

HANDY

DOCUMENTAÇÃO

v 2020.09.20

Matheus Tomazella
Luiz Felipe Costa Soares
Eduardo Gomes Heleno

SUMÁRIO

1 TERMOS DE ABERTURA.....	4
1.1 JUSTIFICATIVA DO PROJETO.....	4
1.2 DESCRIÇÃO DO PROJETO.....	4
1.3 GERENTE DO PROJETO.....	4
1.3.1 Nível de Autoridade.....	4
1.4 PATROCINADOR.....	5
1.5 PARTES INTERESSADAS.....	5
1.6 RESTRIÇÕES.....	5
1.7 PREMISSAS.....	5
1.8 AUTORIZAÇÃO DO PROJETO.....	5
2 DETALHAMENTO DE REQUISITOS.....	6
2.1 ENGENHARIA DA PRÓTESE.....	6
2.1.1 Funcionamento da prótese.....	6
2.1.2 Projetos.....	7
2.1.3 Imagens Renderizadas.....	12
2.1.4 Funcionamento do Circuito.....	13
2.1.5 Projetos do Circuito.....	18
2.1.7 Requisitos Funcionais.....	22
2.2 SOFTWARE WEB.....	22
2.2.1 Requisitos Funcionais.....	22
2.2.2 Requisitos Não Funcionais.....	22
2.3 SOFTWARE DESKTOP.....	23
2.3.1 Requisitos Funcionais.....	23
2.3.2 Requisitos Não Funcionais.....	23
2.4 SOFTWARE MOBILE.....	24
2.4.1 Requisitos Funcionais.....	24
2.4.2 Requisitos Não Funcionais.....	24
2.5 BANCO DE DADOS.....	25
2.5.1 Requisitos Funcionais.....	25
2.5.2 Requisitos Não Funcionais.....	25
2.5.3 Diagramas.....	26
3 CASOS DE USO.....	27
3.1 PINUS (API).....	27
3.1.1 Cadastrar Usuário / Admin.....	28
3.1.2 Buscar Usuário / Admin.....	28
3.1.3 Alterar Usuário / Admin.....	29
3.1.4 Excluir Usuário / Admin.....	30
3.1.5 Enviar Mensagem.....	31
3.1.6 Buscar Mensagem.....	31
3.1.7 Adicionar Contato.....	32
3.1.8 Buscar Contato.....	32
3.1.9 Excluir Contato.....	33
4 DEFINIÇÃO DE RECURSOS.....	34
4.1 STAKEHOLDERS.....	34

4.2 LISTA DE RECURSOS DE CONSTRUÇÃO.....	34
4.2.1 Ferramentas.....	34
4.2.2 Recursos Construção Mecânica.....	34
4.2.3 Recursos Construção Eletrônica.....	35
5 OBSERVAÇÕES, MEDIÇÕES E CÁLCULOS.....	36
5.1 ENERGIA.....	36
5.1.1 Capacidade das Baterias.....	36
5.1.2 Consumo de Energia.....	36

1 TERMOS DE ABERTURA

1.1 JUSTIFICATIVA DO PROJETO

Muitas vezes prótese são muito caras, levando pessoas com membros amputados a não terem condições de pagar por uma. O objetivo do projeto é criar uma solução barata para o cliente, a fim de que possa melhorar sua qualidade de vida até conseguir uma prótese mais tecnológica.

1.2 DESCRIÇÃO DO PROJETO

A ideia principal dessa prótese é a acessibilidade, possibilitando que mais camadas da sociedade tenham condições de possuir uma prótese para seus membros amputados. O produto é pensado para impressão em impressoras 3d, porém pode ser construído segundo a vontade do cliente, e deve ser controlado por movimentos de outras partes do corpo.

Uma página web será criada para compartilhar informações sobre o produto e permitir que interessados se comuniquem com os criadores.

O produto possuirá um software para desktop para acessar os dados enviados por interessados no projeto por meio do site e um software mobile conectado aos eletrônicos da prótese para movimentos e ações mais precisas.

1.3 GERENTE DO PROJETO

MATHEUS TOMAZELLA

1.3.1 Nível de Autoridade

O gerente de projeto deve ter autoridade para coordenar recursos financeiros e humanos disponibilizados, a fim de concluir o projeto.

1.4 PATROCINADOR

Grupo responsável pelo projeto.

1.5 PARTES INTERESSADAS

1. Cliente;
2. Projetista;
3. Designer;
4. Programador.

1.6 RESTRIÇÕES

1. Não existe um orçamento máximo fixo para o projeto, porém deve usar o mínimo possível de recursos financeiros;
2. O projeto deve possuir uma versão totalmente funcional até o final do ano de 2021.

1.7 PREMISSAS

1. Deve priorizar um produto final barato;
2. O produto deve poder ser construído pelo cliente.

1.8 AUTORIZAÇÃO DO PROJETO

Data: 01 de Setembro de 2020

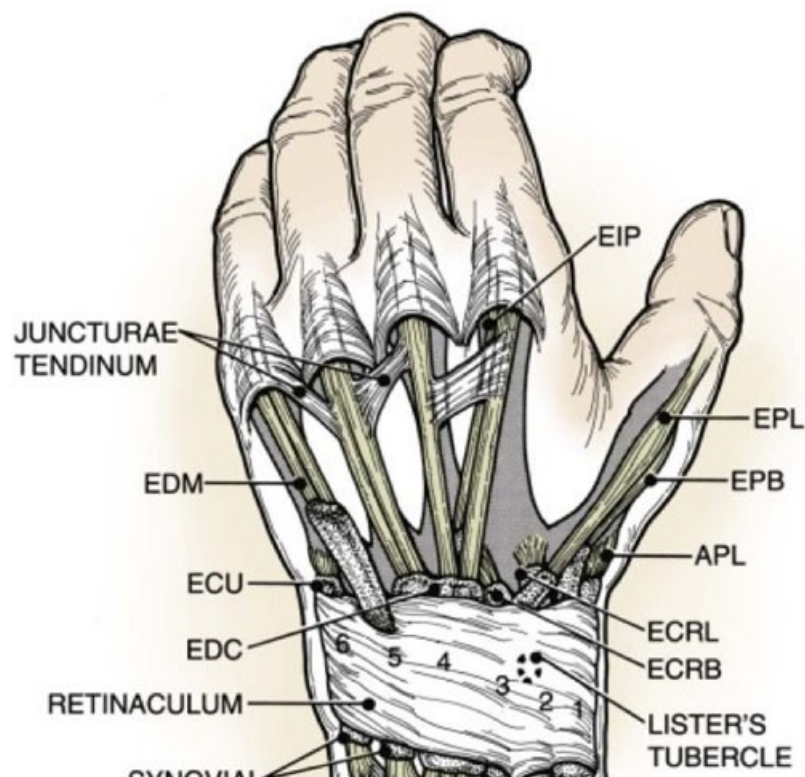
Matheus Tomazella

2 DETALHAMENTO DE REQUISITOS

2.1 ENGENHARIA DA PRÓTESE

2.1.1 Funcionamento da prótese

O movimento da prótese é baseado em fios que correm entre os dedos, ligando a parte mais extrema de cada dedo com seu correspondente atuador. Uma vez que o atuador puxa o fio, a ponta do dedo será puxada e o levará a se dobrar contra a palma, assim como uma mão biológica.

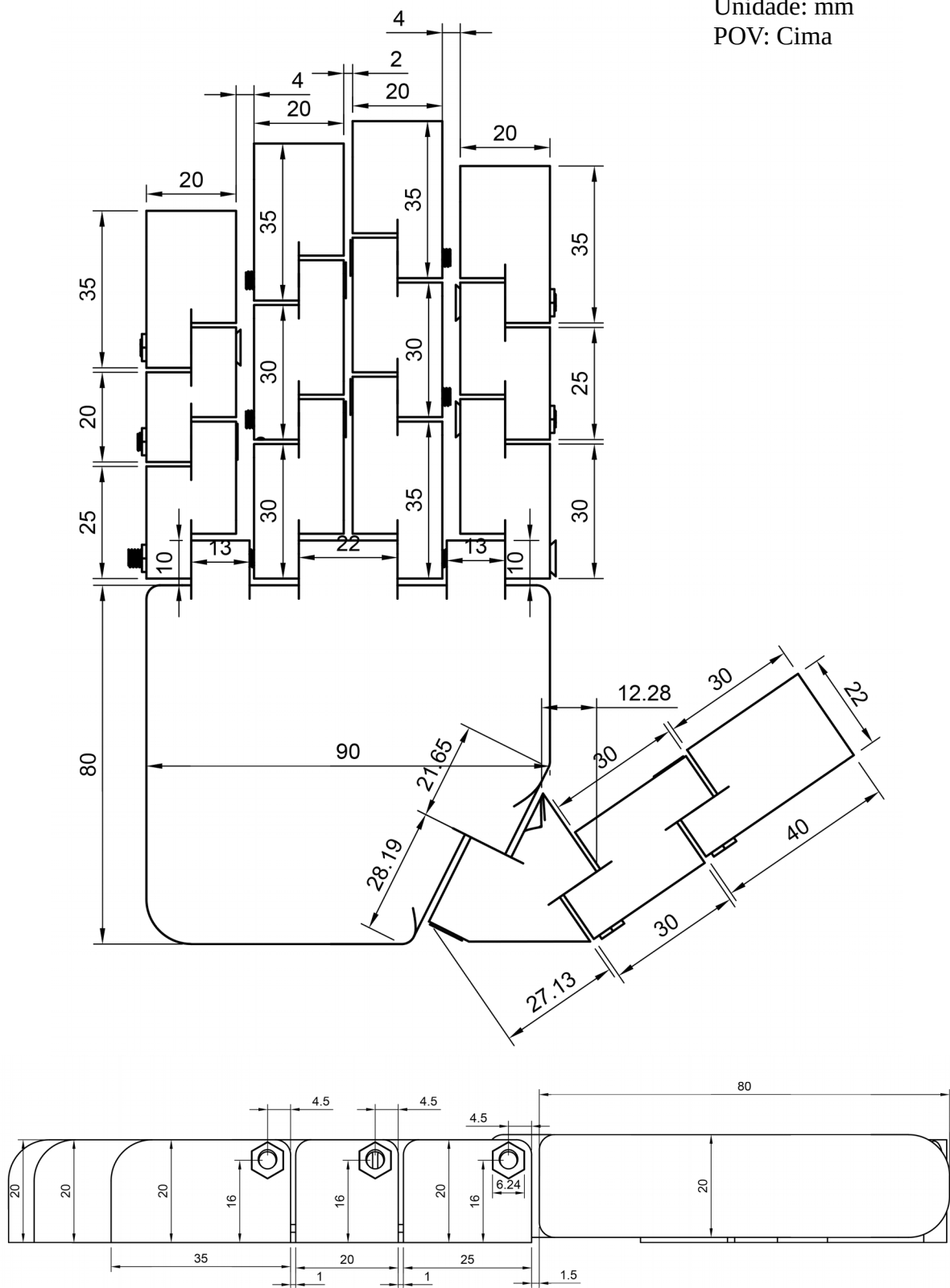


Fonte: <http://feliperoth.com.br/traumatologia-da-mao/lesao-de-tendao/#>

Na parte contrária a palma da mão, um material elástico será responsável por mover os dedos para suas posições originais assim que a força aplicada pelos atuadores parar de agir.

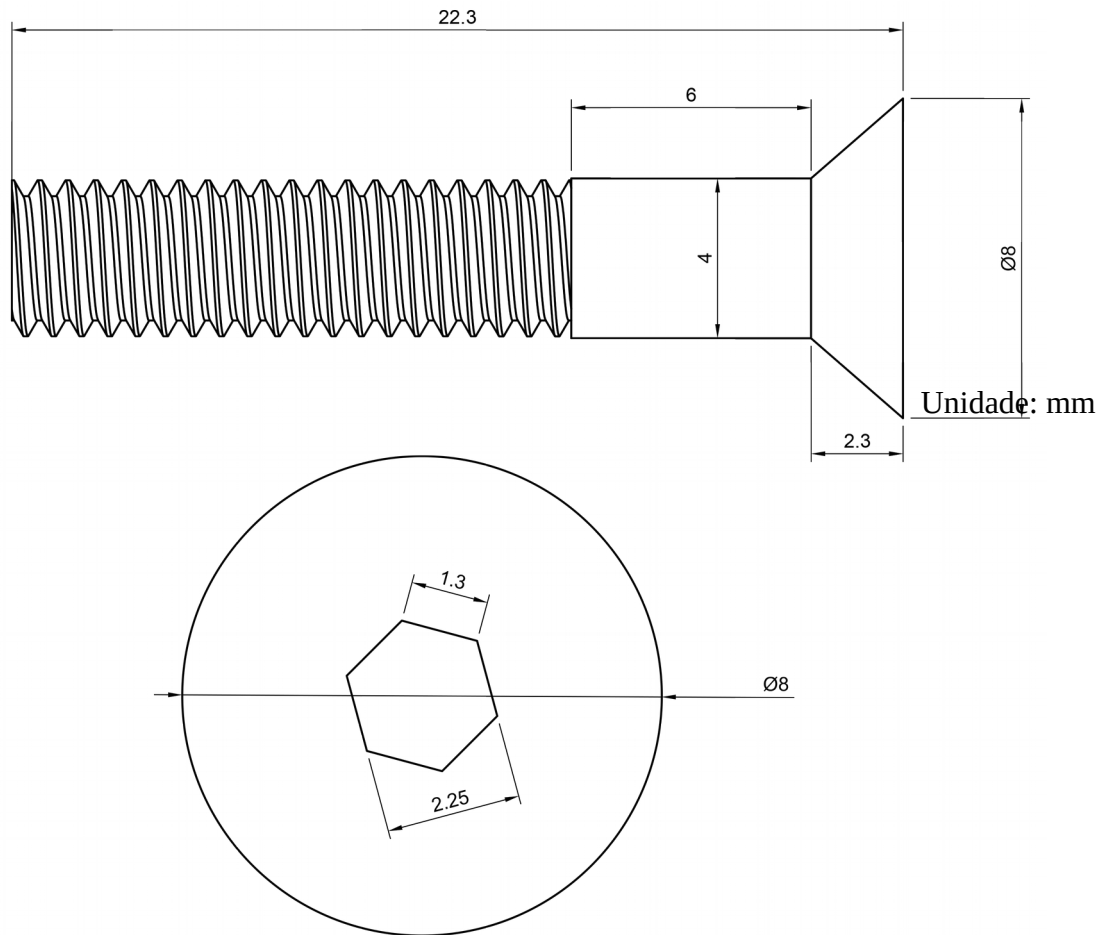
2.1.2 Projetos

Unidade: mm
POV: Cima



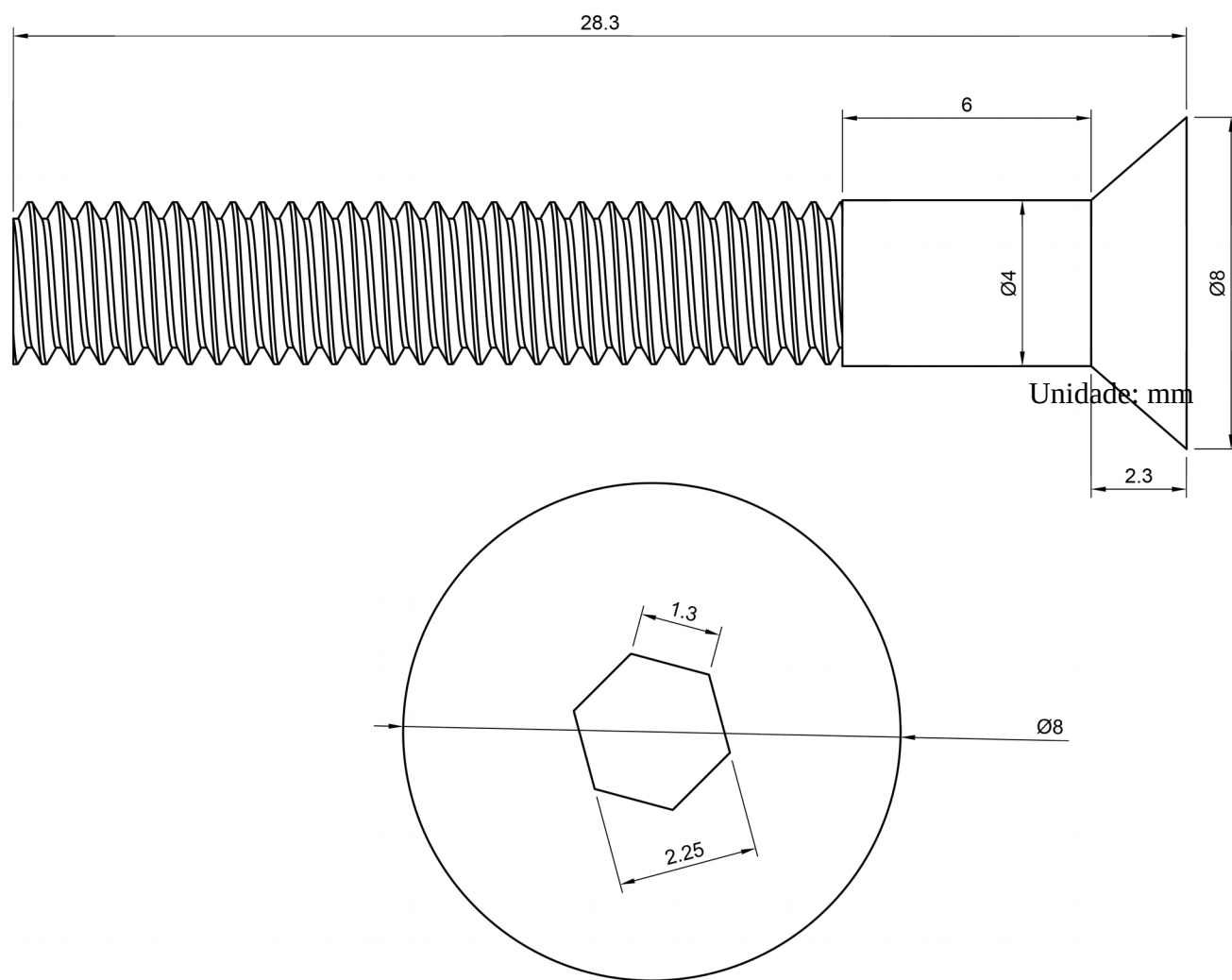
Parafuso ideal para juntas entre as falanges

Parafuso Allen cabeça chata M4 de 20mm de comprimento



Parafuso ideal para a junta entre metacarpo e falanges do dedo polegar e a junta entre metacarpo e carpo do dedo polegar

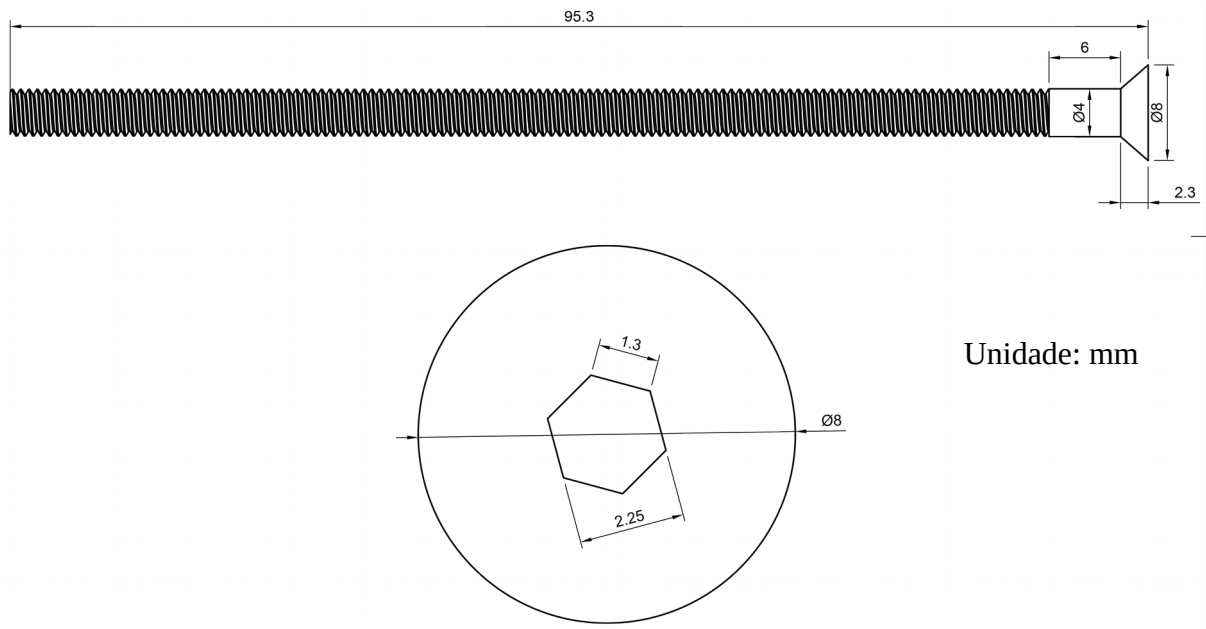
Parafuso Allen cabeça chata M4 de 90mm de comprimento.



Parafuso ideal para as juntas entre metacarpo e falanges

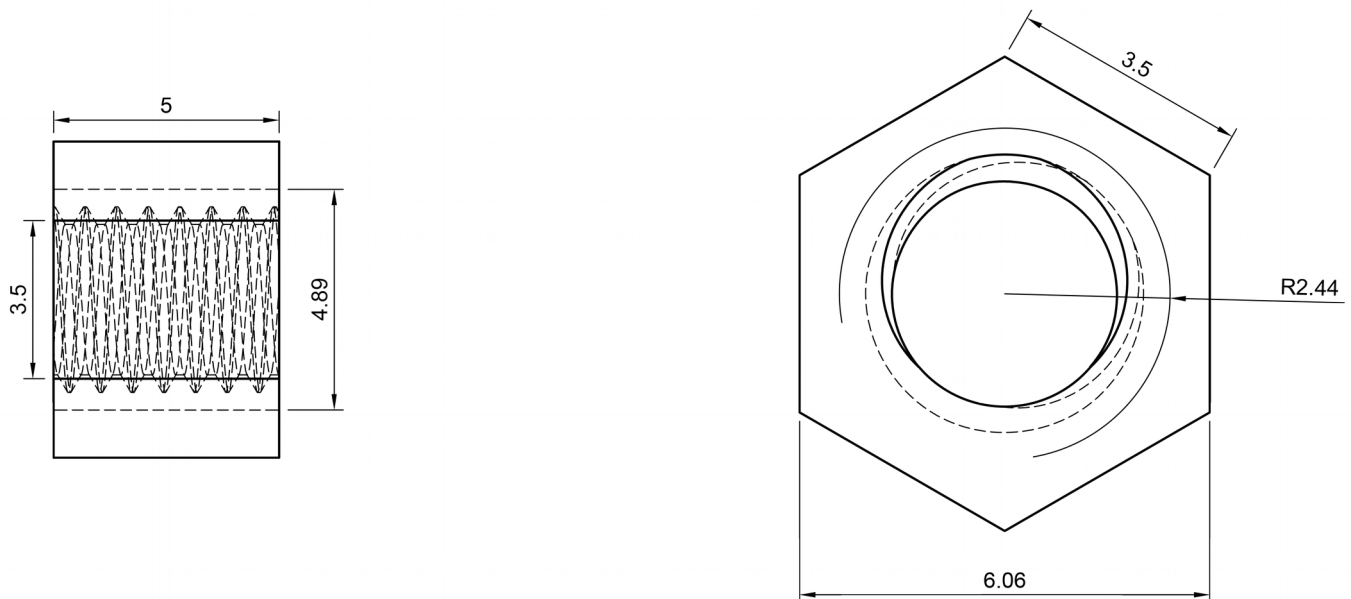
Parafuso Allen cabeça chata M4 de 90mm de comprimento.

Diferente das demais juntas, as juntas entre metacarpo e falanges compartilham o mesmo parafuso.

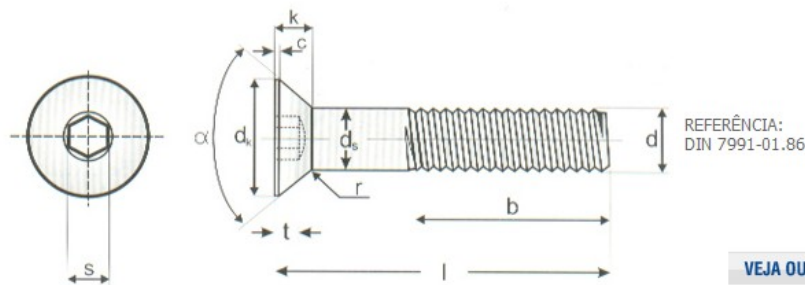


Porca ideal para todas as juntas

Porca Parlock M4 – Projeto simplificado



Tabelas relativas aos parafusos e porcas



[VEJA OUTROS MODELOS](#) >>

[VEJA OUTROS PRODUTOS](#) >>

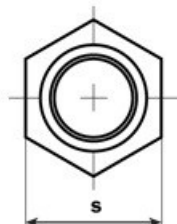
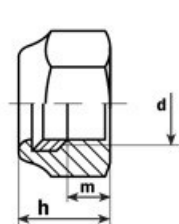
Diâmetro	ds		d		k	r	s		t		c	b		
Nominal	máx.	min.	máx.	min.	máx.	min.	máx.	min.	máx.	min.	~	1	2	3
M3	3,00	2,86	6,00	5,70	1,7	0,1	2,10	2,02	1,20	0,95	0,2	12	---	---
M4	4,00	3,82	8,00	7,64	2,3	0,2	2,60	2,52	1,80	1,55	0,3	14	---	---
M5	5,00	4,82	10,00	9,64	2,8	0,2	3,10	3,02	2,30	2,05	0,3	16	---	---
M6	6,00	5,82	12,00	11,57	3,3	0,3	4,12	4,02	2,50	2,25	0,3	18	24	---
M8	8,00	7,78	16,00	15,56	4,4	0,5	5,14	5,02	3,50	3,20	0,4	22	28	---
M10	10,00	9,78	20,00	19,48	5,5	0,5	6,14	6,02	4,40	4,10	0,5	26	32	45
M12	12,00	11,73	24,00	23,48	6,5	1,0	8,175	8,025	4,60	4,30	0,5	30	36	49
M14	14,00	13,73	27,00	26,48	7,0	1,0	10,175	10,025	4,80	4,50	0,5	34	40	53
M16	16,00	15,73	30,00	29,48	7,5	1,0	10,175	10,025	5,30	5,00	0,5	38	44	57
M18	18,00	17,73	33,00	32,38	8,0	1,0	12,212	12,032	5,50	5,20	0,5	42	48	61
M20	20,00	19,67	36,00	35,38	8,5	1,0	12,212	12,032	5,90	5,60	0,5	46	52	65
M22	22,00	21,67	36,00	35,38	13,1	1,0	14,212	14,032	8,80	8,44	1,0	50	56	69
M24	24,00	23,67	39,00	38,38	14,0	1,6	14,212	14,032	10,30	9,87	1,0	54	60	73

OBS: 1-) Comprimentos de rosca: b1 para $l < \text{ou} = 125\text{mm}$, b2 para $l > 125\text{mm}$ e $l < \text{ou} = 200\text{mm}$ e b3 para $l > 200\text{mm}$

2-) Tolerância para comprimento "l": js 15, vide tabela de ajustes - ISO.

ISO - EN -

DIN 985

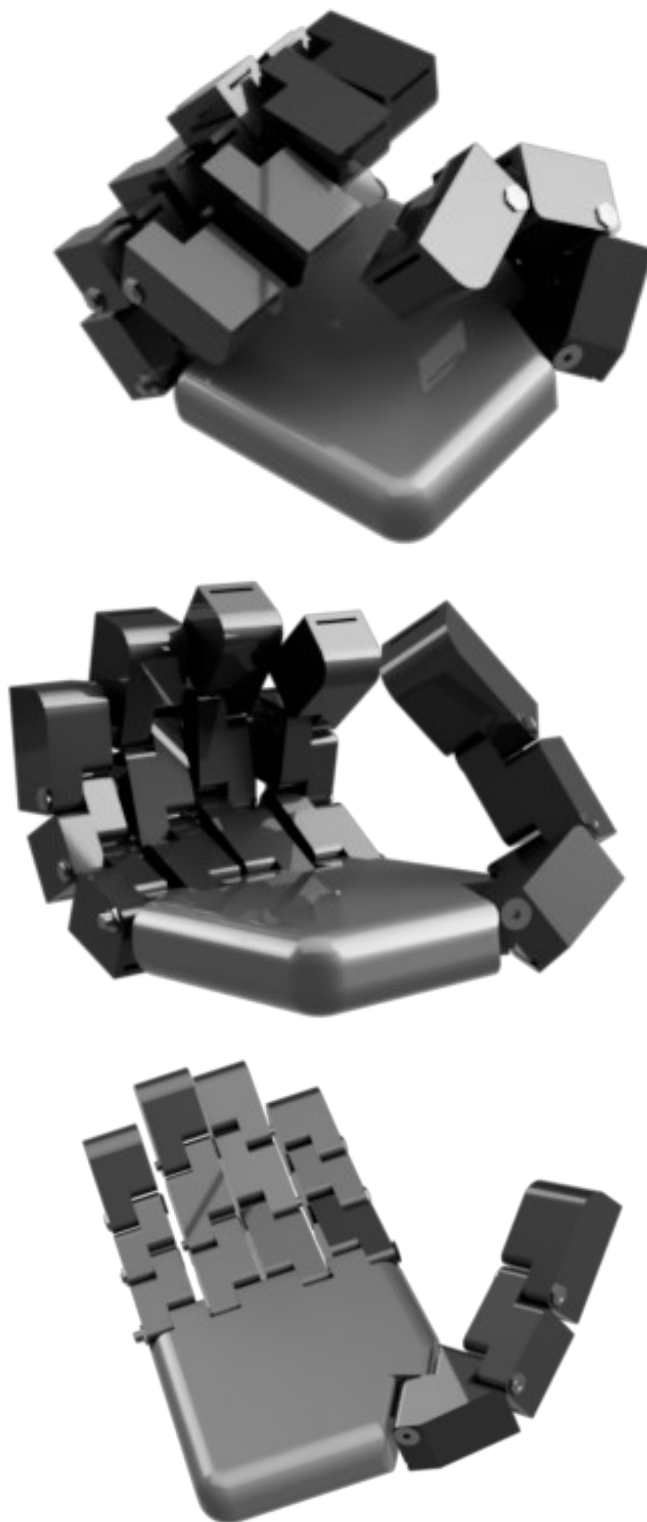


d	M3	M4	M5	M6	M7	M8	M10	M12	M14	M16	M18
p	0,5	0,7	0,8	1	1	1,25	1,5	1,75	2	2	2,5
s	5,5	7	8	10	11	13	17/16*	19/18*	22/21*	24	27
h	4	5	5	6	7,5	8	10	12	14	16	18,5

Peso/Weight 1000 ud. kg

0,500	1,000	1,400	2,400	3,000	5,100	10,600	17,200	26,00	34,00	45,00
-------	-------	-------	-------	-------	-------	--------	--------	-------	-------	-------

2.1.3 Imagens Renderizadas



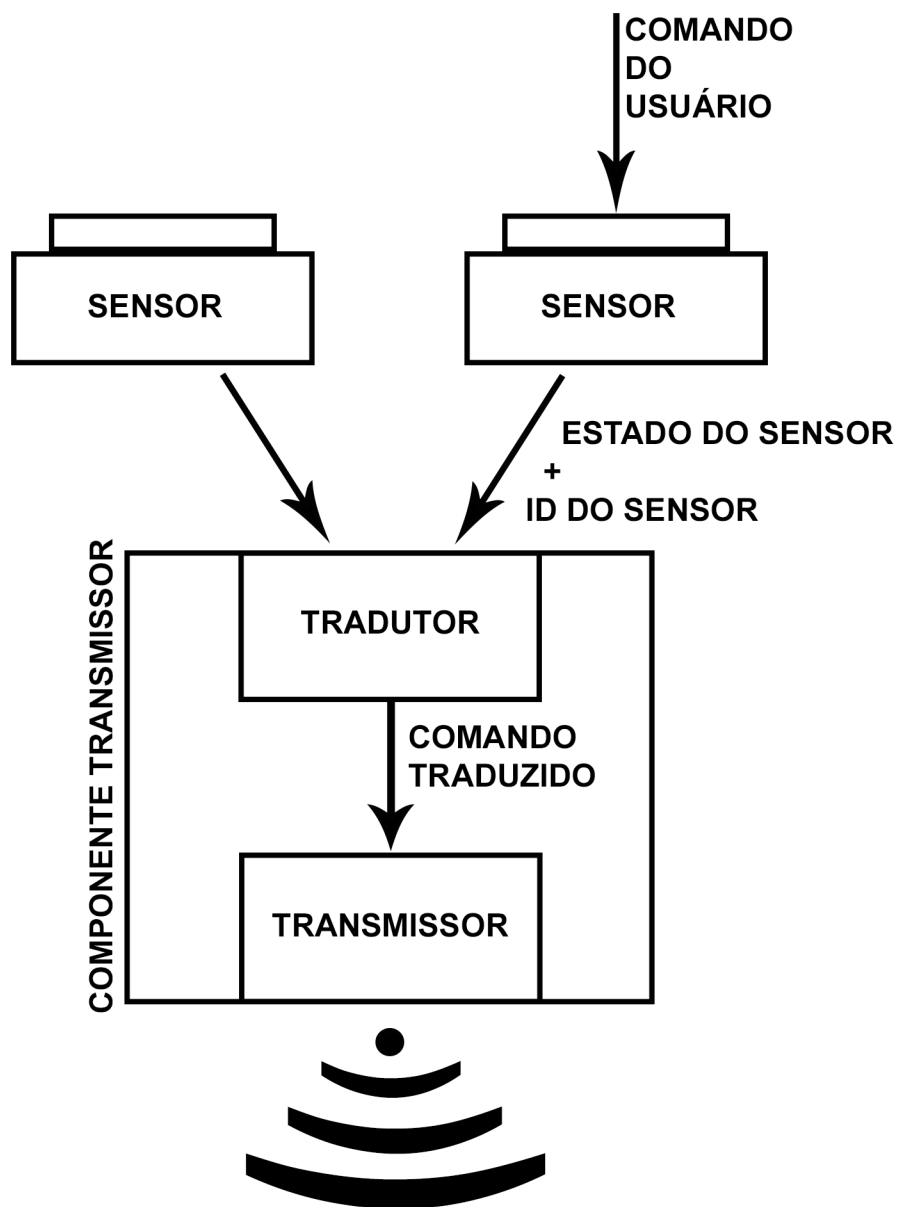
2.1.4 Funcionamento do Circuito

Conjunto de Controle

O Conjunto de Controle tem como função enviar informações do usuário para a prótese, a fim de permitir seu controle. É composto de sensores, que recebem informações, e um componente que as transmite para o atuador.

Tais sensores devem ser escolhidos conforme a parte do corpo que controlará a prótese. A versão sugerida pelo time de desenvolvimento tem como sensores componentes próprios, pensados para perceber a contração muscular de forma barata.

O transmissor será um módulo de transmissão rádio nRF24 combinado com uma placa de microcontrolador pronta, caso o local dos sensores seja longe do braço, mas poderá ser substituído por fios que correm até o conjunto atuador.



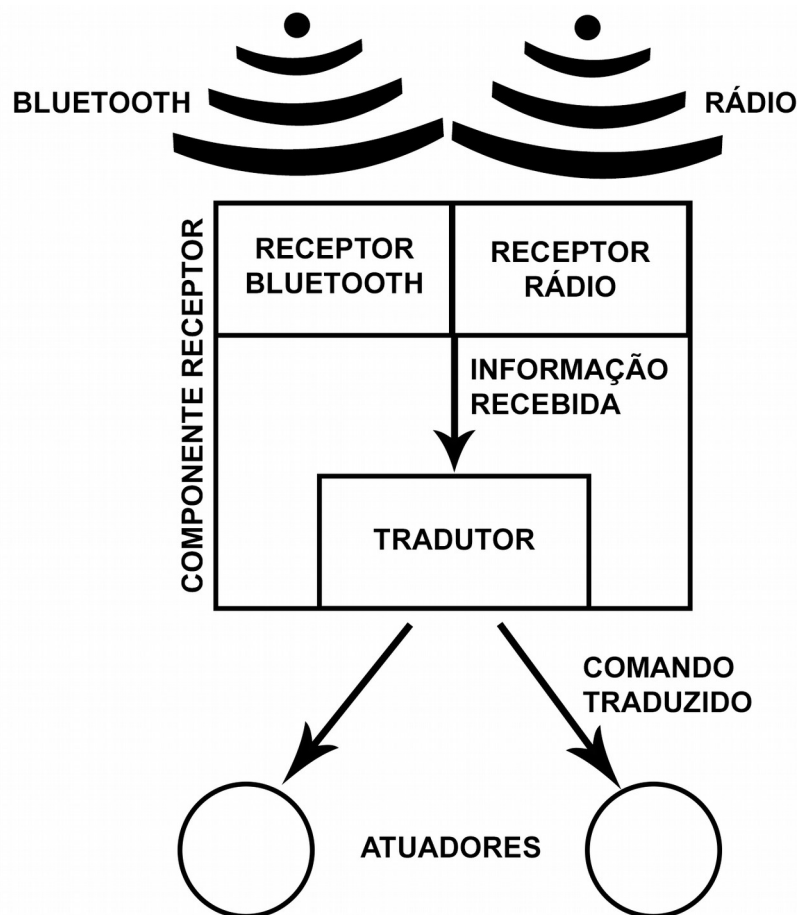
Conjunto Atuador

O Conjunto Atuador tem como função receber as informações do Conjunto de Controle e converter as instruções em ações. O conjunto composto de um componente receptor e atuadores.

O componente receptor é composto por um módulo Bluetooth hc06 ou hc05 para a comunicação com o aplicativo para celular, um módulo de transmissão rádio nRF24 para comunicação com os controles e uma placa de microcontrolador pronta, que é responsável por traduzir as informações recebidas pelos receptores em informações que os atuadores podem entender.

Os atuadores serão motores que devem enrolar as cordas introduzidas anteriormente no título 2.1.1 Funcionamento da Prótese.

O sistema deve se conectar com o celular do utilizador a fim de receber informações sobre os modos de controle. Por meio destas informações, deve controlar os atuadores.



Energia

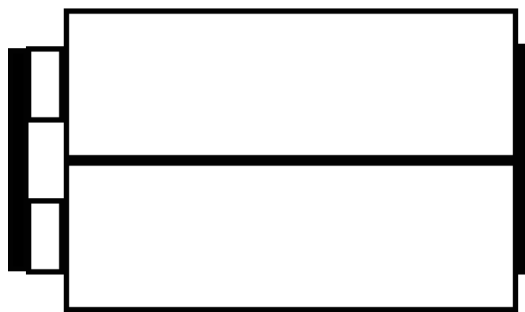
Em produto móvel uma parte fundamental é a autonomia. É de extrema importância que o produto tenha uma capacidade energética condizente com seu uso, porém também é muito importante que esta energia seja armazenada de forma conveniente e segura.

Pensando nestes pontos, as baterias mais condizentes com as necessidades foram as células de Li-ion, por serem comuns e baratas.

Existem problemas com esta escolha, porém, uma vez que baterias de lítio são instáveis em certas situações. Para diminuir os riscos, o sistema de encaixe deverá ser fácil de remover a fim de reduzir quaisquer danos. Além disso, mesmo com esta falha, baterias desse tipo são usadas em quase todos os aparelhos móveis atualmente, tais como celulares, notebooks e até carros elétricos. Mesmo sendo usadas em tantos locais, não é comum ouvirmos notícias de acidentes causados por elas.

O padrão recomendado para o projeto é o 18650, um formato de bateria de lítio que tem a voltagem entre 3.7v e 4.2v. Sua capacidade varia dependendo do modelo. São baterias baratas e encontradas em lojas virtuais do Brasil.

Para o funcionamento dos eletrônicos, uma bateria seria suficiente, porém, para que a autonomia do sistema seja melhorada, é recomendado que sejam adicionadas baterias em paralelo às já presentes, assim aumentando a capacidade energética do arranjo.



Exemplo com duas baterias
em paralelo

Com duas baterias o conjunto total teria, considerando as baterias com capacidade de 2400mAh, uma soma de 4800mAh a uma voltagem de 3.7v a 4.2v.

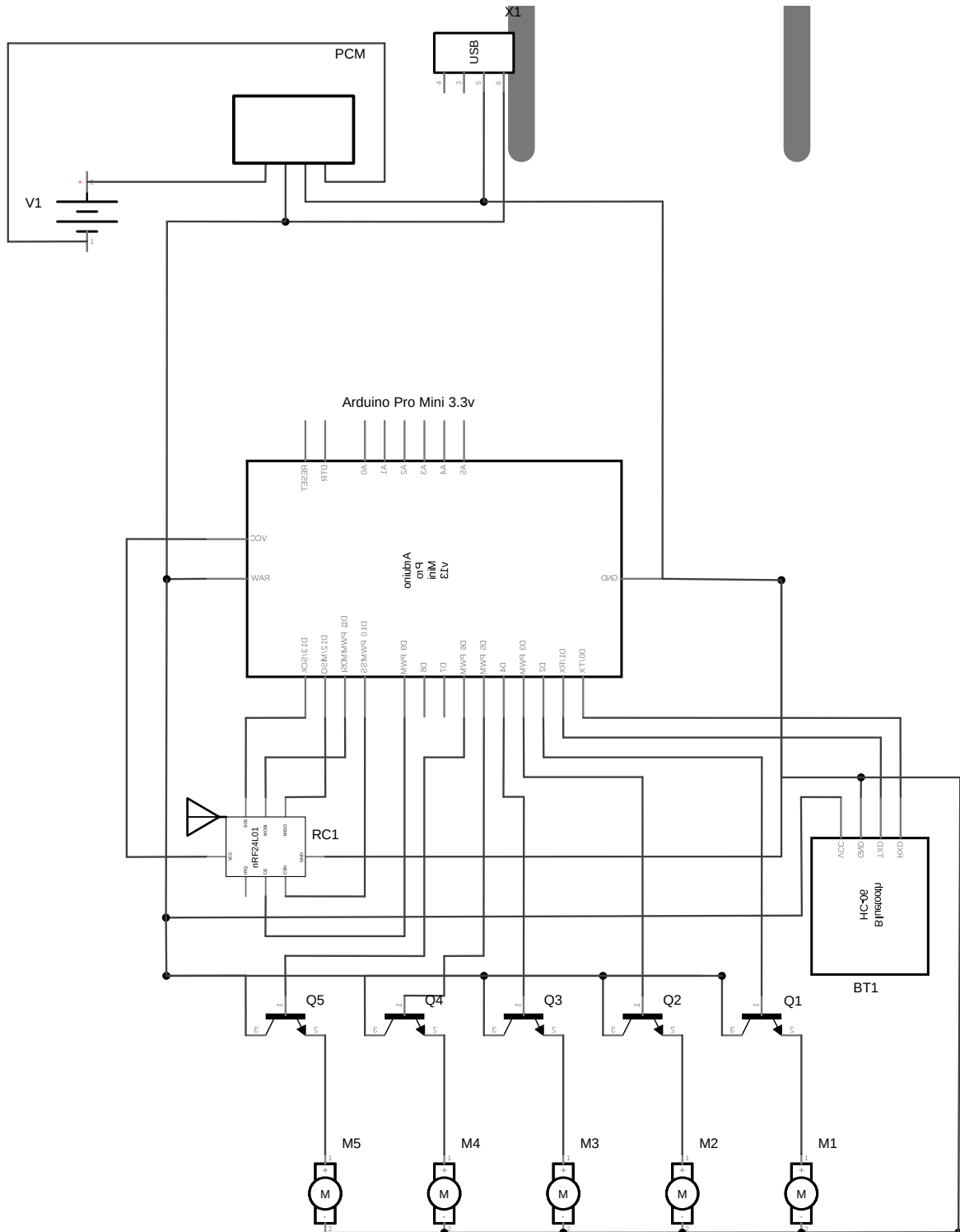
Entretanto, um fato importante a ser levado em conta é que a grande maior parte dos anúncios de bateria mentem e a real capacidade das baterias vendidas normalmente não passa de 600mAh.

É muito difícil saber se a bateria que está sendo comprada realmente cumprirá com a capacidade anunciada. Um meio de contornar este problema seria comparar de fabricantes como Samsung, diretamente de suas lojas, porém isso levaria o valor a ser alto.

A recarga das baterias pode ser feita tanto por meio de carregadores de bateria de lítio, onde é possível carregar uma bateria por vez, ou por meio do circuito já incluído no produto, onde só é necessário conectar uma fonte de energia compatível, como fontes de celular, por exemplo.

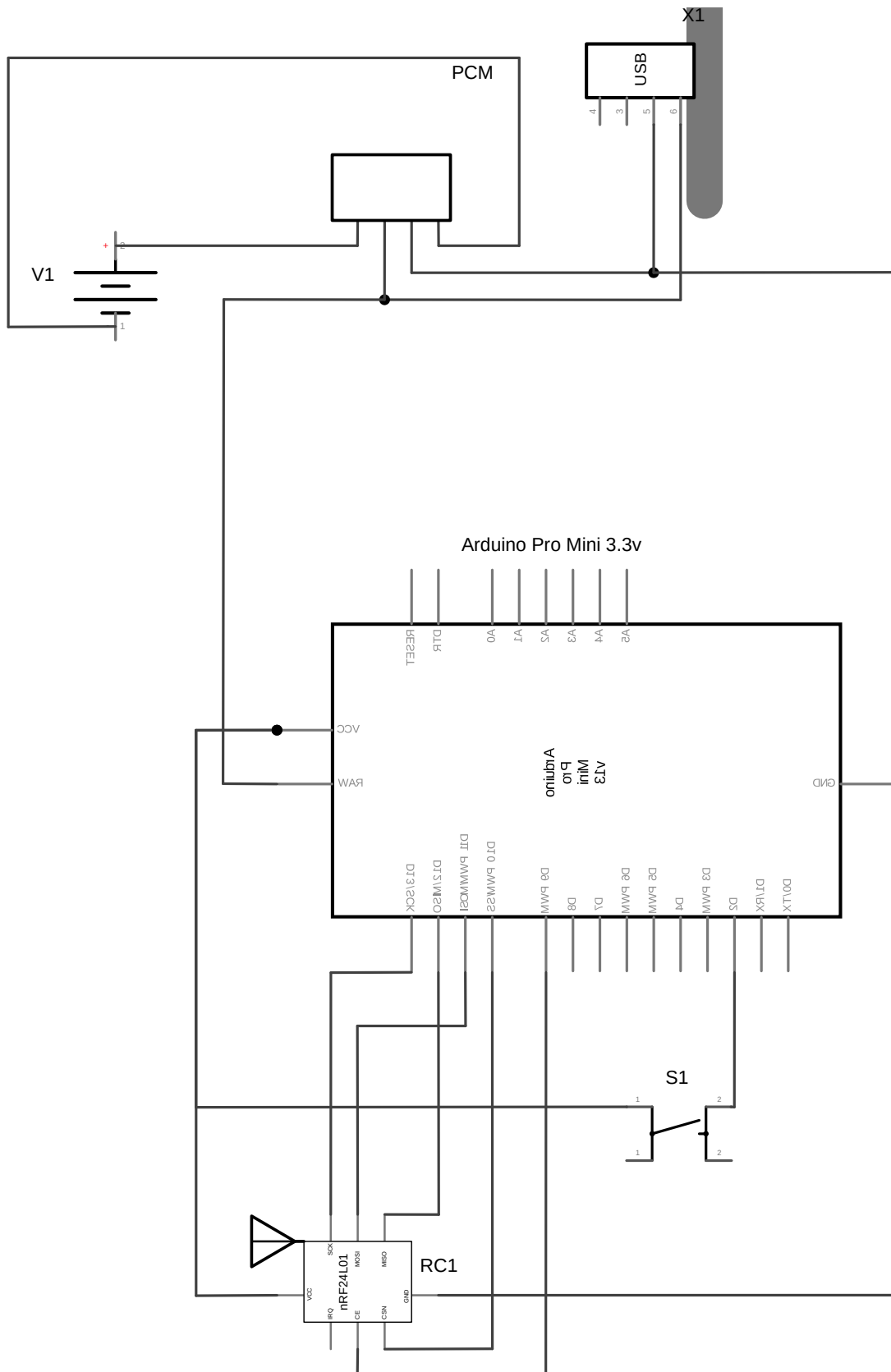
2.1.5 Projetos do Circuito

Conjunto Atuador



Símbolo	Componente	Descrição
PCM	Módulo de Carga e Descarga	Módulo que protege as baterias 18650 contra curto-circuito, sobrecarga e na carga e descarga.
X1	Conector Micro USB	Conexão para a fonte de carregamento da bateria
-	Arduino Nano	Placa baseada no microcontrolador ATmega328p que controla todo o sistema.
RC1	nRF24	Módulo de comunicação rádio de frequência 2.4GHz. Usado para a comunicação com o circuito de controle.
BT1	hc 06 ou hc 05	Módulo de comunicação Bluetooth usado na comunicação com o aplicativo para celular. Ambos os hc 06 ou hc 05 podem ser usados.
R1, R2, R3	Resistores de 220Ohms	Resistores usados para criar um divisor de tensão para não causar dano ao módulo Bluetooth.
Tr1, Tr2, Tr3, Tr4, Tr5	Transistor NPN	Transistor usado para cortar ou liberar corrente dos motores conforme o comando do microcontrolador.
M1, M2,M3, M4, M5	Motor DC	Motor DC que enrolará o fio para que o dedo se feche sobre a palma.

Conjunto de Controle



Símbolo	Componente	Descrição
PCM	Módulo de Carga e Descarga	Módulo que protege as baterias 18650 contra curto-circuito, sobrecarga e na carga e descarga.
-	Arduino Pro Mini 3.3v	Placa baseada no microcontrolador ATmega328p que controla todo o sistema.
RC1	nRF24	Módulo de comunicação rádio de frequência 2.4GHz. Usado para a comunicação com o circuito atuador.
S1	Chave Táctil (Push Button)	Serve como sensor para passar os comandos do usuário para o sistema. É somente um exemplo de sensor, outros podem ser utilizados.
X1	Conector Micro USB	Conexão para a fonte de carregamento da bateria.

2.1.7 Requisitos Funcionais

1. Deve ter a capacidade de segurar objetos;
2. Deve funcionar com uma fonte de energia móvel;
3. Deve seguir os modos de funcionamento dados pelo usuário por meio do aplicativo mobile.

2.2 SOFTWARE WEB

A parte web do projeto terá como função explicar sobre o produto, além de permitir a comunicação entre o interessado e os criadores.

Será composta de explicações sobre o projeto, como funcionamento e motivações, instruções de montagem, imagens de produtos prontos e um chat para que aqueles com problemas possam tirar dúvidas com um time de suporte.

2.2.1 Requisitos Funcionais

1. Deve ter chat para conversa com o time de suporte;
2. Deve possuir uma galeria com imagens do produto;
3. Deve conter informações sobre montagem e funcionamento.

2.2.2 Requisitos Não Funcionais

1. Deve funcionar em qualquer navegador a partir de uma versão 2020, incluindo navegadores de aparelhos móveis.
2. Deve ser desenvolvido com base em HTML, CSS, JavaScript (Puro, JQuery, Vue.js e Node.js).

2.3 SOFTWARE DESKTOP

A parte Desktop do projeto tem como alvo a criação de um sistema de gerenciamento para o time do produto, contando com um sistema para gerenciamento lojas que disponibilizam cada um dos componentes do sistema e um sistema de chat com o usuário do site.

2.3.1 Requisitos Funcionais

1. Deve ter um sistema de gerenciamento de fornecedores de cada componente;
2. Deve suprir a necessidade de um mecanismo de contato com o usuário do site por meio de um sistema de chat;
3. O sistema de chat deve contar com suporte para envio de imagens, melhorando o contato com o usuário para suporte.

2.3.2 Requisitos Não Funcionais

1. Deve funcionar no sistema Windows 10;
2. Deve ser desenvolvido em Java.

2.4 SOFTWARE MOBILE

O sistema Mobile do projeto será composto de um aplicativo que se conectará com o produto e poderá controlá-lo por Bluetooth.

Contará com um sistema de modos que definem como a prótese deve se mover. Cada modo confere dois estados que podem ser alternados pelo usuário usando os sensores da prótese.

2.4.1 Requisitos Funcionais

1. Deve poder se conectar com o sistema da prótese por Bluetooth;
2. Deve poder informar o hardware da prótese sobre como se mover dado o modo definido pelo usuário.

2.4.2 Requisitos Não Funcionais

1. Deve ser compatível com o sistema Android;
2. Deve estar disponível na Play Store;
3. Deve ser desenvolvido em Java ou Kotlin.

2.5 BANCO DE DADOS

O banco de dados do sistema será responsável por armazenar as informações do usuário no site, fazer a ligação do chat entre o site e o aplicativo Desktop, também armazenando estas informações, e armazenar os dados do sistema de gerenciamento presente no sistema para Windows.

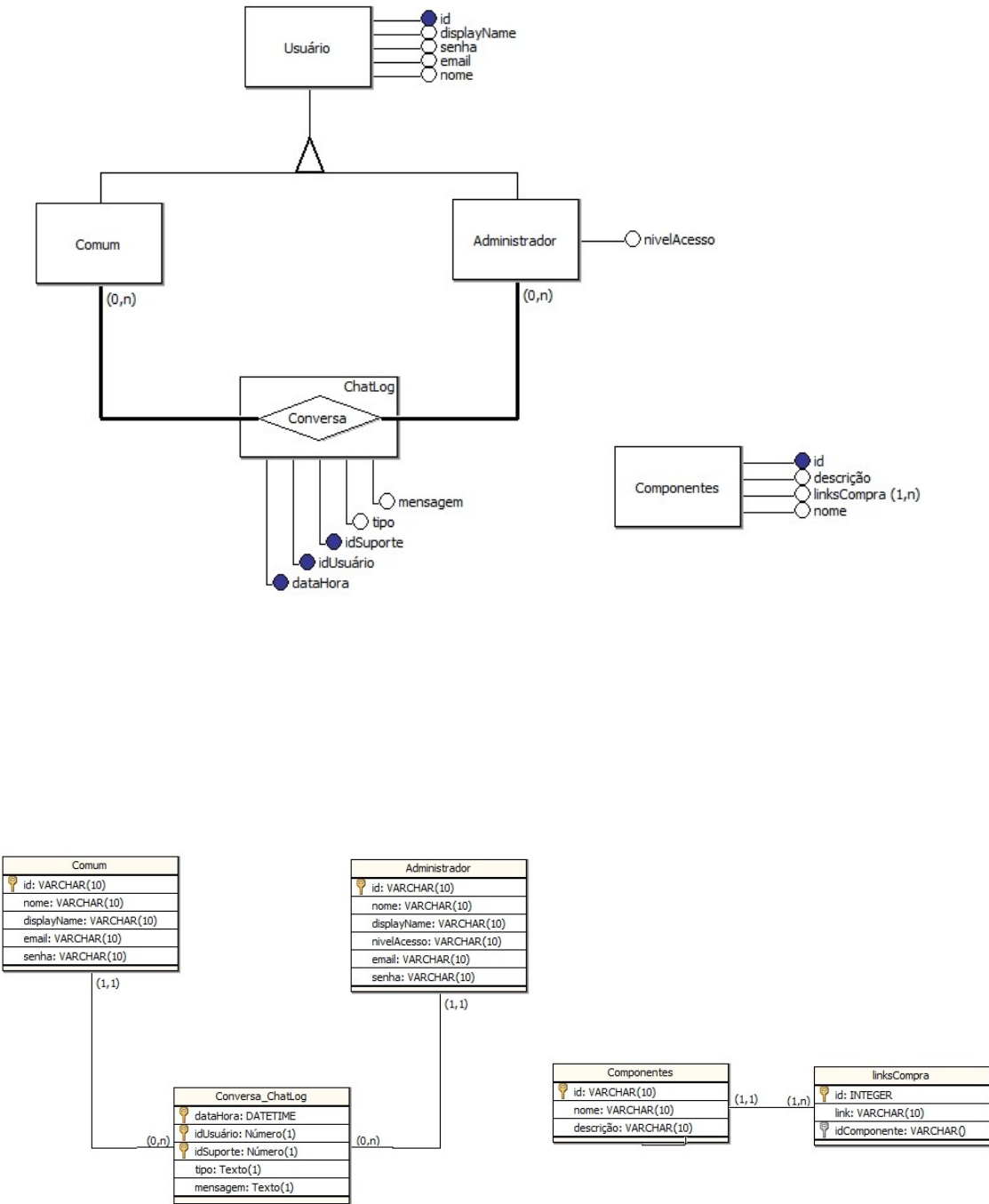
2.5.1 Requisitos Funcionais

1. Deve armazenar as informações do usuário;
2. Deve fazer a ligação do chat entre usuário web e time no Desktop;
3. Deve armazenar as informações de gerenciamento da solução Desktop.

2.5.2 Requisitos Não Funcionais

1. Deve ser desenvolvido com o sistema SQL;

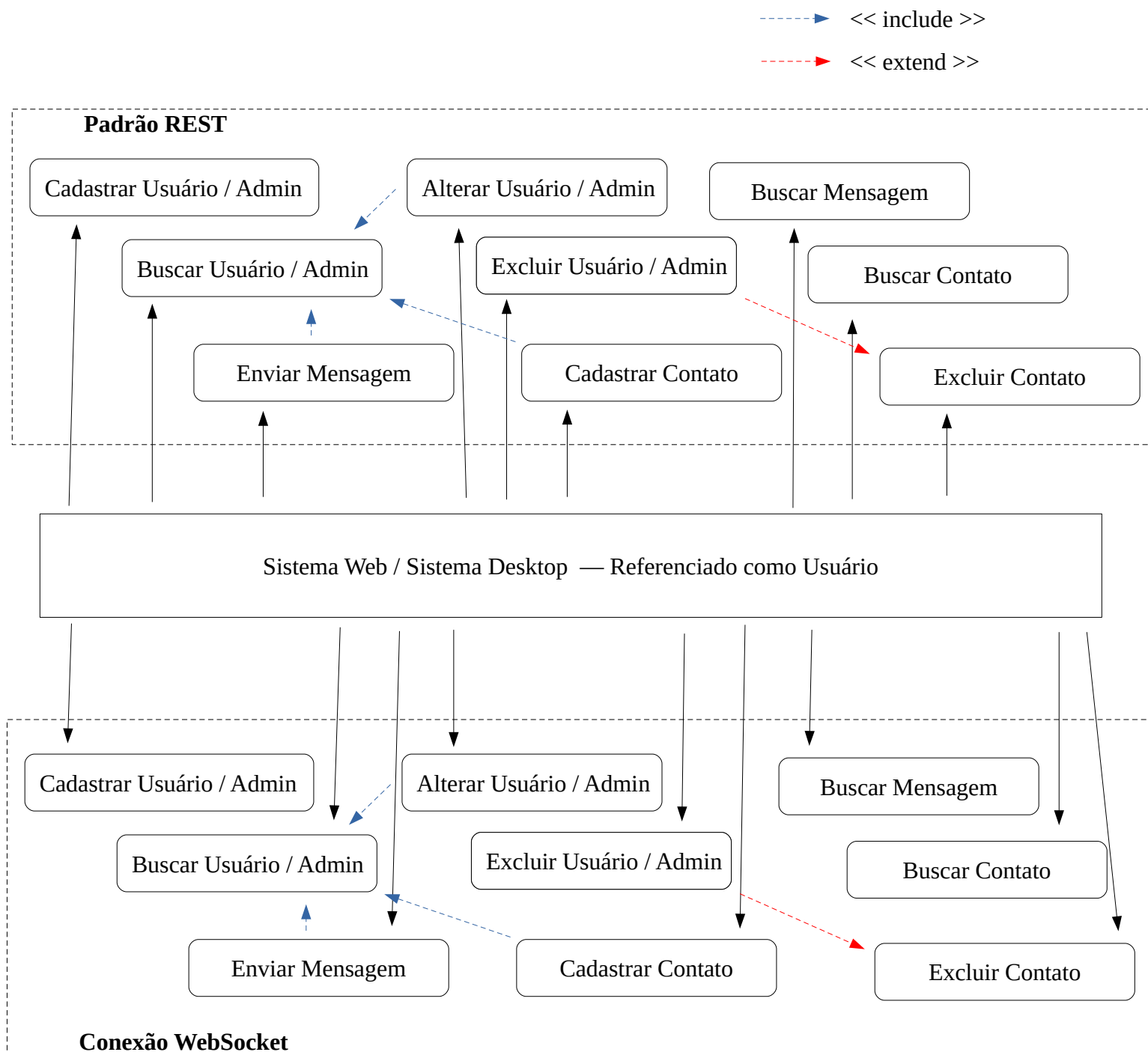
2.5.3 Diagramas



3 CASOS DE USO

3.1 PINUS (API)

Sendo a API desenvolvida em Node.js a base do funcionamento das partes Web e Desktop, essa sessão servirá para ambos os serviços.



3.1.1 Cadastrar Usuário / Admin

1. O usuário se conecta à API por um dos canais (HTTP / WebSocket);
2. O usuário solicita o cadastro, enviando os seguintes dados, sendo todos obrigatórios:
 - Nome de exibição (texto);
 - Nome (texto);
 - E-mail (texto);
 - Senha (texto);
 - Nível de acesso (número) (apenas no caso de cadastro de Admin).
3. O sistema valida as informações;
4. Caso as informações sejam válidas, o sistema cadastra o usuário no banco de dados e retorna as informações no formato de objeto; Caso as informações sejam inválidas, o sistema retorna um erro;

3.1.2 Buscar Usuário / Admin

1. O usuário se conecta à API por um dos canais (HTTP / WebSocket);
2. O usuário solicita uma busca, enviando os identificadores do registro prontamente;
3. O sistema valida as informações;
4. Caso as informações sejam inválidas, o sistema retorna um erro;
5. Caso as informações sejam válidas, o sistema faz uma busca no banco de dados usando os identificadores informados;
6. O sistema retorna uma lista contendo os objetos de cada resultado da busca. Os objetos são compostos do nome da coluna, seguido pelo valor na linha.

3.1.3 Alterar Usuário / Admin

1. O usuário se conecta à API por um dos canais (HTTP / WebSocket);
2. O usuário solicita uma alteração em determinado registro, enviando um objeto com a seguinte formação:

```
{  
  "identifier": {  
    "[ coluna da tabela ]": "[ valor atual ]",  
    ...  
  },  
  "password": "[ senha atual ]",  
  "update": {  
    "[ coluna da tabela ]": "[ novo valor ]",  
    ...  
  }  
}
```

3. O sistema executa a operação no banco de dados;
4. Caso haja um erro, o sistema retorna o erro;
5. Em caso de sucesso, o sistema executa uma **BUSCA (3.1.2)** e retorna seu resultado.

3.1.4 Exlcuir Usuário / Admin

1. O usuário se conecta à API por um dos canais (HTTP / WebSocket);
2. O usuário solicita uma exclusão em determinado registro, enviando um objeto com a seguinte formação:

```
{  
  "identifier": {  
    "[ coluna da tabela ]": "[ valor atual ]"  
  },  
  "password": "[ senha atual ]",  
}
```

3. O sistema executa a operação no banco de dados;
4. Caso haja um erro, o sistema retorna o erro;
5. Em caso de sucesso, retorna uma confirmação de conclusão.

3.1.5 Enviar Mensagem

1. O usuário se conecta à API por um dos canais (HTTP / WebSocket);
2. O usuário solicita o envio de uma mensagem, enviando um objeto com a seguinte formação:

```
{  
    "idAdmin": [ código do admin ],  
    "idUser": [ código do usuário ],  
    "datetime": null, //será preenchido pela API  
    "sender": "[ admin / user ]", //o tipo de usuário que está mandando a mensagem  
    "type": "[ text / image ]", //o tipo da mensagem,  
    "text": "[ texto ]", //Opcional  
    "image": "[ imagem ]" //Opcional  
}
```

3. O sistema adiciona a mensagem ao banco de dados;
4. Caso haja um erro, o erro é retornado;
5. Em caso de sucesso, o sistema envia uma confirmação ao usuário;
6. O sistema busca nas sessões ativas o destinatário da mensagem;
7. Caso o destinatário esteja "logado", o sistema envia a mensagem. Caso não esteja, a mensagem será retornada junto com o restante no momento da inicialização do "chat".

3.1.6 Buscar Mensagem

1. O usuário se conecta à API por um dos canais (HTTP / WebSocket);
2. O usuário solicita a busca de mensagens, enviando as informações que as identificam;
3. O sistema faz uma busca no banco de dados, usando as informações para identificá-las ;
4. Caso haja um erro, o sistema retorna o erro;
5. Em caso de sucesso, o sistema retorna uma lista contendo os objetos das mensagens.

3.1.7 Adicionar Contato

1. O usuário se conecta à API por um dos canais (HTTP / WebSocket);
2. O usuário solicita a adição de um contato, enviando um objeto com a seguinte formação:

```
{  
    "idAdmin": [ código do admin ];  
    "idUser": [ código do usuário ];  
}
```

3. O sistema adiciona o contato ao banco de dados;
4. Caso haja um erro, o sistema retorna o erro;
5. Em caso de sucesso, o sistema retorna um objeto com a mesma forma do enviado, porém com um elemento "id" contendo o número correspondente ao contato no banco de dados.

3.1.8 Buscar Contato

1. O usuário se conecta à API por um dos canais (HTTP / WebSocket);
2. O usuário solicita a busca de contatos, enviando as informações que os identificam;
3. O sistema faz uma busca no banco de dados, usando as informações para identificá-los ;
4. Caso haja um erro, o sistema retorna o erro;
5. Em caso de sucesso, o sistema retorna uma lista contendo os objetos dos contatos.

3.1.9 Excluir Contato

1. O usuário se conecta à API por um dos canais (HTTP / WebSocket);
2. O usuário solicita uma exclusão em determinado registro, enviando um objeto com a seguinte formação:

```
{  
  "identifier": {  
    "[ id / idAdmin / idUser ]": [ valor ] //Código do contato que  
                                     deve ser removido  
  },  
  "from": {  
    "[ idAdmin / idUser ]": [ valor ] //Código usuário que deseja  
                                     remover o contato  
  },  
  "password": "[ senha do usuário que deseja remover o contato ]",  
}
```

3. O sistema executa a operação no banco de dados;
4. Caso haja um erro, o sistema retorna o erro;
5. Em caso de sucesso, retorna uma confirmação de conclusão.

4 DEFINIÇÃO DE RECURSOS

4.1 STAKEHOLDERS

NOME	RESPONSABILIDADES	E-MAIL
Matheus Tomazella	Engenharia, Documentação, Web (back-end)	matheus250504@gmail.com
Luiz Felipe Costa Soares	Web (front-end), Desktop	lfcsoares04@gmail.com
Eduardo Gomes Heleno	Mobile, Firmware	Eduardo.gheleno@gmail.com

4.2 LISTA DE RECURSOS DE CONSTRUÇÃO

4.2.1 Ferramentas

- Impressora 3d;
- Multímetro.

4.2.2 Recursos Construção Mecânica

- **9x** Parafuso M4 cabeça chata 20mm;
- **1x** Parafuso M4 cabeça chata 26mm;
- **1x** Parafuso M4 cabeça chata 90mm;
- **?m** Filamento ??.

4.2.3 Recursos Construção Eletrônica

N = número de controles

- (N+1)x [Arduino Mini 3.3v \(modelo Pro opcional\)](#);
- (N+1)x [Placa de proteção de bateria 18650 BMS 1S 3A 4.2v](#);
- (N+1)x [Módulo de rádio nRF24](#);
- 1x [Módulo Bluetooth hc 06 ou hc 05](#);
- 5x Motor DC;
- 5x Transistor NPN que suporte 8.4v e 2A (opção: [BC368](#));
- 3x Resistor 220Ohms;
- (N+1)x Conector para fonte de celular;
- Chave Táctil (Push Button);
- Baterias Li-ion 18650;
- [Conversor USB Serial TTL](#).

5 OBSERVAÇÕES, MEDIÇÕES E CÁLCULOS

Tenho que construir primeiro pra fazer isso

5.1 ENERGIA

5.1.1 Capacidade das Baterias

5.1.2 Consumo de Energia