



CURSO DE GRADUAÇÃO EM SISTEMAS DA INFORMAÇÃO

ANDRÉ UGOLINI, EVERTON BERLEZI, RICHARD OTTO, VITOR GABRIEL CAGLIARI  
LOLI, VITOR JUAN

**MÉTODOS ÁGEIS BEBOSIM**

Chapecó/SC, 2025

## 1 PAPÉIS INTEGRANTES

Product Owner — EVERTON MARLON BERLEZI

Responsabilidades: priorizar backlog, aceitar entregas, comunicar visão do produto e necessidades do cliente/gerente da cervejaria.

Scrum Master — Richard Leonardo Otto,

Responsabilidades: remover impedimentos, facilitar cerimônias (Daily, Sprint Planning, Review, Retrospective), proteger time de interrupções.

Desenvolvedor(es) (Frontend / Backend) — VITOR GABRIEL CAGLIARI LOLI – VITOR JUAN – ANDRÉ UGOLINI

Responsabilidades: implementar histórias, escrever testes, integrar com banco.

## 2 CONTEXTO

A Cervejaria BeboSim contratou a Avalon – Software do Futuro para desenvolver um sistema de controle de produção e vendas. O sistema deve contemplar o cadastro de produtos (com informações comerciais e de produção), unidades de produção, embalagens utilizadas, equipes de vendas e seus integrantes, incluindo histórico de gerência e movimentação entre equipes, além do cadastro de clientes pessoas jurídicas. Também deve registrar pedidos de venda emitidos pelos vendedores e controlar campanhas publicitárias, com detalhes sobre produtos participantes, preços promocionais, valores e percentuais de retorno esperados.

## 3 BACKLOG

O backlog do produto é uma lista dinâmica e priorizada de tudo o que é necessário para melhorar um produto, como funcionalidades, requisitos, melhorias e correções. Ele é o único local que define o trabalho a ser realizado e é gerido pelo Product Owner, que o organiza em ordem de valor, para que a equipe de desenvolvimento trabalhe nos itens mais importantes primeiro.

### 3.1 ITENS DO BACKLOG

Para este primeiro sprint, foi definido um conjunto de cinco entregas que representam os primeiros passos essenciais para a construção do sistema. Esses itens foram selecionados com base em sua relevância e impacto imediato no andamento do projeto, garantindo que as funcionalidades iniciais sustentem o desenvolvimento das próximas etapas.

A priorização foi realizada de forma estratégica, organizando as tarefas do backlog em uma ordem que vai das mais urgentes e indispensáveis para o funcionamento básico da aplicação, até aquelas que, embora importantes, podem ser tratadas em um segundo momento dentro da sprint. Dessa forma, asseguramos que os esforços da equipe estejam direcionados para o que é realmente crítico, reduzindo riscos e acelerando a entrega de valor.

O foco principal está em estabelecer uma base sólida, tanto em termos de estrutura quanto de usabilidade, criando um alicerce que permita a evolução contínua do sistema ao longo das próximas iterações. Assim, cada entrega prevista contribui diretamente para aproximar o projeto de seus objetivos gerais, garantindo não apenas progresso técnico, mas também alinhamento com as necessidades identificadas.

### 3.1.1 CADASTRO DE PRODUTOS

A cervejaria BEBOSIM trabalha com a confecção de bebidas, sendo alguns deles cerveja branca, cerveja escura, guaraná normal, guaraná light, água mineral com gás, água mineral sem gás, etc. Visando isso o sistema deverá possibilitar o cadastro dos produtos líquidos produzidos pela mesma através de um CRUD, que terá as funcionalidades de:

- Criar/Adicionar novo produto;
- Listar os produtos cadastrados no sistema;
- Atualizar/Editar registros já inseridos no sistema;
- Excluir registros.

Para esse cadastro será necessário realizar o armazenando dos dados principais, sendo todos eles obrigatórios para poder realizar o cadastro, os dados são:

- Nome do produto - (Campo string 255 caracteres).
- Quantidade em estoque - (Campo integer, não deve possibilitar a inserção de caracteres).
- Preço normal de venda - (Campo double, não deve possibilitar a inserção de caracteres).
- Percentual de comissão sobre a venda (Campo integer, não deve possibilitar a inserção de caracteres).
- Fórmula de produção - (Campo string 255 caracteres).

**Regras de negócio da tela:** Ao selecionar no menu a tela de cadastro de produtos, deve ser automaticamente exibido os produtos já cadastrados em uma tabela, cada linha da tabela será um produto com suas respectivas informações listadas acima, e ao final de cada produto (Em uma coluna separada) deve conter dois ícones, o primeiro o de editar produto, e o segundo sendo o excluir produto.

Abaixo da tabela deve conter o botão de adicionar novo produto (Adicionar). Ao ser selecionado deve carregar um pop-up com os campos disponíveis também listados acima, todos os campos devem ser obrigatórios, abaixo dos campos terá o botão de cadastrar, que só será possível selecioná-lo após preencher todos os campos. Após o preenchimento e a confirmação, os dados são salvos no banco de dados.

O botão de editar (localizado ao lado de cada item), será responsável por atualizar os dados no banco de dados, ao selecioná-lo abrirá o mesmo pop-up de cadastro, mas com os campos já preenchidos com as respectivas informações do produto selecionado. Ao confirmar as novas informações serão salvas no banco de dados.

O botão de excluir (também localizado ao lado de cada item), será responsável por remover o produto do banco de dados, ao selecioná-lo exibirá um alerta na tela com a mensagem “Confirmar exclusão do item [NOME DO ÍTEM]”. Onde terá dois botões, o de cancelar, que fechará o pop-up e não removerá o registro, e o de confirmar, que ao ser selecionado removerá o respectivo item do banco de dados.

### 3.1.2 CADASTRO DE EMBALAGENS

A partir do cadastro de produtos o sistema deverá possibilitar o cadastro das embalagens que serão associado(s) com seu(s) respectivo(s) produto(s) já cadastrado(s) anteriormente, dentre as embalagens estão a lata 300ml, garrafa 600ml, garrafa pet 2 litros etc. A tela será também um CRUD, que terá as funcionalidades de:

- Criar/Adicionar nova embalagem;
- Listar as embalagens cadastradas no sistema;
- Atualizar/Editar registros já inseridos no sistema;
- Excluir registros.

Para esse cadastro será necessário realizar o armazenando dos dados principais, sendo todos eles obrigatórios para poder realizar o cadastro, os dados são:

- Nome da embalagem - (Campo string 255 caracteres).
- Material da embalagem (plástico, alumínio etc) - (Campo string 255 caracteres).
- Custo da embalagem - (Campo double, não deve possibilitar a inserção de caracteres).
- Volume e unidade (ml, litros etc) - (Campo double, não deve possibilitar a inserção de caracteres) .
- Associar embalagem a múltiplos produtos e produtos a múltiplas embalagens - (Campo de múltipla escolha, onde lista os produtos cadastrados no sistema).

**Regras de negócios da tela:** Ao selecionar no menu a tela de cadastro de embalagens, deve ser automaticamente exibido as embalagens já cadastrados em uma tabela, cada linha da tabela será uma embalagem com suas respectivas informações listadas acima, e ao final de cada embalagem (Em uma coluna separada) deve conter dois ícones, o primeiro o de editar embalagem, e o segundo sendo o excluir embalagem.

Abaixo da tabela deve conter o botão de adicionar nova embalagem (Adicionar). Ao ser selecionado deve carregar um pop-up com os campos disponíveis também listados acima, todos os campos devem ser obrigatórios, abaixo dos campos terá o botão de cadastrar, que só será possível selecioná-lo após preencher todos os campos. Após o preenchimento e a confirmação, os dados são salvos no banco de dados.

O botão de editar (localizado ao lado de cada item), será responsável por atualizar os dados no banco de dados, ao selecioná-lo abrirá o mesmo pop-up de cadastro, mas com os campos já preenchidos com as respectivas informações do produto selecionado. Ao confirmar as novas informações serão salvas no banco de dados.

O botão de excluir (também localizado ao lado de cada item), será responsável por remover a embalagem do banco de dados, ao selecioná-lo exibirá um alerta na tela com a mensagem "Confirmar exclusão da embalagem [NOME DA EMBALAGEM]". Onde terá dois botões, o de cancelar, que fechará o pop-up e não removerá o registro, e o de confirmar, que ao ser selecionado removerá o respectivo item do banco de dados.

### 3.1.3 CADASTRO DE FILIAIS

O sistema deverá possibilitar o cadastro das filiais (unidades de produção), permitindo associar cada produto produzido à sua respectiva unidade. A tela será desenvolvida como um CRUD completo, disponibilizando as funcionalidades de criar, listar, editar e excluir registros.

Os dados obrigatórios para o cadastro são:

- Nome da filial – (Campo string, até 255 caracteres).
- Endereço completo – (Campo string, até 255 caracteres).
- CNPJ – (Campo string, com máscara de formatação para CNPJ).
- Área construída – (Campo double, para registro da área em m<sup>2</sup>).
- Telefone de contato – (Campo string, com máscara de formatação para telefone).

#### **Regras de negócio da tela:**

Ao acessar o menu de cadastro de filiais, o sistema exibirá uma tabela contendo as unidades já cadastradas, com todos os campos listados acima. Cada linha da tabela apresentará ao final dois ícones: um para editar filial e outro para excluir filial.

Na parte inferior da tela, deverá existir o botão Adicionar nova filial. Ao acioná-lo, será exibido um pop-up com os campos de cadastro, todos de preenchimento obrigatório. Após o preenchimento completo, o botão Cadastrar ficará habilitado, permitindo a gravação das informações no banco de dados.

O botão de edição permitirá a atualização dos dados de uma filial existente, carregando o pop-up preenchido com as informações salvas. O botão de exclusão exibirá um alerta com a mensagem: “Confirmar exclusão da filial [NOME DA FILIAL]”. O usuário poderá confirmar ou cancelar a operação, e somente em caso de confirmação os dados serão removidos definitivamente.

### 3.1.4 CADASTRO DE CLIENTES

O sistema permitirá o cadastro de clientes, restringindo-se exclusivamente a pessoas jurídicas. O módulo também seguirá a estrutura CRUD, possibilitando a inclusão, listagem, edição e exclusão de registros.

Os dados obrigatórios a serem cadastrados são:

- Razão social – (Campo string, até 255 caracteres).
- CNPJ – (Campo string, com máscara de formatação para CNPJ).
- Endereço completo – (Campo string, até 255 caracteres).
- Telefone – (Campo string, com máscara de formatação).
- Pessoa de contato – (Campo string, até 255 caracteres).

#### **Regras de negócio da tela:**

Ao selecionar o menu de cadastro de clientes, será exibida uma tabela contendo os registros existentes, organizados pelas informações principais. Ao final de cada linha, haverá os botões para editar cliente e excluir cliente. Na parte inferior, o botão Adicionar novo cliente abrirá um pop-up contendo os campos obrigatórios. Apenas quando todos estiverem preenchidos, o botão Cadastrar ficará ativo. Após a confirmação, as informações serão armazenadas no banco de dados.

O botão de edição carregará o pop-up com os dados já salvos, permitindo atualizações. Já o botão de exclusão exibirá a mensagem: “Confirmar exclusão do cliente [RAZÃO SOCIAL]”. O usuário poderá optar por confirmar ou cancelar a operação, garantindo maior segurança contra exclusões acidentais

### 3.1.5 CONTROLE DE VENDAS

O sistema permitirá o cadastro de clientes, restringindo-se exclusivamente a pessoas jurídicas. O módulo também seguirá a estrutura CRUD, possibilitando a inclusão, listagem, edição e exclusão de registros.

Os dados obrigatórios a serem cadastrados são:

- Razão social – (Campo string, até 255 caracteres).
- CNPJ – (Campo string, com máscara de formatação para CNPJ).
- Endereço completo – (Campo string, até 255 caracteres).
- Telefone – (Campo string, com máscara de formatação).
- Pessoa de contato – (Campo string, até 255 caracteres).

#### **Regras de negócio da tela:**

Ao selecionar o menu de cadastro de clientes, será exibida uma tabela contendo os registros existentes, organizados pelas informações principais. Ao final de cada linha, haverá os botões para editar cliente e excluir cliente. Na parte inferior, o botão Adicionar novo cliente abrirá um pop-up contendo os campos obrigatórios. Apenas quando todos estiverem preenchidos, o botão Cadastrar ficará ativo. Após a confirmação, as informações serão armazenadas no banco de dados. O botão de edição carregará o pop-up com os dados já salvos, permitindo atualizações. Já o botão de exclusão exibirá a mensagem: “Confirmar exclusão do cliente [RAZÃO SOCIAL]”. O usuário poderá optar por confirmar ou cancelar a operação, garantindo maior segurança contra exclusões acidentais



## 4 DEFINIÇÕES SPRINT

Um Sprint é um ciclo de trabalho definido dentro da metodologia ágil Scrum, com duração fixa que geralmente varia entre uma e quatro semanas. Durante esse período, a equipe de desenvolvimento se compromete a atingir um objetivo claro, chamado de *Sprint Goal*, que consiste em entregar um incremento do produto que seja funcional e gere valor para o cliente ou usuário final. Ao longo do Sprint, as atividades planejadas são executadas de forma colaborativa e acompanhadas diariamente em reuniões curtas, garantindo visibilidade do progresso e identificação de possíveis impedimentos. No encerramento, realiza-se uma revisão para apresentar os resultados alcançados e coletar feedback dos stakeholders, além de uma retrospectiva interna, em que o time reflete sobre o processo e define melhorias para o próximo ciclo. Dessa forma, o Sprint representa a base do Scrum, promovendo entregas frequentes, adaptação contínua e maior alinhamento com as necessidades do projeto.

### 4.1 DURAÇÃO DA SPRINT

Este primeiro Sprint terá duração de duas semanas, tempo considerado adequado para a entrega de um mínimo produto viável do domínio, contemplando os CRUDs com suas funcionalidades básicas, o controle de venda que engloba a parte de contato com os clientes, e a confecção do banco de dados para armazenar todas as informações dos cadastros e controles. Esse período é suficiente para permitir a modelagem inicial do sistema, desenvolvimento dos CRUDs, e construção do banco de dados.

Optar por um Sprint curto garante uma entrega rápida de valor ao cliente, possibilitando testes iniciais e coleta de feedback precoce, além de reduzir riscos. Dessa forma, ao final do ciclo, será possível realizar uma Review com o cliente e, a partir do retorno, ajustar o backlog de forma alinhada às necessidades reais do projeto. Caso a equipe seja muito reduzida ou disponha de poucas horas semanais, o prazo poderá ser estendido para três semanas, mantendo a lógica de entrega ágil e iterativa.

### 4.2 APLICAÇÃO DOS MÉTODOS ÁGEIS

#### 4.2.1 PLANEJAMENTO DA SPRINT

O objetivo da sprint é decomposto em histórias menores (tickets), cada uma com estimativas claras e critérios de aceitação bem definidos. A equipe adota uma Definition of Done abrangente, que inclui código funcional, testes implementados, documentação mínima e build sem erros.

#### 4.2.2 DAILY SCRUM

São realizadas reuniões diárias de 15 minutos, onde cada membro compartilha o que foi feito no dia anterior, o que pretende realizar a seguir e eventuais impedimentos. O Scrum Master atua ativamente na remoção de bloqueios, garantindo fluidez ao progresso.

#### 4.2.4 ENTREGA INCREMENTAL E INTEGRAÇÃO CONTÍNUA

O desenvolvimento é conduzido por meio de commits frequentes, integrados a um pipeline de CI que executa builds e testes automatizados. Isso assegura a detecção precoce de falhas e reduz riscos de retrabalho.

#### 4.2.5 REVISÕES FREQUENTES COM O PO/CLIENTE

Ao final do Sprint, é apresentada uma demo aos stakeholders (cliente, professor ou gerente da cervejaria), demonstrando as funcionalidades concluídas. Esse momento permite coletar feedback direto e ajustar prioridades de acordo com as expectativas reais.

#### 4.2.6 FEEDBACK RÁPIDO E ENTREGAS UTILIZÁVEIS

O backlog é priorizado de forma a favorecer entregas pequenas e já utilizáveis, como por exemplo uma página de listagem de produtos, permitindo validar requisitos essenciais e identificar necessidades de ajuste já no início.

#### 4.2.7 ADAPTAÇÃO A MUDANÇAS

O backlog é continuamente reordenado a partir das revisões com o cliente, aceitando mudanças no escopo sempre que negociadas com o PO, preservando a flexibilidade e o foco no valor entregue.

#### 4.2.8 DEFINITION OF READY / DONE

As histórias só são iniciadas quando possuem critérios de aceitação claros (Definition of Ready). Considera-se concluída (Done) apenas a funcionalidade que tiver passado pelos testes, code review e deploy em ambiente de staging.

#### 4.2.9 PAIR PROGRAMMING E CODE REVIEW

A prática de programação em pares é incentivada, especialmente entre desenvolvedores mais experientes e iniciantes, promovendo aprendizado, redução de falhas e aumento da qualidade do código.

#### 4.2.10 RETROSPECTIVA

Ao término do Sprint, a equipe realiza uma reunião de retrospectiva para identificar ao menos duas ações concretas de melhoria, como aprimorar estimativas, reduzir reuniões desnecessárias ou ampliar a cobertura de testes, aplicando esses aprendizados no Sprint seguinte.