

**UNIVERSIDADE PAULISTA**

**ICET - INSTITUTO DE CIÊNCIAS EXATAS E TECNOLOGIA**

**CURSO SUPERIOR DE TECNOLOGIA EM ANÁLISE E DESENVOLVIMENTO DE SISTEMAS**

**PROJETO INTEGRADO MULTIDISCIPLINAR  
PIM III**

**Sistema integrado para gestão de chamados e suporte técnico baseado em IA**

**Nome R.A**

ALUNO 1 - Andrei Henrique Mancijo G922CG4

ALUNO 2 - Filipe Vitor dos Santos R084353

ALUNO 3 - Jônatas dos Santos Souza G9038F8

ALUNO 4 - Kaique Batista da Silva G03IGG0

ALUNO 5 - Mariozan Damasceno Lacerda Júnior G9884G2

ALUNO 6 - Mateus Teodoro da Silva G9265G4

**SÃO JOSÉ DOS CAMPOS – SP**

**JUNHO/2025**

|  |  |
| --- | --- |
|  | **RA** |
| Aluno 1 - Andrei Henrique Mancijo | G922CG4 |
| Aluno 2 - Filipe Vitor dos Santos | R084353 |
| Aluno 3 - Jônatas dos Santos Souza | G9038F8 |
| Aluno 4 - Kaique Batista da Silva | G03IGG0 |
| Aluno 5 - Mariozan Damasceno Lacerda Júnior | G9884G2 |
| Aluno 6 - Mateus Teodoro da Silva | G9265G4 |

**Sistema integrado para gestão de chamados e suporte técnico baseado em IA**

Projeto Integrado Multidisciplinar (PIM) desenvolvido como exigência parcial dos requisitos obrigatórios à aprovação semestral no Curso Superior de Tecnologia em Análise e Desenvolvimento de Sistemas da UNIP (Universidade Paulista), orientado pelo corpo docente do curso.

**São José dos Campos – SP**

**JUNHO/2025RESUMO**

Este trabalho apresentou o desenvolvimento de um sistema integrado para gestão de chamados e suporte técnico, com base em Inteligência Artificial (IA), visando atender à crescente demanda por soluções tecnológicas que otimizem o atendimento técnico nas organizações. O objetivo principal foi criar uma ferramenta capaz de realizar a triagem inicial, categorização automática de chamados e sugestão de soluções, com o intuito de reduzir a sobrecarga da equipe de TI e agilizar o tempo de resposta aos usuários. A metodologia Scrum foi adotada para o gerenciamento das etapas do projeto, garantindo organização ágil e entrega contínua de resultados. O levantamento de requisitos foi conduzido em parceria com uma empresa de grande porte, permitindo identificar com precisão as necessidades reais do sistema. Como parte do desenvolvimento, elaborou-se um protótipo funcional com acesso administrativo e estruturação de banco de dados em MS SQL Server, abrangendo informações de usuários, equipe de TI, supervisores e triagem. A IA foi integrada ao sistema para atuar como chatbot na triagem inicial e como ferramenta de apoio na análise e encaminhamento de chamados. Foram aplicadas diretrizes da Lei Geral de Proteção de Dados (LGPD) para garantir segurança e conformidade no tratamento dos dados pessoais. A expectativa é que a solução implementada contribua para a redução na quantidade de chamados pendentes, diminua o tempo de espera e potencialmente aumente a satisfação dos usuários, indicando um possível impacto positivo.

**Palavras-chave:** Suporte técnico, Inteligência Artificial, gestão de chamados, Scrum, LGPD, banco de dados, chatbot.

**SUMÁRIO**

|  |  |
| --- | --- |
|  | **Pág.** |
| 1. INTRODUÇÃO | 5 |
| 2. CARACTERIZAÇÃO DA EMPRESA (SUPORTE TÉCNICO) |  |
| 3. DESENVOLVIMENTO DO PROJETO |  |
| 4. CONSIDERAÇÕES FINAIS |  |
| 5. REFERÊNCIAS |  |

**1. INTRODUÇÃO**

**OBJETIVO GERAL**

Realizar o levantamento e análise de requisitos para a criação de um sistema de suporte técnico inteligente, que utilize Inteligência Artificial para triagem inicial, categorização automática de chamados e sugestão de soluções, visando otimizar o tempo de resposta e reduzir a sobrecarga da equipe de TI. A LGPD deve ser aplicada a todos os dados pessoais tratados no sistema. **O desenvolvimento ocorrerá no próximo semestre (PIM IV)**.

**Objetivos Específicos**

Com o propósito de atingir o objetivo geral proposto, serão considerados os seguintes objetivos específicos:

* Aplicar metodologias de levantamento de requisitos e modelagem de dados para um sistema de suporte técnico.
* Desenvolver artefatos UML, incluindo diagramas de caso de uso, classe e sequência.
* Criar protótipos de interface gráfica para desktop, web e mobile, garantindo acessibilidade e usabilidade.
* Estruturar o banco de dados com MS SQL Server e definir os principais relacionamentos entre tabelas.
* Avaliar as melhores práticas para integração com IA para classificação automática de chamados.
* Desenvolver estratégias para garantir conformidade com a LGPD na manipulação de dados dos usuários.

1. Engenharia de Software II

Os testes unitários são uma prática de verificação do software em que partes isoladas do código, como funções, métodos ou classes, são testadas separadamente para garantir que cada uma funciona corretamente (conforme o esperado). O objetivo é detectar os erros mais cedo, nas fases iniciais do desenvolvimento, para que a manutenção do sistema seja mais fácil e segura (SOMMERVILLE, 2011).

Conforme Jorgensen (2013), os testes unitários são normalmente realizados pelos desenvolvedores durante a codificação e fazem parte das práticas de desenvolvimento orientadas a testes (TDD – Test-Driven Development).

Já o **caso de uso descritivo** é uma técnica de modelagem de requisitos que detalha, em linguagem natural estruturada, os fluxos de interação entre o usuário e o sistema. Ele serve como base para o entendimento funcional do sistema, ajudando no alinhamento entre desenvolvedores e stakeholders (PRESSMAN; MAXIM, 2016).

A **documentação de interface** é outro aspecto essencial, especialmente quando aliada aos princípios de **usabilidade**. Uma interface bem projetada deve ser intuitiva, eficiente e acessível, proporcionando uma boa experiência ao usuário. A usabilidade é avaliada com base em critérios como consistência visual, feedback claro e facilidade de navegação (NIELSEN, 2000).

Esses três pilares — teste unitário, modelagem por caso de uso e usabilidade na interface — formam a base de um desenvolvimento de software centrado na qualidade, contribuindo para sistemas mais robustos, funcionais e adequados às necessidades dos usuários.

1. Análise de Sistemas Orientada a Objetos

A Análise de Sistemas Orientada a Objetos (ASOO) é essencial na engenharia de software de hoje, principalmente para fazer sistemas fortes, escaláveis e que respondam às necessidades do usuário (SOMMERVILLE, 2019). A ASOO procura por objetos, modelando entidades do mundo real no sistema, melhorando a ligação entre o problema e a solução de informática.

Esta técnica usa os conceitos de Programação Orientada a Objetos (POO), como encapsulamento, herança e polimorfismo, em todas as etapas do desenvolvimento do software (PRESSMAN, 2016). Os analistas usam a linguagem UML (Unified Modeling Language) para mostrar casos de uso, classes, interações e estruturas do sistema (FOWLER, 2011). Os diagramas mais comuns incluem o diagrama de casos de uso, que define as funções esperadas e quem as usa; o diagrama de classes, que mostra a estrutura do sistema; e o diagrama de sequência, que explica como os objetos se comportam (BOOCH; RUMBAUGH; JACOBSON, 2006).

Com a ASOO, o desenvolvimento se torna mais organizado e claro. As etapas de levantamento de requisitos, modelagem e validação simplificam-se para todos os envolvidos, mesmo aqueles que não são da área técnica. Desse jeito a comunicação entre analistas, desenvolvedores e usuários finais fica mais suave, garantindo que o produto final supere as expectativas do cliente (LARMAN, 2007).

Dentro do contexto do Projeto Integrado Multidisciplinar (PIM III), a aplicação da ASOO é chave para estruturar o sistema de suporte técnico com IA. Utilizando modelagem orientada a objetos, o grupo consegue mapear os processos da empresa simulada, identificando entidades tipo chamados, usuários, técnicos e categorias, definindo suas responsabilidades e interações. Essa organização lógica ajuda a implementar o sistema depois com maior eficiência e menos chances de falhas ou retrabalho.

Outra vantagem da ASOO é sua adaptação com metodologias ágeis e arquiteturas atuais como orientação a serviços (SOA) e microsserviços (SOMMERVILLE, 2019). Focando em objetos e seus comportamentos, o sistema pode ser dividido em módulos independentes, o que simplifica a manutenção e evolução constantes.

A modelagem orientada a objetos possibilita o reaproveitamento eficaz de componentes de software, o que ajuda a baixar os custos e acelera o tempo de criação (PRESSMAN, 2016). A colaboração com ferramentas CASE, de Engenharia de Software Auxiliada por Computador, automatiza uma parte do processo, por exemplo, gerando código ou um banco de dados com base nos modelos.

Enfim, a Análise de Sistemas Orientada a Objetos é crucial para o desenvolvimento com jeito de soluções de tecnologia, igual ao sistema de suporte técnico sugerido. Integrando as bases sólidas da engenharia de software e oferecendo uma visão geral das funcionalidades, ela assegura qualidade, escalabilidade e, ainda, que o projeto case com as necessidades do negócio.

1. Programação Orientada a Objetos I

A Programação Orientada a Objetos (POO) é um paradigma de programação que organiza o software a partir de entidades chamadas objetos, que representam elementos do mundo real ou conceitual. Esses objetos são definidos por meio de classes, que funcionam como moldes.

Segundo Forbellone e Eberspächer (2014), a POO visa tornar o código mais reutilizável, modular e de fácil manutenção, sendo amplamente adotada em linguagens como C# e Java.

1. Classe

A classe é a estrutura que define os atributos (características) e métodos (comportamentos) de um objeto. É a base para a criação de objetos, funcionando como um modelo.

Fonte: Forbellone, A. L. V., & Eberspächer, H. F. – Programação Orientada a Objetos com C#.

2. Objeto

O objeto é a instância concreta de uma classe. Ele possui estados e comportamentos definidos pela classe de origem. Em outras palavras, é quando aquele modelo (classe) é colocado em uso no programa.

Fonte: Microsoft Learn – Introdução à POO em C#.

3. Encapsulamento

Encapsulamento significa proteger os dados de um objeto, restringindo o acesso direto a seus atributos e permitindo a interação por meio de métodos. Isso melhora a segurança e integridade dos dados.

Fonte: Curso Alura – Programação Orientada a Objetos com C#.

4. Herança

A herança é um mecanismo que permite que uma classe herde características e comportamentos de outra. Com isso, é possível reutilizar código e criar hierarquias entre objetos.

Fonte: Microsoft Learn – POO em C#.

5. Polimorfismo

Polimorfismo refere-se à capacidade de objetos diferentes responderem de formas distintas à mesma ação. Ele permite modificar ou estender comportamentos sem alterar o código da superclasse.

Fonte: Curso DIO – Formação Fundamentos do .NET com C#.

6. Abstração

A abstração consiste em esconder detalhes internos e mostrar apenas as informações relevantes. Na prática, ela permite que o desenvolvedor trabalhe com interfaces e conceitos genéricos, focando no “o que” e não no “como”.

Fonte: Forbellone, A. L. V., & Eberspächer, H. F. – Programação Orientada a Objetos com C#.

7. Associação, Agregação e Composição

Esses três conceitos representam formas de relacionamento entre classes:

• Associação: vínculo genérico entre dois objetos.

• Agregação: um objeto faz parte de outro, mas pode existir independentemente.

• Composição: um objeto é parte essencial do outro e não existe separadamente.

Esses conceitos são úteis para modelar a estrutura e o comportamento do sistema de forma mais próxima da realidade.

Fonte: Curso Alura – POO com C#.

1. Projeto de Interface com o Usuário

Porque elementos de usabilidade devem ser garantidos em sistemas profissionais, como um sistema de gerenciamento de chamados, por exemplo?

Em um sistema de gerenciamento de chamado, é de interesse que a interface e a interação do usuário sejam fluidas e orgânicas, já que os usuários que solicitam suporte já estão em situação de estresse e os atendentes destes, deve utilizar o sistema de forma eficiente para evitar perda de tempo e minimizar possíveis danos à corporação. “O design atraente faz as pessoas se sentirem bem, o que por sua vez as faz pensar mais criativamente. Um estado emocional positivo afeta a maneira como o cérebro funciona.” (Norman, 2004)

— Norman, D. A. (2004). Emotional Design: Why We Love (or Hate) Everyday Things. Basic Books.

Ao desenvolver uma interface para o usuário, deve-se considerar importantes conceitos como a experiência do usuário (UX), arquitetura da informação (AI), interface do usuário (UI) e design de interação (IxD) a fim de desenvolver uma interface amigável e receptiva ao usuário, atendendo as necessidades do cliente e desenvolvendo um produto de maior qualidade.

A fim de criar uma interação receptiva e amigável, é de suma importância garantir que a interface do sistema seja fácil de ser utilizada. Uma interface bem elaborada, fácil e receptiva pode trazer maior desempenho do usuário, bem-estar e contribuir para o humor, já que a interface é a principal apresentação do software e onde o usuário irá trabalhar. Logo, uma interface mal elaborada, pode trazer atrasos, perda de tempo, serviço, contribuir com o estresse do colaborador e consequentemente, prejuízos à empresa.

Os usuários passam a maior parte do tempo em outros sites, então eles esperam que seu site funcione como todos os outros sites que eles já conhecem. (Nielsen, ano).

Usabilidade na web Projetando Websites com Qualidade- Jakob Nielsen Hoa Loranger.

Para desenvolver uma interface com boa usabilidade, deve-se garantir que o usuário se familiarize, se sinta seguro e produtivo. Garantindo os princípios citados por Jakob Nielsen e Donald A. Norman, pode-se trazer economias à empresa, pois terá custo menor com treinamentos e a produtividade dos colaboradores será ampliada.

1. Banco de Dados.

O Banco de Dados é um componente essencial na construção de sistemas de informação, permitindo o armazenamento, organização e recuperação eficiente de dados. Em projetos modernos, especialmente aqueles que integram inteligência artificial (IA), uma modelagem bem estruturada do banco é crucial para garantir consistência e escalabilidade.

A modelagem de dados inicia-se com a criação do Diagrama Entidade-Relacionamento (ER), que representa visualmente entidades, atributos e os relacionamentos entre elas. Em um sistema de gestão de chamados com suporte técnico por IA, é fundamental identificar corretamente elementos como:

* Usuários
* Técnicos
* Chamados
* Categorias de problemas
* Interações
* Sugestões automatizadas

Esse planejamento lógico permite compreender melhor o funcionamento do sistema, garantindo uma estrutura flexível e preparada para expansão, além de facilitar a normalização dos dados, evitando redundâncias e garantindo integridade.

## Aplicação Prática

Durante a fase de testes do sistema, os scripts de carga de dados são utilizados para inserir dados fictícios, simulando cenários reais. Isso permite testar consultas SQL, avaliar a resposta da IA e validar funcionalidades do sistema. Em soluções com IA, a preparação de dados históricos bem estruturados é essencial para que os algoritmos identifiquem padrões e façam sugestões relevantes.

## Considerações Finais

A combinação de uma modelagem ER bem definida com dados de teste representativos impacta diretamente a qualidade e eficiência do sistema. No contexto de chamados técnicos automatizados por IA, a estrutura do banco de dados é vital para garantir o desempenho e a confiabilidade da aplicação.

1. Economia e Mercado

O sistema integrado para gestão de chamados e suporte técnico com inteligência artificial foi desenvolvido para atender empresas que lidam com um grande volume de solicitações, tanto de funcionários quanto de clientes. Muitas vezes, essas empresas enfrentam dificuldades para organizar os atendimentos, o que pode resultar na sobrecarga das equipes técnicas. A proposta deste sistema é justamente otimizar esse processo, tornando o atendimento mais rápido, organizado e eficiente.

Atualmente, existem diversas ferramentas de help desk disponíveis no mercado. No entanto, poucas oferecem recursos de inteligência artificial de forma acessível e personalizável. Segundo um levantamento da Unite.AI, soluções de help desk com IA podem reduzir custos operacionais em até 30% e melhorar o tempo de resposta em 52% [10 melhores softwares de suporte ao cliente de IA com recursos de help desk (2025) - Unite.AI](https://www.unite.ai/pt/10-melhores-softwares-de-suporte-ao-cliente-de-IA-com-recursos-de-help-desk/). O diferencial da solução proposta neste projeto é unir automação inteligente com praticidade, oferecendo funcionalidades como sugestões automáticas para resolver chamados e o registro detalhado do histórico de cada atendimento. [As 6 melhores soluções de software de helpdesk com IA para pequenas empresas em 2024](https://pt.glassix.com/article/best-ai-helpdesk-software)

Para avaliar a viabilidade da ideia, foram considerados alguns custos básicos, como o tempo necessário para desenvolver o sistema, os recursos de hospedagem e banco de dados, e eventuais integrações com ferramentas de IA. De acordo com a RapidCanvas, os principais elementos de custo em projetos de IA incluem aquisição de talentos, infraestrutura computacional e manutenção contínua [Understanding The Cost Of AI Projects](https://pt.rapidcanvas.ai/guides/understanding-the-cost-of-ai-projects). O retorno esperado vem da economia de tempo da equipe técnica, da melhoria no tempo de resposta e da satisfação dos usuários.

Pensando na realidade de pequenas e médias empresas, o projeto se mostra viável tanto do ponto de vista técnico quanto financeiro. Estudos indicam que a automação inteligente pode aumentar a eficiência operacional e reduzir erros, permitindo que pequenas empresas ofereçam um atendimento mais ágil e competitivo [Automação inteligente: 5 benefícios para empresas](https://santodigital.com.br/automacao-inteligente/). A estimativa é que, em cerca de seis meses de uso, os ganhos em produtividade e organização já compensem os investimentos iniciais, trazendo benefícios concretos para o dia a dia da operação.

1. Gestão Estratégica de Recursos Humanos.

A Gestão Estratégica de Recursos Humanos (GERH) é um pilar fundamental para o sucesso organizacional, pois alinha as práticas de pessoas com os objetivos estratégicos da empresa. Segundo Chiavenato (2014), a GERH tem como foco desenvolver competências, engajamento e desempenho alinhado aos resultados desejados pela organização.

No contexto do projeto UpDesk, a GERH se manifesta principalmente na organização do atendimento ao cliente interno ou externo, promovendo eficiência, rastreabilidade e valorização do capital humano. Com a implementação do sistema, é possível melhorar a alocação de chamados, priorizar tarefas por nível de complexidade (N1 e N2), monitorar o desempenho individual da equipe de suporte e permitir ações corretivas baseadas em dados reais.

A adoção de inteligência artificial (IA) também contribui com a automação de demandas repetitivas, como classificação de chamados, o que libera os colaboradores para focarem em tarefas mais estratégicas, promovendo um ambiente mais produtivo e menos operacional (MARRAS, 2017).

Além disso, o registro de históricos de chamados, soluções aplicadas e tempo de resposta se torna uma base confiável para avaliação de desempenho, planejamento de treinamentos e reconhecimento de talentos internos.

Dessa forma, o projeto UpDesk vai além da tecnologia: ele se posiciona como uma ferramenta de apoio à gestão de pessoas, contribuindo diretamente para a motivação, produtividade e alinhamento dos times com os objetivos organizacionais.

**CARACTERIZAÇÃO DA EMPRESA (SUPORTE TÉCNICO)**

A empresa em questão é um centro de suporte técnico e gerenciamento de chamados, onde atua internamente dentro de empresas que contratam seu serviço. Basicamente uma empresa terceira que oferece/ suporte de TI.

**1. Objetivo do Sistema Atual**

O sistema é responsável pela gestão de registros de chamados, divididos em três categorias principais:

- Incidentes

- Requerimentos

- Mudanças

Além disso, possui gestão de conhecimento (Base de Conhecimento), permitindo que os times tenham acesso à documentação para auxiliar na resolução das solicitações.

**2. Tipos de Chamados mais comuns**

As solicitações recorrentes envolvem:

- Troca/desbloqueio de senhas

- Instalação de softwares

- Problemas/erros de hardware e software

**3. Registro e Acompanhamento dos Chamados**

Os chamados são registrados e monitorados através da ferramenta ITSM, sendo gerenciados e tratados pelos times responsáveis de forma individualizada, ou seja, cada analista que compõe o time puxa um ou mais chamados para tratativa.

**4. Informações Essenciais no Registro de Chamados**

Cada chamado deve conter:

- Dados do usuário e contato

- Informações do equipamento envolvido

- Localização geográfica

- Descrição detalhada do problema ou solicitação

- Urgência do caso

- Categorização do chamado

- OLA (Operational Level Agreement)

- Dados do time responsável pela tratativa

**5. Fluxo de Atendimento**

O fluxo pode variar conforme o nível de atuação, mas segue as seguintes etapas:

- Análise inicial: Verificação das informações e entendimento do problema.

- Busca na base de conhecimento: Consulta de documentação relacionada.

- Aplicação do processo: Tratamento conforme conhecimento técnico.

- Validação da solução: Teste de eficácia da ação realizada.

- Confirmação do usuário: Encerramento do chamado após validação.

**6. Definição do Nível de Urgência**

O grau de urgência é categorizado como:

- Alta – Necessidade de ação imediata (exemplo: falha crítica).

- Média – Pode aguardar um pouco, mas deve ser resolvida em prazo adequado.

- Baixa – Pode ser resolvida conforme disponibilidade (exemplo: melhorias).

**9. DESENVOLVIMENTO DO PROJETO**

Com base na fundamentação teórica das disciplinas que compõem o semestre (cursadas no modo presencial e no AVA), deverão ser elaboradas propostas para cada uma das situações específicas de cada disciplina, procurando sempre justificar a adoção da solução proposta, por meio da discussão de suas vantagens, viabilidade econômica e/ou disponibilidade da tecnologia.

Nesse capítulo, o grupo deverá apresentar o projeto como um todo, destacando suas virtudes, eventuais limitações e suas perspectivas de aperfeiçoamento.

Aqui o(s) avaliador(es) deve(m) “enxergar” (identificar) o atendimento aos requisitos obrigatórios do PIM III em cada disciplina, além daqueles estabelecidos pelos professores.

**10. CONSIDERAÇÕES FINAIS**

O desenvolvimento do sistema **UpDesk** representou uma oportunidade concreta de aplicar os conhecimentos adquiridos ao longo do curso de Análise e Desenvolvimento de Sistemas. A proposta de integrar funcionalidades tradicionais de gerenciamento de chamados com recursos de **inteligência artificial** ampliou o escopo técnico do projeto e demonstrou a viabilidade de soluções inteligentes no contexto de atendimento e suporte técnico.

Durante a elaboração do projeto, foram realizados o **levantamento de requisitos com uma empresa real**, a criação de protótipos no Figma, modelagem de banco de dados, definição de casos de uso, diagramas de classe com agregações e composições, e planejamento dos relatórios de gestão. Cada etapa foi essencial para transformar a ideia inicial em um sistema funcional e estrategicamente estruturado.

Além dos ganhos técnicos, o projeto contribuiu para a **formação prática dos envolvidos**, fortalecendo a compreensão sobre o ciclo de vida do desenvolvimento de software, gestão de requisitos, modelagem UML e boas práticas em projetos de extensão universitária. A aplicação da IA, mesmo que em estágio inicial, mostrou o potencial de automação e apoio à decisão nos processos organizacionais.

Conclui-se, portanto, que o projeto UpDesk atingiu seus objetivos pedagógicos e técnicos, apresentando-se como uma solução viável, escalável e alinhada às necessidades atuais de empresas que buscam agilidade e eficiência em seus processos de atendimento.

**11. REFERÊNCIAS**

**(Exemplos)**

ALVES, L. **Educação a distância: conceitos e história no Brasil e no mundo**. Revista Brasileira de Aprendizagem Aberta e a Distância, v. 10, 2011. Disponível em: <http[://www.abed.org.br/revistacientifica/Revista\_PDF\_Doc/2011/Artigo\_07.pdf](HTTP://www.abed.org.br/revistacientifica/Revista_PDF_Doc/2011/Artigo_07.pdf)>. Acesso em: 05 fev. 2017.

ALVES, R. M.; ZAMBALDE, A. L.; FIGUEIREDO, C. X. **Ensino a distância**. Lavras: UFLA/FAEPE, 2004.

ARBACHE, F. S., SANTOS, A. G., MONTENEGRO, C., SALLES, W. F. **Gestão de logística, distribuição e trade marketing**. Rio de Janeiro: Editora FGV, 2014. 180p.

CARVALHO, D. T. de; NEVES, M. F. (Org.). **Marketing na nova economia**. São Paulo: Atlas, 2001.

COSTA, K. S.; FARIA, G. G. **EAD: sua origem histórica, evolução e atualidade brasileira face ao paradigma da educação presencial**. Disponível em: [<http://www.abed.org.br/congresso2008/tc/552008104927AM.pdf](file:///D:\ROBERTO\UNIP\POS_EAD\MONOGRAFIA\%3chttp:\www.abed.org.br\congresso2008\tc\552008104927AM.pdf)>. Acesso em: 05 fev. 2017.

FARIA, M. A.; SILVA, R. C. S. **EAD: o professor e a inovação tecnológica**. Revista Brasileira de Aprendizagem Aberta e a Distância. São Paulo, dez. 2007.

GRAVES, P. **Por dentro da mente do consumidor**. Rio de Janeiro: Elsevier, 2011.

GRONROOS, C. **Marketing: gerenciamento e serviços**. 2ª ed. Rio de Janeiro: Elsevier, 2003.

HOOLEY, G.; PIERCY, N. F.; NICOULAUD, B. **Estratégia de marketing e posicionamento competitivo**. 4. ed. São Paulo: Pearson Prentice Hall, 2011.

KOTLER, P. **Administração de marketing: análise, planejamento, implementação e controle**. São Paulo: Atlas, 1994.

KOTLER, P., ARMSTRONG, G. **Introdução ao marketing**. Rio de Janeiro: LTC – Livros Técnicos e Científicos Editora SA, 2000. 371p.

KOTLER, P. **Marketing para o século XXI: como criar, conquistar e dominar mercados**. São Paulo: Futura, 2001.

KOTLER, P., TRIAS de BES, F. **Marketing lateral – Uma abordagem revolucionária para criar novas oportunidades em mercados saturados**. Rio de Janeiro: Elsevier, 2004.

LAMB, C. W. Jr., HAIR, J. F. Jr., MCDANIEL, C. **Princípios de marketing**. São Paulo: Pioneira Thomson Learning, 2004. 644p.

LAS CASAS, A. L. **Marketing de serviços**. 6ª ed. São Paulo: Atlas, 2012.

MAZZA, M. F. **CRM: sucessos & insucessos**. Rio de Janeiro: Brasport, 2009. 226p.

MORAN, J. M. **O que e a educação a distância**. Atualizado em 2002. Disponível em :<http://www.eca.usp.br/prof/moran/dist.html>. Acesso em 05 fev. 2017.

SARAIVA, T. **Educação a distância no Brasil: lições da história**. Em Aberto, Brasília, ano 16, n. 70, abr./jun. 1996.

YANAZE, M. H. **Gestão de marketing e comunicação – avanços e aplicações**. 2ª ed. São Paulo: Saraiva, 2011.

### FICHA DE CONTROLE DO PIM

Ano: 2025 Período: 2°/3° Coordenador: Prof Roberto Cordeiro Waltz

Tema (Identificação da empresa de suporte técnico):\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Alunos

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| RA | Nome | E-mail | Curso | Visto do aluno |
| **G922CG4** | **Andrei Henrique Mancijo** | **andrei.mancijo@aluno.unip.br** | **CST em ADS** |  |
| **R084353** | **Filipe Vitor dos Santos** | **filipevitor6@gmail.com** | **CST em ADS** |  |
| **G9038F8** | **Jônatas dos Santos Souza** | **jonatas.souza20@aluno.unip.br** | **CST em ADS** |  |
| **G03IGG0** | **Kaique Batista da Silva** | **kaique.silva107@aluno.unip.br** | **CST em ADS** |  |
| **G9884G2** | **Mariozan Damasceno Lacerda Júnior** | **mariozanjunior15@gmail.com** | **CST em ADS** |  |
| **G9265G4** | **Mateus Teodoro da Silva** | **mateus.silva329@aluno.unip.br** | **CST em ADS** |  |

Registros

|  |  |
| --- | --- |
| Data do encontro | Observações |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |