

**ROTEIRO PARA ORIENTAR O DESENVOLVIMENTO DO  
TRABALHO DE LABORATÓRIO DE ENGENHARIA DE  
SOFTWARE I**

**DECOM – CEFET/MG**

**Professor:**

*Ismael Santana Silva*

# 1. Escopo do Trabalho

O objetivo do Laboratório de Engenharia de Software I consiste em colocar em prática a execução de um processo de desenvolvimento para construção de um software. Neste semestre os alunos deverão construir o sistema Web, utilizando uma metodologia que consiste na adaptação do SCRUM e o desenvolvimento será dirigido por Caso de Uso (CDU). O grupo deve propor um sistema útil para os professores do CEFET-MG, para isso você pode pesquisar com os professores sistemas que eles gostariam que fossem desenvolvidos para auxiliar nas atividades acadêmicas. Exemplos de sistemas:

- Sistema para entrega de Trabalho de Conclusão de Curso (TCC): coletar requisitos com a Professora Glória;
- Revisor automático de TCC: coletar requisitos com o Professor Ismael;

A metodologia SCRUM prevê os seguintes passos e artefatos:

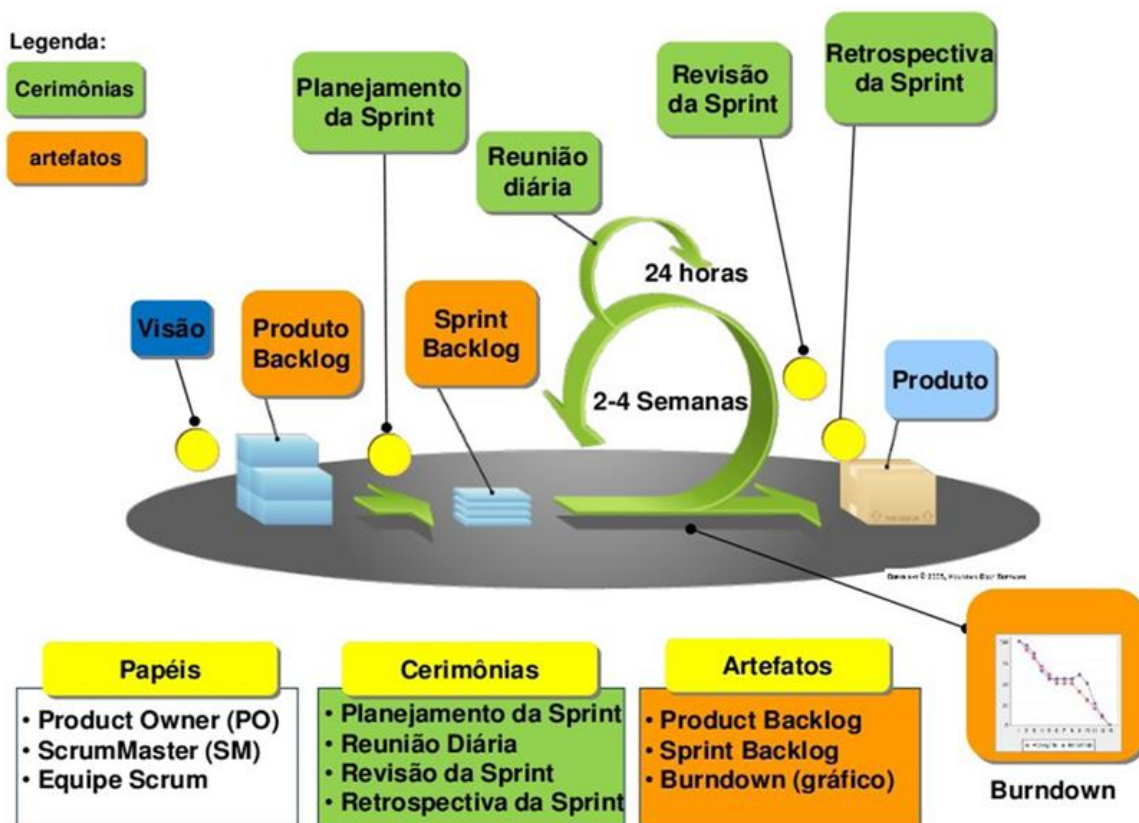


Figura 01. SCRUM - Visão Geral

O trabalho será desenvolvido em grupo formado por no máximo 05 (cinco) pessoas e cada membro deve implementar pelo menos 03 (três) CDUs, ou seja, um grupo com 05 integrantes deve implementar 15 CDUs. Todo o processo deve ser documentado o sistema Trello<sup>1</sup>.

Todo o material da disciplina será disponibilizado no seguinte link do Google Drive: <<https://goo.gl/ZvxeRW>>.

## 1.1. As etapas previstas consistem em 06 entregas:

Data	Entrega	Artefatos Previstos
14/03/2018	Entrega 01 05 pontos	<ul style="list-style-type: none"> <li>- <i>Product Backlog</i> (i.e, lista de funções - CDUs - do sistema), ver exemplo no tópico 2.1.</li> <li>- O documento deve ser adicionado no Trello, exemplo em <a href="https://trello.com/b/U35NxQ7s">https://trello.com/b/U35NxQ7s</a>. O professor deve ser adicionado como membro do projeto no Trello, e-mail ismaelsantana@decom.cefetmg.br.</li> </ul>
04/04/2018	Entrega 02 30 pontos	<ol style="list-style-type: none"> <li>1. Especificação do Software (Ver exemplos no tópico 2.2)               <ol style="list-style-type: none"> <li>a. Diagrama de contexto (Diagrama de Casos de Uso - CDU);</li> <li>b. Diagrama de sequencia do sistema (Fluxo principal do sistema);</li> <li>c. Detalhamento das histórias de usuário (pelo menos 01 Estória por CDU);                   <ol style="list-style-type: none"> <li>i. Cenário;</li> <li>ii. Critérios de Aceitação (i.e., Casos de Testes);</li> </ol> </li> <li>d. Protótipos de tela;</li> <li>e. Classes de domínio do sistema;</li> <li>f. Diagrama de Entidade e Relacionamento;                   <ol style="list-style-type: none"> <li>i. Utilize o MySQL WorkBench para criação do diagrama;</li> </ol> </li> </ol> </li> <li>2. Registro do <i>Product Backlog</i> e planejamento da 1a Sprint (1o <i>Sprint Backlog</i>) no Trello<sup>1</sup>, exemplo de como os artefatos devem ser documentados: <a href="https://trello.com/b/U35NxQ7s">https://trello.com/b/U35NxQ7s</a>. O professor deve ser adicionado como membro do projeto no Trello, e-mail ismaelsantana@decom.cefetmg.br.</li> </ol>
25/04/2018	Entrega 03 15 pontos	<ol style="list-style-type: none"> <li>3. <b>Retrospectiva da 1ª Sprint</b> <ol style="list-style-type: none"> <li>a. Apresentação do Produto da 1a <i>Sprint</i> (Um CDU implementado por aluno, ou seja, se o grupo contém 4 membros, pelo menos 4 CDUs devem ser implementados nesse Sprint)</li> <li>b. Apresentação do <i>Burndown chart</i> para acompanhamento</li> <li>c. Planejamento da 2a Sprint (2o <i>Sprint Backlog</i>) no Trello</li> </ol> </li> </ol>
16/05/2018	Entrega 04 15 pontos	<ol style="list-style-type: none"> <li>4. <b>Retrospectiva da 2ª Sprint</b> <ol style="list-style-type: none"> <li>a. Apresentação do Produto da 2a <i>Sprint</i> (Um CDU implementado por aluno, ou seja, se o grupo contém 4 membros, pelo menos 4 CDUs devem ser implementados nesse Sprint)</li> <li>b. Apresentação do <i>Burndown chart</i> para acompanhamento</li> <li>c. Planejamento da 3a Sprint (3o <i>Sprint Backlog</i>) no Trello</li> </ol> </li> </ol>
06/06/2018	Entrega 05 15 pontos	<ol style="list-style-type: none"> <li>5. <b>Retrospectiva da 3ª Sprint</b> <ol style="list-style-type: none"> <li>a. Desenvolver os casos de uso utilizando a metodologia <i>Test-Driven Development</i> (TDD);</li> <li>b. Apresentação do Produto da 3a <i>Sprint</i> (Um CDU implementado por aluno, ou seja, se o grupo contém 4 membros, pelo menos 4 CDUs devem ser implementados nesse Sprint)</li> <li>c. Apresentação do <i>Burndown chart</i> para acompanhamento</li> <li>d. Apresentação dos resultados dos testes unitários do <i>Sprint</i></li> </ol> </li> </ol>
27/06/2018	Entrega 06 10 pontos	<ol style="list-style-type: none"> <li>1. Manual do usuário</li> <li>2. Manual de configuração do ambiente de execução e implantação do sistema</li> </ol>
-	Práticas 10 pontos	Durante todo o semestre serão realizados exercícios práticos para introdução das tecnologias utilizadas. Caso não seja necessário, esses pontos serão distribuídos entre as <i>sprints</i> de desenvolvimento.

<sup>1</sup> Trello - <https://trello.com/>

## 2. Exemplos de alguns Artefatos para cada Entrega

### 2.1. Entrega 01 - Escopo do Sistema

Exemplo de *Product Backlog* de um Sistema de Controle de Vendas

Nome do Produto		Merci
Missão do Produto		O produto Merci 1.0 visa a oferecer apoio informatizado ao controle de vendas e de compras de pequenas mercearias, implementado com tecnologia atualizada na época da contratação.
Product Backlog (Lista de Funções)		
Código	Nome da Função	Descrição
1	Gestão de mercadorias	Fornecimento de dados a outras funções. Identificação das mercadorias.
2	Controle da operação de venda	Registro de produtos e dos valores vendidos. Viabilização do controle de estoque. Emissão de tíquetes de caixa para o cliente da mercearia.
....	...	....
N	Gestão de fornecedores	Atualização dos dados de cadastro de fornecedores. Atualização da lista de produtos comercializados por cada fornecedor.

## 2.2. Entrega 02 - Especificação do Sistema

### 2.2.1. Exemplo de Diagrama de Contexto (Diagrama de Casos de Uso) - Sistema de Gestão de Reservas de Recursos

OBS.: Cada CDU deve ter um código e nome (e.g., CDU001. Visualizar Disponibilidade de Recursos)

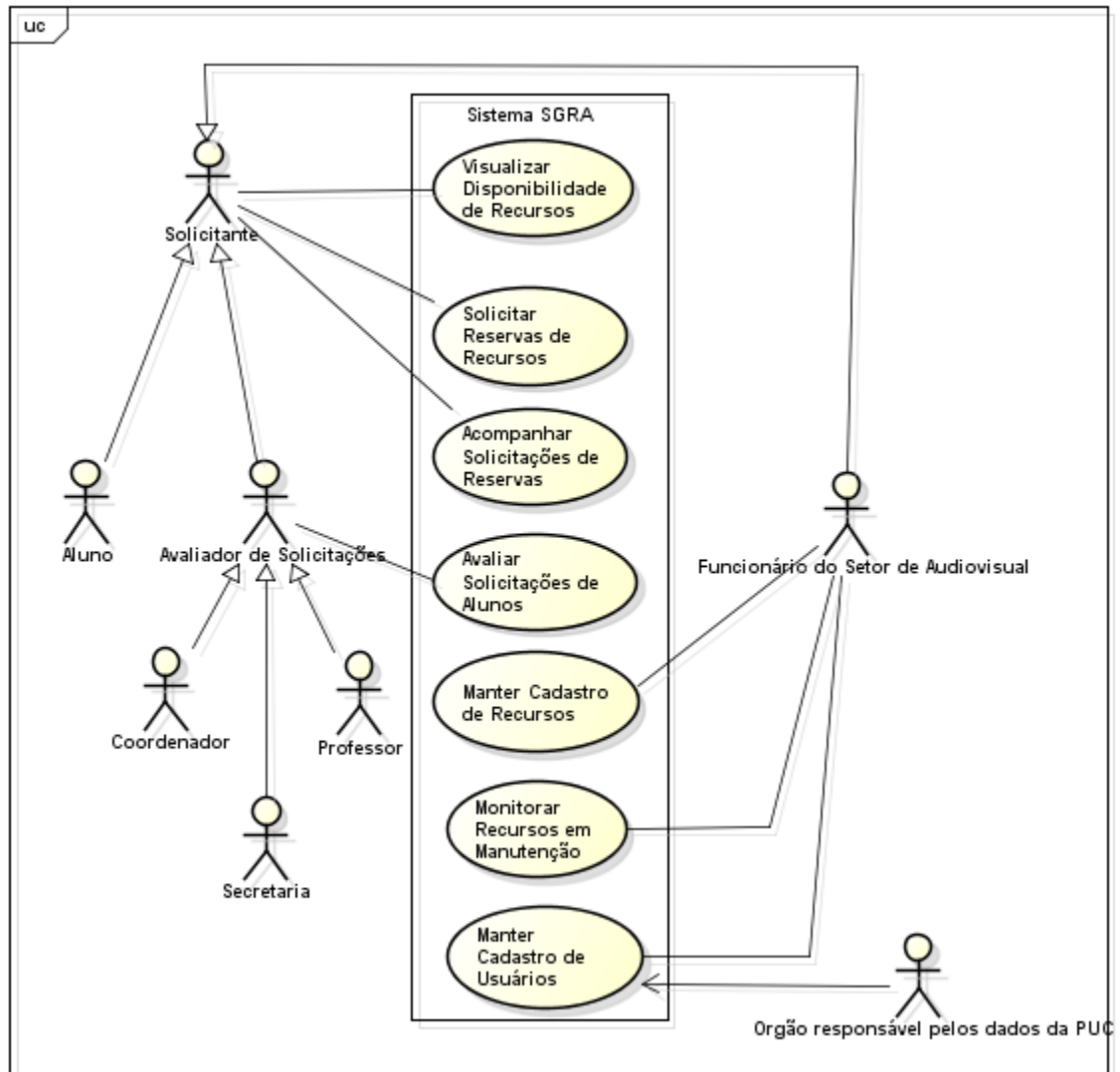


Figura 01. Diagrama de Contexto de um Sistema

### 2.2.2. Exemplo de Estória de Usuário

Nas abordagens ágeis, podemos definir e organizar os requisitos de um sistema utilizando *User Stories* (Estórias de Usuário). *User Stories* fracionam os requisitos para que seja possível (e mais fácil) estimar o esforço. São descrições simples que descrevem uma funcionalidade e é recomendável que sejam escritas segundo o ponto de vista do usuário.

*OBS: Para cada CDU, deve ser apresentada pelo menos 01 estória de usuário*

- Template

Como um [ator] eu quero/preciso de  
/devo/gostaria de [ação] para  
[funcionalidade].

**[Critérios de Aceitação]**

Figura 02. Template de uma *User Story*

Onde:

- **Ator** – O proprietário da *User Story*. De forma simplista é o usuário, o interessado naquela funcionalidade. Mas é recomendado descrever de forma específica quem é o ator para que seja mais fácil identificar o contexto da história dentro do sistema.
  - **Ação** – É o que o ator quer fazer. A partir dessa ação ele espera alcançar seu objetivo dentro do sistema.
  - **Funcionalidade** – É o que o ator espera que aconteça ao executar a ação
  - **Critérios de Aceitação** - Os Critérios de Aceitação são representados por uma lista de itens de negócio que expressam formas de usar a funcionalidade implementada em uma História. O objetivo dessa lista é validar se a História foi implementada de acordo com as solicitações do PO.
- Exemplo:

Como um vendedor responsável pelo setor de livros eu quero procurar por livros filtrando por nome para que seja possível verificar se o livro X está disponível para pronta entrega.

#### Critérios para aceitação:

- O vendedor não pode solicitar a busca se não informar o nome do livro.
- O sistema, encontrando o livro, deve apresentar todos os dados do livro (nome completo, autores, editora, ano de edição) e a quantidade em estoque.
- O sistema, não encontrando o livro, deve informar que o livro não foi encontrado.

Figura 03. Exemplo de uma User Story



### 2.2.3. Exemplo de protótipo de Interface

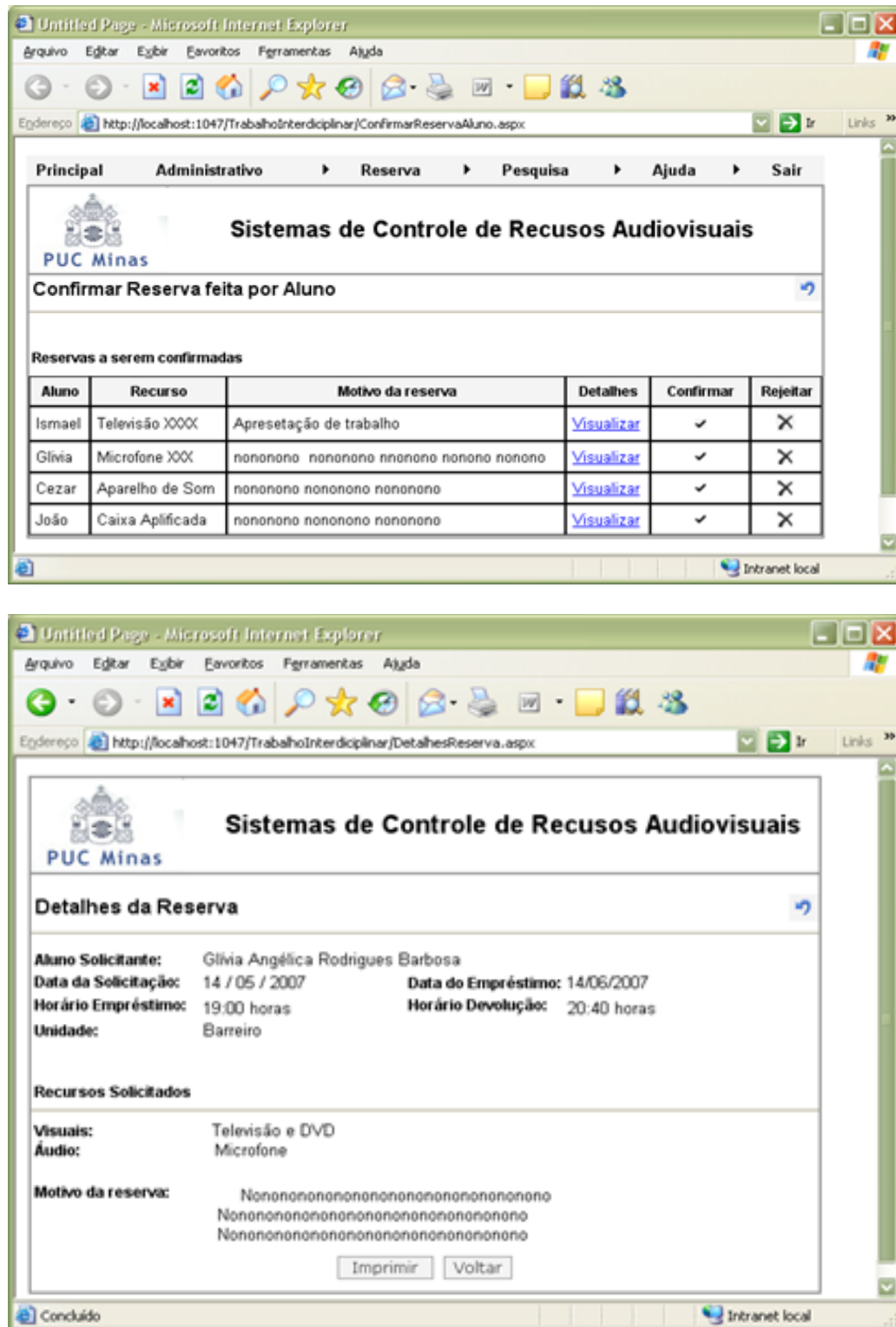


Figura 04. Exemplo de Protótipo



## 2.2.4. Exemplo de Classe do Domínio - Sistema de Gestão de Reservas de Recursos

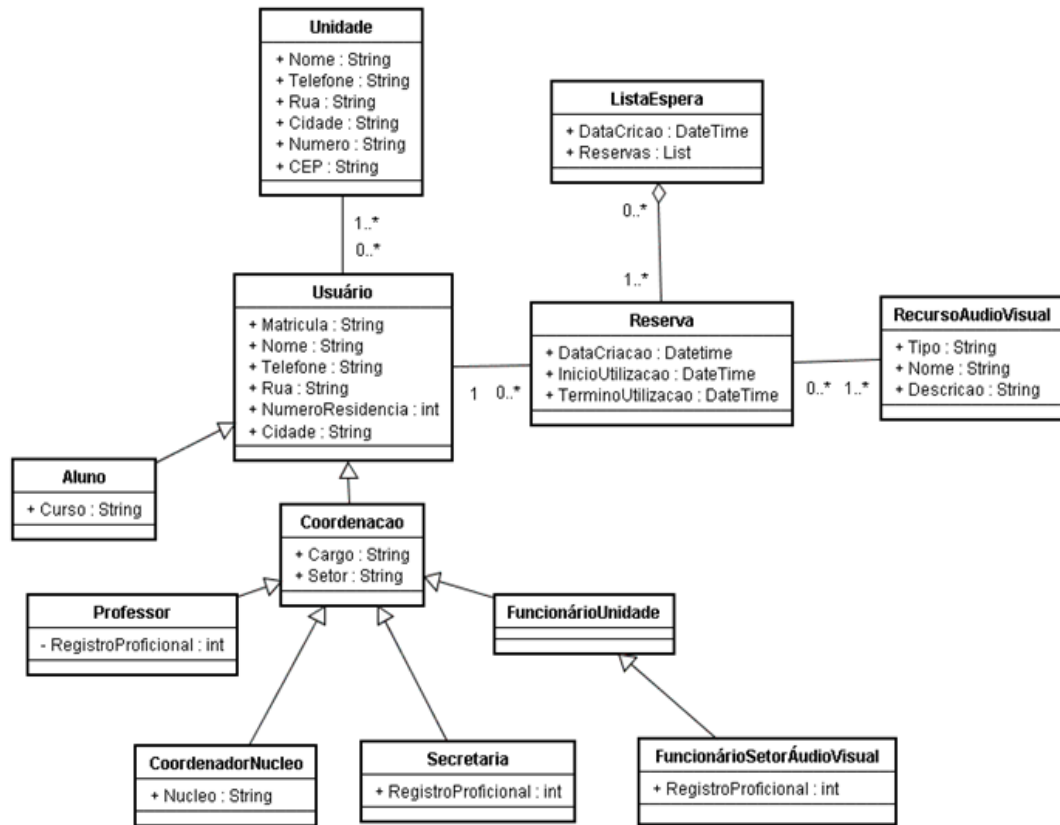
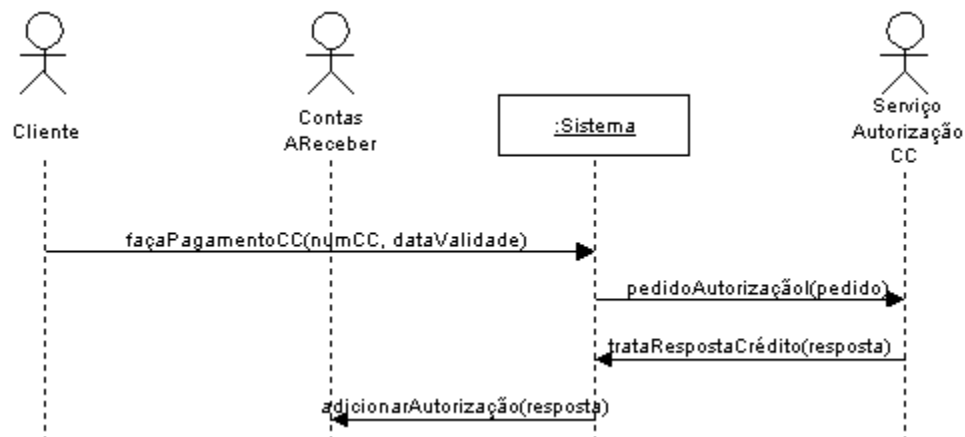


Figura 05. Exemplo de Diagrama de Classes

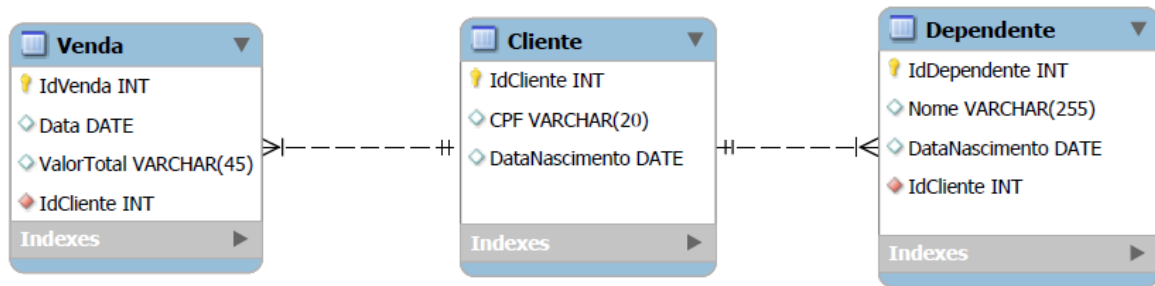
## 2.2.5. Exemplo de Sprint Backlog

Apresentação das tarefas da Sprint a ser executada no Trello com a atribuição dos responsáveis. O nome da tarefa deve apresentar o número da Sprint: **[Sprint Código] Nome da Atividade**. Por exemplo: <https://trello.com/b/U35NxQ7s>.

## 2.2.6. Exemplo de Diagrama de Sequencia do Sistema



### 2.2.7. Exemplo de *Diagrama de Entidade e Relacionamento (DER)*



## 2.3. Entrega 06 - Manuais do Sistema

Exemplos de Manuais do Usuário e de Configuração/Implantação do Sistema disponíveis no Google Drive na pasta *Templates e Exemplos*

# 3. Ferramentas e Tecnologias Exigidas

Para realizar as entregas, o processo de desenvolvimento adotado exige o conhecimento e uso das seguintes ferramentas e/ou tecnologias.

- **Editor de Texto** – Para documentar o *Product Backlog*, bem como detalhar as Estórias de Usuário;
- **Ferramentas Case** (ex: *Astah Community*) – Para modelar os Diagramas de Caso de Uso e Classes de Domínio;
- **Ferramentas de Prototipação** (ex: *Pencil Project*) – Para prototipar interfaces;
- **Trello** – Para o planejamento e gerenciamento das Sprints;
- **SVN ou Git** – Para controle de versionamento dos artefatos gerados durante o processo de desenvolvimento
  - o Gerenciar a documentação do sistema (*Product Backlog*, Diagramas, Estórias de Usuários e Protótipos de Interface);
  - o Gerenciar o código fonte;
- **MVC** – Padrão arquitetural que deverá ser adotado para implementar o sistema Web;
- **Web** – Para desenvolvimento do sistema;
- **Banco de Dados** - Para manutenção do Banco de Dados da aplicação;
- **Framework ORM (Exemplo: Hibernate)** – Framework para mapeamento objeto relacional;

OBS:.

*Caso os alunos não tenham conhecimento prévio sobre as ferramentas e/ou tecnológicas citadas, recomenda-se fortemente que busquem estudá-las, uma vez que esses assuntos não serão abordados em profundidade no laboratório.*

## 4. Orientações Complementares

1. Como informado anteriormente, o desenvolvimento será dirigido por CDUs. Para que haja uma boa rastreabilidade desses requisitos, cada CDU deverá conter um código que permitirá identificá-lo em todas as ferramentas utilizadas no projeto.
2. Templates para Documentação – Os alunos deverão seguir os exemplos listados acima como templates para elaboração dos artefatos previstos nas entregas