

Centro Estadual de Educação Tecnológica Paula Souza
ETEC Júlio de Mesquita
Curso Técnico em Eletrônica Integrado ao Ensino Médio

Alarme T- Security

Bruno Otavio Barboza¹
Leonardo Pelloni Biassio²
Luiz Henrique Araujo Cavalheri³
Matheus de Lima Santos⁴
Ramon Melo Cazetta⁵
Vitor Augusto Ripper Grezzani⁶

Relatório técnico apresentado ao Curso Técnico em Eletrônica da ETEC Júlio de Mesquita, orientado pelo professor Egmar Accetto e Adriana Mariko como requisito parcial para obtenção do título de Técnico em Eletrônica.

Santo André – SP
2025

Resumo

O trabalho apresenta o desenvolvimento de um sistema de segurança voltado para bicicletas, tendo em vista o aumento de furtos e a necessidade de soluções mais eficazes para proteção desse meio de transporte. O projeto integra três recursos principais: rastreamento por GPS, alarme sonoro e trava mecânica, proporcionando ao ciclista maior segurança sobre o seu veículo. A construção do protótipo envolve o uso de softwares de simulação e testes iniciais em protoboard, permitindo ajustes até a obtenção de um dispositivo funcional. Espera-se que a solução proposta seja acessível, eficiente e contribua para a redução de furtos, incentivando o uso da bicicleta como alternativa sustentável de deslocamento.

Palavras-chave: Bicicleta; Alarme; Segurança; GPS; Trava; Arduino.

Sumário:

1. INTRODUÇÃO.....	4
1.1 A bicicleta como meio de transporte sustentável.....	4
1.2 O aumento dos furtos de bicicleta no Brasil.....	6
1.3 Estratégias de prevenção de furtos de bicicletas.....	7
1.4 Justificativa.....	8
1.5 Objetivo.....	9
2. DESENVOLVIMENTO.....	10
2.1 Componentes mais relevantes.....	12
2.2 Arduino Nano.....	12
2.3 Mini Trava Solenóide.....	13
2.4 Módulo GPS GY-NEO6MV2 com Antena.....	14
2.5 Módulo GSM GPRS SIM800L.....	15
2.6 Módulo Buzzer Ativo.....	16
3. CÓDIGO DO SISTEMA.....	18
3.1 Código de acionamento do sistema (Apêndice).....	18
4. LAYOUT E IMPRESSÃO 3D.....	21
5. TABELA DE PREÇOS DO PROJETO.....	23
6. EVOLUÇÃO PARA UM PRODUTO COMERCIAL DE MERCADO.....	24
7. TABELA DE PREÇOS PARA UM PRODUTO COMERCIAL.....	25
8. CONCLUSÃO.....	26
9. REFERÊNCIAS.....	27
APÊNDICE.....	30

1. INTRODUÇÃO

1.1 A bicicleta como meio de transporte sustentável

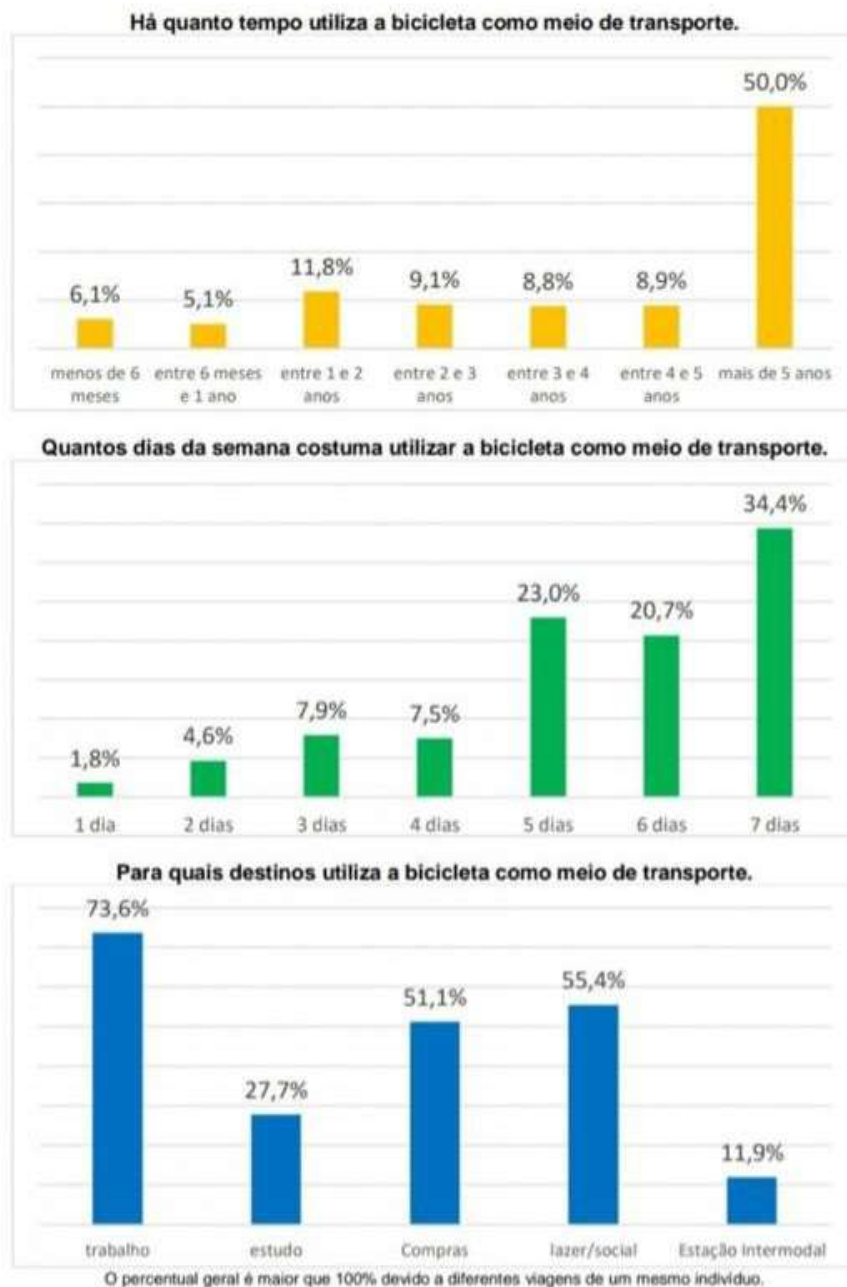
A bicicleta tem ganhado cada vez mais espaço como meio de transporte alternativo nas áreas urbanas, destacando-se por sua praticidade, baixo custo e contribuição para a mobilidade sustentável. Segundo Miranda e Carvalho (2022), o uso da bicicleta cresceu significativamente nas últimas décadas, impulsionado por políticas de mobilidade ativa e pela busca por alternativas ao transporte motorizado tradicional. Esse aumento está associado tanto à conscientização ambiental quanto à necessidade de melhorar a qualidade de vida nas cidades.

No contexto brasileiro, Jânio Santos e Luiz Eduardo Santos (2022) destacam que o uso da bicicleta representa uma nova forma de pensar o planejamento urbano, pois promove o equilíbrio entre os diferentes modais de transporte e contribui para cidades mais democráticas e acessíveis. Além disso, observa-se que muitos ciclistas utilizam a bicicleta não apenas por economia, mas também por valores ligados à sustentabilidade e pelo prazer da atividade, demonstrando uma mudança cultural em direção a hábitos mais conscientes e saudáveis.

A crescente adesão ao uso da bicicleta no Brasil é evidenciada pela pesquisa de Jornal do Commercio de 2025, que mostra que a maioria dos ciclistas utiliza bicicletas cinco ou mais vezes por semana, indicando que esse modal vem ganhando força como meio de transporte cotidiano. Mesmo com limitações como infraestrutura insuficiente e insegurança no trânsito, o aumento do uso é expressivo e reforça a necessidade de investimentos em ciclovias e políticas de mobilidade urbana que incentivem o deslocamento sustentável.

Imagem 1 - Uso Crescente da Bicicletas

Aspectos dos deslocamentos, hábitos e motivações.



Fonte: jc.uol.com.br/colunas/mobilidade/2025/02/10/ciclistas-brasileiros

Esses resultados demonstram que a bicicleta deixou de ser vista apenas como uma opção de lazer e passou a integrar a agenda de mobilidade urbana sustentável, sendo reconhecida como um meio de transporte viável, econômico e ambientalmente responsável.

1.2 O aumento dos furtos de bicicleta no Brasil

Apesar da crescente adoção da bicicleta como meio de transporte, um dos principais desafios enfrentados pelos ciclistas é o aumento dos furtos. Esse problema tem se tornado recorrente em diversas regiões do país, gerando insegurança e desestimulando o uso.

De acordo com dados do Instituto Brasileiro de Geografia e Estatística (2022), cerca de 388 mil bicicletas foram furtadas no Brasil em 2021. Esse número expressivo demonstra a fragilidade da segurança urbana voltada aos ciclistas. No estado do Rio de Janeiro, por exemplo, somente em maio de 2025 foram registradas 413 ocorrências de furto de bicicletas, representando um aumento de 18% em relação ao mesmo período de 2024 (Instituto de Segurança Pública do Rio de Janeiro, 2025).

Um levantamento realizado por Souza, Reis, Queiroz e Formigoni (2024) apontou que a cidade de São Paulo apresenta um índice baixo de segurança contra furto de bicicletas, quando comparada a outras cidades internacionais. Isso evidencia que o crescimento do ciclismo urbano não tem sido acompanhado por políticas públicas eficazes de prevenção ao furto e de proteção aos usuários.

Dessa forma, o aumento dos furtos de bicicletas representa um paradoxo da mobilidade sustentável: enquanto a bicicleta é promovida como um símbolo de progresso ambiental e social, a falta de segurança e de infraestrutura adequada ameaça sua consolidação como meio de transporte confiável.

De acordo com reportagens divulgadas pelo portal G1 (2025) e Jornal Digital Ultimosegundo (2023), o furto de bicicletas vem crescendo ano após ano, tanto em grandes capitais quanto em cidades de médio porte. Esse aumento está relacionado à facilidade com que ladrões conseguem burlar sistemas tradicionais de travamento. Como resultado, muitos usuários deixam de utilizá-la como meio de locomoção diária, motivados pelo receio de perder seu bem.

Imagem 2 - Reportagem.

Furto de bicicleta aumenta 59% no Estado do Rio; veja as áreas com mais casos

Na capital, incidência foi ainda pior no primeiro bimestre deste ano, com aumento de 106%

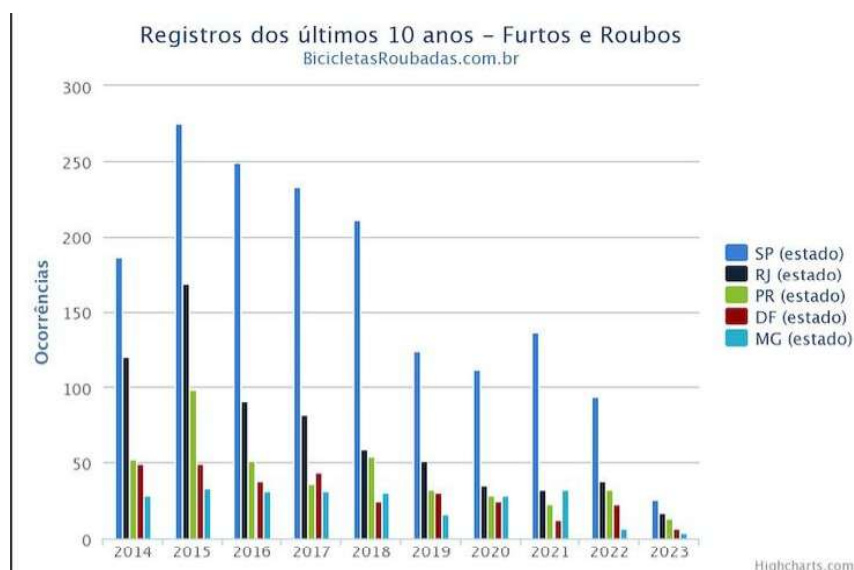
Por Marcos Nunes e Gabriela Germano — Rio de Janeiro
08/04/2025 08:30 Atualizado há 2 meses



Furtos de bicicletas: foram 838 ocorrências em janeiro e fevereiro no Rio, 59% a mais que no mesmo período de 2024 — Foto: Marcia Foleto / Agência O Globo

Fonte: <https://oglobo.globo.com/rio/noticia/2025/04/08>

Imagem 3 - Aumento de furtos



Fonte: ultimosegundo.ig.com.br/brasil/2023-10-26

1.3 Estratégias de prevenção de furtos de bicicletas

Diante do aumento dos furtos, torna-se fundamental discutir estratégias de prevenção que possam garantir mais segurança aos ciclistas e incentivar o uso contínuo da bicicleta como meio de transporte. Segundo Viana, Silva, Oliveira e Campos (2023), as políticas mais eficazes de prevenção combinam tecnologia, infraestrutura e regulamentação.

Entre as medidas mais adotadas em cidades que possuem bons índices de segurança ciclovária estão a instalação de bicicletários monitorados, o uso de sistemas de rastreamento por GPS e a criação de cadastros públicos de bicicletas, que facilitam a recuperação em casos de furto. Além disso, Jânio Santos e Luiz Eduardo Santos (2022) destacam a importância da integração entre políticas de mobilidade e segurança pública, com ações conjuntas entre prefeituras, polícias e associações de ciclistas.

Portanto, investir em educação, infraestrutura e tecnologia de segurança é essencial para que a bicicleta seja consolidada como uma alternativa real e segura de mobilidade sustentável. Sem essas medidas, o avanço do ciclismo urbano continuará limitado pela sensação de insegurança e pelas perdas materiais que afetam diretamente os usuários.

1.4 Justificativa

Diante desse cenário, torna-se necessário buscar soluções tecnológicas mais eficientes para proteção do ciclista e de sua bicicleta. Este projeto apresenta o desenvolvimento de um sistema integrado de segurança composto por três elementos principais: rastreamento por GPS, que permite monitorar a localização do veículo em tempo real; alarme sonoro, que atua na inibição imediata da ação criminosa; e trava mecânica, que forma uma barreira física contra a remoção indevida. Uma abordagem que combina três componentes tecnológicos integrados, formando uma “pilha de defesa” que amplia a segurança do equipamento e reduz o risco efetivo de furto.

Quando essas três tecnologias são integradas elas passam a operar como uma camada de segurança sequencial: a trava dificulta a ação inicial, o alarme gera alerta imediato-local e o rastreador oferece meios de localização e recuperação pós-furto. Em conjunto a essas tecnologias, políticas de infraestrutura (bicicletários seguros, iluminação, vigilância) e regulamentação (registro obrigatório de bicicletas, bancos de dados públicos) precisam operar de forma articulada para que a prevenção seja eficaz.

Considerando que o uso da bicicleta cresce, torna-se crítico que os usuários adotem dispositivos tecnológicos apropriados e que municípios e estados incentivem ou regulamentem essas tecnologias como parte das estratégias de mobilidade urbana sustentável.

1.5 Objetivo

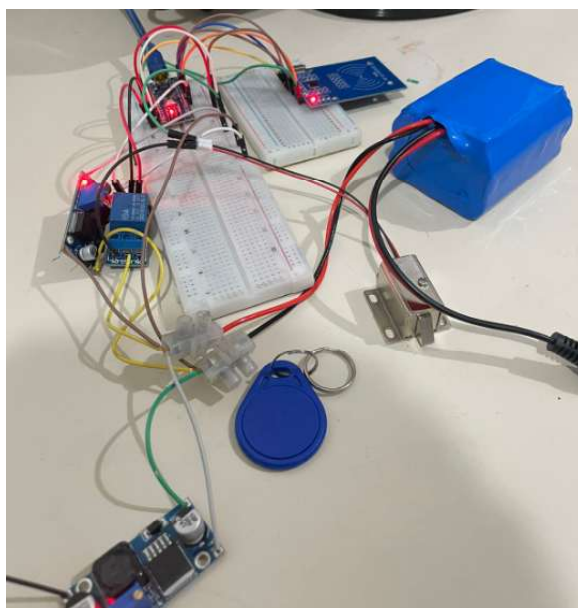
Desenvolver um sistema integrado de segurança para bicicletas que consiste em um circuito eletrônico capaz de controlar o alarme sonoro e a trava mecânica de maneira eficiente, além de implementar um sistema de rastreamento por GPS que permita ao usuário acompanhar a localização da bicicleta em tempo real, com o intuito de reduzir furtos e aumentar a proteção do usuário.

2. DESENVOLVIMENTO

O sistema proposto tem como objetivo oferecer um alarme de proteção inteligente para bicicletas, utilizando componentes eletrônicos integrados ao microcontrolador Arduino Nano, que atua como o cérebro do sistema, coordenando o funcionamento de todos os módulos e componentes para assegurar o pleno desempenho. A bateria de 12V e 4,4A é responsável pela alimentação móvel e recarregável de todo o circuito, garantindo autonomia e praticidade ao dispositivo.

A Mini Trava Solenóide é alimentada diretamente pela bateria, já que necessita de 12V para seu correto funcionamento, sendo acionada por meio do Módulo RFID, que identifica o cartão de acesso e realiza o travamento automático. O módulo relé ativo controla o acionamento da trava de forma mais eficiente, rápida e com menor consumo de energia.

Imagem 4 - Circuito de acionamento Mini Trava Solenóide



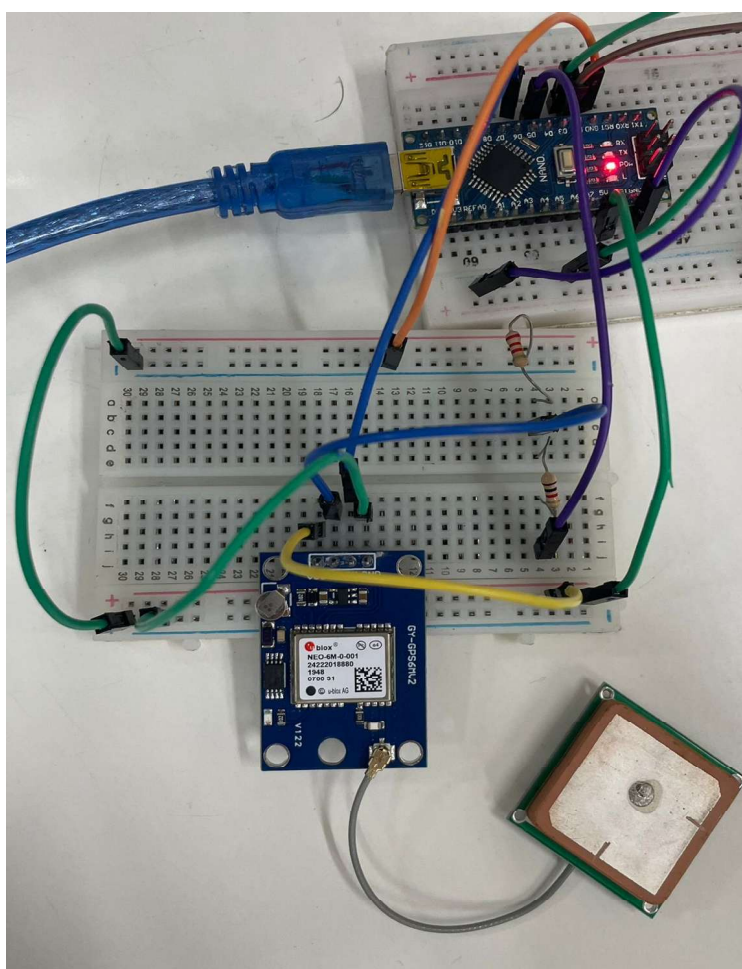
Fonte: Foto de Autoria própria- 19/06/2025

Para o controle de energia, foram utilizados dois Módulos Reguladores de Tensão Ajustável LM2596 3A, sendo um dedicado ao Módulo GSM GPRS SIM800L, que opera em torno de 4V. Para proteger o módulo SIM800L contra picos de corrente, que podem chegar a 2A, foi inserido um capacitor de 1000uF no circuito,

garantindo maior estabilidade e segurança. O segundo regulador tem a função de estabilizar o restante do circuito, visto que o Arduino não é capaz de fornecer corrente suficiente para todos os componentes simultaneamente.

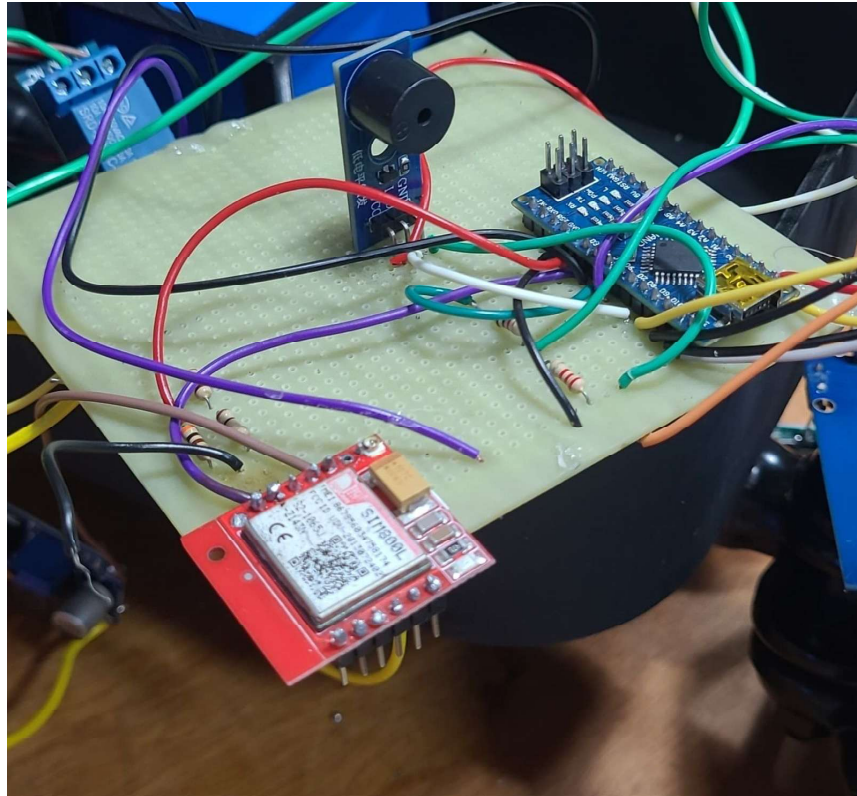
O Módulo GSM GPRS SIM800L, integrado ao sistema, tem a função de enviar a localização da bicicleta via SMS, utilizando dados obtidos pela última localização. O Módulo GPS GY-NEO6MV2 com antena, é responsável por captar as coordenadas geográficas por sinal de satélite.

Imagem 5 - Circuito GPS com Antena



Além disso, o módulo buzzer *ativo* atua sempre que ocorre a leitura de um cartão RFID não autorizado. Quando um usuário tenta desbloquear a trava com um cartão inválido, é identificado uma tentativa suspeita e envia um pulso de ativação ao buzzer, que emite um alerta sonoro durante 1 segundo.

Imagem 6 - Circuito GPS SIM800L e Buzzer



Fonte: Foto de Autoria própria- 28/11/2025

2.1 Componentes mais relevantes

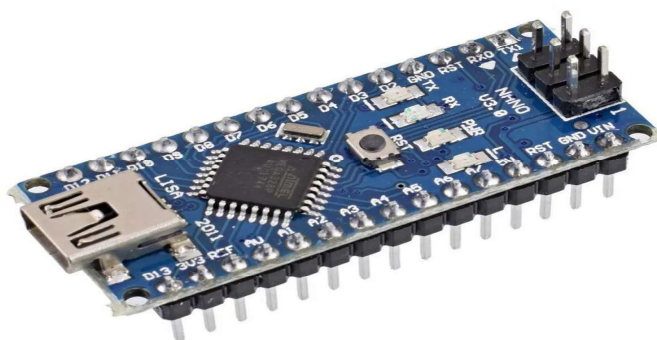
2.2 Arduino Nano

O Arduino Nano é uma placa microcontroladora compacta baseada no ATmega328P, o mesmo chip do Arduino Uno, porém em formato reduzido é ideal para projetos de pequeno porte. Possui 14 pinos digitais (6 PWM), 8 entradas analógicas, oscilador de 16MHz, conexão USB e pinos para alimentação externa, podendo ser tanto por meio da porta USB de 5V quanto por fonte externa entre 7 V e 12 V.

Segundo a documentação oficial da Arduino (2022), o Nano mantém o mesmo desempenho do Uno, oferecendo praticidade em aplicações embarcadas e dispositivos portáteis que exigem baixo consumo e economia de espaço.

No projeto, o Arduino Nano atua como o “cérebro” do sistema de segurança, processando os sinais dos módulos e controlando o buzzer, a trava mecânica e o rastreamento GPS, garantindo o funcionamento coordenado do dispositivo e contribuindo para a prevenção de furtos e proteção do ciclista.

Imagem 7 - Arduino Nano.



Fonte: www.robocore.net/placa-arduino/placa-nano-v3

2.3 Mini Trava Solenóide

A mini trava solenóide de 12 V é um atuador eletromecânico utilizado em sistemas de segurança, como fechaduras eletrônicas e cofres. Para operar corretamente, a mini solenóide é alimentada diretamente pela bateria de 12V, recebendo apenas o comando de acionamento pelo circuito de controle. No projeto, esse comando é fornecido pelo Módulo RFID, que identifica o cartão de acesso e aciona automaticamente a trava. O módulo relé incorporado ao circuito permite uma ação mais rápida, menor consumo e maior eficiência ao controlar a corrente necessária para a solenóide.

De acordo com a HU Infinito (2025), seu funcionamento ocorre a partir do acionamento de uma bobina que, ao receber corrente, gera um campo magnético capaz de puxar o êmbolo metálico e liberar o mecanismo. Quando a alimentação é interrompida, a mola interna devolve o êmbolo à posição original, restabelecendo o travamento. Esse processo é rápido, geralmente inferior a um segundo, e garante respostas precisas no sistema.

Imagem 8 - Mini Trava Solenoide 12V



Fonte: www.robocore.net/mini-trava-eletrica-solenoid-12v

2.4 Módulo GPS GY-NEO6MV2 com Antena

O módulo GPS GY-NEO6MV2 é um dispositivo compacto e preciso de rastreamento baseado no chip u-blox NEO-6. Ele converte sinais de satélites em dados de latitude, longitude, velocidade e horário, transmitidos em formato NMEA por meio da interface UART, pinos Tx e Rx, facilitando a comunicação com microcontroladores.

Segundo a fabricante u-blox (2011), o módulo possui alta sensibilidade (-161 dBm), precisão média de 2,5 metros e inicialização rápida, podendo começar a enviar dados em até 1 segundo após o acionamento. Alimentado entre 3,3 V e 5 V, inclui antena cerâmica integrada e memória EEPROM, garantindo estabilidade e rapidez na aquisição de sinais.

No projeto, o módulo GPS atua como a principal camada de rastreamento, permitindo localização em tempo real em caso de furto. Integrado ao Arduino Nano, ele envia continuamente as coordenadas do veículo, que podem ser monitoradas remotamente. Sua instalação requer apenas a visibilidade do céu e uma conexão simples aos pinos de comunicação e alimentação.

Imagem 9 - Módulo GPS GY-NEO6MV2 com Antena



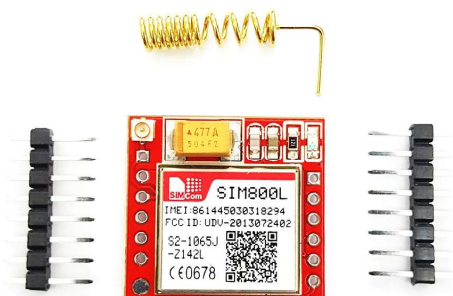
Fonte: www.robocore.net/sensor-robo/modulo-gps-gy-neo6mv2-com-antena

2.5 Módulo GSM GPRS SIM800L

O módulo GSM/GPRS SIM800L é um dispositivo compacto utilizado em projetos de comunicação móvel, possibilitando o envio de SMS, realização de chamadas e transmissão de dados via rede GPRS. O módulo opera em faixa quad-band (850/900/1800/1900 MHz), garantindo compatibilidade com redes de telefonia em diversos países. De acordo com o *SIM800L Hardware Design* fornecido pela própria SIMCom (SIMCOM, 2013), seu funcionamento exige uma alimentação estável entre 3,4 V e 4,4 V, além de capacidade para suportar picos de corrente de até 2 A durante as transmissões GSM.

Para atender a essas exigências elétricas, o sistema utiliza um regulador de tensão LM2596, responsável por reduzir os 12V da bateria para a faixa adequada ao módulo, garantindo fornecimento contínuo e seguro. Complementando essa estabilização, um capacitor de 1000 μF é inserido na linha de alimentação para absorver os picos de corrente e evitar quedas bruscas de tensão, assegurando o desempenho confiável do SIM800L. A comunicação com o microcontrolador é realizada por meio de interface UART, utilizando comandos AT para controle de todas as funções do modem.

Imagem 10 - Módulo GSM GPRS SIM800L



Fonte: www.robocore.net/modulo-gsm-gprs-sim800l

2.6 Módulo Buzzer Ativo

O módulo buzzer ativo é um dispositivo utilizado em projetos com Arduino para geração de alertas sonoros. Possui oscilador interno, o que o diferencia do buzzer passivo ao dispensar sinais PWM para produzir som: basta que o pino de sinal receba nível lógico alto para que o módulo emita um tom contínuo na faixa de 2 kHz a 4 kHz. Sua estrutura é composta por três pinos VCC, GND e SIG permitindo conexão direta ao microcontrolador, já que o próprio módulo integra resistores e transistores de acionamento. Quando o sinal é desligado (nível lógico baixo), o circuito interno cessa a oscilação e o buzzer permanece silencioso.

O funcionamento mostrado corresponde ao padrão de buzzers ativos comerciais, como o modelo Kingstate KPEG158. Segundo seu datasheet, o componente opera em 5V, com frequência típica próxima de 3 kHz e modo self-drive, termo que indica que o buzzer contém o próprio circuito oscilador responsável pela geração do sinal sonoro.

Imagem 11 - Módulo Buzzer Ativo



Fonte: chip.com.br/produtos/modulo-eletronico-buzzer-5v

3. CÓDIGO DO SISTEMA

Essa etapa representa a base lógica e funcional de todo o sistema desenvolvido, atuando na interpretação de sinais, tomada de decisões e execução dos comandos necessários ao funcionamento de cada dispositivo. Dessa forma, o código não apenas define o comportamento eletrônico, mas também assegura a interação eficiente, consolidando o funcionamento confiável da solução proposta.

3.1 Código de acionamento do sistema (Apêndice)

Biblioteca / Função

#include <SPI.h>

#include <MFRC522.h>

#include <SoftwareSerial.h>

#include <TinyGPS++.h>

SPI.h / Comunicação do Arduino com o RFID

MFRC522.h / Leitura e processamento dos cartões RFID

SoftwareSerial.h / Cria portas seriais extras para GPS e SIM800L

TinyGPS++.h / Lê e traduz dados de coordenadas do GPS

Configuração do RFID, relé e buzzer

#define SS_PIN 10

#define RST_PIN 9

#define RELE_PIN 4

#define BUZZER_PIN 8

#define BOTAO_PIN 7

Função de cada componente:

Componente / Função

RELE_PIN / Liga/desliga a trava (solenóide)

BUZZER_PIN / Emite alerta ao cartão não autorizado

SS_PIN / RST_PIN / Comunicação com o leitor RFID

BOTAO_PIN / Reserva para acionamento manual no futuro

A variável **estadoSolenóide** define se a trava está aberta ou fechada.

Módulo SIM800L + GPS

São criadas duas seriais adicionais:

SoftwareSerial sim800(3, 2);

SoftwareSerial gpsSerial(6, 5);

GPS recebe dados continuamente

SIM800L é ativado quando necessário, economizando tempo

O intervalo para checar SMS é de 30 segundos:

const unsigned long INTERVALO_SMS = 30000UL;

Setup:

RFID ativado e antena no ganho máximo

Solenóide começa TRAVADA

Buzzer desligado

GPS e SIM800 iniciados

SIM800 configurado para modo texto

Loop Principal

Funções executam sempre na mesma ordem:

1) Atualização do GPS

```
gpsSerial.listen();
```

```
atualizaGPS();
```

Ele fica capturando coordenadas enquanto o Arduino executa o restante do código.

2) Leitura RFID

```
if (mfr522.PICC_IsNewCardPresent() && mfr522.PICC_ReadCardSerial())
```

```
{
```

```
    // Gera UID do cartão e compara com autorizado
```

```
}
```

Se o UID = "8644F604" → Acesso liberado

Se travada → destrava

Se destravada → trava

Caso contrário:

CARTÃO NÃO AUTORIZADO → Buzzer toca por 1 segundo

Nenhuma alteração no relé

Ao final de cada leitura, o RFID é reiniciado para evitar travamentos:

```
mfr522.PCD_Init();
```

```
mfr522.PCD_SetAntennaGain(...);
```

3) Verificação de SMS a cada 30 segundos

verificaSMSComando();

Se receber um SMS com a palavra Localizacao:

GPS é atualizado por 1 segundo

Se coordenadas forem válidas → envia SMS com latitude/longitude + link

Google Maps

Caso não tenha sinal GPS → SMS de erro

Função / O que faz

buzzerNaoAutorizado() / Emite som por 1s para acesso negado

inicializaSIM800() / Configura módulo para envio/recebimento SMS

atualizaGPS() / Atualiza coordenadas GPS continuamente

coordenadasValidas() / Verifica se GPS capturou localização

enviaSMSLocalizacaoPara() / Envia mensagem com link do Google Maps

verificaSMSComando() / Lê SMS recebido e responde com localização

4. LAYOUT E IMPRESSÃO 3D

A caixa desenvolvida é constituída por cinco componentes principais: o corpo principal, a tampa, o suporte da bateria, o suporte da trava e dois fixadores destinados à instalação no cano do banco da bicicleta. Todas as peças foram fabricadas por meio de impressão 3D utilizando a impressora Bambu Lab A1, sendo empregado como material de fabricação o filamento do tipo PLA (Ácido Polilático), escolhido devido à sua boa resistência mecânica, facilidade de impressão e custo acessível. O conjunto foi projetado com o objetivo de garantir a proteção dos componentes eletrônicos, a organização interna dos elementos e a fixação segura do sistema ao veículo.

Primeiramente, o circuito eletrônico é posicionado no compartimento central interno da caixa, o qual apresenta dimensões compatíveis com a placa, assegurando sua correta acomodação e reduzindo a influência de vibrações durante o deslocamento da bicicleta. Em seguida, a bateria é inserida em seu compartimento específico, fisicamente separado do circuito, de modo a evitar contato direto entre os componentes e possíveis danos causados por movimentações ou aquecimento.

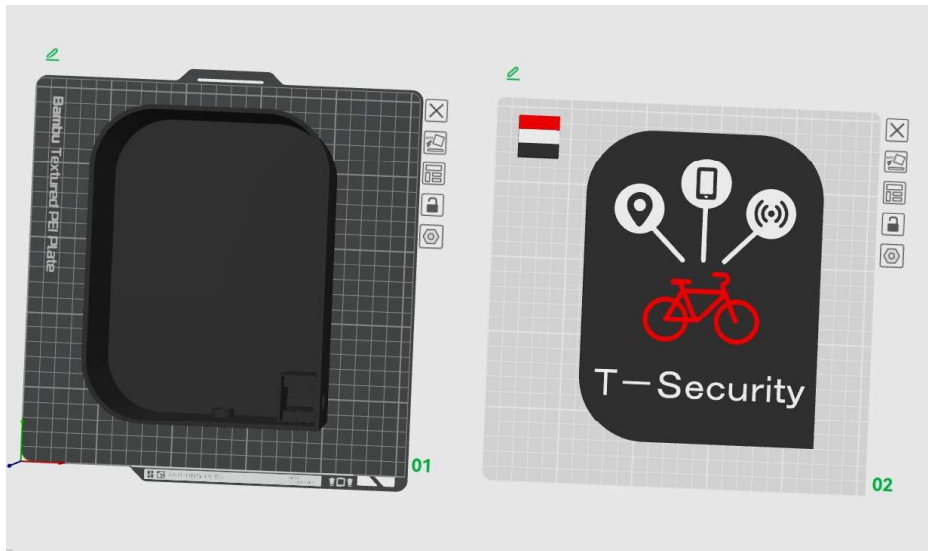
Posteriormente, a trava é instalada em um alojamento independente, projetado para permitir seu encaixe adequado e garantir o pleno funcionamento do mecanismo de segurança do sistema. A separação entre os compartimentos contribui para a organização interna e para a segurança operacional do conjunto.

Após a instalação de todos os componentes internos, a tampa é acoplada ao corpo da caixa, promovendo o fechamento do sistema e garantindo a proteção contra poeira, umidade e impactos externos, fatores comuns ao ambiente de uso da bicicleta.

A fixação da caixa à bicicleta é realizada por meio de dois fixadores posicionados na parte inferior do conjunto, os quais são presos diretamente ao cano do banco da bicicleta. Essa configuração proporciona estabilidade mecânica ao sistema, minimizando deslocamentos indesejados e garantindo a integridade dos componentes durante o uso em diferentes condições de terreno.

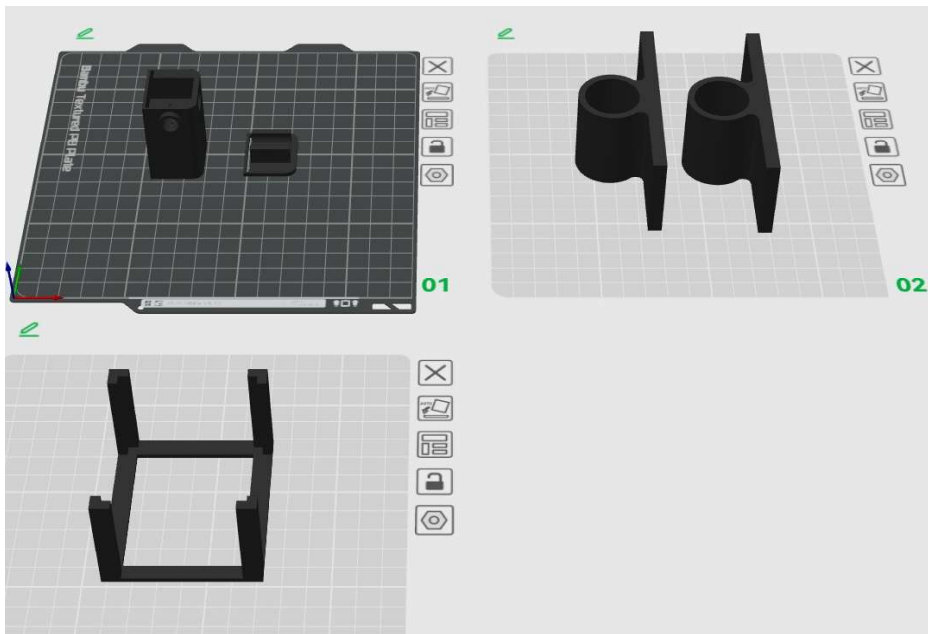
Dessa forma, a montagem resulta em um sistema compacto, seguro e funcional, atendendo aos requisitos de proteção, organização e fixação necessários para a aplicação proposta.

Imagem 13 - Caixa e Tapa



Fonte: Foto de Autoria própria- 27/11/2025

Imagem 14 - Acoplamentos



Fonte: Foto de Autoria própria- 27/11/2025

5. TABELA DE PREÇOS DO PROJETO

Produto	Quantidade	Preço Unitário (R\$)	Total (R\$)
Arduino Nano V3	1	31,50	31,50
Protoboard 40 Pontos	2	6,90	13,80
Jumpers	1	14,00	14,00
Resistores	20	0,50	10,00
Módulo Regulador de Tensão LM2596 3A	2	7,50	15,00
Mini Trava Elétrica Solenoide 12V	1	32,90	32,90
Módulo Buzzer 5V Ativo	1	5,90	5,90
Módulo GPS GY-NEO6MV2 com Antena	1	39,90	39,90
Capacitor 1000 μ F 10V	1	1,50	1,50
Bateria 12V/4400mAh	1	127,00	127,00
Kit RFID MFRC522 (13,56MHz)	1	13,50	13,50
Módulo GSM GPRS SIM800L	1	39,90	39,90
Impressão de Caixa 3D	1	40,00	40,00
Total			R\$ 391,40

6. EVOLUÇÃO PARA UM PRODUTO COMERCIAL DE MERCADO

A partir do desempenho satisfatório do protótipo, propõe-se sua evolução para um produto comercial adequado ao uso real em segurança veicular. Para isso, seria necessário substituir o Arduino por uma placa eletrônica própria (PCB industrial) com microcontrolador de maior desempenho, como STM32 ou ESP32-S3, integrada a proteções elétricas, antena otimizada e fontes chaveadas. O sistema de localização deveria utilizar GNSS profissional (u-blox M8/M10), com suporte às principais constelações de satélite, garantindo maior precisão em áreas urbanas.

Na comunicação, recomenda-se a troca do módulo GSM por tecnologias mais modernas como LTE Cat-M1 e NB-IoT, permitindo envio de localização em tempo real para um servidor dedicado. Essa infraestrutura exigiria também um aplicativo próprio (Android/iOS), banco de dados em nuvem e API segura para monitoramento remoto do dispositivo.

O travamento físico utilizaria atuadores automotivos de alta resistência, solenoides ou fechaduras eletromecânicas, enquanto a segurança seria reforçada com sensores profissionais (acelerômetro, giroscópio, Hall, MEMS) e firmware robusto com criptografia, watchdog, OTA e otimização energética. Para alimentação, sugere-se bateria Li-ion/LiPo com BMS e carregamento inteligente.

Para comercialização, o produto precisaria ser certificado pela Anatel, Inmetro e EMC, além de possuir carcaça ABS IP65, conectores selados e manual com QR Code. Em síntese, o projeto pode se tornar um equipamento comercial competitivo ao ser reestruturado em hardware, comunicação, software e mecânica, contribuindo para reduzir furtos e promover uma mobilidade urbana mais segura.

7. TABELA DE PREÇOS PARA UM PRODUTO COMERCIAL

Componente	Descricao / Exemplo	Preco (R\$)	Qtd	Subtotal (R\$)
Microcontrolador / placa	Modulo Wifi Bluetooth ESP32-WROOM-32U IpeX	62,00	1	62,00
GNSS	Modulo GPS GNSS M10 UbloX NMEA 10Hz	128,00	1	128,00
Fonte regulada	TPS5430 + LT1763 + filtros	77,00	1	77,00
Protecoes eletricas	TVS, fusiveis, MOSFETs	30,00	1	30,00
Modulo LTE	SIM7070E	102,00	1	102,00
Sensores	IMU + Hall + MEMS	164,00	1	164,00
Bateria + BMS	Bateria Li-ion 18650 12V 2400mAh	75,00	1	75,00
Travamento fisico	Motor Síncrono Worm Gear 12V	180,00	1	180,00
PCB + Montagem	PCB custom + SMT	100,00	1	100,00
Carcaca	IBS65 ABS selado com vedacao	60,00	1	60,00
Homologacao	ANATEL / EMC – por lote	15000,00	1	15000,00
			TOTAL	15978,00

8. CONCLUSÃO

A pesquisa e o desenvolvimento do sistema proposto evidenciaram a relevância da criação de dispositivos de segurança acessíveis, especialmente diante do crescimento do uso de bicicletas e do aumento de furtos em áreas urbanas. O protótipo desenvolvido, mesmo utilizando componentes de baixo custo, demonstrou desempenho eficiente na simulação de situações de risco, comprovando que soluções simples, quando bem estruturadas, podem gerar impacto real na proteção de bens pessoais. A integração do Arduino Nano aos módulos RFID, GSM e GPS possibilitou a construção de um sistema completo, capaz de realizar travamento automático da bicicleta, rastreamento por coordenadas geográficas e emissão de alertas sonoros em casos de tentativa de acesso indevido.

Os resultados obtidos reforçam que pequenas iniciativas tecnológicas podem contribuir de forma significativa para a conscientização sobre segurança, principalmente quando embasadas por dados reais e indicadores estatísticos que demonstram a urgência do tema. A análise gráfica aplicada ao estudo evidenciou a vulnerabilidade de bicicletas em centros urbanos, justificando a necessidade de novas estratégias preventivas. Dessa forma, o dispositivo desenvolvido demonstrou-se funcional ao mostrar, de forma prática, que medidas preventivas de baixo custo podem reduzir riscos, aumentar a proteção do usuário e promover maior confiança no uso do modal.

Conclui-se, portanto, que o projeto atendeu plenamente aos objetivos propostos, apresentando-se como uma solução viável, didática e tecnicamente aplicável. Para trabalhos futuros, recomenda-se a incorporação de recursos adicionais, como monitoramento em tempo real via aplicativo, atualização contínua de localização, armazenamento de dados em nuvem e implementação de novos sensores, como acelerômetro e detecção de movimento. Espera-se que este estudo sirva como base para pesquisas mais avançadas e estimule o desenvolvimento de tecnologias inovadoras, sustentáveis e acessíveis, contribuindo para a segurança veicular urbana e para a ampliação de soluções inteligentes de proteção.

9. REFERÊNCIAS

ARDUINO. *Arduino Nano (Datasheet)*. 2022. Disponível em: <https://docs.arduino.cc/hardware/nano>. Acesso em: 7 nov. 2025.

CHIP ELETRÔNICA. Módulo Eletrônico Buzzer 5V. Disponível em: <https://chip.com.br/produtos/modulo-eletronico-buzzer-5v>. Acesso em: 8 nov. 2025.

G1. O número de furtos de bicicletas cresce nas capitais brasileiras. Rio de Janeiro, 2025. Disponível em: <https://g1.globo.com>. Acesso em: 6 nov. 2025.

HU INFINITO. *Trava Solenóide Mini 12 V*. Disponível em: <https://www.huinfinito.com.br/motores/1824-trava-solenóide-mini-12v.html>. Acesso em: 7 nov. 2025.

IBGE – INSTITUTO BRASILEIRO DE GEOGRAFIA E ESTATÍSTICA. *Estatísticas de Segurança Pública e Justiça*. Rio de Janeiro: IBGE, 2022. Disponível em: <https://www.ibge.gov.br>. Acesso em: 6 nov. 2025.

IG – ÚLTIMO SEGUNDO. Roubos e furtos de bicicletas nas cidades: estados com mais ocorrências. 2023. Disponível em: <https://ultimosegundo.ig.com.br/brasil/2023-10-26/roubos-furtos-bicicletas-cidades-estados-mais-ocorrencias.html>. Acesso em: 8 nov. 2025.

ISP-RJ – INSTITUTO DE SEGURANÇA PÚBLICA DO RIO DE JANEIRO. *Indicadores de Segurança do Estado do Rio de Janeiro*. Rio de Janeiro: ISP-RJ, 2025. Disponível em: <https://www.isp.rj.gov.br>. Acesso em: 6 nov. 2025.

KINGSTATE ELECTRONICS CORP. *Datasheet KPEG158*. Disponível em: <https://www.kingstate.com.tw/upload/products/15/KPEG158.pdf>. Acesso em: 8 nov. 2025.

MIRANDA, João; CARVALHO, Lucas. *Evaluation of public opinion regarding the changes implemented to encourage bicycle use in the city of São Paulo*. São Paulo: ResearchGate, 2022. Disponível em: <https://www.researchgate.net/publication/358825413>. Acesso em: 6 nov. 2025.

ROBÔCORE. Mini Trava Elétrica Solenóide 12V. Disponível em: <https://www.robocore.net/mini-trava-eletrica-solenóide-12v>. Acesso em: 8 nov. 2025.

ROBÔCORE. Módulo GPS GY-NEO6MV2 com antena. Disponível em: <https://www.robocore.net/sensor-robo/modulo-gps-gy-neo6mv2-com-antena>. Acesso em: 8 nov. 2025.

ROBÔCORE. Módulo GSM GPRS SIM800L. Disponível em: <https://www.robocore.net/modulo-gsm-gprs-sim800l>. Acesso em: 8 nov. 2025.

ROBÔCORE. Placa Arduino Nano V3. Disponível em: <https://www.robocore.net/placa-arduino/placa-nano-v3>. Acesso em: 8 nov. 2025.

SANTOS, Janio Laurentino de Jesus; SANTOS, Luiz Eduardo Pereira Ferreira dos. *Planejamento e mobilidade urbana no Brasil: o uso da bicicleta como uma nova maneira de pensar e construir a cidade*. Rio de Janeiro, v. 19, n. 2, p. 1–20, 2023. Disponível em: <https://www.e-publicacoes.uerj.br/rdc/article/view/52895>. Acesso em: 6 nov. 2025.

SIMCOM. *SIM800L Hardware Design V1.00 / V2.02*. Shenzhen: SIMCom Wireless Solutions, 2013. Disponível em: https://www.makerhero.com/img/files/download/Datasheet_SIM800L.pdf. Acesso em: 8 nov. 2025.

SOUSA, Izolina Margarida de; REIS, João Gilberto Mendes dos; QUEIROZ, Lucas Santos de; FORMIGONI, Alexandre. *Cyclomobility Development in São Paulo City: A Comparison With the Ten Most Bike-Friendly Cities Worldwide*. South American Development Society Journal, v. 10, n. 28, p. 108–130, 2024. Disponível em: <https://www.sadsj.org/index.php/revista/article/view/695>. Acesso em: 6 nov. 2025.

U-BLOX AG. *NEO-6 Data Sheet (GPS.G6-HW-09005-E)*. Thalwil, Suíça, 2011. Disponível em: https://content.u-blox.com/sites/default/files/products/documents/NEO-6_DataSheet_%28GPS.G6-HW-09005%29.pdf. Acesso em: 8 nov. 2025.

UOL – Jornal do Commercio (JC). Ciclistas brasileiros usam a bicicleta mais de cinco vezes por semana e não pedalam mais porque faltam ciclovias, confirma pesquisa.

2025. Disponível em: <https://jc.uol.com.br/colunas/mobilidade/2025/02/10/...> Acesso em: 8 nov. 2025.

VIANA, André; SILVA, Thiago; OLIVEIRA, Beatriz; CAMPOS, Mariana. *A brief study on cycling technologies*. Journal of Transportation and Sustainable Mobility, São Paulo, v. 5, n. 3, p. 101–114, 2023. Disponível em: <https://www.researchgate.net/publication/372245555>. Acesso em: 6 nov. 2025.

APÊNDICE

```
#include <SPI.h>
#include <MFRC522.h>
#include <SoftwareSerial.h>
#include <TinyGPS++.h>
// -----
// RFID / TRAVA / BUZZER
// -----
#define SS_PIN 10
#define RST_PIN 9
#define RELE_PIN 4
#define BUZZER_PIN 8
#define BOTAO_PIN 7 // reservado se quiser usar botão depois
MFRC522 mfrc522(SS_PIN, RST_PIN);
bool estadoSolenoid = false;
// -----
// SIM800L / GPS
// -----
// SIM800L nos pinos 3 (RX do Arduino) e 2 (TX do Arduino)
SoftwareSerial sim800(3, 2);
// GPS NEO-6M nos pinos 6 (RX Arduino) e 5 (TX Arduino)
SoftwareSerial gpsSerial(6, 5);
TinyGPSPPlus gps;
// Intervalo para checar SMS de comando (mais espaçado pra não atrapalhar RFID)
const unsigned long INTERVALO_SMS = 30000UL; // 30s
unsigned long ultimoCheckSMS = 0;
// -----
// PROTÓTIPOS SIM800/GPS
// -----
void inicializaSIM800();
void atualizaGPS();
bool coordenadasValidas(double &lat, double &lon);
void enviaSMSLocalizacaoPara(double lat, double lon, const String &numeroDestino);
void enviaSMSTexto(const String &numeroDestino, const String &texto);
String lerRespostaSIM(unsigned long timeout);
void verificaSMSComando();
// -----
// BUZZER: 1s se cartão NÃO autorizado
// LEMBRANDO: LOW = TOCA, HIGH = SILÊNCIO
// -----
```

```

void buzzerNaoAutorizado() {
  digitalWrite(BUZZER_PIN, HIGH); // garante silêncio antes
  delay(10);
  digitalWrite(BUZZER_PIN, LOW); // TOCA
  delay(1000); // 1 segundo
  digitalWrite(BUZZER_PIN, HIGH); // volta ao silêncio
}
// -----
// SETUP ÚNICO
// -----
void setup() {
  Serial.begin(9600);
  // --- RFID / TRAVA / BUZZER ---
  SPI.begin();
  mfrc522.PCD_Init();
  // Ganho máximo da antena do leitor RFID
  mfrc522.PCD_SetAntennaGain(mfrc522.RxGain_max);
  pinMode(RELE_PIN, OUTPUT);
  digitalWrite(RELE_PIN, HIGH); // trava padrão
  pinMode(BUZZER_PIN, OUTPUT);
  digitalWrite(BUZZER_PIN, HIGH); // BUZZER SILENCIOSO POR PADRÃO
  pinMode(BOTAO_PIN, INPUT_PULLUP);
  // --- SIM800L / GPS ---
  sim800.begin(9600);
  gpsSerial.begin(9600);
  Serial.println(F("Iniciando sistema RFID + GPS + SIM800L..."));
  inicializaSIM800();
  Serial.println(F("Sistema iniciado e travado."));
}
// -----
// LOOP ÚNICO
// -----
void loop() {
  // 1) Atualiza GPS rapidamente (não bloqueia)
  gpsSerial.listen();
  atualizaGPS();
  // 2) PRIMEIRO: tratar RFID (prioridade total)
  if (mfrc522.PICC_IsNewCardPresent() && mfrc522.PICC_ReadCardSerial()) {
    String uid = "";
    for (byte i = 0; i < mfrc522.uid.size; i++) {

```

```

uid += String(mfrc522.uid.uidByte[i] < 0x10 ? "0" : "");
uid += String(mfrc522.uid.uidByte[i], HEX);
}
uid.toUpperCase();
Serial.print("Cartão lido: ");
Serial.println(uid);
// --- CARTÃO AUTORIZADO ---
if (uid == "8644F604") {
    estadoSolenoide = !estadoSolenoide;
    if (estadoSolenoide) {
        digitalWrite(RELE_PIN, LOW); // destrava
        Serial.println("Solenoide destravada");
    } else {
        digitalWrite(RELE_PIN, HIGH); // trava
        Serial.println("Solenoide travada");
    }
    // Garante silêncio sempre que cartão for autorizado
    digitalWrite(BUZZER_PIN, HIGH);
} else {
    // --- CARTÃO NÃO AUTORIZADO ---
    Serial.println("Cartão não autorizado");
    buzzerNaoAutorizado(); // TOCA 1s (LOW)
}
// Finaliza comunicação com o cartão
mfrc522.PICC_HaltA();
mfrc522.PCD_StopCrypto1();
// REINICIALIZA O LEITOR APÓS CADA LEITURA
mfrc522.PCD_Init();
mfrc522.PCD_SetAntennaGain(mfrc522.RxGain_max);
delay(50); // pequeno tempo pra estabilizar
delay(200); // evita leituras duplicadas muito rápidas
}
// 3) Depois: checar SMS de tempos em tempos
if (millis() - ultimoCheckSMS >= INTERVALO_SMS) {
    ultimoCheckSMS = millis();
    sim800.listen();
    verificaSMSComando();
}
}
}

```

```

// -----
// FUNÇÕES SIM800 / GPS
// -----
// Inicializa SIM800L
void inicializaSIM800() {
    sim800.listen();
    sim800.println(F("AT"));
    delay(500);
    Serial.println(F("Enviado: AT"));
    Serial.println(lerRespostaSIM(1000));
    sim800.println(F("AT+CMGF=1")); // modo texto
    delay(500);
    Serial.println(F("Enviado: AT+CMGF=1"));
    Serial.println(lerRespostaSIM(1000));
    // Garante uso da memória do SIM para SMS
    sim800.println(F("AT+CPMS=\"SM\",\"SM\",\"SM\""));
    delay(500);
    Serial.println(F("Enviado: AT+CPMS=\"SM\",\"SM\",\"SM\""));
    Serial.println(lerRespostaSIM(1000));
    Serial.println(F("SIM800L pronto (modo texto)."));
}

// Atualiza o GPS lendo os bytes
void atualizaGPS() {
    while (gpsSerial.available()) {
        gps.encode(gpsSerial.read());
    }
}

// Relaxa a validação (não usa mais age < 5000)
bool coordenadasValidas(double &lat, double &lon) {
    if (gps.location.isValid()) {
        lat = gps.location.lat();
        lon = gps.location.lng();
        Serial.print(F("Coordenadas lidas: "));
        Serial.print(lat, 6);
        Serial.print(F(", "));
        Serial.println(lon, 6);
        Serial.print(F("Age(ms): "));
        Serial.println(gps.location.age());
        return true;
    }
}

```

```

Serial.println(F("gps.location.isValid() == false"));
return false;
}
// Lê tudo que o SIM800L responder por "timeout" ms
String lerRespostaSIM(unsigned long timeout) {
String resp = "";
unsigned long inicio = millis();
while (millis() - inicio < timeout) {
while (sim800.available()) {
char c = sim800.read();
resp += c;
}
}
return resp;
}
// Envia SMS com link do Google Maps para QUALQUER número
void enviaSMSLocalizacaoPara(double lat, double lon, const String &numeroDestino) {
Serial.print(F("Enviando localizacao para: "));
Serial.println(numeroDestino);

sim800.println(F("AT+CMGF=1"));
delay(300);
sim800.print(F("AT+CMGS=\""));
sim800.print(numeroDestino);
sim800.println(F("\r"));
delay(500); // espera '>'
String msg = "Localizacao via GPS:\nLat: ";
msg += String(lat, 6);
msg += "\nLon: ";
msg += String(lon, 6);
msg += "\n\nGoogle Maps:\nhttps://www.google.com/maps?q=";
msg += String(lat, 6);
msg += ",";
msg += String(lon, 6);
sim800.print(msg);
delay(200);
sim800.write(26); // CTRL+Z
delay(4000);
Serial.println(F("SMS de localizacao enviado (ou tentado)."));
}

```

```

// Função simples para enviar texto (para mensagem de erro, por ex.)
void enviaSMSTexto(const String &numeroDestino, const String &texto) {
    sim800.println(F("AT+CMGF=1"));
    delay(300);
    sim800.print(F("AT+CMGS=\"\");
    sim800.print(numeroDestino);
    sim800.println(F("\");
    delay(500);
    sim800.print(texto);
    delay(200);
    sim800.write(26);
    delay(4000);
    Serial.println(F("SMS de texto enviado (ou tentado)."));
}

// Verifica SMS e responde se encontrar "Localizacao"
void verificaSMSComando() {
    // Pede TODOS os SMS armazenados
    sim800.println(F("AT+CMGL=\"ALL\""));
    // Timeout menor pra não ficar tanto tempo travado (1,5 s)
    String resp = lerRespostaSIM(1500);
    Serial.println(F("Resp CMGL (bruta:)"));
    Serial.println(resp);
    if (resp.length() == 0) return;
    int searchPos = 0;
    while (true) {
        // Acha próximo cabeçalho de SMS
        int idx = resp.indexOf("+CMGL:", searchPos);
        if (idx == -1) break;
        // Pega índice numérico da mensagem
        int comma = resp.indexOf(',', idx);
        if (comma == -1) break;
        String idxStr = resp.substring(idx + 6, comma); // depois de "+CMGL:"
        idxStr.trim();
        int msgIndex = idxStr.toInt();
        // Acha fim da linha de cabeçalho
        int headerEnd = resp.indexOf('\n', idx);
        if (headerEnd == -1) break;
        // Cabeçalho em si (pra extrair número depois)
        String header = resp.substring(idx, headerEnd);
    }
}

```

```

header.trim();
// Agora vamos achar o corpo da mensagem
int p = headerEnd + 1;
// Pula \r, \n e linhas vazias
while (p < (int)resp.length()) {
    char c = resp[p];
    if (c == '\r' || c == '\n') {
        p++;
    } else {
        // achamos começo de uma linha não vazia
        break;
    }
}
int bodyStart = p;
// Vai até o fim dessa linha (até \r ou \n)
while (p < (int)resp.length() && resp[p] != '\r' && resp[p] != '\n') {
    p++;
}
int bodyEnd = p;
String body = resp.substring(bodyStart, bodyEnd);
body.trim();
// Extrai número do remetente do header: +CMGL: idx,"REC ...","+55...",...
String numeroRem = "";
int q1 = header.indexOf("\n"); // abre "REC ..."
int q2 = header.indexOf("\n", q1 + 1); // fecha "REC ..."
int q3 = header.indexOf("\n", q2 + 1); // abre número
int q4 = header.indexOf("\n", q3 + 1); // fecha número
if (q3 != -1 && q4 != -1 && q4 > q3) {
    numeroRem = header.substring(q3 + 1, q4);
}
Serial.print(F("SMS idx "));
Serial.print(msgIndex);
Serial.print(F(" de "));
Serial.print(numeroRem);
Serial.print(F(" body: "));
Serial.println(body);
// Só aceita exatamente "Localizacao"
if (body == "Localizacao") {
    Serial.println(F("Comando Localizacao recebido, atualizando GPS por 1s..."));
    gpsSerial.listen();
}

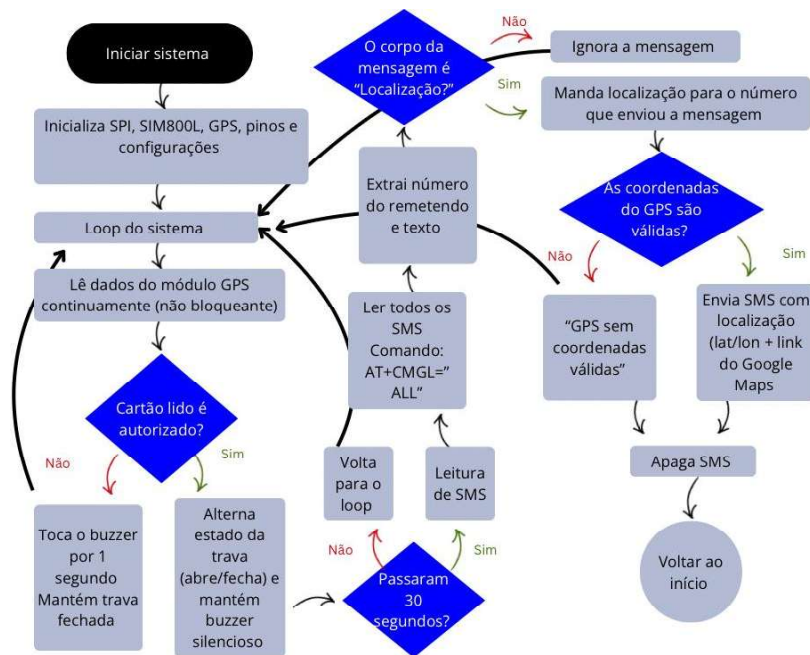
```

```

unsigned long t0 = millis();
while (millis() - t0 < 1000) {
  atualizaGPS();
}
sim800.listen();
double lat, lon;
if (coordenadasValidas(lat, lon) {
  enviaSMSLocalizacaoPara(lat, lon, numeroRem);
} else {
  enviaSMSTexto(numeroRem, "GPS sem coordenadas validas, tente novamente.");
}
}
// Apaga SMS processado pra não repetir
sim800.print(F("AT+CMGD="));
sim800.println(msgIndex);
lerRespostaSIM(500); // também curto
// Continua procurando a partir do fim do corpo
searchPos = p;
}
}

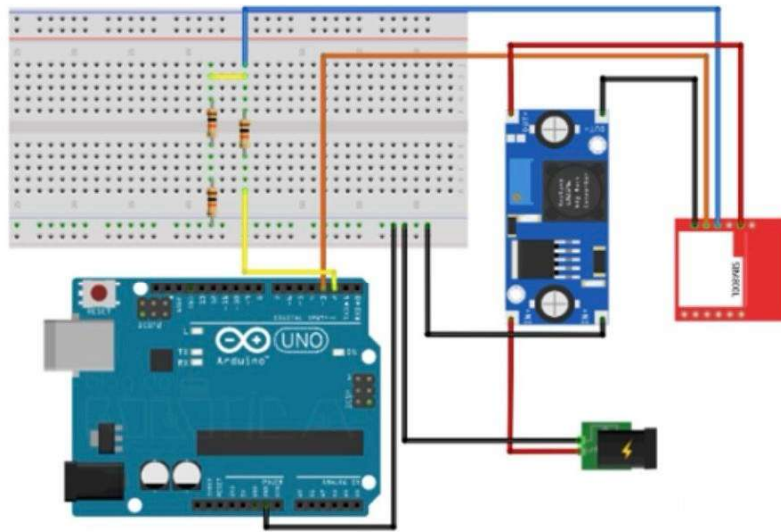
```

Imagem 15 - Fluxograma de Funcionamento do Código



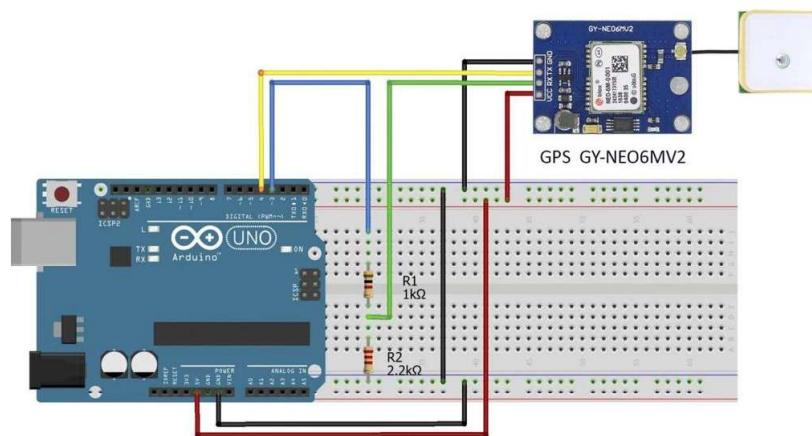
Fonte: Foto de Autoria própria- 27/11/2025

Imagem 16 - Esquema Elétrico Módulo GSM GPRS SIM800L



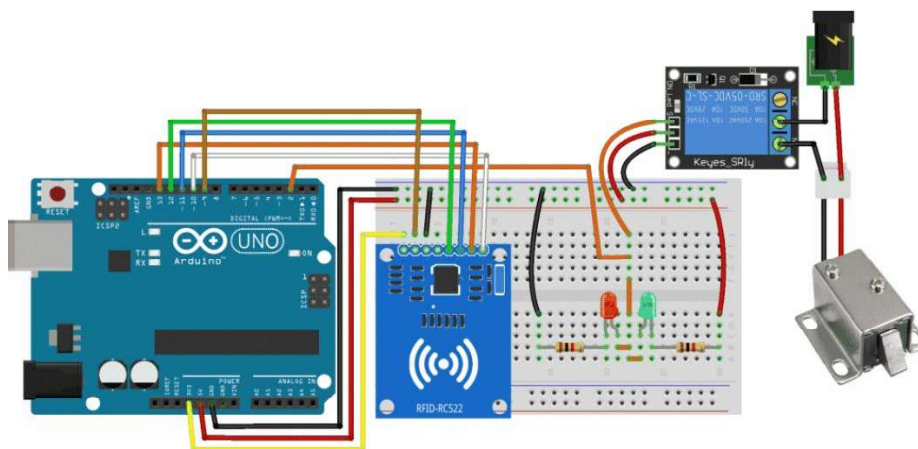
Fonte: blogdarobotica.com/como-enviar-sms-para-celular-utilizando-um-sim800l

Imagem 17 - Esquema Elétrico Módulo GPS GY-NEO6MV2 com Antena



Fonte: blog.eletrogate.com/gps-neo-6m-com-arduino

Imagem 18 - Esquema Elétrico Mini Trava Solenóide e Módulo RFID



Fonte: makerhero.com/blog/acionando-trava-eletrica-com-rfid

Devido à ausência de componentes como o módulo GPS com antena, o módulo SIM800L e o módulo RFID, foram utilizadas referências externas para compreender e simular o funcionamento de cada etapa do projeto.