

Venâncio Assumane Afonso

**Concepção e Implementação de um Sistema de Gestão de Matrículas: Estudo de Caso -
Escola Comunitária Santa Montanha**

Licenciatura em Informática

Universidade Pedagógica de Maputo

Maputo

2023

Venâncio Assumane Afonso

**Concepção e Implementação de um Sistema de Gestão de Matrículas: Estudo de Caso -
Escola Comunitária Santa Montanha**

Monografia apresentada ao curso de Informática, Faculdade de Engenharia e Tecnologia da Universidade Pedagógica de Maputo, para obtenção do grau académico de Licenciatura em Informática com habilitações a Engenharia de Desenvolvimento de Sistemas.

Supervisor(a):

Msc. Cláudia Ivete F. Jovo Gune

Universidade Pedagógica de Maputo

Maputo

2023

Índice.....	Página
Lista de Tabelas.....	IV
Lista de Figuras.....	V
Lista de Abreviaturas.....	VI
Declaração.....	VII
Dedicatória.....	VIII
Agradecimentos.....	IX
Resumo.....	X
Abstract.....	XI
CAPÍTULO I - INTRODUÇÃO.....	1
1.1.Problematização.....	2
1.2.Justificativa.....	3
1.3.Objectivos.....	3
1.3.1.Objectivo Geral.....	3
1.3.2.Objectivos Específicos.....	3
1.5.Hipóteses.....	4
1.6.Metodologia de Trabalho.....	4
1.7.Material Necessário.....	6
CAPÍTULO II - REVISÃO BIBLIOGRÁFICA.....	7
2.1. Gestão.....	7
2.2. Matrículas.....	8
2.3. Gestão de Matrículas Escolares.....	8
2.4. Engenharia de Software.....	9
2.5. Engenharia Web.....	9
2.6. Modelo de Processos.....	10
2.7. Metodologias de Desenvolvimento de <i>Software</i>	11
2.7.1. Rational Unified Process – RUP.....	12
2.8. Linguagens de Modelação UML.....	12
2.8.1. Diagrama de Caso de Uso.....	13
2.8.2. Diagrama de Classe.....	13
2.9. Frameworks de Desenvolvimento Web.....	13
2.9.1. Laravel.....	14
2.9.4. Comparação entre <i>Frameworks</i>	15
2.10. Linguagens de Programação.....	16
2.10.1. PHP.....	16
2.10.2. Javascript (JS).....	17

2.10.3. Comparação entre linguagens de Programação.....	17
2.11. Sistema de Gestão de Base de Dados (SGBD).....	17
2.11.1. Comparação entre SGBD.....	17
CAPITULO III – ANÁLISE E APRESENTAÇÃO DE RESULTADOS.....	18
3.1. Descrição do Local.....	18
3.2. Modelo Actual do Processo de Matrículas.....	18
3.2.1. Limitações do Modelo Actual.....	19
3.3. Modelo Proposto.....	20
3.4. Levantamento de Requisitos do Sistema Proposto.....	20
3.4.1. Requisitos do Utilizador.....	20
3.4.2. Fase de Planeamento.....	21
3.4.2.1. Actores.....	21
3.4.2.2. Requisitos Funcionais.....	21
3.4.2.4. Requisitos Não-funcionais.....	25
3.4.2.5. Análise e Projecto.....	26
3.4.2.5.1. Diagrama de Classe.....	26
3.4.3. Fase de Desenvolvimento.....	26
3.4.3.1. Telas do Sistema.....	26
3.4.4. Fase de Estabilização.....	31
3.4.4.1. Caso de Teste.....	31
3.5. Processo de implementação do sistema de gestão de matrícula.....	33
3.5.1. Requisitos necessários para a implementação do sistema de gestão de matrículas.....	33
3.5.2. Recursos de <i>hardware</i> :.....	33
3.5.3. Recursos de <i>software</i>	33
3.5.4. Implementação do sistema.....	34
3.6. Resultados do sistema.....	40
4. CONSIDERAÇÕES FINAIS, RECOMENDAÇÕES, TRABALHOS FUTUROS E LIMITAÇÕES.....	43
4.1. Considerações Finais.....	43
4.2. Recomendações.....	44
4.3. Trabalhos Futuros.....	44
4.4. Limitações.....	45
REFERÊNCIAS BIBLIOGRÁFICAS.....	46
Apêndice.....	48
Apêndice 1: Guião de entrevista a direcção da escola.....	49

Lista de Tabelas

Tabela 1: Material Necessário.....	6
Tabela 2: Comparação entre <i>Frameworks</i>	15
Tabela 3: Comparação entre Linguagens de Programação.....	17
Tabela 4: Comparação entre SGBD.....	17
Tabela 5: Actores.....	21
Tabela 6: Gerir Usuário.....	21
Tabela 7: Gerir Ano Lectivo.....	22
Tabela 8: Gerir Classe.....	22
Tabela 9: Gerir Turma.....	22
Tabela 10: Gerir Aluno.....	22
Tabela 11: Gerir Matricula.....	23
Tabela 12: Gerir Relatório.....	23
Tabela 13: Efectuar acesso.....	25
Tabela 14: Período de funcionamento.....	25
Tabela 15: Período de Matricula.....	25
Tabela 16: Cadastrar Aluno.....	31
Tabela 17: Cadastrar Matricula.....	32
Tabela 18: Cadastrar Ano Lectivo, Classe e Turma.....	32

Lista de Figuras

Figura 1: Pátio da ECOSAMO.....	18
Figura 2: Modelo actual.....	19
Figura 3: Modelo proposto.....	20
Figura 4: Diagrama de caso de uso.....	24
Figura 5: Diagrama de classe.....	26
Figura 6: Tela de cadastro de aluno.....	27
Figura 7: Tela de Matrícula.....	27
Figura 8: Tela dos alunos que pretendem renovar a matrícula.....	30
Figura 9: Tela de renovação de matrícula.....	31
Figura 10: Tela principal da mozdomains.....	34
Figura 11: Tela de escolha de plano de hospedagem.....	34
Figura 12: Tela de Cpanel.....	35
Figura 13: Tela de <i>MYSQL Databases</i>	35
Figura 14: Tela de criação da base de dados.....	36
Figura 15: Tela de acesso ao phpMyAdmin.....	36
Figura 16: Tela de importação da base de dados na hospedagem.....	37
Figura 17: Tela de acesso ao <i>File Manager</i>	37
Figura 18: Tela que mostra a pasta onde se coloca o sistema.....	38
Figura 19: Tela de <i>upload</i>	38
Figura 20: Tela que ilustra o ficheiro <i>.env</i>	39
Figura 21: Tela de configuração da base de dados no arquivo <i>.env</i>	39
Figura 22: Tela inicial da ECOSAMO.....	40
Figura 23: Tela inicial da Secretaria.....	41
Figura 24: Tela de visualização e impressão da lista.....	41
Figura 25: Lista definitiva.....	42

Lista de Abreviaturas

ECOSAMO	Escola Comunitária Santa Montanha
ESG	Ensino Secundário Geral
UML	Unified Modeling Language (<i>Linguagem de Modelagem Unificada</i>)
RUP	Rational Unified Process (<i>Processo Unificado Racional</i>)
MVC	Model-View-Controller (<i>Modelo-Visão-Controle</i>)
REST	Representational State Transfer (<i>Transferencia Representacional de Estado</i>)
HTTP	HyperText Transfer Protocol (<i>Protocolo de Transferencia de Hipertexto</i>)
MVT	Model-View-Template (<i>Modelo de Visualização do modelo</i>)
HTML	HyperText Markup Language (<i>Linguagem de Marcação de Hipertexto</i>)
URL	Uniform Resource Locator (<i>Locador Padrão de Recursos</i>)
SGBD	Data Base Management System (<i>Sistema de Gestão de Base de Dados</i>)
SQL	Structured Query Language (<i>Linguagem de Consulta Estruturada</i>)

Declaração

Declaro que a presente monografia é resultado de investigação pessoal conjugado com as orientações da supervisora, o conteúdo que nele contem é original e todas as fontes bibliográficas consultadas estão devidamente mencionadas no texto, nas notas de rodapé e na bibliografia final. Declaro ainda que este trabalho não foi apresentado em nenhuma outra instituição nem como trabalho científico, nem para obtenção de qualquer grau académico.

Maputo, _____ de _____ de _____

(Venâncio Assumane Afonso)

Dedicatória

Este trabalho é dedicado aos meus pais Pompílio Afonso e Emília Assumane, pelo apoio incondicional e por fazerem-me acreditar que o meu futuro depende apenas da minha força de vontade.

Agradecimentos

Na realização de qualquer actividade para além da força de vontade individual, existe sempre uma motivação, seja ela positiva ou negativa, como também a presença de inúmeras pessoas que directa ou indirectamente dão o seu contributo para que o trabalho corra da melhor forma. E porque não quero ficar alheio ao apoio recebido ao longo do percurso académico, é chegada a altura de endereçar os meus agradecimentos a cada um dos intervenientes que contribuiu para que o presente trabalho chegasse ao seu culminar.

Quero agradecer aos meus pais por me terem dado a vida, pelas noites que passaram em claro orando pelo meu sucesso, pelos seus conselhos, amor incondicional e confiança que depositaram em mim. Aos meus irmãos (sem excepção alguma) pelo companheirismo, suporte e inspiração.

Agradeço igualmente a minha noiva Matilde de Oliveira pelo apoio, e por tornar-se a minha principal motivação, por trazer-me forças para seguir em frente, mesmo quando eu não tivesse vontade de prosseguir e me visse desanimado.

A gratidão vai de igual modo aos docentes da Universidade Pedagógica, Delegação de Maputo, de modo especial aos docentes da Faculdade de Engenharia e Tecnologia - FET e muito particularmente aos docentes do Curso de Informática, pela paciência e sabedoria que tiveram em transmitir o conhecimento que permitiu o alcance do grau académico, hoje obtido. Agradeço aos meus colegas de curso, pela partilha de conhecimentos.

E porque a lista é vasta, estendo a minha gratidão a todos que directamente ou indirectamente contribuíram em palavras, acções, pensamentos, na minha formação académica, profissional e pessoal.

Resumo

A concepção de um sistema de gestão de matrícula tem-se mostrado uma alternativa bastante eficiente na execução de tarefas, dadas suas inúmeras vantagens. Para permitir que o processo de matrícula decorra de forma eficaz, flexível e em curto tempo, é necessário que haja um sistema que permita a realização de um número considerável de actividades. No entanto, em situações em que a escola recorre a procedimentos arcaicos como o uso de folha A4, o processo de matrícula certamente acaba se tornando um processo moroso e com números limitados de actividades. A presente monografia consiste na Concepção e Implementação de um Sistema de Gestão de Matrículas: Estudo de Caso - Escola Comunitária Santa Montanha. Uma escola que até a chegada do pesquisador recorria ao uso de folha A4 para o registo de dados dos alunos para a sua posterior matrícula e atribuição de turmas. Um processo que acabava se tornando lento, visto que as actividades tinham de ser organizadas em fases. Para a concepção do sistema, usou-se a metodologia *RUP*, aliada ao *framework laravel* e a linguagem *PHP*. No final, foi desenvolvido um sistema *web* que permite a matrícula e/ou registo do aluno, registo do ano lectivo, classes, turmas, numero de vagas para cada turma, permite a impressão de relatório dos alunos matriculados e/ou inscritos.

Palavras-chave: concepção, implementação, sistema, gestão de matrículas

Abstract

The design of a registration management system has proved to be a very efficient alternative in the execution of tasks because of numerous advantages. To allow the registration process to take place efficiently, flexibly and in a short time, it is necessary to have a system that allows for the holding of a considerable number of activities. However, in situations where the school resorts to archaic procedures such as the use of A4 sheets, the enrollment process certainly ends up becoming a lengthy process with limited numbers of activities. This monograph consists of the design and implementation of an enrollment management system: a case study - Santa Montanha Community School. A school that, until the arrival of the researcher, used an A4 sheet to record student data for later registration and assignment of classes. A process that ended up becoming slow, as the activities had to be organized in phases. For the design of the system, was used the RUP methodology, combined with the PHP language and the laravel framework. In the end of the search, was developed a web system that allows the registration of the student, registration of the academic year, classes, classes, number of vacancies for each class, allows the printing of a report of the students registration.

Keywords: design, implementation, system, registration, management

CAPÍTULO I - INTRODUÇÃO

Vivemos num mundo globalizado e cada vez mais digitalizado, onde tanto as ferramentas de trabalho quanto as de aprendizagem mostram-se mais modernizadas, com vista a flexibilizar e encurtar o tempo da realização de certas actividades.

Ao longo dos tempos, o uso de sistemas de gestão académica tem se tornado mais eficaz devido ao seu fácil acesso por parte dos gestores das escolas, docentes e os respectivos estudantes. É nesta ordem de ideia que a presente Monografia expandiu o seu estudo na Concepção e Implementação do Processo de Gestão de Matrículas no intuito de trazer uma nova dinâmica no acto da matrícula dos alunos da Escola Comunitária Santa Montanha, no distrito de Marracuene, província de Maputo, a fim de facilitar as actividades e reduzir o tempo gasto ao longo do processo de matrícula, visto que a concepção e implementação do processo de gestão de matrículas permitirá realizar inúmeras actividades num curto espaço de tempo, diferentemente do modo tradicional de inserção de dados no âmbito da realização da matrícula, em que as actividades são feitas de forma faseada, respeitando uma sequência de actividades e uso de um “extenso tempo”, como por exemplo, realizar a matrícula para noutra altura organizar as listas de turma, entre outras actividades, e todas elas realizadas manualmente.

1.1. Problematização

A Escola Comunitária Santa Montanha (ECOSAMO) é uma instituição de ensino Secundário do primeiro e segundo ciclo, isto é, lecciona desde a 8ª classe a 12ª classe. Cita no bairro Habel Jafar, Distrito de Marracuene.

Em todo início do ano lectivo, os alunos precisam passar pelo processo de matrícula para garantirem a sua vaga na escola. Este processo é feito mediante a apresentação de documentos na secretaria da escola como o certificado de habilitação literária da 7ª classe (para os candidatos a 8ª classe), ou o certificado de habilitação literária da 10ª classe (para os candidatos a 11ª classe) ou uma declaração que comprova a conclusão destas classes; bilhete de identificação pessoal; duas fotos do tipo passe e o talão de depósito bancário com o valor estipulado pela escola referente ao pagamento da matrícula.

Após a apresentação dos documentos outrora mencionados, o funcionário da secretaria fornece o processo do aluno com informações a preencher, tais como: o nome completo, género, data de nascimento, nacionalidade, Distrito, Província, Morada, Bilhete de Identidade, Nome do Pai, Profissão, Telefone, Nome da Mãe, Profissão, Telefone, Encarregado de Educação, Profissão, Telefone. De seguida é entregue o processo já preenchido ao funcionário da secretaria, e este por sua vez escreve o nome completo do aluno, data de nascimento e a classe numa folha A4, feito isto, o aluno já matriculado recebe o comprovativo da matrícula e aguarda pela atribuição das turmas.

Enquanto isto, o funcionário da secretaria, leva a folha A4 com os dados dos alunos matriculados ao director pedagógico. Por sua vez o director pedagógico, serve-se dos dados presentes na folha A4 para a organização das turmas baseando-se nos dados dos alunos registados. Assim, o director pedagógico observa lista por lista, atribui as turmas aos alunos obedecendo o critério por idade (sendo 12 anos a idade mínima para 8ª classe e 15 anos para a 11ª classe) e por fim, as listas das turmas com os respectivos nomes dos alunos são afixadas na vitrina da escola.

1.2. Justificativa

A escolha do tema deve-se às constatações feitas no campo de estudo em relação ao processo de matrículas dos alunos, organização das turmas e atribuição das mesmas. Visto que, todo processo é realizado de forma manual, que, de certa forma, leva muito tempo e traz um impacto negativo no que diz respeito ao tempo de realização de todas as actividades.

No âmbito académico, o presente estudo poderá não apenas servir de apoio para futuros estudos que tenham relação com a concepção e implementação do processo de gestão de matrículas, isto é, servirá de fonte de informação ou base para futuras pesquisas, como também, contribuir no enriquecimento académico e desempenho profissional de quem estiver interessado no estudo do tema.

No âmbito social, o estudo poderá permitir a análise do impacto da implementação de um sistema informático de gestão do processo de matrículas, o que conseqüentemente poderá agregar valores na gestão do processo de matrículas na área em estudo e melhorar no desempenho profissional do pessoal da secretaria, ou seja, o pessoal envolvido no processo de matrículas.

No âmbito económico, poderá permitir a flexibilização das actividades, permitindo a realização de inúmeras actividades num curto espaço de tempo, para além de ajudar a escola a economia em recursos materiais como a compra de folha A4, canetas, canalizando o valor para outras áreas que se considerem de extrema necessidade.

Outrossim, no âmbito pessoal o estudo recai também sob o facto de a pesquisa proporcionar vários benefícios para o autor como obtenção do grau superior de licenciatura em informática, trabalhar e ter experiência com a actividade de gestão de matrículas nas escolas.

1.3. Objectivos

1.3.1. Objectivo Geral

- Conceber e implementar um sistema de gestão de matrícula para a Escola Comunitária Santa Montanha.

1.3.2. Objectivos Específicos

- Identificar as dificuldades enfrentadas no processo de matrículas na Escola Comunitária Santa Montanha.
- Identificar mecanismos que possam ajudar na gestão de matrículas na Escola Comunitária Santa Montanha.

- Propor um meio tecnológico para flexibilizar a gestão de matrículas na Escola Comunitária Santa Montanha.

1.4. Questões de Pesquisa

A pesquisa cingiu-se nas seguintes questões de pesquisa:

- Quais são as dificuldades enfrentadas no processo de matrículas na Escola Comunitária Santa Montanha?
- Qual mecanismos podem ajudar na gestão de matrículas na Escola Comunitária Santa Montanha?
- Que meio tecnológico pode flexibilizar a gestão de matrículas na Escola Comunitária Santa Montanha?

1.5. Hipóteses

H1: A escola gasta muito tempo para suprir com as actividades concebidas no acto de matrícula, como a recolha de dados, preenchimento no processo do aluno, organização das listas provisórias por classe e idade, e por fim organização definitiva e atribuição das turmas.

H2: O mecanismo que podem ajudar na gestão de matrículas na Escola Comunitária Santa Montanha é um sistema informático capaz de integrar os dados dos alunos no processo e que possa atribuir para cada classe, turmas de acordo com a idade, sem que haja a necessidade de preencher em papéis.

H3: A implementação de um sistema de gestão de matrículas pode flexibilizar o processo de gestão de matrículas na Escola Comunitária Santa Montanha, facilitando e reduzindo o tempo de trabalho.

1.6. Metodologia de Trabalho

No tocante a metodologia de trabalho, a presente Monografia baseou-se na leitura de diferentes literaturas de diversos autores que deram suporte à realização deste trabalho com os conteúdos por eles abordados em suas obras ou seus trabalhos que estão directamente relacionados com o tema em estudo.

1.6.1. Tipo de Pesquisa

O presente trabalho, trata-se de uma pesquisa aplicada uma vez que visa gerar conhecimentos para a aplicação prática direccionada à soluções de dificuldades específicas, para além de ajudar a contribuir em fins práticos, com o foco de gerar soluções rápidas, versando sobre os pontos mais importantes em relação a necessidade da solução a implementar.

1.6.2. Delimitação do Universo

No que diz respeito a delimitação do universo, o presente trabalho envolveu funcionários da secretaria e o director pedagógico da Escola Comunitária Santa Montanha.

1.6.3. Técnicas de recolha de dados

Como técnica de recolha de dados, recorreu-se ao uso das seguintes:

- a) **A observação directa**, que resultou na presença do pesquisador no período da realização das matrículas entre o período de 05 a 31 de Janeiro de 2022, observando de forma directa o decorrer do processo de inscrição dos alunos até a atribuição das turmas.
- b) **O questionário** feito a funcionários da secretaria envolvidos no acto da matrícula e a direcção da escola, neste caso o director pedagógico.

O questionário usado foi constituído por um total de 7 (sete) questões, sendo quatro (4) abertas e três (3) fechadas organizadas de forma aleatória. Estas questões foram respondidas por escrito e sem a presença do pesquisador.

A opção pela diferenciação das questões deve-se ao facto da qualidade de informação que se pode apurar delas. As perguntas abertas permitiram a recolha de mais diversificadas informações. Diferentemente das perguntas fechadas que não permitem argumentos. As perguntas fechadas serviram apenas de guia de confirmação da informação que se pretendeu colher.

1.7. Material Necessário

Tabela1: Material Necessário

Designação	Quantidade unidade	Valor Unitário	Valor Total
Resmas	2	180,00mt	360,00mt
Canetas	5	10,00mt	50,00mt
Lápis	5	5,00mt	25,00mt
Comunicação e Internet	5 Recargas	500,00mt	2.500mt
Borrachas	5	10,00mt	50,00mt
Digitação	30 Páginas	20,00mt	600,00mt
Impressão	210 Páginas (7 impressões)	5,00mt	1050,00mt
Lanches	10	100,00mt	1000,00mt
Transporte	6 Voltas	50,00mt	300,00mt
Total	75 Unidades	_____	5,935,00mt

Fonte: Autor

1.8. Organização do Trabalho

O presente trabalho está organizado em três (3) capítulos, obedecendo a descrição que se segue:

Capítulo I: O primeiro capítulo é composto pela introdução, problematização, justificativa, pelo objectivo geral e os objectivos específicos, questões de pesquisa, hipóteses, metodologia do trabalho, tipo de pesquisa, delimitação do universo, técnicas de recolha de dados, material necessário e organização do trabalho.

Capítulo II: O segundo capítulo diz respeito à revisão bibliográfica. Neste capítulo são apresentados conceitos e fundamentos teóricos, ou seja, nesta fase são apresentadas as reflexões de diversos autores que escreveram sobre o tema, que de certa forma está relacionado com o problema em estudo.

Capítulo III: No terceiro capítulo é apresentado a análise e discussão dos resultados. É neste capítulo onde são apresentadas as limitações enfrentadas pela Escola Comunitária Santa Montanha, bem como os passos subsequentes em relação a concepção e implementação do sistema proposto pelo pesquisador.

CAPÍTULO II - REVISÃO BIBLIOGRÁFICA

2.1. Gestão

Em algum momento a noção da gestão confunde-se com a noção da administração, assim, PARO (1991) citado por DOURADO (2012) define gestão como “o processo de planejar para organizar, dirigir e controlar recursos humanos, materiais, financeiros e informacionais visando à realização de objectivos com base na utilização racional de recursos para a realização de fins determinados.

Gestão é apresentada como um processo dentro da acção administrativa, outras vezes apresentam-se como sinónimos...em outros momentos, gestão aparece como uma nova alternativa para o processo político-administrativo da educação.” (*idem.*, p. 24)

Em consonância com as ideias expressas pelo autor a cima citado, pode-se verificar que a gestão é imprescindível para o bom funcionamento de uma instituição, tal como subsidia CARLETO (2009) ao afirmar que “o termo gestão é utilizado para denotar uma nova forma de trabalho administrativo.”

Com isso, leva-se a crer que tanto em uma escola, quanto em qualquer outra instituição, o gestor não actua de forma isolada, porém em uma equipe agregada, empenhada e constituída por diversos actores.

Entretanto a gestão escolar não deve ser vista apenas no sentido administrativo, mas também no sentido pedagógico, permitindo desse modo uma análise geral do ambiente escolar, desta forma, SCHULTZ (2016) propõe que o gestor deverá compreender que as organizações são sistemas abertos que interagem com o ambiente, que existem elementos que compõem os diferentes tipos de ambientes e que avaliar a intensidade da complexidade é essencial, como etapas a serem cumpridas, para a adequada prática da gestão na busca da eficácia organizacional.

Assim, percebe-se que a escolha das ferramentas de análises diferenciadas para contextos diferentes torna-se fundamental para uma boa gestão, pois num ambiente em que haja situações de actuações diferenciadas, torna-se necessário que se tenham abordagens não apenas teóricas, mas também práticas que deão conta aos diversos instrumentos de trabalho.

Conforme TAUCHEN *et al.*, (2013), O objectivo da gestão é a promoção de aprendizagens efectivas e significativas aos sujeitos escolares, contribuindo para o desenvolvimento de competências demandadas pela vida em sociedade.

2.2. Matrículas

A palavra matrícula apresenta em um todo, a ideia de inscrição oficial. Trata-se da incorporação de um aluno a um centro de ensino qualquer como escola, colégio, faculdade, entre outros.¹

Em consonância com excerto retirado do regulamento do ensino secundário, artigo 52, ponto 1 e 2, a matrícula, na ESG, realiza-se na classe inicial de cada um dos ciclos, nomeadamente, na 8ª (para o 1º ciclo do ESG) e 11ª Classes (para o 2º ciclo do ESG). Nas restantes classes de cada ciclo, haverá apenas renovações de matrículas. As matrículas, nas classes iniciais de cada ciclo, são feitas nas duas últimas semanas que precedem a preparação de cada ano lectivo, salvo instruções em contrário por Despacho do Ministro da Educação.

2.3. Gestão de Matrículas Escolares

Para DOURADO (2012) “Discutir gestão escolar leva ao conceito de administração em geral... pois as transformações económicas e tecnológicas, bem como os princípios, funções e maneira de gerir interferem nas práticas sociais e educacionais”

A gestão escolar, tal como a gestão de qualquer outra instituição, pressupõe o controlo de todas as actividades relacionadas com aquela instituição e para o seu bom funcionamento.

A gestão da educação é um processo político administrativo contextualizado, por meio do qual a prática social da educação é organizada, orientada e viabilizada”. (BORDIGNON; 2004).

Portanto, falar da gestão de matrículas escolares, é falar da administração do processo de inscrição dos alunos, desde a observação do cumprimento das exigências necessárias, tais como a reunião de requisitos necessários para o processo (como certificado de habilitação literária da classe anterior, bilhete de identificação pessoal, talão de depósito, duas fotos tipo passe), inscrição na ficha de registro de dados, até a certificação da conclusão do processo de matrículas, tudo isto, considerando as pessoas envolvidas em todo o processo (o pessoal responsável pela matrícula, o director pedagógico, entre outros)

¹ <https://conceitos.com/matricula/>

2.4. Engenharia de Software

SOMMERVILLE (2011), Engenharia de software é uma disciplina de engenharia cujo foco está em todos os aspectos da produção de software, desde os estágios iniciais da especificação do sistema até sua manutenção, quando o sistema já está sendo usado.

Engenharia de software é uma abordagem sistemática para a produção de software; ela analisa questões práticas de custo, prazo e confiança, assim como as necessidades dos clientes e produtores do software. A forma como essa abordagem sistemática é realmente implementada varia dramaticamente de acordo com a organização que esteja desenvolvendo o software, o tipo de software e as pessoas envolvidas no processo de desenvolvimento. Um sistema de software desenvolvido profissionalmente normalmente consiste em uma série de programas separados e arquivos de configuração que são usados para configurar esses programas. Isso pode incluir documentação do sistema, que descreve a sua estrutura; documentação do usuário, que explica como usar o sistema; e sites, para usuários baixarem a informação recente do produto.

(*idem*)

Geralmente ao falar-se de *software*, é comum associar esta palavra à programas de computadores, porém ao falar da engenharia de software, não se trata apenas do programa, mas também de todo o conjunto de informações a ele associados e a dados de configuração indispensáveis para que este programa funcione tal como o esperado.

Portanto, pode-se perceber que ao desenvolver um programa deve-se ter em conta a finalidade, isto é, caso desenvolva-se um programa para o uso individual, não haverá necessidade de produzir com ele uma documentação descritiva para o usuário, mas se temos em conta um potencial usuário, haverá a necessidade de fornecer um documento que sirva de guia de utilizador.

2.5. Engenharia Web

Uma aplicação Web é um *software* de computador no sentido que é uma colecção de instruções executáveis e dados que oferecem informações e funcionalidades para usuários finais.

Assim, para BEDER (2012) A Engenharia Web pode ser definida como a utilização de princípios, conceitos e métodos da Engenharia de Software, de uma maneira

que se adaptem às características inerentes de aplicações Web, tais como Desempenho, Disponibilidade, Evolução Contínua, Urgência e Segurança

O conceito de BEDER (2012) não foge muito do conceito de PRESSMAN (2003), já que este define-a como “sistemas caracterizados por disponibilizar grande quantidade de conteúdo e funcionalidade para grande população de usuários. A Engenharia para a *Web* é, portanto, o processo utilizado para criar aplicativos *Web* de alta qualidade.”

Outro sim, LANCERDA (2013) define a engenharia web como “um termo utilizado para referir as diversas actividades relacionadas com a criação e manutenção de web sites, onde as actividades realizadas no âmbito do desenvolvimento web são, essencialmente, o desenho, a programação e a publicação.

Com a definição do autor acima citado, pode-se concluir que a engenharia web é um conjunto de actividades que podem ser aplicadas tanto para a criação de uma simples página fixa, quanto para a criação de uma aplicação um tanto complexa que possa ser acedida por inúmeros utilizadores em todo o mundo.

O que quer com isso dizer que na engenharia web deve-se ter em conta dois aspectos essenciais, nomeadamente o servidor e o cliente que é o consumidor final.

2.6. Modelo de Processos

De acordo com SOMMERVILLE (2011), um modelo de processo de *software* é uma representação simplificada de um processo de software. Cada modelo representa uma perspectiva particular de um processo e, portanto, fornece informações parciais sobre ele.

Assim, com base nos pressupostos do autor acima citado, um modelo de actividade do processo pode mostrar as actividades e sua sequência, todavia não mostra os papéis das pessoas envolvidas.

Por sua vez PRESSMAN (2002) pressupõe fases preponderantes do modelo de processos que vão desde a formulação, planeamento, análise, projecto, testes, gerência de projecto e configuração, que descreve da seguinte maneira:

- a) **Formulação:** o desenvolvedor deve procurar identificar os objectivos da aplicação, os pontos chaves, além de definir o perfil do utilizador final.
- b) **Planeamento:** o desenvolvedor deve avaliar os riscos e custos associados ao desenvolvimento da aplicação.

- c) **Análise:** o desenvolvedor deve verificar se o conteúdo a ser oferecido pelo aplicativo será definido nesta etapa, assim como cada detalhe das funções, das operações, da configuração onde a aplicação reside e das maneiras como o utilizador poderá interagir com o sistema.
- d) **Projecto:** o desenvolvedor deve definir uma arquitectura para a aplicação que tem a capacidade de evolução ao longo do tempo.
- e) **Testes:** os testes feitos para web exigem um esforço maior, pois, os aplicativos podem ser acessos utilizando diferentes *browsers*, sistemas operacionais, plataformas de hardware, além de estarem disponíveis para um número não definido de usuários.
- f) **Gerência de projecto e a gerência de configuração:** devem ser aplicadas a todas as etapas anteriores, pois, possibilitam o controlo tanto do conteúdo quanto do pessoal envolvido no desenvolvimento.

Ao analisarmos o processo de *software* como o número de actividades relacionadas que levam à produção de um produto de *software*, é necessário considerar que essas actividades podem envolver o desenvolvimento de software a partir do zero em uma linguagem padrão de programação.

Existem diferentes processos de *software*, mas todos devem incluir quatro actividades fundamentais para a engenharia de *software*, tal como recomenda SOMMERVILLE (2011:33), ao enumerar as actividades a serem consideradas, nomeadamente:

Especificação de *software*: a funcionalidade do *software* e as restrições a seu funcionamento devem ser definidas. **Projecto e implementação de *software*:** o *software* deve ser produzido para atender às especificações. **Validação de *software*:** o *software* deve ser validado para garantir que atenda às demandas do cliente. **Evolução de *software*:** o *software* deve evoluir para atender às necessidades de mudança dos clientes.

2.7. Metodologias de Desenvolvimento de *Software*

Segundo OLIVEIRA & SEABRA (2015) As metodologias de desenvolvimento de *software* são criadas por meio da utilização coordenada de métodos, ferramentas e procedimentos, com foco na produção de um produto de *software* e seus artefactos relacionados.

Estas metodologias buscam o aperfeiçoamento da utilização dos recursos, visando reduzir o tempo de execução e a melhoria da qualidade do produto final.

Para a materialização do trabalho, contou-se com o uso da metodologia de desenvolvimento que a seguir apresenta-se:

2.7.1. Rational Unified Process – RUP

Rational Unified Process – RUP, que na língua portuguesa significa “Processo Unificado Racional”, é um processo proprietário de Engenharia de *Software* criado pela *Rational Software Corporation*, que fornece técnicas para o desenvolvimento de *software* com o objectivo de ampliar a sua produtividade no processo de desenvolvimento.

Segundo PERSSMAN (2002), O RUP organiza o desenvolvimento de *software* em 5 fases a saber:

Concepção ou iniciação – diz respeito a interacção com o cliente e o planeamento do projecto tendo em conta os riscos, as prioridades, custos, e prazos estabelecidos. **Elaboração** – é feita a modelagem, a análise de forma detalhada do problema dos dados colhidos na fase da elaboração. **Construção** – é a parte correspondente ao desenvolvimento do *software*, isto é, é a fase do desenvolvimento das componentes e os respectivos recursos do sistema. **Transição** – corresponde a entrega do *software* ao respectivo cliente, isto é, é disponibilizado o sistema para testes, relatórios de feedback sobre os defeitos para posteriores modificações. **Produção** – é feita o monitoramento e fornecido o suporte. Esta fase coincide com a fase da implantação, onde é feito o relatório de defeito e pedidos de alterações.

2.8. Linguagens de Modelação UML

UML é uma tentativa de padronizar a modelagem orientada a objectos de forma que qualquer sistema, seja qual for o tipo, possa ser modelado correctamente, com consistência, fácil de se comunicar com outras aplicações, simples de ser actualizado e compreensível. Existem várias metodologias de modelagem orientada a objectos que até o surgimento da UML causavam uma guerra entre a comunidade de desenvolvedor orientado a objectos. A UML acabou com esta guerra trazendo as melhores ideias de cada uma destas metodologias, e mostrando como deveria ser a migração de cada uma para a UML.²

² Apostila UML linguagem de modelagem unificada em português acesso em <https://www.apostilauml.pdf...> a 18.04.2022

2.8.1. Diagrama de Caso de Uso

Descreve a funcionalidade proposta para um novo sistema que será projectado, é uma excelente ferramenta para o levantamento dos requisitos funcionais do sistema. Pode-se dizer que um *caso de uso* é um "documento narrativo que descreve a sequência de eventos de um actor que usa um sistema para completar um processo".

A descrição do diagrama de caso de uso do sistema proposto, pode ser vista adiante, na página 24.

2.8.2. Diagrama de Classe

O diagrama de classes mostra a estrutura das classes representadas pelas “peças” a serem administradas pela aplicação. Num diagrama de classe, as classes podem relacionar-se umas com as outras por meio de vários mecanismos, como a conexão, dependência, agrupamento, entre outros.

2.9. Frameworks de Desenvolvimento Web

A utilização de *frameworks* reduz os custos, visto que possuem estruturas definidas, onde o desenvolvimento concentre-se na implementação de regras já preestabelecidas.

Como subsidiam WILLEMANN & IBARRA (2007) ao afirmarem que um *framework* “é uma estrutura de suporte definida em que um outro projecto de *software* pode ser organizado e desenvolvido. Um *framework* pode incluir programas de suporte, bibliotecas de código, linguagens de *script* e outros *softwares* para ajudar a desenvolver e juntar diferentes componentes de um projecto de *software*.”

Os *framework* permitem a reutilização de códigos e a factoração de problemas em configurações comuns a várias aplicações, permite também obter sistemas com códigos menos frágeis e com menos defeitos.

Na visão dos autores acima citados, para que um *framework* ofereça um bom suporte ao domínio de aplicações a que se propõe, deve buscar implementar algumas características que irão contribuir muito para o aumento da qualidade do *framework*, entre elas, pode-se citar como principais:

- a) **Alterabilidade:** Repercute a capacidade do *framework* mudar suas funcionalidades em função da necessidade de uma aplicação específica sem que estas alterações resultem em consequências imprevistas no conjunto de sua estrutura.

- b) **Generalidade:** Reflecte a capacidade do *framework* dar suporte às muitas aplicações diferentes de um mesmo domínio, sendo flexível o suficiente para que as características de alterabilidade e extensibilidade possam ser aplicadas.
- c) **Extensibilidade:** reflecte a capacidade do *framework* ampliar sua funcionalidade sem consequências imprevistas no conjunto de sua estrutura, está ligada directamente à manutenção do *framework*, permitindo que sua estrutura evolua por toda sua vida útil, pois à medida que vai sendo utilizado, outros novos recursos vão sendo agregados para que ele se ajuste as novas aplicações a que dá suporte.
- d) **Clareza:** A interface pública das classes do *framework* deve ser tão simples quanto possível, visto que são nas interfaces públicas que o desenvolvedor estará trabalhando, é importante mantê-las não mais complexas que o necessário para que se possa alcançar a funcionalidade desejada.
- e) **Simplicidade:** A estrutura geral do *framework* deve ser de fácil compreensão para que assim o desenvolvedor possa aprendê-lo em pouco tempo. Obviamente todos os detalhes do projecto não poderão ser aprendidos em poucos dias, no entanto, o desenvolvedor deve estar apto para entender seu funcionamento em pouco tempo, deixando o aprendizado de seus detalhes para o decorrer de sua utilização.

Contudo, o presente trabalho irá cingir-se no *framework* **LARAVEL**, versão 8.0.

2.9.1. Laravel

De acordo com STAUFFER (2017) citado por PACHECO (2019) O *Laravel* é um *framework* PHP criado por Taylor Otwell em Junho de 2011 com o objectivo de facilitar a criação de projectos de forma rápida deixando o código mais limpo e com uma boa aprendizagem, facilitando assim manutenções.

O *framework* *Laravel* possui arquitectura MVC (*Model-View-Controller*) e tem, como aspecto principal, o de auxiliar no desenvolvimento de aplicações de forma ágil e simplificada incentivando à boas práticas de programação e utilização de padrões.

O motivo da escolha deste *framework* deve-se as suas inúmeras vantagens, das quais vale mencionar algumas, tais como:

- Define regras que permitam associar caminhos a controladores
- Permite associar caminhos a controladores para que o *framework* mapeie automaticamente os métodos REST (*Representational State Transfer*) presentes nesses controladores.

- Permite adicionar novos métodos a estes controladores, contudo o mapeamento tem de ser realizado manualmente no ficheiro “*routes.php*”.
- Permite associar funcionalidades específicas (métodos) a controladores que estejam declarados nesse mesmo controlador, desde que o nome siga uma determinada nomenclatura, geralmente por um nome com o prefixo do método HTTP.
- Aborda medidas de segurança importantes como autenticação por omissão, isto é, o *framework* utiliza o Eloquent como *driver* de autenticação, agregado a um “Modelo” chamado *User*.

O *Laravel* é um *framework web* com sintaxe expressiva e elegante. O *Laravel* visa tornar o processo de desenvolvimento agradável para o desenvolvedor, sem sacrificar a funcionalidade do aplicativo. O *Laravel* é acessível e poderoso, fornecendo ferramentas poderosas necessárias para aplicativos grandes e robustos³.

2.9.2. Django

Django é um *framework web* de alto nível, que permite o rápido desenvolvimento de sites seguros e de fácil manutenção.⁴

2.9.3. Codeigniter

Codeigniter é um *framework PHP* focado na produtividade, ele utiliza a arquitectura *MVC* e se preocupa com o que o usuário vai ver.⁵

2.9.4. Comparação entre *Frameworks*

Tabela 2: Comparação entre *Frameworks*

	<i>Frameworks</i>		
	<i>Laravel</i>	<i>Django</i>	<i>Codeigniter</i>
Linguagem	PHP	<i>Python</i>	PHP
ORM	<i>Eloquent</i>	Qualquer	Integrado
Arquitetura	MVC (<i>Model View Controller</i>)	MVT (<i>Model View Template</i>)	MVC (<i>Model View Controller</i>)

Fonte: Autor

³ <https://www.djangoproject.com/start/overview>. Site consultado em: 23 de Abril de 2022

⁴ <https://developer.mozilla.org/pt-BR/docs/Learn/Server-side/Django/Introduction> Site consultado em 23 de Abril de 2022

⁵ <https://dicas.codeigniter.com.br/o-que-e-codeigniter/> Site consultado em 23 de Abril de 2022

2.10. Linguagens de Programação

Para WILLEMANN & IBARRA (2007) uma linguagem de programação é um método padronizado para expressar instruções para um computador. É um conjunto de regras sintáticas e semânticas usadas para definir um programa de computador.

Uma linguagem permite que um programador especifique precisamente sobre quais dados um computador vai actuar, como estes dados serão armazenados ou transmitidos e quais acções devem ser tomadas sob várias circunstâncias.

(up cit.)

Uma das principais metas das linguagens de programação é permitir que programadores tenham uma maior produtividade, podendo expressar suas intenções mais facilmente do que quando comparado com a linguagem que um computador entende nativamente (código de máquina)

O presente trabalho acreditou nas linguagens de programação PHP e Java Script.

2.10.1. PHP

PHP é uma linguagem embutida no HTML e que reside no servidor em que o servidor *web* está rodando, isto significa que, o usuário através do *browser* faz a chamada de uma página PHP, o servidor web a processa e, depois, envia as informações de volta para o navegador do usuário.⁶

Segundo BOECHAT (2014) O PHP é uma linguagem de *script open source* projectada para o desenvolvimento web, mas também usada como uma linguagem de programação de propósito geral.

O PHP é uma linguagem que possibilita a criação de sites WEB dinâmicos, permitindo uma interacção com o usuário por meio de formulários, padrões da URL e links. Uma das diferenças de PHP com relação a linguagens semelhantes a *Javascript* é que o código PHP é executado no servidor, sendo enviado para o cliente o html completa.

⁶ <http://www.lyfreitas.com.br/ant/pdf/php.pdf>

2.10.2. Javascript (JS)

JS É uma linguagem de programação aplicada principalmente para desenvolvimento web e desenvolvimento de software, actua na programação *front-end*, a parte “visual” de uma aplicação.⁷

2.10.3. Comparação entre linguagens de Programação

Tabela 3: Comparação entre Linguagens de Programação

Linguagem	Modo de execução	Influências	Paradigma principal	Modelo de tipo de dados
C#	Interpretação e compilação	Java, C++	Orientada a objectos, múltiplos paradigmas	Estético, forte
Java	Interpretação e compilação	C++	Orientada a objectos	Estético, forte
PHP	Interpretação	C e Perl	Orientada a objectos	Dinâmico
Python	Interpretação	ABC, Perl	Orientada a objectos	Dinâmico e forte

Fonte: Fulgham & Gouy (2004)

2.11. Sistema de Gestão de Base de Dados (SGBD)

WILLEMANN & IBARRA (2007) sustentam que “uma base de dados é uma colecção de dados estruturados, que pode ser desde uma simples lista de compras a uma galeria de imagens ou a grande quantidade de informação da sua rede corporativa.

Desse modo, percebe-se que base de dados funciona como um repositório de informações, onde sempre que exigido, é possível buscar e identificar a informação necessária. A título de exemplo, os computadores são exemplos claros de conservação de inúmeros dados, pois a sua base de dados é funcional e estes lidam-se muito bem com grandes quantidades de dados.

2.11.1. Comparação entre SGBD

Tabela 4: Comparação entre SGBD

SGBD	Estrutura de dados	Estabilidade	Licença
MySQL	SQL	Complexo Vertical	GNU
Oracle	Multi-model, SQL	Vertical	Proprietário
PostgreSQL	Objecto-Racional SQL	Vertical	Open-soure

Fonte: Autor

⁷ <https://blog.betrybe.com/javascript/>

CAPITULO III – ANÁLISE E APRESENTAÇÃO DE RESULTADOS

3.1. Descrição do Local

A Escola Comunitária Santa Montanha (ECOSAMO), localizada no bairro – Habel Jafar, é uma instituição vocacionada para o Ensino Secundário Geral, de carácter social, com fins educativos e não lucrativos, leccionando o I e II ciclos. Sita no Bairro de Habel Jafar, Localidade de Michafutene, Distrito de Marracuene, Província de Maputo, em Moçambique. As instalações da ECOSAMO foram construídas com doação da Igreja Metodista da Coreia, Paróquia da Santa Montanha e é propriedade exclusiva da Missão da Coreia e da Igreja Metodista da Coreia.



Figura 1: Pátio da ECOSAMO

Fonte: Autor

3.2. Modelo Actual do Processo de Matrículas

Em todo início do ano lectivo, os alunos precisam passar pelo processo de matrícula para garantirem a sua vaga na escola. Este processo é feito mediante a apresentação de documentos na secretaria da escola como o certificado de habilitação literária da 7ª classe (para os candidatos a 8ª classe), ou o certificado de habilitação literária da 10ª classe (para os candidatos a 11ª classe) ou uma declaração que comprova a conclusão destas classes; bilhete

de identificação pessoal; duas fotos do tipo passe e o talão de depósito bancário com o valor estipulado pela escola referente ao pagamento da matrícula.

Após a apresentação dos documentos outrora mencionados, o funcionário da secretaria fornece o processo do aluno com informações a preencher, tais como: o nome completo, género, data de nascimento, nacionalidade, Distrito, Província, Morada, Bilhete de Identidade, Nome do Pai, Profissão, Telefone, Nome da Mãe, Profissão, Telefone, Encarregado de Educação, Profissão, Telefone. De seguida é entregue o processo já preenchido ao funcionário da secretaria, e este por sua vez escreve o nome completo do aluno, data de nascimento e a classe numa folha A4, feito isto, o aluno já matriculado recebe o recibo e aguarda pela atribuição das turmas.

Enquanto isto, o funcionário da secretaria, leva a folha A4 com os dados dos alunos matriculados ao director pedagógico. Por sua vez o director pedagógico, serve-se dos dados presentes na folha A4 para a organização das turmas.

Em suma, o sistema actual obedece o esquema que é representado a seguir na figura 4, abaixo apresentada:

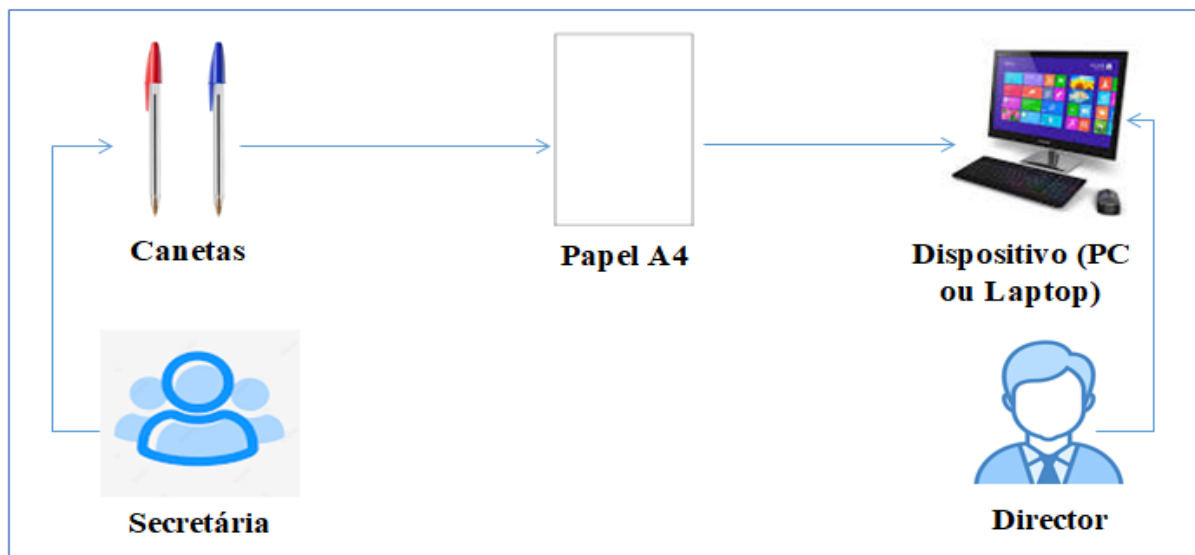


Figura 2: Modelo actual

Fonte: Autor

3.2.1. Limitações do Modelo Actual

Sendo o modelo actual baseado no registo das matrículas de forma obsoleta, sob ponto de vista informático, verificam-se as seguintes limitações:

- O modelo actual não garante segurança e flexibilidade no manuseio dos dados;

- Gasta muito tempo para suprir com as actividades concebidas no acto de matrícula, como a recolha de dados, preenchimento no processo do aluno, organização das listas provisórias por classe e idade, e por fim organização definitiva e atribuição das turmas;
- Recorre primordialmente aos dados registados em folha A4 o que pode comprometer no trabalho final, caso a folha A4 perca ou molhe.

3.3. Modelo Proposto

O modelo proposto será baseado num sistema *web* que terá dois níveis de acesso, em que o primeiro nível dirá respeito ao Director, que será responsável por gerir o ano lectivo, classes e as suas respectivas turmas, alunos, matrículas, relatórios e o nível de secretaria. O segundo nível corresponderá à secretaria, que será responsável apenas por gerir alunos e as matrículas.

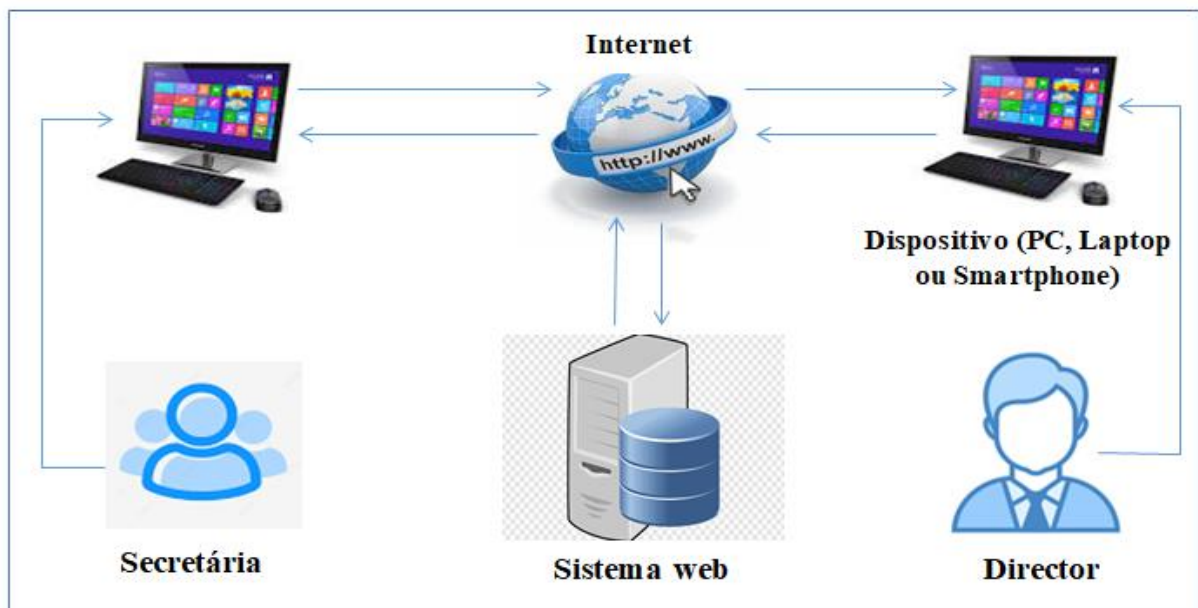


Figura 3: Modelo proposto

Fonte: Autor

3.4. Levantamento de Requisitos do Sistema Proposto

3.4.1. Requisitos do Utilizador

O sistema tem como requisitos do utilizador, os seguintes:

- O sistema deve ser capaz de enviar código para o *email* do utilizador para a definição de uma nova senha, caso esse tenha esquecido a anterior.
- O sistema deve permitir que o Director e os funcionários da secretaria, façam a gestão (registo, lista, remoção, edição e visualização) dos alunos e matrículas;

- O sistema deve permitir que apenas o Director faça a recuperação de dados do aluno, matrículas anteriormente removidas, bem como, registo, edição e remoção de usuários do sistema, do ano lectivo, das classes e das turmas de acordo com número de vagas para o ano lectivo;
- O sistema deve permitir que apenas o Director visualize o relatório dos alunos e das matrículas, assim como, imprimir;
- O sistema deve permitir a visualização das listas das turmas dos alunos após a matrícula e a impressão das mesmas;
- O sistema deve permitir que o relatório seja visualizado através das listas, gráficos de barra e circular

3.4.2. Fase de Planeamento

3.4.2.1. Actores

Actores São as pessoas, os intervenientes que terão o acesso ao sistema, sendo classificados como os usuários, sendo que cada um possuirá uma função específica, de acordo com o papel que lhe será atribuído, tal como ilustra a tabela que se segue:

Tabela 5: Actores

Actor	Desenvolvimento
Director	Utilizador responsável pelo controlo total da aplicação.
Secretaria	Utilizador responsável por gerir alunos e as respectivas matrículas.

Fonte: Autor

3.4.2.2. Requisitos Funcionais

Os requisitos funcionais descrevem as funcionalidades que cada tipo de usuário deve ser capaz de aceder no sistema. Esses requisitos estão relacionados às actividades que o sistema realiza.

Tabela 6: Gerir Usuário

[RF01] Gerir Usuário	
Actor(es)	Director
Prioridade	Alta
Requisitos associados	
Descrição	O sistema deve permitir registar, visualizar, editar e remover dados do usuário

Fonte: Autor

Tabela 7: Gerir Ano Lectivo

[RF02] Gerir Ano Lectivo	
Actor(es)	Director
Prioridade	Alta
Requisitos associados	
Descrição	O sistema deve permitir registar, editar e remover o ano lectivo

Fonte: Autor

Tabela 8: Gerir Classe

[RF03] Gerir Classe	
Actor(es)	Director
Prioridade	Alta
Requisitos associados	[RF02]
Descrição	O sistema deve permitir registar, editar e remover dados da classe

Fonte: Autor

Tabela 9: Gerir Turma

[RF04] Gerir Turma	
Actor(es)	Director
Prioridade	Alta
Requisitos associados	[RF03]
Descrição	O sistema deve permitir registar, editar e remover dados da matrícula

Fonte: Autor

Tabela 10: Gerir Aluno

[RF05] Gerir Aluno	
Actor(es)	Director, Secretaria
Prioridade	Alta
Requisitos associados	[RF04]
Descrição	O sistema deve permitir registar, visualizar, editar e remover os dados do aluno

Fonte: Autor

Tabela 11: Gerir Matricula

[RF06] Gerir Matricula	
Actor(es)	Director, Secretaria
Prioridade	Alta
Requisitos associados	[RF05]
Descrição	O sistema deve permitir registar, visualizar, editar, imprimir listas das turmas e remover os dados da matrícula

Fonte: Autor

Tabela 12: Gerir Relatório

[RF07] Gerir Relatório	
Actor(es)	Director
Prioridade	
Requisitos associados	[RF06]
Descrição	O sistema deve permitir visualizar dados do aluno, matricula e imprimir

Fonte: Autor

3.4.2.3. Diagrama de Caso de Uso

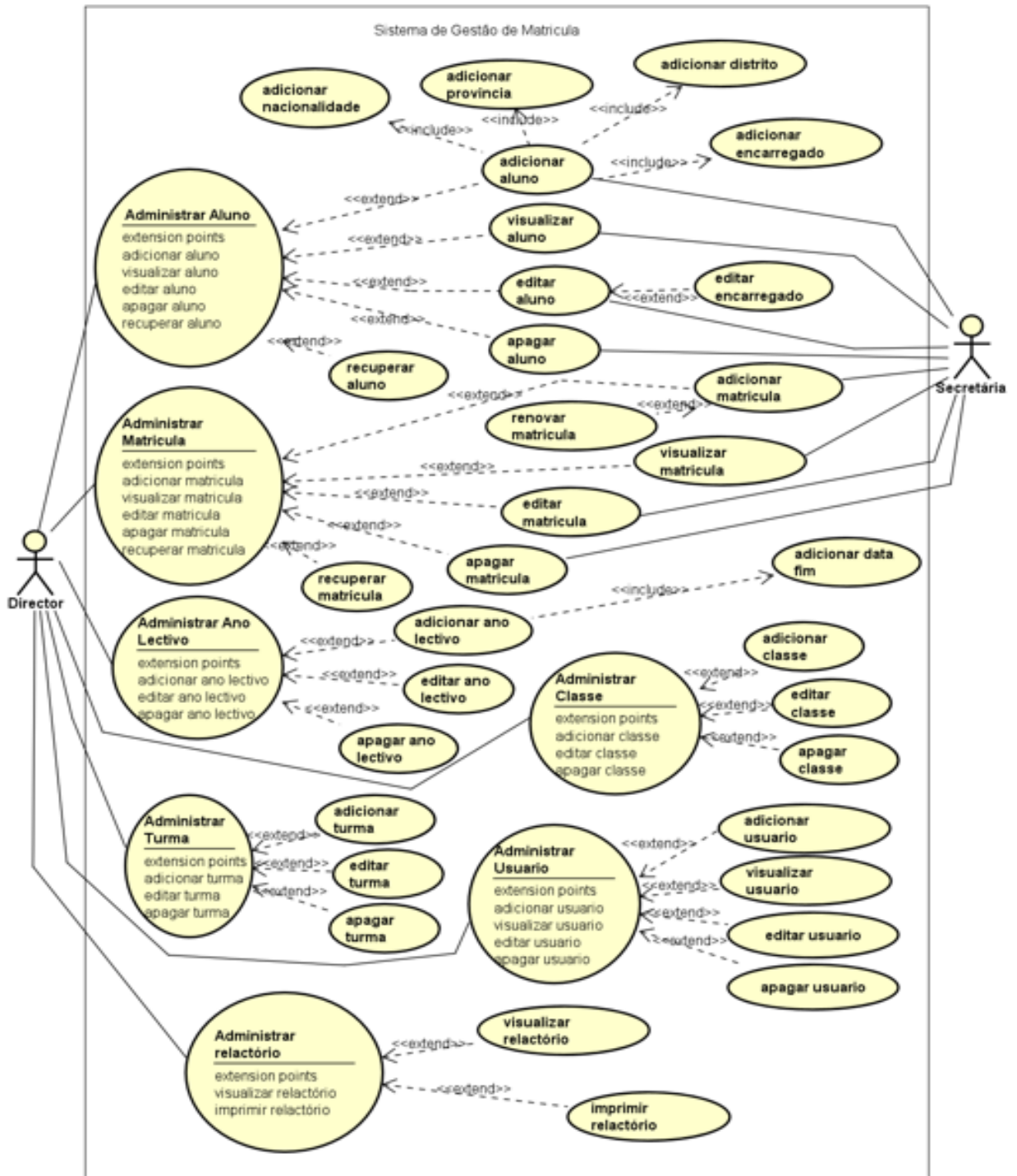


Figura 4: Diagrama de caso de uso
 Fonte: Autor

3.4.2.4. Requisitos Não-funcionais

Os requisitos não-funcionais são aqueles que não se relacionam directamente com as funções do sistema, no entanto, são restrições impostas aos serviços oferecidos pelo sistema.

Tabela 13: Efectuar acesso

[RNF01] Efectuar acesso	
Prioridade	
Descrição	Os utilizadores não devem usar o sistema sem antes estarem autenticados

Fonte: Autor

Tabela 14: Período de funcionamento

[RNF02] Período de funcionamento	
Prioridade	
Descrição	O sistema estará disponível 24h por dia

Fonte: Autor

Tabela 15: Período de Matrícula

[RNF03] Período de Matrícula	
Prioridade	
Descrição	O período da matrícula será amostrado e nenhuma delas será permitida após o prazo.

Fonte: Autor

3.4.2.5. Análise e Projecto

3.4.2.5.1. Diagrama de Classe

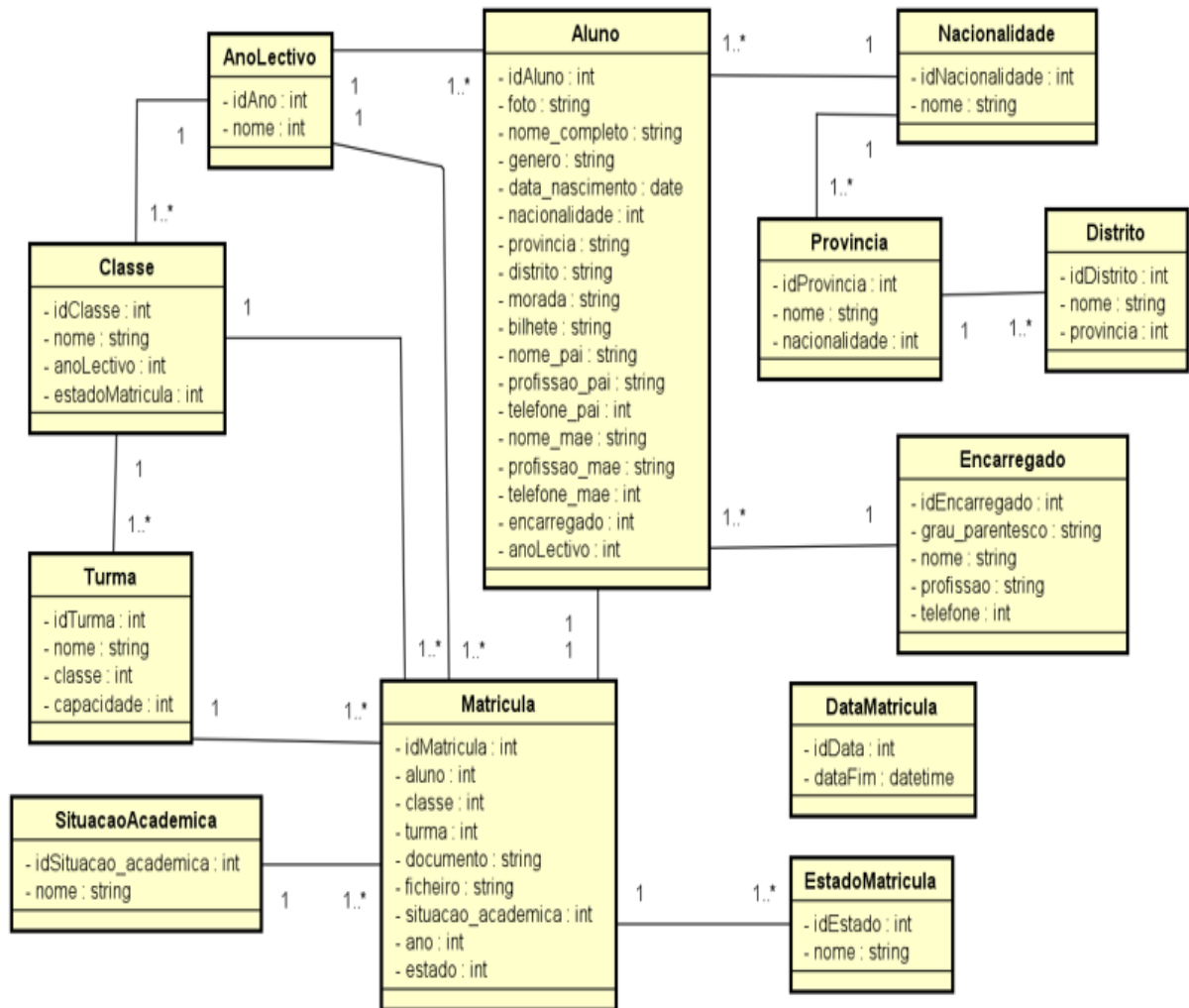


Figura 5: Diagrama de classe

Fonte: Autor

3.4.3. Fase de Desenvolvimento

Nesta fase focou-se na construção do código da aplicação com base na arquitectura MVC, onde utilizou-se a linguagem PHP para a codificação dos sistema, SQL Server para armazenamento dos dados. Utilizou-se também o ambiente do *Visual Studio Code* para o desenvolvimento do sistema.

3.4.3.1. Telas do Sistema

A figura 6 apresenta a tela de cadastro de aluno, filiação e encarregado do aluno, que servirão posteriormente para o cadastro de matrícula.

Figura 6: Tela de cadastro de aluno

Fonte: Autor

A figura 7 apresenta a tela de cadastro de matrícula, nesta tela são necessários os preenchimentos do nome do aluno cadastrado na figura 6, classe, turma, certificado ou declaração.

Figura 7: Tela de Matrícula

Fonte: Autor

Trecho do código que gera a tela de registo de matrícula em PHP

```
<!DOCTYPE html>
<html lang="en">
@include('layouts.header')
<body class="hold-transition sidebar-mini layout-fixed layout-navbar-fixed layout-footer-fixed">
```

```

<div class="wrapper">
  @include('layouts.nav-top')
  @include('layouts.sidebar')
  <div class="content-wrapper">
    <div class="content-header">
      <div class="container-fluid">
        <div class="row mb-2">
          <div class="col-sm-6">
            <h1 class="m-0 text-dark">@if(isset($editaMatricula)) Editar Matricula @else
            Adicionar Matricula @endif</h1>
          </div>
          <div class="col-sm-6">
            <ol class="breadcrumb float-sm-right">
              <li class="breadcrumb-item"><a href="#">Matricula</a></li>
              <li class="breadcrumb-item active">@if(isset($editaMatricula)) Editar @else Adicionar @endif</li>
            </ol></div></div></div></div>
          <section class="content">
            <div class="container-fluid">
              <div class="row">
                <div class="col-md-8">
                  <div class="card card-primary card-outline">
                    <div class="card-header">
                      <h1 class="card-title"><h4>@if(isset($editaMatricula)) Dados para editar o Aluno Matriculado @else
                      Dados para a Matricula do Aluno @endif</h4></h1></div>
                    <div class="card-body">
                      @if(isset($editaMatricula))
                      <form id="formMatricula" action="{ { url('matriculas/editar/$editaMatricula->cod_matricula') } }"
                      method="post" enctype="multipart/form-data">
                        @method('PUT')
                      @else
                      <form id="formMatricula" action="{ { route('matriculas.adicionar') } }" method="post"
                      enctype="multipart/form-data">
                        @endif
                      @csrf
                    <div class="card-header">
                      <div class="row">
                        <h3><div name="data_matricula" id="data_matricula"></div></h3>
                        @foreach($date as $d)
                        <input type="hidden" id="date" value="{{ $d->data_fim }}">
                        @endforeach</div></div><br>
                        <div class="form-group row">
                          <label for="inputComb" class="col-sm-2 col-form-label">Nome<code>*</code></label>
                          <div class="col-sm-10">
                            @if(isset($editaMatricula))
                            <input class="form-control" type="text" name="nomeA" value="{ { mb_strToUpper($editaMatricula->alunos->nome_aluno ?? ' ') } }" disabled="true">
                            @else
                            <select class="form-control selecMatriculado @error('nome_aluno') is-invalid @enderror"
                            name="nome_aluno" id="nome_aluno" style="width: 100%;">
                              <option value="0" selected="true">-- Seleccione o nome do Aluno --</option>
                              @foreach($seleccAlun as $selecAl)
                              <option value="{ { $selecAl->cod_aluno } }">{ { mb_strToUpper($selecAl->nome_aluno) } }</option>@endforeach</select>
                              @error('nome_aluno')
                              <span class="invalid-feedback" role="alert">
                                <strong>{ { $message } }</strong></span>@enderror@endif

```

```

<input type="text" class="form-control nome_aluno" name="nome_alun" id="nome_alun"
style="display:none;"></div></div><br>
<div class="form-group row">
<label for="inputClasse" class="col-sm-2 col-form-label">Classe <code>*</code></label>
<div class="col-sm-3">
<select class="form-control seleccioneClasse @error('classe') is-invalid @enderror" name="classe"
id="classe" style="width: 100%;">
<option value="{{ $editaMatricula->cod_classe ?? }}" selected="true">{{ $editaMatricula->classe-
>nome_classe ?? 'Seleccione a Classe' }}</option>
@foreach($selecClasse as $selecClass)
<option value="{{ $selecClass->cod_classe }}">{{ $selecClass->nome_classe }}</option>
@endforeach</select>
@error('classe')
<span class="invalid-feedback" role="alert">
<strong>{{ $message }}</strong></span>@enderror
<select class="form-control nome_claset" name="nome_claset" id="nome_claset"
style="display:none;"></select></div>
<label for="inputTurma" class="col-sm-2 col-form-label">Turma <code>*</code></label>
<div class="col-sm-3">
<select class="form-control seleccioneTurma @error('turma') is-invalid @enderror" name="turma"
id="turma" style="width: 100%;">
<option value="{{ $editaMatricula->cod_turma ?? }}" selected="true">{{ $editaMatricula->turma-
>nome_turma ?? 'Seleccione a Turma' }}</option></select>
@error('turma')
<span class="invalid-feedback" role="alert">
<strong>{{ $message }}</strong></span>@enderror</div>
<label for="inputVaga" class="col-sm-1 col-form-label">Vagas</label>
<div class="col-sm-1">
<input type="text" class="form-control vagaAluno" name="vaga" id="vaga"></div></div>
<div class="form-group row">
<div class="col-sm-10">
<select class="form-control seleccioneAno" name="cod_ano" id="cod_ano" style="width:
100%;display:none;">
@foreach($anolectivos as $ano)
<option value="{{ $ano->cod_ano }}">{{ $ano->nome_anolectivo }}</option>
@endforeach</select></div></div>
<div class="form-group row">
@if(isset($editaMatricula))
<label for="inputEstado" class="col-sm-1 col-form-label">Estado</label>
<div class="col-sm-3">
<select class="form-control selectAcad" name="cod_sitAcad" id="cod_sitAcad">
<option value="{{ $editaMatricula->situacaoAcad->cod_sit }}">{{ $editaMatricula->situacaoAcad-
>nome_sitAcad }}</option>
@foreach($sitAcademicas as $sitAcademica)
<option value="{{ $sitAcademica->cod_sit }}">{{ $sitAcademica-
>nome_sitAcad }}</option>@endforeach</select></div>@endif
<label for="inputTurma" class="col-sm-0 col-form-label"><code>*</code></label>
<div class="col-sm-4">
<select class="form-control selectDOC @error('documento') is-invalid @enderror"
name="documento" id="documento">
<option value="{{ $editaMatricula->documento ?? }}" selected="selected">{{ $editaMatricula-
>documento ?? 'Seleccione o Documento' }}</option>
<option value="CERTIFICADO">CERTIFICADO</option>
<option value="DECLARAÇÃO">DECLARAÇÃO</option></select>
@error('documento')

```

```

<span class="invalid-feedback" role="alert">
<strong>{{ $message }}</strong></span>@enderror</div>
<label for="inputTurma" class="col-sm-0 col-form-label"><code>*</code></label>
<div class="col-sm-3">
<input class="form-control @error('ficheiro') is-invalid @enderror" type="file" name="ficheiro"
id="ficheiro">
<input class="form-control" name="ficheiroAct" value="{{ $editaMatricula->ficheiro ?? '' }}"
style="display:none;">
@error('ficheiro')
<span class="invalid-feedback" role="alert">
<strong>{{ $message }}</strong></span>@enderror</div>
<div class="col-sm-2">
<input type="text" class="form-control data_nascimento" name="data_nascimento"
id="data_nascimento" style="display:none;"></div></div>
<div class="form-group row"></div>
<div class="form-group row">
<div class="offset-sm-9">
<button type="button" class="btn btn-danger" id="getRequest">Voltar</button>
@if(isset($editaMatricula))
<button type="submit" id="btn_matricula" class="btn btn-success">Atualizar</button>
@else
<button type="submit" id="btn_matricula" class="btn btn-primary">Matricular</button>
@endif</div></div></form></div></div></div>
@include('layouts.footer')
@include('layouts.script')
</body>
</html>

```

A figura 8 apresenta a tela dos alunos que pretendem renovar a matrícula, nesta tela são necessários os preenchimentos dos campos, ano de frequência, classe e a turma. Após o preenchimento desses todos campos, são listados os alunos abaixo que pretendem fazer a renovação de matrícula. Clicando no botão a direita de cada nome obtém-se a figura 9.

Matricula	Nome Completo	Genero	Nascimento	Classe	Turma	Ano	Ações
235	CIPRIANO ARMANDO LOPES	MASCULINO	2010-02-20	8ª CLASSE	A	2021	Renovar
236	ALICIA DA FLÁVIA LITOS	FEMININO	2010-03-21	8ª CLASSE	A	2021	Renovar
239	CALTON HUMBERTO JOSÉ	MASCULINO	2010-10-10	8ª CLASSE	A	2021	Renovar
246	BEATRIZ AFONSO MONDEREIA	FEMININO	2009-06-04	8ª CLASSE	A	2021	Renovar
251	ÂNGELA SALOMÃO MARULIA	FEMININO	2010-02-06	8ª CLASSE	A	2021	Renovar

Figura 8: Tela dos alunos que pretendem renovar a matrícula

Fonte: Autor

A figura 9 apresenta a tela de renovação de matrícula, nesta tela foi carregada a informação do aluno que pretende ser renovado, após o carregamento segue-se ao preenchimento do destino da renovação deste aluno, seleccionando o ano, classe e turma.

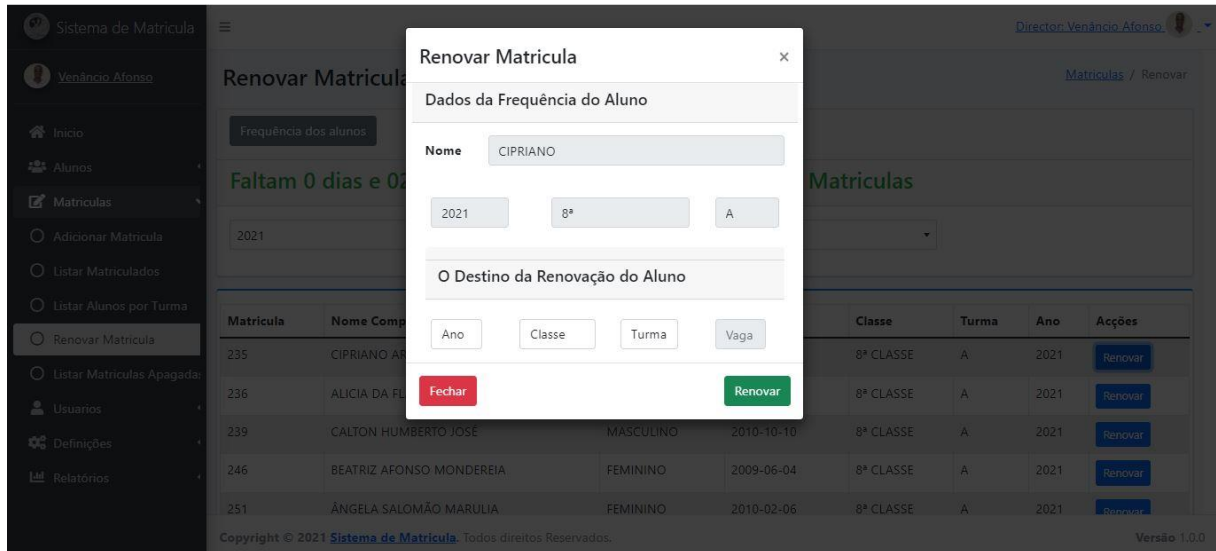


Figura 9: Tela de renovação de matrícula

Fonte: Autor

3.4.4. Fase de Estabilização

3.4.4.1. Caso de Teste

Após a autenticação do utilizador, para o caso de testes verifica-se o resultado nas tabelas abaixo.

Tabela 16: Cadastrar Aluno

[CT01] Cadastrar aluno
Descrição: Verifica se a aplicação está cadastrando dados do aluno.
Procedimento: <ol style="list-style-type: none"> 1. Abrir a tela de Aluno 2. Preencher dados do aluno, principalmente os dados obrigatórios. 3. Clicar no botão “Cadastrar Aluno”
Resultado: Aluno Cadastrado com sucesso

Fonte: Autor

Tabela 17: Cadastrar Matricula

[CT02] Cadastrar matrícula
Desenvolvimento: Verifica se a aplicação está cadastrando dados da matrícula
Procedimento: <ol style="list-style-type: none"> 1. Abrir a tela de matrícula 2. Preencher dados da matrícula, todos os dados são obrigatórios. 3. Clicar no botão “Matricular”
Resultado: Matrícula cadastrada com sucesso

Fonte: Autor

Tabela 18: Cadastrar Ano Lectivo, Classe e Turma

[CT03] Cadastrar Ano Lectivo, Classe e Turma
Descrição: Verifica se a aplicação está cadastrando ano lectivo, classes e turma
Procedimento: <ol style="list-style-type: none"> 1. Preencher o ano lectivo e clicar o botão “Ano” 2. Preencher a classe, seleccionar matrícula ou renovação e clicar no botão “Classe” 3. Preencher a turma, vaga e clicar no botão “Turma” 4. Preenche a data de fim, selecciona a data de fim da matrícula ou da renovação e clicar no botão “Data”
Resultado: Novo Ano Adicionado, Nova Classe Adicionada com Sucesso, Nova Turma Adicionada com Sucesso, Nova Data de Fim Adicionada com Sucesso

Fonte: Autor

3.5. Processo de implementação do sistema de gestão de matrícula

Todo o processo de implantação de um sistema deve ter em conta algumas etapas como o planeamento, análise, parametrização, customização, sensibilização, treinamento e avaliação. Ao falar do processo de implantação de um sistema de gestão de matrículas, pretende-se demonstrar os passos ou as etapas e os recursos obedecidos para a implantação de um dado sistema.

No que se refere a implantação do sistema de gestão de matrículas proposto, importa referir-se dos requisitos de *hardware* e de software usados.

3.5.1. Requisitos necessários para a implementação do sistema de gestão de matrículas

Para a implantação do sistema, primeiramente certificou-se, se o servidor onde seria hospedado o sistema correspondia à necessidade da escola, e que de certa forma foram fundamentais para a instalação e o uso correto do mesmo. Desse modo, foram levantados os requisitos de hardware e de software necessários, que a seguir são descritas:

3.5.2. Recursos de *hardware*:

Como recursos de *hardware*, importa mencionar os seguintes:

- 256MB de memória RAM
- 1 GB de espaço em disco duro (espaço mínimo)
- Placa de rede com acesso a internet

3.5.3. Recursos de *software*

São considerados recursos básicos do software do sistema implantado, os seguintes:

- PHP 7.1 (versão mínima recomendada) compilada com o suporte MySQL.
- Servidor Xampp
- MySQL e Framework LARAVEL
- Navegador: Microsoft Edge

3.5.4. Implementação do sistema

1º. Para a implementação do sistema, acedeu-se ao link <https://www.mozdomains.co.mz> do serviço de hospedagem, como ilustra a página do site abaixo.

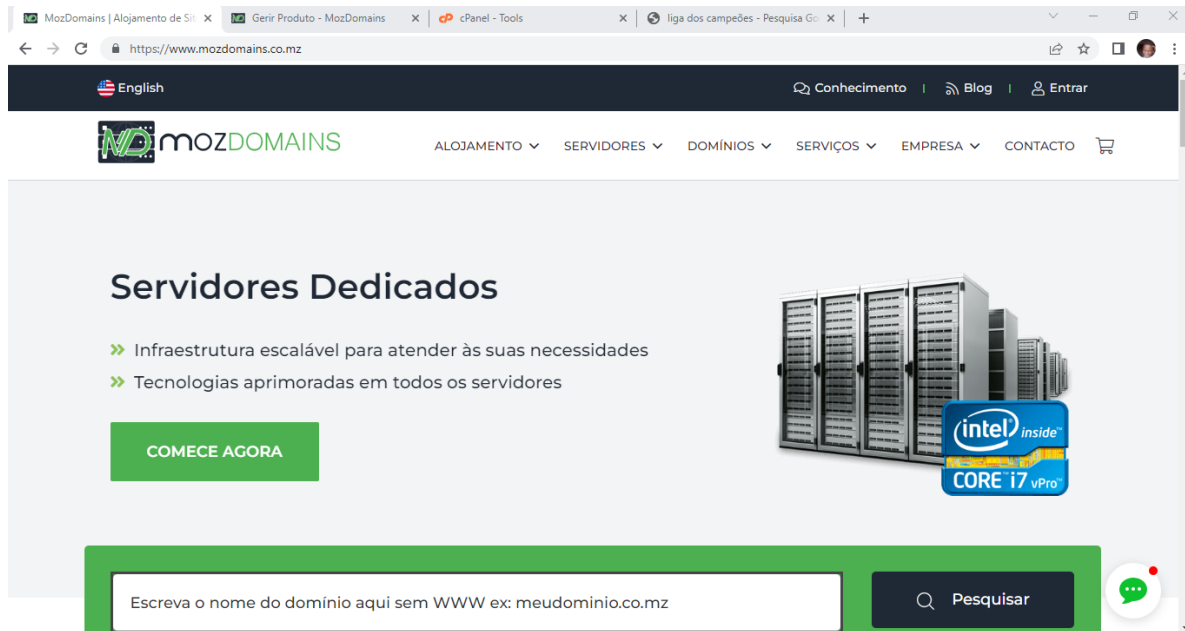


Figura 10: Tela principal da *mozdomains*

Fonte: Autor

2º. Escolhe-se o plano de hospedagem e faz-se a compra do domínio para hospedar o sistema pretendido.

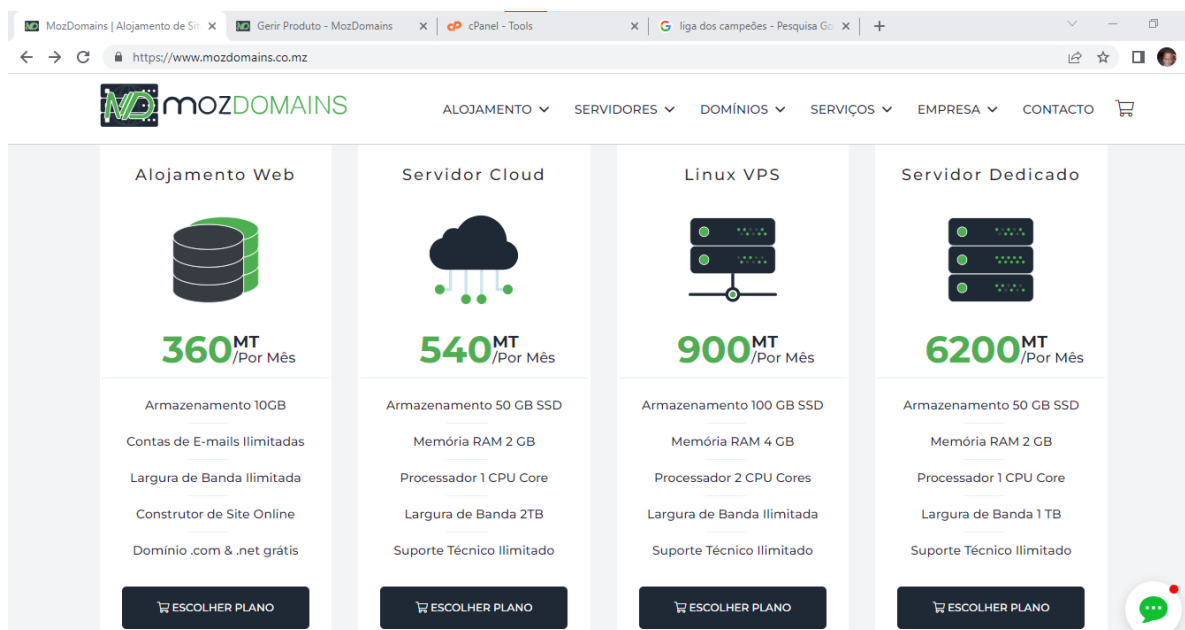


Figura 11: Tela de escolha de plano de hospedagem

Fonte: Autor

3°. Após a compra do plano de hospedagem, dirige-se ao *cpanel* do domínio da hospedagem.

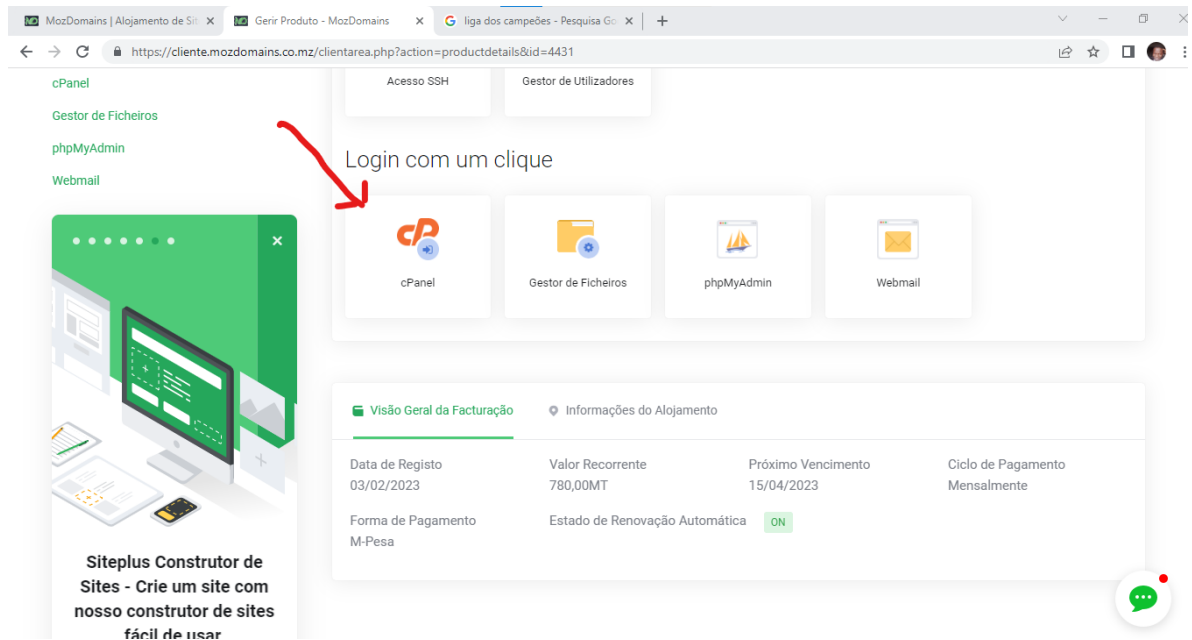


Figura 12: Tela de Cpanel

Fonte: Autor

4° Acede-se ao *cpanel* do domínio e cria-se a base de dados no *MySQL Databases*, como ilustra a figura 13 abaixo.

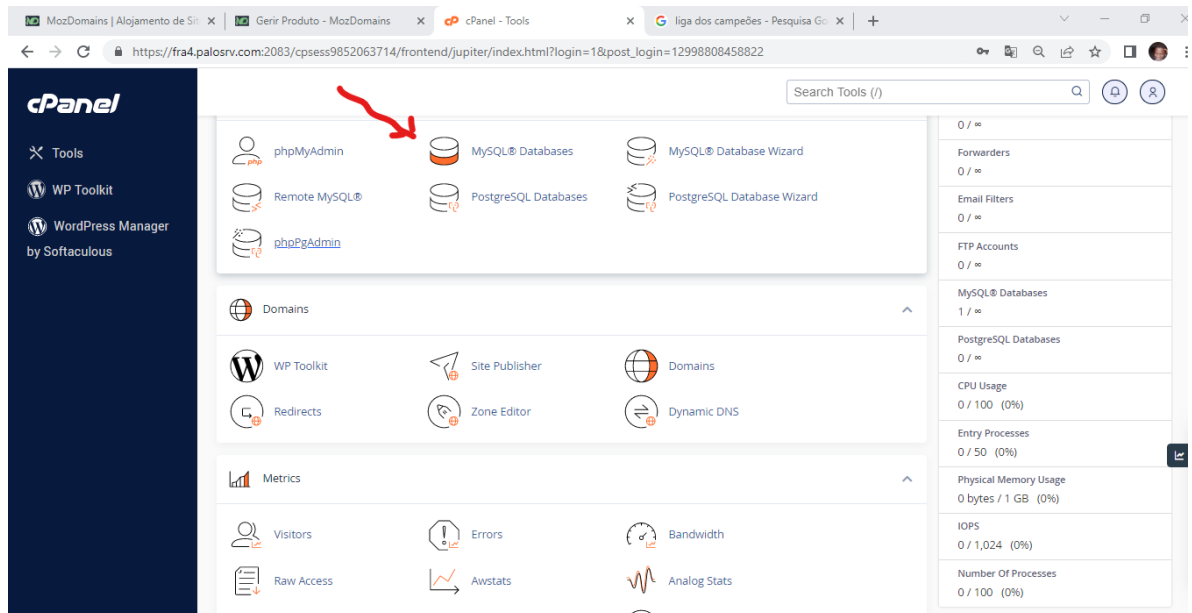


Figura 13: Tela de *MYSQL Databases*

Fonte: Autor

5°. Tendo aberto o *MYSQL Databases*, dá-se um nome a base de dados, preenchendo o campo indicado com a seta vermelha, de seguida clica-se em *create database* como ilustra a figura 14 abaixo.

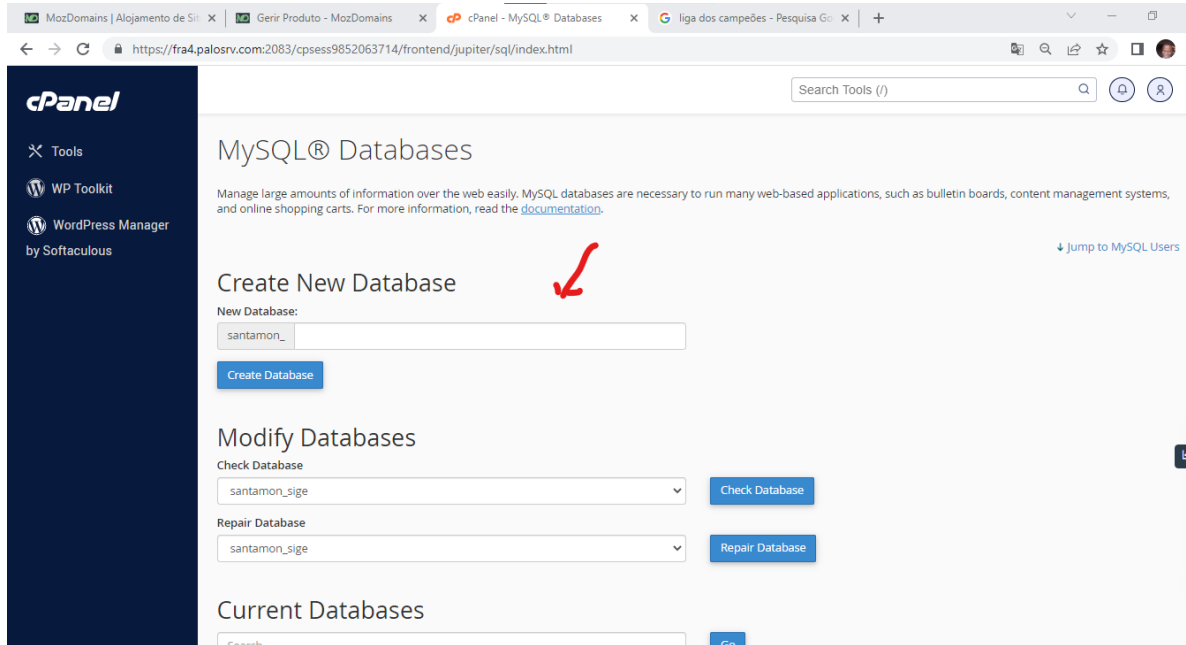


Figura 14: Tela de criação da base de dados

Fonte: Autor

6°. Assim que a base de dados estiver criada na hospedagem, vai-se a *phpMyAdmin*, de modo que se importe a base de dados criada no servidor local *xampp*, como ilustra as figuras 15 e 16.

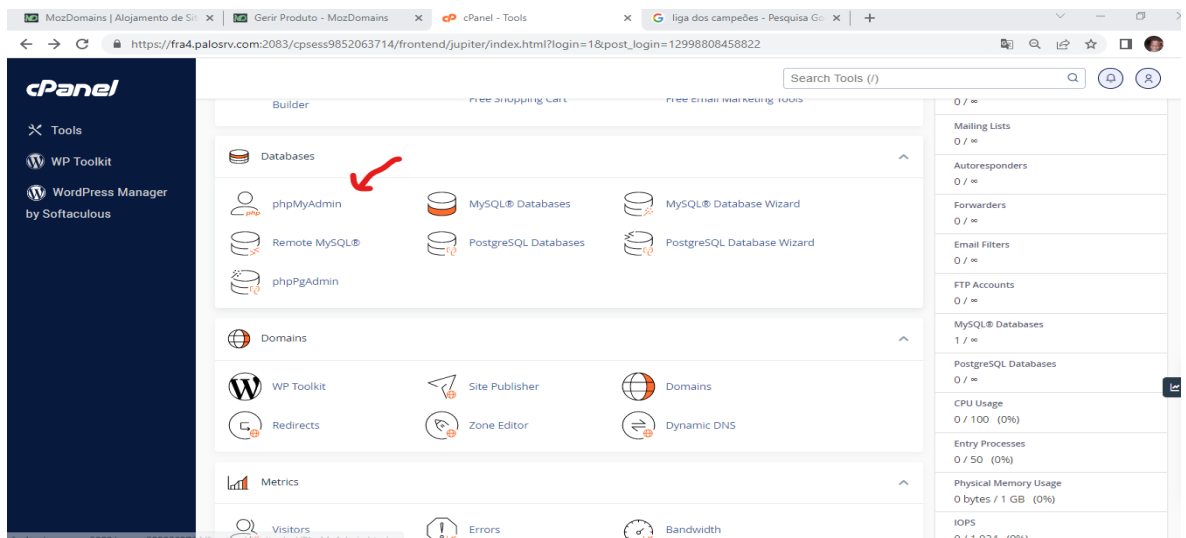


Figura 15: Tela de acesso ao phpMyAdmin

Fonte: Autor

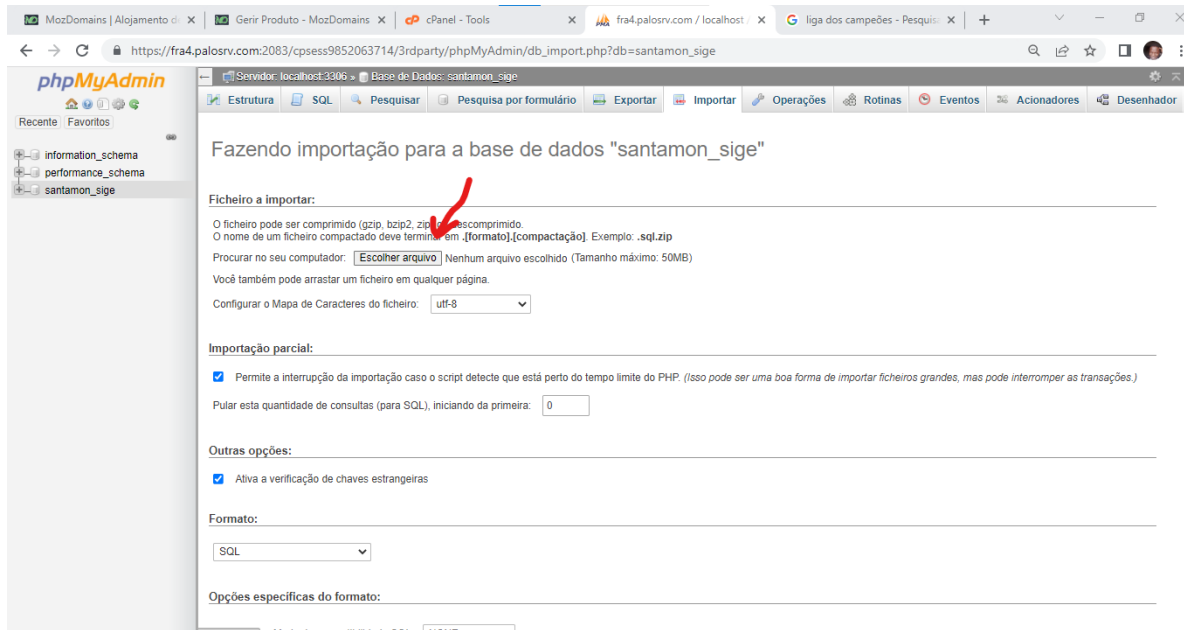


Figura 16: Tela de importação da base de dados na hospedagem

Fonte: Autor

7º. Feita a configuração na base de dados, acede-se ao *cpanel* e direcciona-se ao *File Manager*, onde são importados os ficheiros.

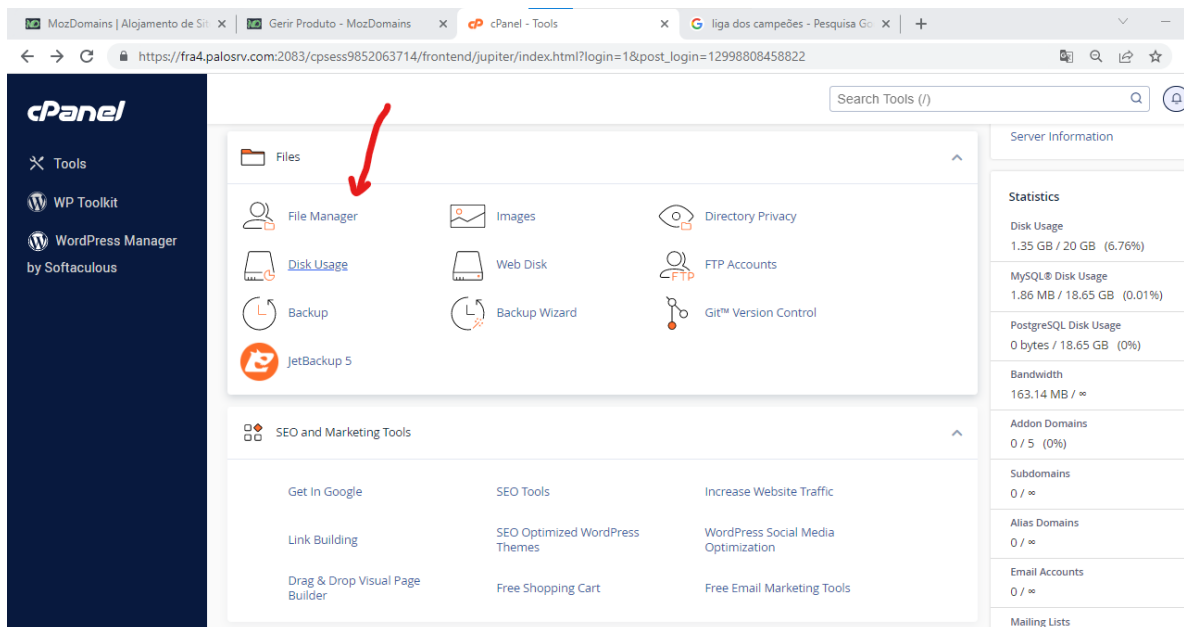


Figura 17: Tela de acesso ao *File Manager*

Fonte: Autor

8°. Estando dentro do File Manager, vai-se a pasta *public_html*, onde serão importados todos os ficheiros do sistema desenvolvido como ilustra a figura 18, e são importados, clicando em *upload* e seleccionando o ficheiro, como ilustra a figura 19.

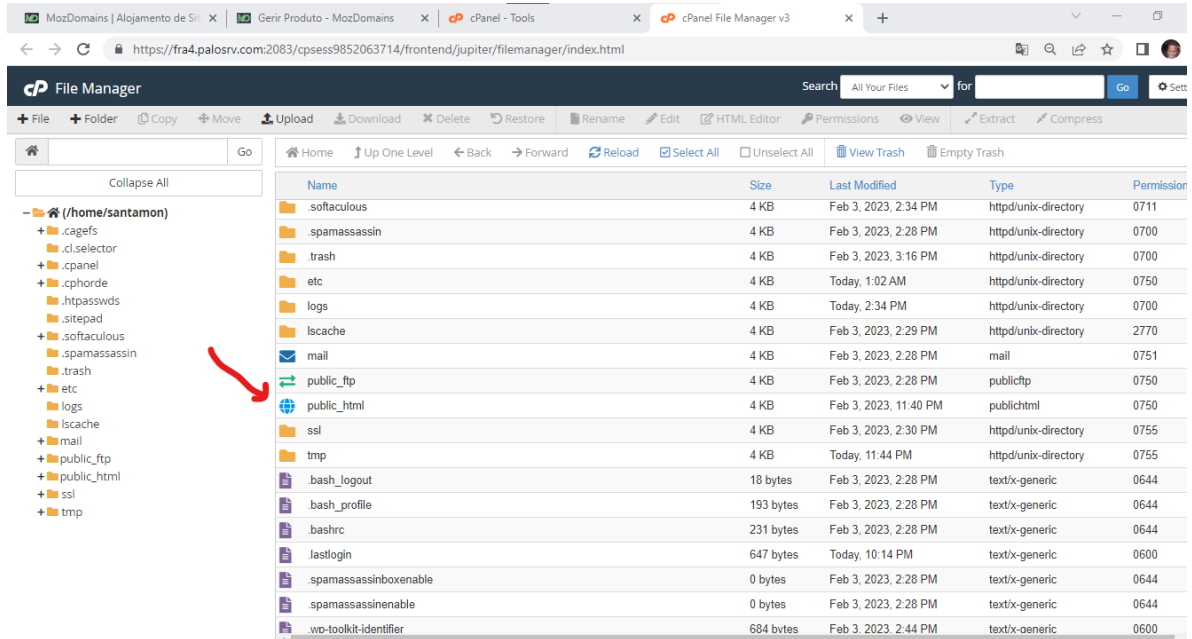


Figura 18: Tela que mostra a pasta onde colocamos o sistema

Fonte: Autor

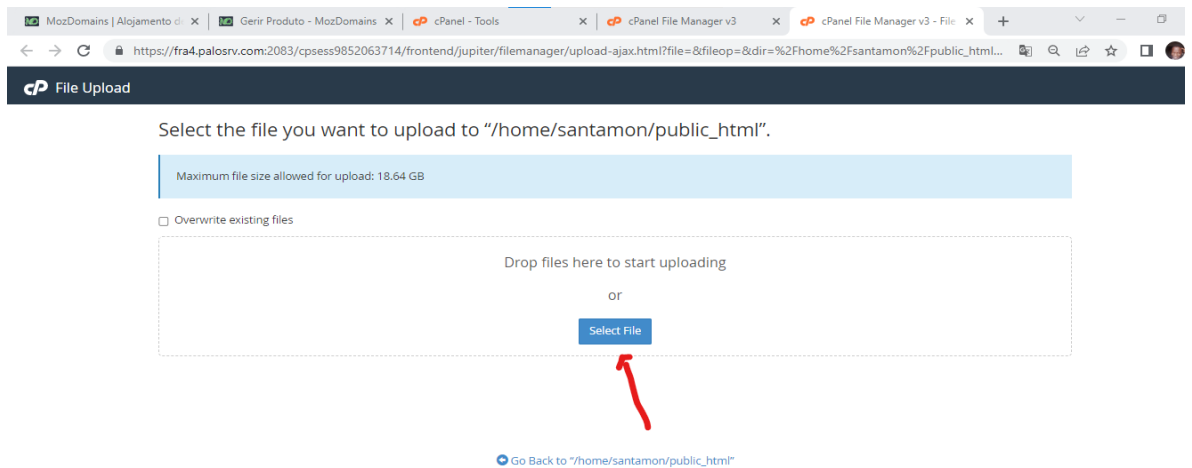


Figura 19: Tela de *upload*

Fonte: Autor

9º. Feito o *upload* do ficheiro do sistema desenvolvido na pasta *public_html*, procura-se pelo ficheiro *.env* para a configuração da base de dados criada na hospedagem. Como ilustra a figura 20 e 21.

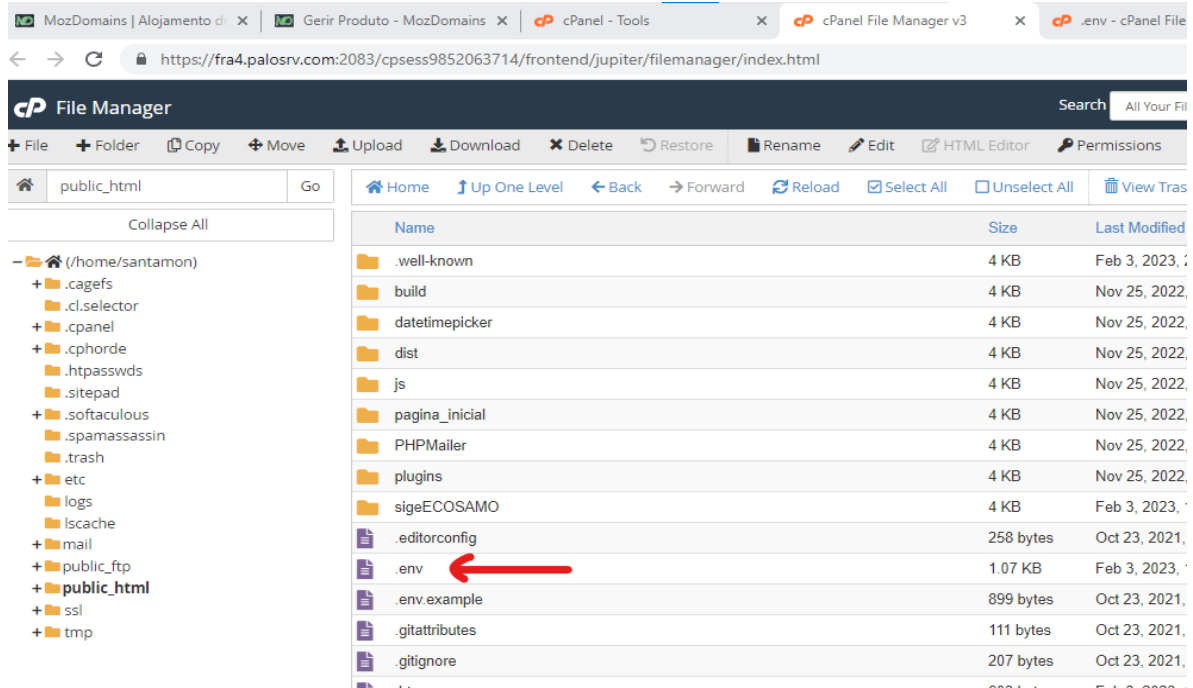


Figura 20: Tela que ilustra o ficheiro *.env*

Fonte: Autor

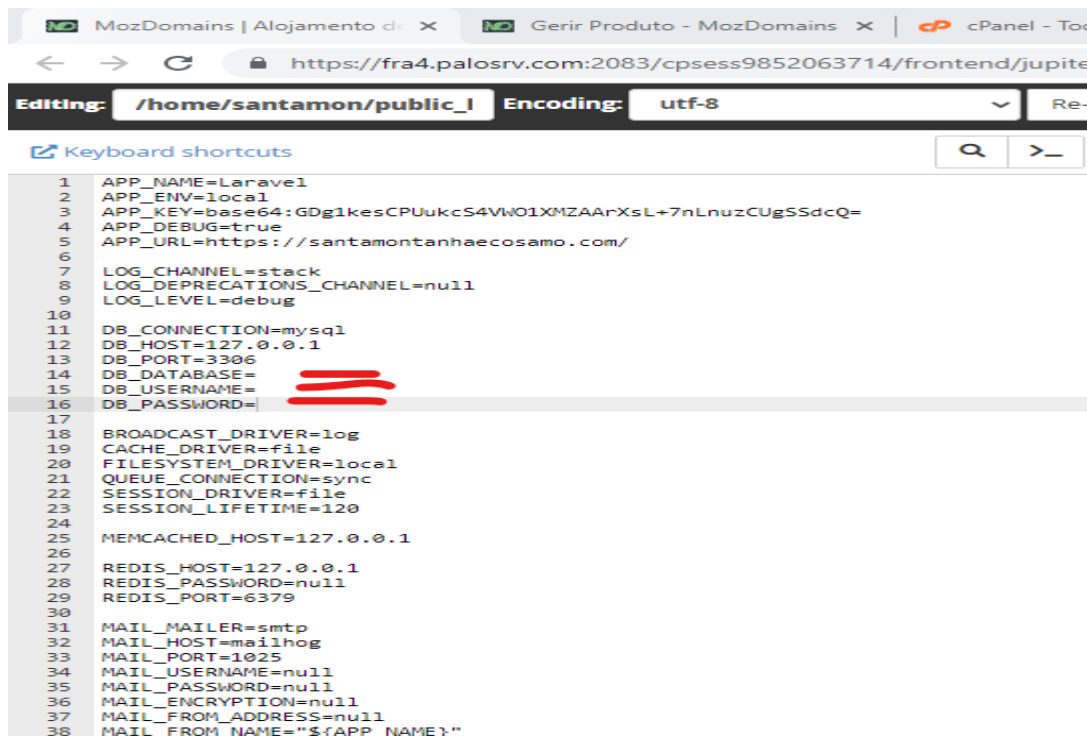


Figura 21: Tela de configuração da base de dados no arquivo *.env*

Fonte: Autor

Terminado o processo de implementação do sistema, o domínio do sistema já está disponível e pronto para ser usado, bastando para tal, aceder o seguinte domínio: <https://santamontanhaecosamo.com>.

3.6. Resultados do sistema

Uma vez que o sistema já foi implementado e está numa fase experimental, não foram abrangidas todas as classes, tendo acordado com a direcção da escola, que só iria se matricular os alunos da 8ª Classe por ser a classe inicial e que o seu ingresso na escola depende da matrícula e, para este ano lectivo 2023, a 8ª Classe possui um total de 7 turmas.

No que diz respeito a execução do sistema, pode notar-se que o mesmo respondeu de forma satisfatória às necessidades da ECOSAMO, visto que o sistema permite matricular o aluno e consequentemente o seu nome passa a fazer parte da lista da turma em que foi atribuída, como se pode observar nas descrições a baixo:

Acedendo o domínio <https://santamontanhaecosamo.com>, obtém-se o seguinte resultado, como ilustra a figura 22.

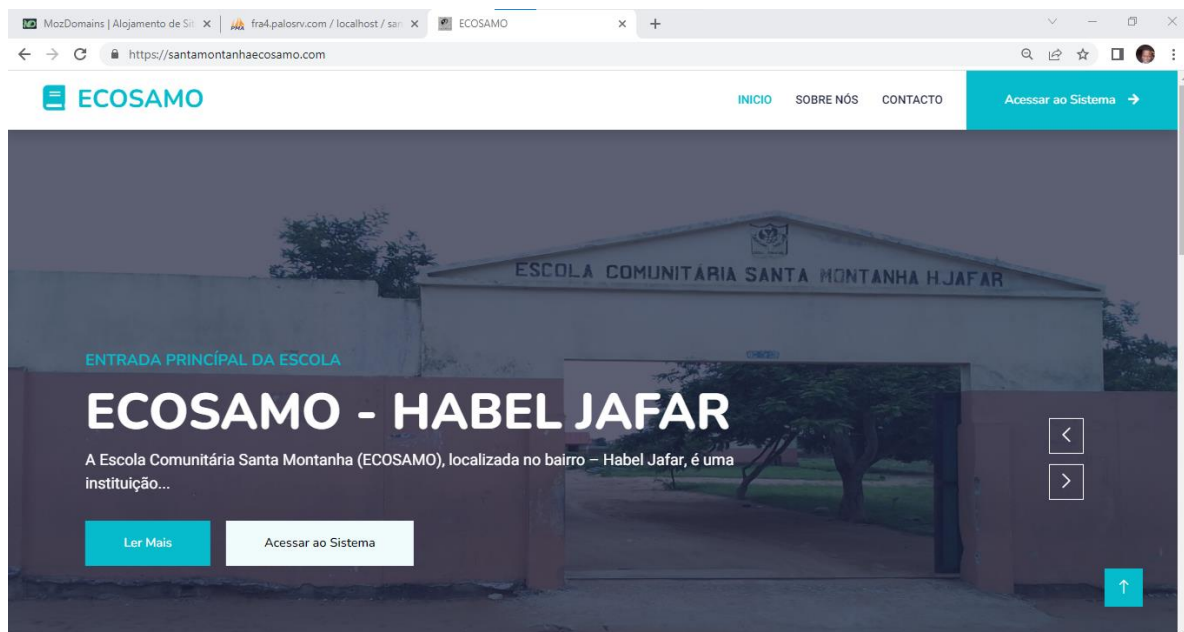


Figura 22: Tela inicial da ECOSAMO

Fonte: Autor

As figuras abaixo, demonstram o resultado final dos passos seguidos anteriormente, que são as listas definitivas das turmas dos alunos matriculados na 8ª Classe, do ano lectivo de 2023.

Neste ano, foram matriculados 357 alunos, como ilustra a figura 23.

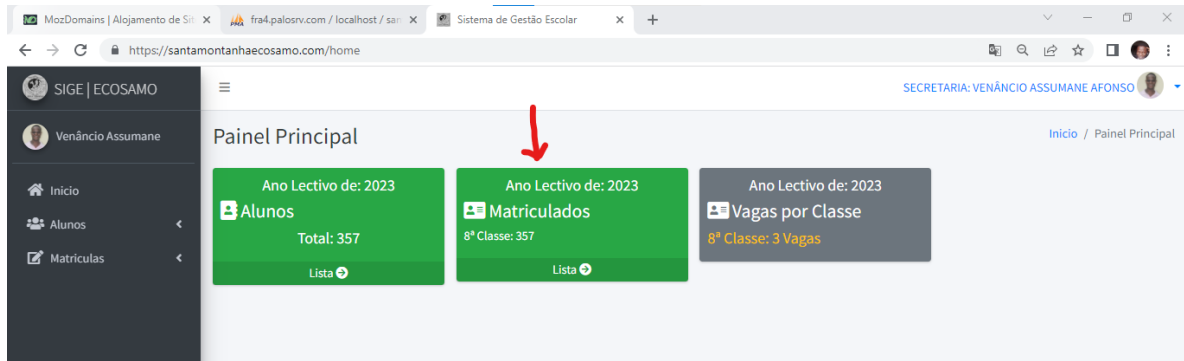


Figura 23: Tela inicial da Secretaria

Fonte: Autor

Para visualizar e imprimir as listas das respectivas turmas, devem ser preenchidos alguns campos considerados obrigatórios, nomeadamente o ano, classe e turma, de seguida confirma-se o preenchimento. Feito isto, o sistema mostra a lista da turma seleccionada, conforme ilustram as figuras 24 e 25 abaixo.

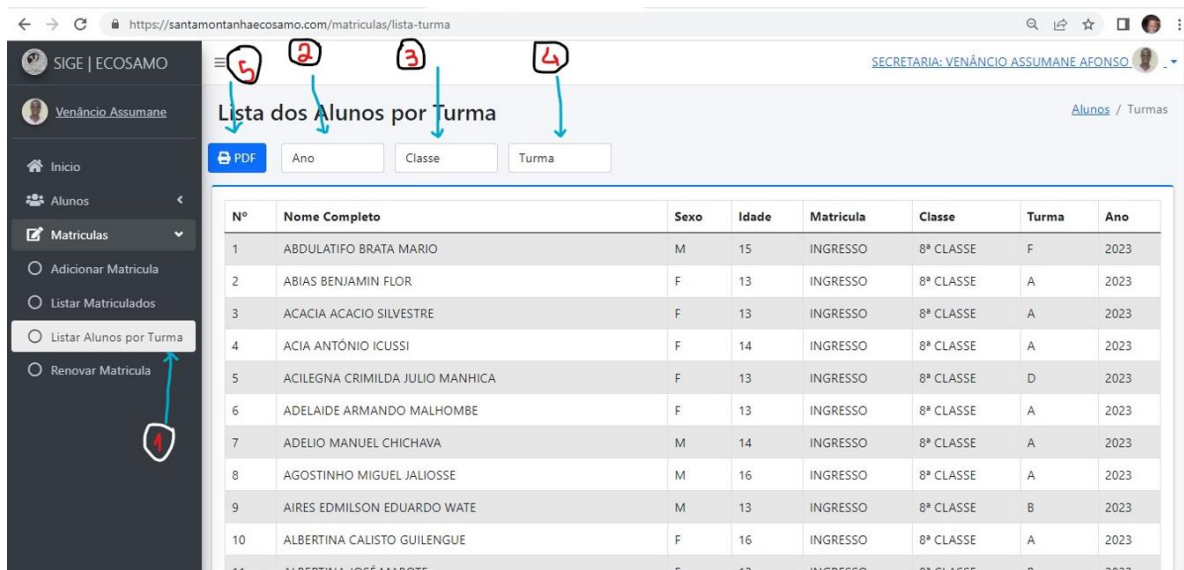



Figura 24: Tela de visualização e impressão da lista

Fonte: Autor

As listas que estão sendo usadas, são apresentadas, como ilustra a figura 25.

 ESCOLA COMUNITÁRIA SANTA MONTANHA - HABEL JAFAR LISTA DEFINITIVA DA 8ª CLASSE TURMA A - 2023						
Nº	NOME COMPLETO	GN	NASCIMENTO	IDADE	PROV	OBSERVAÇÃO
1	Abdias Benjamim Flor	F	08-05-2010	13	E.P.C Mateque	A
2	Acacia Acacio Silvestre	F	08-12-2010	13	E.P.C zintava	A
3	Acia Antônio Ivaasi	F	30-09-2009	14	ECOSAMO	A repetente
4	Adelaide Armando Matibombe	F	04-06-2010	13	E.P.C Mateque	A
5	Adelino Manuel Chichava	M	27-12-2009	14	ECOSAMO	A
6	Agostinho Miguel Jalosse	M	07-08-2007	16	E.P.C Ricafia	A
7	Albertina Calisto Gualingue	F	19-02-2007	16	ECOSAMO	A
8	Arcina Berta Johane	F	22-03-2010	13	E.P.C Guarva	A
9	Ana Eládio Langa	F	13-07-2009	14	E.P.C Jafar	A
10	Arlindo Armando Tivane	M	09-09-2008	15	E.P.C Jafar	A
11	Armando Mario Matola	M	27-02-2009	14	E.P.C Mateque	A
12	Armando Carlos Chavosa	M	19-06-2010	13	E.P.C Mateque	A
13	Augusto Júlio Mafuca	M	11-02-2007	16	E.P.C Mateque	A
14	Aydon Florencio Macaringue	M	01-08-2010	13	E.P.C 16 de junho	A
15	Clelio Jorge Sengo	M	30-08-2010	13	E.P.C Guarva	A
16	Donackton José Cama	M	23-02-2007	16	E.P.C Mateque	A
17	Dylcia Adylson Patrício	F	26-04-2010	13	E.P.C Mateque	A
18	Edílio Eduardo Maduro	M	26-07-2010	13	E.P.C Mateque	A
19	Edito António Chavale	M	09-03-2010	13	E.P.C Guarva	A
20	Eladi Edison Guarube	F	17-05-2010	13	Centro Educacional	A
21	Fernando Arnaldo Matavele	M	13-10-2010	13	Externato Albert Einstein	A
22	Fernando Matoshe Junior	M	21-01-2010	13	E.P.C Jafar	A
23	Fernando Rosa Matombene	M	25-11-2011	12	E.P.C Ricafia	A
24	Filipe Vasco Zita Junior	M	26-02-2010	13	Externato Albert Einstein	A
25	Flávia Julia Langa	F	12-12-2010	13	E.P.C Guarva	A
26	Ibon Jorge	M	21-01-2010	13	E.P.C Muzerno	A
27	Israel Laura	F	29-05-2010	13	E.P.C Mateque	A
28	Isabel António Assado	F	04-09-2010	13	E.P.C Mateque	A
29	Isabel Charles Manango	F	24-07-2007	16	E.P.C Jafar	A
30	Isabel Reginaldo Naicne	F	27-03-2009	14	E.P.C Mali	A
31	Larcia Adelfa Bague	F	16-11-2010	13	colégio beth	A
32	Larissa Cecília Lipsanga	F	22-10-2010	13	E.P.C Jafar	A
33	Larissa David Ngosa	F	04-07-2010	13	E.P.C Mateque	A
34	Lasca Clara Pedro Ferrio	F	10-11-2010	13	E.P.C Mateque	A
35	Laura Adelaide Matibombe	F	17-11-2009	14	E.P.C Ricafia	A
36	Manuel Asser Cambe	M	24-04-2010	13	E.P.C Polana Canico	A
37	Marcela Paulo Miambo	F	12-09-2010	13	E.P.C zintava	A
38	Marcia Helio Nhambo	F	12-09-2007	16	E.P.C Mateque	A
39	Orianda Oriando Mucacho	F	07-04-2010	13	E.P.C Mateque	A
40	Oriando Miguel Macandrea	M	28-09-2009	14	Externato Albert Einstein	A
41	Osoria Da Isabel Mandilovo	F	30-05-2011	12	E.P.C Jafar	A
42	Ovaldo Carlos Zimila	M	14-02-2008	15	E.P.C Mateque	A
43	Oyena Bernardo MacBastene	F	10-04-2010	13	E.P.C Jafar	A
44	Regiana Da Silva Casane	F	12-08-2010	13	E.P.C zintava	A
45	Sakoni Roberto Machaizic	M	16-10-2009	14	E.P.C Guarva	A
46	Shaniel De Pedro Machiana	F	18-05-2010	13	E.P.C Jafar	A
47	Shelton Carlos Dzimba	M	06-09-2009	14	E.P.C Jafar	A
48	Shelton Marcelino Zandiamela	M	31-03-2010	13	Externato Albert Einstein	A
49	Sora Ruben Matosse	M	15-07-2008	15	E.P.C Mateque	A
50	Suzana Estevo Muthamba	F	18-04-2008	15	E.P.C Mateque	A

O Director da Escola
Pedro Alvílio Salatol Gomes

25 Masculino
27 Feminino
50 Total

Figura 25: Lista definitiva

Fonte: Autor

4. CONSIDERAÇÕES FINAIS, RECOMENDAÇÕES, TRABALHOS FUTUROS E LIMITAÇÕES

4.1.Considerações Finais

A presente monografia procurou abordar sobre a concepção e implementação de um sistema de gestão de matrículas: estudo de caso - Escola Comunitária Santa Montanha, uma instituição de ensino Secundário do primeiro e segundo ciclo, isto é, lecciona desde a 8ª classe a 12ª classe. Cita no bairro Habel Jafar, Distrito de Marracuene, que em cada início do ano lectivo, os alunos passam pelo processo de matrícula para garantirem a sua vaga na escola. Este processo é feito mediante a apresentação de documentos na secretaria da escola e o talão de depósito bancário com o valor estipulado pela escola referente ao pagamento da matrícula, seguido de vários procedimentos que são realizados pelos funcionários da secretaria da escola, como verificou-se na descrição da problematização, no capítulo I do presente projecto (monografia).

A concretização desta monografia constituiu não só um grande desafio, como também uma satisfação para o autor, dado que ao longo da sua materialização, o autor procurou aplicar conhecimento e técnicas aprendidas ao longo do seu percurso académico. Assim, a realização da monografia deduz-se como uma oportunidade do autor pôr em prática parte do que foi aprendendo, como também, tornou-se numa fonte de aquisição de experiência no que diz respeito a composição de um trabalho científico, factor bastante gratificante para o autor, para além de fazer parte da vontade de pôr em prática o seu conhecimento em prol do desenvolvimento da educação social no país.

O processo de gestão de matrículas constitui um desafio sobretudo nas escolas primárias e secundárias do nosso país Moçambique, optando-se assim no uso de sistemas remotos, facto que acaba reflectindo no tempo de realização da actividade que muitas vezes acabam levando mais tempo do que o desejado. A Escola Comunitária Santa Montanha não foge desta realidade. Assim, a implementação de um sistema de gestão de matrícula torna-se oportuna, fazendo com que as actividades sejam feitas de forma célere e organizada, ou seja, de forma flexível e em curto tempo.

Para o desenvolvimento do sistema de gestão de matrículas para a Escola Comunitária Santa Montanha usou-se não apenas a metodologia *Rational Unified Process* - RUP com o objectivo de ampliar a produtividade no processo de desenvolvimento de *software*, mas também usou a linguagem PHP.

Após a implementação do sistema, espera-se que o mesmo proporcione um ambiente de trabalho flexível e que seja capaz de integrar os dados dos alunos no processo e que possa atribuir para cada classe, turmas de acordo com a idade, sem que haja a necessidade de preencher em papéis, facilitando e reduzindo desse modo, o tempo de trabalho.

Com a gestão mais eficiente, a escola poderá identificar de forma mais rápida todos os processos (inscrições, classes e listas das turmas) em tempo real. Igualmente, com a implementação deste sistema, os gestores do processo de matrícula poderão ter possibilidade de gerir todas a informações relacionadas outras funcionalidades dentro do sistema deixando de lado as alternativas anteriormente utilizadas, neste caso, papéis.

4.2.Recomendações

Ainda que a direcção da Escola Comunitária Santa Montanha tenham visto com bom grado a implementação de um sistema de gestão de matrículas para dinamização desta actividade na sua instituição, torna-se importante a busca de mais informações para aferir o grau de satisfação a fim de que se faça a sua implementação em outras escolas com problemas similares aos da escola em questão.

Para além disso, recomenda-se uma constante capacitação aos futuros usuários do sistema no que diz respeito ao uso do mesmo, a fim que se garanta uma boa gestão, bem como, para que se evite resistência a mudança. E para que o processo todo corra sem sobressaltos, aconselha-se que os usuários tenham conhecimentos básicos de informática.

Aconselha-se a observância no que diz respeito a realização dos processos de registo, ou qualquer outra operação, de modo que estas sejam concluídas com sucesso, isto é, até que se receba a mensagem de realização da operação com sucesso.

Por fim, recomenda-se a exploração ao máximo do sistema, a fim de se apurar os resultados, ou seja, se o sistema corresponde a inovação tanto desejada para a dinamização das tarefas na ECOSAMO, no âmbito do processo de matrículas.

4.3.Trabalhos Futuros

Como forma de melhorar o sistema, pretende-se posteriormente permitir que o aluno aceda o sistema para a sua inscrição a partir de um local diferente da escola, assim como verificar o seu nome e a lista da turma em que fará parte; permitir que o sistema reconheça o pagamento por entidade e referência feito através de um telemóvel ou de um computador.

4.4.Limitações

Ao longo da realização do trabalho, deparou-se com a falta de estudos que estejam relacionados com o tema. Ao longo da concepção do sistema, a principal limitação teve a ver com o estabelecimento de parâmetros em relação ao período de realização das matrículas, ou seja, definir claramente os intervalos de início e encerramento de matrículas, visto que, o período de matrículas varia de acordo com o ano lectivo.

REFERÊNCIAS BIBLIOGRÁFICAS

1. ABRAHAMSSON, Pekka, SALO, Outi, RONKAINEN, Jussi, e WARSTA, Juhani. *Agile Software Development Methods*. Finland, 2002.
2. BEDER, Delano Medeiros; *Engenharia Web : uma abordagem sistemática para o desenvolvimento de aplicações web*, São Carlos : EdUFSCar, 2012
3. BOECHAT, Gláucia Carreiro; *Apostila de Linguagem de Programação I e II PHP ,IFECT*, Guarulhos, 2014.
4. BOOCH, Grady; RUMBAUGH, James; JACOBSON, Ivar. *UML: Guia do Usuário*. Trad. Fábio Freitas. Rio de Janeiro, campus, 2000
5. BORDIGNON, Genuíno. *Conselhos Escolares: uma estratégia de gestão democrática da educação pública*. Brasília: Ministério da Educação/Secretaria da Educação Básica, 2004.
6. CARLETO, Nivaldo; *Tecnologias De Informação e Comunicação na Gestão Educacional: Possibilidades e Contribuições em Uma Escola Técnica Estadual*; UNESP, Araquara, São Paulo, 2009
7. DOURADO, Luiz Fernandes: *Gestão em Educação Escolar*, 4ª ed. Cuiabá: Universidade Federal de Mato Grosso, 2012
8. FULGHAM, Brent &, GOUY, Isaac «[The Computer Language Benchmarks Game](https://en.wikipedia.org/wiki/The_Computer_Language_Benchmarks_Game)» (2004). https://en.wikipedia.org/wiki/The_Computer_Language_Benchmarks_Game acesso a 24 de Abril de 2022
9. FIREBASE, *Armazenamento e sincronização de dados em tempo real*. Disponível em: <<https://firebase.google.com/products/database/?hl=pt-br>>. Acesso em: 23 de Março de 2018.
10. GERHARDT, T. E. & SILVEIRA, D.T. (org); *Métodos de pesquisa*, Universidade Aberta do Brasil – UAB/UFRGS e SEAD/UFRGS. Editora da UFRGS, Porto Alegre, 2009.
11. GIL, Antônio Carlos. *Métodos e Técnicas de Pesquisa Social*. 6.ed.: Atlas, São Paulo, 2008
12. LANCERDA, José Antônio Vieira; *Análise de Frameworks de Desenvolvimento Web para a Conversão de uma Aplicação Desktop*, IPVC, Brasil, 2013
13. LÜCK, Heloísa. *Dimensões de gestão escolar e suas competências*. Editora Positivo, Curitiba, 2009.
14. KAUARK, Fabiana da Silva; MANHÃES, Fernanda Castro e MEDEIROS, Carlos Henrique. *Metodologia de Pesquisa: Um Guião Prático*. Itabuna – Bahia. Via Litterarum, 2010.
15. LAKATOS, E. M. & MARCONI, M. A. *Fundamentos de Metodologia Científica*. 5ª ed. São Paulo, Atlas, 2003.
16. LAUDON, Kenneth C; LAUDON, Jane P., *sistema de informação com internet*. Rio de Janeiro: LTC, 1999
17. MARCONI, Marina de Andrade e LAKATOS, Eva Maria. *Metodologia Científica*. 6ª ed. São Paulo: Atlas, 2011.
18. OLIVEIRA, F. G. DE, & SEABRA, J. M. P. *Metodologias de Desenvolvimento de Software: Uma Análise no Desenvolvimento de Sistemas Web*, 2015
19. PACHECO, Felipe Guimarães; *Análise das técnicas de segurança do framework laravel contra ataques as aplicações web-* Trabalho de Conclusão de Curso (Graduação em Ciência da Computação) -Universidade Federal Rural de Pernambuco, BR-PE, 2019.
20. PARO, Vitor Henrique. *A gestão da Educação ante as exigências de qualidade e produtividade da escola pública*, 1999.
21. PRESSMAN, R. S. *Engenharia de Software*. 5ª Edição. Rio de Janeiro. Makron Books. 2003.

22. PRESSMAN, R. S. *Engenharia de Software*. 4ª Edição. Rio de Janeiro. Makron Books.2002.
23. PRODANOV, C. C. & FREITAS, E. C. *Metodologia do trabalho científico [recurso electrónico]: métodos e técnicas da pesquisa e do trabalho académico*, 2. ed. Novo Hamburgo: Feevale, 2013
24. SCHULTZ, Glauco: *Introdução à gestão de organizações*; SEAD/UFRGS. – Porto Alegre: Editora da UFRGS, 2016.

25. SERRA, Andreia, *Manual de metodologias De pesquisa*, Instituto Superior Politécnico e Universitário, Escola Superior de Ciências Jurídicas e Sociais, Maputo, 2004

26. SOMMERVILLE, Ian. *Engenharia De Software*. 9ª ed. 2011
27. TAUCHEN, Gionara (org.) *Gestão e Organização Escolar*, Editora da FURG, Rio Grande, 2013.
28. WILLEMANN, D. P. & IBARRA, G.B.; *Framework Java de Apoio ao Desenvolvimento de Aplicações Web com Banco de Dados, utilizando Struts, Tiles e Hibernate*, Universidade Federal de Santa Catarina, Florianópolis – 2007
29. <https://www.infoescola.com/engenharia-de-software/rup/> Visto no dia 09 de Fevereiro de 2021
30. <https://www.devmedia.com.br/metodologia-de-desenvolvimento-de-software/1903/> Visto no dia 09 de Fevereiro de 2021
31. Ferramenta *Rational Unified Process*.
32. <https://developer.mozilla.org/pt-BR/docs/Learn/Server-side/Django/Introduction> /Visto no dia 23 de Abril de 2022.
33. <https://dicas.codeigniter.com.br/o-que-e-codeigniter/> Visto no dia 23 de Abril de 2022.
34. <http://www.lyfreitas.com.br/ant/pdf/php.pdf> /Visto no dia 23 de Abril de 2022
35. <https://blog.betrybe.com/javascript> Visto no dia 23 de Abril de 2022

Apêndice

Apêndice 1: Guião de entrevista a direcção da escola

Questionário

(Dirigido aos directores-pedagógico e aos funcionários da secretaria)

Saudações caro (a), o presente questionário tem o objectivo a recolha de dados sobre “o processo de matrícula” na presente instituição de ensino, para fins da elaboração da monografia científica, como requisito de obtenção do grau de licenciatura em informática com habilitação em engenharia em desenvolvimento de sistema. Pede-se que responda com maior sinceridade e fidelidade para que as informações reflectam a realidade vivenciada nesta instituição a quando da realização do processo de matrícula.

1. A Escola Comunitária Santa Montanha (ECOSAMO) recorre ao processo de matrícula para a admissão dos alunos das classes iniciais do ensino secundário geral? (Marca com X a opção correcta). NÃO SIM

2. Quais são as dificuldades enfrentadas no processo de matrículas na ECOSAMO?

3. Descreve o passo a passo do processo de matricula nesta escola.

4. Acha que existe algum mecanismo que pode ajudar na melhor gestão de matrículas na ECOSAMO? (Marca com X a opção correcta). NÃO SIM

6. Que mecanismos podem ajudar na gestão de matrículas na ECOSAMO?

5. Em termos de tempo, qual é a tua opinião sobre o sistema que actualmente está em uso nesta escola?

a) O tempo é satisfatório b) O tempo é razoável c) O tempo não é satisfatório

d) Outra alternativa (descreve) _____

7. Que meio tecnológico pode flexibilizar a gestão de matrículas na EOCOSAMO?

Obrigado!