

PROJETO

Gestão da Qualidade no Processo de Plantio Agrícola

Joziane Oliveira

Analista de Sistemas | Implantação | Processos | Dados | Requisitos

Estudo de caso completo envolvendo mapeamento de processos (BPMN), requisitos, regras de negócio, modelagem de dados, indicadores e dashboard gerencial.

Projeto desenvolvido para fins de portfólio profissional

Gestão da Qualidade no Processo de Plantio Agrícola	3
Contexto do projeto	3
Objetivo do Projeto	3
Pontos-chave	3
Escopo do projeto	3
Escopo contemplado	3
Fora do escopo	4
Stakeholders	4
Visão Geral do Processo	4
Pontos do Processo	4
Requisitos	5
Requisitos Funcionais	5
Requisitos Não Funcionais	5
Regras de Avaliação	5
Tipos de Regras	5
Incidência	5
Exemplos de Regras	6
Faixas de Tolerância	6
Solução Proposta	6
Benefícios Esperados	6
Indicadores de Qualidade e Desempenho	6
Resultados Esperados	7
Processo de Plantio BPMN — AS IS x TO BE	7
Processo de Plantio Agrícola – AS IS	8
Processo de Plantio Agrícola – TO BE	9
DER — Modelo de Dados (conceitual)	10
Visão Geral	10
Entidades Principais	10
DerDiagram	12
Script SQL	14
DER	16
QUERIES SQL — INDICADORES DO PROCESSO DE PLANTIO	17
Considerações Finais	19

Gestão da Qualidade no Processo de Plantio Agrícola

Contexto do projeto

O plantio agrícola manual é um processo crítico que envolve múltiplas etapas operacionais, diferentes perfis de colaboradores e variáveis que impactam diretamente a produtividade e a qualidade final da cultura.

A ausência de um controle estruturado pode gerar diversos problemas, tais como:

- Desperdício de sementes;
- Falhas na execução do plantio;
- Dificuldade na avaliação da qualidade do serviço;
- Falta de dados confiáveis para apoiar a tomada de decisão gerencial.

Objetivo do Projeto

Este projeto tem como objetivo mapear o processo de plantio agrícola, identificar pontos de controle de qualidade e propor uma solução sistêmica que permita o acompanhamento da produção, da qualidade e da performance das equipes de campo.

Pontos-chave

- Mapear etapas do processo de plantio;
- Identificar pontos de controle e métricas de qualidade;
- Propor ferramentas e procedimentos para coleta de dados;
- Permitir rastreabilidade e suporte à decisão gerencial.

Escopo do projeto

Escopo contemplado

- Mapeamento do processo de plantio agrícola do recebimento das sementes ao pós-plantio
- Identificação de pontos críticos e oportunidades de controle
- Levantamento e documentação de requisitos
- Definição de regras de negócio para avaliação de qualidade
- Proposta de solução sistêmica
- Definição de indicadores de desempenho e qualidade

Fora do escopo

- Desenvolvimento completo da aplicação
- Integrações com sistemas reais de terceiros
- Uso de dados reais de empresas

Stakeholders

Stakeholder	Papel no processo
Gestor Agrícola	Acompanhar produtividade, qualidade e indicadores
Apontador	Registrar produção e qualidade
Plantador	Executar o plantio
Caixeiro	Distribuir sementes
Varredor	Cobrir sementes
Analista de Sistemas	Mapear processos, regras e propor soluções

Visão Geral do Processo

O processo de plantio agrícola foi dividido em sete etapas principais, chamadas de pontos de controle, que permitem acompanhar a operação de forma estruturada e mensurável.

Pontos do Processo

1. Chegada das sementes ao campo
2. Transporte interno das sementes

3. Distribuição nos pontos de coleta
4. Coleta das sementes pelos plantadores
5. Plantio
6. Cobertura das sementes
7. Avaliação pós-plantio

Cada ponto foi analisado considerando atividades executadas, riscos operacionais e oportunidades de controle por meio de sistema.

Requisitos

Requisitos Funcionais

- RF01 – Registrar entrada de sementes no campo
- RF02 – Controlar quantidade ou peso de sementes
- RF03 – Registrar distribuição de sementes
- RF04 – Apontar produção individual
- RF05 – Avaliar qualidade do plantio
- RF06 – Registrar desperdícios
- RF07 – Calcular bonificação
- RF08 – Avaliar pós-plantio

Requisitos Não Funcionais

- RNF01 – Usabilidade em ambiente rural
- RNF02 – Rastreabilidade das informações
- RNF03 – Confiabilidade dos dados
- RNF04 – Segurança de acesso

Regras de Avaliação

As avaliações de qualidade podem ou não impactar a remuneração dos colaboradores, dependendo do modelo definido pelo cliente.

Tipos de Regras

- Sim/Não
- Numérica

Incidência

- Valor fixo
- Porcentagem sobre a produção

Exemplos de Regras

- Plantio raso ou fundo
- Semente deitada
- Semente virada
- Cova sem plantio
- Espaçamento inadequado

Faixas de Tolerância

- 0 a 2 erros: 100% da bonificação
- 3 a 4 erros: 50%
- Acima de 4 erros: 0%

Solução Proposta

A solução proposta consiste em um sistema de apoio à gestão da qualidade do plantio agrícola, composto por:

- Aplicativo mobile para apontamentos em campo
- Backend para regras de negócio e cálculos
- Banco de dados relacional
- Painéis de indicadores gerenciais

Benefícios Esperados

- Padronização do processo
- Redução de desperdícios
- Transparência na avaliação
- Apoio à tomada de decisão

Indicadores de Qualidade e Desempenho

- Quantidade de sementes utilizadas
- Índice de desperdício
- Qualidade do plantio por colaborador
- Produtividade individual

- Percentual de germinação
- Valor de bonificação por colaborador

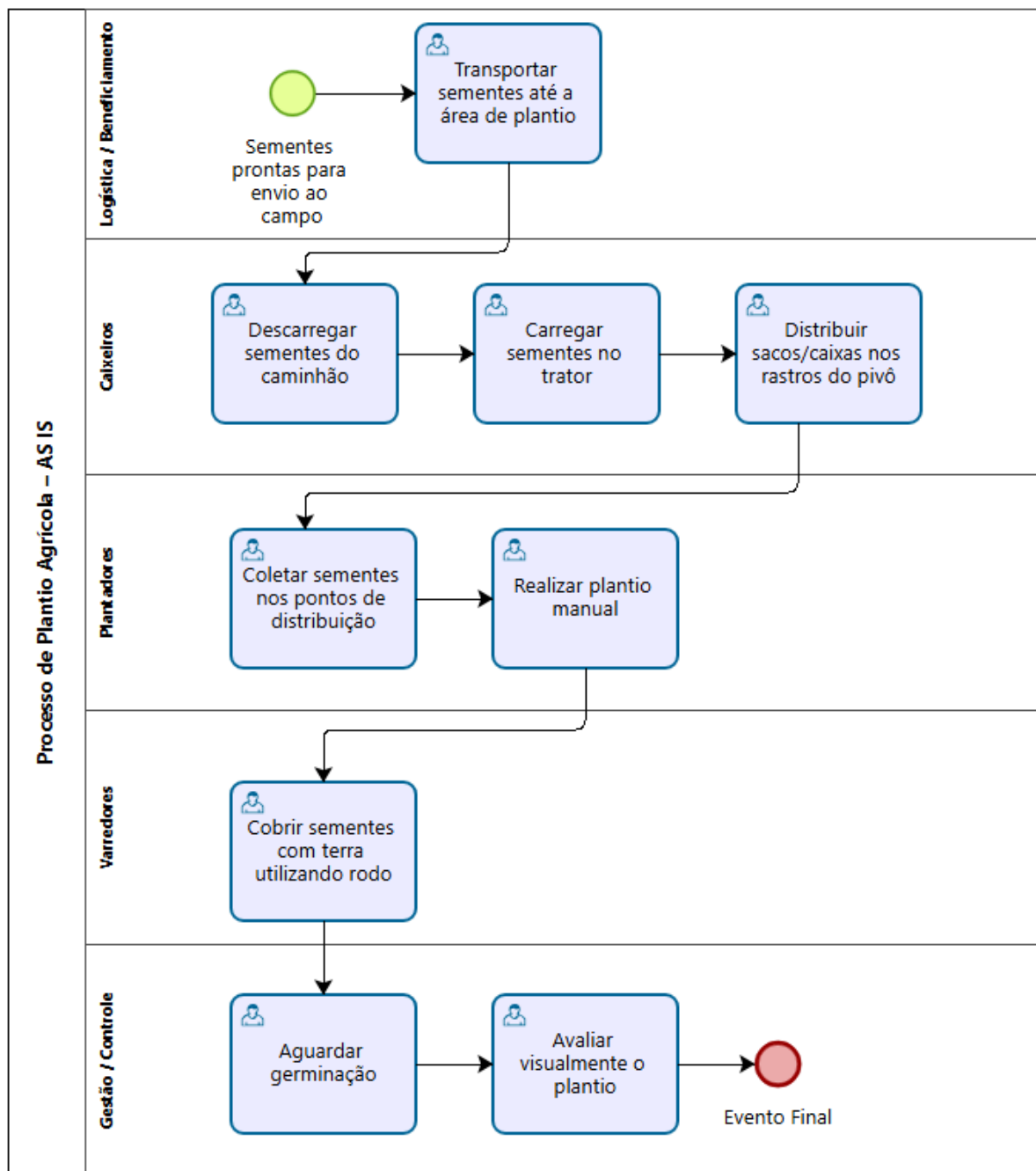
Resultados Esperados

- Melhoria da qualidade do plantio
- Redução de perdas operacionais
- Aumento da produtividade
- Maior controle gerencial
- Base de dados confiável para decisões futuras

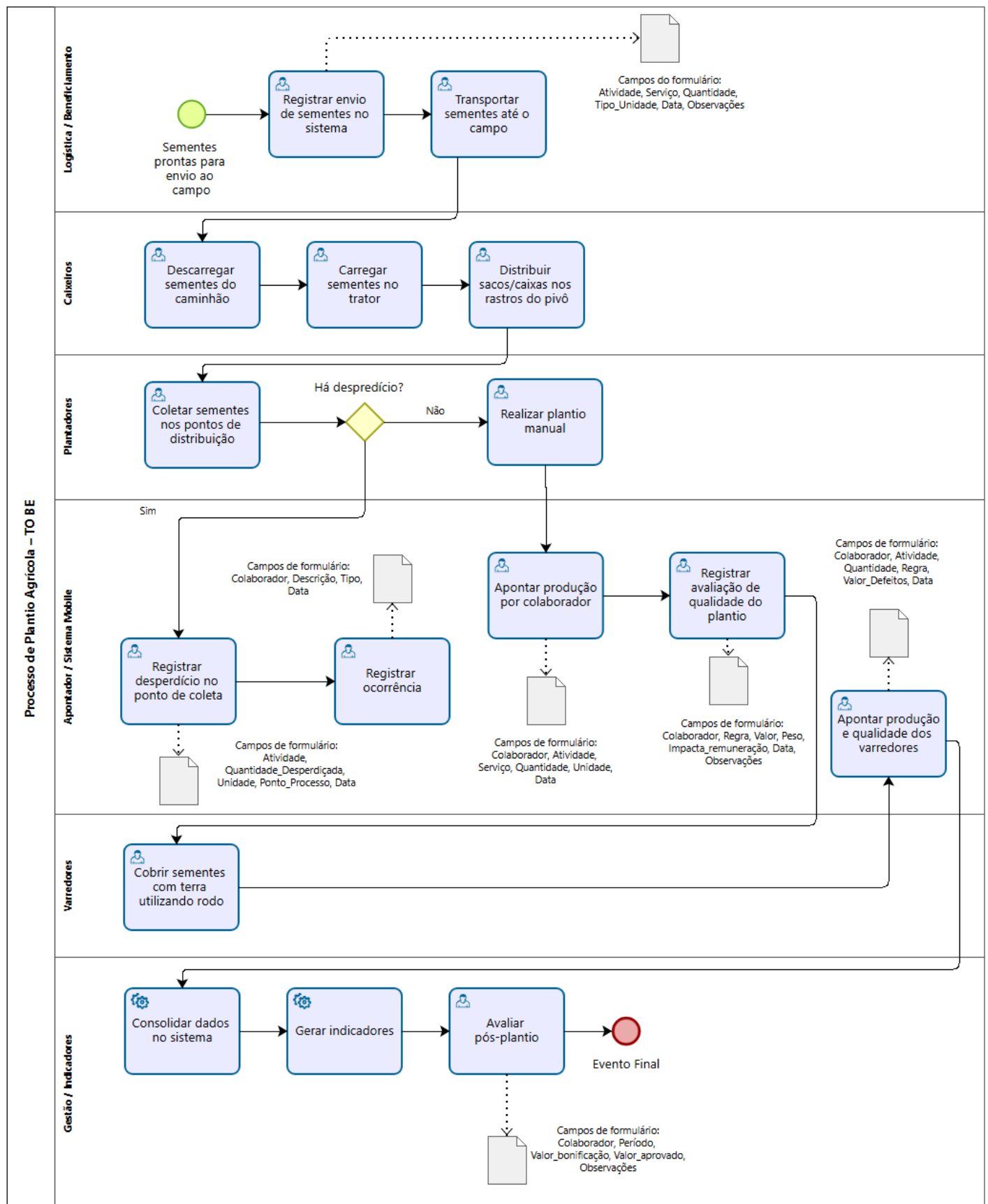
Processo de Plantio BPMN — AS IS x TO BE

AS IS (Situação Atual)	TO BE (Situação Proposta)
Processo manual	Processo monitorado
Avaliação visual	Avaliação sistêmica
Sem indicadores	Indicadores gerenciais
Pouca rastreabilidade	Dados rastreáveis
Decisão empírica	Decisão baseada em dados

Processo de Plantio Agrícola – AS IS



Processo de Plantio Agrícola – TO BE



DER — Modelo de Dados (conceitual)

Visão Geral

O modelo foi pensado para:

- Rastrear quem fez
- o que foi feito
- em qual etapa
- com qual qualidade
- com impacto ou não na remuneração

Entidades Principais

COLABORADOR

Representa qualquer pessoa envolvida no processo (plantador, caixeiro, varredor).

Campos principais:

- id_colaborador (PK)
- nome
- funcao
- ativo

TURMA

Agrupa colaboradores que atuam juntos no plantio.

- id_turma (PK)
- nome_turma
- data_inicio
- data_fim

Relacionamento:

Turma 1 — N Colaborador

SERVICO_CAMPO

Representa uma atividade produtiva (ex: plantio de alho).

- id_servico (PK)
- descricao
- tipo_unidade (ex: metro)
- valor_unitario

ATIVIDADE

Etapa do processo (Plantio, Cobertura, Pós-plantio).

- id_atividade (PK)

- nome
- ponto_processo (1 a 7)

Serviço_Campo 1 — N Atividade

APONTAMENTO_PRODUCAO

Registro da produção realizada pelo colaborador.

- id_apontamento (PK)
- id_colaborador (FK)
- id_atividade (FK)
- quantidade
- data_apontamento

MODELO_AVALIACAO

Define o modelo de avaliação de qualidade.

- id_modelo (PK)
- descricao
- impacta_remuneracao (boolean)
- tipo_incidencia (valor / porcentagem)

REGRA_AVALIACAO

Cada critério avaliado (raso, fundo, deitado etc.).

- id_regra (PK)
- id_modelo (FK)
- descricao
- tipo_regra (sim_nao / numerica)
- peso
- tolerancia_min
- tolerancia_max

APONTAMENTO_AVALIACAO

Registro da avaliação feita no campo.

- id_avaliacao (PK)
- id_colaborador (FK)
- id_regra (FK)
- valor_apontado
- data_avaliacao

BONIFICACAO

Resultado financeiro da avaliação.

- id_bonificacao (PK)
- id_colaborador (FK)
- valor
- data_referencia

Relacionamentos (resumo)

- Colaborador → Turma
- Colaborador → Apontamento_Produção

- Colaborador → Apontamento_Avaliação
- Modelo_Avaliação → Regra_Avaliação
- Atividade → Apontamento_Produção

DerDiagram

TURMA ||--o{ COLABORADOR : possui

COLABORADOR ||--o{ APONTAMENTO_PRODUCAO : realiza

ATIVIDADE ||--o{ APONTAMENTO_PRODUCAO : registra

SERVICO_CAMPO ||--o{ ATIVIDADE : compoe

MODELO_AVALIACAO ||--o{ REGRA_AVALIACAO : define

COLABORADOR ||--o{ APONTAMENTO_AVALIACAO : recebe

REGRA_AVALIACAO ||--o{ APONTAMENTO_AVALIACAO : aplica

COLABORADOR ||--o{ BONIFICACAO : gera

```
TURMA {
  int id_turma PK
  string nome_turma
  date data_inicio
  date data_fim
}
```

```
COLABORADOR {
  int id_colaborador PK
  string nome
  string funcao
  boolean ativo
  int id_turma FK
}
```

```
SERVICO_CAMPO {
  int id_servico PK
  string descricao
  string tipo_unidade
  decimal valor_unitario
}
```

```
ATIVIDADE {
  int id_atividade PK
  string nome
}
```

```
    int ponto_processo
    int id_servico FK
}
```

```
APONTAMENTO_PRODUCAO {
    int id_apontamento PK
    int id_colaborador FK
    int id_atividade FK
    decimal quantidade
    date data_apontamento
}
```

```
MODELO_AVALIACAO {
    int id_modelo PK
    string descricao
    boolean impacta_remuneracao
    string tipo_incidencia
}
```

```
REGRA_AVALIACAO {
    int id_regra PK
    int id_modelo FK
    string descricao
    string tipo_regra
    decimal peso
    int tolerancia_min
    int tolerancia_max
}
```

```
APONTAMENTO_AVALIACAO {
    int id_avaliacao PK
    int id_colaborador FK
    int id_regra FK
    int valor_apontado
    date data_avaliacao
}
```

```
BONIFICACAO {
    int id_bonificacao PK
    int id_colaborador FK
    decimal valor
    date data_referencia
}
```

Script SQL

```
CREATE TABLE turma (  
    id_turma INT AUTO_INCREMENT PRIMARY KEY,  
    nome_turma VARCHAR(100),  
    data_inicio DATE,  
    data_fim DATE  
);
```

```
CREATE TABLE colaborador (  
    id_colaborador INT AUTO_INCREMENT PRIMARY KEY,  
    nome VARCHAR(100),  
    funcao VARCHAR(50),  
    ativo BOOLEAN,  
    id_turma INT,  
    FOREIGN KEY (id_turma) REFERENCES turma(id_turma)  
);
```

```
CREATE TABLE servico_campo (  
    id_servico INT AUTO_INCREMENT PRIMARY KEY,  
    descricao VARCHAR(100),  
    tipo_unidade VARCHAR(30),  
    valor_unitario DECIMAL(10,2)  
);
```

```
CREATE TABLE atividade (  
    id_atividade INT AUTO_INCREMENT PRIMARY KEY,  
    nome VARCHAR(100),  
    ponto_processo INT,  
    id_servico INT,  
    FOREIGN KEY (id_servico) REFERENCES servico_campo(id_servico)  
);
```

```
CREATE TABLE apontamento_producao (  
    id_apontamento INT AUTO_INCREMENT PRIMARY KEY,  
    id_colaborador INT,  
    id_atividade INT,  
    quantidade DECIMAL(10,2),  
    data_apontamento DATE,  
    FOREIGN KEY (id_colaborador) REFERENCES colaborador(id_colaborador),  
    FOREIGN KEY (id_atividade) REFERENCES atividade(id_atividade)  
);
```

```
CREATE TABLE modelo_avaliacao (  
    id_modelo INT AUTO_INCREMENT PRIMARY KEY,  
    descricao VARCHAR(100),  
    impacta_remuneracao BOOLEAN,  
    tipo_incidencia VARCHAR(20)
```

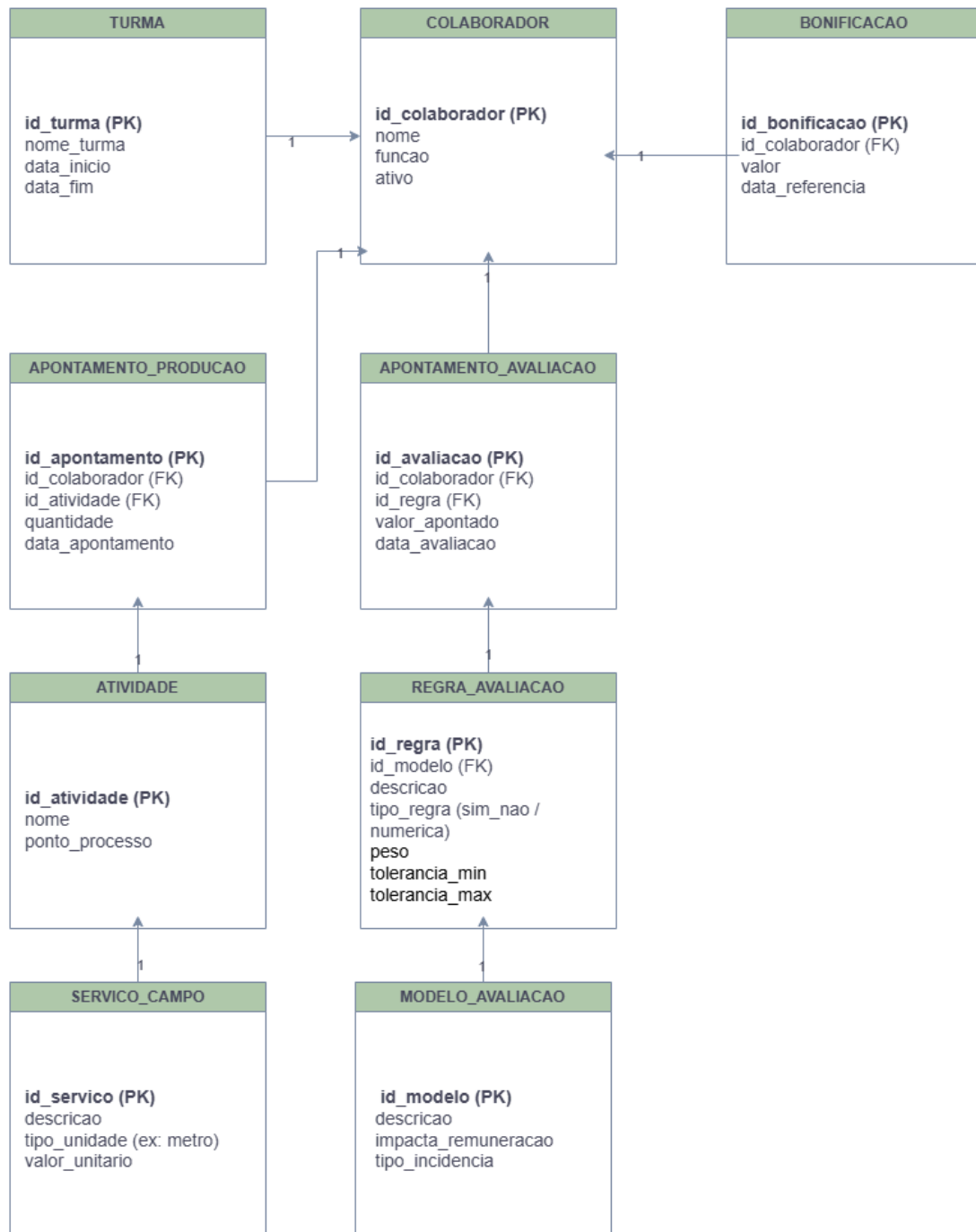
);

```
CREATE TABLE regra_avaliacao (  
  id_regra INT AUTO_INCREMENT PRIMARY KEY,  
  id_modelo INT,  
  descricao VARCHAR(100),  
  tipo_regra VARCHAR(20),  
  peso DECIMAL(5,2),  
  tolerancia_min INT,  
  tolerancia_max INT,  
  FOREIGN KEY (id_modelo) REFERENCES modelo_avaliacao(id_modelo)  
);
```

```
CREATE TABLE apontamento_avaliacao (  
  id_avaliacao INT AUTO_INCREMENT PRIMARY KEY,  
  id_colaborador INT,  
  id_regra INT,  
  valor_apontado INT,  
  data_avaliacao DATE,  
  FOREIGN KEY (id_colaborador) REFERENCES colaborador(id_colaborador),  
  FOREIGN KEY (id_regra) REFERENCES regra_avaliacao(id_regra)  
);
```

```
CREATE TABLE bonificacao (  
  id_bonificacao INT AUTO_INCREMENT PRIMARY KEY,  
  id_colaborador INT,  
  valor DECIMAL(10,2),  
  data_referencia DATE,  
  FOREIGN KEY (id_colaborador) REFERENCES colaborador(id_colaborador)  
);
```

DER



QUERIES SQL — INDICADORES DO PROCESSO DE PLANTIO

1. Produção total por colaborador

Quanto cada colaborador produziu no período

```
SELECT
  c.id_colaborador,
  c.nome,
  SUM(ap.quantidade) AS total_produzido
FROM colaborador c
JOIN apontamento_producao ap
  ON c.id_colaborador = ap.id_colaborador
GROUP BY c.id_colaborador, c.nome
ORDER BY total_produzido DESC;
```

2. Produção por atividade (Plantio, Cobertura, etc.)

Quanto foi produzido em cada etapa do processo

```
SELECT
  a.nome AS atividade,
  SUM(ap.quantidade) AS total_produzido
FROM atividade a
JOIN apontamento_producao ap
  ON a.id_atividade = ap.id_atividade
GROUP BY a.nome;
```

3. Índice de erros por colaborador (qualidade)

Total de erros registrados nas avaliações

```
SELECT
  c.id_colaborador,
  c.nome,
  SUM(aa.valor_apontado) AS total_erros
FROM colaborador c
JOIN apontamento_avaliacao aa
  ON c.id_colaborador = aa.id_colaborador
GROUP BY c.id_colaborador, c.nome
ORDER BY total_erros ASC;
```

4. Erros por tipo de regra

Quais erros ocorrem com mais frequência

```
SELECT
  r.descricao AS regra,
  SUM(aa.valor_apontado) AS total_ocorrencias
FROM regra_avaliacao r
JOIN apontamento_avaliacao aa
```

```
ON r.id_regra = aa.id_regra
GROUP BY r.descricao
ORDER BY total_ocorrencias DESC;
```

5. Qualidade média por colaborador (normalizada)

Indicador simples de qualidade

```
SELECT
  c.id_colaborador,
  c.nome,
  ROUND(AVG(aa.valor_apontado), 2) AS media_erros
FROM colaborador c
JOIN apontamento_avaliacao aa
  ON c.id_colaborador = aa.id_colaborador
GROUP BY c.id_colaborador, c.nome
ORDER BY media_erros ASC;
```

6. Bonificação total por colaborador

Valor financeiro gerado pelas avaliações

```
SELECT
  c.id_colaborador,
  c.nome,
  SUM(b.valor) AS total_bonificacao
FROM colaborador c
JOIN bonificacao b
  ON c.id_colaborador = b.id_colaborador
GROUP BY c.id_colaborador, c.nome
ORDER BY total_bonificacao DESC;
```

7. Produtividade x Qualidade (visão gerencial)

Cruza produção e erros

```
SELECT
  c.id_colaborador,
  c.nome,
  SUM(ap.quantidade) AS total_produzido,
  SUM(aa.valor_apontado) AS total_erros
FROM colaborador c
LEFT JOIN apontamento_producao ap
  ON c.id_colaborador = ap.id_colaborador
LEFT JOIN apontamento_avaliacao aa
  ON c.id_colaborador = aa.id_colaborador
GROUP BY c.id_colaborador, c.nome
ORDER BY total_produzido DESC;
```

8. Produção por turma

Desempenho coletivo

```

SELECT
    t.nome_turma,
    SUM(ap.quantidade) AS total_produzido
FROM turma t
JOIN colaborador c
    ON t.id_turma = c.id_turma
JOIN apontamento_producao ap
    ON c.id_colaborador = ap.id_colaborador
GROUP BY t.nome_turma;

```

9. Qualidade por turma

Erros totais por equipe

```

SELECT
    t.nome_turma,
    SUM(aa.valor_apontado) AS total_erros
FROM turma t
JOIN colaborador c
    ON t.id_turma = c.id_turma
JOIN apontamento_avaliacao aa
    ON c.id_colaborador = aa.id_colaborador
GROUP BY t.nome_turma
ORDER BY total_erros ASC;

```

10. Ranking de colaboradores (performance geral)

Produção alta + poucos erros

```

SELECT
    c.id_colaborador,
    c.nome,
    SUM(ap.quantidade) AS producao,
    SUM(aa.valor_apontado) AS erros,
    (SUM(ap.quantidade) - SUM(aa.valor_apontado)) AS score_performance
FROM colaborador c
LEFT JOIN apontamento_producao ap
    ON c.id_colaborador = ap.id_colaborador
LEFT JOIN apontamento_avaliacao aa
    ON c.id_colaborador = aa.id_colaborador
GROUP BY c.id_colaborador, c.nome
ORDER BY score_performance DESC;

```

Considerações Finais

Este projeto demonstra a importância do mapeamento de processos e da definição clara de regras de negócio como base para o sucesso de soluções tecnológicas.

O papel do Analista de Sistemas se mostra fundamental ao atuar como elo entre operação, negócio e tecnologia, garantindo que as soluções atendam às reais necessidades do processo produtivo.