



INSTITUTO SUPERIOR POLITÉCNICO DE CIÊNCIAS E TECNOLOGIA
CURSO DE ENGENHARIA INFORMÁTICA E SISTEMAS DE INFORMAÇÃO

**Desenvolvimento de uma Plataforma *Web* que Conecta Estudantes Universitários a
Empresas e Oportunidades de Estágio no Contexto Angolano, no Ano de 2026**

Autora: Edvânia António Francisco

Orientador: Pedro Nascimento

Luanda – 2026

DECLARAÇÃO DE APROVAÇÃO DO ORIENTADOR

Eu, Pedro Nascimento na qualidade de orientador, certifico que o presente trabalho atende os requisitos académicos, científicos e metodológicos exigidos pela instituição, reflectindo um esforço genuíno da estudante na investigação e na produção do conhecimento.

Assim, considero o trabalho apto para a sua apresentação e avaliação pela banca examinadora.

Luanda, aos _____ de _____ de 20 _____

Assinatura do Orientador



INSTITUTO SUPERIOR POLITÉCNICO DE CIÊNCIAS E TECNOLOGIA
CURSO DE ENGENHARIA INFORMÁTICA E SISTEMAS DE INFORMAÇÃO

**Desenvolvimento de uma Plataforma *Web* que Conecta Estudantes Universitários a
Empresas e Oportunidades de Estágio no Contexto Angolano, no Ano de 2026**

Autora: Edvânia António Francisco

Orientador: MsC. Pedro Nascimento

Luanda – 2026

TERMO DE APROVAÇÃO

Eu, Edvânia António Francisco, certifico que este trabalho é original_____

Tema: Desenvolvimento de uma Plataforma *Web* que Conecta Estudantes Universitários a Empresas e Oportunidades de Estágio no Contexto Angolano, no Ano de 2026

“Trabalho de Conclusão de Curso apresentado ao Instituto Superior Politécnico de Ciências e Tecnologia como parte dos requisitos para a obtenção do grau académico de Licenciatura em Engenharia Informática e Sistemas de Informação.”

Esta Monografia foi avaliada e aprovada para a obtenção do título de licenciada no curso de Engenharia Informática e Sistemas de Informação.

Mestre prof. Pedro Nascimento
Orientador

Mestre prof. Rouget Ruano
Presidente do Júri

Mestre prof. Irene Monier
Arguente

Licenciado prof. Álvaro Jorge
Arguente

Luanda – 2026

DEDICATÓRIAS

Dedico este trabalho, primeiramente, a Deus, pela força, sabedoria e protecção concedidas em cada etapa desta jornada.

Aos meus pais e familiares que sempre acreditaram no meu potencial e não mediram esforços para apoiar os meus sonhos.

Dedico ainda a todos aqueles que, directa ou indirectamente, fazem parte da minha história e que me inspiraram a nunca desistir.

AGRADECIMENTOS

A realização deste Trabalho de Conclusão de Curso não seria possível sem a contribuição e apoio de várias pessoas e instituições.

Agradeço a Deus pela vida, saúde e pela oportunidade de chegar até aqui.

Aos meus orientadores e docentes, pela dedicação, paciência e conhecimentos transmitidos ao longo da formação.

Aos colegas de curso, pelo companheirismo e partilha de experiências que tornaram esta caminhada mais leve.

À minha família, pela compreensão, incentivo constante e amor incondicional.

À instituição de ensino, pelo acolhimento e pela oportunidade de desenvolver este trabalho que representa a materialização de anos de estudo e esforço.

A todos que, de alguma forma, contribuíram para a concretização deste sonho, o meu muito obrigado.

RESUMO

O presente trabalho tem como objectivo desenvolver uma plataforma *web* destinada a conectar estudantes universitários a empresas e oportunidades de estágio no contexto angolano, como resposta às dificuldades enfrentadas por eles no acesso a experiências profissionais e pelas empresas no recrutamento estruturado de estagiários. A problemática centra-se na inexistência (do ponto de vista prático) de um mecanismo digital integrado que promova a aproximação efectiva entre o meio académico e o mercado de trabalho. Especificamente, pretende-se analisar o cenário actual das limitações no acesso às oportunidades de estágio, levantar os requisitos funcionais e não funcionais necessários ao desenvolvimento da plataforma, projectar a sua arquitectura e *interface* considerando critérios de usabilidade, acessibilidade e adequação ao contexto angolano, bem como desenvolver uma solução tecnológica que permita o registo de estudantes, a divulgação de oportunidades por parte das empresas e a interacção entre ambos. Metodologicamente, a pesquisa caracteriza-se como aplicada, por propor uma solução prática para um problema real; descritiva, por identificar e organizar as necessidades do processo; e qualitativa, por privilegiar a compreensão interpretativa das dificuldades enfrentadas pelos actores envolvidos. Foram utilizados os métodos dedutivo-indutivo, análise-síntese e hipotético-dedutivo, aliados à análise investigativa de natureza conceptual e experiencial, bem como à modelagem do sistema por meio de diagramas UML e protótipos visuais. A relevância do estudo reside no seu contributo para a promoção da empregabilidade juvenil e para o fortalecimento da ligação entre as instituições de ensino superior e o sector empresarial em Angola.

Palavras-chave: Angola; Empregabilidade; Estágio; Estudantes; Plataforma *web*.

SUMÁRIO

INTRODUÇÃO	1
CAPÍTULO I – FUNDAMENTAÇÃO TEÓRICA	7
1.1 Educação superior e o mercado de trabalho em Angola	7
1.1.1 Contextualização da educação superior em Angola	7
1.1.2 Desafios da inserção profissional dos estudantes universitários.....	7
1.1.3 A importância dos estágios no desenvolvimento de competências	8
1.2 Estágios académicos e profissionais.....	8
1.2.1 Conceito e tipologia de estágios	8
1.2.2 O papel dos estágios na transição universidade – mercado de trabalho	9
1.2.3 Oportunidades e desafios de estágios em Angola.....	9
1.3 As plataformas <i>web</i> como mediadoras de oportunidades.....	10
1.3.1 Conceito e características das plataformas <i>web</i>	10
1.3.2 Experiências e modelos internacionais de conexão entre estudantes e empresas....	11
1.3.3 Potencial das plataformas <i>web</i> no contexto angolano.....	11
1.4 Tecnologias <i>web</i>	12
1.4.1 Conceito e evolução das tecnologias <i>web</i>	12
1.4.2 Linguagens de programação e <i>frameworks</i> utilizadas em desenvolvimento <i>web</i>	13
1.4.3 Segurança e escalabilidade em plataformas digitais	13
1.5 Arquitectura de sistemas <i>web</i>	14
1.5.1 Conceito e importância da arquitectura de sistemas	14
1.5.2 Modelos de arquitectura aplicáveis a plataformas <i>web</i>	14
1.5.3 Integração de bases de dados em sistemas <i>web</i>	15
1.6 Padrão MVC	16
1.6.1 Conceito do padrão MVC	16
1.6.3 Vantagens e aplicabilidade do MVC em projectos <i>web</i>	17
1.7 Ferramentas utilizadas para o desenvolvimento da plataforma <i>web</i>	18

1.7.1 PHP	18
1.7.2 <i>JavaScript</i>	19
1.7.3 <i>Bootstrap</i>	20
1.7.4 <i>MySQL workbench</i>	21
1.7.5 Visual <i>studio</i> code	22
1.7.6 XAMPP	23
1.7.7 <i>StarUML</i>	24
CAPÍTULO II – APRESENTAÇÃO DA PLATAFORMA <i>WEB</i>	25
2.1 Desenvolvimento da plataforma	25
2.2 Metodologia utilizada para o desenvolvimento da plataforma <i>web</i>	25
2.2.1 Características, vantagens e desvantagens da metodologia	26
2.2.2 Benefícios	26
2.3 Requisitos funcionais (RF)	26
2.4 Requisitos não funcionais (RNF)	30
2.5 Casos de uso	33
2.5.1 Descrição dos casos de uso	35
2.6 Diagrama de caso de uso	40
2.7 Diagrama de classe	42
2.8 Diagrama entidade-relacionamento (ER)	43
2.9 Diagrama de actividade	44
2.10 Diagrama de sequência	45
CONCLUSÃO	46
REFERÊNCIAS BIBLIOGRÁFICAS	47
APÊNDICES	51
Apêndice n.º 1 – Guião de observação.	51
Apêndice n.º 2 – Algumas telas da plataforma <i>web</i>	54

LISTA DE FIGURAS

Figura 1. (Padrão MVC).....	16
Figura 2. (PHP)	18
Figura 3. (<i>JavaScript</i>).....	19
Figura 4. (<i>Bootstrap</i>).....	20
Figura 5. (<i>MySQL workbench</i>).....	21
Figura 6. (Visual <i>studio</i> code).....	22
Figura 7. (XAMPP).....	23
Figura 8. (<i>StarUML</i>)	24
Figura 9. (<i>Extreme programming</i> – Ciclos de desenvolvimento)	25
Figura 10. (Diagrama de caso de uso).....	41
Figura 11. (Diagrama de classe).....	42
Figura 12. (Diagrama entidade-relacionamento)	43
Figura 13. (Diagrama de actividade – Submeter candidatura).....	44
Figura 14. (Diagrama de sequência – Submeter candidatura)	45
Figura 15. (Tela de <i>login</i>).....	54
Figura 16. (<i>Home</i>)	54
Figura 17. (Painel do administrador).....	55
Figura 18. (Painel da empresa).....	55

LISTA DE TABELAS

Tabela 1. (Requisitos funcionais).....	27
Tabela 2. (Requisitos não funcionais)	30
Tabela 3. (Casos de uso)	33
Tabela 4. (Identificação da observação).....	53
Tabela 5. (Aspectos a observar)	53

LISTA DE ACRÓNIMOS E SIGLAS

AIESPA – Associação das instituições de ensino superior privadas de Angola

APA – *American psychological association* (Associação americana de psicologia)

API – *Application programming interface* (Interface de programação de aplicativos)

ANJE – Associação nacional dos jovens empresários

CSS – *Cascading style sheets* (Folhas de estilo em cascata)

ER – Entidade-relacionamento

GSOC – *Google summer of code* (Verão de código do google)

GTE – Grupo de trabalho de especialistas

HTML – *HyperText markup language* (Linguagem de marcação de hipertexto)

HTTPS – *Hypertext transfer protocol Secure* (Protocolo de transferência de hipertexto seguro)

INEFOP – Instituto nacional de emprego e formação profissional

MVC – *Model–View–Controller* (Modelo–Visualização–Controlador)

MUTW – Mulheres unidas e trabalhadoras de Angola

NGO/ONG – Organização não governamental

OIT – Organização internacional do trabalho

ORM – *Object-relational mapping* (Mapeamento objecto-relacional)

PHP – *Hypertext preprocessor* (Pré-processador de hipertexto)

PNUD – Programa das nações unidas para o desenvolvimento

PROVIP – Programa de valorização da iniciativa privada

RN – Requisitos funcionais

RNF – Requisitos não funcionais

SOA – *Service-oriented architecture* (Arquitetura orientada a serviços)

SQL – *Structured query language* (Linguagem de consulta estruturada)

UML – *Unified modeling language* (Linguagem de modelagem unificada)

UNIA – Universidade Independente de Angola

VN-BASE – Voluntários nacionais base

UC – *Use case* (Caso de uso)

UAN – Universidade Agostinho Neto

VS – Visual *studio*

VS – Versus (vs)

WEB – *World Wide Web* (Rede mundial de computadores)

XAMPP – *Cross-Platform, Apache, MariaDB/MySQL, PHP, Perl*

XP – *Extreme programming* (Programação extrema)

INTRODUÇÃO

O estágio académico constitui uma etapa fundamental no processo de formação superior, permitindo que os estudantes universitários estabeleçam uma ponte entre os conhecimentos teóricos adquiridos e a prática profissional. A nível mundial, diversas plataformas digitais têm sido criadas para aproximar estudantes às empresas, promovendo uma integração mais célere e eficaz no mercado de trabalho. Países com economias emergentes e avançadas recorrem a soluções digitais como instrumentos estratégicos de empregabilidade e inovação, dado o crescimento acelerado da transformação digital e a procura constante por profissionais qualificados.

No contexto africano, o desafio do desemprego juvenil e da inserção profissional de recém-formados constitui um dos maiores obstáculos ao desenvolvimento socioeconómico. Muitos jovens enfrentam dificuldades em acessar informações sobre oportunidades de estágio, programas de *trainee* e ofertas de emprego, o que cria um hiato entre as instituições de ensino superior e o mercado de trabalho. A ausência de mecanismos integrados e acessíveis para conectar estes dois universos acaba por limitar as possibilidades de crescimento e competitividade dos graduados africanos no panorama global.

Em Angola, a realidade não é diferente. O país regista um número crescente de estudantes universitários que concluem a sua formação sem conseguirem acesso imediato a oportunidades de estágio. Este fenómeno tem origem, em grande medida, na inexistência de plataformas eficazes e originariamente nacionais que estabeleçam a ligação entre estudantes e empresas. Muitas organizações, por sua vez, enfrentam dificuldades em recrutar estagiários de forma ágil e transparente, visto que o processo de divulgação de vagas ainda ocorre de forma limitada e pouco sistematizada.

Diante deste cenário, a criação de uma plataforma *web* de matriz nacional que facilite a interação entre estudantes universitários e empresas surge como uma solução inovadora e necessária no contexto angolano. Tal sistema poderá actuar como mediador entre oferta e procura de estágios, promovendo maior inclusão digital, transparência no processo de selecção e, sobretudo, contribuindo para a redução da distância entre a formação académica e as exigências do mercado de trabalho. Assim, este trabalho propõe o desenvolvimento de uma plataforma digital que não apenas centralize oportunidades, mas também fortaleça a ligação entre ensino superior, empresas e sociedade.

Problematização

A integração dos estudantes universitários no mercado de trabalho tem se revelado um dos grandes desafios enfrentados a nível global e, em particular, em Angola. Apesar do crescimento do ensino superior no país, observa-se uma carência significativa de mecanismos estruturados que facilitem a transição entre a vida académica e o ambiente profissional. Essa limitação gera um desfasamento entre a formação teórica e as exigências práticas do mercado, provocando frustrações entre os jovens graduados e impactando negativamente o aproveitamento das suas competências.

As empresas, por sua vez, deparam-se com dificuldades em recrutar estagiários de forma organizada, transparente e eficiente. Muitos processos de selecção ainda são realizados de maneira manual ou através de redes de contactos restritas, o que limita o acesso de estudantes a oportunidades e restringe a diversidade no recrutamento. A ausência de uma plataforma digital integrada originariamente nacional dificulta não apenas o encontro entre estudantes e empresas, mas também o acompanhamento e a avaliação das experiências de estágio que poderiam contribuir para a qualificação contínua dos futuros profissionais.

Constata-se, portanto, a necessidade de criar uma solução tecnológica adequada ao contexto angolano que promova uma interacção ágil e estruturada entre os estudantes universitários e as entidades empregadoras. Tal solução deve responder às carências actuais, reduzindo as barreiras no acesso à informação e proporcionando maior equidade no processo de selecção de estagiários. Essa problemática justifica a relevância de desenvolver uma plataforma *web* que centralize informações, simplifique o acesso a oportunidades de estágio e fortaleça a relação entre instituições de ensino superior, empresas e estudantes.

Pergunta de partida

Como o desenvolvimento de uma plataforma *web* pode facilitar a ligação entre estudantes universitários e empresas, promovendo o acesso a oportunidades de estágio no contexto angolano?

Hipótese

O desenvolvimento de uma plataforma *web*, utilizando a linguagem de programação *JavaScript* e o *framework Bootstrap*, poderá facilitar a ligação entre estudantes universitários e empresas no contexto angolano, ao centralizar e organizar oportunidades de estágio, simplificar os processos de divulgação e candidatura, bem como promover maior transparência e acessibilidade à informação.

Objectivo geral

- Desenvolver uma plataforma *Web* que conecte estudantes universitários a empresas e oportunidades de estágio no contexto angolano.

Objectivos específicos

- Analisar o cenário actual das dificuldades enfrentadas por estudantes universitários angolanos no acesso a oportunidades de estágio e pelas empresas no processo de recrutamento de estagiários.
- Levantar os requisitos funcionais e não funcionais necessários para o desenvolvimento de uma plataforma *web* que conecte estudantes e empresas.
- Projectar a arquitectura e a *interface* da plataforma, considerando usabilidade, acessibilidade e adequação ao contexto angolano.
- Desenvolver a plataforma *web* integrando funcionalidades que permitam o registo de estudantes, a divulgação de oportunidades por parte das empresas e a interacção entre ambos.

Delimitação

A presente pesquisa foi desenvolvida no período compreendido entre o ano de 2024 e 2025, tendo como foco a realidade do ensino superior em Angola, particularmente na cidade de Luanda. O desenvolvimento da plataforma *web* situa-se no contexto académico e empresarial angolano, com ênfase na conexão entre estudantes universitários e empresas que oferecem oportunidades de estágio.

Justificativa

A escolha do tema justifica-se pela crescente necessidade de integração dos estudantes universitários no mercado de trabalho angolano, num contexto em que a formação académica, por si só, não garante inserção profissional. A ausência de mecanismos estruturados e acessíveis que aproximem estudantes e empresas constitui um entrave ao desenvolvimento socioeconómico, tornando essencial a criação de soluções digitais que possam reduzir essa lacuna.

Adicionalmente, a proposta apresenta relevância por explorar uma abordagem inovadora no contexto nacional, visto que, embora existam práticas isoladas de divulgação de vagas e oportunidades, estas ainda se encontram dispersas e pouco sistematizadas.

A plataforma, ao centralizar estas informações, contribuirá não apenas para a democratização do acesso, mas também para a promoção de transparência nos processos de recrutamento e selecção de estagiários.

Por fim, a pertinência do estudo reside no seu impacto potencial para estudantes, empresas e instituições de ensino superior, uma vez que estabelece um canal de comunicação directo, confiável e eficiente. Assim, o desenvolvimento desta plataforma poderá fortalecer a relação entre academia e mercado, aumentar as chances de empregabilidade dos jovens e, ao mesmo tempo, facilitar às empresas a captação de talentos qualificados no cenário angolano.

Metodologia

Classificação da pesquisa

- Quanto à natureza: esta pesquisa é aplicada, pois visa o desenvolvimento de uma plataforma *web* prática e funcional que conecte estudantes universitários a empresas e oportunidades de estágio no contexto angolano. O foco recai sobre a resolução de um problema concreto, buscando implementar uma solução tecnológica que optimize a integração simultânea entre o ambiente académico e o mercado de trabalho.
- Quanto aos objectivos: classifica-se como uma pesquisa descritiva, uma vez que procura identificar, organizar e apresentar informações relevantes sobre os desafios enfrentados por estudantes e empresas no processo de recrutamento de estagiários. Além disso, descreve as funcionalidades e os requisitos necessários para o correcto desenvolvimento da plataforma.
- Segundo a abordagem do problema: a abordagem é qualitativa, pois busca compreender em profundidade as dificuldades enfrentadas pelos estudantes e pelas empresas, propondo com base neste diagnóstico, uma solução digital adequada. Os dados foram analisados de forma interpretativa, considerando tanto o conhecimento empírico quanto as informações institucionais, assim como as experiências dos actores envolvidos.

Métodos de nível teórico ou de abordagem

- Dedutivo-indutivo: a investigação parte de princípios gerais sobre desenvolvimento de sistemas *web*, usabilidade e acessibilidade, aplicando-os ao contexto angolano.
- Análise-síntese: as informações obtidas foram inicialmente analisadas de forma isolada, identificando limitações no acesso às oportunidades de estágio.

- Hipotético-dedutivo: a pesquisa formula a hipótese de que a criação da plataforma *web* contribuirá para a aproximação entre estudantes e empresas. Essa hipótese é verificada por meio do desenvolvimento e da validação do protótipo junto a potenciais utilizadores.

Métodos de nível empírico ou de procedimentos

- Observação directa: recorreu-se à observação directa, fundamentada na experiência da pesquisadora, enquanto estudante universitária e no conhecimento prévio das dificuldades recorrentes associadas à procura e oferta de oportunidades de estágio. Esta análise partiu da identificação do problema com base na vivência directa do processo de candidatura a estágios, bem como na percepção das limitações existentes nos mecanismos tradicionais de divulgação, comunicação e gestão dessas oportunidades.
- Modelagem: foram utilizados diagramas UML e protótipos visuais para representar a estrutura lógica e funcional da plataforma *web*.

Estrutura da monografia

O presente trabalho está organizado da seguinte forma:

Introdução

Apresenta o tema da pesquisa, a delimitação do problema, a justificativa, a pergunta de partida, a hipótese, o objectivo geral e os objectivos específicos, bem como a metodologia adoptada. Esta secção tem carácter introdutório e contextualiza a relevância do estudo.

Capítulo I – Fundamentação teórica

Aborda os principais conceitos e fundamentos relacionados ao desenvolvimento de plataformas *web* e à ligação entre ensino superior e mercado de trabalho. São discutidos tópicos como: empregabilidade juvenil, estágio académico, tecnologias *web* (HTML, CSS, *JavaScript*, PHP e MySQL), usabilidade e acessibilidade digital, *design* responsivo, segurança da informação e metodologias ágeis de desenvolvimento.

Capítulo II – Apresentação da plataforma *web*

Expõe as etapas práticas da criação da plataforma, desde o levantamento de requisitos até a implementação das funcionalidades. Inclui a modelagem com diagramas UML (casos de uso, classes, sequências e actividades), o *design* de *interface*, a estrutura da base de dados e os testes funcionais. Esta secção destaca as ferramentas utilizadas, os desafios enfrentados e as soluções aplicadas.

Conclusão

Apresenta uma síntese dos resultados alcançados, confrontando-os com os objectivos definidos e a hipótese formulada. Expõe as contribuições do trabalho para estudantes, empresas e instituições académicas, bem como limitações da pesquisa e sugestões para futuros estudos.

Referências

Lista todas as fontes consultadas (livros, artigos científicos, relatórios e *sites* institucionais), organizadas segundo as normas da APA (7ª edição).

Apêndices

Contém materiais complementares, como protótipos de tela, tabelas, capturas da plataforma, documentação técnica, diagramas UML e a estrutura detalhada da base de dados.

CAPÍTULO I – FUNDAMENTAÇÃO TEÓRICA

Neste capítulo, é abordada a fundamentação teórica que sustenta o desenvolvimento do presente trabalho, apresentando os principais conceitos, princípios e estudos relacionados ao tema. São discutidos aspectos teóricos sobre plataformas *web*, empregabilidade juvenil, estágios profissionais, bem como as tecnologias envolvidas no processo de desenvolvimento da plataforma.

1.1 Educação superior e o mercado de trabalho em Angola

1.1.1 Contextualização da educação superior em Angola

Desde a independência, o ensino superior em Angola passou por uma expansão significativa, ainda que marcada por limitações estruturais. No período pós-independência, a criação da Universidade Agostinho Neto (UAN) em 1976 foi vista como um passo estratégico para a afirmação institucional e o desenvolvimento académico do país. Segundo Kajibanga (2000), “o ensino superior sofreu com o aumento galopante do número de estudantes, a insuficiência de instalações e de material didáctico” (p. 10-11), o que reflectia a dificuldade de acomodar o crescimento estudantil de forma alinhada à qualidade exigida.

Já no período pós-guerra, entre 2001 e 2010, observou-se um ambicioso esforço de reconstrução do sistema educativo. De acordo com um estudo sobre a reconstrução nacional: “No ensino superior registou-se a criação de 6 novas Universidades públicas em 2009. Esta expansão do ensino superior possibilitou a triplicação do n.º de estudantes matriculados no ensino superior” (p. 9–16).

Entretanto, este crescimento rápido trouxe consigo desafios para além da infraestrutura. Como ressalta Emanuel Nkruma Paim (2024), “a qualidade do ensino é uma preocupação constante, a formação dos docentes muitas vezes não acompanha as exigências modernas do mercado, resultando em lacunas na transmissão de conhecimentos” (p. 18).

1.1.2 Desafios da inserção profissional dos estudantes universitários

A inserção dos universitários no mercado de trabalho enfrenta obstáculos reais que comprometem a transição da teoria para a prática. Segundo Maria Odete Lourenço Miguel (2015), em estudo sobre perspectivas em Angola, “os estudantes consideram como factores mais importantes para a inserção profissional a capacidade de comunicação oral e escrita”; por sua vez, “os empregadores valorizam a capacidade de trabalhar em equipa e de assumir responsabilidades” (p. 34).

Além disso, a desarticulação entre instituições de ensino e mercado leva os estudantes a buscar estágios por conta própria. Como observou o jornal Expansão (2025), “os estudantes ao terem de procurar estágios, sem ajuda institucional, isto ressalta a fragilidade de um sistema educacional que deveria ter maior interligação com o mercado de trabalho” (p. 5).

Esse mesmo cenário é reforçado por normativa recente: conforme o director nacional do Ensino Superior (2022), embora os estágios já devem ser previstos desde o início da formação, “muitos estudantes frequentam os quatro ou cinco anos de licenciatura e não têm a possibilidade de frequentar um estágio curricular”, o que compromete a integralização prática ao currículo.

1.1.3 A importância dos estágios no desenvolvimento de competências

Os estágios são fundamentais para conectar a formação académica à realidade profissional e ao desenvolvimento de habilidades práticas. Segundo um relatório sobre competências profissionais na Huíla (PNUD/OIT), a universidade aproximou-se do sector produtivo para identificar “o perfil dos quadros que necessita para as empresas”, com o objectivo de agregar valor aos recursos humanos formados.

No entanto, muitos estudantes enfrentam dificuldades para acessar estágios estruturados: o jornal Expansão (2025) destacou que, apesar de existir um protocolo entre instituições de ensino superior privadas e entidades empresariais assinado em 2021 “nunca saiu do papel”, deixando os formandos sem suporte institucional (p. 3).

A regulamentação também impõe exigências às instituições: como explicou o órgão nacional (2022), os estágios não devem ocorrer apenas ao final do curso, e as instituições devem apresentar protocolos com empresas locais, mas, na prática, poucos estudantes chegam a realizar estágios curriculares em tempo adequado, limitando o desenvolvimento de competências essenciais.

1.2 Estágios académicos e profissionais

1.2.1 Conceito e tipologia de estágios

Os estágios podem ser compreendidos como experiências práticas que complementam a formação académica, podendo assumir diferentes formas conforme os seus objectivos e duração. Essa definição ressalta a função integradora e preparatória do estágio frente à exigência profissional. Em termos de tipologia, distingue-se frequentemente entre estágios curriculares obrigatórios ou facultativos e estágios extracurriculares ou de iniciação profissional.

Essa variedade reflecte diferentes intenções pedagógicas e administrativas dentro do sistema de ensino superior.

A distinção entre estágios académicos e profissionais também se insinua na realidade angolana. Enquanto estágios académicos servem principalmente a fins formativos e de aprendizagem ainda que muitas vezes ausentes no currículo, os estágios profissionais, quando existentes, visam a real inserção no sector produtivo. Segundo Miguel Joaquim (estudante da UNIA citado pelo Expansão, 2025), “muitos de nós acabaremos por realizar actividades que não estão relacionadas à área em que nos formamos,” devido à falta da componente prática e oportunidades reais de estágio (p. 20). Isso evidencia a fragmentação entre o ensino teórico e a experiência prática esperada.

1.2.2 O papel dos estágios na transição universidade – mercado de trabalho

Os estágios representam uma ponte crucial entre a formação teórica universitária e as exigências práticas do mercado de trabalho. Segundo Karinne Manita (2024), embora existam oportunidades e potencial de crescimento pessoal e profissional, os jovens angolanos enfrentam a contradição de precisarem de experiência para conseguir um emprego, mas sem emprego não conseguem adquirir essa experiência o que torna os estágios ainda mais estratégicos nesse contexto (p. 15).

Além disso, os estágios podem suprir a lacuna entre as competências desenvolvidas na academia e as habilidades exigidas pelo mercado. Em Angola, as instituições muitas vezes não preparam adequadamente os jovens com as habilidades práticas que os empregadores procuram, e isso reforça a importância de estágios estruturados para atenuar esse descompasso.

Num cenário onde existe enorme desigualdade entre oferta e demanda de emprego, os estágios podem ampliar as oportunidades de empregabilidade. Conforme o 1.º Relatório do Emprego em Angola (2023), apenas 11% dos candidatos são absorvidos pelas vagas publicadas, evidenciando a competitividade do mercado; nesse contexto, quem possui experiência, mesmo que por meio de estágios, pode se destacar significativamente (p. 33).

1.2.3 Oportunidades e desafios de estágios em Angola

Desafios estruturais: a ausência de programas de estágio estruturados é frequente no ensino superior angolano.

Conforme a reportagem do Expansão (2025), mesmo após a assinatura de um protocolo em 2021 entre a AIESPA e o GTE para viabilizar estágios profissionais, “o protocolo nunca saiu do papel,” deixando os estudantes à própria sorte na busca por experiências práticas (p. 20).

O difícil acesso a estágios compromete o desenvolvimento de competências relevantes: sem vivência prática, muitos estudantes concluem o curso sem aplicar os conhecimentos adquiridos, reduzindo as suas probabilidades de inserção profissional. Isso reforça o carácter crítico dos estágios como oportunidade de aplicar teoria, adquirir rotina profissional e compreender as exigências reais do mercado.

Oportunidades emergentes: ainda que desafiador, o contexto angolano apresenta caminhos alternativos. A diversificação da economia e o engajamento em sectores emergentes como *fintechs*, *agritechs* e tecnologias digitais criam espaço para novas formas de estágio e aprendizagem no campo prático e inovador. Segundo Karinne Manita (2024), programas governamentais e de ONGs vêm implementando iniciativas de estágio e aprendizagem que não apenas fornecem experiência prática, como também permitem aos estudantes construir redes profissionais essenciais para o sucesso na carreira (p. 35).

1.3 As plataformas *web* como mediadoras de oportunidades

1.3.1 Conceito e características das plataformas *web*

Nos últimos anos, as plataformas *web* têm se consolidado como meios poderosos de conexão entre quem busca formação ou emprego e quem oferece oportunidades. Elas actuam como ambientes digitais estruturados para pesquisa, comparação e acesso a serviços educacionais ou profissionais. Segundo a Wikipedia sobre Educartis (2018), “a Educartis é a primeira plataforma *online* na área da educação em África e estabelece a ligação entre prestadores de serviços de educação e permite que estudantes encontrem, comparem e acessem à educação em território nacional e internacional” (p. 28).

Essas plataformas costumam oferecer filtros avançados, comparações de cursos e suporte à decisão de estudo ou, no caso de emprego, permitem cadastro de currículo, alertas de vagas, aplicação directa e até *networking* profissional. Por exemplo, plataformas como Jobartis, *LinkedIn*, Emprego.co.ao e Ango Emprego são amplamente usadas em Angola para buscar estágios e oportunidades de trabalho.

Ainda, as plataformas podem combinar soluções de aprendizagem e oportunidades profissionais. O modelo conhecido como “academia social *web*” (ou *e-learning*, com recursos colaborativos) utiliza ferramentas tipo *blogs*, *wikis* ou fóruns para promover a troca de conhecimento entre estudantes e profissionais, gerando ambientes de aprendizagem activos e cooperativos.

1.3.2 Experiências e modelos internacionais de conexão entre estudantes e empresas

No panorama internacional, há diversos exemplos de plataformas que conjugam aprendizagem prática e conexão com oportunidades profissionais. Um caso emblemático é o *Google Summer of Code* (GSoC) que conecta estudantes a mentores e projectos de código aberto, oferecendo subsídios financeiros e experiência real. Segundo o estudo sobre GSoC (2019), os estudantes são atraídos não só pelos incentivos monetários, mas sobretudo “pela experiência de trabalho e pela possibilidade de incluírem no seu resumo o nome da organização apoiadora” (p. 56).

Outro modelo internacional relevante é o de micro-*internships* ou estágios curtos e estruturados oferecidos via plataformas colaborativas. Um exemplo prático é a plataforma Atelier que transforma tarefas especializadas em micro-estágios guiados por mentores, ajudando os estagiários a progredir e a absorver boas práticas profissionais. Segundo o estudo sobre Atelier (2016), “Atelier helped interns maintain forward progress and absorb best practices” (p. 34).

Além disso, na esfera da mobilidade académica, a mobilidade mista (*blended mobility*) combina mobilidade física e virtual para promover experiências de estágio ou aprendizagem internacional. Alguns exemplos incluem plataformas como PROVIP, VM-BASE e MUTW que promovem estágios virtuais em empresas ou intercâmbios laborais com base em plataformas.

1.3.3 Potencial das plataformas *web* no contexto Angolano

Em Angola, as plataformas *web* já mostram potencial como mediadoras de oportunidades reais para estudantes e jovens profissionais. A plataforma Educartis, por exemplo, lançou-se no país em 2016, conectando mais de 2.700 instituições com 20.000 programas de estudo, abrindo portas para acesso a cursos nacionais e internacionais mesmo em contexto local.

Além disso, o mercado angolano conta com plataformas especializadas de emprego, como o Ango Emprego que diariamente publica vagas incluindo estágios e facilita a conexão entre empresas e candidatos, sendo considerado um dos *sites* mais acessados neste segmento no país.

Igualmente, ferramentas como *WorkWise Staffing* Angola oferecem categorização clara e actualizada de oportunidades, incluindo vagas de estágio em várias regiões, com aplicação directa *online*.

Por fim, iniciativas como as da ANJE, em parceria com o INEFOP e outras entidades, demonstram como plataformas e acordos institucionais podem ajudar a disponibilizar e mediar milhares de estágios profissionais. Por exemplo, em 2024 a ANJE anunciou a disponibilização de cinco mil vagas de estágio profissional remunerado, fruto de memorandos assinados com entidades de formação e seguro, o que reforça ainda mais o papel destas soluções como ponte entre educação e mercado.

1.4 Tecnologias *web*

1.4.1 Conceito e evolução das tecnologias *web*

A *Web* surgiu inicialmente como um meio de disponibilização de informação estática, evoluindo rapidamente para ambientes interactivos e colaborativos. Segundo Berners-Lee (2007), “a *Web* não é apenas uma rede de máquinas, mas uma rede de pessoas que comunicam e interagem por meio do conhecimento compartilhado” (p. 12). Esse pensamento marcou a transição da *Web* 1.0 para a *Web* 2.0, onde a interacção passou a ser central.

Com o passar dos anos, as tecnologias *web* expandiram-se para incluir não apenas páginas estáticas, mas também serviços interactivos, comércio electrónico, redes sociais e plataformas colaborativas. De acordo com O’Reilly (2005), “a *Web* 2.0 representou uma mudança de paradigma, onde os utilizadores passaram a ser produtores de conteúdo, transformando a *Internet* num espaço de inteligência colectiva” (p. 3). Essa mudança abriu espaço para novos modelos de negócio e práticas sociais.

Na actualidade, discute-se a *Web* 3.0, caracterizada pela descentralização, *blockchain*, inteligência artificial e personalização baseada em dados. Segundo Hendler, Shadbolt e Berners-Lee (2008), “a *Web* Semântica visa estruturar a informação de modo que máquinas e humanos possam cooperar de forma mais eficaz” (p. 61). Esta evolução mostra como a tecnologia *web* acompanha as necessidades sociais e económicas.

1.4.2 Linguagens de programação e *frameworks* utilizadas em desenvolvimento *web*

O desenvolvimento *web* fundamenta-se em linguagens como HTML, CSS e *JavaScript* que fornecem a base para a estrutura, estilo e interactividade das páginas. Segundo Flanagan (2020), “*JavaScript* tornou-se a linguagem universal da *Web*, essencial para aplicações dinâmicas e multiplataforma” (p. 7). Assim, o domínio destas tecnologias é indispensável para qualquer programador *web*. No lado do servidor, linguagens como PHP, *Python*, *Ruby* e Java têm papel central na construção de sistemas robustos.

De acordo com Lerdorf (2002), criador do PHP, “a simplicidade e flexibilidade da linguagem permitiram que ela fosse amplamente adoptada, tornando-se uma das ferramentas mais usadas no desenvolvimento *web back-end*” (p. 15). Essa adopção facilitou a criação de milhões de *websites* em todo o mundo.

Já os *frameworks* surgem como aceleradores do processo de desenvolvimento, trazendo organização e produtividade. Segundo Fowler (2002), “os *frameworks* encapsulam padrões de arquitectura que promovem reutilização de código e reduzem a complexidade do *software*” (p. 34). Exemplos como Laravel (PHP), *Django* (*Python*), *Spring* (Java) e *React* (*JavaScript*) consolidaram-se como soluções de referência.

1.4.3 Segurança e escalabilidade em plataformas digitais

A segurança em plataformas digitais é um dos principais desafios da era da informação. Segundo Stallings (2017), “a protecção de sistemas *web* envolve não apenas a criptografia de dados, mas também o controlo de acessos, auditoria e prevenção contra-ataques” (p. 22). A falta de segurança compromete a confiança dos utilizadores e pode gerar graves consequências económicas e sociais.

Além da segurança, a escalabilidade é um requisito fundamental em plataformas digitais modernas, pois estas precisam responder a milhões de acessos simultâneos. De acordo com Bass, Clements e Kazman (2013), “a escalabilidade é uma propriedade crítica da arquitectura de *software*, permitindo que sistemas cresçam de forma eficiente conforme aumenta a demanda” (p. 89). Plataformas como *Google* e *Facebook* são exemplos práticos desta necessidade.

Outro aspecto essencial é o equilíbrio entre segurança e desempenho. Segundo Anderson (2020), “a engenharia da segurança deve considerar não apenas a protecção de dados, mas também a usabilidade e a eficiência do sistema” (p. 45).

Assim, a construção de plataformas digitais robustas depende de soluções equilibradas que garantem ao mesmo tempo escalabilidade e confiança.

1.5 Arquitectura de sistemas *web*

1.5.1 Conceito e importância da arquitectura de sistemas

A arquitectura de sistemas representa a estrutura fundamental que define como os componentes de *software* e *hardware* interagem para alcançar os objectivos de um sistema. Ela estabelece directrizes que orientam desde o *design* até a manutenção, permitindo que as plataformas digitais sejam escaláveis e eficientes. Segundo Bass, Clements e Kazman (2013), “a arquitectura de *software* é a estrutura de um sistema, que compreendem elementos de *software*, propriedades externamente visíveis desses elementos e os relacionamentos entre eles” (p. 31).

Além de garantir organização e clareza no desenvolvimento, a arquitectura de sistemas fornece uma base sólida para suportar mudanças futuras sem comprometer a qualidade. Isso é particularmente relevante em plataformas *web*, onde há constante evolução de tecnologias e necessidade de adaptação às exigências dos usuários. De acordo com Sommerville (2011), “a arquitectura de um sistema estabelece um vínculo entre requisitos funcionais e não funcionais, actuando como guia estratégico para a evolução do *software*” (p. 92).

Outro ponto essencial é que a arquitectura de sistemas possibilita a redução de riscos no desenvolvimento, uma vez que organiza os módulos de forma a minimizar falhas de integração e de desempenho. Segundo Pressman (2011), “a definição arquitectural é uma actividade crítica que afecta directamente atributos de qualidade, como confiabilidade, desempenho e segurança” (p. 145). Assim, a arquitectura de sistemas é indispensável para plataformas *web* robustas e duradouras.

1.5.2 Modelos de arquitectura aplicáveis a plataformas *web*

Entre os principais modelos de arquitectura aplicáveis ao desenvolvimento *web*, destacam-se a arquitectura em camadas, a cliente-servidor e a orientada a serviços. A escolha do modelo depende directamente da complexidade do sistema e das necessidades de escalabilidade e segurança. Segundo Fowler (2002), “as arquitecturas em camadas permitem separar responsabilidades, favorecendo a manutenção e a evolução contínua do sistema” (p. 78).

A arquitectura cliente-servidor, por sua vez, tornou-se amplamente utilizada devido à simplicidade e à eficiência na comunicação entre os usuários e os servidores.

Esse modelo organiza as interações de forma clara, assegurando que os dados sejam processados de maneira centralizada. De acordo com Tanenbaum e Steen (2007), “o modelo cliente-servidor oferece uma divisão lógica que simplifica a gestão e aumenta a eficiência dos serviços em rede” (p. 56).

Com o avanço das necessidades de integração entre diferentes aplicações, surgiu a arquitetura orientada a serviços (SOA) que viabiliza maior flexibilidade entre sistemas. Segundo Erl (2005), “SOA é um paradigma que utiliza serviços como blocos fundamentais para desenvolver sistemas distribuídos; e garantinte modularidade” (p. 34). Dessa forma, os modelos arquiteturais garantem que plataformas *web* possam evoluir em consonância com as demandas do mercado.

1.5.3 Integração de bases de dados em sistemas *web*

A integração de bases de dados é um elemento central no funcionamento de sistemas *web*, pois garante o armazenamento e a recuperação de informações de maneira eficiente. Sem essa integração, plataformas digitais não seriam capazes de sustentar operações críticas, como autenticação de usuários ou processamento de transações. Segundo Date (2004), “os bancos de dados constituem o núcleo da maioria dos sistemas de informação, sendo indispensáveis para a consistência e confiabilidade dos dados” (p. 19).

Nos últimos anos, diferentes modelos de bases de dados passaram a ser utilizados em plataformas *web*, como os relacionais e os NoSQL, cada um com vantagens específicas. Enquanto os relacionais oferecem consistência e suporte a transações complexas, os NoSQL oferecem escalabilidade e maior flexibilidade de estrutura. De acordo com Silberschatz, Korth e Sudarshan (2011), “o modelo relacional continua sendo amplamente adotado, mas os bancos de dados não relacionais se consolidaram como alternativa viável em sistemas distribuídos e de grande escala” (p. 25).

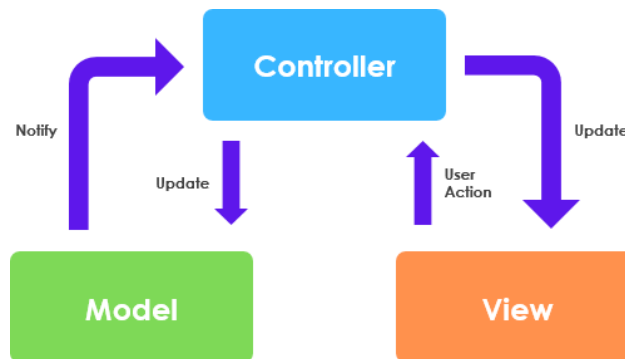
A integração das bases de dados com as aplicações *web* exige mecanismos de segurança e performance, como uso de APIs, ORM (*Object-Relational Mapping*) e criptografia de dados. Segundo Connolly e Begg (2015), “o sucesso de uma aplicação *web* depende não apenas do *design* da base de dados, mas também da forma como ela é integrada ao sistema de forma segura e eficiente” (p. 41). Assim, a integração de bases de dados é um pilar estratégico no desenvolvimento de plataformas digitais modernas.

1.6 Padrão MVC

1.6.1 Conceito do padrão MVC

Figura 1.

Padrão MVC



Fonte: https://www.cybermedian.com/pt/wp-content/uploads/sites/11/2022/02/img_6212df40ef951.png

O padrão *Model-View-Controller* (MVC) é uma abordagem arquitetural que separa a aplicação em três camadas principais: modelo, visão e controlador. Essa divisão visa organizar melhor o código, facilitando a manutenção e a evolução de sistemas de médio e grande porte. Segundo Gamma, Helm, Johnson e Vlissides (2007), “a adoção de padrões arquiteturais como o MVC possibilita uma maior clareza no *design*, favorecendo a reutilização e reduzindo a complexidade de sistemas” (p. 117).

A concepção do MVC surgiu no contexto das primeiras *interfaces* gráficas, com o objectivo de separar regras de negócio da lógica de apresentação. De acordo com Krasner e Pope (2007), “o MVC foi introduzido como um mecanismo para permitir que aplicações pudessem ser construídas de forma modular, promovendo independência entre os componentes de *software*” (p. 39). Isso tornou possível evoluir cada parte da aplicação sem comprometer o todo.

Na prática, o padrão tornou-se um dos mais utilizados no desenvolvimento de sistemas *web*, especialmente em *frameworks* modernos. Conforme Fowler (2002), “o MVC representa não apenas uma separação de responsabilidades, mas também uma estratégia para lidar com a complexidade inerente ao *software* corporativo” (p. 55).

1.6.2 Estrutura e componentes do padrão MVC (*Model*, *View*, *Controller*)

O *Model* é responsável pela lógica de negócio e pelo acesso aos dados, sendo o núcleo da aplicação. Ele garante a integridade das normas que regem o sistema.

Segundo Larman (2004), “o modelo encapsula as regras e o estado da aplicação, assegurando consistência entre as operações e o domínio de negócio” (p. 142).

A *View* trata da apresentação da informação ao utilizador, exibindo os dados processados pelo modelo. De acordo com Bass, Clements e Kazman (2013), “a separação da camada de apresentação permite maior flexibilidade, possibilitando que a *interface* seja alterada sem modificar a lógica interna do sistema” (p. 97). Isso é fundamental em ambientes onde a experiência do utilizador é determinante.

Já o *Controller* actua como mediador entre modelo e visão, recebendo as requisições do utilizador, processando-as e direccionando as respostas adequadas. Para Beck (2007), “o controlador é essencial no fluxo de comunicação da aplicação, pois assegura que as interações entre usuário e sistema sejam consistentes e bem estruturadas” (p. 63).

1.6.3 Vantagens e aplicabilidade do MVC em projectos *web*

O uso do MVC traz benefícios significativos para projectos *web*, especialmente em termos de escalabilidade. Segundo Pressman (2016), “a adopção de arquitecturas baseadas em padrões como o MVC contribui para reduzir custos de manutenção e aumentar a longevidade de sistemas em produção” (p. 221).

Outro ponto importante é a possibilidade de trabalho em equipa de forma mais eficiente. De acordo com Sommerville (2011), “a divisão em camadas como proposta pelo MVC permite que diferentes perfis de desenvolvedores trabalhem de forma paralela, minimizando dependências e conflitos” (p. 188). Isso torna o desenvolvimento mais ágil e estruturado.

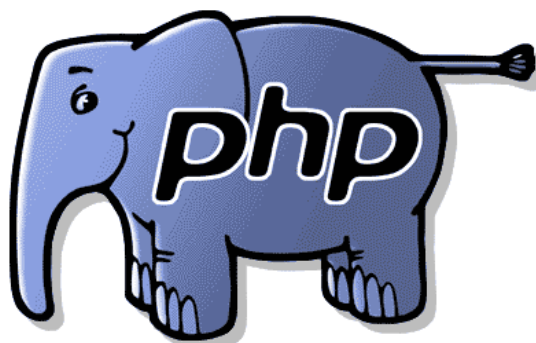
Além disso, o MVC é amplamente aplicável em *frameworks* e linguagens actuais, como Laravel, *Django* e *Spring*. Conforme Freeman e Freeman (2004), “a popularização do MVC em *frameworks* modernos demonstra a sua relevância contínua, pois ele oferece uma base sólida para enfrentar os desafios do desenvolvimento *web*” (p. 201).

1.7 Ferramentas utilizadas para o desenvolvimento da plataforma *web*

1.7.1 PHP

Figura 2.

PHP



Fonte: <https://thiagomellodesign.wordpress.com/wp-content/uploads/2012/10/php.png>

O PHP é uma linguagem de programação voltada para o desenvolvimento *web* que se consolidou como uma das mais utilizadas em aplicações dinâmicas. Segundo Welling e Thomson (2009), “o PHP emergiu como uma linguagem poderosa pela sua simplicidade de integração com o HTML e a sua forte ligação ao desenvolvimento de sistemas baseados em servidores *web*” (p. 22). Essa característica tornou-o adequado para plataformas académicas que necessitem de interacção em tempo real entre utilizadores e base de dados.

Além disso, o PHP oferece suporte a múltiplos *frameworks* e bibliotecas, o que facilita a criação de funcionalidades robustas em pouco tempo. Para Lerdorf (2010), “a grande força do PHP é a sua capacidade de adaptar-se a diferentes contextos de desenvolvimento sem perder desempenho” (p. 14). No contexto da plataforma proposta, essa flexibilidade garante a criação de módulos que respondam às necessidades dos estudantes e das empresas.

O uso do PHP também se justifica pela sua comunidade activa e vasta documentação. De acordo com Ullman (2011), “a aprendizagem contínua e o suporte colaborativo proporcionado pela comunidade PHP são factores determinantes para a sua consolidação no meio académico e corporativo” (p. 37). Essa característica facilita a manutenção futura da plataforma e a integração com outras tecnologias.

1.7.2 JavaScript

Figura 3.

JavaScript



Fonte: https://static.vecteezy.com/system/resources/previews/027/127/463/non_2x/javascript-logo-javascript-icon-transparent-free-png.png

O *JavaScript* é essencial para criar interactividade e dinamismo em páginas *web*. Conforme Flanagan (2011), “o *JavaScript* tornou-se a linguagem da *web*, permitindo que os desenvolvedores criem *interfaces* mais naturais e reactivas para os utilizadores” (p. 45). No caso da plataforma, isso possibilita uma melhor experiência para os estudantes e empresas ao navegarem no sistema.

Outro ponto importante é a evolução do *JavaScript* com bibliotecas e *frameworks* que ampliam o seu poder de aplicação. Segundo Crockford (2008), “a simplicidade estrutural do *JavaScript* permitiu o surgimento de ferramentas que potencializam o desenvolvimento de aplicações complexas e interactivas” (p. 29). Isso é fundamental para implementar funcionalidades como formulários inteligentes e *dashboards* na plataforma.

Por fim, a compatibilidade do *JavaScript* com praticamente todos os navegadores garante a sua eficácia. De acordo com Resig (2013), “a universalidade do *JavaScript* transformou-o num recurso indispensável para qualquer aplicação *web* moderna” (p. 11). Essa característica torna a ferramenta estratégica para atender diferentes utilizadores da plataforma, independentemente do dispositivo utilizado.

1.7.3 *Bootstrap*

Figura 4.

Bootstrap



Fonte: <https://cdn-icons-png.flaticon.com/512/5968/5968672.png>

O *Bootstrap* é um *framework front-end* que facilita a criação de *interfaces* modernas e responsivas. Segundo Spurlock (2013), “o *Bootstrap* revolucionou o desenvolvimento *web* ao disponibilizar componentes prontos que garantem consistência visual e agilidade” (p. 18). Isso é relevante para a plataforma que precisa atender estudantes e empresas em diferentes dispositivos.

A utilização do *Bootstrap* também está associada à padronização de elementos gráficos. Para Finnegan (2014), “a uniformidade trazida pelo *Bootstrap* não apenas reduz erros de *design*, mas também promove uma experiência mais intuitiva ao utilizador” (p. 41). Isso contribui directamente para a atractividade e usabilidade da plataforma.

Outro factor positivo é a sua compatibilidade com outras tecnologias. Conforme Silvers (2015), “a integração do *Bootstrap* com linguagens e *frameworks back-end* torna-o ideal para sistemas que exigem escalabilidade e flexibilidade” (p. 27). Dessa forma, o *framework* fortalece a implementação da solução desenvolvida.

1.7.4 *MySQL workbench*

Figura 5.

MySQL workbench



Fonte: https://images.icon-icons.com/1381/PNG/512/mysqlworkbench_93532.png

O *MySQL Workbench* é uma ferramenta poderosa para modelagem, administração e manutenção de bases de dados. De acordo com DuBois (2013), “o *MySQL Workbench* representa uma evolução na gestão de dados, permitindo que desenvolvedores visualizem, otimizem e implementem estruturas complexas” (p. 33). No projecto, é utilizado para estruturar e gerir informações de estudantes e empresas.

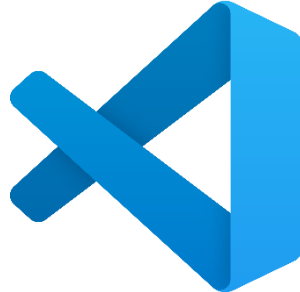
Outro diferencial é a sua *interface* gráfica intuitiva. Segundo Silva (2016), “a visualização gráfica oferecida pelo *MySQL Workbench* simplifica processos de modelagem e reduz a margem de erro na criação de bases de dados relacionais” (p. 56). Essa característica garante maior confiabilidade ao sistema desenvolvido.

Além disso, a ferramenta se destaca pela integração directa com o servidor *MySQL*. Conforme Stephens (2012), “o alinhamento entre o *MySQL Workbench* e o servidor *MySQL* assegura fluidez na administração de dados e na execução de consultas complexas” (p. 22). Isso torna o ambiente mais eficiente para a manipulação das informações necessárias.

1.7.5 Visual *studio* code

Figura 6.

Visual studio code



Fonte: <https://icons.veryicon.com/png/o/business/vscode-program-item-icon/vscode.png>

O Visual *Studio Code* (VS *Code*) é um editor de código amplamente adoptado pela sua leveza e extensibilidade. Segundo Han et al. (2017), “o VS *Code* destaca-se por oferecer recursos avançados de edição sem comprometer o desempenho” (p. 15). Essa característica torna-o ideal para o desenvolvimento ágil da plataforma.

Outro aspecto importante são as suas extensões. Para Rees (2018), “o ecossistema de *plugins* do VS *Code* permite que cada desenvolvedor personalize o ambiente de acordo com as suas necessidades” (p. 40). Isso facilita a integração com PHP, *JavaScript*, *MySQL* e outras tecnologias da plataforma.

Além disso, a sua *interface* simplificada contribui para maior produtividade. De acordo com Brown (2019), “a curva de aprendizagem do VS *Code* é baixa, mas a sua capacidade de adaptação a projectos complexos é elevada” (p. 27). Isso garante uma melhor organização e manutenção do código do sistema.

1.7.6 XAMPP

Figura 7.

XAMPP



Fonte: <https://cdn.freebiesupply.com/logos/large/2x/xampp-logo-png-transparent.png>

O XAMPP é uma solução que integra servidor Apache, *MySQL*, PHP e *Perl* num único pacote. Conforme Lam (2010), “o XAMPP facilita a configuração de ambientes de desenvolvimento locais, eliminando barreiras técnicas iniciais” (p. 21). Isso é essencial para testar a plataforma antes da sua implantação.

A sua simplicidade é outro ponto de destaque. Segundo Ranjan (2012), “o XAMPP democratizou o desenvolvimento ao permitir que iniciantes e especialistas pudessem configurar servidores de forma prática e rápida” (p. 38). Dessa forma, a equipa de desenvolvimento pode validar funcionalidades com maior agilidade.

Além disso, o XAMPP é multiplataforma e amplamente difundido. Para Singh (2014), “a popularidade do XAMPP está associada à sua flexibilidade e à capacidade de replicar ambientes de produção em máquinas locais” (p. 44). Essa característica reforça a sua importância no ciclo de desenvolvimento da plataforma.

1.7.7 *StarUML*

Figura 8.

StarUML



Fonte: https://www.nicepng.com/png/full/247-2477083_staruml-definition.png

O *StarUML* é uma ferramenta amplamente utilizada para a criação de diagramas UML de forma gráfica e intuitiva. Segundo Roques (2017), “o *StarUML* promove a produtividade ao oferecer uma *interface* visual que facilita a modelagem de sistemas complexos” (p. 19). Isso contribui para uma documentação mais clara e estruturada da plataforma.

Outro ponto relevante é a colaboração em equipa. Para Brown e Smith (2019), “a *interface* amigável do *StarUML* possibilita que equipas multidisciplinares colaborem no processo de modelagem, mesmo sem conhecimentos profundos em notação UML” (p. 25). Esse aspecto amplia a compreensão do sistema entre todos os envolvidos.

Além disso, a sua integração com outros ambientes de desenvolvimento é uma vantagem. De acordo com Fowler (2010), “a capacidade do *StarUML* de exportar e importar modelos, bem como de se integrar a diferentes linguagens e *frameworks*, fortalece a consistência e a rastreabilidade dos projectos de *software*” (p. 67). Essa característica é fundamental para manter a documentação técnica da plataforma sempre actualizada.

CAPÍTULO II – APRESENTAÇÃO DA PLATAFORMA *WEB*

Neste capítulo, é apresentada a plataforma *web* desenvolvida, descrevendo a sua estrutura, funcionalidades e principais componentes. Este capítulo visa demonstrar de que modo a plataforma foi planeada para conectar estudantes e empresas, promovendo um ambiente digital que facilite o acesso a oportunidades de estágio e torna o processo de recrutamento mais eficiente, transparente e inclusivo.

2.1 Desenvolvimento da plataforma

O desenvolvimento da plataforma *web* foi orientado pela necessidade de criar um ambiente digital capaz de integrar estudantes universitários e empresas, promovendo o acesso facilitado a oportunidades de estágio. O processo de concepção envolveu etapas de análise de requisitos e modelagem, com foco na usabilidade, acessibilidade e adequação ao contexto angolano. A arquitectura da plataforma foi estruturada de modo a permitir a gestão eficiente de utilizadores, o registo e actualização de perfis, a publicação de vagas e a interacção entre os intervenientes.

2.2 Metodologia utilizada para o desenvolvimento da plataforma *web*

Figura 9.

Extreme programming (Ciclo de desenvolvimento)



Fonte: https://artia.com/wp-content/uploads/2024/09/Imagens_Prancheta-1-01-780x780.png

A metodologia utilizada no desenvolvimento baseou-se em princípios ágeis, mais especificamente na abordagem *Extreme Programming* (XP), escolhida pela sua flexibilidade e capacidade de adaptação a mudanças frequentes. Essa metodologia orientou o trabalho em ciclos curtos de desenvolvimento, permitindo a revisão constante das funcionalidades e o aprimoramento contínuo do sistema. A escolha dessa abordagem justifica-se pela natureza dinâmica do projecto que exige respostas rápidas às necessidades dos utilizadores e a entrega de resultados progressivos. Assim, a metodologia aplicada contribuiu para a eficiência do processo de desenvolvimento, para a redução de erros e para a construção de uma solução tecnológica robusta e ajustada aos objectivos do estudo.

2.2.1 Características, vantagens e desvantagens da metodologia

A metodologia *Extreme Programming* (XP) caracteriza-se pela sua natureza ágil, interactiva e colaborativa, promovendo a entrega contínua de *software* funcional através de pequenos ciclos de desenvolvimento. Entre as suas principais características destacam-se a comunicação constante entre os membros da equipa, o *feedback* rápido dos utilizadores, o refinamento contínuo do código e a simplicidade das soluções.

Entre as vantagens da XP estão a alta qualidade do produto final, a rápida adaptação a mudanças de requisitos, a redução de erros e o forte envolvimento do cliente no processo de desenvolvimento. No entanto, a metodologia também apresenta desvantagens, como a necessidade de grande comprometimento por parte da equipa e do cliente, a dificuldade de aplicação em projectos de grande escala e a dependência de uma comunicação eficaz para que os resultados sejam satisfatórios.

2.2.2 Benefícios

A adopção da metodologia *Extreme Programming* trouxe benefícios significativos para o desenvolvimento da plataforma *web*. Entre eles, destaca-se o melhor alinhamento entre as expectativas dos utilizadores e as funcionalidades implementadas, devido à constante interacção e *feedback* durante o processo. O ritmo interactivo permitiu identificar e corrigir falhas precocemente, garantindo um produto mais estável e eficiente.

2.3 Requisitos funcionais (RF)

São descrições das funcionalidades que o sistema deve oferecer para satisfazer as necessidades dos utilizadores.

Tabela 1.*Requisitos funcionais*

Nº	Tipo	Requisito	Descrição
RF01	Funcional	Cadastro de estudante	A plataforma permite o cadastro de estudantes mediante preenchimento de formulário com dados pessoais e académicos.
RF02	Funcional	Cadastro de empresa	A plataforma permite o cadastro de empresas com informações institucionais, área de actuação e documentos de validação.
RF03	Funcional	<i>Login</i> por perfil	A plataforma permite que Estudantes, Empresas e Administradores acessem utilizando <i>email</i> e senha previamente cadastrados.
RF04	Funcional	Recuperação de senha	A plataforma permite que qualquer perfil recupere a senha através de envio de <i>link</i> de redefinição por <i>email</i> .

RF05	Funcional	Gestão de vagas	A plataforma permite que as Empresas criem, editem e removam anúncios de estágios com definição de requisitos, modalidade e prazo de candidatura.
RF06	Funcional	Busca de vagas	A plataforma permite que os Estudantes pesquisem vagas através de filtros por área de actuação, localização, modalidade e duração.
RF07	Funcional	Candidatura a vagas	A plataforma permite que os Estudantes submetam candidatura a uma vaga com o envio de currículo e carta de motivação.
RF08	Funcional	Histórico de candidaturas	A plataforma permite que os Estudantes consultem todas as candidaturas já realizadas, incluindo estado (pendente, aceite ou rejeitado).

RF09	Funcional	Painel do estudante	A plataforma apresenta ao Estudante um painel com oportunidades disponíveis, estado de candidaturas e dados do perfil académico.
RF10	Funcional	Painel da empresa	A plataforma apresenta à Empresa um painel com todas as vagas publicadas, candidaturas recebidas e ferramentas de filtragem.
RF11	Funcional	Painel do administrador	A plataforma permite que o Administrador tenha acesso ao painel de gestão com controlo de usuários, vagas e estatísticas do sistema.
RF12	Funcional	Relatórios estatísticos e	A plataforma dispõe de um painel de relatórios sobre número de vagas publicadas por mês, candidaturas recebidas por vaga, empresas mais activas e áreas mais procuradas.

RF13	Funcional	Notificações por <i>email</i>	A plataforma envia <i>emails</i> automáticos para Estudantes e Empresas sobre candidaturas, actualizações de vagas ou alterações de estado.
-------------	-----------	-------------------------------	---

Fonte: Elaborado pela autora, 2026.

2.4 Requisitos não funcionais (RNF)

São características de qualidade ou restrições que o sistema deve cumprir, mas que não estão directamente relacionadas com as funcionalidades.

Tabela 2.

Requisitos não funcionais

Nº	Tipo	Requisito	Descrição
RNF01	Não funcional	Usabilidade	A plataforma deve possuir uma <i>interface</i> intuitiva, responsiva e adaptada a diferentes dispositivos.
RNF02	Não funcional	Segurança de autenticação	Os dados de <i>login</i> devem ser armazenados de forma segura (<i>hash</i> de senha) e transmitidos via protocolo HTTPS.

RNF03	Não funcional	Disponibilidade	A plataforma deve estar disponível pelo menos 99% do tempo, garantindo acesso contínuo para todos os perfis.
RNF04	Não funcional	Desempenho	As páginas principais devem carregar em menos de 3 segundos em conexões de <i>internet</i> padrão.
RNF05	Não funcional	Escalabilidade	A plataforma deve suportar crescimento no número de usuários, vagas e candidaturas sem perda significativa de desempenho.
RNF06	Não funcional	Compatibilidade	A plataforma deve ser compatível com os navegadores mais utilizados (<i>Chrome, Edge, Firefox, Safari</i>).
RNF07	Não funcional	Privacidade dos dados	Os dados pessoais dos usuários devem ser protegidos de acordo com normas de protecção de dados, garantindo confidencialidade e não partilha sem autorização.

RNF08	Não funcional	<i>Backup</i> e recuperação	O sistema deve realizar <i>backups</i> automáticos periódicos e permitir a recuperação de informações em caso de falha.
RNF09	Não funcional	Confiabilidade	A plataforma deve manter a integridade dos dados, mesmo em situações de falha ou encerramento inesperado.
RNF10	Não funcional	Manutenibilidade	O sistema deve ser desenvolvido de forma modular, facilitando futuras actualizações e correcções.
RNF11	Não funcional	Acessibilidade	A plataforma deve seguir boas práticas de acessibilidade (uso de contraste, textos alternativos, navegação por teclado) para atender usuários com necessidades especiais.

RNF12	Não funcional	Notificações em tempo real	O sistema deve garantir o envio de notificações por <i>email</i> em tempo real após acções críticas (candidatura, aprovação ou rejeição).
--------------	---------------	----------------------------	---

Fonte: Elaborado pela autora, 2026.

2.5 Casos de uso

Um Caso de Uso é uma técnica da modelagem de sistemas que descreve, de forma textual ou gráfica, as funcionalidades que um sistema deve oferecer e as interacções entre os actores e o próprio sistema para atingir um objectivo específico.

Tabela 3.

Casos de uso

Código	Nome do caso de uso	Actor (es)	Descrição
UC01	Efectuar cadastro	Estudante e empresa	O estudante e a empresa preenchem um formulário com os seus dados pessoais e documentos (opcional) para efectuar o cadastro na plataforma.
UC02	Efectuar <i>login</i> por perfil	Estudante, empresa e administrador	Os usuários fornecem <i>email</i> e senha para acessar ao respectivo painel conforme o seu perfil.

UC03	Recuperar senha	Estudante, empresa e administrador	Os usuários solicitam redefinição de senha e recebem um <i>link</i> de recuperação no <i>email</i> associado à sua conta.
UC04	Gerir vagas de estágio	Empresa	A empresa pode criar, editar ou remover anúncios de estágio, definindo requisitos, modalidade e prazo de candidatura.
UC05	Pesquisar vagas	Estudante	O estudante utiliza filtros (área de actuação, localização, modalidade e duração) para procurar oportunidades de estágio disponíveis.
UC06	Submeter candidatura	Estudante	O estudante candidata-se a uma vaga através do envio de currículo e carta de motivação.
UC07	Consultar histórico de candidaturas	Estudante	O estudante consulta todas as candidaturas realizadas, visualizando o estado (pendente, aceite ou rejeitado).

UC08	Acessar ao painel	Estudante, empresa e administrador	Os usuários têm acesso ao seu painel com funcionalidades recomendadas.
UC09	Visualizar painel estatístico	Administrador	O administrador consulta relatórios sobre número de vagas publicadas por mês, candidaturas recebidas, empresas mais activas e áreas mais procuradas.
UC10	Enviar notificações por <i>email</i>	Plataforma (Automático)	A plataforma envia <i>emails</i> automáticos para estudantes e empresas sobre candidaturas, actualizações de vagas ou alterações de estado.

Fonte: Elaborado pela autora, 2026.

2.5.1 Descrição dos casos de uso

UC01 – Efectuar cadastro:

- **Actores:** estudante, empresa.
- **Objectivo:** permitir que estudantes e empresas realizem o cadastro inicial na plataforma.
- **Pré-condições:** o usuário deve ter acesso à *internet* e um endereço de *email* válido.
- **Fluxo principal:**
 1. O usuário selecciona a opção de cadastro.
 2. Preenche o formulário com os seus dados pessoais (estudante) ou institucionais (empresa).

3. Faz *upload* de documentos de validação.
 4. Submete o formulário.
 5. A plataforma valida os dados e confirma o cadastro.
- **Fluxo alternativo:**
 - Se os dados estiverem incompletos, o sistema solicita correcção antes da submissão.
 - **Pós-condições:** o usuário recebe uma conta activa na plataforma.
 - **Excepções:** *email* já cadastrado ou formato inválido → o sistema informa erro.

UC02 – Efectuar *login* por perfil

- **Actores:** estudante, empresa e administrador.
- **Objectivo:** permitir acesso ao painel de acordo com o perfil.
- **Pré-condições:** o usuário já deve estar cadastrado.
- **Fluxo principal:**
 1. O usuário acessa a página de *login*.
 2. Fornece *email* e senha.
 3. A plataforma autentica as credenciais.
 4. Redirecciona o usuário ao painel correspondente ao perfil (estudante, empresa ou administrador).
- **Fluxo alternativo:**
 - Se a senha estiver incorrecta, o sistema sugere recuperar senha.
- **Pós-condições:** usuário logado com acesso ao painel adequado.

UC03 – Recuperar senha:

- **Actores:** estudante, empresa e administrador.
- **Objectivo:** permitir redefinição de senha em caso de esquecimento.
- **Pré-condições:** o usuário já deve ter conta cadastrada.

- **Fluxo principal:**

1. O usuário selecciona “Esqueci minha senha”.
2. Informa o *email* associado à conta.
3. A plataforma valida a existência do *email*.
4. Um *link* de recuperação é enviado por *email*.
5. O usuário redefine a senha através do *link*.

- **Fluxo alternativo:**

- *Email* não encontrado → o sistema exibe aviso.

- **Pós-condições:** senha actualizada e conta acessível.

- **Excepções:** *link* expirado → solicitar novo *link*.

UC04 – Gerir vagas de estágio:

- **Actor:** empresa.

- **Objectivo:** permitir que empresas façam gestão de oportunidades de estágio.

- **Pré-condições:** a empresa deve estar cadastrada e autenticada.

- **Fluxo principal:**

1. A empresa acessa o painel de vagas.
2. Cria uma nova vaga, preenchendo título, requisitos, modalidade e prazo.
3. (Opcional) Edita ou remove vagas já publicadas.
4. A plataforma publica ou actualiza a vaga.

- **Fluxo alternativo:**

- Se faltar informação obrigatória, o sistema solicita correcção.

- **Pós-condições:** vaga disponível na plataforma.

- **Excepções:** tentativa de editar vaga expirada → o sistema bloqueia alteração.

UC05 – Pesquisar vagas:

- **Actor:** estudante.
- **Objectivo:** facilitar a busca de vagas de estágio com filtros específicos.
- **Pré-condições:** o estudante deve estar cadastrado e logado.
- **Fluxo principal:**
 1. O estudante acessa a área de pesquisa.
 2. Aplica filtros (área, localização, modalidade, duração).
 3. O sistema lista vagas correspondentes.
 4. O estudante visualiza os detalhes da vaga.
- **Fluxo alternativo:**
 - Se nenhum resultado for encontrado, o sistema sugere ampliar filtros.
- **Pós-condições:** lista de vagas disponível para candidatura.
- **Excepções:** filtros inválidos ou erro de conexão.

UC06 – Submeter candidatura:

- **Actor:** estudante.
- **Objectivo:** permitir que o estudante se candidate a uma vaga.
- **Pré-condições:** o estudante deve estar logado e ter um currículo/carta de motivação preparados.
- **Fluxo principal:**
 1. O estudante escolhe uma vaga.
 2. Faz *upload* do currículo e da carta de motivação.
 3. Confirma a candidatura.
 4. A plataforma regista a candidatura.
- **Fluxo alternativo:**
 - Se faltar documento obrigatório, o sistema solicita envio antes da submissão.

- **Pós-condições:** a candidatura é associada ao histórico do estudante.
- **Excepções:** vaga expirada → candidatura rejeitada.

UC07 – Consultar histórico de candidaturas:

- **Actor:** estudante.
- **Objectivo:** permitir acompanhar todas as candidaturas já realizadas.
- **Pré-condições:** o estudante deve ter submetido pelo menos uma candidatura.
- **Fluxo principal:**
 1. O estudante acessa o histórico.
 2. Visualiza lista de candidaturas realizadas.
 3. O sistema mostra o estado (pendente, aceite, rejeitado).
- **Fluxo alternativo:**
 - Sem candidaturas → o sistema informa “nenhuma candidatura encontrada”.
- **Pós-condições:** estudante tem visão consolidada das suas candidaturas.

UC08 – Acessar ao painel:

- **Actores:** estudante, empresa e administrador.
- **Objectivo:** disponibilizar área personalizada conforme o perfil.
- **Pré-condições:** o usuário deve estar autenticado.
- **Fluxo principal:**
 1. Após *login*, o sistema identifica o perfil do usuário.
 2. Exibe o painel correspondente:
 - Estudante: vagas e candidaturas.
 - Empresa: gestão de vagas.
 - Administrador: estatísticas e gestão global.
- **Pós-condições:** o usuário interage com as funcionalidades específicas.

UC09 – Visualizar painel estatístico:

- **Actor:** administrador.
- **Objectivo:** consultar relatórios e estatísticas da plataforma.
- **Pré-condições:** administrador autenticado.
- **Fluxo principal:**
 1. O administrador acessa a área de estatísticas.
 2. Selecciona o tipo de relatório (vagas, candidaturas, empresas activas, áreas procuradas).
 3. O sistema gera gráficos e relatórios.
- **Pós-condições:** administrador obtém visão geral do uso da plataforma.

UC10 – Enviar notificações por *email*:

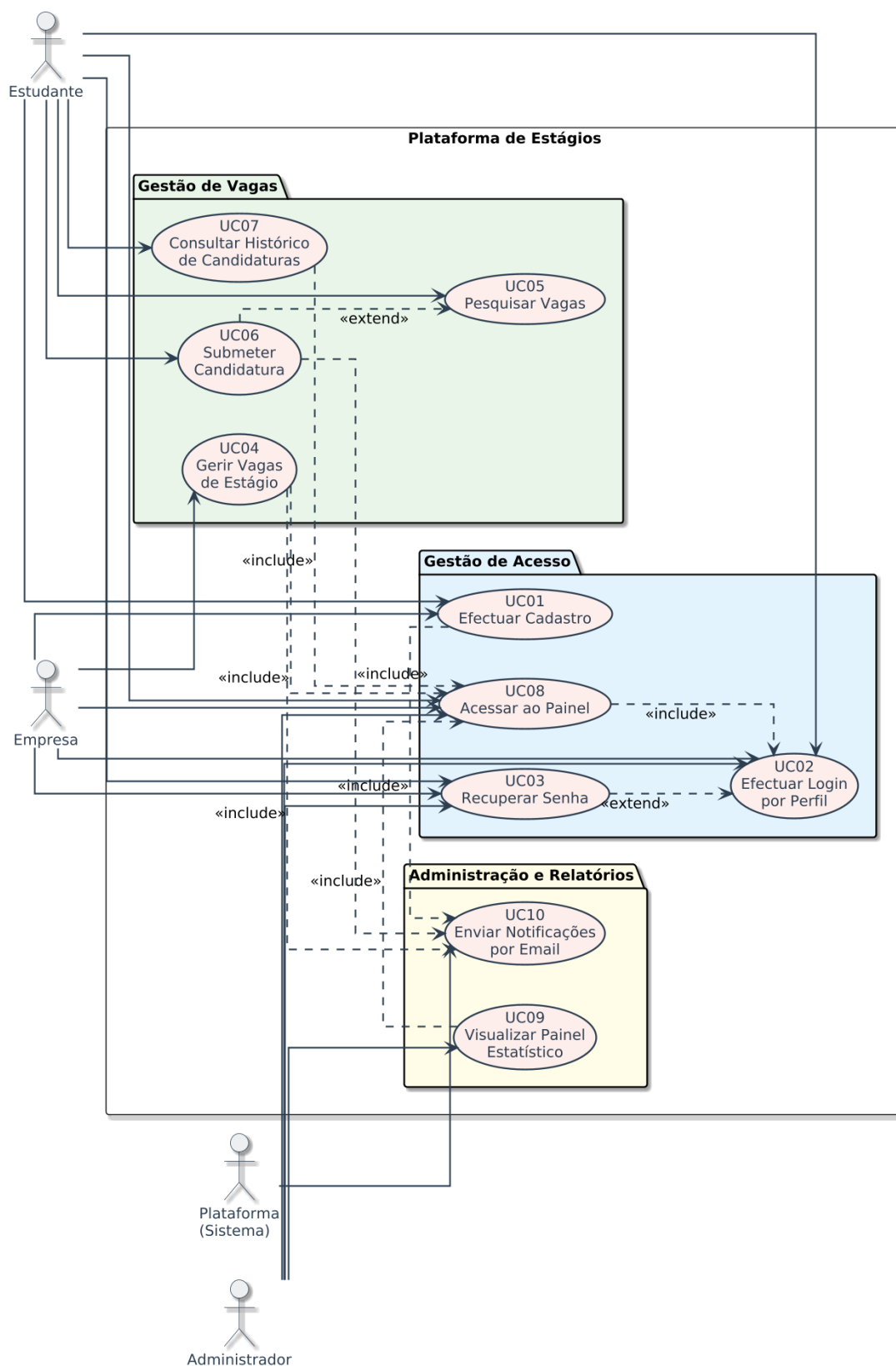
- **Actor:** plataforma (automático).
- **Objectivo:** garantir comunicação eficiente sobre actividades na plataforma.
- **Pré-condições:** o usuário deve ter *email* válido cadastrado.
- **Fluxo principal:**
 1. O sistema identifica eventos relevantes (nova vaga, candidatura recebida, actualização de estado).
 2. Gera automaticamente o conteúdo do *email*.
 3. Envia notificação para o estudante ou empresa.
- **Pós-condições:** usuário informado sobre actualizações importantes.
- **Excepções:** falha no envio do *email* → o sistema tenta reenviar ou alerta administrador.

2.6 Diagrama de caso de uso:

É uma representação gráfica em UML que mostra as interacções entre os actores (utilizadores ou sistemas externos) e o sistema em estudo. Demonstra as principais funcionalidades (casos de uso) oferecidas, de forma simples e acessível, ajudando a compreender o que o sistema fará sob a perspectiva do utilizador.

Figura 10.

Diagrama de caso de uso



Fonte: Elaborado pela autora, 2026.

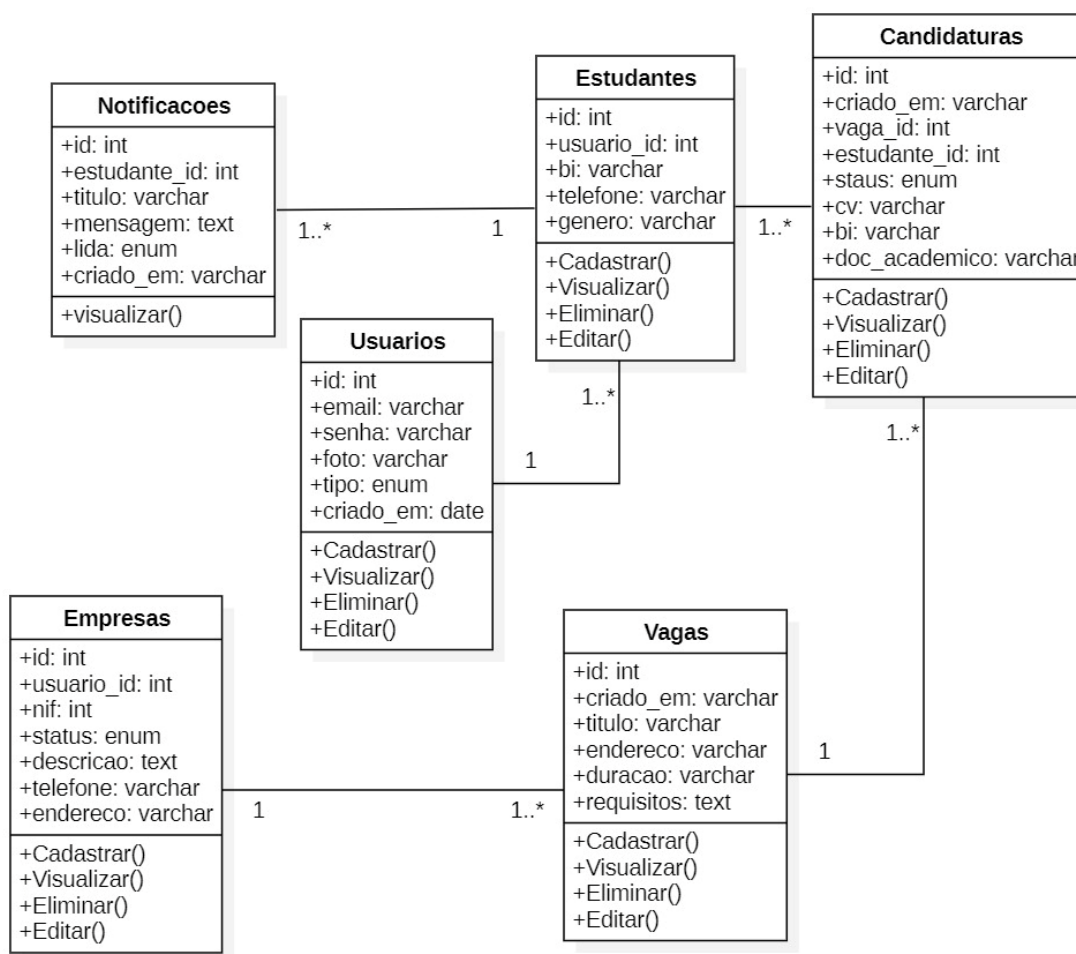
Figura 10 – Diagrama de caso de uso: este diagrama representa as principais funcionalidades da plataforma de estágios, destacando os diferentes actores (estudante, empresa, administrador e a própria plataforma) e os casos de uso relacionados, como cadastro, *login*, gestão de vagas, submissão de candidaturas e envio de notificações, incluindo ainda os relacionamentos de *include* e *extend* que mostram dependências e fluxos alternativos.

2.7 Diagrama de classe

É um modelo em UML que descreve a estrutura estática do sistema. Mostra as classes (objectos do mundo real ou lógico), os seus atributos (dados), os métodos (funções) e as relações entre elas (associação, herança e composição). É essencial para o desenho da arquitectura orientada a objectos.

Figura 11.

Diagrama de classe



Fonte: Elaborado pela autora, 2026.

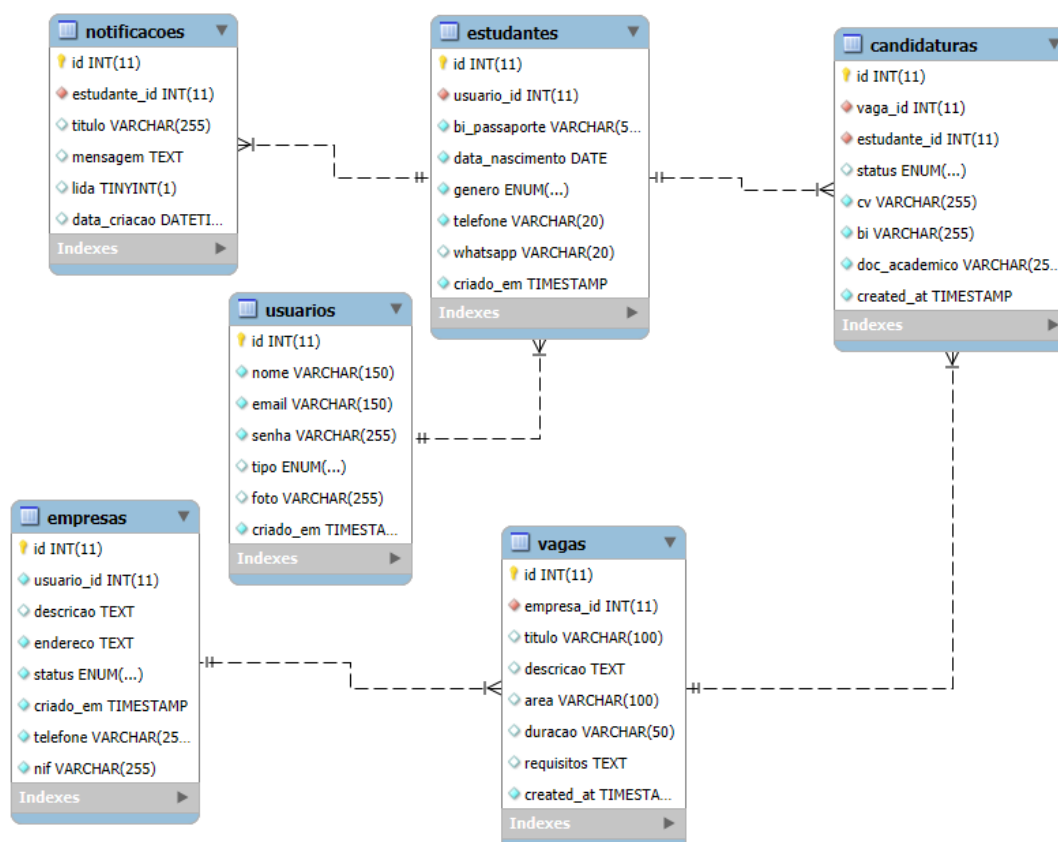
Figura 11 – Diagrama de classe: este diagrama representa a estrutura lógica do sistema de gestão de estágios, mostrando as principais classes (usuários, estudantes, empresas, vagas, candidaturas e notificações) com os seus atributos e métodos. Ele evidencia também os relacionamentos entre as classes, como por exemplo: um usuário pode estar associado a um estudante ou a uma empresa; um estudante pode realizar várias candidaturas; e as candidaturas estão sempre ligadas a vagas específicas. Além disso, as notificações são direccionadas aos estudantes, reforçando o papel da plataforma em interligar acções e garantir a comunicação automática.

2.8 Diagrama entidade-relacionamento (ER)

É uma representação gráfica usada na modelagem de bases de dados. Mostra as entidades (tabelas), os seus atributos (campos/colunas) e os relacionamentos entre essas entidades (um-para-um, um-para-muitos, muitos-para-muitos). É usado para estruturar a base de dados de forma lógica e consistente.

Figura 12.

Diagrama entidade-relacionamento (ER)



Fonte: Elaborado pela autora, 2026.

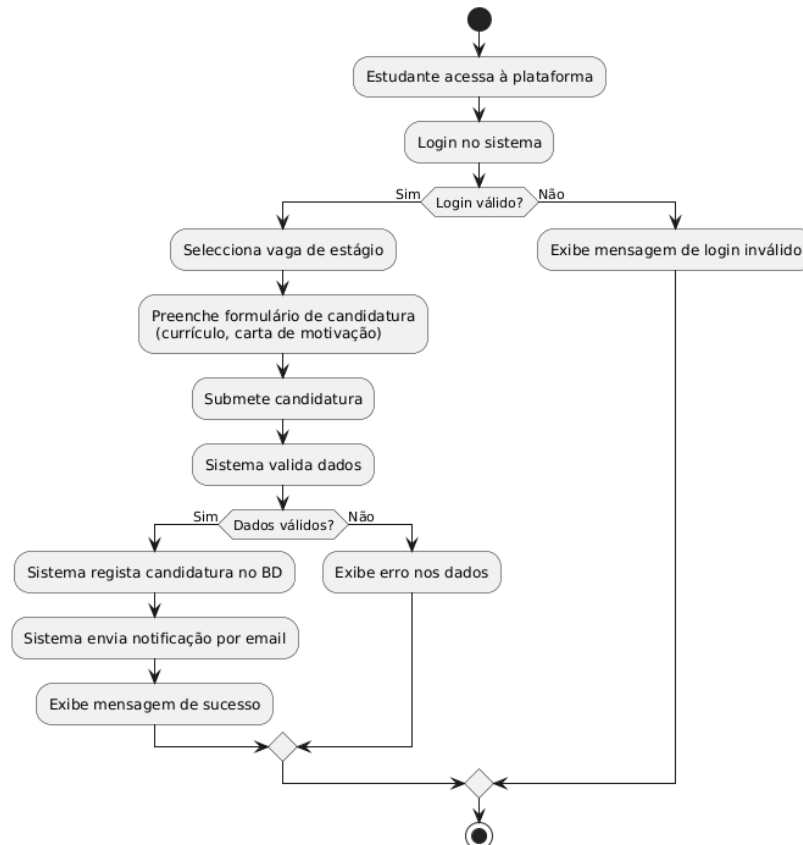
Figura 12 – Diagrama entidade-relacionamento “ER”: este diagrama descreve a organização física dos dados no banco, detalhando entidades, “atributos e chaves primárias/estrangeiras”. Mostra, por exemplo, que a tabela “usuários” contém informações centrais de *login* e perfil, enquanto as tabelas “estudantes” e “empresas” herdam a relação com “usuários”. As entidades “vagas” e “candidaturas”; estabelecem a ligação entre “empresas” e “estudantes”, permitindo o registo de aplicações a “oportunidades”. Já a tabela notificações está associada aos estudantes e armazena avisos gerados automaticamente pelo sistema. Assim, o diagrama ER traduz em nível de banco de dados as mesmas relações de negócio vistas no diagrama de classes, mas com foco em implementação.

2.9 Diagrama de actividade

É um diagrama UML que representa o fluxo de actividades ou processos dentro de um sistema. Mostra a sequência de acções, as decisões, as bifurcações e os pontos de sincronização. É muito usado para modelar fluxos de trabalho, processos de negócio ou lógica de algoritmos.

Figura 13.

Diagrama de actividade (Submeter candidatura)



Fonte: Elaborado pela autora, 2026.

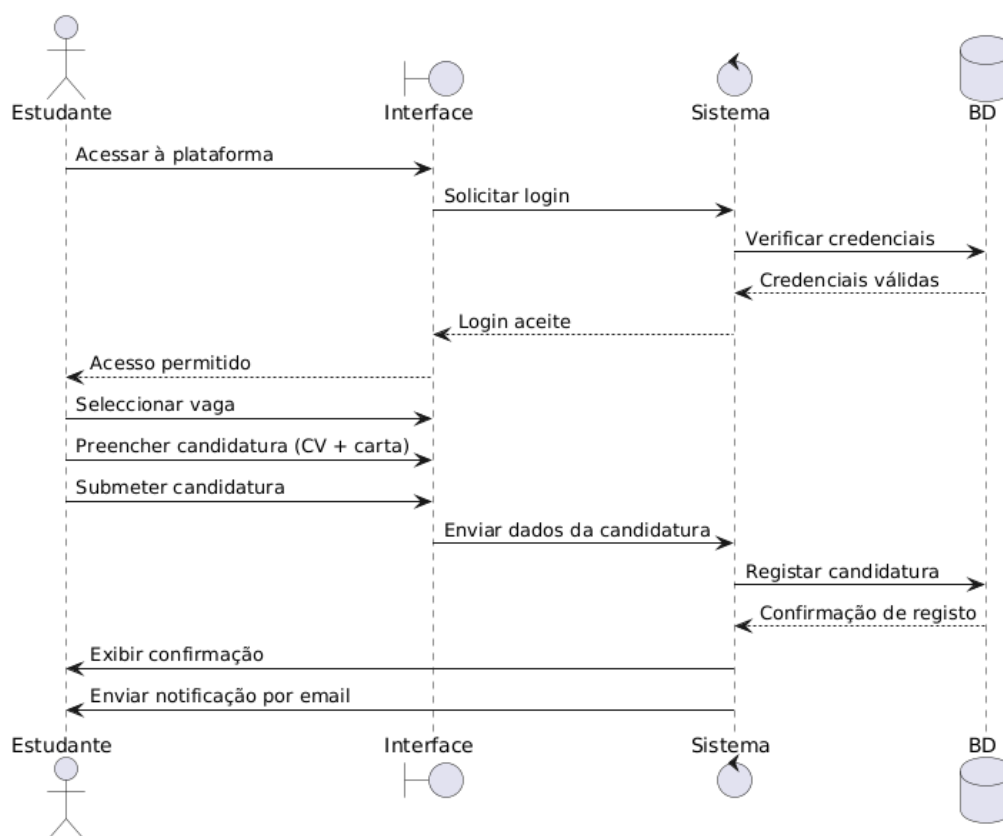
Figura 13 – Diagrama de actividade: este diagrama descreve de forma sequencial o fluxo de acções do estudante ao candidatar-se a uma vaga, desde o acesso e *login* até ao envio da candidatura, validação dos dados e notificação de sucesso ou erro, evidenciando os pontos de decisão.

2.10 Diagrama de sequência

É um diagrama UML que mostra a interacção entre objectos ao longo do tempo. Representa mensagens enviadas entre objectos, a ordem cronológica dessas interacções e a forma como as funcionalidades são executadas passo a passo. É útil para compreender a dinâmica de execução de um caso de uso.

Figura 14.

Diagrama de sequência (Submeter candidatura)



Fonte: Elaborado pela autora, 2026.

Figura 14 – Diagrama de sequência: este diagrama mostra a interacção dinâmica entre os elementos envolvidos (Estudante, *Interface*, Sistema e Base de Dados) durante o processo de *login* e submissão de candidatura, especificando a troca de mensagens entre os componentes e como o sistema responde ao estudante em cada etapa.

CONCLUSÃO

A plataforma concebida mostrou-se viável e pertinente, visto que reúne funcionalidades essenciais para aproximar estudantes e empresas de forma prática, acessível e transparente. A modelagem realizada com base em requisitos funcionais e não funcionais permitiu estruturar uma plataforma que responde às necessidades identificadas durante a investigação. Dessa forma, comprovou-se que a solução proposta contribui para a democratização do acesso às oportunidades de estágio e fortalece a relação entre ensino superior e mercado de trabalho.

Por fim, esta pesquisa reforça a importância da integração entre tecnologia e educação, demonstrando que iniciativas digitais podem desempenhar um papel crucial na promoção da empregabilidade juvenil em Angola. Apesar das limitações inerentes ao estudo, a plataforma criada abre caminho para futuras melhorias, parcerias institucionais mais abrangentes e a expansão para outras províncias. Dessa forma, o trabalho não apenas responde à situação-problemática, mas também se apresenta como um ponto de partida para inovações que poderão beneficiar toda a comunidade académica e empresarial.

REFERÊNCIAS BIBLIOGRÁFICAS

- Anderson, R. (2020). *Security engineering: A guide to building dependable distributed systems* (3rd ed.). Wiley.
- Atelier. (2016). *Micro-internships and guided professional learning*.
- Bass, L., Clements, P., & Kazman, R. (2013). *Software architecture in practice* (3rd ed.). Addison-Wesley.
- Beck, K. (2007). *Implementation patterns*. Addison-Wesley.
- Berners-Lee, T. (2007). *Weaving the Web: The original design and ultimate destiny of the World Wide Web*. HarperCollins.
- Brown, A. (2019). *Modern code editors and developer productivity*.
- Brown, T., & Smith, J. (2019). *Collaborative modeling with UML tools*.
- Connolly, T., & Begg, C. (2015). *Database systems: A practical approach to design, implementation, and management* (6th ed.). Pearson.
- Crockford, D. (2008). *JavaScript: The good parts*. O'Reilly Media.
- Date, C. J. (2004). *An introduction to database systems* (8th ed.). Pearson/Addison-Wesley.
- DuBois, P. (2013). *MySQL cookbook* (2nd ed.). O'Reilly Media.
- Educartis. (2018). *Educartis: Plataforma online de educação em África*. Wikipedia.
- Erl, T. (2005). *Service-oriented architecture: Concepts, technology, and design*. Prentice Hall.
- Expansão. (2025). *Estudantes enfrentam dificuldades na inserção profissional por falta de estágios*. Jornal Expansão.
- Finnegan, J. (2014). *Responsive web design with Bootstrap*.
- Flanagan, D. (2011). *JavaScript: The definitive guide* (6th ed.). O'Reilly Media.
- Flanagan, D. (2020). *JavaScript: The definitive guide* (7th ed.). O'Reilly Media.
- Fowler, M. (2002). *Patterns of enterprise application architecture*. Addison-Wesley.
- Fowler, M. (2010). *UML distilled: A brief guide to the standard object modeling language* (3rd ed.). Addison-Wesley.

- Freeman, E., & Freeman, E. (2004). *Head first design patterns*. O'Reilly Media.
- Gamma, E., Helm, R., Johnson, R., & Vlissides, J. (2007). *Design patterns: Elements of reusable object-oriented software*. Addison-Wesley.
- Google. (2019). *Google Summer of Code program evaluation*.
- Han, X., Li, Y., & Zhao, Q. (2017). *Lightweight development environments and code editors*.
- Hendler, J., Shadbolt, N., & Berners-Lee, T. (2008). *The Semantic Web revisited*. *IEEE Intelligent Systems*, 23(3), 61–64.
- Kajibanga, V. (2000). *O ensino superior em Angola: Desafios e perspectivas*. Luanda: Universidade Agostinho Neto.
- Krasner, G. E., & Pope, S. T. (2007). *A description of the model-view-controller user interface paradigm in the Smalltalk-80 system*. *Journal of Object-Oriented Programming*.
- Lam, C. (2010). *Setting up local web development environments*.
- Larman, C. (2004). *Applying UML and patterns: An introduction to object-oriented analysis and design and iterative development* (3rd ed.). Prentice Hall.
- Lerdorf, R. (2002). *Programming PHP*. O'Reilly Media.
- Lerdorf, R. (2010). *PHP internals and performance*. O'Reilly Media.
- Manita, K. (2024). *Juventude, empregabilidade e desafios da experiência profissional em Angola*. Luanda.
- Miguel, M. O. L. (2015). *Inserção profissional dos estudantes universitários em Angola: Perspectivas de estudantes e empregadores*. Luanda.
- O'Reilly, T. (2005). *What is Web 2.0: Design patterns and business models for the next generation of software*. O'Reilly Media.
- Paim, E. N. (2024). *Qualidade do ensino superior e exigências do mercado de trabalho em Angola*. Luanda.
- Pressman, R. S. (2011). *Software engineering: A practitioner's approach* (7th ed.). McGraw-Hill.

Pressman, R. S. (2016). *Engenharia de software: Uma abordagem profissional* (8.^a ed.). McGraw-Hill Education.

Programa das Nações Unidas para o Desenvolvimento & Organização Internacional do Trabalho. (s.d.). *Relatório sobre competências profissionais e aproximação universidade–sector produtivo na província da Huíla*. PNUD/OIT.

Ranjan, R. (2012). *Open-source web server environments for developers*.

Rees, M. (2018). *Extensible code editors and plugin ecosystems*.

República de Angola. (2023). *1.º Relatório do emprego em Angola*. Luanda: Ministério da Administração Pública, Trabalho e Segurança Social.

Resig, J. (2013). *Secrets of the JavaScript ninja* (2nd ed.). Manning Publications.

Roques, P. (2017). *UML in practice*. Wiley.

Silva, J. (2016). *Modelagem de bases de dados com MySQL Workbench*.

Silvers, A. (2015). *Front-end frameworks and scalable interfaces*.

Singh, R. (2014). *Cross-platform web development environments*.

Sommerville, I. (2011). *Software engineering* (9th ed.). Addison-Wesley.

Spurlock, J. (2013). *Bootstrap: Responsive web development*. O'Reilly Media.

Stallings, W. (2017). *Effective cybersecurity: A guide to using best practices and standards*. Addison-Wesley.

Stephens, R. (2012). *Beginning MySQL database design and optimization*. Wiley.

Tanenbaum, A. S., & Steen, M. van. (2007). *Distributed systems: Principles and paradigms* (2nd ed.). Pearson Prentice Hall.

Ullman, L. (2011). *PHP and MySQL for dynamic web sites* (4th ed.). Peachpit Press.

Welling, L., & Thomson, L. (2009). *PHP and MySQL web development* (4th ed.). Addison-Wesley.

<https://thiagomellodesign.wordpress.com/wp-content/uploads/2012/10/php.png>

https://static.vecteezy.com/system/resources/previews/027/127/463/non_2x/javascript-logo-javascript-icon-transparent-free-png.png

<https://cdn-icons-png.flaticon.com/512/5968/5968672.png>

https://images.icon-icons.com/1381/PNG/512/mysqlworkbench_93532.png

<https://icons.veryicon.com/png/o/business/vscode-program-item-icon/vscode.png>

<https://cdn.freebiesupply.com/logos/large/2x/xampp-logo-png-transparent.png>

https://www.nicepng.com/png/full/247-2477083_staruml-definition.png

https://artia.com/wp-content/uploads/2024/09/Imagens_Prancheta-1-01-780x780.png

APÊNDICES

Apêndice n.º 1 – Guião de observação.

Este guião de observação integra-se na investigação sobre o desenvolvimento de uma plataforma *web* que conecta estudantes universitários a empresas e oportunidades de estágio no contexto angolano. A observação adoptada é de natureza directa, fundamentada na vivência enquanto estudante universitária e no conhecimento prévio das dificuldades recorrentes associadas à procura e oferta de estágios.

A observação directa baseou-se no contacto quotidiano com outros estudantes, na análise empírica das experiências de candidatura a estágios, na procura de oportunidades e na interacção com os mecanismos tradicionais de divulgação existentes. Esta abordagem incluiu ainda a utilização prática dos meios disponíveis para acesso às informações sobre estágios, tais como murais informativos, redes sociais, contactos institucionais e plataformas digitais informais.

Título da pesquisa:

Desenvolvimento de uma Plataforma *Web* que Conecta Estudantes Universitários a Empresas e Oportunidades de Estágio no Contexto Angolano.

Tipo de observação:

Observação directa.

Objectivo geral:

Desenvolver uma Plataforma *Web* que Conecte Estudantes Universitários a Empresas e Oportunidades de Estágio no Contexto Angolano.

1. Identificação

Tabela 4.

Identificação da observação

Elemento	Descrição
Fontes de informação	Vivência académica da investigadora, interacção com estudantes universitários e observação directa dos processos de procura, divulgação e candidatura a oportunidades de estágio.
Período de observação	Junho a Julho de 2025
Local de acesso	Ambiente académico universitário, interacções estudantis, instituições de ensino superior, empresas e plataformas digitais utilizadas pelos alunos.
Observadora	Edvânia António Francisco

Fonte: Elaborado pela autora, 2026.

2. Aspectos a Observar

Tabela 5.

Aspectos a observar

Categoria	Elementos de observação	Indicadores / Perguntas de apoio
A. Divulgação de estágios	Existência de canais formais de divulgação de oportunidades de estágio.	Os estudantes sabem onde encontrar oportunidades de estágio? Existem canais institucionais claros?
B. Informação sobre as vagas	Conteúdo e clareza das informações disponibilizadas.	As informações incluem requisitos, prazos, áreas e entidades promotoras?

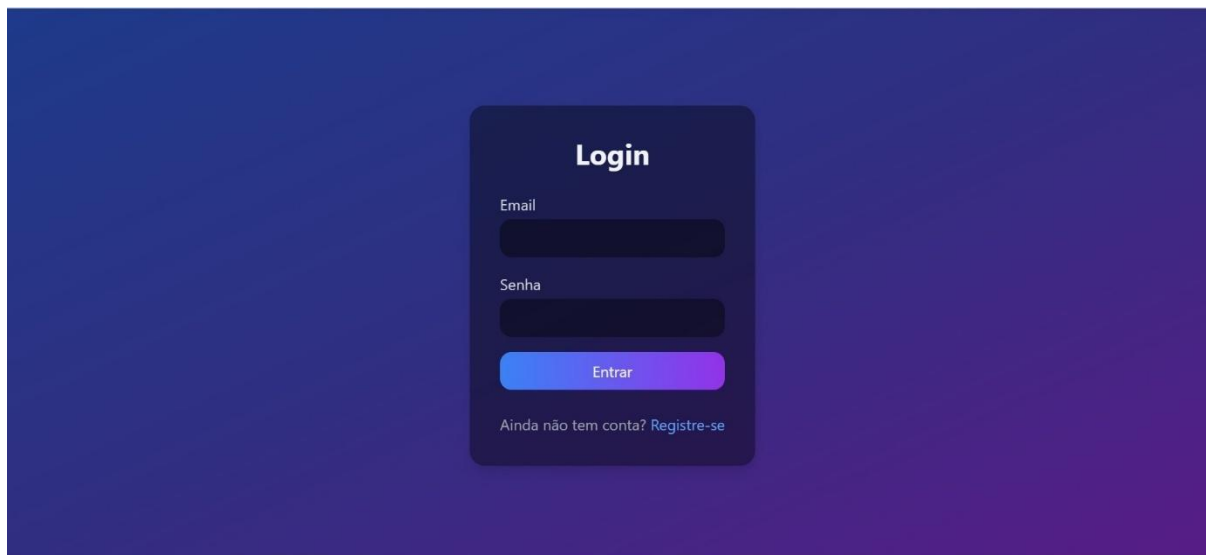
C. Processo de candidatura	Procedimentos exigidos para candidatura.	O processo de candidatura é simples ou complexo? Está bem explicado?
D. Comunicação	Interacção entre estudantes e entidades promotoras de estágio.	Há facilidade de contacto e resposta por parte das entidades?
E. Acesso à informação	Meios utilizados para obter informações sobre estágios.	Os estudantes precisam recorrer a contactos informais ou deslocações físicas?
F. Identificação de lacunas	Dificuldades e ausência de informações relevantes.	Quais informações são mais difíceis de obter? Que dados deveriam estar centralizados numa plataforma?

Fonte: Elaborado pela autora, 2026.

Apêndice n.º 2 – Algumas telas da plataforma *web*.

Figura 15.

Tela de login



Fonte: Elaborado pela autora, 2026.

Figura 16.

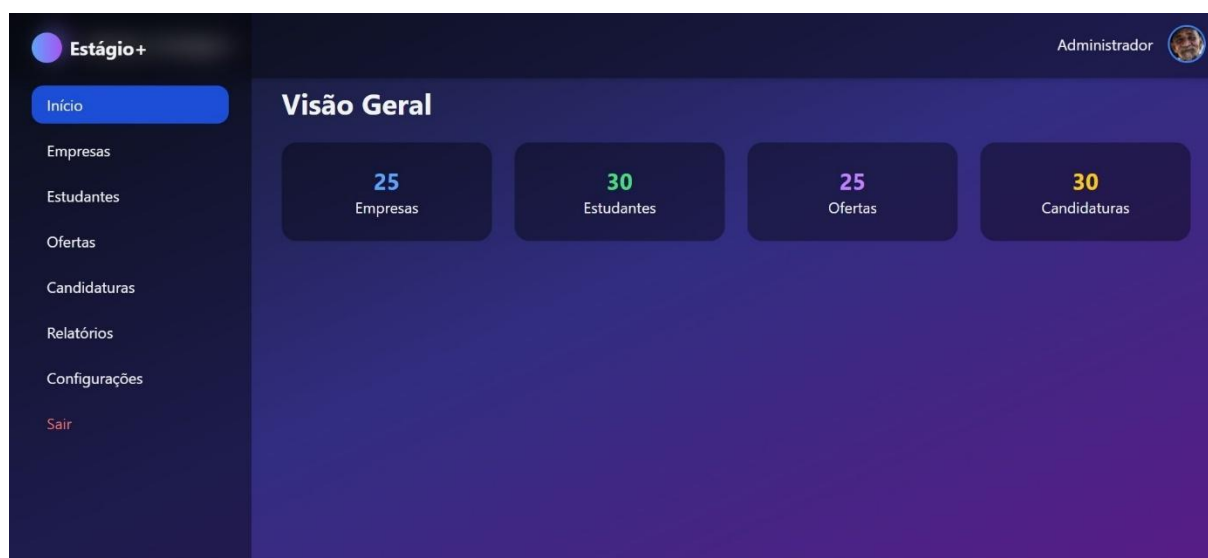
Home



Fonte: Elaborado pela autora, 2026.

Figura 17.

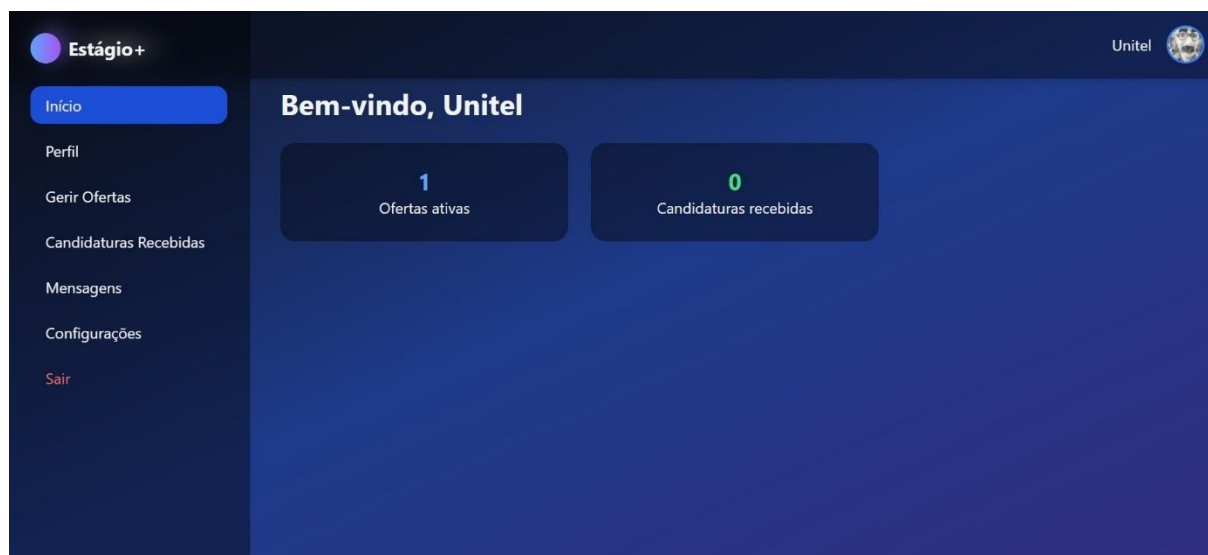
Painel do administrador



Fonte: Elaborado pela autora, 2026.

Figura 18.

Painel da empresa



Fonte: Elaborado pela autora, 2026.