**Engenharia de Software**

2023-01

|  |  |
| --- | --- |
| **Professor(a)(es)** | RADAMÉS PEREIRA |

**UNIVERSIDADE COMUNITÁRIA REGIONAL DE CHAPECÓ - UNOCHAPECÓ**

**ÁREA DE CIÊNCIAS EXATAS E AMBIENTAIS**

**CIÊNCIA DA COMPUTAÇÃO**

Projeto da Fábrica de Concreto

Jacson de Vargas Erthal

Chapecó - SC, 08 de junho de 2023

Introdução

## Resumo do Projeto (Descrição textual)

O sistema permite que os clientes preencham um formulário de pedido, fornecendo informações sobre o tipo e quantidade de concreto desejado para entrega. Esses pedidos são escalonados pelo departamento de vendas e encaminhados, quando necessário, ao encarregado de produção. O sistema também permite que o encarregado da produção acesse as informações dos pedidos e solicite os materiais necessários ao departamento de estoque.

O operador de máquinas é responsável pela execução dos controles de fabricação, contando com o auxílio dos operários responsáveis pela mistura dos componentes e do funcionário encarregado do controle de fluxo de trabalho. Caso algum produto da matéria-prima fique abaixo do nível de segurança, o encarregado da produção emite uma ordem para o departamento de compras providenciar o pedido dos materiais. Após a entrega, o setor de recebimento e distribuição verifica se a mercadoria foi entregue conforme solicitado.

O sistema também inclui um módulo de transporte, permitindo que a transportadora da empresa visualize as informações dos pedidos e organize as rotas de entrega. Após a fabricação, o produto é entregue na construção e o cliente realiza o pagamento, que é registrado pelo departamento financeiro da indústria.

O sistema possui um módulo de estoque, que mantém um registro atualizado das matérias-primas, como cimento, areia, cascalho e água. O encarregado do almoxarifado pode solicitar o reabastecimento das matérias-primas quando necessário.

No aspecto financeiro, o sistema registra todas as transações de pagamento realizadas pelos clientes e fornece ao departamento financeiro acesso em tempo real às informações dessas transações.

O sistema deve ter capacidade para lidar com um grande volume de pedidos e transações simultaneamente, garantindo um tempo de resposta máximo de 5 segundos. Medidas de segurança, como criptografia de dados e autenticação de usuário, são implementadas para proteger o sistema. Além disso, é fundamental que o sistema seja fácil de usar e intuitivo para os usuários, possuindo uma interface amigável e clara. A confiabilidade do sistema é uma prioridade, a fim de garantir a segurança das informações dos clientes e das transações financeiras.

## Plataforma de desenvolvimento (O equipamento dos desenvolvedores e ferramentas de software)

A PlantUML é uma ferramenta de diagramação que permite criar diagramas UML (Unified Modeling Language) usando uma sintaxe simples baseada em texto. Para desenvolver um projeto de engenharia de software, você pode criar vários tipos de diagramas usando a PlantUML.

## Plataforma de operação (O equipamento do cliente/usuário do sistema)

* + 1. **Dispositivos de Computação:** São utilizados para executar o software de controle e monitoramento da fábrica. Esses dispositivos podem variar entre computadores desktop, laptops e até dispositivos móveis, de acordo com as necessidades e preferências do usuário.
    2. **Software de Controle:** Trata-se de um programa específico desenvolvido para gerenciar os diferentes aspectos da produção de concreto. Esse software inclui recursos como o monitoramento da dosagem de ingredientes, controle da mistura, supervisão de temperaturas e pressões, entre outros. Ele oferece uma interface intuitiva para que o usuário possa interagir e controlar os processos de produção de forma eficiente.
    3. **Interfaces e Painéis de Controle:** São dispositivos de exibição que fornecem informações cruciais sobre o processo de produção de concreto, como fluxo de ingredientes, dosagens, níveis de estoque, temperatura, pressão, entre outros. Essas interfaces e painéis de controle podem ser interativos e sensíveis ao toque, permitindo que o usuário faça ajustes e modificações conforme necessário.
    4. **Sensores e Dispositivos de Monitoramento:** São empregados para coletar dados em tempo real sobre várias variáveis do processo de produção, como temperatura, umidade, pressão e níveis de ingredientes. Esses sensores estão conectados à plataforma de operação e fornecem informações precisas ao software de controle.

## Perspectiva do produto

### Modos de operação (Meios de acesso ao sistema/Arquitetura do sistema)

1. **Modo Back-End-Front-End:** Neste modo de operação, a fábrica de concreto possui uma arquitetura de sistema distribuída em que existe uma separação clara entre o back-end e o front-end. O back-end é responsável pelo processamento de dados, armazenamento e gerenciamento das operações principais da fábrica, como controle de estoque, programação de produção e gestão de pedidos.
2. **Modo Stand-Alone:** o sistema da fábrica de concreto é projetado para operar independentemente, sem a necessidade de conexão com uma rede externa. Nesse caso, todas as funcionalidades essenciais são executadas localmente, sem dependência de serviços online. Esse modo pode ser útil em situações em que a conectividade é limitada ou em casos de contingência em que a conexão com a rede pode ser interrompida.

## Funções do produto (Funções básicas, R1.1 ..., R1.2 ...)

Visão geral do sistema

O sistema gerencia a produção e venda de concreto em uma fábrica, incluindo as áreas de vendas, escritório e planta industrial.

Os clientes fazem pedidos de concreto, que são escalonados pelo departamento de vendas e produzidos pela planta industrial.

A produção é gerenciada pela área de produção e de matéria-prima, com auxílio de operários, controles de fabricação e fluxo de trabalho.

R 1. Requisitos funcionais:

R 1.1 O sistema deve permitir que os clientes preencham uma ordem de pedido com o tipo e quantidade de concreto desejado.

R 1.2 O sistema deve permitir que o departamento de vendas escalone os pedidos e os encaminhe para a indústria.

R 1.3 O sistema deve permitir que a indústria produza o concreto a partir dos pedidos recebidos, solicitando os materiais necessários ao encarregado do almoxarifado e estocagem.

R 1.4 O sistema deve permitir que o operador das máquinas execute os controles de fabricação com o auxílio dos operários e do funcionário encarregado do controle de fluxo de trabalho.

R 1.5 O sistema deve permitir que o encarregado da produção emita ordens para o departamento de compras providenciar pedidos de mercadorias que estejam abaixo do nível de segurança.

R 1.6 O sistema deve permitir que o setor de recebimento e distribuição verifique se os pedidos de mercadorias foram entregues conforme solicitado.

R 1.7 O sistema deve permitir que o produto final seja entregue na construção e que o cliente efetue o pagamento que será recebido pelo departamento financeiro da indústria.

R 2. Requisitos não funcionais:

R 2.1 O sistema deve ter uma interface de usuário intuitiva e fácil de usar para permitir que os usuários façam seus pedidos e gerenciem a produção.

R 2.2 O sistema deve ser seguro, com acesso restrito aos dados e funcionalidades apenas por usuários autorizados.

R 2.3 O sistema deve ser escalável e capaz de lidar com muitos pedidos e produção de concreto.

R 2.4 O sistema deve ter um bom desempenho e ser capaz de processar os pedidos e produzir o concreto de forma rápida e eficiente.

R 2.5 O sistema deve ser confiável e estar disponível a maior parte do tempo, minimizando o tempo de inatividade para manutenção e atualizações.

## Características dos usuários (identificação da formação dos usuários para suas especialidades no uso do sistema, tal como ensino-médio, graduação, especialista, engenheiro, doutor, etc...)

1. Na indústria de fabricação de concreto, os profissionais envolvidos no sistema possuem diferentes formações educacionais, que variam de acordo com a natureza do trabalho e o nível de especialização necessário. Abaixo estão algumas possíveis formações educacionais para diferentes especialidades dentro de uma fábrica de concreto:
   1. **Operadores de Máquinas:** Para operar equipamentos como betoneiras, bombas de concreto e misturadores, pode ser exigido um diploma de ensino médio ou treinamento específico fornecido pela fábrica.
   2. **Técnicos em Concreto:** Esses profissionais geralmente possuem formação técnica ou cursos profissionalizantes relacionados à indústria da construção civil. Podem ter concluído um curso técnico em áreas como tecnologia do concreto, construção civil ou materiais de construção.
   3. **Engenheiros Civis:** Engenheiros civis desempenham um papel importante na fábrica de concreto, sendo responsáveis pelo projeto, controle de qualidade, desenvolvimento de misturas e supervisão geral do processo de produção. Eles normalmente possuem um diploma de graduação em engenharia civil ou em uma área relacionada.
   4. **Especialistas em Materiais de Construção:** Esses profissionais possuem um conhecimento aprofundado dos materiais utilizados na fabricação de concreto, incluindo aglomerantes, aditivos, agregados e outros componentes. Podem ter formação de graduação em engenharia de materiais, química ou áreas afins.
   5. **Pesquisadores em Concreto:** Pesquisadores nessa área geralmente possuem formação avançada, como um mestrado ou doutorado, em engenharia civil ou ciência dos materiais. Eles conduzem pesquisas para desenvolver novas técnicas, materiais e processos visando melhorar a qualidade e o desempenho do concreto.

## Restrições (Lei Geral de proteção de dados, etc...)

Uma fábrica de concreto deve cumprir uma série de restrições e diretrizes para garantir a proteção adequada dos dados pessoais que coleta, armazena e processa.

1. Consentimento: A fábrica de concreto deve obter o consentimento explícito dos indivíduos antes de coletar e processar seus dados pessoais. O consentimento deve ser livre, informado e específico para cada finalidade do processamento.

2. Finalidade específica: A coleta de dados pessoais pela fábrica de concreto deve ter uma finalidade específica e legítima. Os dados só devem ser utilizados para os fins declarados e não devem ser processados de maneira incompatível com esses propósitos.

3. Minimização de dados: A fábrica de concreto deve coletar apenas os dados pessoais necessários para atingir a finalidade declarada. Deve-se evitar a coleta excessiva ou desnecessária de informações.

4. Transparência: A fábrica de concreto deve fornecer informações claras e transparentes sobre como os dados pessoais são coletados, usados, compartilhados e armazenados. Isso geralmente é feito por meio de uma política de privacidade ou aviso de privacidade.

5. Segurança de dados: A fábrica de concreto deve implementar medidas técnicas e organizacionais adequadas para proteger os dados pessoais contra acesso não autorizado, perda, destruição ou divulgação. Isso pode envolver a criptografia dos dados, a restrição de acesso apenas a pessoas autorizadas e a implementação de políticas de segurança da informação.

6. Direitos dos titulares dos dados: A fábrica de concreto deve respeitar os direitos dos titulares dos dados, conforme estabelecido pela LGPD. Isso inclui o direito de acesso aos dados, o direito de retificação, o direito de exclusão, o direito de oposição ao processamento e outros direitos relevantes.

7. Transferência internacional de dados: Se a fábrica de concreto transferir dados pessoais para fora do Brasil, ela deve garantir que a transferência atenda aos requisitos legais, como a obtenção de consentimento ou a utilização de salvaguardas adequadas, como as cláusulas contratuais padrão.

8. Registro de atividades de processamento: A fábrica de concreto deve manter registros adequados de suas atividades de processamento de dados, incluindo informações sobre as finalidades do processamento, as categorias de dados envolvidos, os períodos de retenção, entre outros detalhes relevantes.

# Requisitos específicos

## Requisitos funcionais

### Diagramas de casos de uso (Modelo UML de Casos de Uso)

@startuml

left to right direction

actor "Cliente" as cliente

actor "Representante" as representante

actor "Encarregado de Produção" as encarregadoProducao

actor "Encarregado do Almoxarifado" as encarregadoAlmoxarifado

actor "Operador de Máquinas" as operadorMaquinas

actor "Funcionário de Controle de Fluxo" as funcionarioControleFluxo

actor "Departamento de Compras" as departamentoCompras

actor "Setor de Recebimento e Distribuição" as setorRecebimentoDistribuicao

actor "Departamento Financeiro" as departamentoFinanceiro

actor "Setor de Transporte" as setorTransporte

rectangle "Fábrica de Concreto" {

setorTransporte --> (Entregar pedido na obra)

cliente --> (Preencher ordem de pedido)

cliente --> (Efetuar pagamento)

representante --> (Entregar ordem de pedido ao cliente)

departamentoFinanceiro --> (Escalonar ordem de pedido)

departamentoFinanceiro --> (Encaminhar ordem de pedido à produção)

departamentoFinanceiro --> (Receber pagamento do cliente)

(Executar ordem de produção) <-- encarregadoProducao

encarregadoProducao --> (Emitir pedido de compra de matéria-prima)

encarregadoProducao --> (Solicitar materiais ao almoxarifado)

(Entregar materiais à produção) <-- encarregadoAlmoxarifado

(Executar controles de fabricação) <-- operadorMaquinas

(Realizar mistura dos componentes) <-- operadorMaquinas

funcionarioControleFluxo --> (Auxiliar na execução dos controles de fabricação)

(Providenciar pedido de mercadoria) <-- departamentoCompras

(Verificar entrega de mercadoria conforme pedido) <-- setorRecebimentoDistribuicao

(Preencher ordem de pedido) <.. (Escalonar ordem de pedido) : <<include>>

(Escalonar ordem de pedido) <.. (Encaminhar ordem de pedido à produção) : <<include>>

(Encaminhar ordem de pedido à produção) <.. (Executar ordem de produção) : <<include>>

(Solicitar materiais ao almoxarifado) <.. (Entregar materiais à produção) : <<include>>

(Executar controles de fabricação) <.. (Realizar mistura dos componentes) : <<extend>>

(Providenciar pedido de mercadoria) <.. (Verificar entrega de mercadoria conforme pedido) : <<include>>

(Emitir pedido de compra de matéria-prima) <.. (Providenciar pedido de mercadoria) : <<include>>

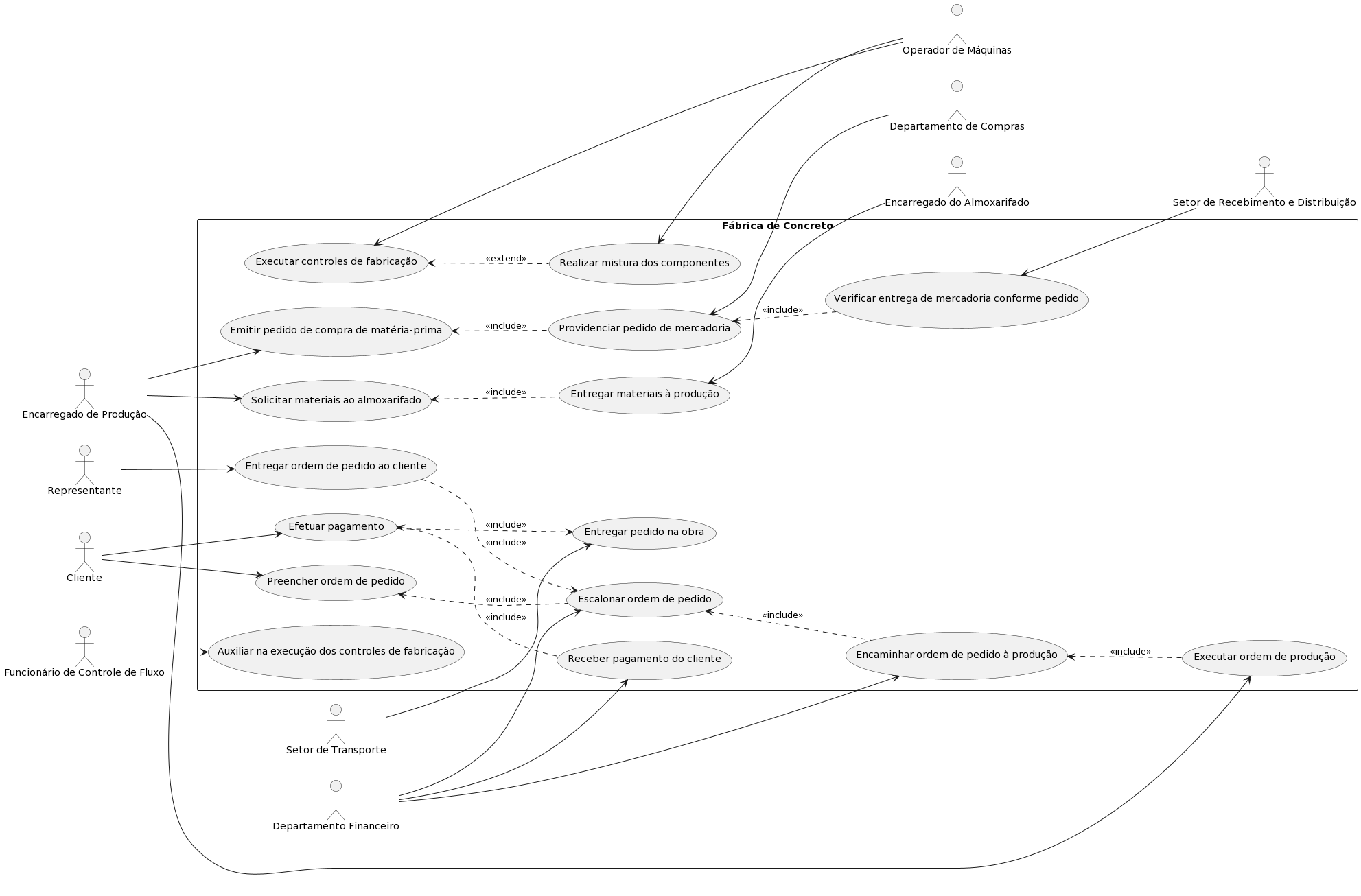
(Entregar ordem de pedido ao cliente) ..> (Escalonar ordem de pedido) : <<include>>

(Efetuar pagamento) <.. (Receber pagamento do cliente) : <<include>>

(Efetuar pagamento) ..> (Entregar pedido na obra) : <<include>>

}

@enduml



### Fluxos dos casos de uso (Casos de Uso Expandidos e Diagramas de Atividades)

1. **Caso de Uso: Gerenciar Pedidos e Produção de Concreto**

**Ator Principal:** Cliente

**Outros Atores:** Representante de Vendas, Encarregado de Produção, Encarregado do Almoxarifado e Estocagem, Operador das Máquinas, Funcionário Encarregado do Controle de Fluxo de Trabalho, Departamento de Compras, Setor de Recebimento e Distribuição, Departamento Financeiro.

**Objetivo:** Este caso de uso descreve o processo de gerenciamento de pedidos e produção de concreto na fábrica, desde o momento em que o cliente faz um pedido até a entrega do produto e o recebimento do pagamento pela empresa.

**Pré-condições:** A fábrica de concreto está operacional e possui os recursos necessários, incluindo receptáculos de armazenagem, misturador, balanças, válvula de medição, controle de descarga de água, esteiras de transporte e equipe de funcionários.

**Fluxo Básico:**

1. O cliente preenche uma ordem de pedido, fornecendo detalhes sobre o tipo e a quantidade de concreto necessários, bem como a data de entrega desejada.
2. O representante de vendas recebe a ordem de pedido e a escalona no sistema.
3. O representante de vendas encaminha a ordem de pedido ao encarregado de produção.
4. O encarregado de produção recebe a ordem de pedido e verifica a disponibilidade dos materiais necessários.
5. Caso algum dos produtos de matéria-prima esteja abaixo do nível de segurança, o encarregado de produção emite uma ordem de compra para o departamento de compras.
6. O departamento de compras recebe a ordem de compra, providencia a compra dos materiais em falta e os entrega no local.
7. O setor de recebimento e distribuição verifica se os materiais entregues estão conforme o pedido feito.
8. O encarregado do almoxarifado e estocagem entrega os materiais necessários à produção.
9. O operador das máquinas inicia os controles de fabricação, com o auxílio dos operários responsáveis pela mistura dos componentes e do funcionário encarregado do controle de fluxo de trabalho.
10. A produção executa a ordem de pedido, preparando o concreto de acordo com as especificações fornecidas.
11. Após a fabricação, o produto é entregue na construção, de acordo com a data agendada na ordem de pedido.
12. O cliente recebe o produto e efetua o pagamento.
13. O departamento financeiro registra o pagamento recebido.

**Fluxos Alternativos:**

1ª - Se a ordem de pedido contém informações incompletas ou inconsistentes, o representante de vendas pode entrar em contato com o cliente para obter os detalhes corretos antes de prosseguir com o processo.

3ª - Se a ordem de pedido não for aprovada pelo representante de vendas, ela pode ser rejeitada e o cliente será notificado.

4ª - Se os materiais necessários para a produção do concreto não estiverem disponíveis, o encarregado de produção pode informar ao representante de vendas sobre a impossibilidade de atender à ordem de pedido.

7ª - Se os materiais entregues não estiverem de acordo com o pedido feito, o setor de recebimento e distribuição notifica o departamento de compras para tomar as medidas necessárias.

11ª - Se a entrega do produto for adiada por algum motivo, o cliente será informado sobre a nova data de entrega.

12ª - Se o cliente não efetuar o pagamento dentro do prazo estabelecido, o departamento financeiro pode tomar as medidas apropriadas para cobrar o pagamento em atraso.

**2.    Caso de Uso: Gerenciar Estoque de Matéria-Prima**

**Ator Principal:** Encarregado de Produção

**Outros Atores:** Encarregado do Almoxarifado e Estocagem, Departamento de Compras

Objetivo: Este caso de uso descreve o processo de gerenciamento do estoque de matéria-prima na fábrica de concreto, incluindo o monitoramento dos níveis de estoque, a solicitação de reposição de materiais e a recepção das entregas.

**Pré-condições:** A fábrica de concreto está operacional, os materiais de matéria-prima estão armazenados corretamente no almoxarifado e estocagem, e há um sistema de gerenciamento de estoque em funcionamento.

**Fluxo Básico:**

1. O encarregado de produção acessa o sistema de gerenciamento de estoque.
2. O encarregado de produção verifica os níveis de estoque dos materiais de matéria-prima, incluindo cimento, areia, cascalho e água.
3. Se algum dos materiais estiver abaixo do nível de segurança, o encarregado de produção inicia o processo de reposição.
4. O encarregado de produção gera uma ordem de compra para o departamento de compras, especificando os materiais que precisam ser adquiridos.
5. O departamento de compras recebe a ordem de compra e inicia o processo de aquisição dos materiais.
6. Após a compra dos materiais, o departamento de compras programa a entrega dos mesmos.
7. O encarregado do almoxarifado e estocagem recebe os materiais entregues pelo fornecedor.
8. O encarregado do almoxarifado e estocagem verifica se os materiais entregues estão de acordo com o pedido feito.
9. Os materiais são devidamente armazenados no local apropriado do almoxarifado e estocagem.
10. O sistema de gerenciamento de estoque é atualizado com as informações da nova entrada de materiais.

**Fluxos Alternativos:**

2ª - Se os níveis de estoque estiverem acima do limite máximo, o encarregado de produção pode tomar medidas para reduzir o estoque, como ajustar a produção ou buscar alternativas para utilização dos materiais existentes.

3ª - Se os materiais estiverem abaixo do nível de segurança, mas não for possível iniciar o processo de reposição imediatamente, o encarregado de produção pode tomar medidas temporárias, como ajustar a produção com base nos materiais disponíveis.

8ª - Se os materiais entregues não estiverem de acordo com o pedido feito, o encarregado do almoxarifado e estocagem notifica o departamento de compras para tomar as medidas necessárias.

## Objetos/Classes

### Modelo Conceitual/Classes de Análise/Modelo de Domínio (Classes, Associações, nomes das associações, Multiplicidades e Atributos)

class FábricaDeConcreto {

  - planta: PlantaIndustrial

  - vendas: DepartamentoVendas

  - escritório: Escritório

}

class PlantaIndustrial {

  - receptáculos: List<Receptáculo>

  - misturador: Misturador

  - balanças: List<Balança>

  - válvulaMedição: VálvulaMedição

  - controleDescargaÁgua: ControleDescargaÁgua

  - esteirasTransporte: List<EsteiraTransporte>

}

class Receptáculo {

  - tipo: String

}

class Misturador {

  - preparaçãoConcreto: String

}

class Balança {

  - pesarComponentes: String

}

class VálvulaMedição {

  - medirComponentes: String

}

class ControleDescargaÁgua {

  - controleDescarga: String

}

class EsteiraTransporte {

  - levarIngredientes: String

}

class DepartamentoVendas {

  - ordensPedido: List<OrdemPedido>

  - encaminharIndústria: EncarregadoProdução

}

class OrdemPedido {

  - tipo: String

  - quantidade: int

  - diasEntrega: int

  - transportadora: Transportadora

}

class Transportadora {

  - nome: String

  - entregaConcreto: String

}

class EncarregadoProdução {

  - executarPedido: String

  - áreaProdução: ÁreaProdução

}

class ÁreaProdução {

  - solicitarMateriais: EncarregadoAlmoxarifado

  - controlesFabricação: OperadorMáquinas

}

class EncarregadoAlmoxarifado {

  - entregarMateriais: String

}

class OperadorMáquinas {

  - controlesFabricação: String

  - operários: List<Operário>

  - funcionárioControleFluxo: FuncionárioControleFluxo

}

class Operário {

  - misturarComponentes: String

}

class FuncionárioControleFluxo {

  - auxiliarControles: String

}

class DepartamentoCompras {

  - emitirPedido: String

}

class SetorRecebimentoDistribuição {

  - verificarPedido: String

}

class Construção {

  - receberProduto: String

}

class Cliente {

  - preencherOrdemPedido: Representante

  - realizarPagamento: DepartamentoFinanceiro

}

class Representante {

  - entregarOrdemPedido: String

}

class DepartamentoFinanceiro {

  - receberPagamento: String

}

class Pedido {

  - entregaProduto: Construção

}

FábricaDeConcreto "1" --> "1" PlantaIndustrial : possui

FábricaDeConcreto "1" --> "1" DepartamentoVendas : possui

PlantaIndustrial "1" --> "\*" Receptáculo : possui

PlantaIndustrial "1" --> "1" Misturador : possui

PlantaIndustrial "1" --> "\*" Balança : possui

PlantaIndustrial "1" --> "1" VálvulaMedição : possui

PlantaIndustrial "1" --> "1" ControleDescargaÁgua : possui

PlantaIndustrial "1" --> "\*" EsteiraTransporte : possui

DepartamentoVendas "1" --> "\*" OrdemPedido : recebe

OrdemPedido "\*" <-- "1" Transportadora : possui

EncarregadoProdução "1" --> "1" ÁreaProdução : gerencia

ÁreaProdução "1" <-- "1" EncarregadoAlmoxarifado  : atende

ÁreaProdução "1" <-- "1" OperadorMáquinas : atende

OperadorMáquinas "1" --> "\*" Operário

OperadorMáquinas "1" --> "1" FuncionárioControleFluxo

OperadorMáquinas --> DepartamentoCompras

DepartamentoCompras --> SetorRecebimentoDistribuição

SetorRecebimentoDistribuição --> Construção

Cliente "\*" --> "\*" Pedido

Cliente --> Representante

Cliente --> DepartamentoFinanceiro

Representante --> OrdemPedido

Pedido "1" --> "1" Construção

Construção "1" --> "\*" Pedido

@enduml

Diagrama

Descrição gerada automaticamente

### DSS – Diagramas de Sequência do Sistema, Contratos

@startuml

actor Cliente

actor Representante

actor DepartamentoVendas

actor EncarregadoProdução

actor ÁreaProdução

actor EncarregadoAlmoxarifado

actor OperadorMáquinas

actor Operário

actor FuncionárioControleFluxo

actor DepartamentoCompras

actor SetorRecebimentoDistribuição

actor Construção

actor Transportadora

actor DepartamentoFinanceiro

Cliente -> Representante: preencherOrdemPedido()

Representante -> DepartamentoVendas: entregarOrdemPedido()

DepartamentoVendas -> EncarregadoProdução: encaminharIndústria()

EncarregadoProdução -> ÁreaProdução: executarPedido()

ÁreaProdução -> EncarregadoAlmoxarifado: solicitarMateriais()

EncarregadoAlmoxarifado -> OperadorMáquinas: entregarMateriais()

OperadorMáquinas -> Operário: misturarComponentes()

OperadorMáquinas -> FuncionárioControleFluxo: auxiliarControles()

OperadorMáquinas -> DepartamentoCompras: emitirPedido()

DepartamentoCompras -> SetorRecebimentoDistribuição: verificarPedido()

SetorRecebimentoDistribuição -> Construção: receberProduto()

Transportadora -> Construção: entregarPedido()

OperadorMáquinas -> OperadorMáquinas: controlesFabricação()

OperadorMáquinas -> DepartamentoVendas: controlesFabricação()

DepartamentoVendas -> Cliente: acompanharStatusPedido()

Cliente -> DepartamentoFinanceiro: realizarPagamento()

DepartamentoFinanceiro -> Cliente: receberPagamento()

@enduml

Diagrama

Descrição gerada automaticamente com confiança baixa

### Classes de Implementação - Diagrama de Classes (Classes, Associações, nomes das associações, Multiplicidades, Atributos e Métodos)

class FabricaDeConcreto {

+ vendas: DepartamentoVendas

+ escritorio: Escritorio

+ plantaIndustrial: PlantaIndustrial

}

class DepartamentoVendas {

+ ordensDePedido: List<OrdemDePedido>

+ escalarOrdemDePedido(ordemDePedido: OrdemDePedido): void

}

class Escritorio {

+ encaminharParaProducao(ordemDePedido: OrdemDePedido): void

}

class PlantaIndustrial {

- receptaculosDeArmazenagem: List<ReceptaculoArmazenagem>

- misturador: Misturador

- balancas: List<Balanca>

- valvulaMedicao: ValvulaMedicao

- controleDescarga: ControleDescarga

- esteirasTransporte: List<EsteiraTransporte>

+ executarPedido(ordemDePedido: OrdemDePedido): void

+ solicitarMateriais(): void

}

class ReceptaculoArmazenagem {

- tipo: TipoMaterial

- capacidade: int

}

class Misturador {

- capacidade: int

+ misturar(): void

}

class Balanca {

- capacidade: int

+ pesarComponentes(): void

}

class ValvulaMedicao {

+ medir(): void

}

class ControleDescarga {

+ controlar(): void

}

class EsteiraTransporte {

- capacidade: int

+ transportar(): void

}

class OrdemDePedido {

+ cliente: Cliente

+ tipoConcreto: TipoConcreto

+ quantidade: int

+ dataEntrega: Date

}

class Cliente {

- nome: String

- endereco: String

- contato: String

}

class TipoConcreto {

- nome: String

}

class EncarregadoProducao {

+ executarPedido(ordemDePedido: OrdemDePedido): void

+ solicitarMateriaisFaltantes(): void

}

class EncarregadoAlmoxarifado {

+ entregarMateriais(): void

}

class OperadorMaquinas {

+ controlarFabricacao(): void

}

class Operario {

+ misturarComponentes(): void

}

class FuncionarioControleFluxo {

+ controlarFluxoTrabalho(): void

}

class DepartamentoCompras {

+ providenciarPedido(mercadoria: Mercadoria): void

}

class SetorRecebimentoDistribuicao {

+ verificarPedido(mercadoria: Mercadoria): void

}

class Mercadoria {

- nome: String

- quantidade: int

}

class DepartamentoFinanceiro {

+ receberPagamento(valor: float): void

}

FabricaDeConcreto "1" --> "1" DepartamentoVendas

FabricaDeConcreto "1" --> "1" Escritorio

FabricaDeConcreto "1" --> "1" PlantaIndustrial

PlantaIndustrial "1" --> "\*" ReceptaculoArmazenagem

PlantaIndustrial "1" --> "1" Misturador

PlantaIndustrial "1" --> "\*" Balanca

PlantaIndustrial "1" --> "1" ValvulaMedicao

PlantaIndustrial "1" --> "1" ControleDescarga

PlantaIndustrial "1" --> "\*" EsteiraTransporte

DepartamentoVendas "1" --> "\*" OrdemDePedido

OrdemDePedido "1" --> "1" Cliente

OrdemDePedido "1" --> "1" TipoConcreto

PlantaIndustrial "1" --> "1" EncarregadoProducao

EncarregadoProducao "1" --> "1" EncarregadoAlmoxarifado

EncarregadoProducao "1" --> "1" OperadorMaquinas

OperadorMaquinas "1" --> "\*" Operario

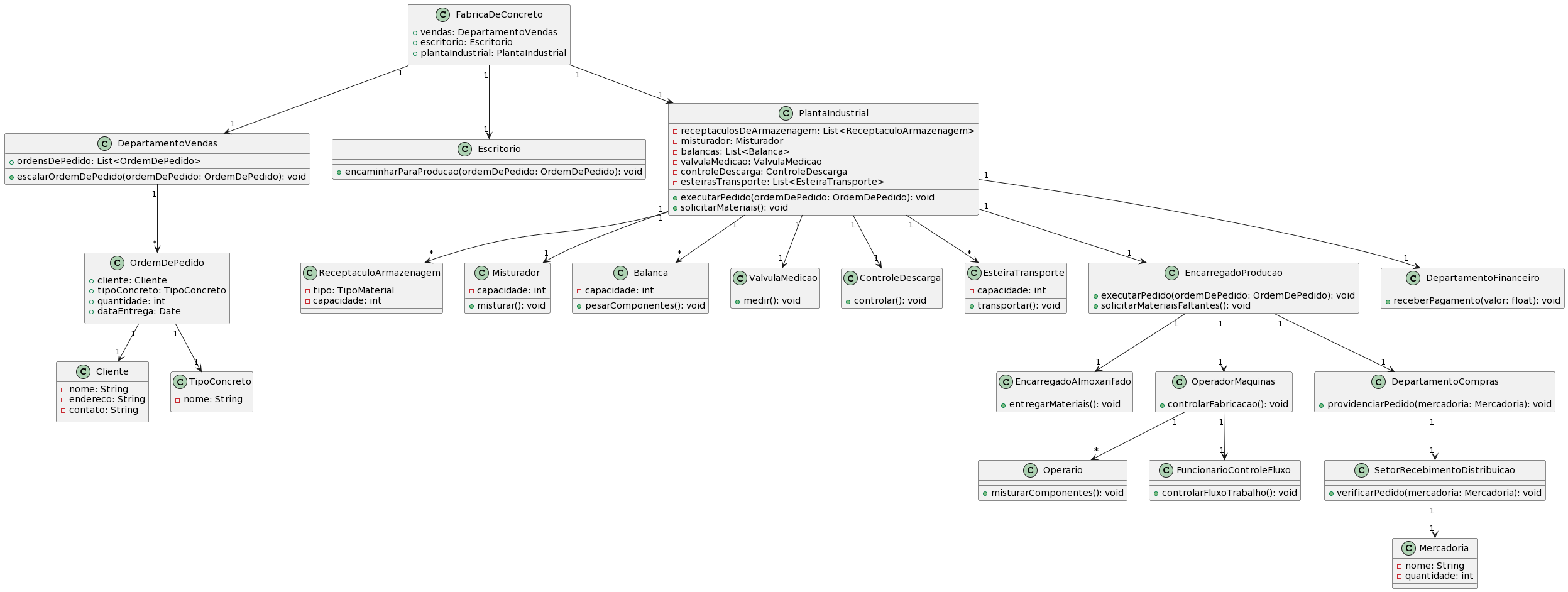
OperadorMaquinas "1" --> "1" FuncionarioControleFluxo

EncarregadoProducao "1" --> "1" DepartamentoCompras

DepartamentoCompras "1" --> "1" SetorRecebimentoDistribuicao

SetorRecebimentoDistribuicao "1" --> "1" Mercadoria

PlantaIndustrial "1" --> "1" DepartamentoFinanceiro



### Diagrama de atividades.

@startuml

title Solicitar Materiais ao Almoxarifado

(\*) --> "Verificar Níveis de Estoque"

if "Níveis de Estoque Baixos?" then

  -->[sim] "Emitir Pedido de Compra ao Departamento de Compras"

  if "Recebimento do Pedido" then

    -->[sim] "Verificar Pedido Recebido pelo Setor de Recebimento e Distribuição"

    --> (\*)

  else

    -->[não] (\*)

  endif

else

  -->[não] (\*)

endif

@enduml

Diagrama

Descrição gerada automaticamente

@startuml

title Entregar Pedido na Obra

(\*) --> "Preparar Pedido para Entrega"

if "Pedido Pronto?" then

  -->[sim] "Carregar Pedido no Veículo de Transporte"

  --> "Transportar Pedido até a Obra"

  --> "Entregar Pedido ao Cliente"

  --> (\*)

else

  -->[não] (\*)

endif

@enduml

Diagrama

Descrição gerada automaticamente