

# Documentação do Mínimo Produto Viável (MVP) do Módulo de Certificados para o Sistema Magister

Versão 1.0

# Sumário

<b>Sumário</b>	<b>1</b>
<b>Introdução</b>	<b>2</b>
<b>Problemática</b>	<b>2</b>
<b>Requisitos</b>	<b>2</b>
Requisitos Funcionais	2
Requisitos Não Funcionais	3
Requisitos Postergados	4
<b>Casos de Uso</b>	<b>4</b>
Diagrama de Casos de Uso	4
Descrição dos Casos de Uso	5
<b>Arquitetura do Banco de Dados</b>	<b>8</b>
<b>Tecnologias Utilizadas</b>	<b>9</b>
<b>Referências</b>	<b>10</b>

# Introdução

Este documento apresenta uma visão abrangente do Mínimo Produto Viável (*MVP*) do Módulo de Certificados para o Sistema Magister. Nas seções seguintes estão dispostos: A apresentação da problemática que este *MVP* está disposto a solucionar, os requisitos do sistema, os casos de uso por meio da Linguagem de Modelagem Unificada (*UML*), arquitetura do banco de dados, as tecnologias utilizadas para a construção do *MVP*.

## Problemática

Desenvolvimento de um sistema de validação de horas extracurriculares, onde desde o primeiro período os alunos poderão anexar seus certificados e ter controle de quantas horas foram concluídas até então.

## Requisitos

Requisitos são descrições dos serviços que um sistema deve prestar e as restrições a sua operação (Sommerville; 2018; p.85). Os requisitos podem ser funcionais e não funcionais, as próximas seções irão explicar e descrever cada um deles (Sommerville; 2018; p.88).

### Requisitos Funcionais

Requisitos funcionais são as funcionalidades que um sistema deve fornecer. Neste MVP são (Sommerville; 2018; p.88):

[RF001] Cadastro de certificados					
<b>Descrição:</b>	O sistema deve permitir o cadastro de certificados.				
<b>Prioridade:</b>	<input checked="" type="checkbox"/>	Essencial	<input type="checkbox"/>	Importante	<input type="checkbox"/> Desejável

[RF002] Envio de arquivos					
<b>Descrição:</b>	O sistema deve permitir anexar o certificado no cadastro.				
<b>Prioridade:</b>	<input checked="" type="checkbox"/>	Essencial	<input type="checkbox"/>	Importante	<input type="checkbox"/> Desejável

[RF003] Validação de certificados					
<b>Descrição:</b>	O sistema deve possibilitar a auditoria dos certificados.				
<b>Prioridade:</b>	<input checked="" type="checkbox"/>	Essencial	<input type="checkbox"/>	Importante	<input type="checkbox"/> Desejável

[RF004] Cálculo da carga horária total dos certificados do aluno					
<b>Descrição:</b>	O sistema deve calcular o total de horas dos certificados aprovados.				
<b>Prioridade:</b>	<input checked="" type="checkbox"/>	Essencial	<input type="checkbox"/>	Importante	<input type="checkbox"/> Desejável

[RF005] Listagem de certificados do aluno					
<b>Descrição:</b>	O sistema deve listar todos os certificados do aluno, informando: nome do certificado, tipo de atividade, carga horária e situação do documento (homologado, em análise ou não-homologado).				
<b>Prioridade:</b>	<input checked="" type="checkbox"/>	Essencial	<input type="checkbox"/>	Importante	<input type="checkbox"/> Desejável

[RF006] Login de acesso					
<b>Descrição:</b>	O sistema deve possibilitar que apenas usuários cadastrados o acessem.				
<b>Prioridade:</b>	<input type="checkbox"/>	Essencial	<input checked="" type="checkbox"/>	Importante	<input type="checkbox"/> Desejável

[RF007] Perfis de acesso					
<b>Descrição:</b>	O sistema deve ter dois níveis de acesso. O de gestores e o de alunos.				
<b>Prioridade:</b>	<input type="checkbox"/>	Essencial	<input checked="" type="checkbox"/>	Importante	<input type="checkbox"/> Desejável

[RF008] Listagem de todos os alunos					
<b>Descrição:</b>	O sistema deve listar todos os alunos cadastrados.				
<b>Prioridade:</b>	<input type="checkbox"/>	Essencial	<input checked="" type="checkbox"/>	Importante	<input type="checkbox"/> Desejável

[RF009] Listagem de todos os certificados					
<b>Descrição:</b>	O sistema deve listar todos os certificados cadastrados.				
<b>Prioridade:</b>	<input type="checkbox"/>	Essencial	<input checked="" type="checkbox"/>	Importante	<input type="checkbox"/> Desejável

[RF010] Visualização de certificados					
<b>Descrição:</b>	O sistema deve possibilitar a visualização dos dados do certificado.				
<b>Prioridade:</b>	<input type="checkbox"/>	Essencial	<input checked="" type="checkbox"/>	Importante	<input type="checkbox"/> Desejável

[RF011] Edição de certificados					
<b>Descrição:</b>	O sistema deve possibilitar a edição dos dados do certificado.				
<b>Prioridade:</b>	<input type="checkbox"/>	Essencial	<input checked="" type="checkbox"/>	Importante	<input type="checkbox"/> Desejável

[RF012] Gravação de Logs					
<b>Descrição:</b>	O sistema deve registrar logs de acesso de todas as ações dos usuários no sistema.				
<b>Prioridade:</b>	<input type="checkbox"/>	Essencial	<input checked="" type="checkbox"/>	Importante	<input type="checkbox"/> Desejável

## Requisitos Não Funcionais

Requisitos não funcionais são restrições sobre as funcionalidades fornecidas pelo sistema. Neste MVP são (Sommerville; 2018; p.89):

[RNF001] Limite de certificados cadastrados					
<b>Descrição:</b>	O sistema deve bloquear o cadastro de novos certificados caso o aluno já tenha atingido a carga horária de horas complementares definida no curso que está matriculado.				
<b>Prioridade:</b>	<input type="checkbox"/>	Essencial	<input checked="" type="checkbox"/>	Importante	<input type="checkbox"/> Desejável

[RNF002] Não edição de certificados homologados					
<b>Descrição:</b>	O sistema não deve permitir a edição de certificados homologados.				
<b>Prioridade:</b>	<input type="checkbox"/>	Essencial	<input checked="" type="checkbox"/>	Importante	<input type="checkbox"/> Desejável

[RNF003] A interface do sistema deve ser usável					
<b>Descrição:</b>	Interface deve ser simples e intuitiva para o usuário baseado nas heurísticas adaptadas de Jakob Nielsen. (da Rocha and Baranauskas; 2003; p.170).				
<b>Prioridade:</b>	<input type="checkbox"/>	Essencial	<input checked="" type="checkbox"/>	Importante	<input type="checkbox"/> Desejável

## Requisitos Postergados

Requisitos postergados são requisitos a serem considerados e até implementados, mas no futuro. Neste MVP são:

[RP001] Limite máximo de certificados cadastrados de um mesmo tipo					
<b>Descrição:</b>	O sistema deve bloquear o cadastro de novos certificados de um tipo específico, caso o máximo de horas destinadas para ele já tenha sido homologado.				
<b>Prioridade:</b>	<input type="checkbox"/>	Essencial	<input type="checkbox"/>	Importante	<input checked="" type="checkbox"/> Desejável

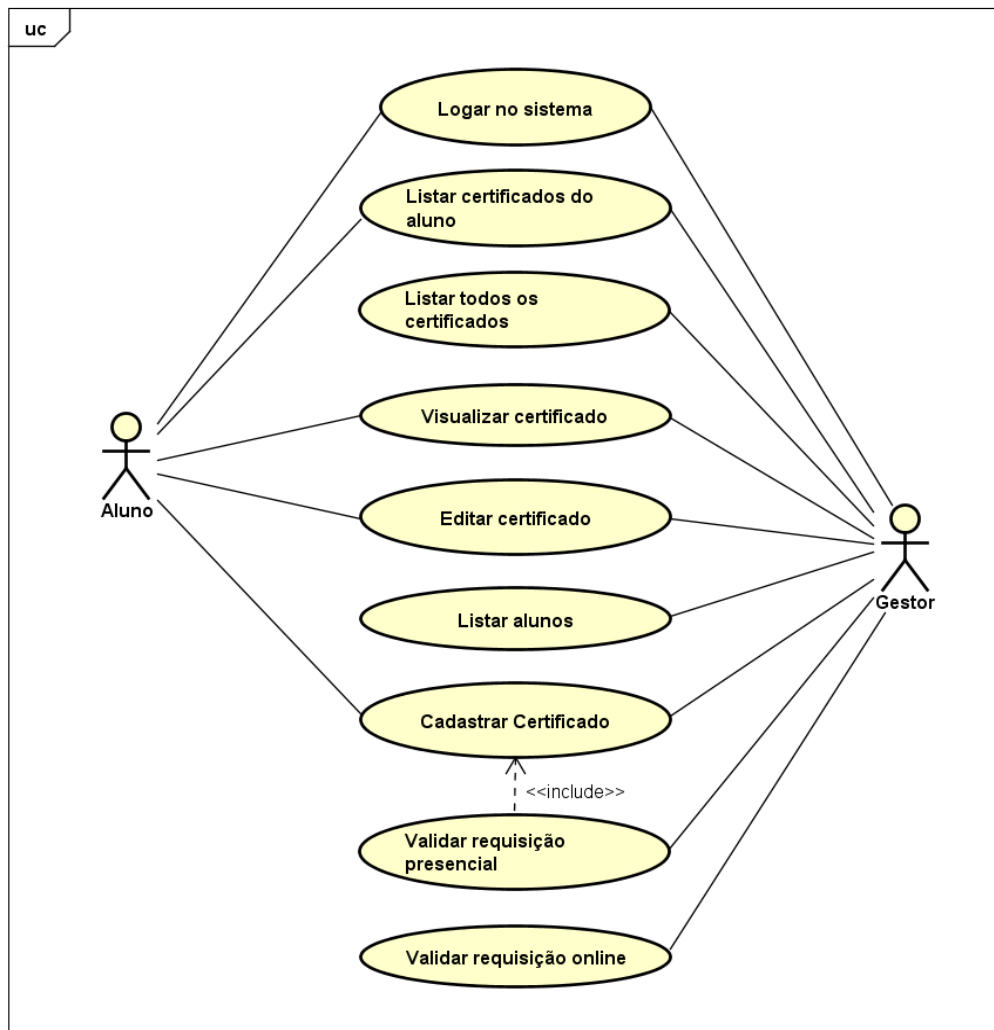
[RP002] Comentários de análise do certificado					
<b>Descrição:</b>	O sistema deve possibilitar o cadastro de comentários que justifiquem a mudança de situação do certificado.				
<b>Prioridade:</b>	<input type="checkbox"/>	Essencial	<input type="checkbox"/>	Importante	<input checked="" type="checkbox"/> Desejável

## Casos de Uso

Os Casos de uso ajudam a determinar a funcionalidade e as características de um sistema sob o ponto de vista dos usuários (Pressman and Maxim; 2016; p.875).

## Diagrama de Casos de Uso

O diagrama de casos de uso é um dos diagramas que compõem a Linguagem de Modelagem Unificada (*UML*), linguagem-padrão de modelagem adotada internacionalmente pela indústria de engenharia de software (Guedes; 2018; p.18). O diagrama de casos de uso tem como objetivo ilustrar os tipos de usuários que interagiram com o sistema, quais papéis eles irão assumir e quais funções um usuário específico poderá requisitar (Guedes; 2018; p.54). Abaixo está o diagrama de casos de uso do *MVP* em foco:



powered by Astah

## Descrição dos Casos de Uso

Para facilitar a compreensão do diagrama apresentado na seção anterior, foi elaborado uma descrição para cada caso de uso identificado até o momento.

UC001 – Logar no sistema	
Objetivo:	Acessar o sistema.
Requisitos:	RF006, RF007, RF012.
Atores:	Aluno ou Gestor.
Gatilho:	Acessar o sítio eletrônico do sistema.
Pré-condição:	O ator deve estar cadastrado no sistema.
Pós-condição:	O log da ação é gravado no sistema.
Fluxo principal	
Sistema	Aluno ou Gestor
	1. Insere os dados de acesso.
	2. Confirma os dados no sistema.
3. Faz a verificação dos dados.	
4. Concede acesso ao sistema.	
Fluxo Alternativo	
Sistema	Aluno ou Gestor
	2. Envia dados inconsistentes.
3. Não concede acesso ao sistema.	
4. Exibe mensagem de erro.	

<b>UC002 – Listar certificados do aluno</b>	
<b>Objetivo:</b>	Visualizar todos os certificados de um determinado aluno.
<b>Requisitos:</b>	RF005, RF012
<b>Atores:</b>	Aluno ou Gestor.
<b>Gatilho:</b>	Seleção da opção correspondente ao caso de uso.
<b>Pré-condição:</b>	O ator deve estar logado no sistema; Existir no sistema, no mínimo, um certificado cadastrado e vinculado ao aluno selecionado.
<b>Pós-condição:</b>	O log da ação é gravado no sistema.
<b>Fluxo principal</b>	
<b>Sistema</b>	<b>Aluno ou Gestor</b>
1. Busca todos os certificados do aluno selecionado.	
2. Encontra todos os certificados do aluno selecionado.	
3. Exibe os registros encontrados.	
<b>Fluxo Alternativo</b>	
<b>Sistema</b>	<b>Aluno ou Gestor</b>
2. Não encontra registros de certificados.	
3. Exibe mensagem informando que nenhum registro foi encontrado.	

<b>UC003 – Listar todos os certificados</b>	
<b>Objetivo:</b>	Visualizar todos os certificados cadastrados.
<b>Requisitos:</b>	RF009, RF012.
<b>Atores:</b>	Gestor.
<b>Gatilho:</b>	Seleção da opção correspondente ao caso de uso.
<b>Pré-condição:</b>	O ator deve estar logado no sistema; Existir no sistema, no mínimo, um certificado cadastrado no sistema.
<b>Pós-condição:</b>	O log da ação é gravado no sistema.
<b>Fluxo principal</b>	
<b>Sistema</b>	<b>Gestor</b>
1. Busca todos os certificados.	
2. Encontra todos os certificados.	
3. Exibe os registros encontrados.	
<b>Fluxo Alternativo</b>	
<b>Sistema</b>	<b>Gestor</b>
2. Não encontra registros de certificados .	
3. Exibe mensagem informando que nenhum registro foi encontrado.	

<b>UC004 – Listar alunos</b>	
<b>Objetivo:</b>	Visualizar todos os alunos cadastrados.
<b>Requisitos:</b>	RF008, RF012.
<b>Atores:</b>	Gestor.
<b>Gatilho:</b>	Seleção da opção correspondente ao caso de uso.
<b>Pré-condição:</b>	O ator deve estar logado no sistema; Existir no sistema, no mínimo, um alunos cadastrado.
<b>Pós-condição:</b>	O log da ação é gravado no sistema.
<b>Fluxo principal</b>	
<b>Sistema</b>	<b>Gestor</b>
1. Busca todos os registros de alunos.	

2. Encontra todos os registros de alunos.	
3. Exibe os registros encontrados.	
<b>Fluxo Alternativo</b>	
<b>Sistema</b>	<b>Gestor</b>
2. Não encontra registros de alunos.	
3. Exibe mensagem informando que nenhum registro foi encontrado.	

<b>UC005 – Visualizar certificados</b>	
<b>Objetivo:</b>	Visualizar todos os dados de um certificado cadastrado.
<b>Requisitos:</b>	RF010, RF012.
<b>Atores:</b>	Gestor.
<b>Gatilho:</b>	Seleção da opção correspondente ao caso de uso.
<b>Pré-condição:</b>	O ator deve estar logado no sistema; Existir no sistema, no mínimo, um certificado cadastrado.
<b>Pós-condição:</b>	O log da ação é gravado no sistema.
<b>Fluxo principal</b>	
<b>Sistema</b>	<b>Gestor</b>
1. Busca o registro dos certificados.	
2. Encontra o registro dos certificados.	
3. Exibe os dados do registro encontrado.	

<b>UC006 – Cadastrar certificado</b>	
<b>Objetivo:</b>	Guardar os dados de um novo certificado
<b>Requisitos:</b>	RF001, RF002, RF004, RF012.
<b>Atores:</b>	Aluno ou Gestor.
<b>Gatilho:</b>	Seleção da opção correspondente ao caso de uso.
<b>Pré-condição:</b>	O ator deve estar logado no sistema;
<b>Pós-condição:</b>	O log da ação é gravado no sistema.
<b>Fluxo principal</b>	
<b>Sistema</b>	<b>Aluno ou Gestor</b>
	1. Preenche o formulário.
	2. Confirma os dados no sistema.
3. Faz a verificação dos dados.	
4. Calcula a carga horária total de certificados do aluno.	
5. Grava os dados no banco de dados com êxito.	
6. Exibe mensagem de êxito.	

<b>UC007 – Editar certificado</b>	
<b>Objetivo:</b>	Atualiza os dados de um certificado
<b>Requisitos:</b>	RF011; RF002, RF004, RF012.
<b>Atores:</b>	Aluno ou Gestor.
<b>Gatilho:</b>	Seleção da opção correspondente ao caso de uso.
<b>Pré-condição:</b>	O ator deve estar logado no sistema; Para o ator Aluno o certificado não pode estar na situação de homologado.
<b>Pós-condição:</b>	O log da ação é gravado no sistema.
<b>Fluxo principal</b>	
<b>Sistema</b>	<b>Aluno ou Gestor</b>
	1. Altera os dados
	2. Confirma os dados no sistema.



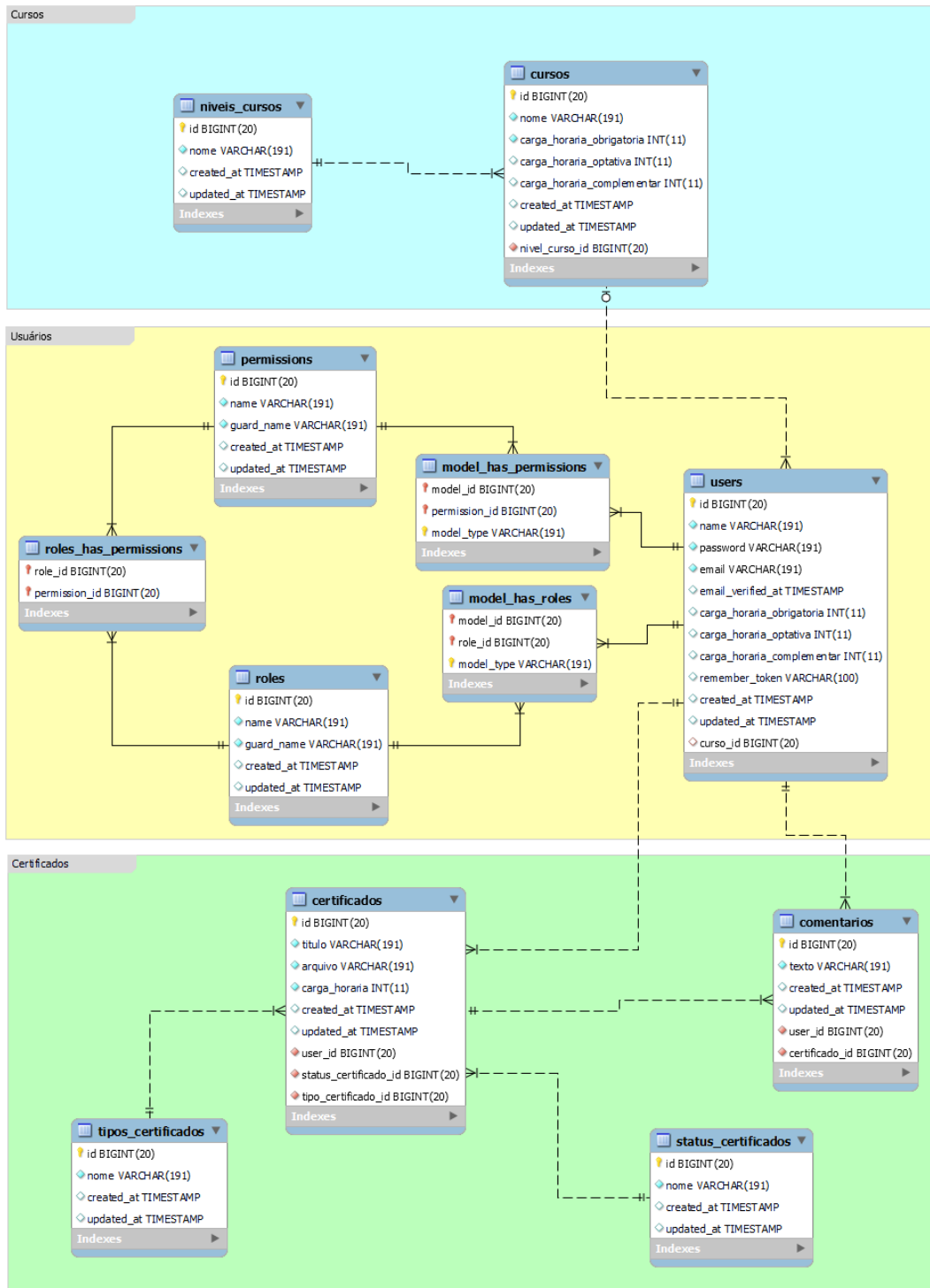
3. Faz a verificação dos dados.	
4. Calcula a carga horária total de certificados do aluno.	
5. Grava os dados no banco de dados com êxito.	
6. Exibe mensagem de êxito.	

UC008 – Validar requisições presenciais	
<b>Objetivo:</b>	Guardar os dados de um novo certificado
<b>Requisitos:</b>	RF011; RF003, RF012
<b>Atores:</b>	Gestor.
<b>Gatilho:</b>	Solicitação do aluno;
<b>Pré-condição:</b>	O ator deve estar logado no sistema;
<b>Pós-condição:</b>	O log da ação é gravado no sistema.
Fluxo principal	
Sistema	Gestor
	1. Recebe o certificado físico do aluno.
	2. Analisa o certificado.
	3. Digitaliza o certificado.
	4. Executa o caso de uso “UC006”.

UC009 – Validar requisições online	
<b>Objetivo:</b>	Homologar ou não um certificado
<b>Requisitos:</b>	RF011; RF003, RF012.
<b>Atores:</b>	Gestor.
<b>Gatilho:</b>	Edição de um certificado.
<b>Pré-condição:</b>	O ator deve estar logado no sistema.
<b>Pós-condição:</b>	O log da ação é gravado no sistema.
Fluxo principal	
Sistema	Gestor
	1. Analisa o certificado.
	2. Edita o campo situação para ‘homologado’ ou ‘não homologado’.
	3. Confirma os dados no sistema.
4. Faz a verificação dos dados.	
5. Calcula a carga horária total de certificados do aluno.	
6. Grava os dados no banco de dados com êxito.	
7. Exibe mensagem de êxito.	

## Arquitetura do Banco de Dados

Para demonstrar a organização lógica do banco de dados do *MVP* em foco, foi criado o diagrama entidade-relacionamento estendido (*EER*) apresentado abaixo:



## Tecnologias Utilizadas

Para a construção deste *MVP* foi utilizado o *framework* — pacote de códigos prontos que podem ser utilizados no desenvolvimento de sistemas — Laravel. Ele dispõe de uma sintaxe simples e elegante e conta com um conjunto de ferramentas úteis para o desenvolvimento de sistemas baseados em arquiteturas de micro-serviços ou monolíticas, como esta. A seguir serão listados os recursos do Laravel utilizados:

- **Migrations:** facilita a construção de tabelas em diferentes bancos de dados relacionais, como o *MySQL* que foi utilizado neste projeto (Laravel).
- **Seeds:** permite povoar com dados pré-definidos as tabelas criadas.
- **Eloquent:** torna agradável a interação do sistema com o banco de dados.

- *Routes*: permite criar os endereços que serão acessados pelos usuários do sistema.
- *Models*: gerenciam o comportamento dos dados utilizados considerando as regras de negócio inerentes ao sistema.
- *Controllers*: organizam a lógica relacionada a cada endereço definido no arquivo de *routes*.
- *Views*: responsável por apresentar informações para os usuários do sistema<sup>1</sup>.
- *Validation*: responsável por validar os dados preenchidos nos formulários das *views* pelos usuários do sistema.

## Referências

- Sommerville, Ian. **Engenharia de Software**. 10 ed., São Paulo, Pearson, 2018.
- Pressman, Roger S., and Bruce R. Maxim. **Engenharia de Software: Uma Abordagem Profissional**. 8 ed., Porto Alegre, Bookman, 2016.
- Guedes, Gilleanes T. A. **UML 2: Uma Abordagem Prática**. 3 ed., São Paulo, Novatec, 2018.
- da Rocha, Heloísa Vieira, and Maria Cecília Baranauskas. **Design e Avaliação de Interfaces Humano-Computador**. UNIVERSIDADE ESTADUAL DE CAMPINAS, 2003.

---

<sup>1</sup> O Laravel apresenta um conjunto de tecnologias específicas para a construção de *views*, neste MVP foram utilizados:

- *HTML*: linguagem de marcação de texto, muito utilizada para estruturar páginas *web*.
- *CSS*: código utilizado para estilizar páginas *web*.
- *SASS*: linguagem de programação que após compilada gera código CSS.
- *JavaScript*: permite a implementação de recursos mais elaborados e complexos, como animações, em páginas *web*.
- *Blade*: possibilita utilizar recursos específicos do Laravel nas *views*.