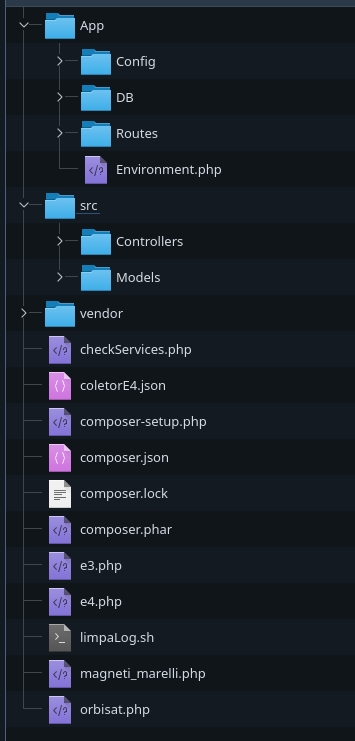
## Coletor de Dados GPS IdTracker

Esse manual tem como objetivo explicar o máximo possível a estrutura de funcionamento, o fluxo de dados e as lógicas aplicadas ao Coletor de Dados GPS IdTracker.

### Estrutura do Sistema



Acima podemos ver a estrutura de pastas do sistema Coletor, na pasta src ficam os controllers e models para manipulação dos dados recebidos e persistência dos mesmo em banco de dados. A pasta App é responsável por manter os arquivos de configuração e constantes do sistema.

### Lógica e fluxo de dados

Como o sistema é multiplataforma, ou seja, coleta dados de 4 modelos diferentes de rastreador a lógica aplicada foi de criar um subsistema para cada tipo de rastreador.

Na pasta raiz temos os arquivos de entrada e3.php, e4.php, orbisat.php e magneti\_marelli.php, cada um deles deve ser iniciado por um gestor de processos para que fiquem em execução sem interrupção aguardando dados vindos dos rastreadores. No desenvolvimento e mesmo em produção foi utilizado o pm2 como gerenciador de processos e o mesmo teve um resultado muito bom.

Os arquivos de entrada nada mais fazem que carregar algumas configurações e classes de Log para utilização no sistema, no final criam uma instância da Classe criada para cada rastreador e chamam o método runSocket que é quem vai de fato criar o método de escuta e aguardar os dados.

Os arquivos de classe para cada coletor ficam dentro do diretório /src/Controllers/ e existe uma classe para cada rastreador como já dito, exemplo: E3Sockets.php, etc…

Dentro dos arquivos de classe a lógica é a mesma, criar uma escuta em uma porta pré determina no arquivo /App/Config/Config.php onde cada rastreador escuta uma porta distinta.

Sempre que novos dados chegam dos rastreadores é feito um tratamento inicial e verificado se o rastreador que tenta a comunicação é um rastreador válido, registrado na plataforma. Estando tudo ok o sistema da continuidade no processamento dos dados recebidos.

Dentro da classe de Socket é seguida uma lógica de checagem de identidade, checagem de comandos para execução, identificação do tipo da mensagem recebida, necessidade de resposta ou não, etc… tudo que o manual de cada rastreador pede é executado ali.

Depois de todas as checagens e separação dos dados recebidos é feita a chamada do método que envia os dados para o Translate do sistema, esse Translate traduz as informações recebidas muitas vezes em hexadecimal.

Na classe Translate tudo é separado, processado, traduzido e armazenado para utilização, a partir desse processamento é identificado o tipo de mensagem e chamado o método adequado para persistência no banco de dados.

Para serem armazenadas no banco de dados as coordenadas precisam atender a alguns requisitos mínimos de confiabilidade que são:

* Ter um mínimo de satélites GPS conectados para garantir uma precisão adequada;
* Estar no tamanho e formato correto de cada protocolo de transmissão de dados;
* Ser uma mensagem enviada em um timestamp antes da hora do recebimento.

Estando tudo certo, o método de persistência no banco de dados é chamado e só então os dados são armazenados.

### Envio de comandos aos Rastreadores

A cada comunicação recebida o sistema verifica se existe algum tipo de comando a ser enviado para o rastreador.

No caso de positivo dentro da classe de Socket antes mesmo do envio da mensagem para processamento e armazenagem, é feito o envio dos comandos.

Foi adotada essa prioridade para garantir que a comunicação não fosse prejudicada ou perdida durante o processamento do sistema, isso garante que caso exista um comando de bloqueio de veículo ou ativação de pânico esse comando seja enviado antes de qualquer coisa.

Em nosso entendimento é vital em primeira instância a comunicação da plataforma com o rastreador tão logo esse tente se comunicar com o coletor.

Todos os comandos ficam armazenados em banco de dados e tem os detalhes de cada comando anotados, bem como os horários de inserção do comando no banco de dados e de execução do mesmo.

### Processamento de Status

Todos os modelos homologados no coletor de dados de gps IdTracker enviam informações referentes ao seu funcionamento e possíveis problemas a serem considerados, são as mensagens de Status.

Essas mensagens também são interpretadas e processadas pelo coletor e aparecem em forma de notificações tanto no App Mobile quanto na interface Web.

Diferente dos comandos, as mensagens de status são as últimas a serem processadas pelo coletor. Por terem um carácter informativo não existe prejuízo em sua análise mais tardia.

As mensagens de status ficam exibidas até que sejam marcadas como lida pelo usuário e dessa forma são marcadas para não serem mais exibidas.