

## Ferramenta de auxílio à organização de atividades escolares

**Cristhian da Silveira Maia, Rafael H. D. Zottesso**

Instituto Federal do Paraná - Campus Paranavaí (IFPR)  
Paranavaí – PR – Brazil

crsthian.smaia@gmail.com, rafael.zottesso@ifpr.edu.br

**Resumo.** *Esse artigo foi produzido para documentar um sistema web responsivo que consiste em uma “agenda” comum para alunos de uma turma de qualquer instituição de ensino, onde alunos poderão criar uma turma no sistema e cadastrar avisos como provas, trabalhos e tarefas além de algumas outras funcionalidades que o sistema proporciona buscando aumentar o desempenho do aluno em relação às tarefas escolares. Para o desenvolvimento do sistema foi utilizado o Django, um framework baseado em Python que possibilita o desenvolvimento de aplicações web por meio do paradigma Model Template View (MTV).*

### 1. Introdução

Tendo em vista a quantidade de matérias na grade de um único aluno e a gama de tarefas, trabalhos e provas que são direcionadas aos mesmos em relação ao número de matérias faz com que os alunos entrem em um grande compromisso com prazos de entrega, geralmente determinado por uma data de entrega final.

Levando ainda em consideração que os alunos tenham outras tarefas além de suas atividades escolares e, com a total quantidade dessas atividades podem acabar por cair no descuido da desatenção de tal prazo.

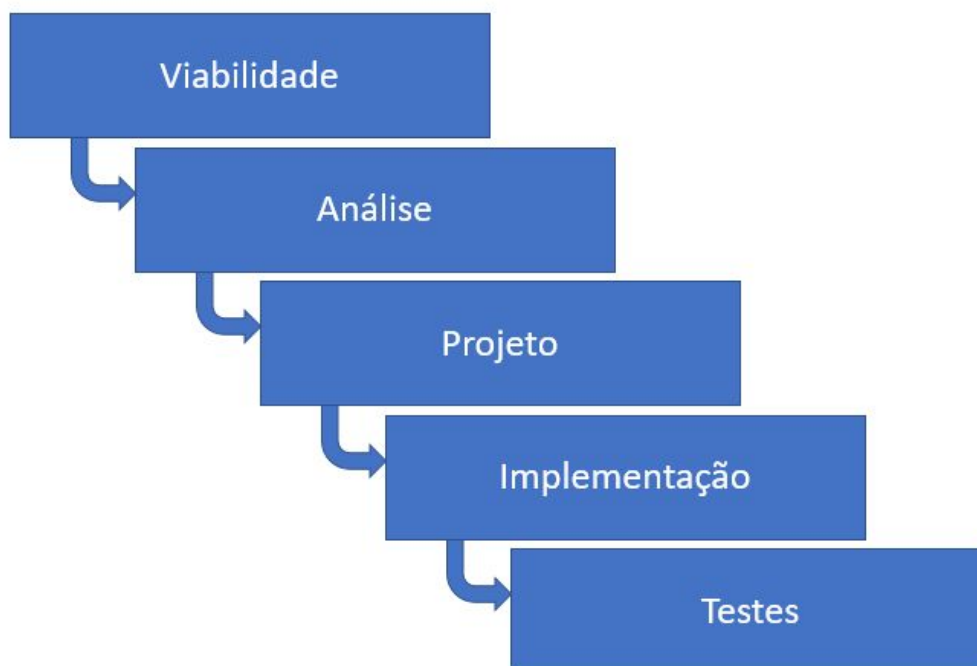
Com esses problemas encontrados por muitos estudantes, convincente devido à quantidade de tarefas, foi proposto o desenvolvimento de um sistema web com o objetivo de auxiliar os alunos em relação às atividades escolares. Por meio desse recurso, será possível uma comunicação do sistema para com os alunos em relação às suas atividades escolares que serão organizados à uma turma cadastrada no sistema para certificar que atividades sejam realizadas dentro do prazo proposto à turma.

Levando em consideração as muitas vezes na qual um aluno se depara com o dia da entrega de uma atividade sem saber da existência do mesmo, isso poderia ser resolvido por meio de uma agenda/caderno de lembretes. Porém, com a ideia de uma “agenda” comum, a qual possibilitaria a postagens de avisos, pode garantir que os alunos estarão cientes de todas as atividades fazendo com que seja mais eficiente para os alunos de uma turma.

### 2. Metodologia

Para o desenvolvimento do sistema foi utilizada a metodologia cascata, que consiste em

passos sequenciais para alcançar o resultado final esperado.



**Figura 1. Modelo cascata**

Inicialmente foi analisada a viabilidade econômica, operacional, técnica, de cronograma sobre a produção da ideia proposta. Em seguida a análise da ideia, onde é discutida as funcionalidades do sistema e como pode ser maior aproveitado. Na etapa Projeto, iniciamos a prototipação do mesmo, colocando todas as funcionalidades necessárias em conjunto, dizendo como os mesmos irão interagir por meio da modelagem de software produzindo então os requisitos do sistema. Então tudo isso servirá de base para todas as etapas da implementação do software. Após toda a implementação dos requisitos, haverá teste total do sistema, devendo atender todas as funcionalidades e requisitos apontadas na etapa de análise e projeto, porém não houve implementação para testes em um ambiente real.

### **3. Desenvolvimento**

O sistema consiste em um aplicativo web destinado a alunos de instituições de ensino, que possibilitará a persistência de dados por operações de Cadastro, Consulta, Atualização e Exclusão (CRUD) realizados pelo representante de cada turma, enquanto usuários normais poderão apenas visualizar tais dados.

#### **3.1. Descrição do sistema**

O aplicativo tem como base as turmas de uma instituição de ensino, que podem ser criadas por usuários que tem a permissão de “representante”. A turma criada considera o usuário que efetuou o cadastro como “representante”. O gerenciamento das turmas permite cadastrar dados como: professores, matérias, atendimentos, provas, trabalhos, tarefas e avisos gerais. Já um usuário normal poderá apenas visualizar informações das turmas em que está inserido.

A função “administrador” tem pouco contato com o sistema, realizando apenas cadastros de colégios e de representantes, além das manutenções do sistema sempre que necessárias.

Os requisitos funcionais irão representar as funções que deverão ser fornecidas ao usuário por meio do sistema e os não funcionais definirão como o sistema deverá realizar, expondo as necessidades que não podem ser atendidas através de funcionalidades.

A criticidade de cada requisito é definida por (1) essencial, (2) importante e (3) desejável.

|                                                                                                                                                                                                                                                     |
|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <p><b>ID:</b> RF001. <b>Nome:</b> Manter Usuários. <b>Prioridade:</b> Essencial.</p> <p><b>Descrição:</b> O sistema deverá possibilitar o auto cadastro de novos usuários e manter suas informações pessoais.</p>                                   |
| <p><b>ID:</b> RF002. <b>Nome:</b> Cadastrar colégios. <b>Prioridade:</b> Essencial.</p> <p><b>Descrição:</b> O usuário deverá ser capaz de cadastrar novos colégios no sistema.</p>                                                                 |
| <p><b>ID:</b> RF003. <b>Nome:</b> Manter turmas. <b>Prioridade:</b> Essencial.</p> <p><b>Descrição:</b> O usuário deverá ser capaz de criar, alterar e excluir turmas no sistema.</p>                                                               |
| <p><b>ID:</b> RF004. <b>Nome:</b> Manter alunos. <b>Prioridade:</b> Essencial.</p> <p><b>Descrição:</b> O representante deverá ser capaz de inserir e remover alunos de sua turma.</p>                                                              |
| <p><b>ID:</b> RF005. <b>Nome:</b> Manter representantes. <b>Prioridade:</b> Essencial.</p> <p><b>Descrição:</b> O usuário deverá ser capaz de se definir como representante bem como transferir o cargo para outro usuário da turma em questão.</p> |
| <p><b>ID:</b> RF006. <b>Nome:</b> Manter professores. <b>Prioridade:</b> Essencial.</p> <p><b>Descrição:</b> O representante deverá ser capaz de criar, alterar e excluir professores por turma no sistema.</p>                                     |
| <p><b>ID:</b> RF007. <b>Nome:</b> Manter matérias. <b>Prioridade:</b> Essencial.</p> <p><b>Descrição:</b> O representante deverá ser capaz de criar, alterar e excluir matérias por turma no sistem</p>                                             |
| <p><b>ID:</b> RF008. <b>Nome:</b> Manter matérias. <b>Prioridade:</b> Essencial.</p> <p><b>Descrição:</b> O representante deverá ser capaz de criar, alterar e excluir horários de atendimentos dos professores da turma em q</p>                   |

|                                                                                                                                                                                                                                                                              |
|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <p><b>ID:</b> RF009. <b>Nome:</b> Manter avisos. <b>Prioridade:</b> Essencial.</p> <p><b>Descrição:</b> O representante deverá ser capaz de criar, alterar e excluir avisos destinados à turma.</p>                                                                          |
| <p><b>ID:</b> RF010. <b>Nome:</b> Consultar turmas. <b>Prioridade:</b> Essencial.</p> <p><b>Descrição:</b> Os usuários devem ter a possibilidade de visualizar as turmas em que está como representante ou como aluno.</p>                                                   |
| <p><b>ID:</b> RF011. <b>Nome:</b> Emitir relatório de alunos. <b>Prioridade:</b> Importante.</p> <p><b>Descrição:</b> Os usuários devem ter a possibilidade de visualizar os alunos das turmas.</p>                                                                          |
| <p><b>ID:</b> RF012. <b>Nome:</b> Emitir relatório de alunos. <b>Prioridade:</b> Importante.</p> <p><b>Descrição:</b> Os usuários devem ter a possibilidade de visualizar as matérias da turma selecionada.</p>                                                              |
| <p><b>ID:</b> RF013. <b>Nome:</b> Emitir relatório de professores. <b>Prioridade:</b> Importante.</p> <p><b>Descrição:</b> Os usuários devem ter a possibilidade de visualizar os professores da turma selecionada.</p>                                                      |
| <p><b>ID:</b> RNF001. <b>Nome:</b> Possibilitar alteração de representante de turma. <b>Prioridade:</b> Importante.</p> <p><b>Descrição:</b> Os usuários devem ter a possibilidade de alterar o representante da turma, desde que o usuário seja representante da turma.</p> |
| <p><b>ID:</b> RNF002. <b>Nome:</b> Enviar email para os alunos quando um novo aviso for cadastrado. <b>Prioridade:</b> Desejável.</p> <p><b>Descrição:</b> Os representantes devem ter a possibilidade de alterar o representante de sua turma selecionada.</p>              |
| <p><b>ID:</b> RNF003. <b>Nome:</b> O aluno ter a possibilidade de aceitar ou recusar o convite de uma turma. <b>Prioridade:</b> Desejável.</p> <p><b>Descrição:</b> Os usuários devem ter a possibilidade de receber convites de participação em turmas.</p>                 |

### 3.2. Recursos Utilizados

Foi utilizado o software *Astah UML 7.2.0/1ff236*, provido por *Change Vision, Inc.* para a projetar o caso de uso e o diagrama de classes.

O protótipo foi construído com o propósito de se utilizar em todos os dispositivos com um navegador e acesso à Internet, para isso foi utilizado o Bootstrap 4.1.1 no desenvolvimento *front-end* do site para que a navegação se tornasse

responsiva, atendendo assim todos as dimensões de tela e englobasse os dispositivos móveis tornando-se assim um sistema multiplataforma.

Para a codificação do sistema foi utilizado o *Django 2.0.5*, um *framework* com base em Python de desenvolvimento de aplicativos para a web, que utiliza o paradigma *Model Template View* (MTV), baseado no *Model View Controller* (MVC), um padrão que separa o desenvolvimento em 3 camadas, o *Model* é onde se implementa o banco de dados, *Template* é responsável pela interface do sistema e *View*, o qual processa dados vindos da camada *Model* e enviados ao *Template*.

No desenvolvimento também foi utilizado alguns *plugins*, o *Django Bootstrap4* (*django-bootstrap4*) na versão 0.0.6 para que fosse possível utilizar componentes do bootstrap nos formulários gerados pelas *Views* e o *Datatables 1.10.18*, um *plugin* jQuery que adiciona uma tabela flexível com interações de usuário avançadas.

Para o banco foi utilizado o *SQLite*, por se tratar de um protótipo, o que garante poucas requisições e pouca preocupação com a segurança, leve e de fácil integração nos diversos computadores e seus sistemas operacionais utilizados no desenvolvimento deste trabalho.

### 3.3. Projeto de Objetos

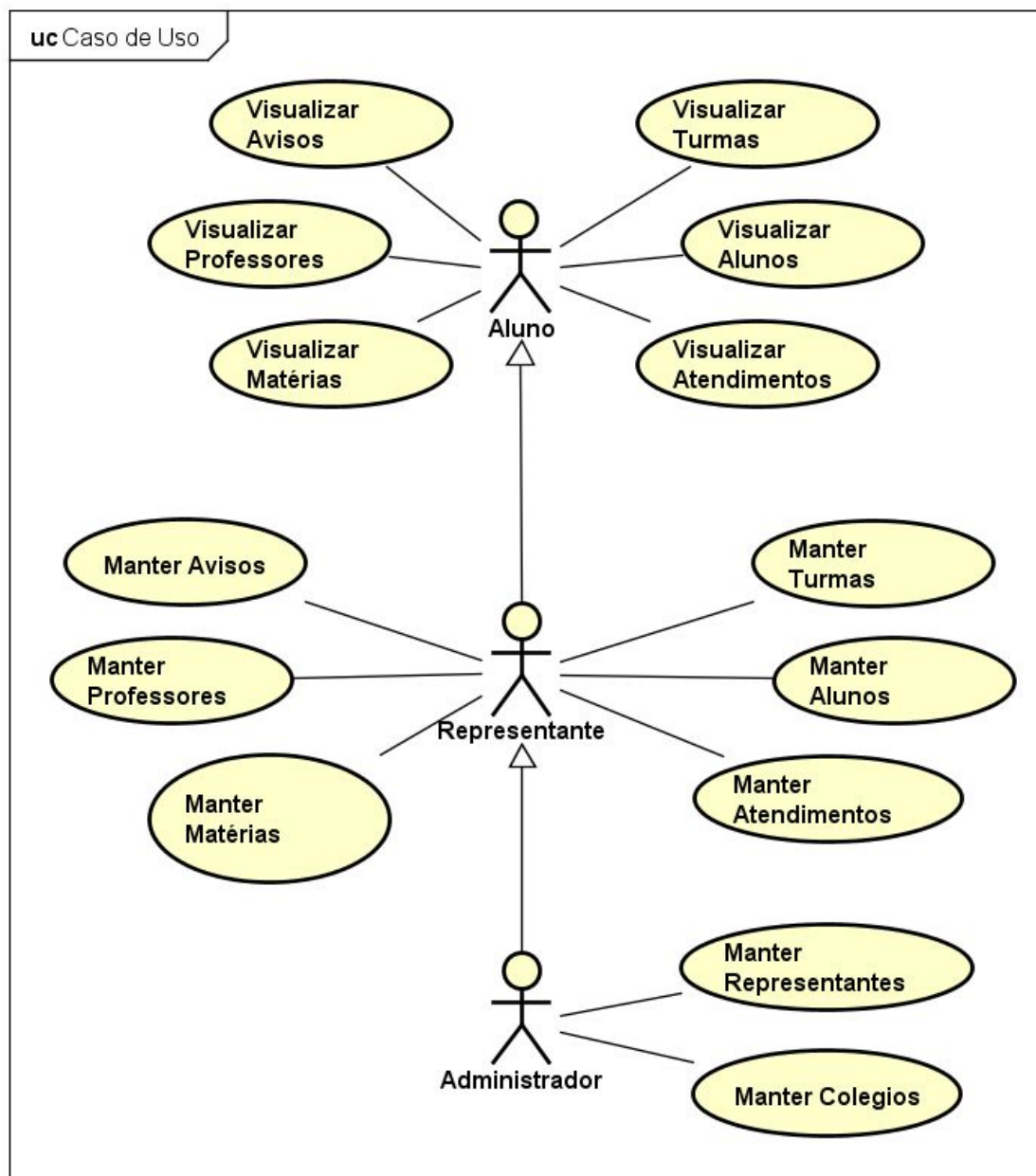
Para o desenvolvimento da aplicação, é necessário representar os objetos que devem ser implementados de acordo comportamento esperado do sistema por meio da ferramenta *Astah*.

As figuras devem apresentar os seguintes objetos: Alunos, Colégio, Professor, Materia, Atendimento, Turma e Avisos.

#### 3.3.1. Diagrama de Caso de Uso

A figura 2 apresenta o Diagrama de Caso de Uso, este diagrama tem a finalidade de definir o escopo do sistema.

Nele está representado as funções de todos os 3 atores do sistema: Administrador, Representante e Aluno. No qual Aluno poderá apenas realizar seu login no sistema e visualizar dados já cadastrados desde que esteja concedidos ao mesmo por meio de sua inserção a uma Turma.



**Figura 2. Diagrama de Caso de Uso**

O ator Administrador poderá realizar CRUD dos colégios no sistema além de poder conceder permissões de Representantes para Alunos (usuários comuns) por meio dos níveis de usuários sempre que necessário. O administrador tem a permissão de todos os outros atores por meio da associação de Generalização, porém poucas funções específicas a fim de um sistema “autônomo”.

O ator Representante é um usuário com nível de usuário acima de aluno e abaixo de administrador, quando nesse nível, o usuário tem permissão no sistema para realizar as operações CRUD das classes Turma, Atendimentos, Avisos, Professores e Materiais além das demais permissões concedidas pela Generalização ao ator Aluno, considerado um dos atores mais importantes do sistema, levando em consideração de ser um sistema

“autônomo”, isto é, sem muita influência do ator Administrador no funcionamento do sistema.

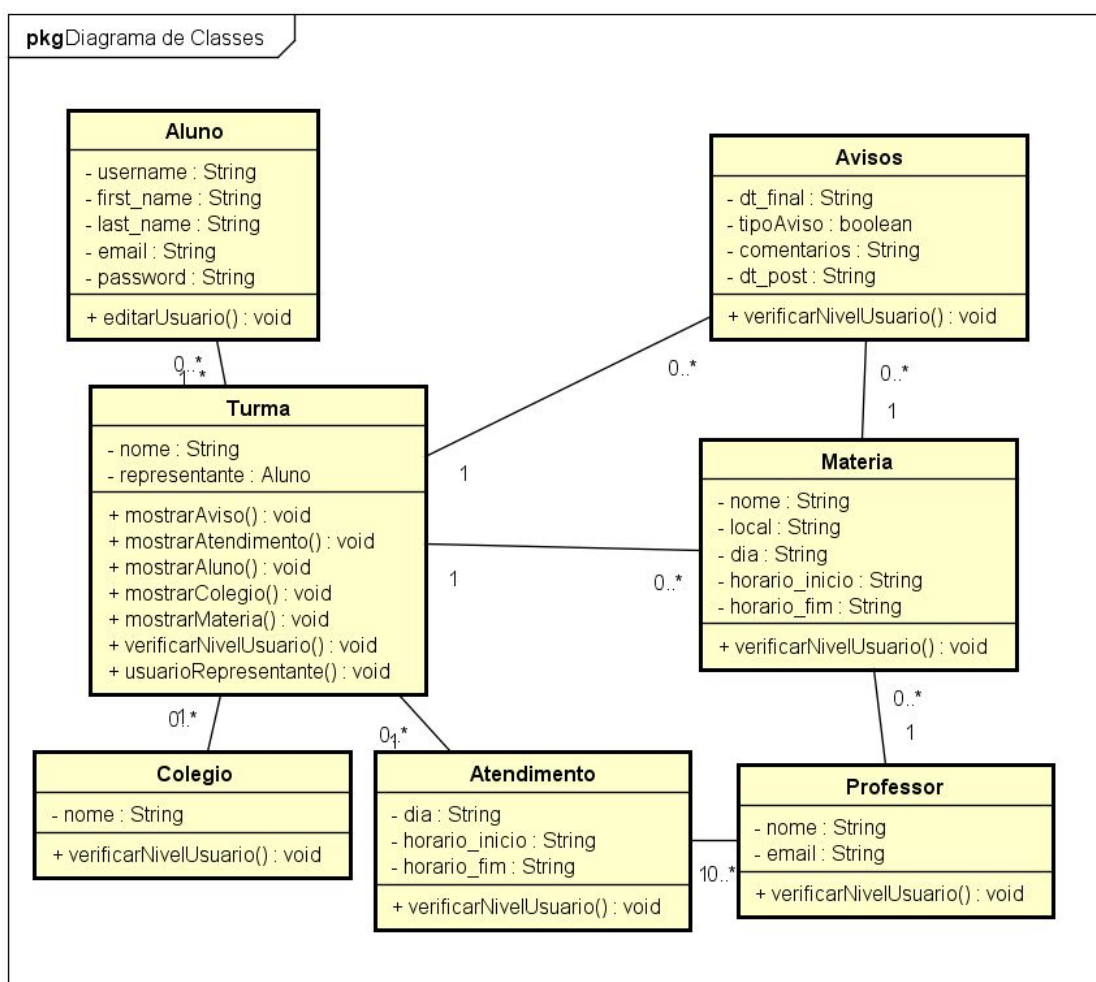
O ator Aluno apresentado na figura é apenas o usuário em primeiro nível de usuário que poderá apenas consultar dados anteriormente cadastrados por atores com tal poder de gerenciamento.

O caso Alunos, que foi representado com uma dependência à Turmas, expressa a ideia de que o representante pode inserir alunos na classe Turma no qual os alunos seriam os usuários do sistema.

O caso Aviso presente na figura 2 tem a possibilidade de cadastrar diferentes tipos de avisos, os quais seriam: Provas, Trabalhos, Tarefas e Aviso, na interface para o usuários eles são tratados como diferentes porém no esquema de dados eles são tratados como uma única tabela.

### 3.3.2. Diagrama de classes

A figura 3 apresenta Diagrama de Classes, onde se pode visualizar o relacionamento entre as tabelas do banco de dados e seus atributos.



**Figura 3. Modelo de dados**

Todos os atributos foram nomeados de acordo com a nomenclatura padrão do

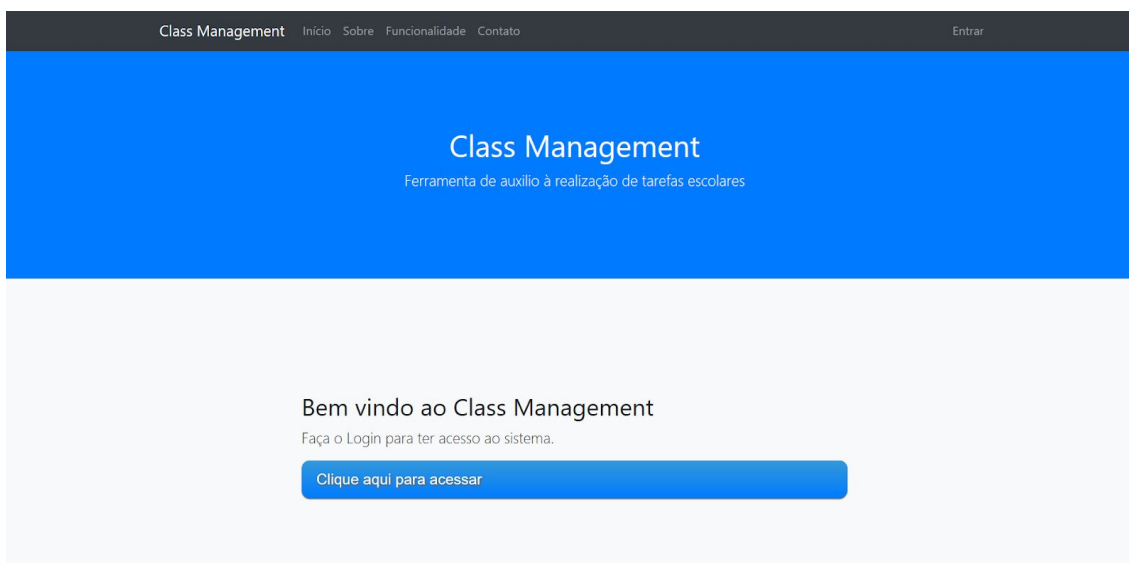


framework Django.

Em destaque a classe Turma a qual será a principal, o seu cadastro no sistema é importante para que as demais classes (desconsiderando as classes de Aluno, Colégio e Professor) possam realizar suas movimentações, pois são criadas em função de uma turma. Já os cadastros da classe professor não possuem associação com Turma, tendo associação somente com Matérias e Atendimentos, todas essas classes podem ter inserções de dados realizadas por Representantes e Administradores. A classe de Colégios e a definição de Representantes é uma tarefa exclusiva ao Administrador, um Colégio pode também estar em diversas turmas.

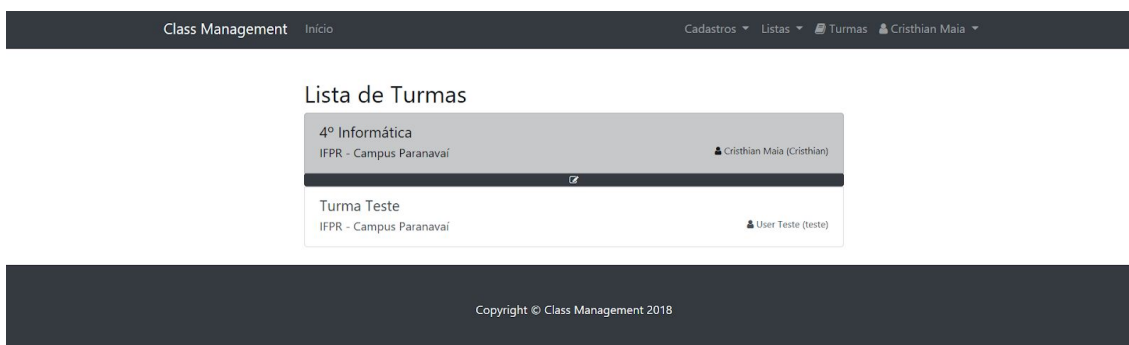
#### 4. Telas do Protótipo

A figura 4 apresenta a tela inicial de boas vindas do sistema sem a autenticação de usuário.



**Figura 4. Tela inicial do protótipo.**

A figura 5 apresenta a tela de turmas que o usuário autenticado participa. As Turmas que possuem fundo cinza são turmas na qual o usuário é representante, o mesmo pode ser identificado por seu nome de usuário no lado direito inferior de cada turma, as mesmas podem ser editadas por meio do botão mais escuro que está no rodapé da turma.



**Figura 5. Tela de Turmas.**



A figura 6 exibe a tela de Turma, onde é possível visualizar todos os avisos da turma, alunos nela cadastrados, horários de atendimentos e horários aulas, alguns componentes nessa tela remetem às movimentações CRUD, elas surgem somente quando o usuário autenticado é representante dessa mesma Turma.

The screenshot shows the 'Class Management' interface for a representative user. The page title is '4º Informática' and the user is 'Representante: Crísthian Maia'. The navigation bar includes 'Class Management', 'Início', 'Cadastros', 'Listas', 'Turmas', and 'Cristhian Maia'. The main content area is titled 'Avisos' and contains several cards for notices and assignments.

| Aviso                                                                                                                                  | Prova                                                                                                                 | Prova                                                                                                           |
|----------------------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------|
| <b>Educacao Fisica II</b><br>Não esquecer a roupa para a aula pratica<br>Data de Entrega: 10 de Outubro de 2018<br>5 de Agosto de 2018 | <b>Empreendedorismo</b><br>1ª Atividade avaliativa<br>Data de Entrega: 12 de Outubro de 2018<br>9 de Setembro de 2018 | <b>Fisica III</b><br>2ª Atividade avaliativa<br>Data de Entrega: 13 de Outubro de 2018<br>9 de Setembro de 2018 |
| <b>Trabalho</b><br><b>Fisica III</b><br>Explicar a morte do cavalo<br>Data de Entrega: 13 de                                           |                                                                                                                       |                                                                                                                 |

**Figura 6. Tela da Turma, representante autenticado.**

A figura 7 representa a tela de Turma quando um aluno está autenticado, o que o restringe apenas à consulta de dados

The screenshot shows the 'Class Management' interface for a student user. The page title is '4º Informática' and the user is 'Representante: Crísthian Maia'. The navigation bar includes 'Class Management', 'Início', 'Turmas', and 'User Teste'. The main content area is titled 'Avisos' and contains several cards for notices and assignments.

| Aviso                                                                                                                                  | Prova                                                                                                                 | Prova                                                                                                           |
|----------------------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------|
| <b>Educacao Fisica II</b><br>Não esquecer a roupa para a aula pratica<br>Data de Entrega: 10 de Outubro de 2018<br>5 de Agosto de 2018 | <b>Empreendedorismo</b><br>1ª Atividade avaliativa<br>Data de Entrega: 12 de Outubro de 2018<br>9 de Setembro de 2018 | <b>Fisica III</b><br>2ª Atividade avaliativa<br>Data de Entrega: 13 de Outubro de 2018<br>9 de Setembro de 2018 |
| <b>Trabalho</b><br><b>Fisica III</b><br>Explicar a morte do cavalo<br>Data de Entrega: 13 de Novembro de 2018                          |                                                                                                                       |                                                                                                                 |

**Figura 7. Tela da Turma, aluno autenticado.**

A figura 8 apresenta a tela de cadastro de aviso, exemplificando a base de um formulário de cadastro de dados, este presente em todas as telas de cadastro.

The screenshot shows a web interface for 'Class Management'. The main heading is 'Cadastrar Aviso'. The form contains the following elements: a 'Turma' dropdown menu, a 'Materia' dropdown menu, a 'Tipo de Aviso' dropdown menu, an 'Obrigatorio' checkbox, a 'Comentarios' text area, and a 'Data Final' input field. The top navigation bar includes links for 'Inicio', 'Cadastros', 'Listas', 'Turmas', and a user profile for 'Cristhian Maia'.

**Figura 8. Tela de Cadastro**

## 5. Considerações finais

Espera-se que, com o constante uso do sistema os usuários apresentem um melhor resultado em relação ao desempenho/comprometimento para com as atividades escolares por meio do auxílio que o aplicativo proporciona.

O sistema possui algumas funções aos usuários que são inseridos ao grupo representante no sistema. Esses usuários possuem a possibilidade de criar uma Turma e realizar cadastros das diversas classes do sistema em turmas nas quais somente ele tenha criado. Um exemplo seria, ao realizar um cadastro de um aviso, conforme a figura 7, as possibilidades de escolha do usuário na menu dropdown seria apenas as turmas na qual ele é representante (que havia criado anteriormente), posteriormente as matérias teriam apenas opções de matérias cadastradas em turmas das quais ele é representante.

Durante o desenvolvimento do sistema houve muita modificação das representações dos diagramas em relação ao desenvolvimento prático como o fato dos níveis de usuários serem definidos por grupos por meio do sistema Django enquanto na representação dos diagramas ele esteja como apenas atributos em classes.

Contudo, é importante salientar que o sistema não está pronto para um uso massivo, principalmente pela falta de um servidor dedicado, um banco de dados com uma boa segurança e que garanta maior desempenho conforme a quantidade de requisições além da ausência de algumas funções que em uma aplicação real seria necessário como por exemplo, um usuário comum (aluno) desejar desvincular sua conta de uma certa Turma.

## 6. Referências

Django Software Foundation. Django Web framework, 2.1.7, <https://www.djangoproject.com/>. Agosto.

Mary Lotz. (2018) “Waterfall vs. Agile: Which is the Right Development Methodology for Your Project?”, <https://www.seguetech.com/waterfall-vs-agile-methodology/>,

Agosto.

Douglas Hughey. (2009) “The Traditional Waterfall Approach”, <http://umsl.edu/~hugheyd/is6840/waterfall.html>, Agosto.

Lima, Adilson da Silva. “UML 2.3: do requisito à solução”, Brasil.

Stackify. (2017) “What are CRUD Operations: How CRUD Operations Work, Examples, Tutorials & More”, <https://stackify.com/what-are-crud-operations/>, Agosto.

Change Vision. Astah UML, 7.2.0/1ff236, <http://astah.net>. Agosto.

Andrews Medina. (2007) “O MVC, o MTV e o Django”, <http://pyman.blogspot.com/2007/04/o-mvc-o-mtv-e-o-django.html>, Agosto.

Zostera. django-bootstrap4, 1.0.1, <https://pypi.org/project/django-bootstrap4/>. Agosto.

SpryMedia Ltd. DataTables, 1.10.18, <https://cdn.datatables.net/1.10.18/>. Agosto.

D. Richard Hipp. SQLite, 3.22.0, <https://www.sqlite.org/>. Agosto.