Nome da Empresa: MoveExpress

Ramo da Empresa: Transporte e Logística

Link GitHub:
https://github.com/brendonuehara/Projeto-Planejamento_em_informatica

Equipe Responsável: Brendon Minoru Uehara – 2222103543

Curso: Tecnologia em Análise e Desenvolvimento de Sistemas

Turma: 40

Semestre: 1

Ano: 2024

ÍNDICE

ESCOPO DO PROJETO

O presente documento detalha o escopo do projeto proposto pela empresa MoveExpress. O projeto tem como objetivo principal a implementação de um sistema de roteirização avançado e aprimoramento da gestão de frota, visando otimizar a logística de entrega de mercadorias. Este sistema será integrado com tecnologias de geolocalização em tempo real, permitindo uma alocação mais eficiente de recursos e uma redução significativa nos tempos de trânsito.

Além disso, o projeto busca modernizar os processos de comunicação e rastreamento, proporcionando aos clientes uma maior transparência e controle sobre suas encomendas. Ao centralizar e automatizar as operações logísticas, espera-se reduzir os custos operacionais e minimizar possíveis falhas na entrega.

Dessa forma, a MoveExpress visa não apenas melhorar a competitividade no mercado de logística e transporte, mas também garantir a satisfação e fidelidade dos clientes, oferecendo um serviço de entrega confiável, ágil e adaptado às necessidades do mercado contemporâneo.

SERVIÇOS

A empresa MoveExpress oferecerá os seguintes serviços:

Roteirização avançada: Implementação de um sistema inteligente de roteirização que permite a otimização das rotas de entrega, levando em consideração variáveis como tráfego, condições climáticas e prioridades de entrega. Isso resultará em entregas mais rápidas e eficientes, reduzindo custos operacionais e aumentando a satisfação do cliente.

Gestão de frota integrada: Desenvolvimento de uma plataforma centralizada para monitoramento e controle da frota de veículos, permitindo uma gestão eficiente de manutenção, abastecimento e alocação de recursos. Com essa solução, a empresa poderá maximizar o uso dos seus veículos, reduzindo o tempo de inatividade e garantindo a disponibilidade necessária para atender à demanda crescente.

Além disso, a MoveExpress disponibilizará o seguinte produto:

Sistema de rastreamento em tempo real: Oferta de uma solução de rastreamento que permite aos clientes acompanharem o status de suas encomendas em tempo real, desde a coleta até a entrega final, proporcionando maior segurança e confiança no processo de entrega. Com esse sistema, os clientes poderão receber notificações automáticas sobre o progresso de suas entregas, melhorando a comunicação e a experiência do cliente.

APRENDIZADO DE MÁQUINA

Nesse tópico será feito três entregas, onde temos a Exploração de dados e pré-processamento, Implementação de Modelos de Aprendizado de Máquina e Otimização e validação do modelo.

Entrega 1 – Exploração de dados e pré-processamento

```
from scipy import stats
from sklearn.metrics import confusion_matrix
import matplotlib.pyplot as plt
import seaborn as sns

# Extraindo dados do arquivo csv
import pandas as pd

tabela = pd.read_csv("CSV/dados_usuarios.csv", sep=",")
print(tabela)

# Tratamentos dos dados

dados_limpos = tabela.dropna()

dados_limpos = tabela.dropna(axis=1)

z_scores = stats.zscore(tabela['valor_pago'])

outliers = tabela[(z_scores > 3) | (z_scores < -3)]

dados_inconsistentes = tabela[tabela['idade'] < 0]
dados_inconsistentes = tabela[tabela['valor_pago'] < 0]

tabela.loc[tabela['idade'] < 0, 'idade'] = 0
tabela.loc[tabela['valor_pago'] < 0, 'valor_pago'] = 0

tabela['valor_pago'] = tabela['valor_pago'].apply(lambda x: x * 1000 if
x < 1000 else x)

# Matriz Confusão
```

```
y_true = [1, 0, 1, 1, 0, 1]

y_pred = [1, 0, 1, 0, 0, 1]

matriz_confusao = confusion_matrix(y_true, y_pred)

plt.figure(figsize=(6, 4))
sns.heatmap(matriz_confusao, annot=True, fmt='d', cmap='Blues',
cbar=False)
plt.xlabel('Rótulos Previstos')
plt.ylabel('Rótulos Verdadeiros')
plt.title('Matriz de Confusão')
plt.show()
```

Entrega 2 – Implementação de Modelos de Aprendizado de Máquina

```
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.ensemble import RandomForestClassifier
from sklearn.metrics import accuracy_score
from sklearn.metrics import recall_score
from sklearn.metrics import f1_score



dados = pd.read_csv("CSV/dados_usuarios.csv", sep=",")

X = dados.drop(columns=["status_pedido", "data_pedido",
"codigo_rastreio"])
y = dados["status_pedido"]

X = pd.get_dummies(X)

X_train, X_test, y_train, y_test = train_test_split(X, y,
test_size=0.2, random_state=42)



#Escolha do algoritmo de classificação
modelo = RandomForestClassifier(random_state=42)

#Treino do modelo
```

```
modelo.fit(X_train, y_train)

#Avaliação da precisão, recall e F1-Score
previsoes = modelo.predict(X_test)


precisao = accuracy_score(y_test, previsoes)
print("Precisão do modelo:", precisao)


recall = recall_score(y_test, previsoes, average='macro')

print("Recall do modelo:", recall)


f1 = f1_score(y_test, previsoes, average='macro')

print("F1-score (macro):", f1)
```

Entrega 3 – Otimização e Validação do Modelo

```
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.ensemble import RandomForestClassifier
from sklearn.model_selection import GridSearchCV
from sklearn.model_selection import cross_val_score

dados = pd.read_csv("CSV/dados_usuarios.csv", sep=",")

X = dados.drop(columns=['status_pedido'])
y = dados['status_pedido']

X = pd.get_dummies(X)

X_train, X_test, y_train, y_test = train_test_split(X, y,
test_size=0.2, random_state=42)

modelo = RandomForestClassifier()

# Definir os hiperparâmetros para busca
parametros = {
    'n_estimators': [50, 100, 200],
```

```
    'max_depth': [None, 10, 20],
    'min_samples_split': [2, 5, 10]
}

grid_search = GridSearchCV(estimator=modelo, param_grid=parametros,
cv=5, scoring='accuracy')

grid_search.fit(X_train, y_train)

print("Melhores parâmetros:", grid_search.best_params_)



# Definindo o modelo com os melhores hiperparâmetros encontrados
modelo = RandomForestClassifier(max_depth=None, min_samples_split=2,
n_estimators=200)

# Realizando a validação cruzada
scores = cross_val_score(modelo, X, y, cv=5)

print("Scores de validação cruzada:", scores)
print("Acurácia média:", scores.mean())
```

Documentação do Processo de Construção e Treinamento do Modelo Introdução:

Este documento oferece uma visão detalhada do processo de construção e treinamento de um modelo de aprendizado de máquina para a tarefa específica. Ele descreve as etapas,

os parâmetros selecionados e os resultados obtidos durante o desenvolvimento do modelo.

Objetivo:

O objetivo principal deste modelo é prever o status de entrega de pedidos com base em diversas características dos clientes e dos produtos.

Etapas do Processo:

1. Exploração de Dados e Pré-processamento:

Coleta de Dados:

-Fontes de Dados: Os dados foram coletados de um sistema de gerenciamento de pedidos online.

-Lista de Variáveis/Features: As features incluem nome do cliente, idade, tipo de pagamento, tipo de produto, data do pedido, código de rastreio e valor pago.

Limpeza e Pré-processamento:

-Tratamento de Valores Ausentes: Valores ausentes foram tratados através de imputação ou exclusão de registros.

-Identificação de Outliers: Outliers foram identificados e tratados com métodos como winsorização ou remoção.

-Transformações: As features categóricas foram codificadas com one-hot encoding ou label encoding, e as datas foram transformadas em características relevantes,

como dia da semana ou mês.

2. Implementação de Modelos de Aprendizado de Máquina:

Escolha de Algoritmos:

Algoritmos Utilizados:

-Foram selecionados algoritmos de classificação como RandomForestClassifier e GradientBoostingClassifier.

-Justificativa: Esses algoritmos foram escolhidos devido à sua capacidade de lidar com dados categóricos e numéricos, e por sua eficácia em problemas de classificação.

Implementação:

-Detalhes da Implementação: Os modelos foram implementados utilizando a biblioteca Scikit-learn em Python.

-Parâmetros Iniciais: Foram utilizados os parâmetros padrão dos algoritmos como ponto de partida.

3. Otimização e Validação do Modelo:

Otimização de Hiperparâmetros:

-Processo de Otimização: A otimização foi realizada utilizando Grid Search e Random Search para encontrar os melhores hiperparâmetros.

-Hiperparâmetros Ajustados: Os hiperparâmetros ajustados incluíram max_depth, n_estimators, min_samples_split, entre outros.

Validação Cruzada:

-Realização: A validação cruzada foi realizada com k-folds, utilizando 5 ou 10 folds para avaliar a performance do modelo.

-Resultados: Foram obtidos scores de validação cruzada para métricas como acurácia, precisão, recall e f1-score.

Parâmetros do Modelo:

-Hiperparâmetros Finais: Os hiperparâmetros finais foram selecionados com base nos resultados da otimização.

-Outros Parâmetros: Outros parâmetros relevantes incluem o número de features utilizadas e o método de tratamento de classes desbalanceadas.

Métricas de Avaliação:

-Métricas Utilizadas: As métricas utilizadas para avaliar o desempenho do modelo incluíram acurácia, precisão, recall e f1-score.

-Resultados Específicos: Os resultados específicos foram apresentados para cada métrica, destacando a performance do modelo em prever o status de entrega dos pedidos.

CIÊNCIA DE DADOS

Nesse tema será apresentado duas entregas, onde nelas teremos a análise descritiva dos dados e a modelagem estatística.

Entrega 1 – Análise Descritiva dos Dados

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns

dados = pd.read_csv("CSV/dados_usuarios.csv", sep=",")

#Parte 01

media_idade = dados['idade'].mean()
print(f"Média das idades: {media_idade}")

mediana_idade = dados['idade'].median()
print(f"Mediana das idades: {mediana_idade}")

desvio_padrao_idade = dados['idade'].std()
print(f"Desvio padrão das idades: {desvio_padrao_idade}")

media_valor_pago = dados['valor_pago'].mean()
print(f"Média dos valores pagos: {media_valor_pago}")

mediana_valor_pago = dados['valor_pago'].median()
print(f"Mediana dos valores pagos: {mediana_valor_pago}")

desvio_padrao_valor_pago = dados['valor_pago'].std()
print(f"Desvio padrão dos valores pagos: {desvio_padrao_valor_pago}")


## Parte 02

contagem_produto = dados['tipo_produto'].value_counts()

plt.figure(figsize=(10, 6))
```

```
sns.barplot(x=contagem_produto.index, y=contagem_produto.values,
palette="viridis")
plt.title('Contagem de Pedidos por Tipo de Produto')
plt.xlabel('Tipo de Produto')
plt.ylabel('Contagem de Pedidos')
plt.xticks(rotation=45)
plt.show()

sns.countplot(data=dados, x='tipo_pagamento', palette='pastel')
plt.title('Contagem de Tipos de Pagamento')
plt.xlabel('Tipo de Pagamento')
plt.ylabel('Contagem')

plt.tight_layout()
plt.show()



## Parte 03

padrao_tipo_pagamento = dados['tipo_pagamento'].value_counts()
print("Padrões no Tipo de Pagamento:")
print(padrao_tipo_pagamento)
print()

padrao_tipo_produto = dados['tipo_produto'].value_counts()
print("Padrões no Tipo de Produto:")
print(padrao_tipo_produto)
print()

padrao_status_pedido = dados['status_pedido'].value_counts()
print("Padrões no Status do Pedido:")
print(padrao_status_pedido)
```

Entrega 2 – Modelagem Estatística

```
import pandas as pd
import numpy as np
import statsmodels.api as sm
import seaborn as sns
import matplotlib.pyplot as plt
from statsmodels.formula.api import ols
from scipy import stats
from sklearn.tree import DecisionTreeClassifier
```

```
from sklearn.model_selection import train_test_split
from sklearn.tree import plot_tree
from sklearn.linear_model import LinearRegression
from sklearn.metrics import mean_squared_error, r2_score
from sklearn.metrics import accuracy_score, classification_report,
confusion_matrix

dados = pd.read_csv("CSV/dados_usuarios.csv", sep=","""")

##Parte 01

#Regressão Linear
X = dados['idade']
y = dados['valor_pago']

X = sm.add_constant(X)

modelo = sm.OLS(y, X).fit()

print(modelo.summary())

plt.figure(figsize=(10, 6))
sns.regplot(x='idade', y='valor_pago', data=dados, line_kws={"color":
"red"})
plt.title('Regressão Linear: Valor Pago vs. Idade')
plt.xlabel('Idade')
plt.ylabel('Valor Pago (R$)')
plt.show()

#Análise de Variância
modelo_anova = ols('valor_pago ~ C(tipo_pagamento)', data=dados).fit()

anova_tabela = sm.stats.anova_lm(modelo_anova, typ=2)

print(anova_tabela)

plt.figure(figsize=(10, 6))
sns.boxplot(x='tipo_pagamento', y='valor_pago', data=dados,
palette='pastel')
plt.title('ANOVA: Valor Pago por Tipo de Pagamento')
plt.xlabel('Tipo de Pagamento')
plt.ylabel('Valor Pago (R$)')
plt.show()



## Parte 02 / Parte 03 / Parte 04 / Parte 05

#Regressão Linear
```

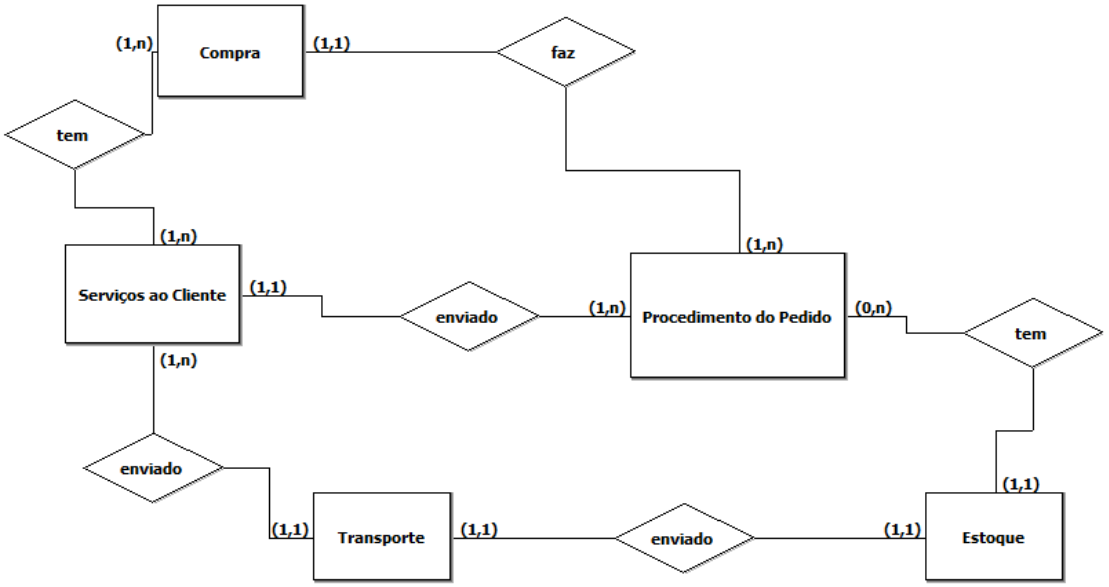
```
X = dados.drop(columns=['nome', 'status_pedido', 'data_pedido',
'codigo_rastreio', 'valor_pago'])
y = dados['valor_pago']

X = pd.get_dummies(X)

X_train, X_test, y_train, y_test = train_test_split(X, y,
test_size=0.3, random_state=42)

modelo_regressao = LinearRegression()
modelo_regressao.fit(X_train, y_train)

y_pred = modelo_regressao.predict(X_test)

mse = mean_squared_error(y_test, y_pred)
r2 = r2_score(y_test, y_pred)

print("Coeficientes do Modelo:")
print(modelo_regressao.coef_)
print("\nIntercepto do Modelo:")
print(modelo_regressao.intercept_)
print("\nMean Squared Error (MSE):", mse)
print("R² Score:", r2)



#Classificação usando Árvore de Decisão

dados['status_entregue'] = np.where(dados['status_pedido'] ==
'entregue', 1, 0)

dados = pd.get_dummies(dados, columns=['tipo_pagamento'],
drop_first=True)

X = dados[['idade', 'valor_pago', 'tipo_pagamento_cartão de crédito',
'tipo_pagamento_pix']]
y = dados['status_entregue']

X_train, X_test, y_train, y_test = train_test_split(X, y,
test_size=0.3, random_state=42)

modelo_arvore = DecisionTreeClassifier(random_state=42)
modelo_arvore.fit(X_train, y_train)

y_pred = modelo_arvore.predict(X_test)

accuracy = accuracy_score(y_test, y_pred)
class_report = classification_report(y_test, y_pred)
```

```
conf_matrix = confusion_matrix(y_test, y_pred)

print("Acurácia do Modelo:", accuracy)
print("\nRelatório de Classificação:")
print(class_report)
print("\nMatriz de Confusão:")
print(conf_matrix)

plt.figure(figsize=(20, 10))
plot_tree(modelo_arvore, feature_names=X.columns, class_names=['Não
Entregue', 'Entregue'], filled=True)
plt.title("Árvore de Decisão")
plt.show()

#Parte 06

mse_regressao = mean_squared_error(y_test, y_pred)
r2_regressao = r2_score(y_test, y_pred)

accuracy_arvore = accuracy_score(y_test, y_pred)

if mse_regressao < accuracy_arvore:
    print('O modelo de regressão linear é mais eficaz.')
else:
    print('O modelo de classificação com árvore de decisão é mais
eficaz.')
```

MODELAGEM DE DADOS

Nesse tópico será apresentado três entregas, onde nelas teremos a modelagem conceitual, modelagem lógica e normalização e a entrega do dicionário de dados de uma simulação de cadastro.

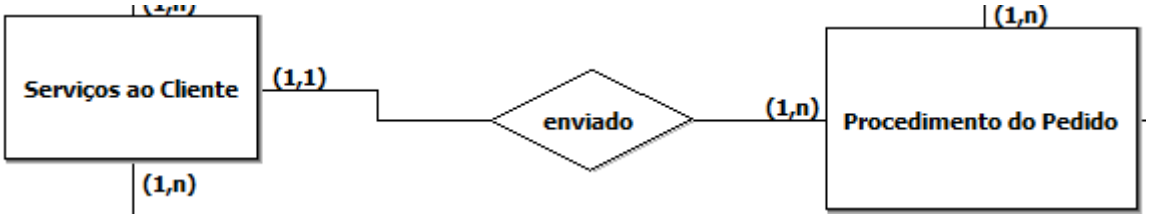
Entrega 1 – Modelagem Conceitual



Principais Entidades:

-Serviços ao Cliente

-Procedimento do Pedido



Entrega 2 – Modelagem Lógica e Normalização

Tabelas Normalizadas:

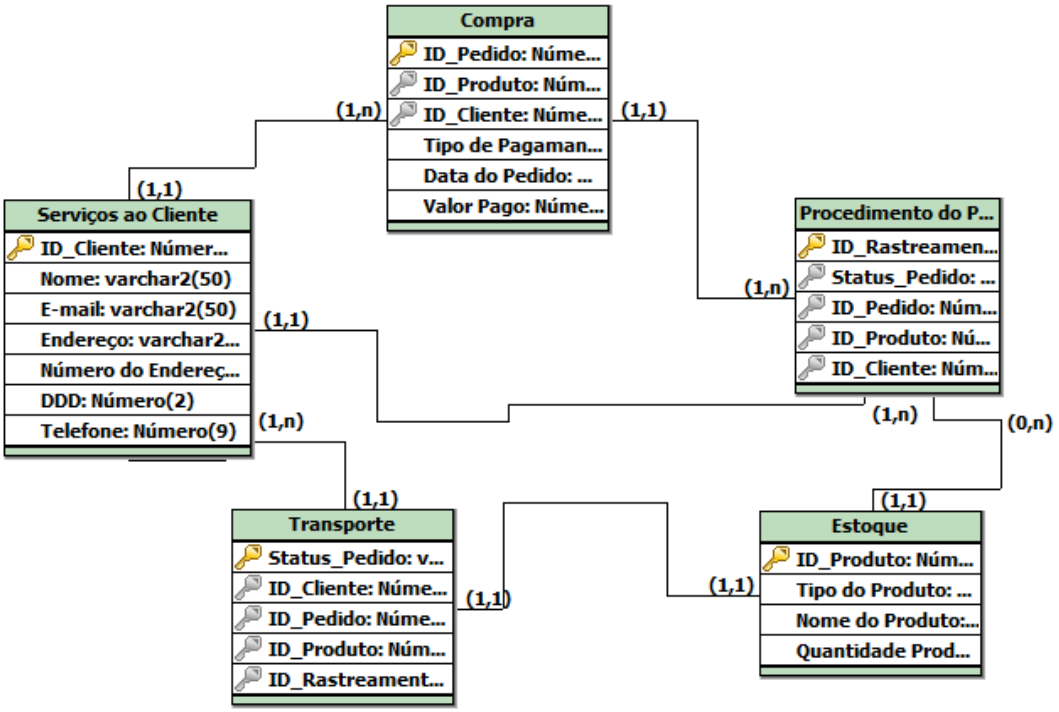
-Serviços ao Cliente (ID_Cliente, Nome, Email, Endereco, Num_Endereco, DDD, Telefone)

-Compra (ID_Pedido, ID_Cliente, ID_Produto, Tipo_Pagamento, Data_Pedido, Valor)

-Procedimento do Pedido (Status_Pedido, ID_Rastreamento, ID_Pedido, ID_Produto, ID_Cliente)

-Estoque (ID_Produto, Tipo_Produto, Nome_Produto, Quantidade_Produto)

-Transporte (ID_Rastreamento, ID_Cliente, ID_Pedido, ID_Produto, Status_Pedido)



Entrega 3 – Entregar Dicionário de Dados uma simulação de cadastro

Listagens das Tabelas:

compra

Column name	DataType	PK	FK	NN	UQ	BIN	UN	ZF	AI	Default	Comment
ID_Pedido	INT(4)	✓		✓							
ID_Produto	INT(4)		✓	✓							
ID_Cliente	INT(4)		✓	✓							
Tipo_Pagamento	VARCHAR(20)			✓							
Data_Pedido	DATE			✓							
Valor_Pago	DECIMAL(10,2)			✓							

estoque

Column name	DataType	PK	FK	NN	UQ	BIN	UN	ZF	AI	Default	Comment
ID_Produto	INT(4)	✓		✓							
Tipo_Produto	VARCHAR(50)			✓							
Nome_Produto	VARCHAR(50)			✓							
Quantidade_Produto	INT(9)			✓							

procedimento_do_pedido

Column name	DataType	PK	FK	NN	UQ	BIN	UN	ZF	AI	Default	Comment
ID_Rastreamento	INT(9)	✓		✓							
Status_Pedido	VARCHAR(50)		✓	✓							
ID_Pedido	INT(4)		✓	✓							
ID_Produto	INT(4)		✓	✓							
ID_Cliente	INT(4)		✓	✓							

servicos_ao_cliente

Column name	DataType	PK	FK	NN	UQ	BIN	UN	ZF	AI	Default	Comment
ID_Cliente	INT(4)	✓		✓							
Nome	VARCHAR(50)			✓							
Email	VARCHAR(50)			✓							
Endereco	VARCHAR(50)			✓							
Num_Endereco	INT(5)			✓							
DDD	INT(2)			✓							
Telefone	INT(9)			✓							

transporte

Column name	DataType	PK	FK	NN	UQ	BIN	UN	ZF	AI	Default	Comment
Status_Pedido	VARCHAR(25)	✓		✓							
ID_Cliente	INT(4)		✓	✓							
ID_Pedido	INT(4)		✓	✓							
ID_Produto	INT(4)		✓	✓							
ID_Rastreamento	INT(9)		✓	✓							

Simulação de cadastro:

Tabela: Serviços ao Cliente

ID_Cliente	Nome	Email	Endereco	Num_Endereco	DDD	Telefone
1	João Pedro	joao@example.com	Rua A	123	11	999999999
2	Ana Clara	ana@example.com	Rua B	456	11	888888888

Tabela: Compra

ID_Pedido	ID_Cliente	ID_Produto	Tipo_Pagamento	Data_Pedido	Valor_Pago
101	1	1	Cartão de Crédito	07/05/2024	150.00
102	2	2	Boleto Bancário	08/05/2024	200.00

Tabela: Procedimento do Pedido

ID_Rastreamento	Status_Pedido	ID_Pedido	ID_Produto	ID_Cliente
123456789	Em Trânsito	101	1	1
987654321	Entregue	102	2	2

Tabela: Estoque

ID_Produto	Tipo_Produto	Nome_Produto	Quantidade_Produto
1	Eletrônico	Smartphone	50
2	Vestuário	Camiseta	100

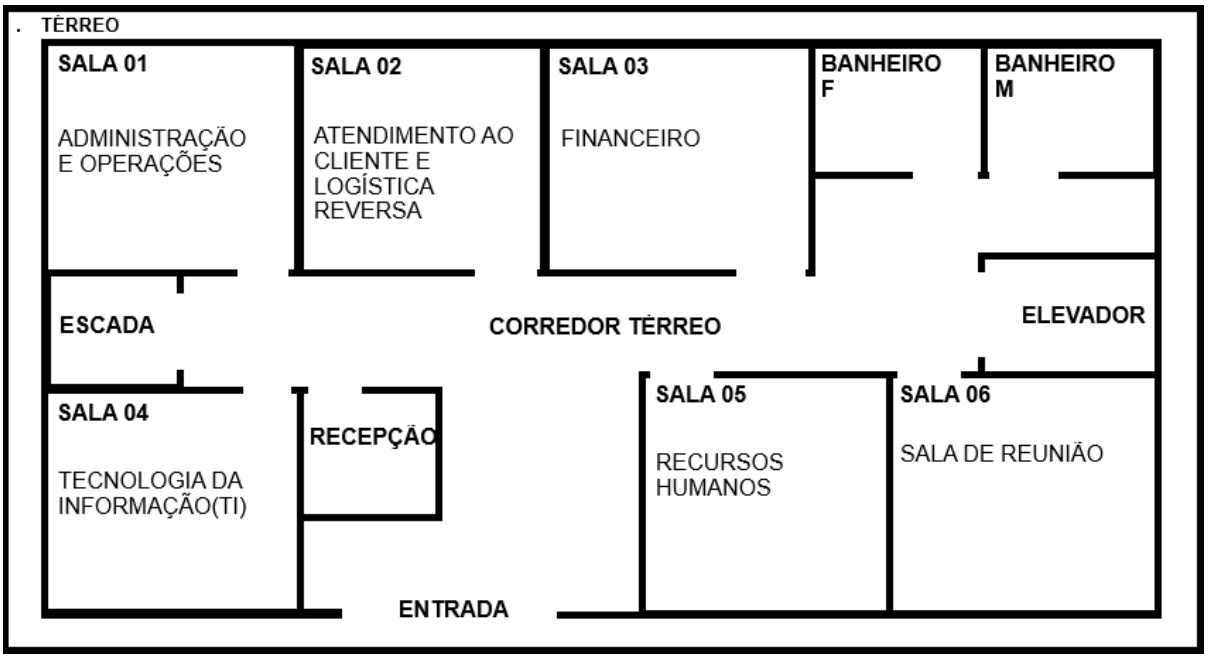
Tabela: Transporte

Status_Pedido	ID_Rastreamento	ID_Pedido	ID_Produto	ID_Cliente
Em Trânsito	123456789	101	1	1
Entregue	987654321	102	2	2

REDES DE COMPUTADORES

Nesse tópico terá duas entregas, onde a primeira será feita a montagem da planta baixa de Rede da Empresa, já na segunda será entregue as configurações de IP de todos os equipamentos que contém na empresa.

Entrega 1 – Montar a planta baixa de Rede da Empresa



Equipamentos:

Administração e Operações:

Computadores e equipamentos de escritório (mesas, cadeiras, armários).

Impressoras, scanners e outros dispositivos de escritório.

Software de gestão empresarial (ERP).

Telefones e sistemas de comunicação interna.

Atendimento ao Cliente e Logística Reversa:

Computadores e equipamentos de escritório para atendimento ao cliente.
Softwares de gestão de relacionamento com o cliente (CRM).
Telefones e sistemas de comunicação para contato com clientes.
Equipamentos de embalagem e envio para logística reversa.

Financeiro:

Computadores e monitores.
Telefones.
Armários para arquivos financeiros.
Sistemas de videoconferência para reuniões.
Impressoras e scanners.
Calculadoras financeiras.
Software de contabilidade e gestão financeira.
Cofre/arquivo seguro para documentos confidenciais.

Tecnologia da Informação (TI):

Servidores de rede e equipamentos de infraestrutura de TI.
Computadores e dispositivos para equipe de suporte técnico.
Equipamentos de rede (roteadores, switches, cabos).
Software de segurança cibernética e firewalls.
Sistemas de rastreamento de remessas e softwares de logística.

Recursos Humanos:

Computadores e monitores

Mesas e cadeiras ergonômicas

Armários para arquivos de funcionários

Equipamento de videoconferência

Software de gestão de RH (folha de pagamento, recrutamento, etc.)

Entrega 2 – Configuração de IP de todos os equipamentos

Classe de rede: Classe C (192.168.0.1) a (192.168.0.254)

Padrão de Rede:

Administração e Operações:

Faixa de endereços IP: 192.168.0.1 a 192.168.0.49

Financeiro:

Faixa de endereços IP: 192.168.0.50 a 192.168.0.99

Recursos Humanos:

Faixa de endereços IP: 192.168.0.100 a 192.168.0.149

Atendimento ao Cliente e Logística Reversa:

Faixa de endereços IP: 192.168.0.150 a 192.168.0.199

Tecnologia da Informação (TI):

Faixa de endereços IP: 192.168.0.200 a 192.168.0.254

SEGURANÇA DA INFORMAÇÃO

Nesse tópico será abordado dois importantes assuntos na área da segurança da informação que é a Análise de Riscos e Implementações de Medidas de Segurança.

Entrega 1 - Análise de Riscos

Riscos Identificados:

Ataques de malware: Infecção por vírus, worms, trojans, etc.

Phishing: Tentativas de enganar funcionários para obter informações confidenciais.

Ataques de engenharia social: Manipulação psicológica para obter acesso não autorizado a informações.

Vazamento de dados confidenciais: Roubo ou exposição não autorizada de informações sensíveis.

Ataques de negação de serviço (DDoS): Sobrecarga de servidores ou redes, causando interrupção do serviço.

Ataques de ransomware: Criptografia de dados por hackers em troca de resgate.

Falhas de segurança de software: Vulnerabilidades em sistemas operacionais, aplicativos ou firmware.

Falhas de segurança de hardware: Vulnerabilidades em dispositivos físicos, como roteadores ou firewalls.

Ataques de força bruta: Tentativas repetidas de login para adivinhar credenciais de usuário.

Furto ou perda de dispositivos: Roubo ou extravio de laptops, celulares ou unidades USB.

Acesso não autorizado: Funcionários ou terceiros obtendo acesso não autorizado a sistemas ou áreas restritas.

Espionagem corporativa: Coleta de informações confidenciais por concorrentes ou agentes externos.

Fraude financeira: Manipulação de transações ou desvio de fundos por funcionários internos.

Vulnerabilidades de rede Wi-Fi: Acesso não autorizado à rede sem fio da empresa.

Falhas de backup e recuperação de desastres: Incapacidade de recuperar dados após uma falha ou desastre.

Exploração de falhas de segurança física: Entrada não autorizada em instalações ou data centers.

Ataques de injeção SQL: Inserção maliciosa de código SQL em formulários web ou bancos de dados.

Roubo de identidade: Uso não autorizado de informações pessoais para acessar contas ou serviços.

Falhas de conformidade regulatória: Não conformidade com regulamentos de privacidade, como GDPR ou LGPD.

Desastres naturais: Eventos como incêndios, enchentes ou terremotos que podem interromper as operações.

Análise de Riscos:

Ataques de malware:

Impacto: Alto (pode resultar em perda de dados, interrupção dos negócios, danos à reputação).

Probabilidade de ocorrência: Moderada a Alta (depende da eficácia das medidas de segurança e da conscientização dos usuários).

Phishing:

Impacto: Moderado a Alto (pode levar a roubo de credenciais, acesso não autorizado a informações confidenciais).

Probabilidade de ocorrência: Moderada (depende da capacidade dos atacantes e da conscientização dos usuários).

Ataques de engenharia social:

Impacto: Moderado a Alto (pode resultar em divulgação de informações sensíveis ou acesso não autorizado).

Probabilidade de ocorrência: Baixa a Moderada (depende da sofisticação dos ataques e da conscientização dos funcionários).

Vazamento de dados confidenciais:

Impacto: Alto (pode resultar em danos à reputação, multas regulatórias, perda de confiança dos clientes).

Probabilidade de ocorrência: Moderada (depende da eficácia das medidas de segurança e do valor dos dados para os atacantes).

Ataques de negação de serviço (DDoS):

Impacto: Alto (pode resultar em indisponibilidade dos serviços, perda de receita, danos à reputação).

Probabilidade de ocorrência: Baixa a Moderada (depende da visibilidade e importância da empresa para os atacantes).

Ataques de ransomware:

Impacto: Alto (pode resultar em perda de dados, interrupção dos negócios, danos financeiros significativos).

Probabilidade de ocorrência: Moderada (depende da exposição da empresa e da eficácia das medidas de segurança).

Falhas de segurança de software:

Impacto: Moderado a Alto (pode resultar em acesso não autorizado, vazamento de dados, interrupção dos serviços).

Probabilidade de ocorrência: Moderada (depende da quantidade e gravidade das vulnerabilidades presentes nos sistemas).

Falhas de segurança de hardware:

Impacto: Moderado (pode resultar em interrupção dos serviços, perda de dados, acesso não autorizado).

Probabilidade de ocorrência: Baixa a Moderada (depende da qualidade e manutenção dos dispositivos físicos).

Ataques de força bruta:

Impacto: Baixo a Moderado (pode resultar em acesso não autorizado, comprometimento de contas de usuário).

Probabilidade de ocorrência: Moderada (depende da exposição dos sistemas e da capacidade dos atacantes).

Furto ou perda de dispositivos:

Impacto: Moderado (pode resultar em perda de dados, acesso não autorizado, comprometimento da segurança).

Probabilidade de ocorrência: Moderada (depende das práticas de segurança física e conscientização dos funcionários).

Acesso não autorizado:

Impacto: Moderado (pode resultar em divulgação de informações sensíveis, comprometimento da integridade dos dados).

Probabilidade de ocorrência: Moderada (depende da eficácia dos controles de acesso e da conscientização dos funcionários).

Espionagem corporativa:

Impacto: Moderado a Alto (pode resultar em perda de dados confidenciais, comprometimento da estratégia empresarial).

Probabilidade de ocorrência: Baixa a Moderada (depende do interesse de concorrentes ou agentes externos).

Fraude financeira:

Impacto: Moderado a Alto (pode resultar em perdas financeiras significativas, danos à reputação).

Probabilidade de ocorrência: Baixa a Moderada (depende da presença de controles financeiros e da integridade dos funcionários).

Vulnerabilidades de rede Wi-Fi:

Impacto: Moderado (pode resultar em acesso não autorizado à rede, vazamento de informações).

Probabilidade de ocorrência: Moderada (depende da segurança da rede Wi-Fi e da visibilidade externa da empresa).

Falhas de backup e recuperação de desastres:

Impacto: Alto (pode resultar em perda irreparável de dados, interrupção prolongada dos negócios).

Probabilidade de ocorrência: Baixa a Moderada (depende da qualidade dos processos de backup e da prevenção de desastres).

Exploração de falhas de segurança física:

Impacto: Moderado a Alto (pode resultar em acesso não autorizado a instalações críticas, perda de equipamentos).

Probabilidade de ocorrência: Baixa a Moderada (depende da eficácia dos controles de segurança física).

Ataques de injeção SQL:

Impacto: Moderado a Alto (pode resultar em acesso não autorizado a dados sensíveis, comprometimento da integridade dos dados).

Probabilidade de ocorrência: Moderada (depende da presença de vulnerabilidades nos sistemas e da exposição a ataques).

Roubo de identidade:

Impacto: Moderado (pode resultar em comprometimento de contas pessoais e corporativas, perda de dados sensíveis).

Probabilidade de ocorrência: Baixa a Moderada (depende da exposição dos dados pessoais e da sofisticação dos atacantes).

Falhas de conformidade regulatória:

Impacto: Moderado a Alto (pode resultar em multas regulatórias, perda de confiança dos clientes).

Probabilidade de ocorrência: Baixa a Moderada (depende do escopo das regulamentações aplicáveis e do cumprimento dos requisitos).

Desastres naturais:

Impacto: Alto (pode resultar em danos físicos às instalações, interrupção prolongada dos negócios).

Probabilidade de ocorrência: Baixa a Moderada (depende da localização geográfica e da frequência de desastres na região).

Entrega 2 - Implementação de Medidas de Segurança

Política de Controle de Acesso:

Princípio do menor privilégio:

Política que concede aos usuários apenas os privilégios de acesso necessários para realizar suas funções, minimizando assim o risco de acesso não autorizado a informações sensíveis.

Autenticação de dois fatores (2FA):

Política que requer autenticação de dois fatores para acessar sistemas ou informações críticas, reduzindo o risco de acesso não autorizado em caso de comprometimento de credenciais.

Política de senha forte:

Política que define requisitos rigorosos para senhas, como comprimento mínimo, uso de caracteres especiais e atualização periódica, visando proteger contra-ataques de força bruta e comprometimento de contas.

Monitoramento de acesso e auditoria:

Política que estabelece a monitorização contínua dos acessos aos sistemas e informações sensíveis, registrando atividades de usuário e realizando auditorias regulares para identificar possíveis anomalias ou comportamentos suspeitos.

Controle de acesso físico:

Política que estabelece medidas de segurança física, como sistemas de controle de acesso com cartões de identificação, para proteger instalações e equipamentos contra acesso não autorizado.

Gerenciamento de identidades e acessos (IAM):

Política que define processos e procedimentos para gerenciar de forma eficaz as identidades dos usuários, atribuir e revogar acessos de acordo com as necessidades operacionais e garantir a conformidade regulatória.

Restrição de acesso a dados confidenciais:

Política que restringe o acesso a informações confidenciais apenas a usuários autorizados, implementando controles de acesso baseados em funções (RBAC) ou criptografia de dados.

Política de acesso remoto seguro:

Política que estabelece requisitos para conexões remotas, como o uso de VPNs (Redes Privadas Virtuais) e autenticação multifator, para garantir a segurança das comunicações e dos dados durante o acesso remoto.

Revisão periódica de acessos:

Política que estabelece revisões regulares dos direitos de acesso dos usuários, removendo acessos desnecessários ou não utilizados e garantindo que os privilégios de acesso estejam alinhados com as responsabilidades do usuário.

Treinamento de conscientização em segurança:

Política que requer treinamento regular para todos os funcionários sobre práticas de segurança, incluindo a importância do controle de acesso, reconhecimento de ameaças cibernéticas e procedimentos de resposta a incidentes.

Sistema de Detecção de Intrusão:

Firewalls de Próxima Geração (NGFW):

Implementar firewalls NGFW que oferecem funcionalidades avançadas de inspeção de tráfego para identificar e bloquear ameaças de rede, como ataques de negação de serviço (DDoS), malware e tentativas de intrusão.

Sistemas de Detecção e Prevenção de Intrusões (IDS/IPS):

Configurar IDS/IPS para monitorar o tráfego de rede em tempo real e identificar atividades suspeitas ou maliciosas, como tentativas de exploração de vulnerabilidades ou tráfego anômalo.

Análise de Comportamento de Usuários (UBA):

Implementar sistemas de UBA para monitorar o comportamento dos usuários e identificar atividades incomuns ou maliciosas, como acessos a recursos não autorizados ou transferências de dados suspeitas.

Sistemas de Detecção de Malware:

Utilizar soluções de detecção de malware que empregam técnicas avançadas, como análise heurística e sandboxing, para identificar e bloquear ameaças de malware em tempo real.

Monitoramento de Logs de Segurança:

Configurar sistemas de monitoramento de logs para coletar e analisar registros de eventos de segurança em toda a infraestrutura, permitindo a detecção precoce de atividades suspeitas ou violações de segurança.

Segmentação de Rede:

Implementar segmentação de rede para dividir a infraestrutura em zonas isoladas e controlar o tráfego entre elas, reduzindo assim a superfície de ataque e limitando o impacto de um eventual comprometimento.

Atualizações e Patches de Segurança:

Manter todos os sistemas e software atualizados com os patches de segurança mais recentes para corrigir vulnerabilidades conhecidas e reduzir o risco de exploração por parte de invasores.

Políticas de Acesso e Controle de Privacidade:

Implementar políticas de acesso granular e controle de privacidade para restringir o acesso a informações sensíveis apenas a usuários autorizados e garantir a conformidade com regulamentos de privacidade, como GDPR ou LGPD.

Monitoramento de Atividade de Administradores:

Monitorar e registrar as atividades dos administradores de sistema para detectar comportamentos suspeitos ou não autorizados, como acessos fora do horário de expediente ou modificações não autorizadas de configurações.

Treinamento em Conscientização de Segurança:

Fornecer treinamento regular em conscientização de segurança para todos os funcionários, destacando os riscos de segurança, boas práticas de segurança e procedimentos de relato de incidentes.