

**UNIVERSIDADE JOSÉ EDUARDO DOS SANTOS**

**ESCOLA SUPERIOR POLITÉCNICA DO BIÉ**

**­­­­­­­­­­­­­­­­­­­­­­­DEPARTAMENTO DE INFORMÁTICA**

**REGISTO Nº\_\_\_\_\_\_\_**

**SISTEMA WEB PARA GESTÃO ESCOLAR DO LICEU FELIZBERTO D. NGUNGA DE CATABOLA**

**Trabalho de Conclusão do Curso de Engenharia Informática**

**A Ser Apresentado por Eusébio Salomão Bango em 2023**

**Orientado pelo professor Engenheiro Paulo Beijio Calei**

**BIÉ – 2023UNIVERSIDADE JOSÉ EDUARDO DOS SANTOS**

**ESCOLA SUPERIOR POLITÉCNICA DO BIÉ**

**­­­­­­­­­­­­­­­­­­­­­­­DEPARTAMENTO DE INFORMÁTICA**

**SISTEMA WEB PARA GESTÃO ESCOLAR DO LICEU FELIZBERTO D. NGUNGA DE CATABOLA**

Trabalho subordinado ao tema “Sistema web para Gestão Escolar do Liceu Felizberto D. Ngunga de Catabola” a ser apresentado à Universidade José Eduardo dos Santos, como parte dos requisitos para aquisição do grau de Engenheiro em Informática.

**BIÉ - 2023**

**Reflexão**

*“Todos os ribeiros vão para o mar, contudo, o mar não se enche; para o lugar para onde os ribeiros vão, para ai tornam eles a ir. Todas essas coisas cansam tanto, que ninguém o pode declarar; 'os olhos nào se fartam de ver, nem os ouvidos de ouvir. O que foi, isso é o que há de ser; e o que se fez, isso se tornará a fazer; de modo que nada há novo debaixo do sol.”*

ECLESIASTES 1:7-9

**Dedicatória**

*À duas mulheres que deram um significado e direção já mais esperado de um outro ser humano, minha mãe e minha querida e inesquecível esposa, aos meus irmãos, meus colegas de curso e à todos que corajosamente decidirem veredar para a área de tecnologias de informação.*

**Agradecimentos**

A Deus todo-poderoso pela vida e saúde que nos proporciona a cada dia, a meu pai(em memória) que sempre apostou na minha formação, minha querida mãe que nunca desistiu de mim, me insentivando desde a iniciação escolar, à minha querida e inesquecível esposa pelo apoio paciência e compreensão durante a formação superior, aos meus familiares, ao colectivo de professores pelos conhecimentos transmitidos com destaque ao meu tutor Engenheiro Paulo Beijio Calei que detalhadamente acompanhou o desenvolvimento deste projecto, aos colegas pela harmonia durante os cincos anos de formação e por se tornarmos uma família, e a todos que direta e indirectamente contribuíram para que, o que era apenas um sonho tornar-se realidade.

**Índice**

[Lista de tabelas I](#_Toc137384730)

[Lista de figuras II](#_Toc137384731)

[Lista de símbolos, abreviaturas e siglas IV](#_Toc137384732)

[Resumo V](#_Toc137384733)

[Abstract VI](#_Toc137384734)

[INTRODUÇÃO 1](#_Toc137384735)

[CAPÍTULO I - FUNDAMENTAÇÃO TEÓRICA 6](#_Toc137384736)

[1.1. Antecedentes históricos dos sistemas informáticos. 6](#_Toc137384737)

[1.1.1. Sistemas Informáticos (SI) 7](#_Toc137384738)

[1.1.2. Automatização dos processos nas organizações 7](#_Toc137384739)

[1.2. Gestão escolar e TICs 8](#_Toc137384740)

[1.3. Sistema informático de Gestão escolar 9](#_Toc137384741)

[1.4. Situação actual do processo de Gestão escolar do Liceu Felizberto D. Ngunga de Catabola 9](#_Toc137384742)

[1.5. Ferramentas para o desenvolvimento de um sistema informático 10](#_Toc137384743)

[1.5.1. Metodologias de desenvolvimento de um software 10](#_Toc137384744)

[1.5.2. Linguagens de programacão web 16](#_Toc137384745)

[1.6. Sistemas web de Gestão escolar 25](#_Toc137384746)

[1.6.1. – Sistemas existentes 25](#_Toc137384747)

[1.7. Conclusões do Capitulo I 27](#_Toc137384748)

[CAPÍTULO II: EXPLORAÇÃO, PLANEJAMENTO E IMPLEMENTAÇÃO 28](#_Toc137384749)

[2.1. Requisitos funcionais e não funcionais 28](#_Toc137384750)

[2.1.2. Duração das iterações das HU 36](#_Toc137384751)

[2.2. Arquitectura de Software 37](#_Toc137384752)

[2.3. Tarefas de engenharia 37](#_Toc137384753)

[2.3.1. Descrevendo as tarefas de engenharia 40](#_Toc137384754)

[2.4. Conclusões parciais do capitulo II 43](#_Toc137384755)

[CAPÍTULO III: PROVAS DE ACEITAÇÃO DO SISTEMA WEB PARA GESTÃO ESCOLAR DO LICEU FELIZBERTO D. NGUNGA DE CATABOLA 44](#_Toc137384756)

[3.1. Princípios de desenho de interface 44](#_Toc137384757)

[3.2. Tratamento de erros 45](#_Toc137384758)

[3.3. Segurança do sistema 45](#_Toc137384759)

[3.4. Testes de software 46](#_Toc137384760)

[3.5. Especificações dos testes correspondentes a metodologia XP 47](#_Toc137384761)

[3.6. Conclusões parciais do capitulo III 55](#_Toc137384762)

[CONCLUSÕES GERAIS 56](#_Toc137384763)

[Recomendações 57](#_Toc137384764)

[Referências Bibliográficas 58](#_Toc137384765)

[Anexos](#_Toc137384766)

# Lista de tabelas

[Tabela 1 - Descrição dos Requisitos Funcionais 29](#_Toc137400994)

[Tabela 2 -Histórias de Usuários 32](#_Toc137400995)

[Tabela 3 -Descrição de HU Gerir usuário 33](#_Toc137400996)

[Tabela 4 - Descrição de HU Gerir configurações 34](#_Toc137400997)

[Tabela 5 - Descrição de HU Gerir informações de escola 34](#_Toc137400998)

[Tabela 6 - Descrição de HU Fazer inscrição de novos 35](#_Toc137400999)

[Tabela 7 - Descrição de HU Fazer lançamento de notas 35](#_Toc137401000)

[Tabela 8 - Lista e duração das iterações 36](#_Toc137401001)

[Tabela 9 - Tarefas de engenharia 38](#_Toc137401002)

[Tabela 10 - Tarefa de engenharia Criar Usuário 40](#_Toc137401003)

[Tabela 11 - Tarefa de engenharia inscrever candidato 40](#_Toc137401004)

[Tabela 12 - Tarefa de engenharia consultar candidatura 41](#_Toc137401005)

[Tabela 13 - Tarefa de engenharia Lançar Notas 41](#_Toc137401006)

[Tabela 14 - Tarefa de engenharia visualizar notas para alunos 42](#_Toc137401007)

[Tabela 15 - Caso de teste cadastrar usuário funcionário 47](#_Toc137401008)

[Tabela 16 - Caso de teste consultar usuário candidato 48](#_Toc137401009)

[Tabela 17 - Caso de teste lista de anos lectivos 49](#_Toc137401010)

[Tabela 18 - Caso de teste editar candidato 50](#_Toc137401011)

[Tabela 19 - Caso de teste Manter Administrador 52](#_Toc137401012)

[Tabela 20 - Caso de teste matricular aluno 53](#_Toc137401013)

# Lista de figuras

[Figura 1 - Fase de desenvolvimento de um projecto 11](#_Toc137407003)

[Figura 2 - PHP 17](#_Toc137407004)

[Figura 3 - Líguagem de programação JavaScript 18](#_Toc137407005)

[Figura 4 - ASP.NET 19](#_Toc137407006)

[Figura 5 - Linguagem de programação Java 20](#_Toc137407007)

[Figura 6 - Linguagem de programação Pyton 20](#_Toc137407008)

[Figura 7 - Linguagem de programação 21](#_Toc137407009)

[Figura 8 - Linguagem de programação C# 21](#_Toc137407010)

[Figura 9 - Linguagem de marcação de texto HTML 21](#_Toc137407011)

[Figura 10 - Editor de código VS Code 22](#_Toc137407012)

[Figura 11 - Node JS 22](#_Toc137407013)

[Figura 12 - Xamp 23](#_Toc137407014)

[Figura 13 - React 23](#_Toc137407015)

[Figura 14 - Biblioteca Bootstrap 24](#_Toc137407016)

[Figura 15 - MySQL 24](#_Toc137407017)

[Figura 16 - MongoDB – Banco de dado 24](#_Toc137407018)

[Figura 17 - Arquitectura de Software 37](#_Toc137407019)

[Figura 18 - Página principal do sistema proposto 45](#_Toc137407020)

[Figura 19 - Página login 46](#_Toc137407021)

[Figura 20 - Página cadastrar funcionário 48](#_Toc137407022)

[Figura 21 - Página de consulta de candidato 49](#_Toc137407023)

[Figura 22 - Página consulta anos lectivos 50](#_Toc137407024)

[Figura 23 - Página detalhes de candidatos 51](#_Toc137407025)

[Figura 24 - Página editar candidato 52](#_Toc137407026)

[Figura 25 - Página Administrador 53](#_Toc137407027)

[Figura 26 - Página fazer matrícula 54](#_Toc137407028)

# Lista de símbolos, abreviaturas e siglas

|  |  |
| --- | --- |
| TIC | Tecnologias de informação e comunicação |
| FORTRAN | FORmula TRANslation |
| SI | Sistemas informáticos |
| RUP | Rational Unified Process |
| PU | Processo Unificado |
| XP | *Extreme Programming* |
| PHP | Hypertext Pre-Processor |
| URL | Uniform Resource Locator |
| ODBC | Open Database connectivity |
| HTML | HyperText Markup Language |
| JS | JavaScript |
| ASP | Active Server Pages |
| VS Code | Visual Studio Code |
| XAMMP | MySQL e Apache com suporte as linguagens PHP e Perl. |
| FTP | File Transfer Protocol |
| WWW | *World* *Wide* *Web* |
| SQL | Structured Query Language |
| JSX | JavaScript Sintax Extension |
| JSON | JavaScript Object Notation |
| DB | Database |
| HU | Histórias de Usuário |
| MAC | Média das avalições contínuas |
| MT | Média do Trimestre |
| MF | Média Final |
| JWT | Json Web Token |
| BI | Bilhete de Identidade |
| IU | Interface de Usuário |
| SWGELFC | Sistema web de gestão escolar do Liceu Felizberto D. Ngunga de Catabola |

# Resumo

<escreva o resumo>

A dependência cada vez maior da tecnologia de informação (TI) torna software seguro um elemento chave para a continuidade dos serviços de nossa sociedade atual. Nos últimos anos, instituições públicas e privadas aumentaram seus investimentos em segurança da informação, mas a quantidade de ataques vem crescendo mais rapidamente do que a nossa capacidade de poder enfrentálos, colocando em risco a propriedade intelectual, a relação de confiança de clientes e a operação de serviços e negócios apoiados pelos serviços de TI. Especialistas em segurança afirmam que atualmente boa parte dos incidentes de segurança da informação ocorrem a partir de vulnerabilidades encontradas no software, componente presente em boa parte dos sistemas de informação. Para tornar o software fidedigno em relação à segurança, a criação e o uso de métricas de segurança serão fundamentais para gerenciar e entender o impacto dos programas de segurança nas empresas. Porém, métricas de segurança são cobertas de mistério e consideradas bastante difíceis de serem implementadas. Este trabalho pretende mostrar que hoje ainda não é possível termos métricas quantitativas capazes de indicar o nível de segurança que o software em desenvolvimento virá a ter. Necessitam-se, então, outras práticas para assegurar níveis de segurança a priori, ou seja, antes de se por o software em uso.

# Abstract

<escreva o abstract>

# INTRODUÇÃO

Actualmente nas organizações, a gestão de informação é um factor que se não pode contornar, devido os benefícios indispensáveis que ela dispõe para o desenvolvimento de qualquer uma instituição. Os serviços desempenhados nas instituições passaram a ser realizados com ajuda de softwares para facilitar o armazenamento e proteção das informações em diversas instituições, quer sejam elas públicas ou privadas (Sousa et al. 2018).

É inevitável a utilização de sistemas de informação porque eles proporcionam cada vez mais vantagens no gerenciamento das organizações (Panno et al. 2020)

Em uma organização, a informação de qualidade, disponibilizada em tempo útil, só se torna possível se a gestão adoptada for eficiente e eficaz. Nos dias de hoje, sendo este objectivo extremamente importante para os gestores, é possível realizá-lo por meio de suportes tecnológicos adequados que garantem a confiabilidade e que sejam fácil de usar (Resende, 2012).

As organizações com uma visão mais holística perceberam a necessidade de usar as informações de uma maneira eficiente. Várias instituições decidiram automatizar seus processos de gerência pelo facto de entenderem a importância que há nesta prática, tornando assim mais produtivos seus trabalhos (Panno at al. 2020).

Estamos numa era digital onde as novas tecnologias estão cada vez mais presentes no ensino, professores já não têm dificuldades no uso de diversos softwares como nos anos 90 uma vez que a maioria está habituada em fazer o uso dos mesmos para redigir um texto ou mesmo para auxiliá-lo na planificação (Kaufmann, 2005).

As instituições de ensino ao perceberem os benefícios existentes no uso de sistemas de informação para a gerência dos processos administrativos, optam por utilizar software de gestão escolar.

Martins et al. (2012), consideram que o sistema de informação é todo processo administrativo por intermédio do uso da tecnologia de informação, de estrutura e pessoas que compõem uma organização que servem para aperfeiçoar a gestão dentro dela.

O presente trabalho propõe a criação de um software de gestão escolar para o Liceu Felizberto D. Ngunga, situada no município de Catabola, província do Bié.

Com base nas pesquisas e análises feitas no Liceu Felizberto D. Ngunga de Catabola, constatou-se que o processo de gestão da informação na referida escola é feito com ajuda de ferramentas do pacote Office, nomeadamente **Word** e **Excel**.

Os processos que são levados em consideração nesta Escola são: processo de inscrição e de matrícula, processo de lançamento de notas, processo de solicitação de certificados habilitacionais e de declarações, processo de impressão de relatórios, processo de solicitação de declaração de serviço e o processo de justificação de faltas.

Refletindo nas actividades acima mencionadas o processo que o Liceu Felizberto D. Ngunga de Catabola adoptou, não é viável e faz com que a gestão seja lenta e não segura, causando desgaste mental e é menos eficiente para o atendimento das solicitações exigidas actualmente.

Tendo em consideração as informações levantadas, o processo não é eficaz para os dias actuais, o que permitiu identificar as seguintes **manifestações problemáticas**:

* Desgastes mentais ao pesquisar documentos anteriormente elaborados;
* Facilidade de esquecimento da ordem de numeração dos documentos;
* Atraso no lançamento de notas devido a limitações de usuários aos computadores indicados;
* Desespero dos alunos por causa da morosidade no levantamento de certificados e declarações;
* Eliminação de dados de funcionários ou alunos para criar um novo documento, como declarações de serviço e habilitacional;
* Custo de tempo na elaboração de diversos relatórios

**Problema de investigação**

Das referidas informações sobre a pesquisa e das manifestações problemáticas nos leva ao seguinte **problema científico: c**omo melhorar o processo de gestão escolar no Liceu Felizberto D. Ngunga de Catabola?

**Objeto de estudo**

Para dar solução ao problema da situação actual, tem – se como objecto de estudo: processos de Gestão escolar.

**Campo de acção**

Ferramentas informáticas para o desenvolvimento de sistemas Web para gestão escolar.

**Objectivo geral**

Desenvolver um sistema web de gestão escolar para o Liceu Felizberto D. Ngunga de Catabola;

**Hipótese**

Considerando o problema que se pretende dar solução e o objectivo a ser alcançado, caso se desenvolva um Sistema web de gestão escolar para aquela entidade, melhorará os serviços prestados pela instituição, garantindo o controlo de alunos e funcionários, e facilitará o processo de gestão da informação.

**Objectivos específicos ou tarefas**

* Caracterizar os antecedentes históricos e conceituais de processos de Gestão escolar e softwares de gestão escolar;
* Caracterizar a situação atual do processo de gestão escolar no Liceu Felizberto D. Ngunga de Catabola;
* Caracterizar as ferramentas informáticas utilizadas para o desenvolvimento web de gestão escolar do Liceu Felizberto D. Ngunga de Catabola;
* Analisar e desenhar o sistema web de gestão escolar para o Liceu Felizberto D. Ngunga de Catabola;
* Implementar o sistema web de gestão escolar no Liceu Felizberto D. Ngunga de Catabola;
* Validar o sistema web de gestão escolar do Liceu Felizberto D. Ngunga de Catabola por meio de teste de aceitação;

**Métodos e técnicas da investigação**

* **Análise documental –** Que serviu para interpretação de documentos do tipo: arquivos, registos estatísticos, diários, biografias, jornais, revistas, entre outros, que ajudaram na pesquisa para o desenvolvimento do sistema Web de Gestão escolar para o Liceu Felisberto D. Ngunga de Catabola.
* **Entrevista -** realizada no princípio e ao longo do desenvolvimento do sistema para o levantamento de informações sobre o processo de Gestão no Liceu Felizberto D. Ngunga de Catabola, posteriormente fez-se a análise sobre os dados recolhidos na entrevista para entender como ocorre o processo de Gestão escolar no Liceu Felizberto D. Ngunga de Catabola.

**População e amostra**

O Liceu Felizberto D. Ngunga de Catabola está constituído por um Director geral, dois subdirectores, um chefe de secretaria geral, quatro auxiliares administrativos e 28 professores pré-fazendo uma população de 36 elementos. No total de elementos já mencionados, foi selecionada a amostra de 7 elementos dos quais, cinco do corpo directivo e dois professores.

**Tipo de investigação**

Uma vez que a pesquisa realizada objectiva levar o conhecimento adquirido à prática, para gerar um produto de software, a investigação feita para o presente trabalho é considerada, quanto a sua natureza, investigação **aplicada**.

**Contribuição prática.**

Com o desenvolvimento do sistema web específico para a gestão escolar do Liceu Felizberto D. Ngunga de Catabola, haverá melhorias significativas nos processos de gerenciamento, tais como:

* Armazenamento de dados de forma segura no sistema informático;
* Informações integradas em uma base de dados;
* Rapidez na inserção e consultas de informações dos alunos e funcionários;
* Selecção automática de alunos admitidos para a matrícula na fase de inscrição;
* Rapidez em gerar documentos diversos (lista de funcionários, lista de alunos, Declaração de serviço e de alunos, Certificados, relatórios).
* Backup e restauro dos dados.

**Caracterização geral do relatório de investigação**

No que se refere a estrutura do presente trabalho, está dividido, para além da introdução, em três capítulos e culminando com as conclusões, recomendações, referencias bibliográficas e anexos; quanto aos capítulos tem-se:

O **Capítulo I:** apresenta os antecedentes históricos dos sistemas informático, conceitos de softwares relacionados ao processo de gestão escolar, assim como a metodologia a ser aplicada, linguagem e tecnologias a serem utilizadas para o desenvolvimento do sistema web de Gestão escolar para o Liceu Felizberto D. Ngunga de Catabola, e uma análise do estudo actual da entidade.

O **Capítulo II:** apresenta as descrições dos usuários e aspectos ligados a engenharia de *software* analisando de maneira detalhada as histórias e as tarefas de Engenharia com alta prioridade no sistema proposto ao Liceu Felizberto D. Ngunga de Catabola.

No **Capítulo III:** será apresentada a análise, desenho das interfaces e a implementação do sistema, tratamento de erros, assim como a planificação e execução dos testes de software e teste de aceitação.

# 

# CAPÍTULO I - FUNDAMENTAÇÃO TEÓRICA

Este capitulo apresenta a fundamentação teórica dos antecedentes históricos e conceituais das escolas do subsistema de ensino geral em Angola, o impacto das tecnologias de informação e comunicação(TIC) sobre as instituições de ensino, os conceitos de sistema de gestão escolar, os softwares já existentes, metodologias usadas para o desenvolvimento de software, caracterização das ferramentas e tecnologias para o desenvolvimento de um sistema web.

## Antecedentes históricos dos sistemas informáticos.

Os sistemas informáticos começaram a evoluir na década de 50, quando foi criada a primeira linguagem de programação, denominada FORTRAN, passos notáveis foram empregados nas tecnologias de desenvolvimento de aplicativos que permitem a gestão automática de informação (Veiga, 2004, p. 52).

O Século XX é marcado como o advento da Era da Informação. Foi a partir dele que a informação começou a fluir com uma velocidade superior a dos corpos físicos.

Desde a invenção do telégrafo elétrico em 1837, passando pelos meios de comunicação de massa, e até ao surgimento da Internet, o ser humano tem convivido com um crescimento exponencial do volume de dados disponíveis.

Antes da evolução e popularização dos computadores, os sistemas de informação nas organizações baseavam-se em técnicas de arquivamento e recuperação de informações de grandes arquivos. Geralmente existia a figura do bibliotecário, que é a pessoa responsável em organizar os dados, registrá-los, catalogá-los e recuperá-los quando necessário.

De acordo com Silva I. (2023) No continente africano, a gestão de informação e comunicação tornou-se mais generalizada e integrada, desde a Conferência Internacional sobre a Gestão da Informação dos Parlamentos em África, onde tratou-se dos Desafios e Oportunidades das TICs para o Reforço da democracia e Governação Parlamentar, realizada em Nairobi, Quénia, de 10 a 11 de Fevereiro de 2005, onde participaram Presidentes dos Parlamentos e delegações de alto nível dos Parlamentos da Argélia, Angola, Camarões, Gana, Quénia, Madagáscar, Malawi, Maurícias, Moçambique, Ruanda, São Tomé e Príncipe, Somália, Sudão, Tanzânia, Tunísia, Uganda, Zâmbia e Zimbabwe.

### Sistemas Informáticos (SI)

A evolução das tecnologias de informação e Comunicação (TIC), fez com que nos dias actuais, instituições adotassem o uso de sistemas informáticos (SI) para facilitar suas tarefas no gerenciamento de informações.

Sistema de informação é um termo utilizado para descrever um processo automático de gestão da informação. Implica a integração de várias partes, de hardware e software para gerir um fluxo de informações de maneira confiável e menos burocrática (Veiga F. J., 2006, p. 2).

Segundo Caldeira (2011, p. 11), sistema de informação é um tipo especial de sistema por ser constituído por pessoas, procedimentos e equipamentos interdependentes com diversos mecanismos de controlo nos dados processados.

Os autores Panno, Senger, Franciscato, e Silveira (2020, p. 17) defendem a ideia de que sistema informático refere-se a uma associação de ferramentas tecnológicas, altamente sofisticadas, concretamente, computadores, softwares, hardwares, banco de dados, infraestrutura de telecomunicações, redes, intranet e internet, entre outros.

### Automatização dos processos nas organizações

Desde a invenção do computador, a sociedade corporativa notou o início de uma revolução na história da humanidade. As mudanças científicas e tecnológicas que surgiram como resultado desse facto causaram modificações acentuadas nas diversas áreas como sociais, económicas, políticas, entre outras. Estas mudanças requerem uma rápida adaptação (Cruz, 2019 , p. 13).

As ferramentas de tecnologia da informação têm gerado benefícios em vários sentidos dentro de uma organização devido à amplitude que elas proporcionam no processo decisório baseado na capacidade de processamento de informações (Lemos, 2015, p. 15).

Segundo Lemos (2015, p. 16) “Com os recursos das tecnologias da informação, a organização flexível tem a possibilidade de interagir eletronicamente com seus fornecedores, com seus distribuidores e consumidores e, complementarmente, com entidades parceiras em alianças estratégicas”.

Com o uso dos recursos tecnológicos, as organizações não são limitadas na interação com sua comunidade, fornecedores, clientes e outras empresas, pois torna – se possível a comunicação além do espaço físico da organização com a garantia de uma conexão completa com a sociedade (Lemos, 2015, p. 16).

Bancovsky (2008, p. 28), acredita que os gestores das escolas podem exercer suas funções administrativas e pedagógicas com o auxilio das TICs para aumentarem suas produtividades. Para tal é necessário que os mesmos tenham cada vez mais uma formação para melhor integração com estes recursos.

## Gestão escolar e TICs

As instituições estão cada vez mais veredando para o uso das tecnologias de informação e comunicação (TIC), lá se foram os modelos de armazenamento de documentos em apenas armários criando um acervo empilhado de arquivos que frustravam os gestores escolares. O mundo mudou com a transformação proporcionada pelas tecnologias digitais.

Estamos numa era globalizada onde a utilização das TICs é indispensável no nosso quotidiano. Estão presentes e influenciando na facilidade de gerir informações por meio de dispositivos pessoais e outros. A escola, estando ela inserida nesta sociedade global, não fica de fora nos benefícios deste crescente evento (Gonçalves, 2012, p. 1).

“Esta nova sociedade cresceu e desenvolveu-se paralelamente com a rápida evolução das TIC, que se tornaram de fácil acesso e passaram a ser indispensáveis na vida pessoal e profissional de cada indivíduo, influenciando o seu modo de vida e a forma como se organiza. O uso das TIC, especialmente da internet, veio possibilitar a troca de informação na comunidade académica e científica, proporcionando novas formas de comunicação e de colaboração” (Domingues, 2017, p. 3)

Segundo Júnior et al. (2010, p. 423), já há um tempo que as TICs conquistaram seu espaço nas organizações, assumindo um papel importante na gestão. Muitas instituições as têm como elemento chave para o sucesso da organização, tornando mais eficiente o processo de gestão organizacional.

As TICs são ferramentas úteis e aplausíveis no processo de gestão escolar porque garantem maior controle dos dados armazenados, permitem o acesso rápido às informações contidas no sistema. Actuam como aliados do processo de gestão das escolas ajudando na tomada de decisões (Júnior et al., 2010, p. 428).

## Sistema informático de Gestão escolar

Considerando que a escola está inserida em um ambiente global, tecnológico é necessário que ela esteja preparada para atender as exigencias dos novos tempos.

A gestão escolar deve ser feita com uso de recursos tecnológicos que possam permitir um acesso imediato pela comunidade escolar tornando mais fácil a tarefa dos professores. Para isso, o gestor educacional deve ter a iniciativa de introduzir inovações tecnológicas na sua instituição (Silva, Lima, & Souza, 2019, p. 45).

“Uma das principais decisões que um gestor pode tomar com o objetivo de otimizar processos e prover avanços consideráveis em uma instituição de ensino é a escolha de um sistema de Gestão Escolar (ou Sistema de Gestão Educacional) apropriado ao ambiente em que atua. Nesta ótica, a implantação de um bom Sistema de Gestão Escolar pode trazer inúmeros benefícios à organização, que vão desde a organização da secretaria até o controle das finanças e a comunicação com os pais dos alunos” (Panno et al., 2020, p. 47 apud Wpensar, 2017).

## Situação actual do processo de Gestão escolar do Liceu Felizberto D. Ngunga de Catabola

A gestão da informação no Liceu Felizberto D. Ngunga de catabola, é feita à base de manuscritos, e com auxílio das ferramentas do pacote Microsoft Office, especificamente o Microsoft Word e Microsoft Excel.

## Ferramentas para o desenvolvimento de um sistema informático

É necessário definir ferramentas e tecnologias adequadas para construir uma aplicação web, como por exemplo, a linguagem de programação a utilizar, base de dados, entre outras, para que se tenha um desempenho esperado na produção.

### Metodologias de desenvolvimento de um software

A metodologia é o estudo dos métodos ou regras que nos permite chegar a um determinado objectivo. É a área da ciência que se preocupa com a descrição e análise dos métodos (SOUSA, LOPEZ, & ANDRADE, 2007, p. 4).

As metodologias de desenvolvimento de software são encaradas como um processo de construção de software que caracteriza um modelo com perspectivas particular e informações parciais sobre o mesmo. É uma sequência de actividades para a produção de um software (Sommerville, 2011, p. 19).

De acordo com (Neto, 2016, p. 23), podemos observar que existem modelos de processos de desenvolvimento de um software que vão diferenciando nas suas descrições de como são realizadas as tarefas. O autor descreve assim a existência de metodologias tradicionais e Metodologias ágeis.

#### Metodologias Tradicionais

##### Modelo Cascata

O modelo Cascata é uma das metodologias tradicional, muito conhecido e ainda utilizado no desenvolvimento de software. É caracterizado por meio de etapas bem definidas, o ciclo que o software cumprirá durante o período compreendido entre sua concepção e sua descontinuidade. O ciclo de vida de um software é constituído de seguintes fases: concepção, construção, implantação, implementações, maturidade, declínio, manutenção e descontinuidade (Neto, 2016, p. 23).

##### Rational Unified Process (RUP)

O RUP é um modelo de desenvolvimento de software que une os processos dos modelos tradicional anteriores tornando assim um modelo de processo moderno. É um modelo de processo hibrido devido a junção de actividades oriundas das metodologias que lhe antecedem (Sommerville, 2011, p. 34). Este modelo reconhece que os modelos de processo convencionais apresentam uma única visão do processo.

Segundo (Sommerville, 2011), o RUP é descrito em três perspectivas:

1. Uma perspectiva dinâmica, que mostra as fases do modelo ao longo do tempo;
2. Uma perspectiva estática, que mostra as atividades realizadas no processo;
3. Uma perspectiva prática, que sugere boas práticas a serem usadas durante o processo.

O RUP identifica quatro fases no processo de construção de software. Estas fases estão estreitamente relacionadas ao negócio, e não a assuntos técnicos; tais fases são:

1. Concepção;
2. Elaboração;
3. Construção;
4. Transição.

Para (Pressman, 2011, p. 72), o processo unificado (RUP), apresenta cinco actividades metodologicas como qualquer outro modelo de processo de software como ilustra a imagem a seguir.

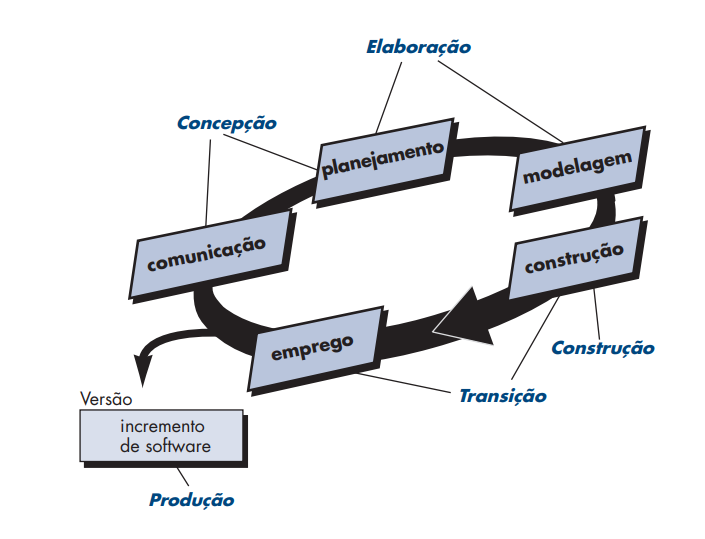


Figura 1 - Fase de desenvolvimento de um projecto

De acordo com o autor:

A fase de **concepção** envolve tanto a atividade de comunicação com o cliente como a de planejamento. Colaborando com os interessados, identificam-se as necessidades de negócio para o software; propõe-se uma arquitetura rudimentar para o sistema e se desenvolve um planejamento para a natureza iterativa e incremental do projeto decorrente.

Requisitos de negócio fundamentais são descritos por meio de um conjunto de casos práticos preliminares, descrevendo quais recursos e funções cada categoria principal de usuário deseja. Nesta fase é arquitectada um esquema provisório dos principais subsistemas e da função e dos recursos que os compõem. Posteriormente, a arquitetura será refinada e expandida para um conjunto de modelos que representarão visões diferentes do sistema.

A fase de **elaboração** envolve atividades de comunicação e modelagem do modelo de processo genérico. A elaboração refina e expande os casos práticos preliminares, desenvolvidos como parte da fase de concepção, e amplia a representação da arquitetura.

A fase de **construção** do é idêntica à atividade de construção definida para o processo de software genérico. Tendo como entrada (input) o modelo de arquitetura, a fase de construção desenvolve ou adquire componentes de software; esses componentes farão com que cada caso prático (de uso) se torne operacional para os usuários finais.

A fase de **transição** do abarca os últimos estágios da atividade da construção genérica e a primeira parte da atividade de emprego genérico: entrega e realimentação (feedback). Entrega- -se o software aos usuários finais para testes beta e o feedback dos usuários relata defeitos e mudanças necessárias.

A fase de **produção** do PU coincide com a atividade de emprego do processo genérico. Durante essa fase, monitora-se o uso contínuo do software, disponibiliza-se suporte para o ambiente (infraestrutura) operacional, realiza-se e avalia-se relatórios de defeitos e solicitações de mudanças.

Para além das fases apresentadas, o RUP apresenta fluxos de trabalho que ajudam a uma boa pratica no processo de construção de produto de software conforme se pode descrever a seguir.

1. Requisitos - este fluxo visa obter um conjunto de requisitos de um produto, acordado entre cliente e fornecedor.
2. Análise - objectivo deste fluxo é detalhar, estruturar e validar os requisitos, de forma que estes possam ser usados como base para o planejamento detalhado.
3. Desenho - formular um modelo estrutural do produto que sirva de base para a implementação.
4. Implementação - fluxo cujo objectivo é realizar o desenho em termos de componentes de código.
5. Testes - fluxo cujo objectivo é verificar os resultados da implementação.

#### Metodologias Ágeis

Para Fontoura (2019, p. 14), as metodologias ágeis surgiram para diminuir os problemas que as metodologias tradicionais não conseguiam dar respostas. Métodos ágeis possibilitam trabalhar de uma forma iterativa e incremental, permitindo que se avança uma fase sem depender da outra, o que não é possível nos modelos tradicionais.

A metodologia ágil adaptasse incrivelmente aos requisitos do cliente uma vez que estes na maioria das vezes não são permanentes. O Cliente é imprevisível, os processos ágeis devem ser capazes de administrar esta situação, ser adaptável, mas de uma maneira incremental para não impedir o progresso do processo (Pressman, 2011, p. 84).

Existem 12 princípios da agilidade para quem quiser a ter (Pressman (2011):

1. A maior prioridade é satisfazer o cliente por meio de entrega adiantada e contínua de software valioso.
2. Acolha bem os pedidos de alterações, mesmo atrasados no desenvolvimento. Os processos ágeis se aproveitam das mudanças como uma vantagem competitiva na relação com o cliente.
3. Entregue software em funcionamento frequentemente, de algumas semanas para alguns meses, dando preferência a intervalos mais curtos.
4. O pessoal comercial e os desenvolvedores devem trabalhar em conjunto diariamente ao longo de todo o projeto.
5. Construa projetos em torno de indivíduos motivados. Dê a eles o ambiente e apoio necessários e confie neles para ter o trabalho feito.
6. O método mais eficiente e efetivo de transmitir informações para e dentro de uma equipe de desenvolvimento é uma conversa aberta, de forma presencial.
7. Software em funcionamento é a principal medida de progresso.
8. Os processos ágeis promovem desenvolvimento sustentável. Os proponentes, desenvolvedores e usuários devem estar capacitados para manter um ritmo constante indefinidamente.
9. Atenção contínua para com a excelência técnica e para com bons projetos aumenta a agilidade.
10. Simplicidade - a arte de maximizar o volume de trabalho não efetuado - é essencial.
11. As melhores arquiteturas, requisitos e projetos emergem de equipes que se auto-organizam.
12. Os intervalos regulares, a equipe se avalia para ver como tornar-se mais eficiente, então sintoniza e ajusta seu comportamento de acordo.

“Nem todo modelo de processo ágil aplica esses 12 princípios atribuindo-lhes pesos iguais, e alguns modelos preferem ignorar (ou pelo menos relevam) a importância de um ou mais desses princípios. Entretanto, os princípios definem um espírito ágil mantido em cada um dos modelos de processo apresentados neste capítulo.” (Pressman, 2011, p. 85)

Existem diversas metodologias ágeis para o desenvolvimento de um produto de software. Neste trabalho, destacou-se, dentre tantas, a metodologia Scrum e XP.

##### Metodologia Scrum

É uma das metodologias ágeis muito conhecida que tem como vantagem o foco no andamento de um projecto considerando três factores importantes que permitem a colaboração de todas as partes envolvidas no processo de uma forma efectiva: Scrum Master (guardião da metodologia na organização), Product Owner (cliente) e Team (equipe).

O Scrum é utilizado pelas grandes empresas na actualidade. Ela pode ser aplicada tanto no desenvolvimento de software ou noutras áreas. Nesta metodologia, o desenvolvimento do projecto, começa com o levantamento dos requisitos do sistema que permite elaborar uma lista dos mesmos a partir do usuário do sistema, de uma maneira ordenada, afim de serem executadas as prioridades identificadas (SANTI & BEGOSSO, 2015, p. 9).

Esta metodologia enfatiza um conjunto de padrões de processos de software que são eficazes para projetos que apresentem requisitos mutáveis ao longo do processo, projectos que tem um grau de execução urgente e curto prazos de entrega (Pressman, Engenharia de Sooftware uma Abordagem profissional, 2016).

##### *Extreme Programming* (XP)

*Extreme Programming* (XP) é considerado como o mais conhecido e mais utilizado dos métodos ágeis. O nome foi atribuído pelo  engenheiro de software Kent Beck, na sua abordagem desenvolvida para impulsionar boas práticas, recomendáveis para o desenvolvimento iterativo (Sommerville, 2011).

A metodologia XP é recomendável para equipes pequenas e médias onde se pode perceber um vazio nos requisitos, com a intensão de modificá-los rapidamente. Em XP as boas práticas de programação são usadas de forma intensiva pelo facto de que os códigos e outras tarefas serem passados em revisão constante através da programação em pares (Fontoura, 2019, p. 26 apud Beck, 1999).

XP restringe aos desenvolvedores a projetar apenas para as necessidades imediatas, limita – se no princípio com as necessidades futuras, porque o foco inicial é criar um projeto simples que possa ser facilmente implementado em código. Se o projeto tiver que ser melhorado, ele poderá ser refabricado posteriormente (Pressman, 2011, p. 87).

Por razões apresentadas anteriormente, escolheu-se a metodologia ágil *Extremme Programming* (XP) para o desenvolvimento do sistema web para gestão escolar do Liceu Felizberto D. Ngunga de Catabola. Também por ser uma metodologia projetada para projetos pequenos, possibilitando o desenvolvedor realizar diversos papéis, e, que por sua vez se encacha muito bem no desenvolvimento deste sistema, e por ser uma metodologia conhecida pelo desenvolvedor.

### Linguagens de programacão web

A seleção de uma linguagem de programação pode determinar o sucesso ou fracasso do processo de construção do sistema, e até mesmo o produto de software. A linguagem de programação bem definida contribui positivamente no desenvolvimento de aplicações web, no caso de um sistema web que é o foco deste trabalho. Por isso, é preciso ter a atenção voltada para, por exemplo, se queremos uma linguagem de programação que seja Server-side ou client-side, pois cria impacto no tráfego na rede, bem como do lado dos servidores (Carvalho, 2015, p. 11).

Para desenvolver um software é importante selecionar uma linguagem de programação que possa possibilitar a equipe desenvolvedora a codificar os requisitos do sistema, de acordo com os requisitos desejados pelo cliente. Actualmente existem diversas linguagens de programação a dispor dos programadores. Para que o critério de seleção da linguagem de programação seja bem definido, apresenta – se abordagem de 3 linguagens de programação web, nomeadamente PHP, JavaScript e ASP.NET, descrevendo algumas de suas vantagens e desvantagens.

**PHP** é uma linguagem de programação que permite criar aplicações web, possibilitando uma interação com o usuário através de formulários, parâmetros da URL e links.

A licença de uso e edição no PHP é Open Source, ou seja, é de código fonte aberto, possibilita aos utilizadores explorar e modificar. Este sistema de licença não traz lucro aos programadores, pois estes disponibilizam tudo para o público gratuitamente. (Carvalho, 2015, p. 12).

O autor Carvalho (2015), aponta algumas vantagens e desvantagens no PHP.



Figura 2 - PHP

**Principais vantagens do PHP**

* O facto de ser grátis - não possui encargos para se utilizar;
* Linguagem de programação de fácil aprendizagem - o PHP tem elementos do Perl, Java e C. Como a maioria dos programadores conhece pelo menos uma destas linguagens, a aprendizagem fica mais facilitada;
* Acesso a bases de dados - o PHP permite implementar facilmente a ligação entre
* sistemas compatíveis com o padrão ODBC3;
* Multiplataforma - o PHP funciona em qualquer plataforma onde for possível instalar um servidor Web (Linux, Solaris, Windows, etc).

**Desvantagens do PHP**

* Incompatibilidade entre versões - falta alguma padronização, pois por exemplo, um comando que funciona em determinada versão do PHP pode não funcionar noutra;
* Documentação incompleta - os recursos surgem antes de estarem documentados;
* Suporte para datas - é possível fazer cálculos utilizando datas, mas é mais limitado quando comparado com o ASP, por exemplo.

**JavaScript**

É uma linguagem de programação muito utilizada no desenvolvimento de aplicações para a Web. Surgiu com o aparecimento de uma linguagem de criação de Scripts executados no servidor pela Netscape. é uma linguagem de programação que é executada do lado cliente, e possibilita adicionar recursos dinâmicos às páginas HTML.

O código do JavaScript é executado do lado do cliente, o que facilita libertar recursos do lado do servidor. Permite o desenvolvimento do código dentro do código HTML, através das tags “<script> e </script>”.



Figura 3 - Líguagem de programação JavaScript

**Principais vantagens do *JavaScript***

Interatividade com o utilizador - o JavaScript melhora a interatividade do utilizador em websites ao utilizar recursos, como por exemplo caixas pop-up e ferramentas de navegação;

Suporte de imagens como links – permite o uso de imagens como links em documentos Web. Pode-se utilizar esta vantagem para criar animações, por exemplo;

Suporta validações do lado cliente - o *JavaScript* pode ser validado do lado do cliente.

Não é necessário enviar os dados para a validação do servidor. Enviar dados para o servidor para validação, origina mais tráfego, portanto, o *JavaScript* pode ser usado para reduzir o tempo de validação;

Suporta identificação do browser - o *JavaScript* pode automaticamente identificar o tipo e versão do browser. É possível utilizar o código *JavaScript* para enviar comandos diferentes para navegadores diferentes;

Suporta deteção de plug-in - O *JavaScript* pode detetar automaticamente os plug-ins do browser. Pode-se utilizar o *JavaScript* para facilitar a instalação de plug-ins (caso o browser do utilizador não o tenha instalado). Se o *JavaScript* deteta um browser sem plug-ins, ele ignora os plug-ins utilizados no código e modifica a página Web para permitir que ela seja executada no browser

**Desvantagens do *JavaScript***

Sobreposição de funcionalidades - pode apresentar em relação aos estilos CSS alguma sobreposição de funcionalidade, como por exemplo *mouse hover* (que pode ser obtido com outro recurso). A aplicação dos dois recursos ao mesmo objeto pode gerar conflito;

Capacidade de operação limitada, devido a restrições de segurança - o *JavaScript* não pode interferir com as configurações do sistema operativo, como por exemplo ativar programas instalados. Também não pode alterar o layout de uma página com origem noutro domínio.

Capacidade de operação limitada pelo próprio utilizador, nas opções dos browsers. Muitos utilizadores desativam os scripts dos seus browsers para abrir pop-ups, escrita de textos na barra de estado.

***ASP.NET***

O *ASP.NET* é uma tecnologia de desenvolvimento de aplicações Web, que foi lançada oficialmente em 2002 juntamente com a *framework .NET 1.0*. Na altura era conhecido como *ASP+* por ser o sucessor direto do ASP 3.0 (Clássico). Foi o primeiro *framework* de desenvolvimento web da *Microsoft*.



Figura 4 - ASP.NET

**Linguagem web adoptada**

Após a análise das diversas linguagens de programação web, achou-se conveniente optar por utilizar o *JavaScript* por ser uma linguagem que não possui custos de utilização, suporta validações do lado cliente, factor que permite aliviar a carga na rede e no servidor, e normalmente é compatível com diversos browsers sem precisar de *plugins* adicionais, e acima de tudo é a linguagem de desenvolvimento web que o programador domina.

**Outras Linguagens de programação e ferramentas de desenvolvimento de software**

**JAVA**

Java é uma linguagem de programação orientada a objectos, robusta, e com todas as características das linguagens modernas, é utilizada nos mais diversos ambientes, desde aplicações simples no desktop até complexos aplicativos web. Com ela se pode fazer a programação de robôs, redes de sensores, celulares e televisão digital interativa e outros.



Figura 5 - Linguagem de programação Java

**Python**

Trata-se de uma linguagem multiplataforma que serve para quase tudo, ou seja, permite desenvolver aplicações para celulares, desktop e web, entre outros. Além disso, permite integrar diversos serviços e sistemas de forma rápida e eficaz.



Figura 6 - Linguagem de programação Pyton

**Ruby**

É uma das linguagens de programação mais simples, elegantes e fáceis de aprender: fácil de programar tem foco em produtividade, é poderosa e eficiente. Ficou famosa ao ser usada com o *framework Rails (Ruby on Rails).* A demanda por programadores *Ruby* tem crescido nos últimos anos.



Figura 7 - Linguagem de programação

**Linguagem C#**

É uma linguagem de programação criada pela *Microsoft* como parte da plataforma *NET*. Por ter uma sintaxe parecida com o Java C++, o C# pode ser indicada mesmo para programadores iniciantes, que já conhecem a programação orientada a objectos. Com a linguagem C#, pode se criar desde aplicações web, desktop, mobile e também jogos eletrónicos.

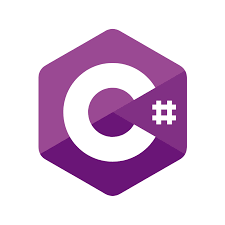


Figura 8 - Linguagem de programação C#

**HTML5**

*Hypertext Markup Language* (Linguagem de Marcação de Hipertexto), uma linguagem de marcação ou *markup* que funciona através de “marcações” chamadas tags ou etiquetas, com o propósito de estruturar e definir a natureza do conteúdo de uma página.



Figura 9 - Linguagem de marcação de texto HTML

**Visual code (VS Code)**

O Visual Studio Code é um editor de código de código aberto desenvolvido pela Microsoft. Funciona em plataforma Windows, Mac e Linux. É criado com Electron, ferramenta criada pelo GitHub que permite a criação de sistemas web.



Figura 10 - Editor de código VS Code

**Node Js**

Node.JS é uma ferramenta amplamente utilizada no universo de desenvolvimento. É um runtime, que nada mais é do que um conjunto de códigos, API’s, ou seja, são bibliotecas responsáveis pelo tempo de execução. De uma forma mais clara, node js é um interpretador de *JavaScript* fora do ambiente do navegador web.

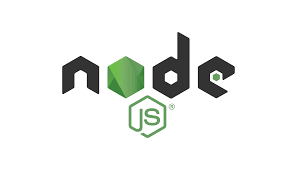


Figura 11 - Node JS

**XAMMP**

O XAMPP é um pacote com os principais servidores de código aberto do mercado, incluindo FTP, banco de dados *MySql* e Apache com suporte as linguagens PHP. Com ele, é possível rodar sistemas como *WordPress* e *Drupal* localmente, o que facilita e agiliza o desenvolvimento.

Na prática actual, *XAMPP* é utilizado atualmente para servir sites web na WWW (*World* *Wide* *Web*), e com algumas modificações é geralmente seguro para uso em servidor público. Uma ferramenta especial é incluída para proteger facilmente as partes mais importantes e sensíveis do pacote. O XAMPP possui muitos aplicativos, dentre eles o Apache, *MySQL*, *phpMyAdmin*, *FileZilla* FTP Server, *OpenSSL*.



Figura 12 - Xamp

**React Js**

Segundo Lins ( 2019, p. 12), *ReactJS* é uma biblioteca *Front*-end baseada na linguagem *JavaScript* que permite desenvolver interfaces baseadas em componentes para aplicações web. Alguns conhecimentos necessário para sua utilização, são componentização, estado, propriedades, sintaxe JavaScript Sintax Extension (JSX).

Esta biblioteca pode ser utilizada tanto para o desenvolvimento de aplicações web quanto para aplicações mobile. O *React* é capaz de manipular elementos visuais de forma que se possa criar componentes reutilizáveis.

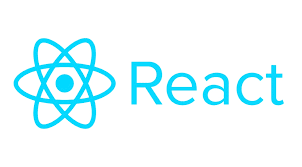


Figura 13 - React

***Bootstrap***

*Bootstrap* é um dos *frameworks* mais populares para construção de sites responsáveis ​​compatíveis tanto com computadores desktop, *smartphones* ou dispositivos com telas de diversos tamanhos.

Em sua quinta versão, *Bootstrap* traz uma série de novidades, como nomes de classes mais simplificados, novos recursos em *JavaScript* (*jQuery* não é mais necessário) e finalmente *Bootstrap* passou a ter uma biblioteca de ícones (Mariano, 2022, p. 8).



Figura 14 - Biblioteca Bootstrap

#### Sistema de Banco de dados

**MySQL**

MySQL é um Sistema de Gerenciamento de Bancos de Dados relacional. Um banco de dados relacional armazena dados em tabelas separadas em vez de colocar todos os dados em um só local. MySQL é um software cujo código fonte é aberto. Código fonte aberto significa que é possível para qualquer um usar e modificar o programa. Qualquer pessoa pode fazer download do MySQL pela Internet e usá-lo sem pagar nada (o MySQL só é cobrado em alguns poucos casos) (Quesada, 2001).

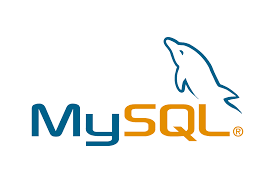


Figura 15 - MySQL

**MongoDB**

O *MongoDB* é um banco de dados baseado em documentos (*document database*) no formato JSON. Apresenta informações necessária, sem a restrição dos bancos relacionais. Em um documento podem existir um valor simples, como um número, uma palavra ou uma data, assim como uma lista de valores. Os documentos são agrupados em *collections* que formam um banco de dados (Boaglio, 2015, p. 25).



Figura 16 - MongoDB – Banco de dado

Vários desenvolvedores consideram o MongoDB um banco de dados que melhor se integra com o javascript devido a forlulação em formato JSON que é o mesmo da linguagem adotada para o desenvolvimente deste trabalho. Por esta razão e pela simplicidade que este apresenta, achou-se conveniente a utilização do MongoDB para o armazenamento de dados no desenvolvimento deste sistema.

## Sistemas web de Gestão escolar

As instituições de ensino podem fazer uso de um Sistema de Gestão Escolar. Actualmente os softwares de gestão abrangem diferentes segmentos educacionais, que vão desde o jardim de infância até aos níveis superiores (Panno, Senger, Franciscato, & Silveira, 2020, p. 47 apud Wpensar, 2017)

### – Sistemas existentes

**Software de Gestão Cacimbo**

O Cacimbo é um software de gerenciamento adaptavel para alguns serviços. Dentre estes, a gestão escolar. Várias instituições ja adotaram este software, tem ajudado no processo de gestão escolar permitindo fazer control de alunos e funcionários, gestão financeira e outras funcionalidades.

Embora com vantagens recentemente mencionadas, o cacimbo tem apresentado algumas falhas, como por exemplo, pouca restrições, que tem permitido violar a integridade dos dados, pois pessoas não autorizadas não poderiam ter acesso a certas funcionalidades, alem disso, não atende as necessidades do Liceu Felizberto D. Ngunga de Catabola, por ser um software que fuciona Descktop. Por tanto, não é o mais adequado para a proposta de gestão escolar que se quer melhorar.

**Software de gestão escolar Control+**

Control+ é um Sistema de gestão escolar online que permite fazer o cadastro de alunos e funcionários, gerenciar actividades administrativas da escola e impressões de relatórios.

Sua desvantagem centra-se na configuração de um novo ano lectivo e em algumas funcionalidades que não cumpre com o requisito de simplicidade, pois a sua criação seria para evitar dar vários clikis mas isto não foi atingido porque há um grau de complexidade exagerada. Além disso, não atende a necessidade de visualizar as notas pelo aluno, lançar nota pelo professor, requisitos estes que se pretende implementar na proposta para Liceu Felizberto D. Ngunga de Catabola. Por esta razão não se recomenda a adoção deste software para a entidade Cliente.

## Conclusões do Capitulo I

Neste capítulo foram apresentado várias abordagem que se deve levar em consideração para desenvolver um sistema web de gestão escolar, assim como as ferramentas e tecnologias adequadas que possam dar qualidade ao processo de construção de modo a atingir os objectivos desejados na instituição que se propõe o projecto e na comunidade em geral.

# CAPÍTULO II: EXPLORAÇÃO, PLANEJAMENTO E IMPLEMENTAÇÃO

Este capítulo apresenta o desenvolvimento do sistema web para gestão escolar do Liceu Felizberto D. Ngunga de Catabola, utilizando a metodologia XP, como foi anunciado no capitulo anterior, pelo qual se obteve a compreensão do negócio possibilitando extrair os requisitos funcionais e não funcionais do sistema em questão. Segue também neste capítulo a descrição dos artefactos que propõe a metodologia selecionada, as histórias de usuários, tarefas de engenharia e provas de aceitação.

## 2.1. Requisitos funcionais e não funcionais

Os requisitos de um sistema têm a haver com as definições do que o sistema deverá fazer assim como os serviços que oferece e as restrições declaradas para o seu funcionamento. Refletem as necessidades dos clientes e a garantia do correcto funcionamento do sistema (Sommerville, 2011).

Analisar requisitos de um sistema garante fornecer mecanismo adequado para entender o desejo do cliente com relação ao que o software deve fazer. No estudo de requisitos são analisados diversos factores importantes para o funcionamento do sistema, como por exemplo, as necessidades, avaliando a viabilidade, negociando uma solução razoável, especificando a solução sem ambiguidades, validando a especificação e outros.

Ainda (Pressman, 2011, p. 151), afirma que análise de requisitos resulta na especificação de características operacionais do software. Além disso, indica elementos que colaboram para a facilidade de uso, como a interface do software e a segurança do sistema garantindo restrições que o software deve atender.

Entendendo assim as visões dos autores, sobre os requisitos, podemos perceber duas categorias que se denominam requisitos funcionais e requisitos não funcionai.

**Requisitos funcionais**

Os requisitos funcionais descrevem o que o sistema deverá fazer tendo em conta tipo de software a ser desenvolvido, e as necessidades impostas pelo usuário (Sommerville, 2011, p. 59).

Apos ter sido feito o levantamento dos requisitos no Liceu Felizberto D. Ngunga de Catabola, analisando-os, apurou-se os seguintes requisitos funcionais, conforme detalhados na tabela a seguir.

Tabela 1 - Descrição dos Requisitos Funcionais

|  |  |  |
| --- | --- | --- |
| **N/O** | **Requisitos Funcionais /Função** | **Descrição** |
|  | Logar | Tarefa inicializada pelo usuário do sistema(funcionários de instituição ou alunos) |
|  | Autenticar usuário | Ação da responsabilidade do sistema que é disparada para terminar a função Logar. |
|  | Gerir usuários | Acção exclusiva do administrador de sistema. |
|  | Gerir configurações | Acção da inteira responsabilidade do administrador do sistema. |
|  | Fazer inscrição de novos alunos | Ação inicializada pelo funcionário(usuário) autorizado ou pelo administrador do sistema. |
|  | Matricular alunos admitidos | Acção realizada pelo usuário (Funcionário da instituição indicado para efeito de inscrições e matrículas) do sistema. |
|  | Confirmar matricula | Acção realizada pelo usuário (Funcionário da instituição indicado para efeito de inscrições e matrículas) do sistema ou pelo aluno de modo particular. |
|  | Fazer lançamento de notas | Ação realizada pelo funcionário(usuário) autorizado tendo em conta as restrições nas disciplinas. |
|  | Visualizar notas, para alunos | Acção realizada de modo particular, pelos alunos, professores e outros usuários autorizados |
|  | Imprimir relatórios | Ação autorizada apenas para funcionários administrativos com nível indicado para o efeito. |
|  | Imprimir declarações e certificados | Ação restrita de acordo ao nível de acesso |
|  | Actualizar dados armazenados | Ação restrita de acordo ao nível de acesso |
|  | Fazer cópia de segurança e restauro | Acção da responsabiliddade do Administrador do sistema. |
|  | Gerir informações da escola | Acção restrita de acordo a área |
|  | Notificar alunos ou encarregados de educação | Acção restrita de acordo a área |
|  | Realizar consultas de dados | Acção restrita de acordo a área |

**Requisitos não funcionais**

São aqueles que não estão directamente ligados com as funcionalidades do sistema. Podem ter relações directa com as propriedades do sistema que garantem a confiança e a facilidade de uso do mesmo (Sommerville, 2011, p. 60).

Para o referido sistema pretende-se garantir as seguintes qualidades de acordo aos seguintes requisitos não funcionais:

**Segurança**: O sistema deverá ser protegido, por meio de senhas de autenticação criptografia e restrições de acesso. Só o administrador do sistema poderá ter acesso a todas funcionalidades do Sistema e efetuar certas operações apenas a ele restritas. O sistema deve usar algum tipo de encriptação.

**Confidencialidade**: neste requisito se pretende restringir certas informações para que não sejam acessada por qualquer um. Para tal as Informações deverão ser acessadas pelos usuários devidamente cadastrados e de acordo ao seu nível de acesso no sistema.

**Integridade**: os dados devem estar protegidos com o maior nível de segurança, armazenado de modo a não serem mudados ou corrompidos; devem acima de tudo ser consistentes.

**Usabilidade**: o sistema deve ser de fácil uso de modo a facilitar ao usuário final sua execução.

Interface Externa

* Muito legível
* Simples de usar
* Interativo

**Suporte**: testar constantemente o software para que o mesmo não fique estático e sofra alterações com melhorias.

**Portabilidade**: o sistema é web, permitirá que diversas plataformas possam acessá-lo usando um navegador.

**Software**: deve ter instalado o um navegador, caso essteja em ofline, e usar o sistema gestor de base de dados MongoDB.

**Hardware**: um computador com capacidade mínima de 1 GB RAM, e 10 a 20 GB de espaço livre no disco rígido para melhor desempenho.

#### 2.1.1. Histórias de Usuários (HU)

Histórias de usuários podem ser encaradas como a base que sustentam o planejamento dos desenvolvedores, divididas em tarefas, que constituem detalhes mais preciso, no caso de serem extensas demais para apenas um dia de trabalho (Neto, 2016).

As HU são parte da versão de software pelo facto de descrever detalhadamente uma determinada acção. No entanto, pode-se testar a aceitação de uma operação junto do cliente (Pressman, 2011, p. 90).

Para calcular o tempo de execução de uma HU, em seus pontos estimados e real, definiu-se que 0.2 refere-se a um dia, 0.4 refere-se a dois dias, 1.0 refere-se a uma semana que são cinco dias de trabalho, 2.0 refere-se a duas semanas de trabalho

Com base nos requerimentos funcionais, a tabela a seguir apresenta as HU e suas estimativas:

Tabela 2 -Histórias de Usuários

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Iteração** | **Nº** | **História de Usuário** | **Ponto estimado** | **Ponto real** |
| 1 | 1 | Gerir usuários | 0.4 | 1.0 |
| 2 | 2 | Gerir candidatos | 2.0 | 3.0 |
| 3 | Gerir configurações | 1.0 | 3.0 |
| 3 | 4 | Gerir matricula | 0.4 | 1.0 |
| 5 | Fazer lançamento de notas | 1.0 | 2.0 |
| 6 | Editar notas | 0.6 | 0.6 |
| 7 | Visualizar notas, para alunos | 0.8 | 1.0 |
| 4 | 8 | Gerir informações da escola | 0.8 | 2.0 |
| 9 | Imprimir relatórios | 0.4 | 0.4 |
| 10 | Imprimir declarações e certificado de Habilitação | 0.4 | 0.4 |
| 5 | 11 | Fazer cópia de segurança e restauro | 0.4 | 0.4 |
| 12 | Notificar alunos ou encarregados de educação | 0.4 | 0.4 |
| 13 | Realizar consultas de dados | 0.4 | 1.0 |

As tabelas a seguir descrevem algumas histórias de usuário com mais detalhes de acordo com as especificações do cliente.

Tabela 3 -Descrição de HU Gerir usuário

|  |  |
| --- | --- |
| **HU Gerir usuário** | |
| **Número:** 1 | **Nome da HU:** Gerir usuário |
| **Desenvolvedor:** Eusébio Salomão Bango | |
| **Usuário:** Administrador | **Iteração:** 1 |
| **Prioridade no negócio:** Alta | **Pontos estimados:** 0.4 |
| **Risco no desenvolvimento:** Médio | **Pontos reais:** 1.0 |
| **Descrição:** O sistema possui opções para gerenciar usuários, onde o Administrador do Sistema pode ter as seguintes informações: Nome, senha, o cargo do usuário e o nível de acesso. | |
| **Nota:** Somente os usuários com nível de acesso de administrador têm acesso a esta funcionalidade. | |

Tabela 4 - Descrição de HU Gerir configurações

|  |  |
| --- | --- |
| **HU Gerir configurações** | |
| **Número:** 3 | **Nome da HU:** Gerir configurações |
| **Desenvolvedor:** Eusébio Salomão Bango | |
| **Usuário:** Administrador | **Iteração:** 2 |
| **Prioridade no negócio:** Alta | **Pontos estimados:** 1.0 |
| **Risco no desenvolvimento:** Médio | **Pontos reais:** 3.0 |
| **Descrição:** O sistema possui usuário admin que gerência as configurações do sistema sem nenhuma restrição. | |
| **Nota:** Somente os usuários com nível de acesso de administrador têm acesso a esta funcionalidade. | |

Tabela 5 - Descrição de HU Gerir informações de escola

|  |  |  |
| --- | --- | --- |
| **HU Gerir informações da escola** | | |
| **Número:** 9 | **Nome da HU:** Gerir informações da escola | |
| **Desenvolvedor:** Eusébio Salomão Bango | | |
| **Usuário:** Administrador | | **Iteração:** 4 |
| **Prioridade no negócio:** Média | | **Pontos estimados:** 0.8 |
| **Risco no desenvolvimento:** Médio | | **Pontos reais:** 2.0 |
| **Descrição:** O sistema possui um espaço onde são visualizadas as informações de consumo público. | | |
| **Nota:** Somente os usuários com nível de acesso de secretário podem gerir as informações. Os visualizadores podem reagir. | | |

Tabela 6 - Descrição de HU Fazer inscrição de novos

|  |  |  |
| --- | --- | --- |
| **HU Fazer inscrição de novos** | | |
| **Número:** 2 | **Nome da HU:** Fazer inscrição de novos | |
| **Desenvolvedor:** Eusébio Salomão Bango | | |
| **Usuário:** Ação realizada por qualquer visitante na pagina durante a faze de inscrição | | **Iteração:** 2 |
| **Prioridade no negócio:** Alta | | **Pontos estimados:** 0.6 |
| **Risco no desenvolvimento:** Médio | | **Pontos reais:** 0.8 |
| **Descrição:** O sistema possui um espaço onde qualquer um que encontrar a pagina pode realizar a sua inscrição, no tempo determinado das inscrições. | | |
| **Nota:** A gerência de alunos inscritos é realizada pelo usuário com nível de acesso definido. | | |

Tabela 7 - Descrição de HU Fazer lançamento de notas

|  |  |  |
| --- | --- | --- |
| **HU: Fazer lançamento de notas** | | |
| **Número:** 5 | **Nome da HU:** Fazer lançamento de notas | |
| **Desenvolvedor:** Eusébio Salomão Bango | | |
| **Usuário:** Acção realizada pelo usuário (Funcionário da instituição indicado para efeito de lançamento de notas) do sistema. | | **Iteração:** 3 |
| **Prioridade no negócio:** Alta | | **Pontos estimados:** 1.0 |
| **Risco no desenvolvimento:** Médio | | **Pontos reais:** 2.0 |
| **Descrição:** O sistema permite que se faça inserção das notas das avaliação dos alunos, de acordo a disciplina, e calcula as médias de acordo a especificação (MAC, MT e MF); | | |
| **Nota:** Esta acção está restringida de acordo ao nível de acesso, classe e disciplina. | | |

### 2.1.2. Duração das iterações das HU

Tabela 8 - Lista e duração das iterações

|  |  |
| --- | --- |
| **Iteração** | **Tempo de duração** |
| 1 | 1 Semana |
| 2 | 10 Semanas |
| 3 | 4 Semanas e 3 dias |
| 4 | 2 Semanas e 4 dias |
| 5 | 1 Semana e 2 dias |
| Tempo total de duração do projecto | 4 Meses e dois dias |

## 2.2. Arquitectura de Software

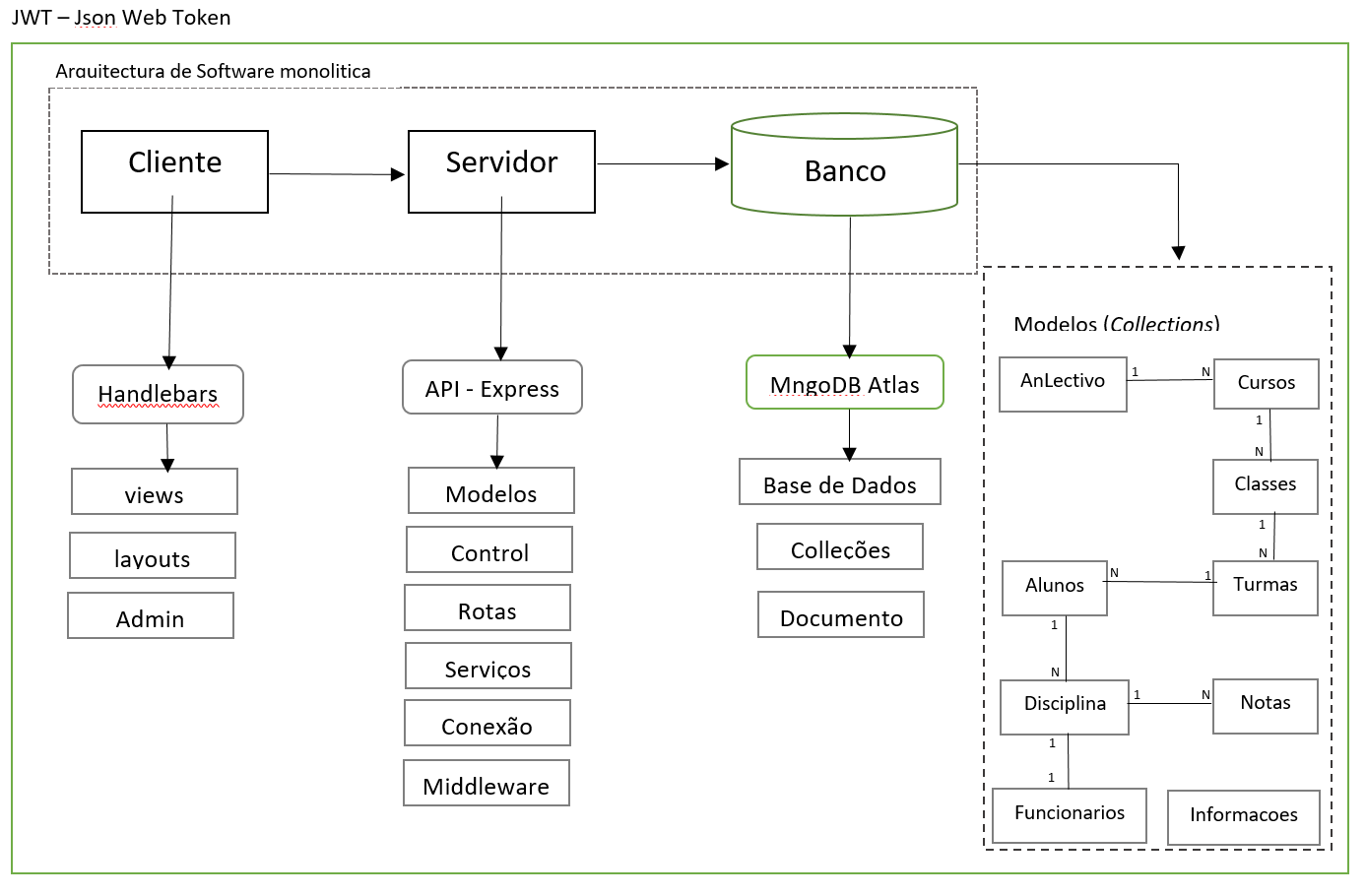


Figura 17 - Arquitectura de Software

## 2.3. Tarefas de engenharia

As tarefas de engenharia são partes constituintes das HU, ou seja, para concluir a execução de uma HU são executadas diversas actividades, dependendo da complexidade da história, com métodos e técnicas adequadas permitindo assim o escalonamento do projecto. Estas actividades executadas parcialmente são chamadas de tarefas de engenharia.

A tabela a seguir enumera as tarefas de engenharia a serem executadas, baseando-se na HU, com suas estimações e pontos reais.

Tabela 9 - Tarefas de engenharia

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **História** | **N/O** | **Tarefa** | **Pontos estimados** | **Data-Inicio** | **Data-Final** |
| 1 | 1.1 | Criar usuário | 0.2 | 08/02/2023 | 09/02/2023 |
| 1.2 | Autenticar usuário | 0.2 | 09/02/2023 | 10/02/2023 |
| 1.3 | Encerrar sessão | 0.2 | 10/02/2023 | 10/02/2023 |
| 1.4 | Listar usuários | 0.2 | 14/02/2023 | 15/02/2023 |
| 1.5 | Actualizar usuário | 0.2 | 15/02/2023 | 16/02/2023 |
| 1.6 | Excluir usuário | 0.4 | 16/02/2023 | 17/02/2023 |
| 2 | 2.1 | Inscrever candidato | 0.6 | 07/03/2023 | 11/03/2023 |
| 2.2 | Consultar candidatura | 0.4 | 12/03/2023 | 14/03/2023 |
| 2.3 | Actualizar candidatura | 0.6 | 15/03/2023 | 18/03/2023 |
| 2.4 | Deletar inscrição | 0.8 | 18/03/2023 | 21/03/2023 |
| 2.5 | Admitir candidato | 0.2 | 21/03/2023 | 23/03/2023 |
| 3 | 3.1 | Configurações | 1.0 | 09/05/2023 | 30/05/2023 |
| 4 | 4.1 | Fazer matrícula | 0.4 | 22/03/2023 | 24/03/2023 |
| 4.2 | Confirmar matrícula | 0.4 | 25/03/2023 | 30/03/2023 |
| 5 | 5.1 | Lançar notas | 1.0 | 03/04/2023 | 15/04/2023 |
| 6 | 6.1 | Editar notas | 0.6 | 17/04/2023 | 19/04/2023 |
| 7 | 7.1 | Visualizar notas, para alunos | 0.8 | 19/04/2023 | 26/04/2023 |
| 8 | 8.1 | Publicar informação | 0.4 | 26/04/2023 | 02/05/2023 |
| 8.2 | Editar informação | 0.4 | 03/05/2023 | 08/05/2023 |
| 9 | 9.1 | Relatório de candidato | 0.0 | 18/02/2023 | 19/02/2023 |
| 9.2 | Relatório de alunos matriculados | 0.2 | 20/02/2023 | 20/02/2023 |
| 10 | 10.1 | Gerar Declaração de estudante | 0.1 | 21/02/2023 | 21/02/2023 |
| 10.2 | Gerar Certificado de Habilitações | 0.2 | 21/02/2023 | 22/02/2023 |
| 10.3 | Gerar Declaração de serviço | 0.1 | 22/02/2023 | 23/02/2023 |
| 11 | 11.1 | Realizar cópia de segurança | 0.4 | xxxx | xxxx |
| 11.2 | Restaurar as informações |  | xxxx | xxxx |
| 12 | 12.1 | Notificar alunos ou encarregados de educação | 0.2 | 24/02/2023 | 26/02/2023 |
| 13 | 13.1 | Realizar consultas de dados | 0.6 | 05/06/2023 | 10/06/2023 |

## 2.3.1. Descrevendo as tarefas de engenharia

Tabela 10 - Tarefa de engenharia Criar Usuário

|  |  |
| --- | --- |
| **Tarefa de engenharia** | |
| **Nº da Tarefa:** 1.1 | **HU:** Gerir usuário |
| **Nome da tarefa:** Criar usuário | |
| **Tipo de Tarefa:** Desenvolvimento | **Pontos estimados:** 0.2 |
| **Data inicial:** 08/02/2023 | **Data final:** 09/02/2023 |
| **Programadores:** Eusébio Salomão Bango | |
| **Descrição**: para criar um usuário, o administrador deve preencher um formulário conforme a imagem no capitulo a seguir e fazer click no botão de cadastrar. Ao criar um usuário com os dados cadastrados por meio do objeto para usuário, faz-se a requisição da rota que executa o método add() que recebe como parâmetro “requisição” e “resposta”; este por sua vez faz a chamada do método save() no banco de dados. | |

Tabela 11 - Tarefa de engenharia inscrever candidato

|  |  |
| --- | --- |
| **Tarefa de engenharia** | |
| **Nº da Tarefa: 2**.1 | **HU:** Fazer inscrição de novos alunos |
| **Nome da tarefa:** Inscrever candidato | |
| **Tipo de Tarefa:** Desenvolvimento | **Pontos estimados:** 0.6 |
| **Data inicial:** 07/03/2023 | **Data final:** 11/03/2023 |
| **Programadores:** Eusébio Salomão Bango | |
| **Descrição**: para se inscrever, o candidato ou funcionário da instituição deve preencher um formulário conforme a imagem no capitulo a seguir e fazer click no botão continuar, em seguida deve confirmar os dados inseridos e clicar no botão Confirmar. Ao confirmar o sistema faz a requisição da rota que executa o método saveCandidato() que recebe como parâmetro “requisição” e “resposta”; este por sua vez faz a chamada do método save() no banco de dados. | |

Tabela 12 - Tarefa de engenharia consultar candidatura

|  |  |
| --- | --- |
| **Tarefa de engenharia** | |
| **Nº da Tarefa:** 2.3 | **HU:** Fazer inscrição de novos alunos |
| **Nome da tarefa:** Consultar candidatura | |
| **Tipo de Tarefa:** Desenvolvimento | **Pontos estimados:** 0.4 |
| **Data inicial:** 12/03/2023 | **Data final:** 14/03/2023 |
| **Programadores:** Eusébio Salomão Bango | |
| **Descrição**: Para consultar o estado da candidatura de um escrito o funcionário ou proprietário deve informar o número do Bilhete de identidade e fazer um click no botão Consultar, em seguida o sistema apresentará os dados do candidato e o estado da candidatura. Ao consultar a candidatura o sistema faz a requisição da rota que executa o método consultCand() que recebe como parâmetro “requisição” e “resposta”; este por sua vez faz uma busca no banco de dados de acordo a informação inserida (BI). | |

Tabela 13 - Tarefa de engenharia Lançar Notas

|  |  |
| --- | --- |
| **Tarefa de engenharia** | |
| **Nº da Tarefa:** 5.1 | **HU:** Fazer lançamento de notas |
| **Nome da tarefa:** Lançar notas | |
| **Tipo de Tarefa:** Desenvolvimento | **Pontos estimados:** 1.0 |
| **Data inicial:** 03/04/2023 | **Data final:** 15/04/2023 |
| **Programadores:** Eusébio Salomão Bango | |
| **Descrição**: para lançar nota dos alunos o funcionário deverá fazer login com seus dados de acesso e selecionar a turma do aluno e posteriormente inserir a nota na disciplina correspondente fazer o click no botão confirmar. | |

Tabela 14 - Tarefa de engenharia visualizar notas para alunos

|  |  |
| --- | --- |
| **Tarefa de engenharia** | |
| **Nº da Tarefa:** 7.1 | **HU:** Visualizar notas para alunos |
| **Nome da tarefa:** Visualizar notas para alunos | |
| **Tipo de Tarefa:** Desenvolvimento | **Pontos estimados:** 0.8 |
| **Data inicial:** 19/04/2023 | **Data final:** 26/04/2023 |
| **Programadores:** Eusébio Salomão Bango | |
| **Descrição**: Para visualizar a nota, o aluno deve aceder ao sistema com os seus dados de login e fazer um clik em notas. | |

## 2.4. Conclusões parciais do capitulo II

Ao definir as histórias de usuários do sistema, percebe-se as características e funcionalidades que este deve apresentar. As histórias de usuário descrevem com mais detalhes uma determinada acção funcional do sistema.

As HU devem ser particionadas em tarefas de engenharias que permitam delegar responsabilidades e garantir colaborações no desenvolvimento do sistema para que as funcionalidades adquiridas em cada história sejam mais específicas. As tarefas de engenharia ajudam a estimar o custo e o tempo de desenvolvimento do sistema.

O uso da metodologia XP permitiu ter uma compreensão mais clara de uma arquitectura para a construção das fases de implementação do sistema desenvolvido.

# CAPÍTULO III: PROVAS DE ACEITAÇÃO DO SISTEMA WEB PARA GESTÃO ESCOLAR DO LICEU FELIZBERTO D. NGUNGA DE CATABOLA

Necessário foi apresentar as provas de aceitação do sistema proposto para medir o grau de satisfação da entidade em questão; essas provas incluem um conjunto de acções/actividades como; apresentar as interfaces do sistema, perceber como cada interface foi implementado assim como cada usuário vai poder interagir com o mesmo; o tratamento de erros ou vulgo excepções. Todos esses elementos também têm por finalidade medir a segurança do sistema.

## Princípios de desenho de interface

Os autores Queiroz e Peixoto (2011) afirmam que a utilização de métodos, técnicas e ferramentas para a geração de interface torna-se relevante pois boa parte se não mesmo mais de 70% no que concerne interação Homem-Computador, depende dela, uma vez que é o meio de comunicação entre os dois (Homem-Computador). Dentro do processo de modelagem de interfaces de usuário existem abordagens para a definição do modelo conceitual, propondo um conjunto de etapas na criação de componentes de software a partir de elementos abstratos.

O processo de implementação de uma interface de usuário (IU) é uma etapa que mais consome tempo na fase criação no desenvolvimento de um sistema. Pode ser visto, por exemplo, que nos sistemas de software atuais a fração de código dedicada à interface com o usuário vem aumentando, correspondendo em média a aproximadamente 50% do software (Oliveira Neto, 2006 apud QUEIROZ & PEIXOTO, 2011). A interface com o usuário é uma parte fundamental de um software, podendo ser considerado o elemento mais importante de um sistema ou produto baseado em computador. (PRESSMAN, 2006)

Nesse contexto metodologias e ferramentas foram utilizadas para que se deixasse todas interfaces amigáveis aos usuários do sistema.



Figura 18 - Página principal do sistema proposto

## Tratamento de erros

O tratamento de erro ajuda o usuário a não comer erros na hora de executar as tarefas; aqui as excepções entram em acção, pois um dos maiores erros e que procurou-se prevenir no momento da implementação do sistema é na hora de fazer inserções de dados no sistema. Por exemplo, se o campo a preencher pede um dado do tipo numérico e o usuário por um e outro descuido insere dado do tipo texto, alguma coisa deverá ocorrer dentro do banco de dados, ou seja, nestes casos o sistema reage disparando uma notificação de que a tarefa não tenha sido finalizada com sucesso por uma falha ao informar os dados.

O ideal é evitar que os erros ocorram, mas, se isto não for possível (e nem sempre será), as mensagens apresentadas devem auxiliar o usuário e não confundi-lo ou frustrá-lo por não conseguir identificar o mesmo e ultrapassar determinada situação.

## Segurança do sistema

As aplicações de software se tornaram o alvo preferencial de ataques de criminosos, por isso é preciso garantir que as aplicações sejam projetadas, desenvolvidas e implantadas com um nível de segurança aceitável.

A gestão da segurança da informação é um assunto sério, é estratégico e deve estar entre as preocupações centrais da alta direção da instituição. Regulamentos, normas de conduta e diversos mecanismos de defesa têm sido desenvolvidos nos últimos anos, num cenário em que, cada vez mais, as empresas dependem de seus sistemas de informação e do comportamento da sua força de trabalho, para assegurar que seus dados estejam protegidos. (Filho, et al., 2012)

Pela relevância que se impõe o assunto e não ficando de fora o software proposto, levou-se em consideração todos os aspectos para garantir a protecção e a segurança dos dados; no que tange a segurança o software possui:

* Segurança e protecção contra o acesso não autorizado;
* Protecção e segurança contra a modificação de informações;
* Diferentes níveis de acessos.

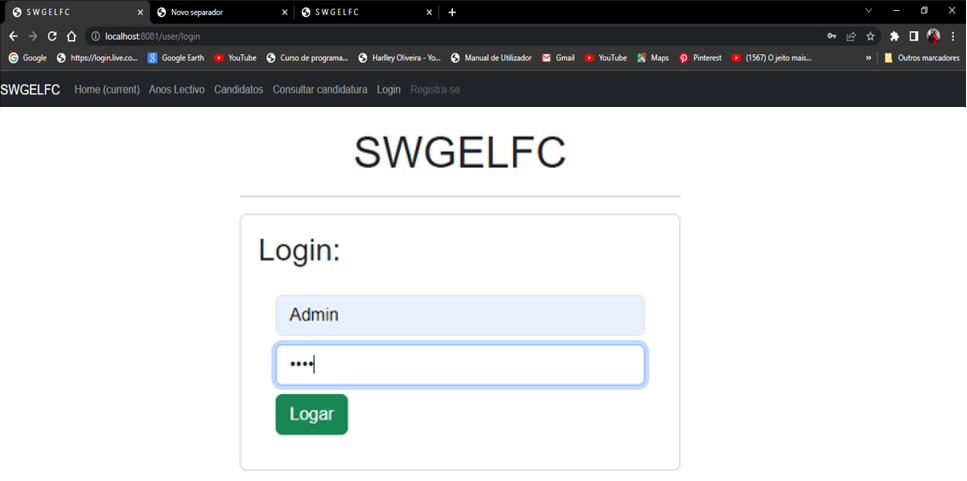


Figura 19 - Página login

## Testes de software

A Engenharia de Software tem como objetivo primário o aprimoramento da qualidade dos produtos de software e o aumento da produtividade dos engenheiros de software, além do atendimento aos requisitos de eficácia e eficiência, ou seja, efetividade (Maffeo, 1992)

## Especificações dos testes correspondentes a metodologia XP

De acordo com Sommerville (2007 apud Assis, 2012) o teste de aceitação é o estágio final do processo de teste, antes que o sistema seja aceito para uso operacional, e o sistema é testado com dados fornecidos pelo cliente, em vez de dados simulados de testes.

São apresentadas assim algumas tabelas de testes de aceitação ao software proposto tido de acordo a metodologia xp.

Tabela 15 - Caso de teste cadastrar usuário funcionário

|  |  |
| --- | --- |
| **Teste de Aceitação** | |
| **Código:** 1 | **História de usuário:** Gerir Usuário |
| **Nome:** Cadastrar Usuário | |
| **Descrição:** o sistema permite o cadastro de novos usuários e neste caso o funcionário que poderá acessar diversas funcionalidades; essa função é especificamente para o administrador. | |
| **Condição de execução:** os dados solicitados deverão todos ser preenchidos correctamente. | |
| **Entrada / saída de execução:**   1. Acessar o site como administrador; 2. Escolher a função cadastrar; 3. Dados preenchidos, finalizar clicando em “Cadastrar”. | |
| **Avaliação do teste:** satisfatória | |

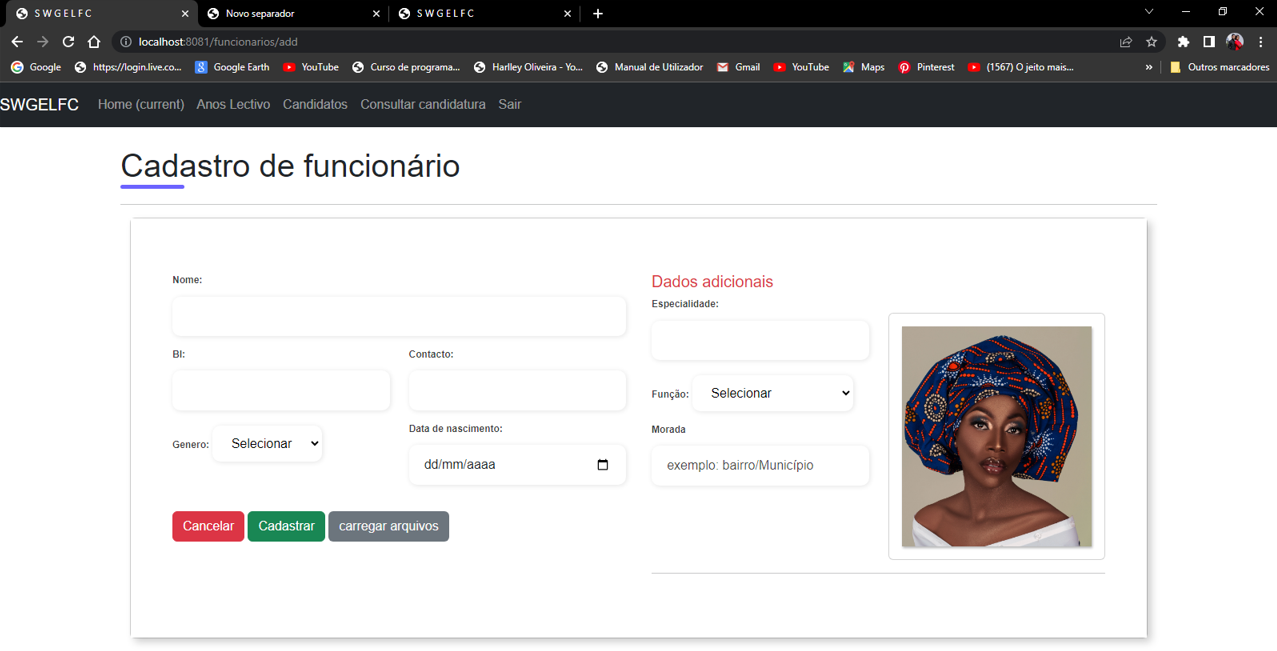


Figura 20 - Página cadastrar funcionário

Tabela 16 - Caso de teste consultar usuário candidato

|  |  |
| --- | --- |
| **Teste de Aceitação** | |
| **Código:** 2 | **História de usuário:** Gerir Candidato |
| **Nome:** Cadastrar Usuário | |
| **Descrição:** o sistema permite efectuar buscas a todos inscritos; essas buscas podem ser específicas (por nome, número do bilhete e ainda permite fazer mais alguns filtros) ou geral conforme mostra a figura a seguir. | |
| **Condição de execução:** em caso de quer visualizar os dados de cada candidato, clicar em detalhes. | |
| **Entrada / saída de execução:**   1. Acessar a página principal/site; 2. Clicar no menu consultar candidatura; | |
| **Avaliação do teste:** satisfatória | |

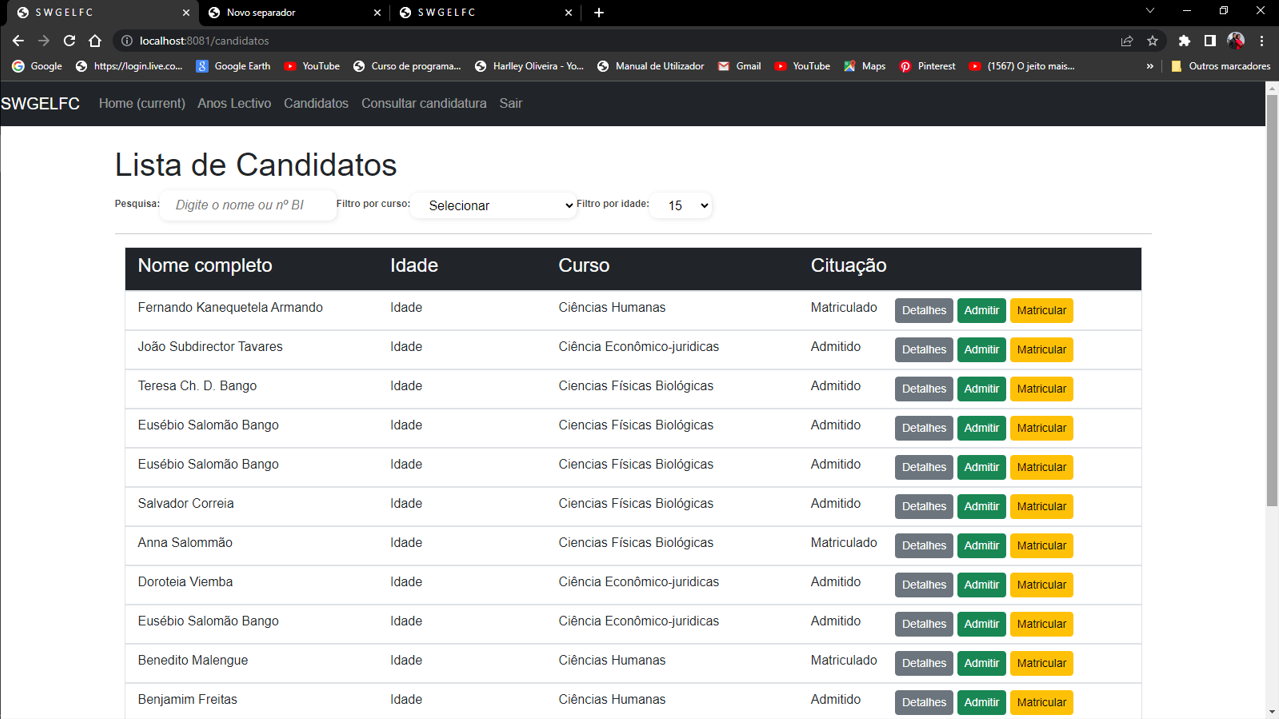


Figura 21 - Página de consulta de candidato

Tabela 17 - Caso de teste lista de anos lectivos

|  |  |
| --- | --- |
| **Teste de Aceitação** | |
| **Código:** 4 | **História de usuário:** Gerir Matricula |
| **Nome:** lista de anos lectivos | |
| **Descrição:** o sistema permite visualizar o ano lectivo actual e também os anos lectivos já vencidos. | |
| **Condição de execução:** logar como administrador | |
| **Entrada / saída de execução:**   1. Entrar como administrador; 2. Clicar em “Anos Lectivos”. | |
| **Avaliação do teste:** satisfatória | |

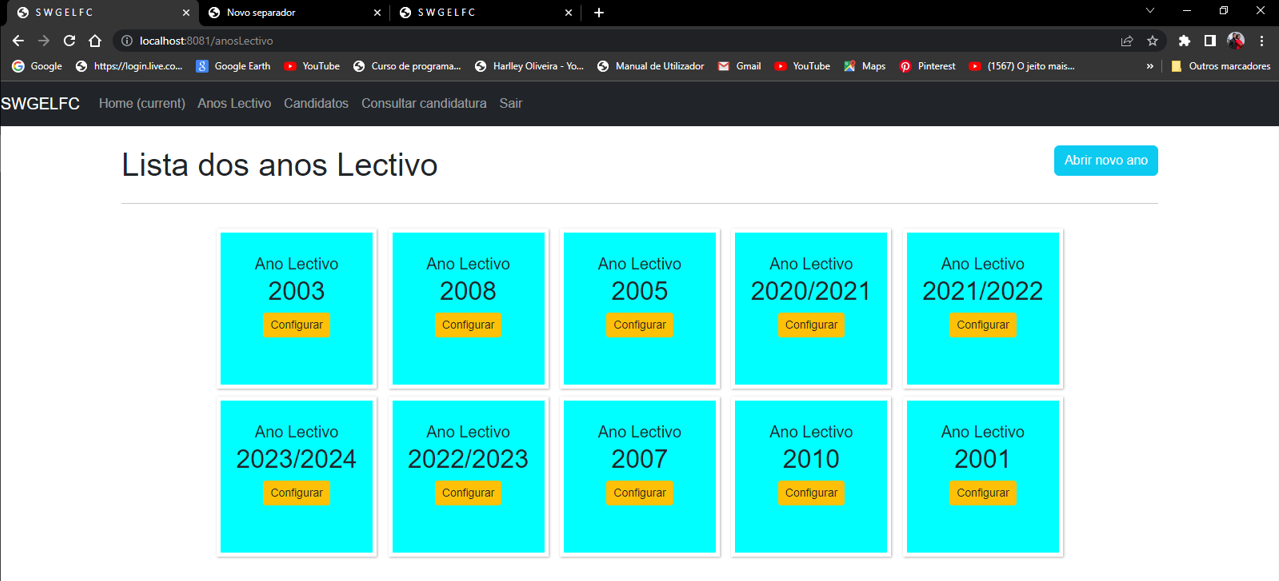


Figura 22 - Página consulta anos lectivos

Tabela 18 - Caso de teste editar candidato

|  |  |
| --- | --- |
| **Teste de Aceitação** | |
| **Código:** 2 | **História de usuário:** Gerir Candidatos |
| **Nome:** Cadastrar Usuário | |
| **Descrição:** o usuário com a devida permissão, e com muita atenção pode escolher os campos que pretenda alterar e o sistema deve responder às alterações. | |
| **Condição de execução:** o utente deverá preencher todos os dados conforme elucida o formulário. | |
| **Entrada / saída de execução:**   1. Acessar o sistemo como funcionário ou administrador; 2. Consultar candidatos; 3. Clicar em detalhes; 4. Clicar em editar dados; 5. Dados fornecidos, finalizar clicando em “Salvar alterações”. | |
| **Avaliação do teste:** satisfatória | |

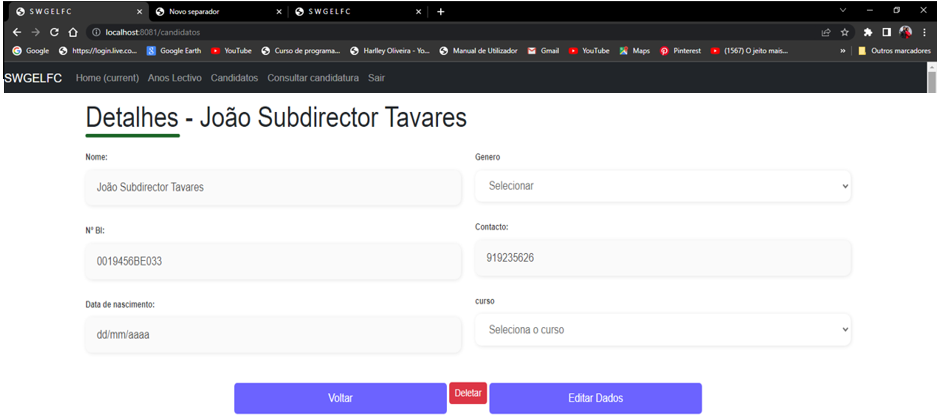


Figura 23 - Página detalhes de candidatos

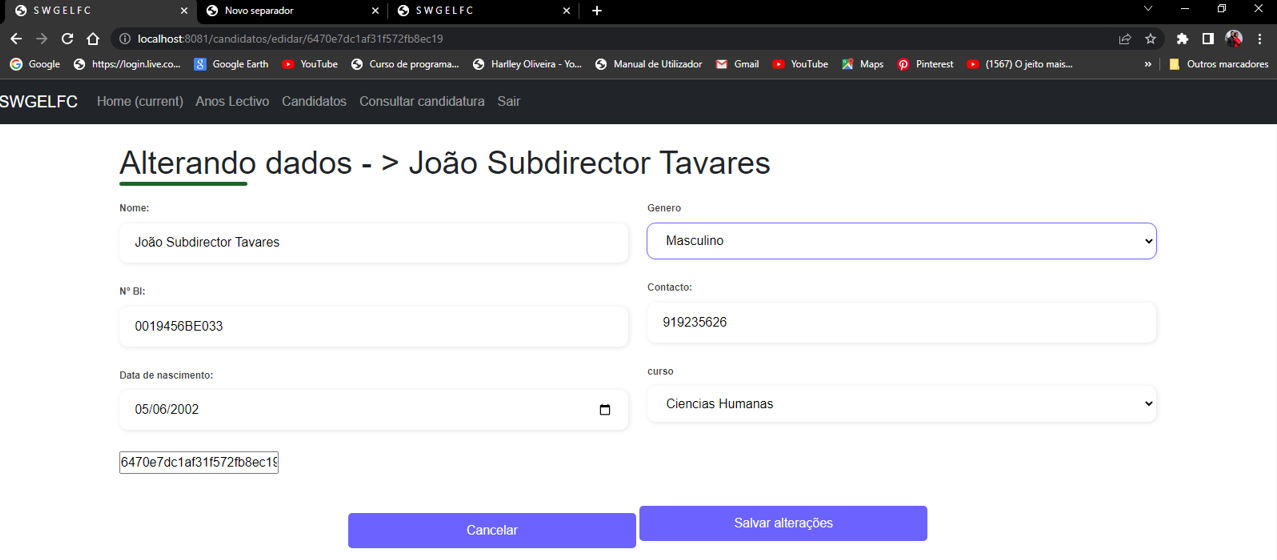


Figura 24 - Página editar candidato

Tabela 19 - Caso de teste Manter Administrador

|  |  |
| --- | --- |
| **Teste de Aceitação** | |
| **Código:** 1 | **História de usuário:** Gerir Usuário |
| **Nome:** Gerir Usuário | |
| **Descrição:** visualizar área de administrador | |
| **Condição de execução:** acessar a área de administrador | |
| **Entrada / saída de execução:**   1. Logar como administrador; | |
| **Avaliação do teste:** satisfatória | |

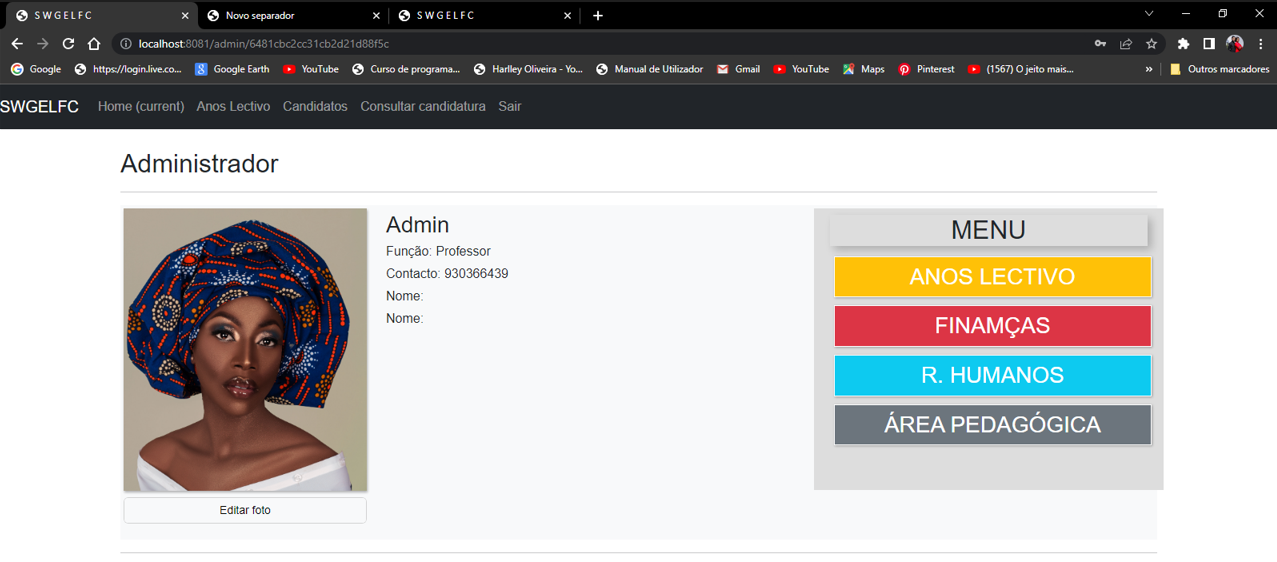


Figura 25 - Página Administrador

Tabela 20 - Caso de teste matricular aluno

|  |  |
| --- | --- |
| **Teste de Aceitação** | |
| **Código:** 4 | **História de usuário:** Gerir matrícula |
| **Nome:** Cadastrar Usuário | |
| **Descrição:** o sistema permite efectuar matricula de novos alunos. | |
| **Condição de execução:** o usuário deverá preencher todos os dados conforme elucida o formulário. | |
| **Entrada / saída de execução:**   1. Acessar a página como funcioário; 2. Consultar Candidatos; 3. Na página seguinte clicar em “Matricular”, válido apenas aos candidatos admitidos; 4. Dados fornecidos correctamente, finalizar clicando em “Matricular”. | |
| **Avaliação do teste:** satisfatória | |

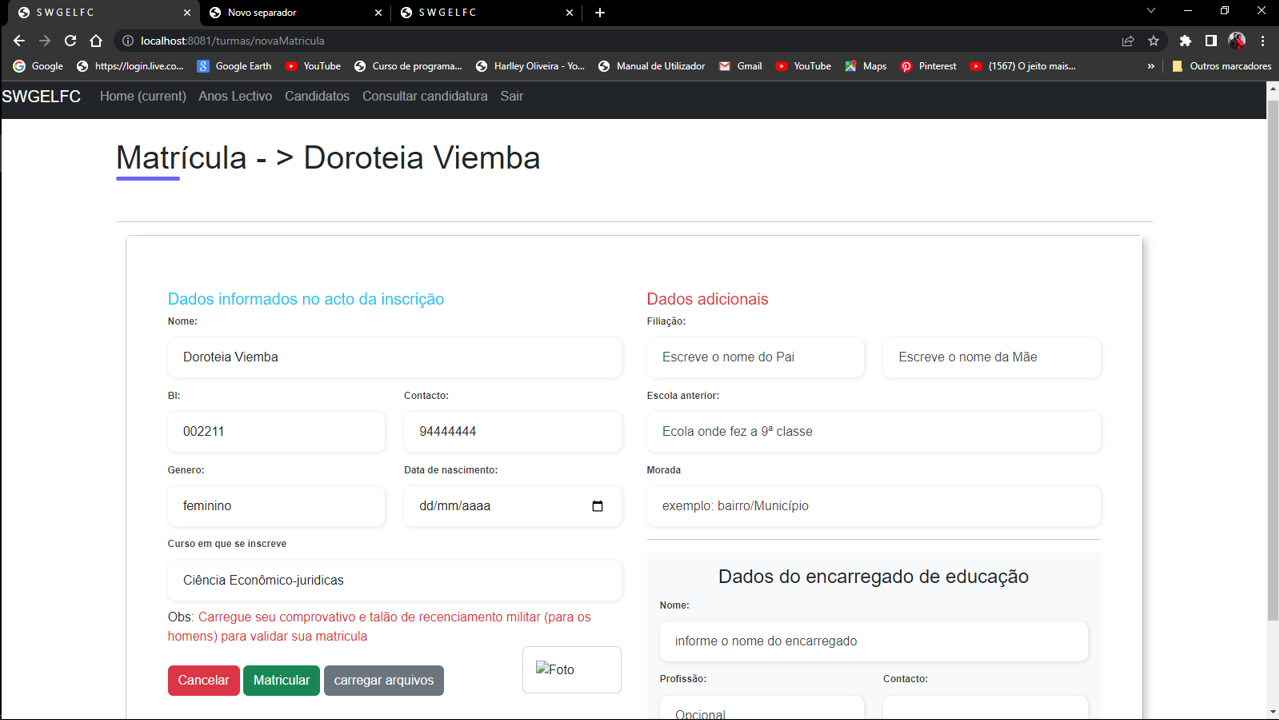


Figura 26 - Página fazer matrícula

## Conclusões parciais do capitulo III

Os testes/provas de aceitação permitiram mostrar como o Sistema Web para Gestão Escolar do Liceu Felizberto D. Ngunga de Catabola foi implementado ao nível de interface, tratamento de erros e segurança do mesmo; além disso serviu como termómetro de medição quanto a satisfação do cliente e do próprio sistema em si.

O software foi desenvolvido tendo em conta os padrões e todos os pressupostos para que não fuja do paradigma geral de um aplicativo; assim, todos os requisitos mencionados aqui neste capítulo foram implementados com sucesso.

# CONCLUSÕES GERAIS

<é so responder as esses objectivos>

* Caracterizar os antecedentes históricos e conceituais de processos de Gestão escolar e softwares de gestão escolar;
* Caracterizar a situação atual do processo de gestão escolar no Liceu Felizberto D. Ngunga de Catabola;
* Caracterizar as ferramentas informáticas utilizadas para o desenvolvimento web de gestão escolar do Liceu Felizberto D. Ngunga de Catabola;
* Analisar e desenhar o sistema web de gestão escolar para o Liceu Felizberto D. Ngunga de Catabola;
* Implementar o sistema web de gestão escolar no Liceu Felizberto D. Ngunga de Catabola;
* Validar o sistema web de gestão escolar do Liceu Felizberto D. Ngunga de Catabola por meio de teste de aceitação;

# Recomendações

<por favor coleque suas recomendações>

# Referências Bibliográficas

Assis, F. B. (2012). *DESENVOLVIMENTO DE SOFTWARE DIRIGIDO POR TESTE DE ACEITAÇÃO.* Belo Horizonte. Obtido em 02 de Junho de 2023, de http://hdl.handle.net/1843/BUOS-94LMVC

Bancovsky, R. (2008). *Formação e prática de gestão escolar com o uso das tecnologias.* SÃO PAULO. Obtido de https://tede.pucsp.br/bitstream/handle/10085/1/Renata%20Bancovsky.pdf

Boaglio, F. (2015). *MongoDB; Constua novas aplicações com novas tecnologias.* São Paulo: Casa de Código.

Caldeira, C. P. (2011). *INTRODUÇÃO AOS SISTEMAS DE GESTÃO DE INFORMAÇÃO.* Évora.

Carvalho, A. A. (2015). *Articulação de várias linguagens de programação/tecnologias no desenvolvimento aplicacional.* Porto.

Chavesa, L. C., Ensslina, L., Ensslina, S. R., Valmorbidaa, S. M., & Shinohara, K. J. (2013). *SEGURANÇA DE SOFTWARE: UMA ABORDAGEM MULTICRITÉRIO PARA AVALIAÇÃO DE DESEMPENHO.* Rio de Janeiro: Universidade Federal de Santa Catarina.

Cruz, J. M. (2019 ). *O papel das Instituições de Formação Inicial de Professores na criação e desenvolvimento de competências TIC: o caso do ISCED – Benguela. .* Lisboa.

Domingues, N. O. (2017). *As Tecnologias de Informação e Comunicação: um recurso na promoção das aprendizagens.* Gulpilhares.

Filho, A. B., Guerra, A. C., Grégio, A. A., Rosa, F. d., Teracine, E. C., Duarte, L. O., & Colombo, R. M. (Maio de 2012). Segurança de Sistemas da Informação. *Programa Brasileiro da Qualidade e Produtividade em Software - Projetos Ciclo 2011*.

Fontoura, F. C. (2019). *Uso de Metodologias de Desenvolvimento de Software e de Engenharia de Requisitos em empresas de Tecnologia: um estudo a partir de um Survey.* Natal-RN. Obtido em 28 de Janeiro de 2023, de Junhttps://www.overleaf.com/project/5cc221041b92714e94e02996ho, 2019

Gonçalves, A. R. (2012). *O Papel das TIC na Escola, na Aprendizagem e na Educação.* Lisboa.

Júnior, A. A., Schmitz, H., & Neto, A. C. (2010). *TICs nos processos da Administração e Gestão Escolar.* Sergipe.

Kaufmann, M. (2005). *Implementação de um Sistema unificado de gestão escolar e e-learning.* Porto.

Lemos, G. B. (2015). *Otimização de Processos Organizacionais.* Rio de Janeiro.

Lins, G. d. (2019). *UTILIZANDO REACTJS PARA O DESENVOLVIMENTO DE UM SISTEMA DE ALOCAÇÃO E RESERVA DE SALAS NO CAMPUS DA UFC EM QUIXADÁ.* QUIXADÁ: CDD 005.

Maffeo, B. (1992). *Engenharia de Software e Especificação de Sistemas.* Rio de Janeiro.

Mariano, D. (2022). *BOOTSTRAP 5 – GUIA RÁPIDO PARA INICIANTES.* Lagoa Santa,: MG.

Martins, P. L., Melo, B. M., Queiroz, D. L., Souza, M. S., & Borges, R. d. (2012). *Tecnologia e Sistemas de Informação e Suas Influencias na Gestão e Contabilidade.*

Neto, R. M. (2016). *Engenharia de Software.* Londrina: r Editora e Distribuidora Educacional S.A. Obtido em 28 de Janeiro de 2023, de http://cm-kls-content.s3.amazonaws.com/201601/INTERATIVAS\_2\_0/ENGENHARIA\_DE\_SOFTWARE/U1/LIVRO\_UNICO.pdf

Panno, F., Senger, I., Franciscato, R., & Silveira, S. R. (2020). *Sistema de Informação para Gestão Educacional.* Santa Maria: UFSM, NTE.

PRESSMAN, R. S. (2006). *Engenharia de Software.* São Paulo: McGrawHill.

Pressman, R. S. (2011). *Engenharia de Software: Uma Abordagem Profi ssional.* Porto Alegre: 7.

Pressman, R. S. (2016). *Engenharia de Sooftware uma Abordagem profissional.* São Paulo: AMGH EDITORA LTDA.

QUEIROZ, M. A., & PEIXOTO, C. S. (08 de 11 de 2011). GERAÇÃO AUTOMATIZADA DE INTERFACE DE USUÁRIO BASEADO NA MÉTODOLOGIA. *Ambiente e sustentabilidade*.

Quesada, A. (2001). *Apostila de MySQL.* SÃO PAULO.

Resende, F. M. (2012). *A Gestão da informação na dmstração das escolas Pública.* Lisboa.

SANTI, R. d., & BEGOSSO, L. R. (2015). METODOLOGIAS ÁGEIS PARA O DESENVOLVIMENTO DE SOFTWARE. Em R. d. SANTI, & L. R. BEGOSSO, *METODOLOGIAS ÁGEIS PARA O DESENVOLVIMENTO DE SOFTWARE.*

Silva, I. (9 de Abril de 2023). *Noticias de Angola*. Obtido de Noticia de Angola: https://noticiasdeangola.co.ao/huawei-e-uniao-africana-intensificam-transformacao-digital-em-africa/

Silva, J. G., Lima, S. F., & Souza, V. M. (2019). A GESTÃO ESCOLAR E A INSERÇÃO DAS TECNOLOGIAS DIGITAIS DE INFORMAÇÃO E COMUNICAÇÃO NA PRÁTICA PEDAGÓGICA. *Revista mais Educação*. Obtido de https://www.revistamaiseducacao.com/ArtigosV2N6AGO2019/05

Sommerville, I. (2011). *Engenharia de Software.* São Paulo: Person Education do Brasil. Obtido em 28 de Janeiro de 2023, de https://www.facom.ufu.br/~william/Disciplinas%202018-2/BSI-GSI030-EngenhariaSoftware/Livro/engenhariaSoftwareSommerville.pdf

SOUSA, A. J., LOPEZ, A. P., & ANDRADE, S. C.-R. (2007). *Metodologia, método e técnica de investigação científica em Ciência da Informação.* Brasília. Obtido em 28 de Janeiro de 2023, de https://brapci.inf.br/index.php/res/v/181714

Sousa, J., Souza, G. B., Aujo, M. V., & Souza, A. d. (2018). *SISTE DE GESTÃO ESCOLAR.* Roraima.

Veiga, F. J. (16 de Outubro de 2006). Evolução dos Sistemas de Informação. *Gestão de Sistemas de Informação*, p. 18.

Veiga, P. (2004). *TECNOLOGIAS E SISTEMAS DE INFORMAÇÃO, REDES E SEGURANÇA.* Porto: SPI – Sociedade Portuguesa de Inovação.

# Anexos