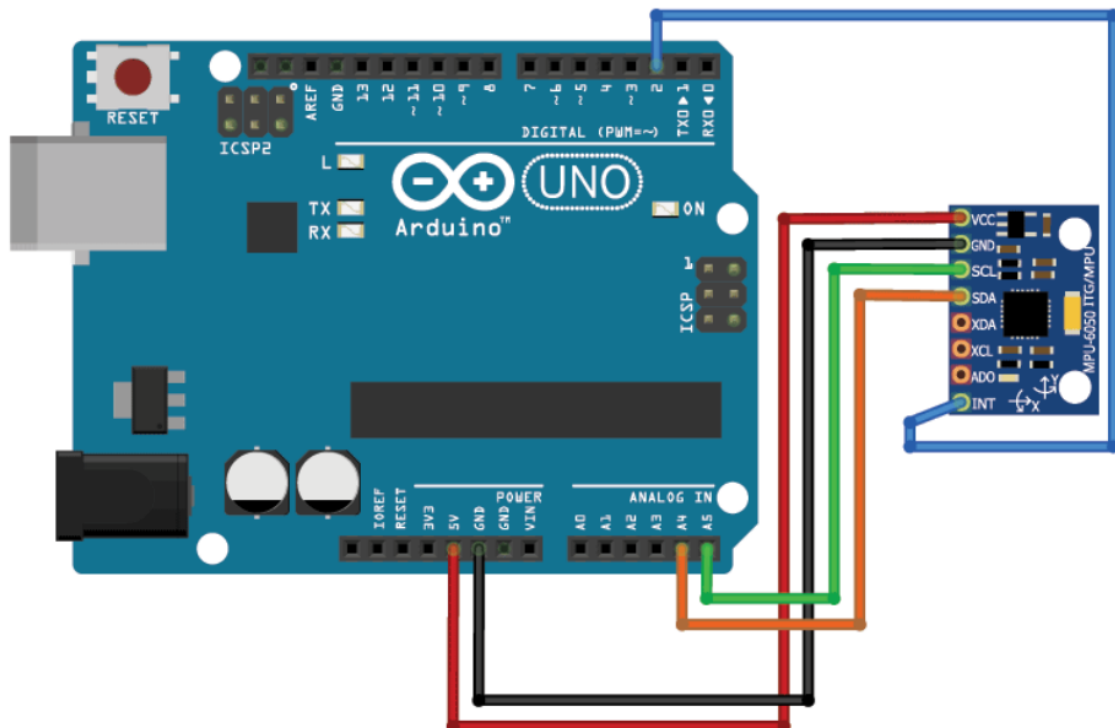
O módulo MPU6050, tem protocolo de comunicação serial I2C, como o MAX30100. Portanto, para comunicar o acelerômetro com o Arduino foram utilizadas as mesmas conexões: VIN (módulo) ligado ao 5 V (Arduino), GND (módulo) ligado ao GND (Arduino). O SDA e SCL foram conectados às portas de entrada analógica do Arduino (A4 e A5, respectivamente) e o INT foi ligado a uma porta de entrada digital do Arduino (porta 2). Representado pelas Figuras 12 e 13 (Componentes101, 2018).

Figura 12 - Pinagem MPU6050



Fonte: Components101, 2018.

Figura 13 - Esquema de Montagem do Sensor MPU6050 no Arduino UNO

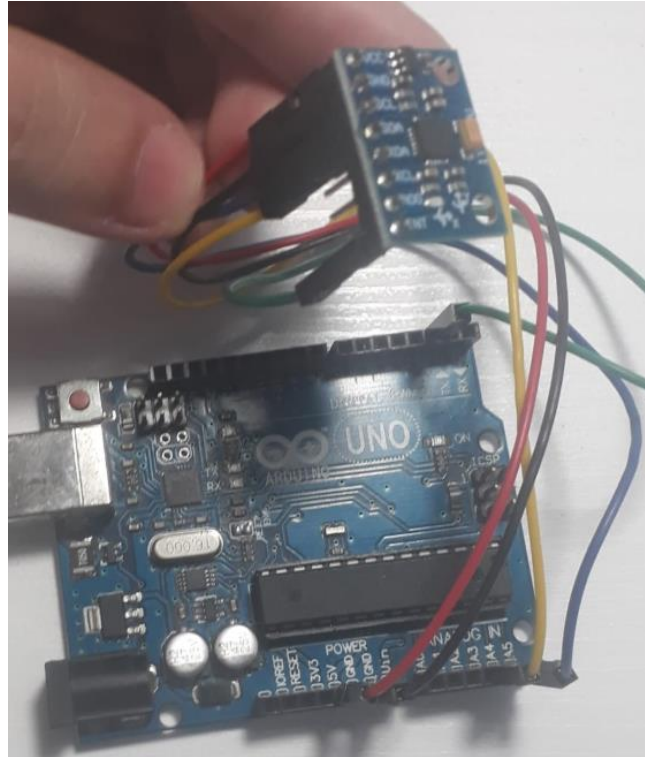


Fonte: Components101, 2018.

O sensor foi testado no *software* do Arduino, Figura 14, e obteve-se sucesso em seu funcionamento. Para fins de resultado, o sensor MPU6050, Figura 15, foi programado para quando mudar da posição base gerar um sinal de interrupção que acorda o sensor

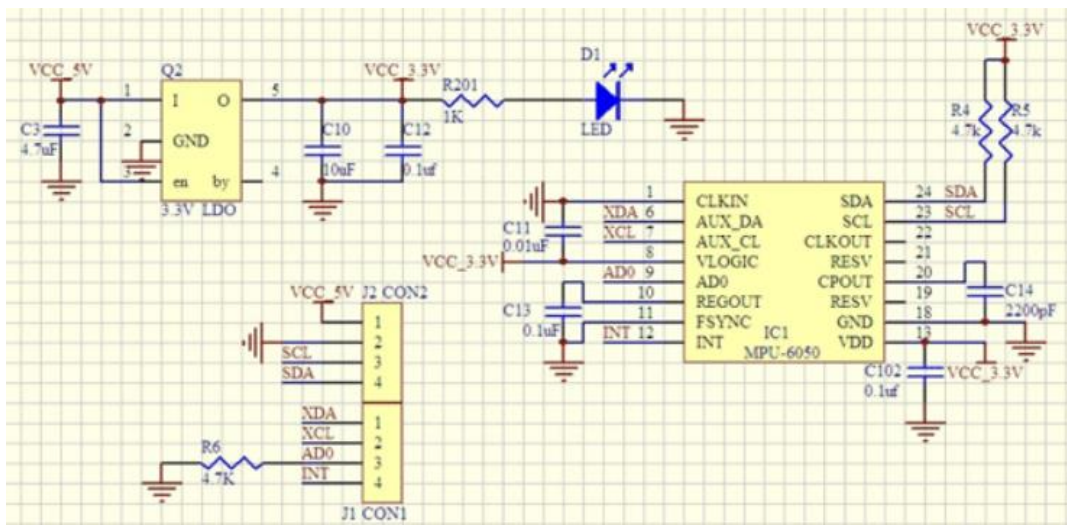
MAX30100 (saindo do *sleep mode*). A posição base representou a posição ideal de sono em que o bebê deve estar.

Figura 14 - Sensor MPU6050 ligado ao Arduino UNO



Fonte: Os Autores, 2019.

Figura 15 - Esquema Elétrico do MPU6050



Fonte: Components101, 2018.

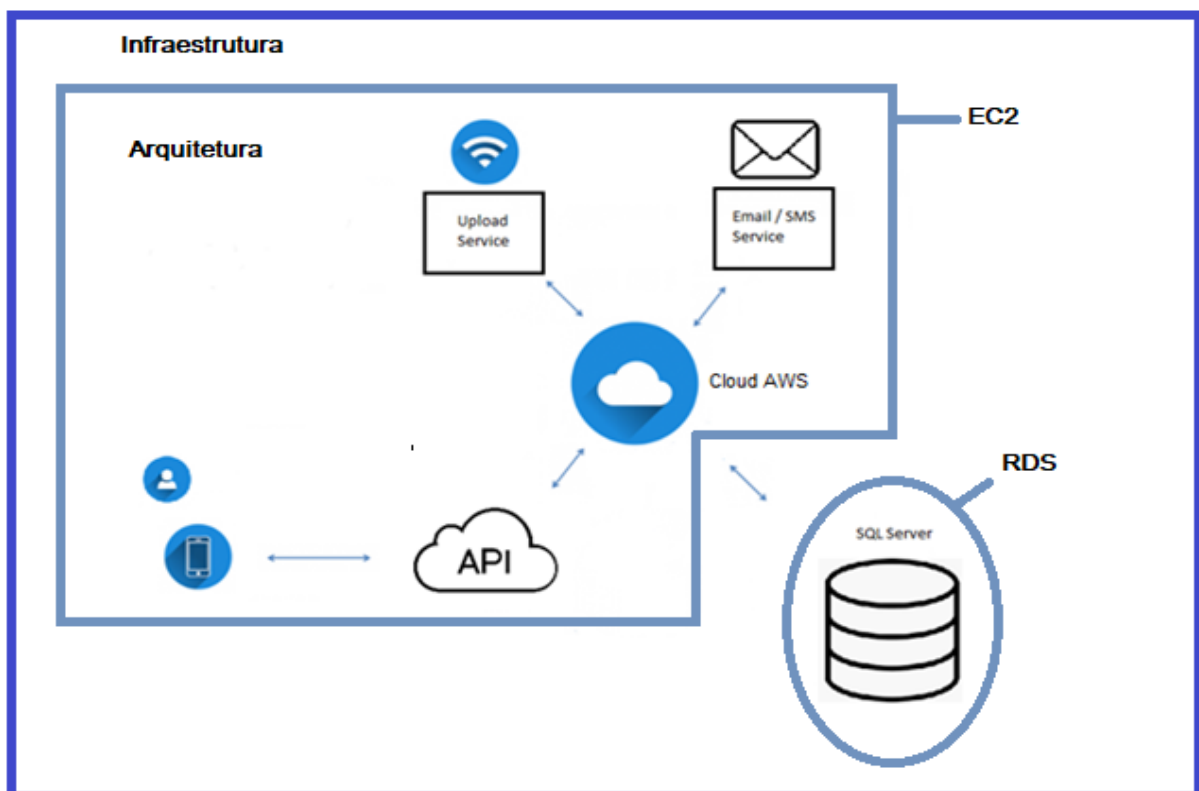
3.2 SOFTWARE

O sistema desenvolvido tem por finalidade trabalhar os dados transmitidos pelo *hardware*, a fim de disponibilizar informações úteis para os usuários através do aplicativo *mobile*.

3.2.1 ARQUITETURA DO SISTEMA

O sistema desenvolvido pode ser subdividido em quatro partes: aplicativo *mobile*, API, banco de dados e serviços. O fluxo de comunicação se encontra na Figura 16.

Figura 16 – Arquitetura do sistema



Fonte: Os Autores, 2019.

A arquitetura desenvolvida foi baseada em micro serviços, isto é, o sistema foi dividido em pequenas aplicações em que cada aplicação foi responsável por uma pequena atividade dentro do sistema. A arquitetura em micro serviços foi vantajosa pois permitiu que um sistema

de grande porte seja escalável, de fácil manutenção e menos propenso a quedas de conexão na rede.

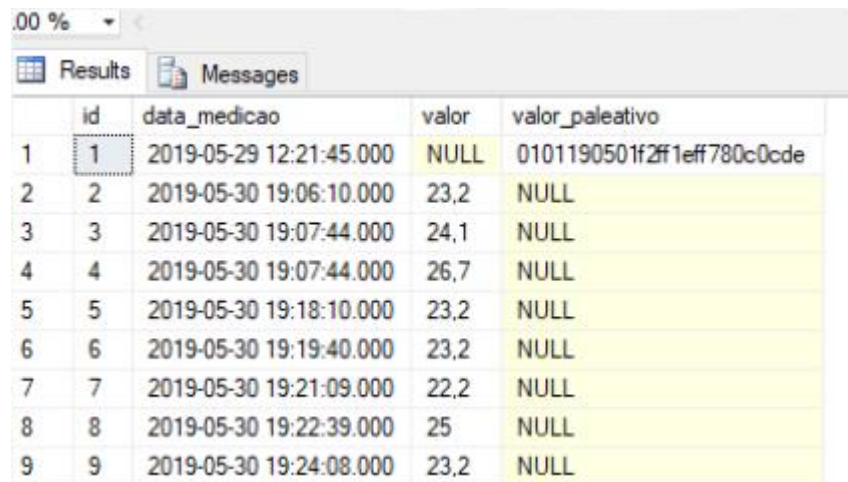
3.2.2 BANCO DE DADOS

O banco de dados foi a parte do sistema responsável por armazenar todos os tipos de dados:

- a) cadastro de usuários: nome, CPF, endereço, número de telefone, idade;
- b) cadastro de bebês: nome, idade, dispositivo LoRaWAN associado, ID único do usuário associado, ID único do alarme associado;
- c) configuração de alarmes: intervalo de valores de cada tipo de medição, tipo de envio de alarme, ID único do bebê associado ao alarme;
- d) medições: registros de todos os tipos de medições, com a data de cada medição;
- e) relacionamento entre os dados: vínculos entre bebês e usuários, bebês e alarmes, bebês e dispositivos, bebês e medições.

O banco de dados utilizado foi o SQL Server, Figura 17. Sua função foi prioritariamente armazenar, relacionar e dispor os dados de forma coerente. Qualquer lógica de negócio que envolva cálculos, funções condicionais e *loops* não foram implementados nesta parte do sistema.

Figura 17 - Exemplo de medição salva no banco de dados.



	id	data_medicao	valor	valor_paleativo
1	1	2019-05-29 12:21:45.000	NULL	0101190501f2ff1eff780c0cde
2	2	2019-05-30 19:06:10.000	23,2	NULL
3	3	2019-05-30 19:07:44.000	24,1	NULL
4	4	2019-05-30 19:07:44.000	26,7	NULL
5	5	2019-05-30 19:18:10.000	23,2	NULL
6	6	2019-05-30 19:19:40.000	23,2	NULL
7	7	2019-05-30 19:21:09.000	22,2	NULL
8	8	2019-05-30 19:22:39.000	25	NULL
9	9	2019-05-30 19:24:08.000	23,2	NULL

Fonte: Os Autores, 2019.

3.2.3 API

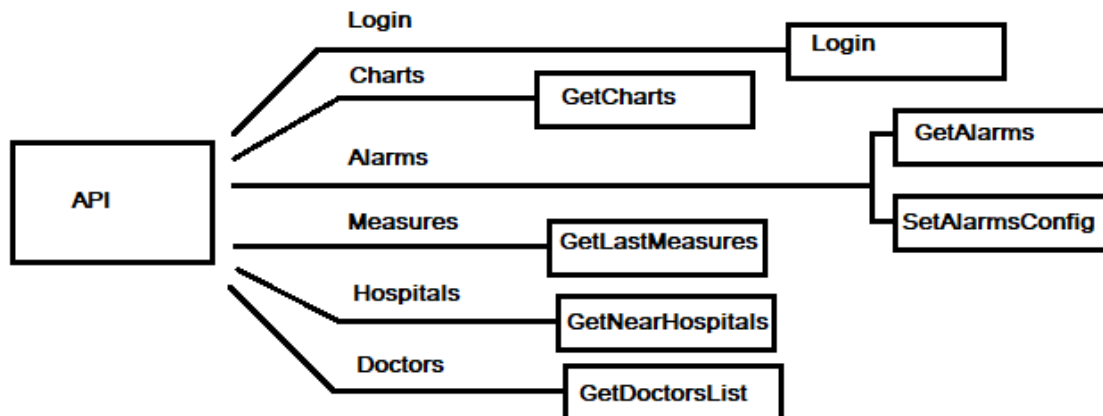
A API, *Application Programming Interface*, gerenciou o fluxo de dados do sistema, Figura 18. Dados que precisaram ser buscados no banco de dados e exibidos no aplicativo, por exemplo, precisaram necessariamente passar pela API.

A estrutura da API foi dividida em rotas específicas para a aplicação que requisitou algum dado. Portanto, criou-se rotas específicas para requisições que vem do aplicativo, que diferentes das rotas para requisições dos serviços de *e-mail* e *upload*, e assim sucessivamente.

Os cálculos de regras de negócio foram implementados nas rotas da API. Os tratamentos necessários para disponibilizar gráficos com a média das medições, por exemplo, foram feitos na rota de gráficos da API.

Questões como acessibilidade das rotas e segurança dos dados também fizeram parte desta parte do sistema. A API gerou *tokens* criptografados de acesso às requisições dos aplicativos, verificou se o usuário tem acesso à rota que está requisitando e criou *logs* em casos de erro.

Figura 18 - Rotas da API



Fonte: Os Autores, 2019.

A API foi desenvolvida em linguagem C#, utilizando o *Visual Studio* como interface de programação. O *Web Service* para publicação foi instanciado em uma máquina virtual da *Amazon Web Services*.

3.2.4 SERVIÇOS

Os serviços desenvolvidos funcionaram como *Jobs*. *Jobs* são aplicações que executam uma determinada tarefa periodicamente, de acordo com a sua função.

3.2.4.1 Upload

O *upload* foi o serviço que capturou o *payload* (pacote) transmitido pelo *hardware*. Ele recebeu a mensagem, decodificou o conteúdo, criou um objeto representativo e enviou para a API chamando a rota correspondente.

3.2.4.2 E-mail e SMS

O serviço de *e-mail* e SMS pode também ser chamado de serviço de alarmes, pois ele disparou os alarmes de acordo com a configuração de cada usuário.

Esse serviço foi executado em intervalos de um minuto, chamando a rota de alarmes da API. A API, por sua vez, executou uma *procedure* (procedimento) no banco de dados que verificou

se houve alguma medição anormal naquele minuto, de acordo com a configuração de cada usuário. Após ter processado os dados no banco de dados, a API retornou para o serviço um objeto contendo uma lista dos alarmes de todos os usuários que devem ser disparados. Por fim, o serviço montou e enviou os e-mails para cada um dos usuários cadastrados.

3.2.4.3 Tecnologia Utilizada

Os serviços foram desenvolvidos utilizando a linguagem C# e o *Visual Studio* como IDE fazendo uso do serviço da *Amazon Web Services*, como *contêiner* para as aplicações.

3.2.5 APLICATIVO MOBILE

O aplicativo foi a plataforma de visualização do usuário. Portanto, foi necessário que ele fosse simples, atrativo e de fácil navegação.

A principal função do aplicativo foi disponibilizar os dados úteis para os usuários, tais como últimas medições do *hardware*, as configurações dos alarmes, gráficos do histórico de medições, informações dos bebês (idade, nome) e diagnósticos.

O funcionamento de comunicação foi similar ao descrito no item 5.1.3: o aplicativo enviou uma requisição para a API, que buscou os dados no banco de dados e retornou o objeto para ser disponibilizado ao usuário.

O aplicativo utilizou, principalmente, as rotas de medições, alarmes, diagnósticos, pediatras, gráficos e hospitais.

Como falado anteriormente, o aplicativo necessitou ser de fácil navegação. Para tanto, o aplicativo foi subdividido de acordo com seus recursos. Os recursos são ferramentas úteis aos usuários que fornecem utilidades e soluções, agregando valor à plataforma sendo desenvolvidas as seguintes: alarmes, gráficos, últimos dados registrados, lista de pediatras, diagnósticos e hospitais próximos.

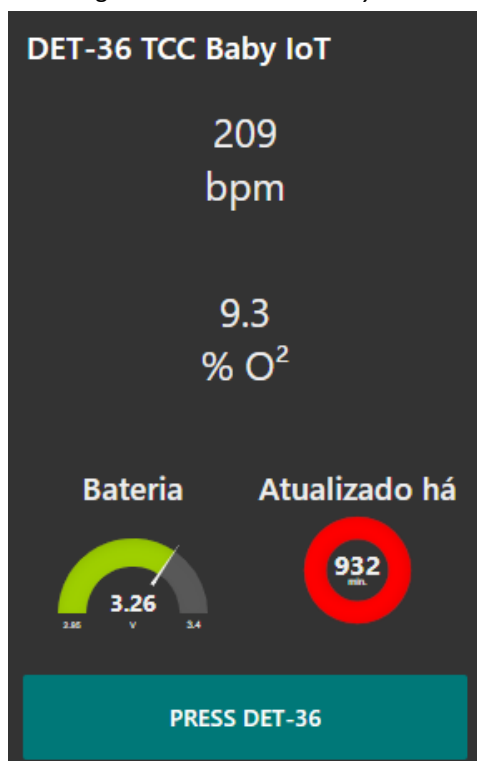
Os valores coletados pelos sensores foram demonstrados por meio de um *dashboard*, presente no *site* do Smart Campus Mauá, relatando seu funcionamento operacional. Diferentemente do aplicativo desenvolvido, que tem como finalidade deixar as informações mais acessíveis, com uma melhor visualização, sendo o produto final.

3.2.5.1 *Dashboard*

Com o *dashboard* da Figura 19, presente no *site* Smart Campus Mauá (2019), foi possível verificar o cadastro do dispositivo denominado como DET-36, sendo uma monitoração vinculada com o setor do Centro de Pesquisas, evidenciando as seguintes informações:

- a) Nomenclatura do módulo: DET-36 TCC *Baby* IoT
- b) Batimentos cardíacos em bpm.
- c) Taxa de oximetria.
- d) Nível de tensão da bateria do módulo desenvolvido.
- e) Tempo de atualização das informações em minutos.

Figura 19 - *Dashboard Baby IoT*



Fonte: Smart Campus Mauá, 2019.

3.2.5.2 Funcionalidades

A solução gráfica do aplicativo será descrita a seguir.

Primeiramente foi desenvolvido a tela de *Login* dos usuários e com o *slogan* do produto, Figura 20.



Fonte: Os Autores, 2019.

A tela seguinte, Figura 21, foi determinada como a tela principal, sendo mostrado variadas informações como os bebês que estão cadastrados pelo usuário, a última medição realizada e os valores do monitoramento do recém-nascido e o ambiente, mostrando em verde caso o valor esteja dentro dos parâmetros e em vermelho caso a medição esteja fora do padrão correto.

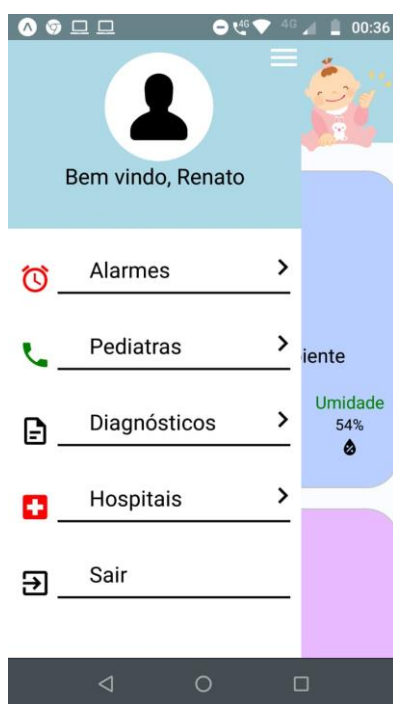
Figura 21 - Tela Inicial



Fonte: Os Autores, 2019.

Com o seguinte menu inicial foi possível verificar o nome do usuário e diversas funções no aplicativo, alarmes, ligação para pediatras, diagnósticos e localização de hospitais, vide Figura 22.

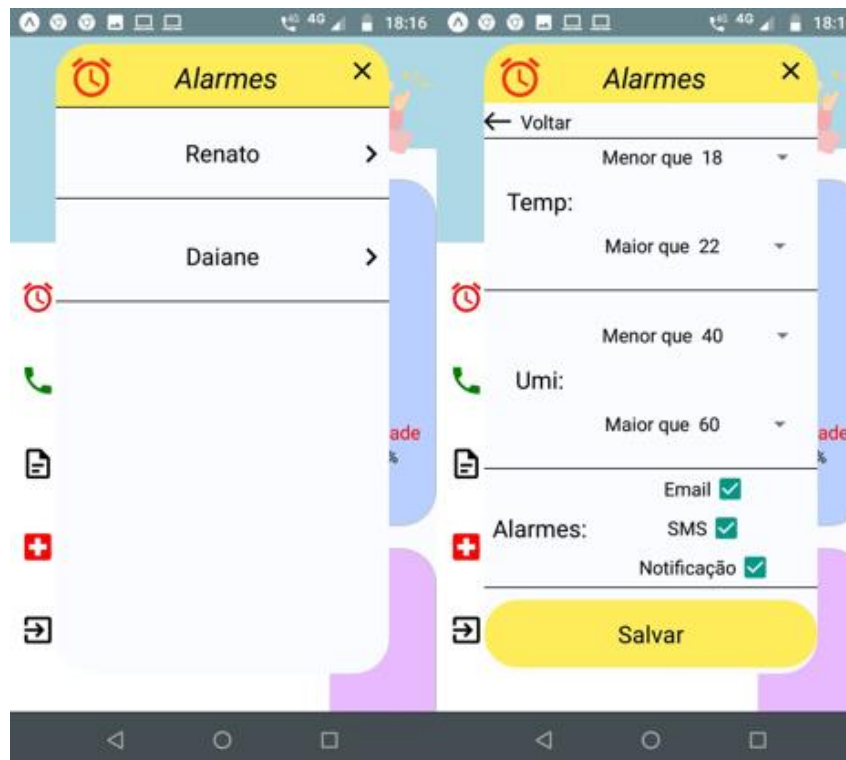
Figura 22 - Menu do aplicativo



Fonte: Os Autores, 2019.

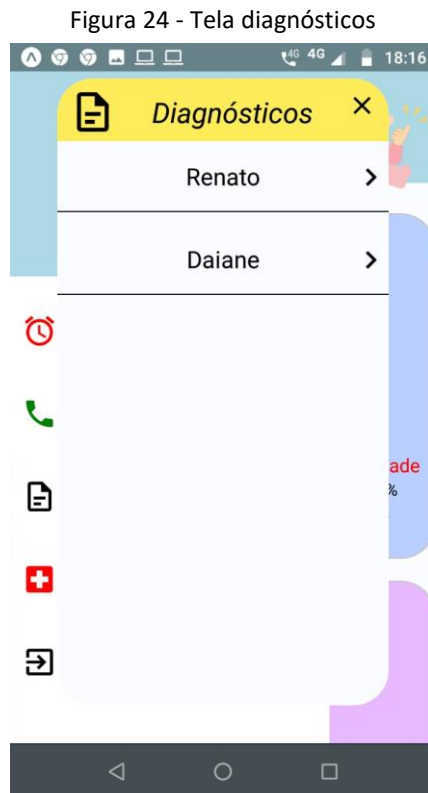
Com os alarmes, Figura 23, foi possível determinar para cada recém-nascido a escala de variação dos parâmetros, sendo um local destinado aos pediatras.

Figura 23 - Tela de alarmes



Fonte: Os Autores, 2019.

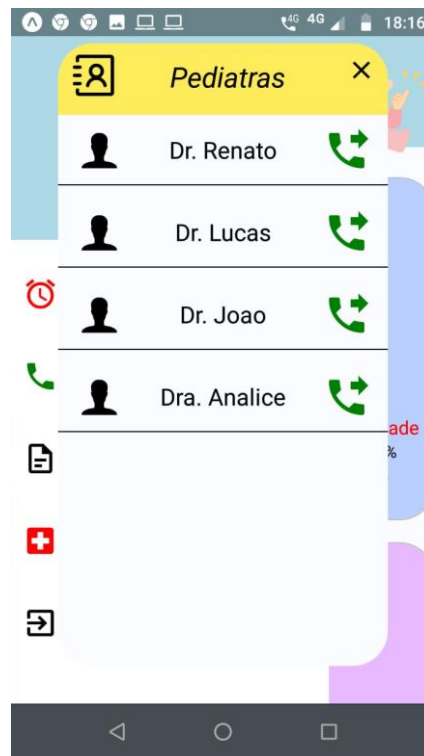
A aba diagnóstico, Figura 24, foi criada com a intenção de alocar documentos como exames de sangue ou receitas médicas que o pediatra tenha solicitado, tendo armazenado todo o histórico de exames dos recém-nascidos.



Fonte: Os Autores, 2019.

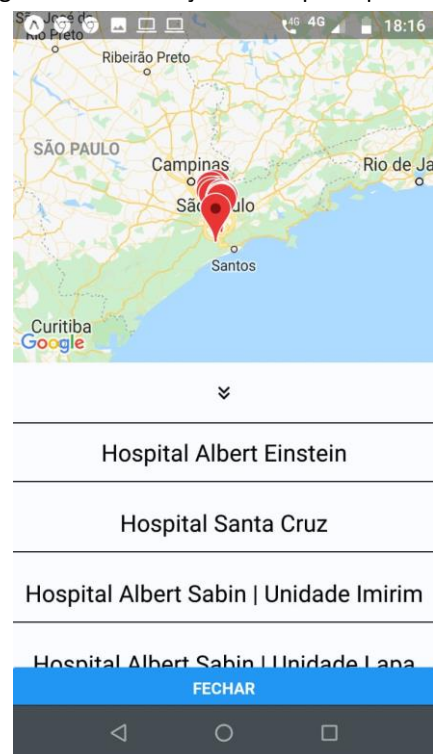
A ligação aos pediatras como também a localização de hospitais, Figura 25 e 26, teve como prioridade na atuação de emergências, vinculando os contatos com a agenda do telefone e o mapa com os hospitais mais próximos, habilitando que o aplicativo requisite a localização e utilize em outra plataforma, como Waze ou Google Maps, para indicar o caminho correto ao hospital escolhido, Figura 27.

Figura 25 - Tela dos contatos dos pediatras



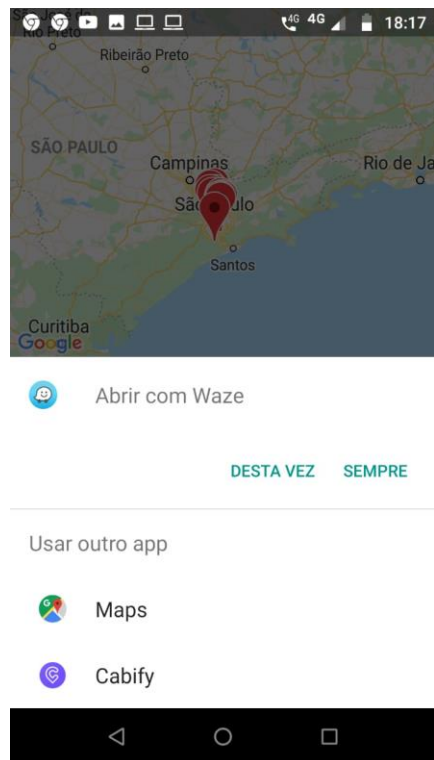
Fonte: Os Autores, 2019.

Figura 26 - Localização de hospitais próximos



Fonte: Os Autores, 2019.

Figura 27 - Aplicativos de rotas vinculados com o *Baby IoT*



Fonte: Os Autores, 2019.

Por fim, o aplicativo foi publicado, mostrado na Figura 28, na Google Play Store (Google, 2019), habilitando o seu *download* para celulares androides.

Figura 28 - Publicação do aplicativo



Fonte: Google Play, 2019.

4 RESULTADOS DO PROJETO

4.1 MONTAGEM DE APLICAÇÃO

A montagem para a aplicação de todo o sistema desenvolvido foi dividida nas seguintes etapas:

- a) determinação da parte da roupa e suporte para aplicação do produto.
- b) desenvolvimento de uma caixa flexível para a placa eletrônica.

4.1.1 APLICAÇÃO DO PRODUTO

Para determinar onde seria testada a placa eletrônica foi determinado um local para melhores medições e a questão do tamanho necessário para ser alocada a placa.

Foi utilizado o programa SolidWorks® para o desenho da caixa, visando atribuir uma imagem infantil no desenho 3D realizado Figura 29.

Figura 29 - Desenho realizado no programa SolidWorks®



Fonte: Os Autores, 2019.

Após o desenho do projeto mecânico, o arquivo foi convertido para habilitar sua impressão 3D, utilizando os recursos disponíveis no FabLab do Instituto Mauá de Tecnologia. O material utilizado foi o ABS, pois é leve, resistente e apropriado para a montagem do protótipo do produto.

Após concluída a impressão 3D da peça, foi preciso lixa-la para retirar qualquer imperfeição impressa. Feito isso foi realizada a pintura com tinta guache, vide Figura 30.

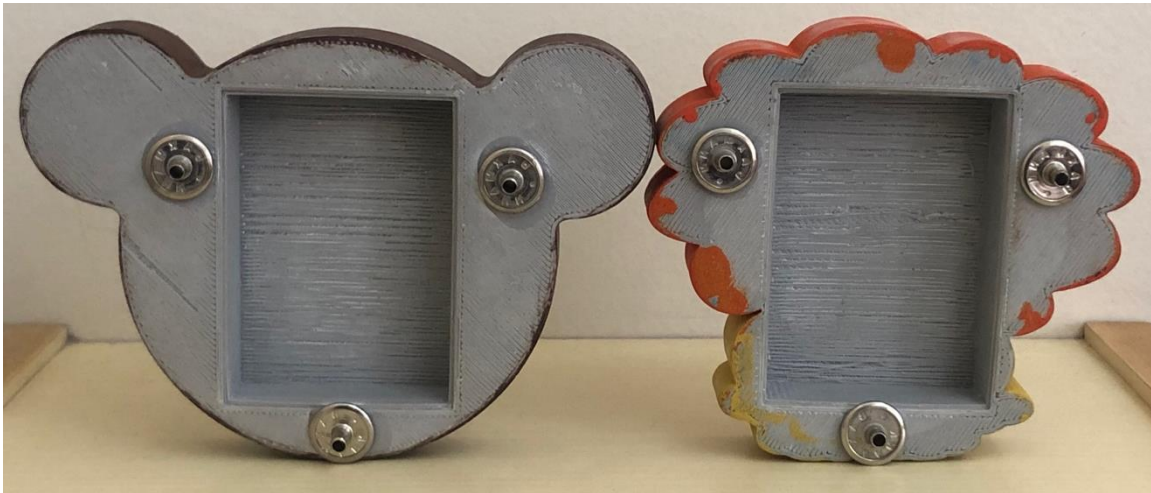
Figura 30 - Processo de pintura



Fonte: Os Autores, 2019.

Feita a pintura, Figura 32, foi preciso colocar botões que se prenderam por pressão, com o intuito de fixar o boneco no macacão do bebê e possibilitar a retirada para lavagem dos macacões, Figura 31.

Figura 31 - Botões de fixação



Fonte: Os Autores, 2019.

A parte de trás dos bonecos possuem os locais para o acoplamento do circuito de monitoramento, este circuito após acomodado foi protegido por uma tampa de acrílico. O acrílico possui a finalidade de proteger o sensor para que o mesmo não fique em contato direto com a pele do bebê. Este método não é invasivo ao bebê e os equipamentos ficam protegidos. Visto isso, o sensor necessita ser calibrado para que o acrílico não interfira nos resultados das medições.

Figura 32 – Peças 3D finalizadas



Fonte: Os Autores, 2019.

Visando o maior conforto do recém-nascido durante a utilização do equipamento, foi desenvolvido um novo protótipo utilizando um material mais flexível feito de silicone.

Com o desenho impresso na impressora 3D, colocou-se a peça dentro de uma caixa, para ser preenchida com silicone, representado na Figura 33.

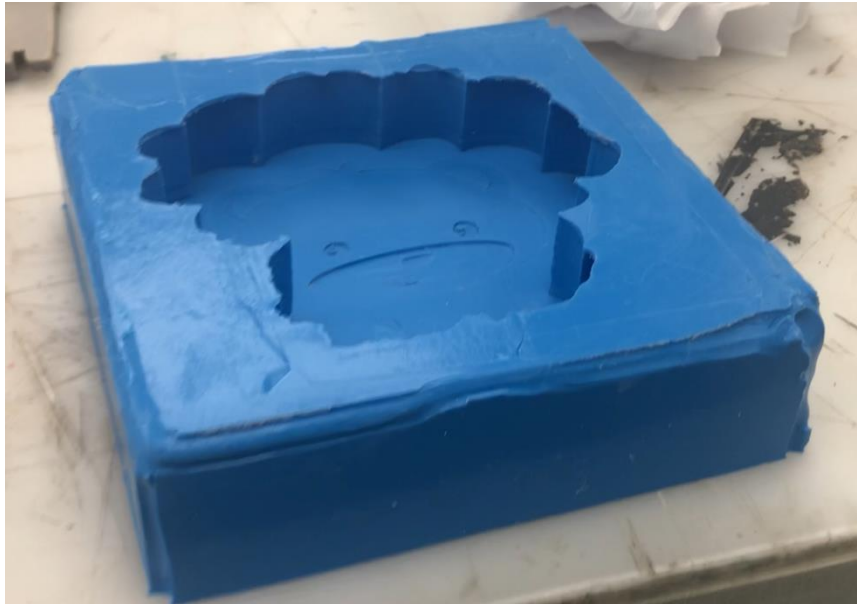
Figura 33 - Caixa para moldar o boneco



Fonte: Os Autores, 2019.

Após o boneco ser posicionado e preso com fita dupla-face ao fundo, foi derramado o silicone na cor azul. O processo de cura do silicone demorou 24 horas. Em seguida foi retirado o boneco do silicone, mostrado na Figura 34.

Figura 34 - Molde de silicone



Fonte: Os Autores, 2019.

Passados 24 horas o boneco de silicone foi retirado do molde com cuidado e o resultado deste processo foi mostrado na Figura 35.

Figura 35 – Peça desenvolvida com silicone



Fonte: Os Autores, 2019.

Após o desenvolvimento da peça, foram aplicadas mudanças no macacão do recém-nascido, colocando pinos para possibilitar o encaixe do protótipo com a roupa conforme a Figura 36.

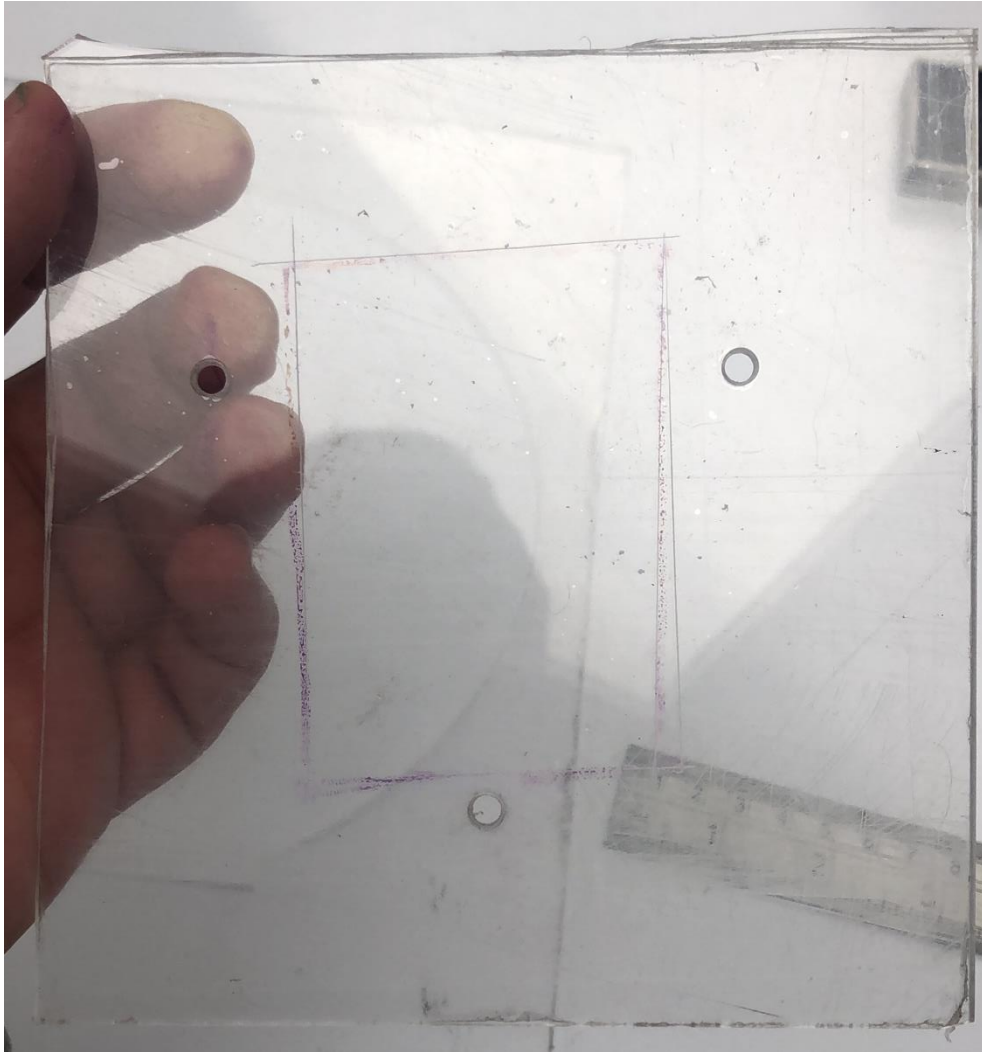
Figura 36 - Roupa adaptada para o protótipo



Fonte: Os Autores, 2019.

O macacão pode ser de qualquer marca, modelo, tamanho e gosto, pois o padrão de furação foi mantido conforme o molde mostrado na Figura 37.

Figura 37 - Molde para furação



Fonte: Os Autores, 2019.

4.2 APLICAÇÕES ASSOCIADAS AO PROJETO

As soluções podem se adequar de forma diferente dependendo do usuário final. Portanto, a fim de organizar de forma clara como o projeto se encaixa no mercado, esta sessão foi segmentada de acordo com os diferentes tipos de usuários do sistema.

4.2.1 PEDIATRAS

Tendo em vista o usuário final sendo um médico pediatra, o aplicativo pode facilitar a comunicação entre o médico e os pais, além de possibilitar o acompanhamento em tempo real do bebê pelo médico.

O médico pode configurar os alarmes de acordo com a necessidade de cada bebê. Caso ele esteja acompanhando um bebê prematuro, por exemplo, é possível restringir o intervalo dos alarmes para que qualquer medição, mesmo que apenas um pouco abaixo ou acima do normal, seja detectada e enviada por *e-mail* ou SMS.

O gráfico contendo os históricos das medições, associado com as últimas medições, podem auxiliar em casos em que o pediatra cuida de muitos bebês ao mesmo tempo. Pelo aplicativo, ele tem acesso ao histórico de medições de cada bebê, em forma de gráfico ou valores, tendo a possibilidade de fazer acompanhamentos com uma periodicidade flexível, podendo dar mais atenção a bebês que necessitem de acompanhamentos especiais.

4.2.2 PAIS

Visando os pais como usuário final, o aplicativo impacta principalmente na segurança e alívio. Naturalmente, os pais do bebê se sentiriam mais seguros e confortáveis sabendo que existe um médico acompanhando o bebê a todo momento.

A lista com os pediatras e o mapa com os hospitais mais próximos gera uma tranquilidade ainda maior, pois caso haja alguma emergência, o aplicativo já possui estas ferramentas para tomar algum tipo de ação imediata. Também é possível que os pais acompanhem o diagnóstico disponibilizado pelo pediatra, caso sintam necessidade de receber notícias diretamente do médico.

4.2.3 HOSPITAIS

No caso de hospitais, o uso das placas para medição de grandezas do ambiente é de extrema importância. Tomando como exemplo o cenário de um berçário, o ideal é que, além do

sensoriamento dos bebês, haja um sistema que monitore o ambiente em que os bebês se encontram. Desta forma, é possível agrupar as medições de acordo com os bebês que estão dentro do mesmo berçário, trazendo mais eficiência em um monitoramento em massa.

As soluções e ferramentas utilizadas pelo hospital são similares aos dos pediatras: alarmes na ocorrência de medições fora dos limites, gráficos e diagnósticos.

4.3 CUSTOS E MERCADOS

Em algumas lojas dos Estados Unidos é possível a compra do equipamento Smart Sock 2 Baby Oxygen Level and Heart Rate Monitor (OWLET, 2019), no valor de 299 dólares americanos. Este equipamento monitora os batimentos cardíacos e a oximetria do bebê através de sensores colocados na meia do bebê, os sensores se comunicam com o celular que possui um aplicativo para visualização do monitoramento. Equipamento mostrado na Figura 38.

Figura 38 - Equipamento de monitoramento



Fonte: Os Autores, 2019.

Este aplicativo possui a função de receber uma imagem em tempo real do bebê através de uma câmera vendida separadamente.

No Brasil é possível encontrar o *Baby Hug*, que diz ser “o primeiro *wearable* para bebês do Brasil” (BABYHUG,2019) Este promete fornecer os mesmos dados visto até então nesse trabalho, entretanto a comunicação dos sensores à Internet é através de um celular com comunicação Bluetooth Esta comunicação se limita a um raio de 8 metros. Não se sabe valores e os sites de compras sugeridos pelo fabricante não comercializam o dispositivo.

Tratando-se do projeto realizado nesse trabalho, foi feito o levantamento de gastos para a fabricação de uma unidade de monitoramento, sendo eles listados a seguir:

- a) Módulo de rádio frequência LoRaWAN: R\$ 90,00
- b) Sensores: R\$ 60,00
- c) Boneco: R\$ 35,00
- d) Macacão: R\$ 20,00
- e) Componentes eletrônicos, microcontrolador (PIC): R\$ 100,00
- f) *Software open-source*, desenvolvido durante o trabalho.

O valor inicial para um dispositivo seria de aproximadamente R\$ 300,00. Para produção em larga escala, todos estes valores informados seriam revistos, possibilitando um barateamento, sendo necessário aplicar o desenvolvimento e manutenção do aplicativo.

5 CONCLUSÃO

5.1 CONSIDERAÇÕES FINAIS

A preocupação no cuidado das pessoas tem se mostrado cada vez mais promissor e importante. O objetivo foi desenvolver um sistema para o monitoramento do bem-estar de recém-nascidos. O cuidado e a atenção que os bebês merecem durante um curto período de sua vida podem ser essenciais para o crescimento.

A engenharia e a medicina se encontraram para proporcionar melhorias no estilo de vida atual e o fruto de estudos durante o trabalho comprovaram que há uma ajuda mútua. Foi possibilitado o monitoramento de bebês através de métodos não invasivos com uma grande capacidade de tornar ações em padrões comportamentais fornecidos pelos dados recebidos através do dispositivo desenvolvido. Estes dados então podem ser analisados por pessoas capazes de distinguir o verdadeiro bem-estar de um recém-nascido.

A praticidade e a tecnologia disponíveis foram capazes de acoplar, em um protótipo, diversos sensores. Com estas peças presas na roupa do bebê, através de botões de pressão que seguem um padrão de fixação, o intuito foi não ficar restrito a um tipo de roupa.

Foi possível, em tempo real, analisar, agir e tomar decisões instantâneas de acordo com as informações passadas ao aplicativo desenvolvido, diminuindo a falta de comunicação e a distância entre os pediatras e os recém-nascidos, trazendo um maior conforto e segurança aos pais que utilizam o dispositivo.

5.2 TRABALHOS FUTUROS

Para trabalhos futuros foi possível sugerir ações em diversas áreas e cada uma delas com seu campo de ação e desenvolvimento. Abaixo foram citadas algumas das possíveis iniciativas

O aperfeiçoamento do dispositivo de *hardware* para tornar o sistema mais específico para a aplicação desejada, tornando-o mais barato e mais compacto.

O monitoramento de recém-nascidos, pode ser aplicável no cuidado de idosos, com a possibilidade de concentrar todas as informações de saúde do idoso no próprio aplicativo desenvolvido, além de criar alarmes para que o paciente tome seus remédios no horário certo, saiba seu prontuário médico, agendamentos de exames, datas e lembretes importantes. Registro de vacinas e até acesso rápido e direto com o plano de saúde ou seus médicos de confiança.

Por fim, o cuidado dos *pets* pode ser levado em conta, visto que no âmbito social diversos casais têm preferido ter animais de estimação. O cuidado com o *pet* pode se tornar possível implementando também um sensoriamento na coleira do animal para localização quando perdido, e como dito anteriormente, concentrar os dados médicos no aplicativo.

REFERÊNCIAS

NOGALES,A. **Evolução da mortalidade na infância nos últimos 10 anos (2007 a 2016)**. Arquivos Ministérios da Saúde, 2018. Disponível em <https://portalarquivos2.saude.gov.br/images/pdf/2018/setembro/13/Oficina-mortalidade-materna-e-infantil-CIT-MESA-Ana-Nogales.pdf>.

SIM. **Porcentagem das causas de óbitos evitáveis**. Sistema de Informação sobre Mortalidade, 2018. Disponível em sim.saude.gov.br

SIM. **Taxa de mortalidade de acordo com a idade**. Sistema de Informação sobre Mortalidade, 2018. Disponível em sim.saude.gov.br

CHEN, Hongyu; XUE, Mengru; MEI, Zhennig; OETOMO, Sirdarto; CHEN, Wei. **A Review of Wearable Sensor Systems for Monitoring Body Movements of Neonates**. Ncbi, 2016. Disponível em <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5191114/>.

LEE, E. **Baby by the Numbers**. IEEE Spectrum, Piscataway, p. 24, 2015. Disponível em <https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=7115553>.

IANNELLI, Vincent. **Normal Heart Rate for Children**. Verywellfamily, 2019. Disponível em <https://www.verywellfamily.com/normal-pulse-rates-for-kids-2634038>.

AGRAWAL, Susan. **Normal Vital Signs in Children: Heart Rate, Respirations, Temperature, and Blood Pressure**. ComplexChild, 2009. Disponível em <http://www.articles.complexchild.com/march2009/00114.pdf>.

Maxim Integrated Products, Inc. **Pulse Oximeter and Heart-Rate Sensor IC for Wearable Health**. Maxim Integrated Products, 2014. Disponível em <https://datasheets.maximintegrated.com/en/ds/MAX30100.pdf>.

STROGONOV, Raivis. **Implementing pulse oximeter using MAX30100**. Microcontrolles, 2017. Disponível em <https://morf.lv/implementing-pulse-oximeter-using-max30100>.

Ali, Zahid. **Introduction to MAX30100**. The Engineering Projects, 2019. Disponível em <https://www.theengineeringprojects.com/2019/05/introduction-to-max30100.html>.

Teach Me Micro. **How to Use the MAX30100 as Arduino Heart Rate Sensor**. Microcontroller Tutorials, 2018. Disponível em <https://www.teachmemicro.com/max30100-arduino-heart-rate-sensor>.

Mikro Electron. **MAX30100 Pulse Oximeter Heart-Rate Sensor Module**. Mikro Electron, 2019. Disponível em <https://mikroelectron.com/Product/MAX30100-Pulse-Oximeter-Heart-Rate-Sensor-Module/>.

Components101. MPU6050 - **Accelerometer and Gyroscope Module**. Components101, 2018. Disponível em <https://components101.com/sensors/mpu6050-module>.

Smart Campus Mauá. **Smart Campus Mauá Dashboard**. Multiplataforma em rede LPWAN voltada para aplicações em Internet das Coisas (IoT – Internet of Things), 2019. Disponível em <https://smartcampus.maua.br/dash/>

Google Play. **Aplicativo Baby Iot**. Google, 2019. Disponível em https://play.google.com/store?hl=pt_BR

Owlet. **Know Your Baby is Okay**. Owlet, 2019. Disponível em: <https://owletcare.com/>.

Babyhug. **A tranquilidade de estar sempre conectado ao seu bebê**. Baby Hug, 2019 Disponível em <http://www.babyhug.com.br/>.

ANEXO A – CÓDIGO DO MAX30100 EM ARDUINO

```

#include <Wire.h>
#include "MAX30100.h"
MAX30100* pulseOxymeter;
int bpm[10];
int oxy[10];
int sumbpm = 0;
int sumoxy = 0;
float mediabpm = 0;
float mediaoxy = 0;
int i = 0;
int counter = 0;
void setup()
{
  Wire.begin();
  Serial.begin(115200);
  Serial.println("Pulse oxymeter test!");
  pulseOxymeter = new MAX30100(); //pulseOxymeter = new MAX30100(
DEFAULT_OPERATING_MODE, DEFAULT_SAMPLING_RATE, DEFAULT_LED_PULSE_WIDTH,
DEFAULT_IR_LED_CURRENT, true, true );
  pinMode(2, OUTPUT);
}
void loop()
{
  delay( 27 ); //You have to call update with frequency at least 37Hz.
But the closer you call it to 100Hz the better, the filter will work.
  pulseoxymeter_t result = pulseOxymeter->update();
  if(result.pulseDetected == true )
  {
    Serial.println( result.heartBPM );
    if ( result.heartBPM < 220 && result.heartBPM > 10 )
    {
      bpm[i] = result.heartBPM;
      oxy[i++] = result.SaO2;
    }
  }
  if (i >= 10)
  {
    i = 0;
    sumbpm = 0;
    sumoxy = 0;
    for (counter = 0; counter < 10; counter++)
    {
      sumbpm += bpm[counter];
    }
    mediabpm = sumbpm / 10;
    mediaoxy = sumoxy / 10;
    Serial.print( "BPM:" );
    Serial.print( mediabpm );
    Serial.print( " SaO2:" );
    Serial.print( mediaoxy );
    Serial.println();
  }
}

```

ANEXO B – CÓDIGO DO MPU6050 EM ARDUINO

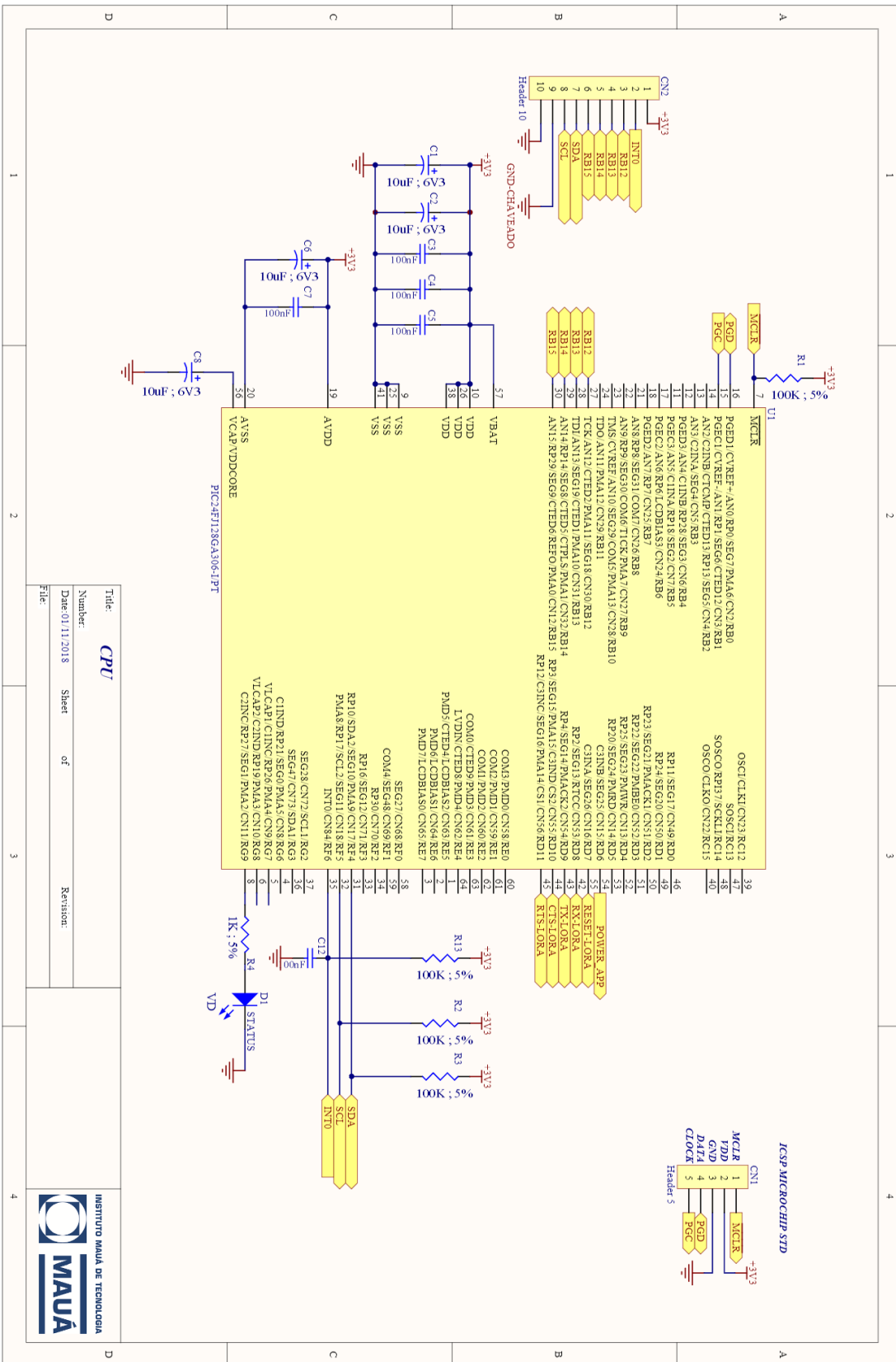
```

#include "TimerOne.h"
#include<Wire.h>
#include <stdio.h>
const int MPU=0x68;
int AcX,AcY,AcZ,Tmp,GyX,GyY,GyZ;
int timer;
int vAcX[2];
int vAcY[2];
int vAcZ[2];
int difx = 0;
int dify = 0;
int difz = 0;
int i = 0;
int counter = 0;
void setup()
{
Wire.begin();
Timer1.initialize(1000000); // Inicializa o Timer1 e configura para um
período de 1 segundos
Timer1.attachInterrupt(callback, 1000000); // Configura a função callback()
como a função para ser chamada a cada interrupção do Timer1
Serial.begin(115200);
Wire.begin();
Wire.beginTransmission(MPU);
Wire.write(0x6B);
//Inicializa o MPU-6050
Wire.write(0);
Wire.endTransmission(true);
pinMode(2, OUTPUT);
}
void callback()
{
Serial.println("medindo...");
AcX=Wire.read()<<8|Wire.read(); //0x3B (ACCEL_XOUT_H) & 0x3C
(ACCEL_XOUT_L)
AcY=Wire.read()<<8|Wire.read(); //0x3D (ACCEL_YOUT_H) & 0x3E
(ACCEL_YOUT_L)
AcZ=Wire.read()<<8|Wire.read(); //0x3F (ACCEL_ZOUT_H) & 0x40
(ACCEL_ZOUT_L)
Tmp=Wire.read()<<8|Wire.read(); //0x41 (TEMP_OUT_H) & 0x42 (TEMP_OUT_L)
GyX=Wire.read()<<8|Wire.read(); //0x43 (GYRO_XOUT_H) & 0x44 (GYRO_XOUT_L)
GyY=Wire.read()<<8|Wire.read(); //0x45 (GYRO_YOUT_H) & 0x46 (GYRO_YOUT_L)
GyZ=Wire.read()<<8|Wire.read(); //0x47 (GYRO_ZOUT_H) & 0x48 (GYRO_ZOUT_L)
vAcX[i]=AcX;
vAcY[i]=AcY;
vAcZ[i]=AcZ;
if(i==1)
{
difx = vAcX[0] - vAcX[1];
dify = vAcY[0] - vAcY[1];
difz = vAcZ[0] - vAcZ[1];
}
}

```

```
    delay(500);
    if(abs(difx)>=3000 || abs(dify)>=3000 || abs(difz)>=3000){
        Serial.println("Vou acordar");
    }
    delay(500);
    difx=0;
    dify=0;
    difz=0;
    i=0;
    i--;
    }
    i++;
}
void loop()
{
    Wire.beginTransaction(MPU);
    Wire.write(0x3B); // starting with register 0x3B (ACCEL_XOUT_H)
    Wire.endTransmission(false);
    Wire.requestFrom(MPU,14,true);
}
```

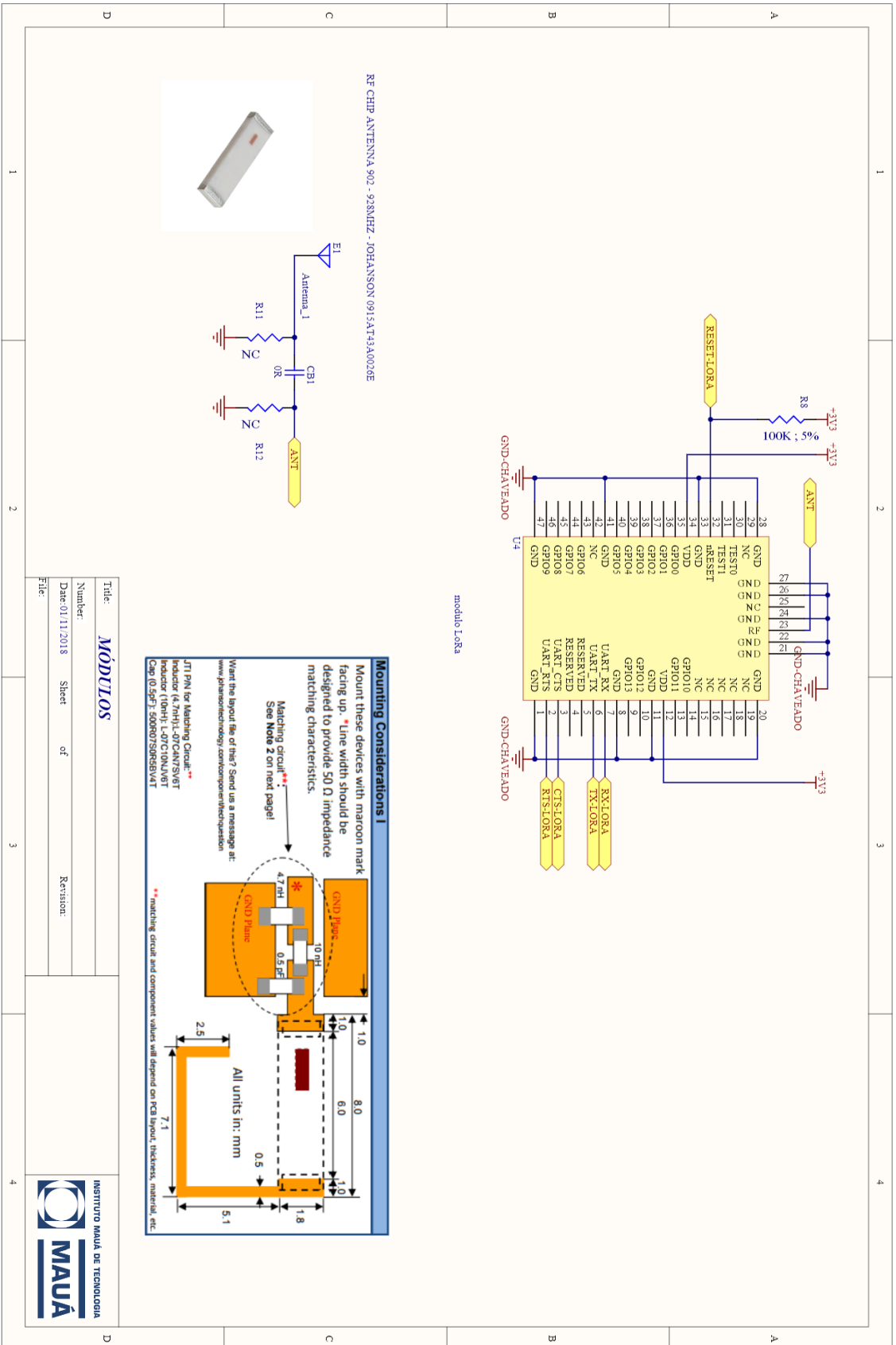
ANEXO C – ESQUEMA ELÉTRICO PIC



Title:	CPU		
Number:	Sheet	of	Revision
Date: 01/11/2018			
File:			



ANEXO D – ESQUEMA ELÉTRICO LORA



Title:	MÓDULOS		
Number:	Sheet	of	Revision:
Date: 01/11/2018			
File:			



ANEXO E – PROGRAMAÇÃO DO APLICATIVO

```

import React, {Component} from 'react';
import {Text, View, Button, StyleSheet, TextInput, TouchableOpacity, Image,
KeyboardAvoidingView, Modal} from 'react-native';
export default class Login extends Component{
  constructor(props) { ...
  }
  login() //chama rota da API para validar usuario ...
  }
  render() {
    return(
      <View>
        <Modal
          animationType="slide"
          transparent={false}
          visible={this.state.modalVisible}
        >
          <View>
            <Image
              source={require(loading.gif)}
            >
            </Image>
            <Text>
              Loading...
            </Text>
          </View>
        </Modal>
        <View>
          <View>
            <Image
              source={require(bebe.png)}
            >
            </Image>
            <Image
              source={require(heart.png)}
            >
            </Image>
          </View>
          <Text>
            Baby IoT
          </Text>
          <Text>
            Monitoramento de recém-nascidos
          </Text>
          <Text>
            Versão Beta
          </Text>
        </View>
        <KeyboardAvoidingView
          behavior = "padding"
        >
          <TextInput

```



```
        placeholder="Usuário"
        autoCorrect={false}
        value={this.state.username}
    />
    <TextInput
        placeholder="Senha"
        autoCorrect={false}
        secureTextEntry={true}
        value={this.state.password}
    />
    <TouchableOpacity
        onPress={this.login.bind(this)}>
        <Text>Login</Text>
    </TouchableOpacity>
    </KeyboardAvoidingView>
</View>
)
}
}
```

ANEXO F – FLUXOGRAMA DO APLICATIVO

