Story Polaroids

Nomes dos participantes:

Guilherme Ribeiro Araujo Dos Santos RA: 2223205236

Beatriz Vieira de Jesus RA: 2223203960

Leonardo de Jesus Ribeiro RA: 2223204095

Índice

Escopo

O presente documento detalha o escopo do projeto proposto pela empresa Story Polaroids. O projeto tem como objetivo uma empresa real que tem como objetivo criar fotos, quadros, chaveiros, entre outros itens, para todos os tipos de comemorações, como casamentos, festas, presentes para outras pessoas e para você mesmo guardar de lembrança. Os benefícios incluem guardar essas lembranças não só na memória, mas também em formato físico, permitindo revê-las a qualquer momento.

Serviços prestados:

Serviços:

°Revelação de Fotos Polaroids °Personalização de Quadros Polaroids °Chaveiros Personalizados com Fotos °Cubos Giratórios de Fotos Personalizados °Venda de Produtos Personalizados com Fotos Polaroids

Tecnologias:

°Plataforma de E-commerce °Sistema de Gestão Empresarial °Ferramentas de Design Gráfico °Estratégias de Marketing Digital °Sistema de Impressão de Alta Qualidade

Estrutura interna da empresa

1 Aprendizado de Máquina


```
import pandas as pd
import matplotlib.pyplot as plt
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression
from sklearn.metrics import mean_absolute_error, mean_squared_error, r2_score


dados = pd.read_csv('test.csv')


print(dados.head())
```

```
    age  anaemia  creatinine_phosphokinase  diabetes  ejection_fraction  \
0  75.0        0                       582         0                 20
1  55.0        0                      7861         0                 38
2  65.0        0                       146         0                 20
3  50.0        1                       111         0                 20
4  65.0        1                       160         1                 20

   high_blood_pressure  platelets  serum_creatinine  serum_sodium  sex  \
0                    1  265000.00               1.9           130    1
1                    0  263358.03               1.1           136    1
2                    0  162000.00               1.3           129    1
3                    0  210000.00               1.9           137    1
4                    0  327000.00               2.7           116    0

   smoking  time  DEATH_EVENT
0        0     4            1
1        0     6            1
2        1     7            1
3        0     7            1
4        0     8            1
```

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


from sklearn.metrics import confusion_matrix
y_true = [1, 0, 1, 0, 1, 0, 0, 1]
y_pred = [1, 1, 0, 0, 1, 1, 0, 1]
matriz_confusao = confusion_matrix(y_true, y_pred)
print(matriz_confusao)
```

```
[[2 2]
 [1 3]]
```

```
from sklearn.model_selection import train_test_split

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

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

modelo.fit(X_train, y_train)
```

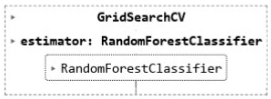
```
▾ RandomForestClassifier
RandomForestClassifier()
```

```
from sklearn.metrics import precision_score, recall_score, f1_score


y_pred = modelo.predict(X_test)


precisao = precision_score(y_test, y_pred)
recall = recall_score(y_test, y_pred)
f1 = f1_score(y_test, y_pred)



print(f'Precisão: {precisao}, Recall: {recall}, F1-score: {f1}')
```

```
Precisão: 0.75, Recall: 0.48, F1-score: 0.5853658536585366
```

```
from sklearn.model_selection import GridSearchCV
from sklearn.ensemble import RandomForestClassifier


parametros = {'n_estimators': [50, 100, 150], 'max_depth': [None, 10, 20]}
```

```
modelo = RandomForestClassifier()


grid_search = GridSearchCV(modelo, parametros, cv=5)


grid_search.fit(X_train, y_train)
```




```
print("Melhores hiperparâmetros:", grid_search.best_params_)
```

```
Melhores hiperparâmetros: {'max_depth': 20, 'n_estimators': 150}
```

```
y_pred = grid_search.predict(X_test)
precisao = precision_score(y_test, y_pred)
recall = recall_score(y_test, y_pred)
f1 = f1_score(y_test, y_pred)
print(f'Precisão: {precisao}, Recall: {recall}, F1-score: {f1}')
```

```
Precisão: 0.9285714285714286, Recall: 0.52, F1-score: 0.6666666666666666
```

```
from sklearn.model_selection import cross_val_score


pontuacoes = cross_val_score(modelo, X, y, cv=5)


print("Pontuações de validação cruzada:", pontuacoes)
```

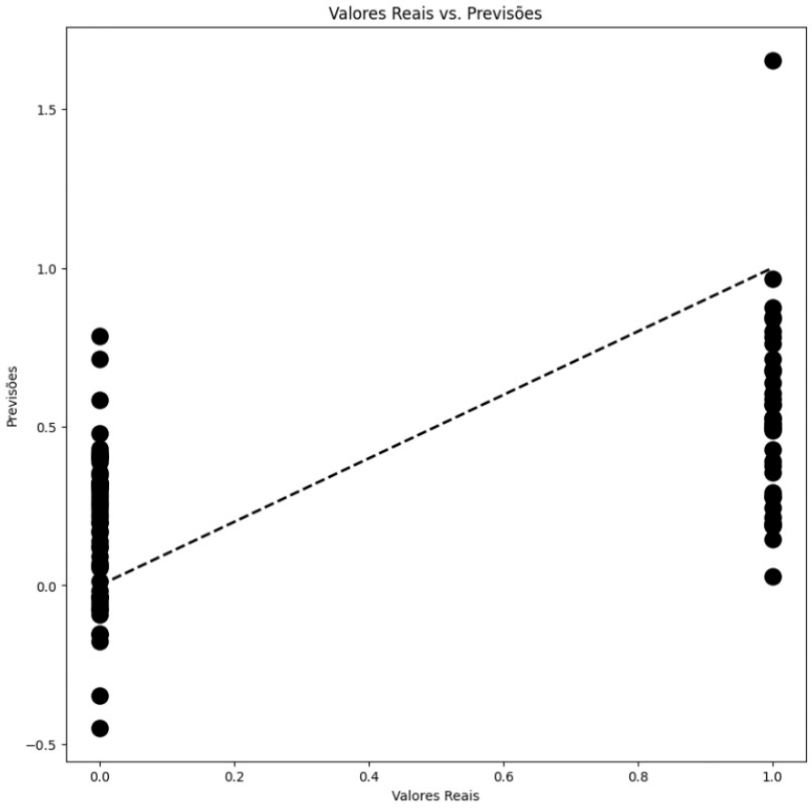
```
Pontuações de validação cruzada: [0.41666667 0.81666667 0.88333333 0.7        0.71186441]
```

```
print("Média das pontuações:", pontuacoes.mean())
print("Desvio padrão das pontuações:", pontuacoes.std())
```

```
Média das pontuações: 0.7057062146892654
Desvio padrão das pontuações: 0.15967339147010895
```

```
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression
from sklearn.metrics import mean_squared_error, r2_score
import matplotlib.pyplot as plt

X_train, X_val, y_train, y_val = train_test_split(X, y, test_size=0.3, random_state=42)

model = LinearRegression()

model.fit(X_train, y_train)

predictions = model.predict(X_val)

mse = mean_squared_error(y_val, predictions)
r2 = r2_score(y_val, predictions)

print("Erro Quadrático Médio (MSE):", mse)
print("R-quadrado (R²):", r2)

plt.figure(figsize=(10, 10,))
plt.scatter(y_val, predictions, color='black', alpha=1, marker='o', s=150)
plt.plot([y_val.min(), y_val.max()], [y_val.min(), y_val.max()], 'k--', lw=2)
plt.xlabel('Valores Reais')
plt.ylabel('Previsões')
plt.title('Valores Reais vs. Previsões')
plt.show()
```

```
Erro Quadrático Médio (MSE): 0.18028241159969233
R-quadrado (R²): 0.2553352708018828
```



Etapas do processo

As fontes utilizadas foram obtidas no site Kaggle. O nome do conjunto de dados é "Prever a sobrevivência de pacientes com insuficiência cardíaca".

As variáveis utilizadas, ou seja, os nomes das colunas, foram: age, anaemia, creatinine_phosphokinase, diabetes e ejection_fraction.

Escolhi esse algoritmo por sua facilidade de uso e porque foi explicado na aula. Era o que eu conhecia para colocar em prática.

Utilizei a biblioteca Scikit-learn porque é conhecida e tem uma variedade de ferramentas que facilitam para quem está começando a entender e colocar em prática. Com ela, é possível criar modelos e realizar treinamentos.

Utilizei o Grid Search para otimizar o modelo, pois nos vídeos que assisti e nas pesquisas que fiz, foi o método recomendado e que eu entendi melhor.

Para validar o modelo, utilizei a validação cruzada com 5 folds, ou seja, o modelo foi treinado 5 vezes para teste, proporcionando uma avaliação mais completa do seu desempenho.

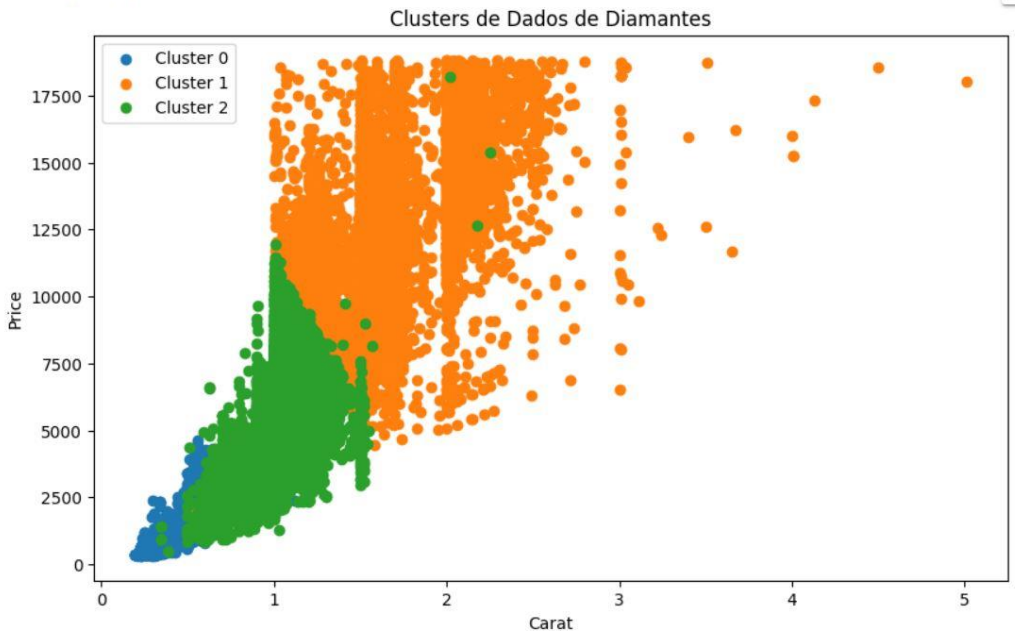
Os hiperparâmetros foram otimizados utilizando o Grid Search.

As métricas de avaliação utilizadas foram o MSE (Erro Quadrático Médio) e R2, entre outras informações que indicaram um bom ajuste no modelo.

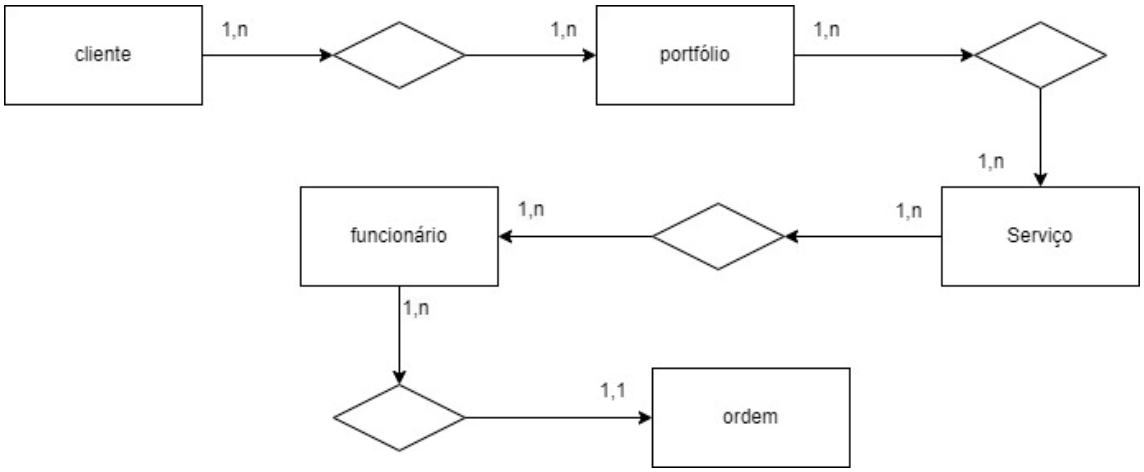
```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
from sklearn.cluster import KMeans
from sklearn.preprocessing import StandardScaler, OneHotEncoder
from sklearn.compose import ColumnTransformer

df = pd.read_csv("diamonds.csv")

print(df.head())

X = df.drop(['carat'], axis=1)

numeric_features = X.select_dtypes(include=['float64', 'int64']).columns
numeric_transformer = StandardScaler()

categorical_features = ['cut', 'color', 'clarity']
categorical_transformer = OneHotEncoder()

preprocessor = ColumnTransformer(
    transformers=[
        ('num', numeric_transformer, numeric_features),
        ('cat', categorical_transformer, categorical_features)])

X_processed = preprocessor.fit_transform(X)

n_clusters = 3

kmeans = KMeans(n_clusters=n_clusters, random_state=42)
kmeans.fit(X_processed)

df['cluster'] = kmeans.labels_

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

for cluster in range(n_clusters):
    cluster_data = df[df['cluster'] == cluster]
    plt.scatter(cluster_data['carat'], cluster_data['price'], label=f'Cluster {cluster}')

plt.title('Clusters de Dados de Diamantes')
plt.xlabel('Carat')
plt.ylabel('Price')
plt.legend()
                                        plt.show()
```

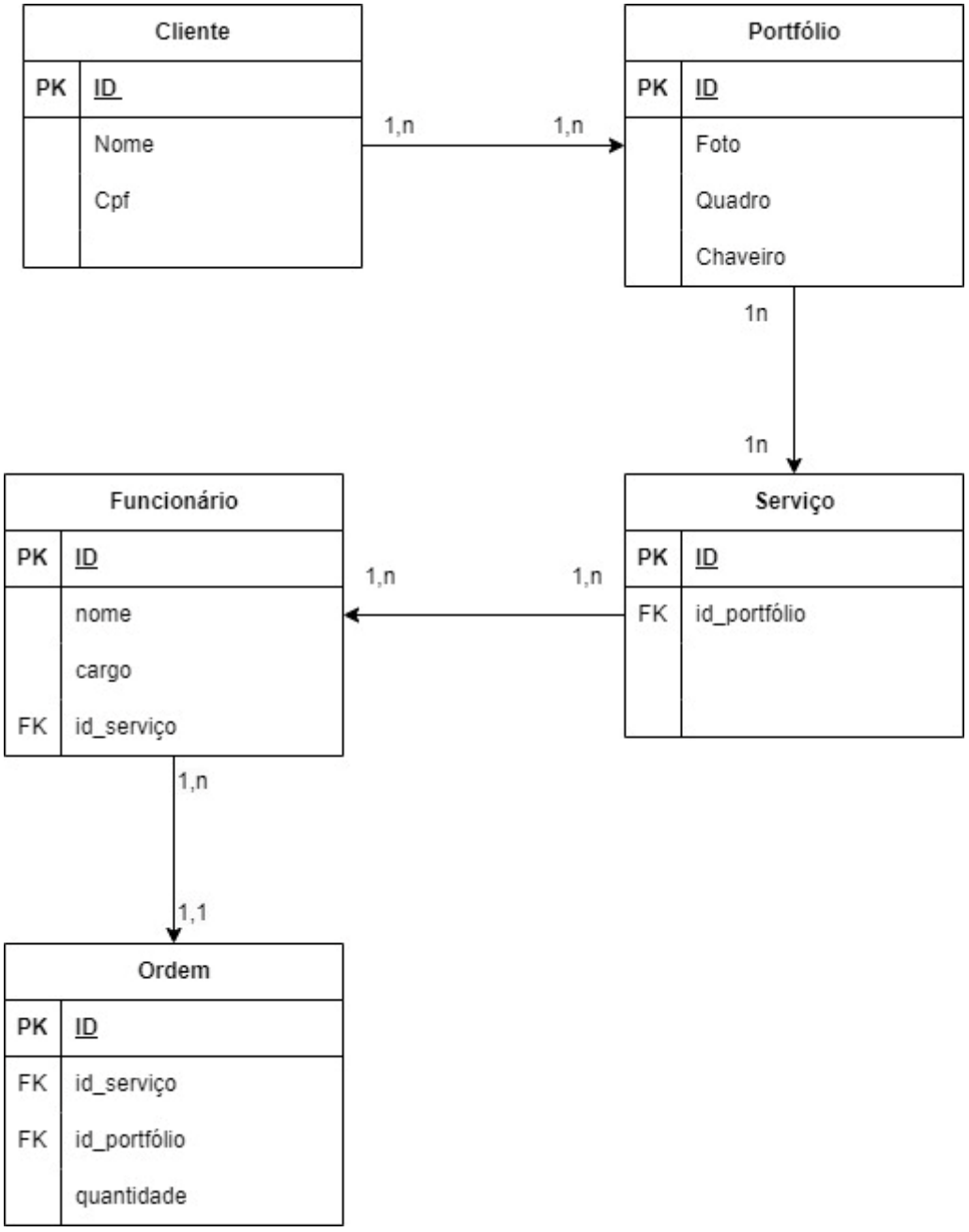
	carat	cut	color	clarity	depth	table	price	x	y	z
0	0.23	Ideal	E	SI2	61.5	55.0	326	3.95	3.98	2.43
1	0.21	Premium	E	SI1	59.8	61.0	326	3.89	3.84	2.31
2	0.23	Good	E	VS1	56.9	65.0	327	4.05	4.07	2.31
3	0.29	Premium	I	VS2	62.4	58.0	334	4.20	4.23	2.63
4	0.31	Good	J	SI2	63.3	58.0	335	4.34	4.35	2.75

```
warnings.warn(
```



3 Modelagem de banco de Dados




Cliente
PK
ID
Nome
Cpf
1,n
1,n
Portfólio
PK
ID
Foto
Quadro
Chaveiro
1n
1n
Funcionário
PK
ID
nome
cargo
FK
id_serviço
1,n
1,n
Serviço
PK
ID
FK
id_portfólio
1,n
1,1
Ordem
PK
ID
FK
id_serviço
FK
id_portfólio
quantidade

cliente		
id	nome	cpf
1	Guilherme Ribeiro	909899878-43
2	Leonardo Ribeiro	894989343-96
3	Beatriz de Jesus	232343454-32

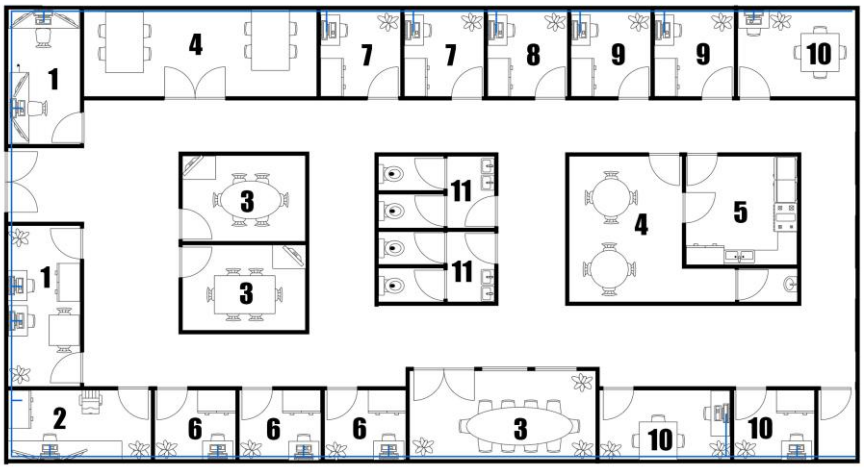
Portfólio		
id	trabalho	preço
1	foto	R$1,30
2	quadro	R$75,50
3	chaveiro	R$4,50

serviços		
id	id_portfólio	id_cliente
1	2	3
2	3	2
3	1	1

funcionario		
id	nome	cargo
1	ronaldo	gerente
2	romario	auxiliar de produção
3	ronaldinho	produção

ordem			
id	id_serviços	id_portfólio	quantidade
1	2	1	4
2	3	3	6
3	1	2	5

4 Redes de computadores



Planta da empresa - Story Polaroid

1-Recepção
2-RH
3-Sala de reunião
4-Refeitorio
5-Cozinha
6-Produção de Fotos
7-Produção de Quadros
8-Produção de Chaveiros Personalizados
9-Produção de Cubos Giratórios de Fotos Personalizados
10-Prospecção de clientes
11-Banheiro

Configuração de IP de todos os equipamentos

Recepção: 192.168.0.1 a 192.168.0.4
RH: 192.168.0.5
Sala de Reunião: 192.168.0.6 a 192.168.0.7
Produção de Fotos: 192.168.0.8 a 192.168.0.10
Produção de Quadros: 192.168.0.11 a 192.168.0.12
Produção de Chaveiros: 192.168.0.13
Produção de Cubo: 192.168.0.14 a 192.168.0.15
Prospecção de Clientes: 192.168.0.16 a 192.168.0.18

5 Segurança da informação

1. **Entrega 1: Análise de Riscos**
   Identificação e avaliação dos riscos de segurança para a empresa.

   Roubou ou perda de dados: Dados de clientes, como fotos e informações pessoais, podem ser perdidos ou roubados.

   Acesso não autorizado: Pessoas não autorizadas podem acessar as instalações ou sistemas da empresa.

   Ataques Cibernéticos: Ataques como ransomware, phishing e malware podem comprometer os sistemas da empresa.

   Vazamento de informações: Informações sensíveis podem ser divulgadas inadvertidamente.

   Danos físicos: Danos aos equipamentos ou instalações podem interromper as operações.

   Fraudo ou uso indevido de informações: Funcionários podem usar informações dos clientes de forma inadequada.

   Análise de vulnerabilidades e ameaças potenciais.

   Infraestrutura de TI desatualizada: Equipamentos e softwares desatualizados podem ter vulnerabilidades conhecidas que não foram corrigidas.

   Falta de controle e de acesso: Sistemas que não implementam controle de acesso adequado podem permitir que pessoas não autorizadas obtenham informações sensíveis.

   Ausência de políticas de segurança: A falta de políticas claras de segurança pode resultar em práticas inconsistentes e descuidadas por parte dos funcionários.

   Backup insuficiente: Falta de backups regulares e confiáveis pode resultar na perda permanente de dados.

   Uso de redes Wi-Fi inseguras: Conexões Wi-Fi não seguras podem ser exploradas por atacantes para interceptar dados

   Falta de treinamento em segurança: Funcionários mal informados são mais suscetíveis a erros que podem levar a brechas de segurança.
   Ataques ransomware: Atacantes podem bloquear o acesso aos dados e exigir um resgate para restaurá-los.

Ataques de engenharia social: Emails ou mensagens falsas que tentam enganar os funcionários para revelar informações sensíveis ou instalar malware.

Ataques de negação de serviço: Atacantes sobrecarregam os sistemas da empresa, tornando-os inacessíveis para usuários legítimos.

Roubo ou perda de dispositivo: Dispositivos como laptops ou smartphones contendo informações sensíveis podem ser perdidos ou roubados.

Insider threats: Funcionários descontentes ou mal-intencionados podem comprometer a segurança intencionalmente.

Vazamento de informações: Informações sensíveis podem ser divulgadas inadvertidamente ou intencionalmente

2. **Entrega 2: Implementação de Medidas de Segurança**
   Implementação de políticas de controle de acesso aos sistemas e dados.

   Avaliação de necessidades: Identifique quais dados e sistemas são críticos para o negócio e quem precisa de acesso a eles.

   Definição de níveis de acesso: Estabeleça diferentes níveis de acesso baseados nas funções dos usuários, como administrador, funcionário regular, etc.

   Seleção de tecnologias: Escolha as ferramentas e tecnologias adequadas para implementar o controle de acesso, como sistemas de gerenciamento de identidade e acesso.

   Desenvolvimento de políticas: Crie políticas claras que definam quem pode acessar o quê, quando e como.

   Treinamento dos funcionários: Eduque os funcionários sobre as novas políticas de controle de acesso e por que elas são importantes.

   Implementação técnica: Configure as ferramentas e tecnologias escolhidas de acordo com as políticas definidas.

   Monitoramento e auditoria: Monitore regularmente o acesso aos sistemas e dados e realize auditorias para garantir conformidade com as políticas.

   Senha forte: Exigir senhas fortes e a alteração regular das mesmas.
   Procedimentos de acesso: Definir como os usuários devem solicitar acesso, como ele será concedido e revogado.

Registros e auditoria: Registrar todas as tentativas e atividades de acesso para fins de auditoria e conformidade.

Políticas de saída: Estabelecer procedimentos para revogar o acesso quando um funcionário deixa a empresa ou muda de função.

Configuração de sistemas de detecção de intrusão e prevenção de ataques.

Avaliação de requisitos: Identifique as necessidades específicas da empresa, como tipos de tráfego, sistemas e aplicações que precisam ser monitorados.

Seleção de tecnologia: Escolha a solução de IDS/IPS adequada às necessidades da empresa, considerando fatores como escalabilidade, compatibilidade e recursos.

Planejamento de implementação: Defina a arquitetura do sistema, incluindo a localização dos sensores, configuração de políticas e procedimentos de resposta a incidentes.

Configuração e tuning: Configure as regras e políticas de detecção/prevenção para se alinhar com o perfil de risco da empresa e minimizar falsos positivos.

Interação com outros sistemas: Integre o IDS/IPS com outros sistemas de segurança, como firewalls, sistemas de gerenciamento de eventos e informações de segurança (SIEM) e sistemas de gerenciamento de vulnerabilidades.

Testes e validação: Realize testes de validação para garantir que o IDS/IPS esteja funcionando conforme esperado e detectando/prevenindo ataques de forma eficaz.

Monitoramento e manutenção: Monitore continuamente o desempenho do IDS/IPS, aplique atualizações de segurança e realize manutenções regulares.

Atualização regular: Mantenha as assinaturas de ameaças e regras de detecção atualizadas para garantir a eficácia contra as ameaças mais recentes.

Segmentação de rede: Separe a rede em segmentos para limitar o tráfego entre diferentes partes da rede e reduzir a superfície de ataque.

Logging e armazenamento de dados: Configure o armazenamento de logs de maneira adequada para suportar a análise forense e conformidade com regulamentações.

Resposta automatizada: Configure ações automáticas para bloquear tráfego malicioso em tempo real, conforme as políticas de prevenção configuradas.

Formação de equipe: Treine a equipe de segurança e operações para interpretar os alertas do IDS/IPS, investigar incidentes e responder adequadamente.

Auditoria e revisão: Realize auditorias regulares e revisões de desempenho para garantir a conformidade com as políticas e identificar áreas de melhoria.

Backup e recuperação: Estabeleça procedimentos de backup e recuperação para os sistemas de IDS/IPS para garantir a continuidade das operações em caso de falhas ou incidentes.