

# Gerenciamento de Transporte Escolar Público: Sistema SYS STADIA

Bruno Assunção, Gabriel dos Santos Motroni, Gledson da Silva, Matheus Marques Silva,  
Priscila Da Dalt

Orientador: Professor Esp. Victor Williams Stafusa da Silva

Faculdade Impacta de Tecnologia  
São Paulo, SP, Brasil  
15 de junho de 2021

**Resumo.** O projeto se inicia com a demanda de fazer o controle de gerenciamento de transporte escolar da E. E. Prof<sup>a</sup> Rita Pinto de Araujo, onde se via necessário informatizar todo o processo. Iniciou com a realização do levantamento das necessidades e entendimento de qual seria a melhor solução. Após isso, foi iniciado o mapeamento dos requisitos e qual seria a melhor solução para entregar um programa que pudesse fazer todo esse processo. Ao evoluir com os levantamentos e desenvolvimento das aulas, optou-se por fazer um sistema desenvolvido em Python na plataforma Flask e com banco de dados MariaDB, todo em *web* utilizando o HTML e CSS. Durante todo o desenvolvimento foi utilizado o Git para controle de versão e o GitHub para hospedagem do conteúdo. No final do projeto, toda a aplicação foi implantada na nuvem no Heroku para operação do aplicativo.

**Palavras-chaves:** gerenciamento de transporte, desenvolvimento, processo.

## 1. Introdução

A solução desenvolvida visa resolver o problema na gestão de transporte escolar gratuito em todas as suas etapas, desde a solicitação até a geração de relatórios gerencial. Todo o sistema foi desenvolvido em linguagem *web* para facilitar o acesso de qualquer lugar e sendo de simples utilização e com ferramentas de gerenciamento das demandas.

### 1.1. Apresentação da empresa E.E. Prof.<sup>a</sup> Rita Pinto de Araújo

Fundada em 1976 sob Decreto nº 7.517, de 03 de fevereiro de 1976 sob o nome Escola Estadual de 1º Grau da Adutora Rio Claro hoje chamada E.E. Prof.<sup>a</sup> Rita Pinto de Araújo, jurisdicionada a Diretoria de Ensino Leste 3, atualmente atende o segmento ensino fundamental anos iniciais mas já atendeu também o EJA, situada no bairro 3ª divisão, local em que existem inúmeras invasões de terrenos e construções de barracos em áreas ilegais, tornando a população de seu entorno carente e com recursos precários, possui inúmeras ruas sem calçadas, ruas de barro, poucos locais de instrução e entretenimento, fazendo da escola um pólo central de socialização da comunidade.

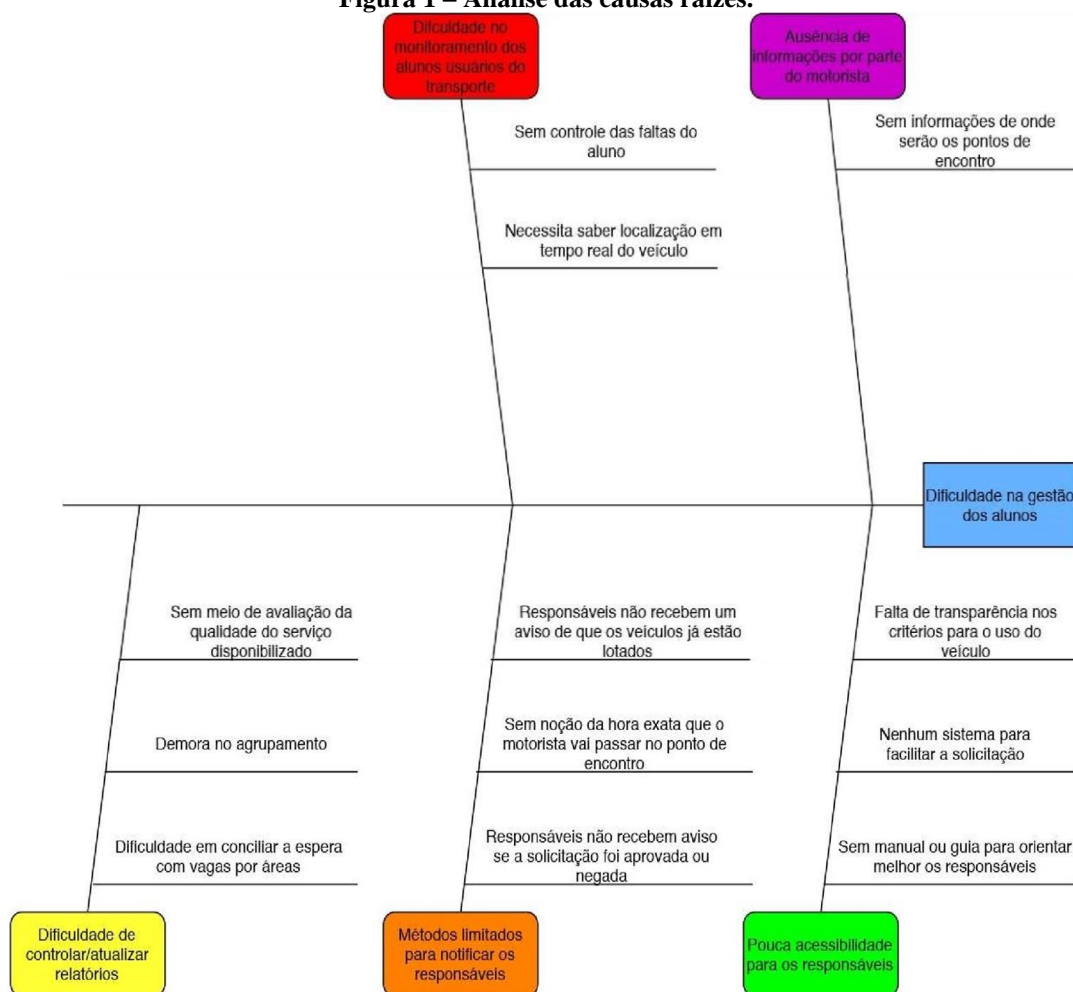
### 1.2. O problema da gestão do transporte escolar

O problema da dificuldade na gestão do transporte escolar dos alunos, afeta alunos, responsável pelo aluno, funcionários da secretaria, motoristas, monitores, fiscal e diretor do núcleo de compras, devido demora para disponibilização de vaga, deslocamento de casa até a escola a pé ou pagando transporte particular, grande número de crianças para gerenciar, os endereços residenciais dos alunos serem irregulares dificultando sua identificação no mapa, dificuldade em montar pontos de encontro, conciliação de diferentes desejos dos pais x

motoristas em relação a escola, dificuldade de parâmetros para alocar ou não os alunos nas vagas.

Diante dos problemas apresentados faz-se necessário um sistema que gerencie efetivamente o grande número de alunos eleitos ou não ao transporte, visando atendimento de legislação específica, fornecendo relatórios rapidamente aos gestores escolares de grupos de alunos de determinado bairro, rua, idade, etc.; propiciar a seus usuários uma forma de obter facilmente informações de quantidades de vagas em cada ônibus, critérios para inclusão ou não dos interessados em tal transporte, saber em qual ônibus alocar qual estudante de qual endereço e, também, que o sistema forneça para impressão um documento de autorização para ser entregue aos novos alunos habilitados.

**Figura 1 – Análise das causas raízes.**



Fonte: Os autores.

### 1.3. Stakeholders e restrições

**Quadro 1 – Stakeholders.**

Usuários	Descrição
Diretor do núcleo de compras	Usará o sistema SYS_STADIA, para: <ul style="list-style-type: none"> <li>• Gerar relatórios de quantidade de alunos atendidos por trajeto (descritivo).</li> <li>• Gerar relatórios de quantidade de alunos na fila de espera.</li> </ul>

Usuários	Descrição
Diretor da escola	Usará o sistema SYS_STADIA, para: <ul style="list-style-type: none"> <li>• Gerar relatório de quantidade de motoristas, monitores e veículos e suas identificações.</li> <li>• Cadastrar ocorrências e seu status.</li> <li>• Gerar relatório de quantidade de ocorrências por trajetos.</li> <li>• Cadastrar atrasos na entrada ou saída por parte dos alunos ou motorista.</li> <li>• Gerar relatório de atrasos por parte dos alunos ou motoristas.</li> </ul>
Funcionário da secretaria	Usará o sistema SYS_STADIA, para: <ul style="list-style-type: none"> <li>• Cadastrar pedidos de transporte feito pelos responsáveis.</li> <li>• Cadastrar status do pedido de transporte.</li> <li>• Cadastrar quantidade e tipo de veículos.</li> <li>• Cadastrar motoristas e monitores de cada veículo.</li> <li>• Gerar lista de alunos já transportados para entrega aos motoristas.</li> <li>• Visualizar número de vagas ocupadas ou não por trajeto e turno de ensino.</li> <li>• Cadastrar aluno homologado na vaga.</li> <li>• Cadastrar aluno na fila de espera na ausência de vaga.</li> <li>• Monitorar chegada do veículo na escola/residência do aluno.</li> </ul>

Outros Stakeholders	Descrição
Monitor	<ul style="list-style-type: none"> <li>• Usará lista de alunos para registrar a presença no transporte.</li> <li>• Usará lista de alunos do seu veículo com informação de turno, ano, turma, endereço.</li> <li>• Solicitará registro de ocorrências no veículo.</li> <li>• Consultará calendário de dias que deve haver transporte.</li> </ul>
Motorista	<ul style="list-style-type: none"> <li>• Verificará relatório de quantidade de alunos por endereço para definição do ponto de encontro e melhor rota.</li> </ul>
Responsável pelo aluno	<ul style="list-style-type: none"> <li>• Acompanhará trajeto do aluno em tempo real.</li> <li>• Verificará andamento de seu pedido.</li> <li>• Verificará em qual veículo, motorista e monitor seu filho foi atendido.</li> <li>• Consultará carteirinha do aluno no transporte.</li> </ul>
Equipe de desenvolvimento	<ul style="list-style-type: none"> <li>• Modela e cria sistema para gerenciamento do transporte.</li> </ul>
Fiscal	<ul style="list-style-type: none"> <li>• Verificará quantidade de alunos na fila de espera para disponibilizar mais veículos para a escola desde que já autorizados em contrato.</li> <li>• Verificará quantidade de alunos por trajeto.</li> <li>• Consultará calendário de dias que deve fornecer transporte.</li> </ul>

## 2. Solução proposta

A solução apresentada foi o desenvolvimento de uma solução *web*, onde será possível fazer toda gestão do transporte escolar desde a locação dos lugares até a gestão dos carros e motorista. Também terá uma visão gerencial de toda demanda e análise de localidade e onde há mais carência e vagas no transporte.

## 2.1. Requisitos do sistema

Quadro 2 – Requisitos.

ID	REQUISITO
SSS – 01	O Sistema DEVE registrar as solicitações de transporte.
SSS – 02	O Sistema DEVE incluir as barreiras já homologadas nos trajetos.
SSS – 03	O Sistema DEVE registrar os trajetos dos transportes, com ponto de encontro de no mínimo 5 alunos.
SSS – 04	O Sistema DEVE registrar as vagas dos veículos por trajetos.
SSS – 05	O Sistema DEVE registrar o status da solicitação.
SSS – 06	O Sistema DEVE consultar as solicitações já efetuadas.
SSS – 07	O Sistema DEVE verificar se aluno atende os requisitos previamente cadastrados.
SSS – 08	O Sistema DEVE registrar aluno já homologado na fila de espera, caso não haja vaga no trajeto.
SSS – 09	O Sistema DEVE priorizar alunos PcD tanto na fila de espera quanto nas solicitações.
SSS – 10	O Sistema DEVE registrar os períodos manhã ou tarde e ano letivo na solicitação.
SSS – 11	O Sistema DEVE registrar os tipos de veículos ônibus, microônibus e van, e nome da empresa e data de início.
SSS – 12	O Sistema DEVE registrar os alunos que possuem bilhete único ativo.
SSS – 13	O Sistema DEVE emitir aviso se excedida idade máxima do aluno de 12 anos.
SSS – 14	O Sistema DEVE registrar dados do veículo: nome do motorista, nome da monitora, turno que atende, placa.
SSS – 15	O Sistema DEVE emitir lista de alunos com nome do motorista, nome da monitora, tipo de veículo, turno, trajeto, nº de vagas, nome do aluno, R.A.
SSS – 16	O Sistema DEVE registrar nome do aluno, R.A., ano/turma, PcD, <i>internet</i> .
SSS – 17	O Sistema DEVE registrar ocorrências com nome do requerente, aluno envolvido, motorista envolvido, monitor envolvido, dia da ocorrência, turno, breve relato da ocorrência, <i>status</i> , parecer da direção.
SSS – 18	O Sistema DEVE emitir relatório de ocorrências com filtro por data, trajeto, motorista, monitora, aluno, <i>status</i> .
SSS – 19	O Sistema DEVE emitir ficha da ocorrência para assinatura dos envolvidos.
SSS – 20	O Sistema DEVE possuir acessibilidade na interface com o usuário.
SSS – 21	O Sistema DEVE permitir acompanhamento do percurso do aluno em tempo real.
SSS – 22	O Sistema DEVE permitir agrupar alunos por endereço.
SSS – 23	O Sistema DEVE permitir cadastrar calendário com dias que haverá o serviço de transporte.
SSS – 24	O Sistema DEVE permitir cancelamento da solicitação a qualquer tempo.
SSS – 25	O Sistema DEVE controlar as alterações de endereço emitindo alerta no <i>dashboard</i> .
SSS – 26	O Sistema DEVE possuir <i>dashboard</i> na tela inicial com as últimas notificações.
SSS – 27	O Sistema DEVE emitir carteirinha para uso do aluno no transporte.
SSS – 28	O Sistema DEVE emitir lista de alunos por trajeto para monitor fazer a chamada diariamente.
SSS – 29	O Sistema DEVE registrar atrasos do transporte na entrada/saída da escola.
SSS – 30	O Sistema DEVE permitir importar/exportar dados.

Fonte: Os autores.

## 2.2. Regras de negócio

Quadro 3 – Regras de negócio.

Regra	Descrição
RN 01	Secretaria Escolar Digital envia detalhes da ficha do aluno. Envio imediato mediante acesso a SED.
RN 02	SPTrans envia Detalhes do Cadastro de Bilhete Único. Envio imediato mediante acesso a SPTrans.
RN 03	Diretoria de Ensino envia Resposta da Homologação. A homologação tem que ser feita com no máximo 30 dias antes do início do ano letivo.
RN 04	Responsável pelo aluno solicita cancelamento do pedido. É possível cancelar a qualquer tempo, porém deve requerer por escrito.
RN 05	Funcionário da Secretaria solicita informações para análise. Somente se houver pedido de transporte pelo responsável.
RN 06	Diretoria de Ensino não envia resposta de Homologação. Extemporaneamente o funcionário da secretaria cobrará resposta da homologação.
RN 07	Funcionário da secretaria envia lista de alunos para motorista. Todo início de ano letivo o funcionário deve enviar a lista de alunos ao motorista, após 6 meses enviar outra lista atualizada.
RN 08	Funcionário da secretaria recebe os dados referente a quantidade de lugares disponíveis. Após finalizada a licitação por parte da diretoria de ensino, ela envia detalhes para a escola referente a quantidade de veículos disponíveis.
RN 09	Funcionário da secretaria envia lista para diretoria de ensino. Deve ocorrer antes do início da licitação por parte da diretoria de ensino.

Fonte: Os autores.

## 2.3. Descrição das funcionalidades

O sistema foi desenvolvido em 6 abas principal onde cada uma delas tem seus *sublinks*. Todas as telas onde será informado o CEP, já está integrado por uma API com os correios e assim já busca as informações do endereço e bairro direto na base dos correios, abaixo os *links* principais do sistema:

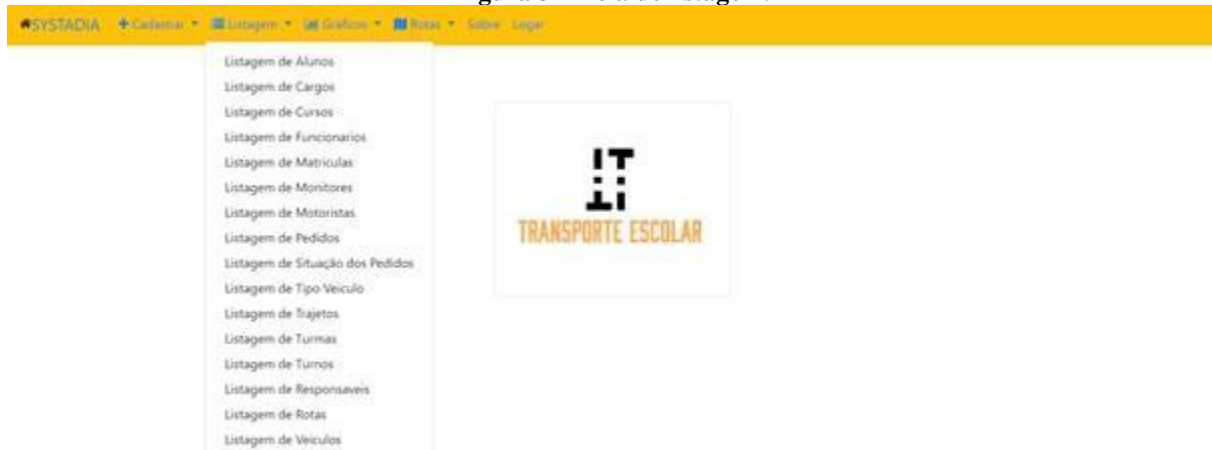
Cadastrar – Nessa sessão será onde irá ser realizado os cadastros do sistema, tais como aluno, curso, cargo, funcionário, matrícula, motorista, monitor, pedido, situação do pedido, trajeto, tipo veículo, turmas, turnos, responsável, rota e veículo

**Figura 2 – Tela cadastrar.**



**Fonte: Os autores.**

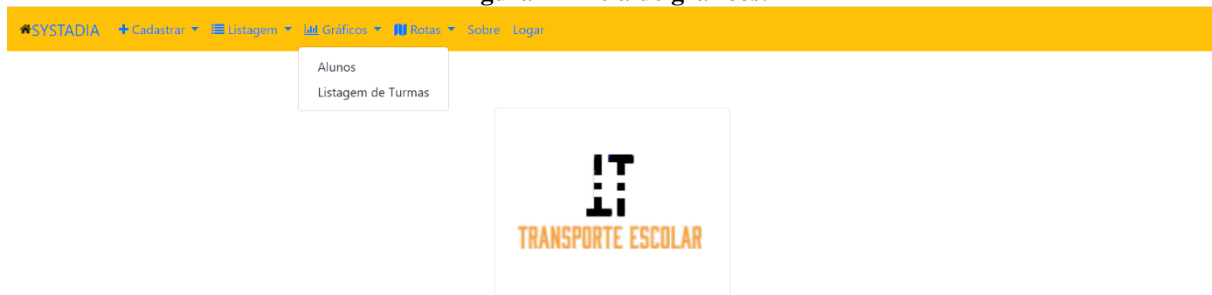
**Figura 3 – Tela de listagem.**



**Fonte: Os autores.**

Listagem – Nessa sessão temos todas as listagens dos cadastros já feitos no sistema, onde temos as listagens de alunos, cargos, cursos, funcionários, matrículas, monitores, motoristas, pedidos, situação de pedidos, tipo veículo, trajetos, turmas, turnos, responsáveis, rotas e veículos.

**Figura 4 – Tela de gráficos.**



**Fonte: Os autores.**

Gráficos – Tela onde será possível gerar gráficos para auxiliar na análise e administração dos pedidos de transporte escolar, onde temos os gráficos dos alunos e da listagem de turmas.

### 2.3.1. Login

Figura 5 – Tela de login.



IT  
TRANSPORTE ESCOLAR

Email Institucional  
@educacao.sp.gov.br

Senha  
utilize a mesma senha da SED

☐ Lembrar de mim (7 dias)

Entrar

Esqueci Minha Senha

Fonte: Os autores.

Nessa tela será onde o usuário irá colocar suas credenciais para acessar como o sistema como “supervisor” e ter acesso as alterações. Os campos para validar as credenciais serão o *e-mail* como usuário e sua senha. Também há o campo “esqueci minha senha” caso o usuário não lembre mais suas credenciais. Os erros que podem acontecer seria de esquecer as credenciais de *e-mail* ou de senha.

### 2.3.2. Cadastro de alunos

Nessa tela será onde será cadastrado todas as informações do aluno. Nela, há os campos nome, R.A., CEP, endereço, bairro, sexo, data de nascimento e CPF do aluno. O campo CEP está integrado com API dos correios e após informar os dados já preenche as informações do endereço e bairro automaticamente. Todos os campos são obrigatórios e nessa tela também será vinculado o responsável do aluno, após ser cadastrado na tela responsável.

**Figura 6 – Cadastrar aluno.**

## Cadastrar Aluno

Nome:

RA:

CEP:

Endereço:

Bairro:

Sexo:

Sexo: Masculino

Data de Nascimento:

dd/mm/aaaa

CPF do Aluno

**Fonte: Os autores.**

### 2.3.3. Cadastro responsável

Nessa tela serão cadastradas as informações do responsável pelo aluno, com os campos de nome, telefone principal, telefone secundário, *e-mail*, senha, usuário, CEP, endereço e bairro. Aqui será criado o usuário e senha para que o responsável acompanhe a solicitação de transporte escolar gratuito.



**Figura 7 – Cadastrar responsável.**

## Cadastrar Responsavel

Nome:

Telefone Principal:

Telefone Secundário:

Email:

Senha:

Usuario:

CEP:

Endereço:

Bairro:

Fonte: Os autores.

### 2.3.4. Cadastrar pedido

Nesta tela é onde cadastra-se o pedido do transporte escolar, onde existem os campos de seleção do aluno, do responsável do aluno solicitante, da rota e a descrição. Somente no campo “descrição” é permitido o preenchimento e os outros campos já irá trazer as informações de outras telas.

**Figura 8 – Cadastrar pedido.**

■ SYSTADIA   ■ Cadastrar   ■ Listagem   ■ Gráficos   ■ Rota   ■ Sobre   ■ admin

### Cadastrar Pedido

Selecione o Aluno:

Responsável do Aluno Solicitante:

Selecione a Rota:

Descrição:

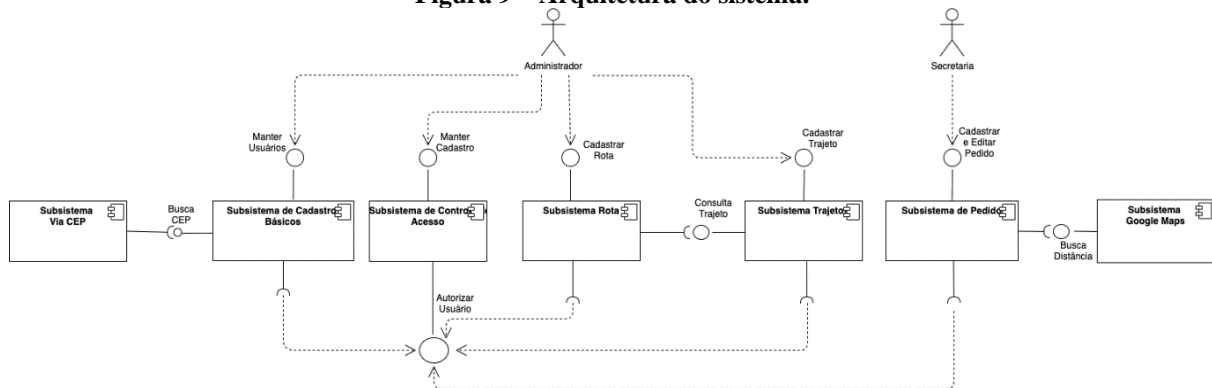
Fonte: Os autores.

## 3. Projeto, análise e implementação

A solução foi desenvolvida em equipe usando GitHub para controle de versionamento onde todos os membros se basearam inicialmente no código fonte de CRUD base para desenvolver os demais módulos para posteriormente integrar ao código-fonte novamente. Atualmente tanto a aplicação como o banco de dados estão hospedados na plataforma em nuvem Heroku. O projeto foi desenvolvido para *web* utilizando framework Flask no *back-end*

devido à baixa curva de aprendizagem e conhecimento comum entre os membros desenvolvedores do grupo, assim como Bootstrap no *front-end* e MariaDB como SGBD.

**Figura 9 – Arquitetura do sistema.**



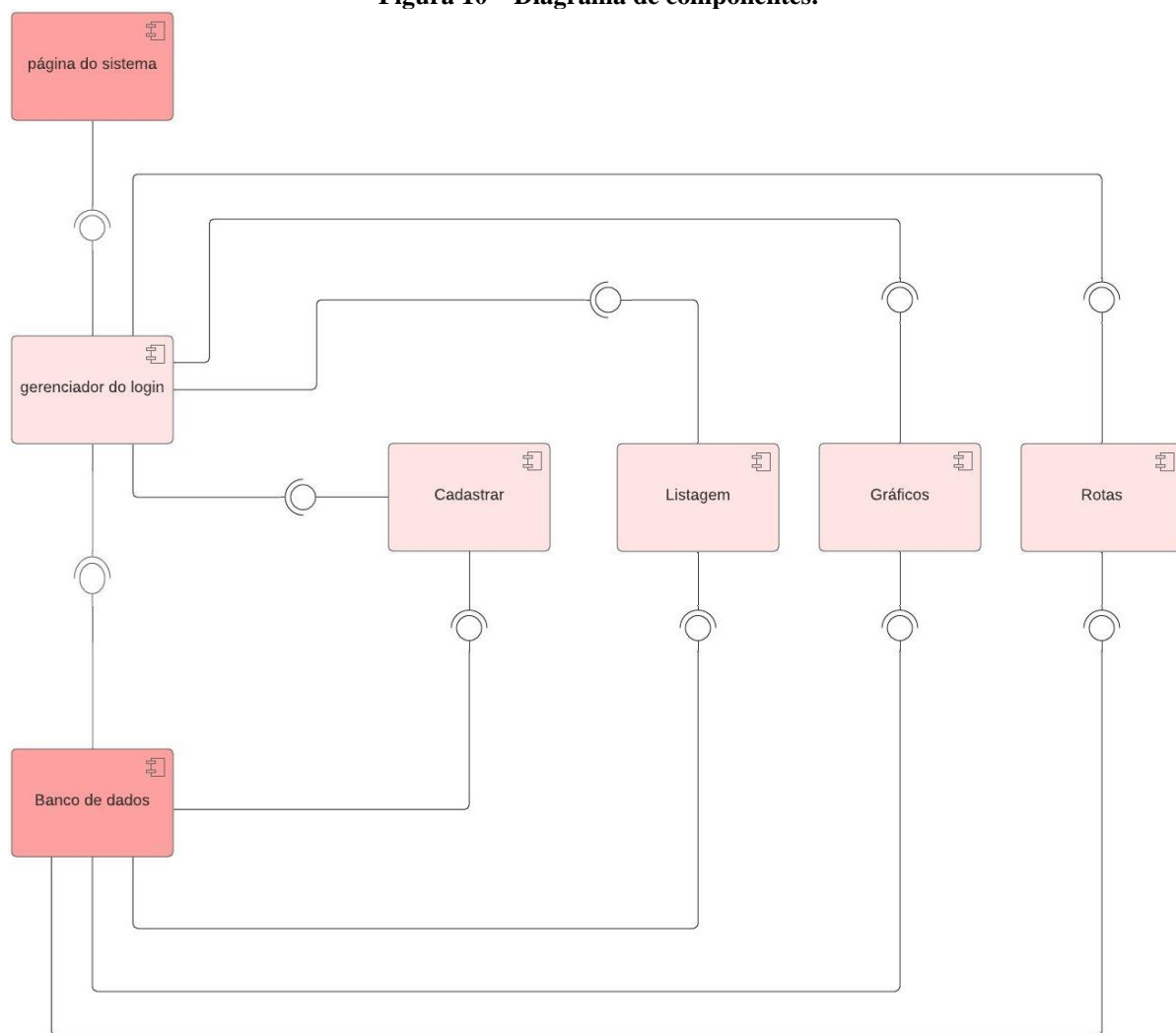
**Fonte: Os autores.**

### 3.1. Arquitetura, módulos e subsistemas

No desenvolvimento do sistema usamos para o *front-end* as tecnologias web HTML, CSS e Bootstrap, já no *back-end* usamos como base a linguagem Python juntamente com o Flask e para gerenciamento de banco de dados o MariaDB. Utilizando o Flask como *framework* desenvolvemos todos os requisitos HTML baseada nessa aplicação.

O banco de dados está hospedado no Heroku juntamente com a aplicação e assim conseguimos um custo baixo e com alta disponibilidade e performance. Com um sistema baseado na linguagem *web* conseguimos abrir em qualquer navegador disponível no mercado e assim atender a demanda necessária para a fácil navegação.

**Figura 10 – Diagrama de componentes.**



**Fonte: Os autores.**

### **3.1.1. Subsistema de cadastros básicos**

Cadastrar dados do aluno, responsável, motorista, monitor, fiscal, rotas e empresa de transporte como por exemplo: nome, número de identificação, endereço, telefones, ano letivo do cadastro, ano/turma do aluno; dados estes que serão utilizados para posteriormente abrir solicitação de vaga no transporte.

### **3.1.2. Subsistema de solicitação de transporte**

Registrar manifestação de interesse em usar o transporte. A solicitação é feita com os dados do aluno e responsável já cadastrado.

### **3.1.3. Controle de acesso**

Responsável por gerir as contas mantidas pela instituição escolar em relação ao sistema, realizando perfilamento dos usuários.

### 3.1.4. Subsistema de registro de ocorrências

Registram formalmente as ocorrências manifestadas pelos responsáveis dos alunos e motoristas ocorridas durante alguma viagem do transporte.

### 3.1.5. ViaCEP

Busca CEP informado no momento da solicitação de transporte e nas telas de cadastro de endereço.

### 3.1.6. Google Maps Distance Matrix

Busca e retorna a distância em quilômetros do CEP do aluno em contraste ao CEP da escola.

## 3.2. Projeto de banco de dados

O banco de dados utilizado foi o MariaDB, por ser um banco de dados gratuito e de simples implementação. Nele, existem as seguintes tabelas implementadas para atender os requisitos do sistema:

**Funcionário** – Tabela onde será cadastrado os funcionários da escola.

```
(
codigo_funcionario INT AUTO_INCREMENT PRIMARY KEY,
nome varchar (30),
codigo_cargo int null,
codigo_turno int null,
usuario varchar (40) not null,
senha varchar (40) not null,
email varchar (60) not null,
CONSTRAINT fk_pk_cargo FOREIGN KEY (codigo_cargo) REFERENCES Cargo
(codigo_cargo)
)
```

**Cargo** – Tabela onde será cadastrado os cargos dos funcionários.

```
(
codigo_cargo INT AUTO_INCREMENT PRIMARY KEY
nome varchar (30)
departamento varchar (30) not null
)
```

**Responsavel** – Tabela onde serão cadastrados o responsável pelo aluno.

```
(
codigo_responsavel INT AUTO_INCREMENT PRIMARY KEY,
nome varchar (60) not null,
telefone_principal varchar (12),
telefone_secundario varchar (12),
codigo_aluno int not null,
email varchar (60),
usuario varchar (60) not null,
senha varchar (60) not null,
endereco varchar (50) not null,
bairro varchar (50) not null,
cep char (8) not null,
CONSTRAINT fk_pk_aluno_responsavel FOREIGN KEY (codigo_aluno) REFERENCES
Aluno(codigo_aluno)
)
```

)

**Turma** – Tabela onde será cadastrado as turmas.

```
(
codigo_turma INT AUTO_INCREMENT PRIMARY KEY
codigo_curso int not null
nome varchar (50) not null
codigo_turno int not null
sigla char (5) not null
ano_letivo smallint not null
fk_pk_turma_curso FOREIGN KEY (codigo_curso)
)
```

**Curso** – Tabela onde será cadastrado os cursos.

```
(
codigo_curso INT AUTO_INCREMENT PRIMARY KEY
descricao varchar (30)
)
```

**Aluno** – Tabela onde será cadastrado os alunos e suas informações.

```
(
codigo_aluno INT AUTO_INCREMENT PRIMARY KEY,
codigo_situacao_aluno smallint not null,
ra_aluno varchar (10),
nome varchar (50) not null,
endereco varchar (255) not null,
bairro varchar (50) not null,
cep char (8) not null,
sexo char (1) not null,
CONSTRAINT fk_pk_Aluno_situacao_aluno FOREIGN KEY (codigo_situacao_aluno)
)
```

**SituacaoAluno** – Tabela onde será cadastrado se o aluno está ativo ou inativo.

```
(
codigo_situacao_aluno smallint PRIMARY KEY,
descricao_situacao_aluno varchar (14) not null
)
```

**Matricula** – Tabela onde será cadastrado os alunos matriculados.

```
(
codigo_matricula INT AUTO_INCREMENT PRIMARY KEY
codigo_aluno int not null
codigo_turma int not null
codigo_situacao_matricula int not null
ano_matricula smallint,
CONSTRAINT fk_pk_matricula_turma FOREIGN KEY (codigo_turma) REFERENCES
Turma(codigo_turma)
CONSTRAINT fk_pk_matricula_aluno FOREIGN KEY (codigo_aluno) REFERENCES
Aluno(codigo_aluno)
CONSTRAINT fk_pk_matricula_situacao_matricula FOREIGN KEY
(codigo_situacao_matricula) REFERENCES
SituacaoMatricula(codigo_situacao_matricula)
)
```

**SituacaoMatricula** – Tabela onde será cadastrado se o aluno está ativo, inativo ou transferido.

```
(
codigo_situacao_matricula int primary key,
descricao_situacao varchar (14) not null
)
```

**Turno** – Tabela onde será cadastrado o turno da turma.

```
(
codigo_turno TINYINT AUTO_INCREMENT PRIMARY KEY,
descricao char (5) not null
)
```

**FrequenciaAluno** – Tabela onde será cadastrada a frequência do aluno na utilização do transporte escolar.

```
(
codigo_frequencia INT AUTO_INCREMENT PRIMARY KEY,
codigo_matricula int not null,
data_presenca date not null,
codigo_turma int not null,
codigo_tipo_frequencia int not null,
CONSTRAINT fk_pk_frequencia_aluno_matricula FOREIGN KEY (codigo_matricula)
REFERENCES Matricula(codigo_matricula),
CONSTRAINT fk_pk_turma_frequencia_aluno FOREIGN KEY (codigo_turma)
REFERENCES Turma(codigo_turma),
CONSTRAINT fk_pk_tipoFrenquecia FOREIGN KEY (codigo_tipo_frequencia)
REFERENCES TipoFrenquecia(codigo_tipo_frequencia)
)
```

**TipoFrequencia** – Tabela onde será cadastrado o tipo da frequência que será utilizada na tabela “FrequenciaAluno”.

```
(
codigo_tipo_frequencia INT AUTO_INCREMENT PRIMARY KEY,
descricao CHAR (2)
)
```

**Pedido** – Tabela onde será cadastrado o pedido do transporte escolar.

```
(
codigo_pedido INT AUTO_INCREMENT PRIMARY KEY,
codigo_aluno int not null,
codigo_rota int not null,
codigo_situacao_pedido tinyint,
codigo_responsavel int not null,
abertura_data_hora datetime not null,
CONSTRAINT fk_pk_aluno_pedido FOREIGN KEY (codigo_aluno) REFERENCES
Aluno(codigo_aluno),
CONSTRAINT fk_pk_rota_pedido FOREIGN KEY (codigo_rota) REFERENCES
Rota(codigo_rota),
CONSTRAINT fk_pk_situacao_pedido FOREIGN KEY (codigo_situacao_pedido)
REFERENCES SituacaoPedido(codigo_situacao_pedido),
CONSTRAINT fk_pk_responsavel_pedido FOREIGN KEY (codigo_responsavel)
REFERENCES Responsavel(codigo_responsavel)
)
```

**SituacaoPedido** – Tabela onde será cadastrada a situação do pedido do transporte escolar pelo responsável pelo aluno, sendo essas situações uma das seguintes: “negado: endereço não atende”, “em análise: documentos”, “homologação diretoria”, “aprovado: fila de espera” ou “aprovado: regular”.

```
(
codigo_situacao_pedido TINYINT PRIMARY KEY,
descricao_situacao_pedido varchar (40) not null
)
```

**HistoricoPedido** – Tabela onde será armazenado os pedidos do transporte escolar e seus deferimentos ou indeferimento.

```
(
codigo_historico_pedido INT AUTO_INCREMENT PRIMARY KEY,
codigo_pedido int not null,
codigo_situacao_pedido tinyint not null,
data_abertura datetime not null,
data_fechamento datetime not null,
codigo_funcionario int not null,
descricao varchar (60),
CONSTRAINT fk_pk_historico_pedido_pedido FOREIGN KEY (codigo_pedido)
REFERENCES Pedido(codigo_pedido),
CONSTRAINT fk_pk_historico_pedido_situacao_pedido FOREIGN KEY
(codigo_situacao_pedido) REFERENCES SituacaoPedido(codigo_situacao_pedido),
CONSTRAINT fk_pk_historico_pedido_funcionario FOREIGN KEY
(codigo_funcionario) REFERENCES Funcionário(codigo_funcionario)
)
```

**Motorista** – Tabela onde será cadastrado os motoristas que será alocado em determinado veículo/rota.

```
(
codigo_motorista TINYINT AUTO_INCREMENT PRIMARY KEY,
nome varchar (50) not null,
codigo_turno tinyint,
CONSTRAINT fk_pk_turno_motorista FOREIGN KEY (codigo_turno) REFERENCES
Turno(codigo_turno)
)
```

**Monitor** – Tabela onde será cadastrada o monitor que irá trabalhar em determinado veículo/rota.

```
(
codigo_monitor TINYINT AUTO_INCREMENT PRIMARY KEY,
nome varchar (50) not null,
codigo_turno tinyint,
CONSTRAINT fk_pk_turno_monitor FOREIGN KEY (codigo_turno) REFERENCES
Turno(codigo_turno)
)
```

**Veiculo** – Tabela onde será cadastrado as informações do veículo.

```
(
codigo_veiculo INT AUTO_INCREMENT PRIMARY KEY,
codigo_tipo_veiculo smallint not null,
vagas tinyint not null,
placa char (7) not null,
CONSTRAINT fk_pk_veiculo_tipo_veiculo FOREIGN KEY (codigo_tipo_veiculo)
REFERENCES TipoVeiculo (codigo_tipo_veiculo)
)
```

**Rota** – Tabela onde será cadastrado os tipos de rotas.

```
(
codigo_rota INT PRIMARY KEY,
descricao varchar (255) not null,
codigo_veiculo int not null,
codigo_monitor tinyint not null,
codigo_motorista tinyint null,
CONSTRAINT fk_pk_rota_veiculo FOREIGN KEY (codigo_veiculo) REFERENCES
Veiculo(codigo_veiculo),
CONSTRAINT fk_pk_rota_monitor FOREIGN KEY (codigo_monitor) REFERENCES
Monitor(codigo_monitor),
)
```

```

CONSTRAINT fk_pk_rota_motorista FOREIGN KEY (codigo_motorista) REFERENCES
Motorista(codigo_motorista)
)

```

**Trajeto** – Tabela onde será cadastrado as rotas que será feita pelos veículos/motoristas.

```

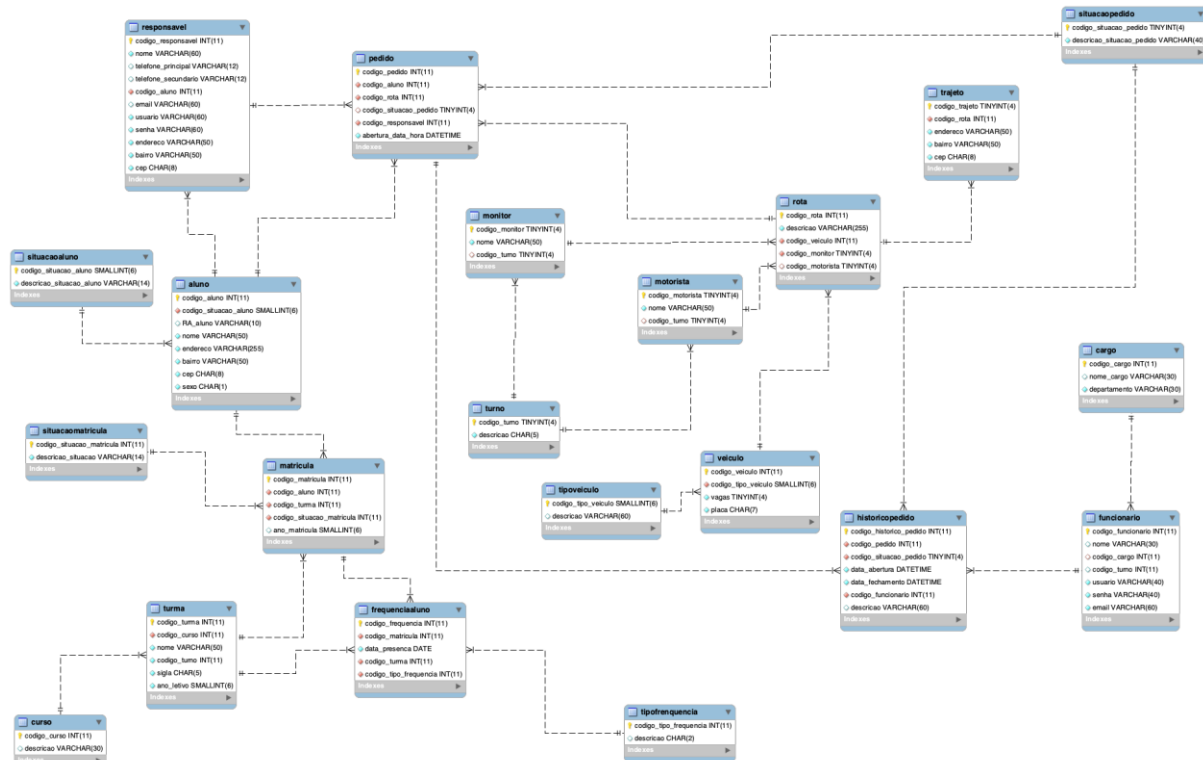
(
codigo_trajeto TINYINT AUTO_INCREMENT PRIMARY KEY,
codigo_rota int not null,
endereco varchar (50) not null,
bairro varchar (50) not null,
cep char (8) not null,
CONSTRAINT fk_pk_rota FOREIGN KEY (codigo_rota) REFERENCES
Rota(codigo_rota)
)

```

### 3.3. Estrutura do sistema

A arquitetura do sistema internamente está em MVC utilizando *framework* Flask e o banco de dados relacional MariaDB, conforme as figuras 11 e 12 respectivamente.

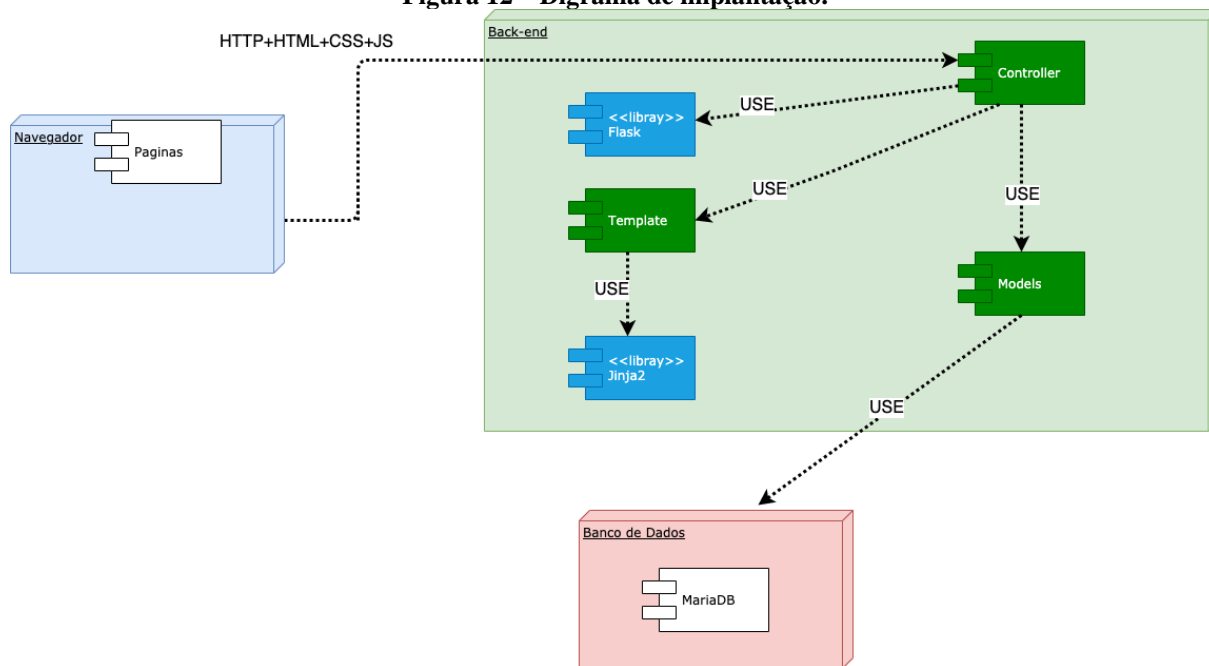
Figura 11 – Modelo de dados.



Fonte: Os autores.



**Figura 12 – Digrama de implantação.**



Fonte: Os autores.

### 3.4. Tecnologias utilizadas

A URL do projeto no GitHub é <https://github.com/gabrielmotroni-prog/sys-stadia-ope.git>

**Quadro 3 – Tecnologias utilizadas.**

Tecnologia	Justificativa
Git	Sistema de versionamento de código distribuído entre todos os desenvolvedores. Necessário também devido as ferramentas de hospedagem escolhidas.
GitHub	Site onde o código-fonte é hospedado e versionado.
Python	Linguagem de programação utilizada para desenvolver o <i>back-end</i> do projeto.
MariaDB	Sistema gerenciador de banco de dados relacional utilizado para persistirem-se os dados cadastrais do projeto.
Flask	Biblioteca em Python utilizada para servir-se as funcionalidades do <i>back-end</i> por meio do protocolo HTTP.
HTML	Padrão no qual as páginas servidas ao navegador cliente estão codificadas.
CSS	Padrão utilizado pelo navegador para estilizar e formatar as páginas clientes adequadamente.
Heroku	Utilizado para hospedar o <i>back-end</i> e o banco de dados.
Bootstrap	Biblioteca utilizada para simplificar e padronizar a estilização do <i>front-end</i> .

Fonte: Os autores.

## Considerações finais

Foi apresentado o primeiro MVP ao cliente, o qual foi muito bem recebido e também já apontados alguns pontos de melhoria. Além de abordar os requisitos mínimos do negócio, a solução já conta com recursos que oferecem mais inteligência ao processo de negócio. A experiência foi super válida ao grupo e cliente, principalmente ao grupo devido a experiência em campo no acompanhamento e desenvolvimento do *software* desde os artefatos até sua primeira versão funcional. O projeto desenvolvido pode ser acessado por meio da URL <https://systadia.herokuapp.com/>.

## Referências bibliográficas

Quadro 4 – Referências bibliográficas.

Fonte	Descrição
Site da escola	<a href="https://eeritapintoaraujo.webnode.com/">https://eeritapintoaraujo.webnode.com/</a>
Facebook da escola	<a href="https://www.facebook.com/eeritapintodearaujo?ref=bookmarks">https://www.facebook.com/eeritapintodearaujo?ref=bookmarks</a>
Secretaria de Educação	<a href="https://www.educacao.sp.gov.br/transporte-escolar">https://www.educacao.sp.gov.br/transporte-escolar</a>
Site da diretoria de ensino - Núcleo de Compras e Serviços	<a href="https://deleste3.educacao.sp.gov.br/nucleo-de-compras-e-servicos-ncs/">https://deleste3.educacao.sp.gov.br/nucleo-de-compras-e-servicos-ncs/</a>
Site da diretoria de ensino - Núcleo de Administração	<a href="https://deleste3.educacao.sp.gov.br/nucleo-de-administracao-nad/">https://deleste3.educacao.sp.gov.br/nucleo-de-administracao-nad/</a>
Blog do Núcleo Pedagógico - Educação Especial	<a href="https://educacaoespecialleste3.blogspot.com/search?q=transporte">https://educacaoespecialleste3.blogspot.com/search?q=transporte</a>
Blog de Boas Práticas	<a href="https://padlet.com/nucleoleste3/83c8pn281r6ct3rn">https://padlet.com/nucleoleste3/83c8pn281r6ct3rn</a>
RESOLUÇÃO SE 27 09-05-2011	Disciplina a concessão de transporte escolar para assegurar aos alunos o acesso às escolas públicas estaduais <a href="https://www.educacao.sp.gov.br/a2sitebox/arquivos/documentos/724.pdf">https://www.educacao.sp.gov.br/a2sitebox/arquivos/documentos/724.pdf</a>

Fonte: Os autores.

## Glossário

Termo, conceito ou abreviação	Definição
R.A.	Abreviação de Registro do Aluno, número gerado pelo sistema acadêmico da instituição no ato da matrícula.
SED	Secretaria Escolar Digital é um programa inovador criado para centralizar e facilitar todas as operações que envolvem a gestão diária da administração escolar, professores, alunos e seus responsáveis. A nova plataforma é personalizada para cada tipo de usuário.
E. E.	Abreviação de Escola Estadual.

EJA	Abreviação de Educação de Jovens e Adultos. É uma modalidade de ensino criada pelo Governo Federal que perpassa todos os níveis da Educação Básica do país, destinada aos jovens, adultos e idosos que não tiveram acesso à educação na escola convencional na idade apropriada. Permite que o aluno retome os estudos e os conclua em menos tempo e, dessa forma, possibilitando sua qualificação para conseguir melhores oportunidades no mercado de trabalho.
SARESP	O Sistema de Avaliação do Rendimento Escolar do Estado de São Paulo, é uma prova externa, aplicada anualmente, desde 1996, pela Secretaria da Educação do Estado de São Paulo para avaliar sistematicamente o Ensino Básico na rede estadual, e produzir um diagnóstico do rendimento escolar básico paulista.
SAEB	O Sistema Nacional de Avaliação da Educação Básica é um conjunto de sistemas de avaliação do ensino brasileiro, desenvolvido e gerenciado pelo Instituto Nacional de Estudos e Pesquisas Educacionais Anísio Teixeira, autarquia do Ministério da Educação e é baseado na Teoria da Resposta ao Item.
ANA	A Avaliação Nacional da Alfabetização é um dos instrumentos do Sistema de Avaliação da Educação Básica (Saeb) e mede os níveis de alfabetização e letramento em língua portuguesa, a alfabetização em matemática e as condições de oferta do ciclo de alfabetização das redes públicas.
AAP	O objetivo da Avaliação de Aprendizagem em Processo é diagnosticar o nível de aprendizado dos estudantes matriculados na rede estadual de ensino. O caderno de perguntas e respostas é formado por uma redação, questões dissertativas e de múltipla escolha de língua portuguesa e matemática, tendo como base o conteúdo do Currículo Oficial do Estado de São Paulo. O exame é aplicado duas vezes ao longo do ano letivo, em fevereiro e agosto, para alunos a partir do 2º Ano do Ensino Fundamental, Anos Finais do Ensino Fundamental e todas as séries do Ensino Médio, totalizando um número superior a 3,9 milhões de participantes.
	Os índices extraídos são utilizados pela Educação para produzir orientações aos educadores, desenvolver programas e projetos que atuem nas dificuldades dos alunos.
PROVA BRASIL	A Prova Brasil e o Sistema Nacional de Avaliação da Educação Básica (Saeb) são avaliações para diagnóstico, em larga escala, desenvolvidas pelo Instituto Nacional de Estudos e Pesquisas Educacionais Anísio Teixeira (Inep/MEC). Têm o objetivo de avaliar a qualidade do ensino oferecido pelo sistema educacional brasileiro a partir de testes padronizados e questionários socioeconômicos.
PcD	Atualmente, o termo oficial e CORRETO que foi definido pela Convenção das Nações Unidas sobre o Direito das Pessoas com Deficiência é PcD que significa Pessoa com Deficiência, pois ele esclarece que há algum tipo de deficiência sem que isso inferiorize quem a tem.
Barreira	Barreira física, ou obstáculo que impeça ou dificulte o seu acesso à escola, ou lhe prejudique a liberdade de movimento, a circulação com segurança, a integridade, como por exemplo: rodovia e ferrovia sem passarela, ou faixa de travessia sem semáforo; rio, lago, lagoa, brejo, ribeirão, riacho, braços de mar, sem pontes ou passarelas, trilhas em matas, serras, morros, ou locais desertos; divisória física fixa (muro ou cerca), linha eletrificada, vazadouro (lixão).