**Relatório de Especificação e Modelagem de Casos de Uso para o Sistema de Biblioteca Digital**

**1. Introdução à Engenharia de Software e Contextualização do Projeto**

Este relatório visa aplicar os princípios fundamentais da Engenharia de Software, especificamente a **especificação de requisitos** e a **modelagem de Casos de Uso (UC) em UML**, ao projeto de um novo **Sistema de Biblioteca Digital** para uma instituição universitária. O objetivo principal do novo sistema é modernizar e otimizar o acesso ao acervo acadêmico, resolvendo as dificuldades atuais enfrentadas por alunos, professores e bibliotecários.

A fase de requisitos é crucial, pois define o "o quê" do sistema, garantindo que o produto final atenda às necessidades das partes interessadas (stakeholders). A modelagem por Casos de Uso, como ferramenta visual e narrativa, será empregada para documentar as interações funcionais do sistema, servindo como uma base sólida para as etapas subsequentes de desenvolvimento.

**2. Levantamento de Requisitos e Atores**

A primeira etapa consiste em identificar os atores envolvidos e as funcionalidades (Casos de Uso) essenciais que o sistema deve suportar para resolver os problemas e atender às expectativas do cenário proposto.

**2.1. Atores Identificados**

Os atores representam papéis que interagem diretamente com o sistema ou que são impactados por ele.

|  |  |
| --- | --- |
| **Ator Principal** | **Descrição** |
| **Aluno** | Usuário primário que busca e utiliza o acervo para fins de estudo. |
| **Professor** | Usuário que utiliza o acervo, geralmente com maiores privilégios de empréstimo/reserva, e pode ser um ator secundário em outras funções (ex.: sugestão de compra). |
| **Bibliotecário** | Ator com permissões administrativas, responsável pela gestão e manutenção do acervo e dos registros de empréstimos. |
| **Sistema** | Ator secundário que realiza ações automáticas, como o envio de notificações. |

**2.2. Casos de Uso Principais**

Estes são os serviços funcionais de alto nível que o sistema deve fornecer aos atores.

|  |  |  |
| --- | --- | --- |
| **Grupo Funcional** | **Casos de Uso (UC)** | **Atores Principais** |
| **Acesso e Consulta** | UC01: Autenticar-se | Aluno, Professor, Bibliotecário |
|  | UC02: Consultar Catálogo | Aluno, Professor |
| **Gestão do Acervo** | UC03: Cadastrar Nova Obra | Bibliotecário |
|  | UC04: Atualizar Informações de Obra | Bibliotecário |
| **Gestão de Empréstimos** | UC05: Realizar Empréstimo | Aluno, Professor |
|  | UC06: Renovar Prazo | Aluno, Professor |
|  | UC07: Reservar Título | Aluno, Professor |
|  | UC08: Acompanhar Empréstimos | Bibliotecário |
| **Notificações** | UC09: Enviar Notificação de Devolução Próxima | Sistema |
|  | UC10: Devolver Obra | Aluno, Professor, Bibliotecário |

**2.3. Requisitos Fundamentais (Funcionais e Não Funcionais)**

Estes são requisitos essenciais para o funcionamento adequado do Sistema de Biblioteca Digital.

**Requisitos Funcionais (RF)**

1. **RF-01 (Autenticação):** O sistema DEVE permitir a autenticação de Alunos, Professores e Bibliotecários por meio de credenciais válidas da universidade.
2. **RF-02 (Busca):** O sistema DEVE permitir a busca no catálogo por título, autor, ISBN, palavra-chave e categoria, com resultados em tempo real.
3. **RF-03 (Disponibilidade):** O sistema DEVE exibir o *status* de disponibilidade de cada obra (disponível, emprestado, reservado) e sua localização física/digital.
4. **RF-04 (Empréstimo):** O sistema DEVE registrar um empréstimo somente se o usuário estiver ativo e não exceder o limite de obras.
5. **RF-05 (Notificação):** O sistema DEVE enviar automaticamente um e-mail de notificação aos usuários 3 dias antes da data de devolução programada.
6. **RF-06 (Restrição de Reserva):** O sistema DEVE permitir reservas apenas para obras que estejam com o *status* "Emprestado".

**Requisitos Não Funcionais (RNF)**

1. **RNF-01 (Usabilidade):** A interface do usuário DEVE ser responsiva e intuitiva, com tempo de carregamento inferior a 3 segundos em conexões padrão.
2. **RNF-02 (Segurança):** O sistema DEVE criptografar todas as senhas armazenadas e utilizar protocolo HTTPS/SSL.
3. **RNF-03 (Performance):** A consulta ao catálogo (RF-02) DEVE ser concluída em menos de 1 segundo, independentemente do tamanho do acervo.
4. **RNF-04 (Disponibilidade):** O sistema DEVE estar acessível 24 horas por dia, 7 dias por semana (24/7) com um *uptime* mínimo de 99,5%.

**3. Diagrama de Casos de Uso UML**

O Diagrama de Casos de Uso UML representa a fronteira do sistema e as relações entre os atores e as funcionalidades identificadas.

**3.1. Representação Gráfica e Relações**

A estrutura do diagrama é descrita abaixo, utilizando a notação padrão da UML, focando nas associações e nas relações de dependência.

**Sistema: Biblioteca Digital**

|  |  |  |
| --- | --- | --- |
| **Ator** | **Casos de Uso (UC) Associados** | **Relações Específicas** |
| **Aluno / Professor** | Consultar Catálogo (UC02) | Autenticar-se (UC01) |
|  | Realizar Empréstimo (UC05) | Autenticar-se (UC01) |
|  | Renovar Prazo (UC06) | Autenticar-se (UC01) |
|  | Reservar Título (UC07) | Autenticar-se (UC01) |
|  | Devolver Obra (UC10) | Autenticar-se (UC01) |
| **Bibliotecário** | Autenticar-se (UC01) | Associa-se diretamente a UC01 |
|  | Cadastrar Nova Obra (UC03) | Autenticar-se (UC01) |
|  | Atualizar Informações de Obra (UC04) | Autenticar-se (UC01) |
|  | Acompanhar Empréstimos (UC08) | Autenticar-se (UC01) |
|  | Devolver Obra (UC10) | Autenticar-se (UC01) |
| **Sistema** | Enviar Notificação de Devolução Próxima (UC09) | Associa-se a um Cronômetro (Evento Temporal) |

**Exemplo de Relações:**

* **Relação de Inclusão:** O Caso de Uso **"Realizar Empréstimo (UC05)"** **INCLUI** o Caso de Uso **"Autenticar-se (UC01)"**. Isso significa que a autenticação é uma etapa obrigatória e comum que ocorre sempre que o UC05 é executado.
* **Relação de Extensão:** Poderíamos modelar a funcionalidade de **"Solicitar Compra de Obra"** como uma EXTENSÃO do Caso de Uso **"Consultar Catálogo (UC02)"**. O ponto de extensão seria o usuário não encontrar a obra desejada.

**4. Especificação Completa do Caso de Uso "Realizar Empréstimo" (UC05)**

O Caso de Uso **Realizar Empréstimo** é fundamental e será detalhado a seguir, incluindo seu protótipo de tela.

|  |  |
| --- | --- |
| **Campo** | **Detalhamento** |
| **Nome do Caso de Uso** | UC05: Realizar Empréstimo |
| **Ator Principal** | Aluno, Professor |
| **Ator Secundário** | Bibliotecário (Confirmação Física/Registro), Sistema (Verificação de Status) |
| **Resumo** | Permite que um usuário autenticado solicite o registro de uma obra disponível para empréstimo, respeitando as restrições e limites definidos pelo sistema. |
| **Pré-condições** | 1. O Ator Principal (Aluno ou Professor) deve estar autenticado no sistema (UC01). 2. A obra desejada deve ter o *status* "Disponível" e não estar reservada por outro usuário. 3. O Ator Principal não deve ter pendências (multas, devoluções atrasadas). 4. O Ator Principal não deve ter atingido o limite máximo de obras emprestadas. |
| **Pós-condições** | 1. O *status* da obra é alterado para "Emprestado". 2. Um novo registro de empréstimo é criado no banco de dados, contendo Ator, Obra, Data de Empréstimo e Data de Devolução Prevista. 3. A quantidade disponível da obra é decrementada em uma unidade. 4. O Ator Principal recebe um comprovante eletrônico do empréstimo. |
| **Fluxo Principal de Eventos** | **1. Início do UC:** Ator Acessa a função de Empréstimo. **2. Ator:** Informa o código/ISBN da obra e seu próprio ID/Matrícula. **3. Sistema:** Verifica se o Ator está autenticado e ativo. **4. Sistema:** Consulta se a obra está disponível e se o Ator atingiu o limite de empréstimos (Restrição). **5. Sistema:** Se as validações forem aprovadas, o sistema registra o empréstimo (cria o registro e calcula a Data de Devolução). **6. Sistema:** Atualiza o *status* da obra no catálogo para "Emprestado". **7. Sistema:** Exibe a mensagem de sucesso e o comprovante com a data de devolução. **8. Fim do UC.** |
| **Fluxos Alternativos ou de Exceção** | **FA-01: Limite Excedido:** No Passo 4, o sistema verifica que o Ator já atingiu o limite (ex.: Aluno com 5 livros). O sistema informa: "Limite de empréstimos excedido." O UC termina. **FE-02: Obra Indisponível/Reservada:** No Passo 4, a obra não está disponível. O sistema informa: "Obra indisponível no momento. Deseja Reservar? (UC07)". O UC termina. **FE-03: Multa Pendente:** No Passo 3, o sistema identifica uma multa não paga. O sistema informa: "Empréstimo bloqueado devido a pendências financeiras." O UC termina. |
| **Restrições e Validações** | **Regra de Negócio:** **Limites:** Aluno: 5 livros por 15 dias. Professor: 10 livros por 30 dias. **Validação:** Não permitir empréstimo para usuários com pendências (multas ou atrasos). **Validação:** Não permitir empréstimo de obras reservadas, a menos que o Ator seja o reservante. |

**5. Análise Crítica da Modelagem por Casos de Uso**

A modelagem por Casos de Uso (UC) é uma técnica poderosa, mas, como toda ferramenta, possui vantagens e limitações. Uma análise crítica de seu uso é fundamental para o sucesso do projeto.

**5.1. Facilitação do Entendimento dos Requisitos**

Os Casos de Uso servem como uma ponte de comunicação indispensável entre os três principais grupos envolvidos no ciclo de vida do software:

* **Clientes e Usuários Finais:** A natureza narrativa do UC permite que o cliente (a universidade) e os usuários (alunos, professores, bibliotecários) compreendam o sistema em sua própria linguagem, focada em suas tarefas e objetivos. Eles não precisam entender a tecnologia subjacente; basta ler o **Fluxo Principal de Eventos** para validar se o sistema fará o que eles precisam. O Diagrama de UC oferece uma visão de alto nível, permitindo que o cliente confirme se todas as suas necessidades foram mapeadas.
* **Desenvolvedores:** Para a equipe técnica, o UC detalhado (como o do **Realizar Empréstimo**) é a base para o desenvolvimento. O fluxo de eventos se traduz diretamente em código e lógica de negócio. As **Pré-condições**, **Pós-condições** e **Restrições** fornecem as regras de negócio e os limites de teste que a aplicação deve respeitar, servindo como requisitos de teste.

**5.2. Vantagens e Limitações**

**Vantagens**

|  |  |
| --- | --- |
| **Vantagem** | **Explicação** |
| **Clareza e Comunicação** | Os UCs são escritos em linguagem natural, facilitando a comunicação e minimizando ambiguidades, pois definem o que o sistema *faz* de maneira simples e concisa. |
| **Rastreabilidade** | Cada UC é uma unidade de funcionalidade. Isso permite rastrear o requisito desde a documentação inicial, passando pelo código que o implementa, até os casos de teste que o validam. |
| **Definição da Fronteira do Sistema** | O Diagrama de UC define claramente o que está dentro (funcionalidades) e o que está fora (o ambiente e os atores) do escopo do projeto, evitando o desvio de escopo (*scope creep*). |
| **Priorização** | Os UCs podem ser priorizados, permitindo que a equipe foque nos mais importantes para o cliente, como o **Realizar Empréstimo**. |

**Limitações**

|  |  |
| --- | --- |
| **Limitação** | **Explicação** |
| **Foco Apenas Funcional** | Os UCs são excelentes para descrever *o que* o sistema faz (funcionalidade), mas são fracos para descrever *como* (desempenho, segurança, usabilidade). Isso deve ser complementado por Requisitos Não Funcionais separados (RNFs), como visto no item 2.3. |
| **Ausência de Detalhes Técnicos** | O UC não descreve a arquitetura, a tecnologia de banco de dados ou as classes internas do sistema, sendo necessária a complementação com outros diagramas UML (Classes, Sequência, Componentes) para o desenvolvimento. |
| **Custo de Elaboração** | A especificação completa e detalhada de todos os UCs de um sistema grande pode ser demorada e custosa, exigindo um alto esforço inicial de documentação. |

**5.3. Comparação com Histórias de Usuário (*User Stories*)**

|  |  |  |
| --- | --- | --- |
| **Característica** | **Casos de Uso (UC)** | **Histórias de Usuário (User Stories)** |
| **Formato** | Narrativa estruturada e detalhada (tabela). | Cartão conciso (Formato: "Como Ator, eu quero Objetivo para Benefício). |
| **Detalhe** | Alto nível de detalhe (pré/pós-condições, fluxos alternativos). | Baixo nível de detalhe (reforçado pela "Conversa" e "Confirmação"). |
| **Adequação de Contexto** | Sistemas grandes, complexos, de missão crítica ou com requisitos regulatórios rigorosos (ex.: Bancos, Telecomunicações). | Projetos Ágeis e enxutos, onde a flexibilidade e a entrega rápida de valor são prioridades (ex.: Startups, Módulos Web). |
| **Foco** | O sistema e sua interação formal. | O valor e o benefício para o usuário. |

Os Casos de Uso são mais adequados quando a **integridade da documentação** é crítica e a complexidade exige o mapeamento explícito de todas as condições e exceções. As Histórias de Usuário são preferíveis em ambientes Ágeis onde a **colaboração constante** substitui a documentação extensa.

**5.4. Alinhamento e Qualidade da Documentação**

O uso de Casos de Uso contribui significativamente para o alinhamento entre a equipe técnica e o cliente:

1. **Validação Compartilhada:** O documento de UC (narrativa) é o artefato que o cliente pode entender e assinar, garantindo que o que será construído é o que ele realmente precisa. Isso é a base do alinhamento.
2. **Qualidade da Documentação:** A especificação de UC força a equipe a pensar em todos os caminhos possíveis (*happy path*, fluxos alternativos, exceções). Isso resulta em uma documentação de requisitos mais completa e menos ambígua. A clareza das **Pré-condições** e **Pós-condições** elimina suposições sobre o estado do sistema antes e depois de cada ação, o que é um pilar da qualidade do requisito.

**Conclusão**

Este relatório demonstrou a aplicação bem-sucedida da **especificação de requisitos** e da **modelagem por Casos de Uso (UML)** no projeto do Sistema de Biblioteca Digital. A identificação clara dos atores (**Aluno, Professor, Bibliotecário**) e o mapeamento funcional detalhado, exemplificado pelo Caso de Uso **Realizar Empréstimo**, permitiram delimitar o escopo e as regras de negócio de forma rigorosa. O uso desta metodologia não apenas garante a **rastreabilidade** e a clareza para os desenvolvedores, mas, sobretudo, estabelece o **alinhamento** crucial entre as necessidades da Universidade e a solução técnica proposta, assegurando a qualidade final da documentação e do sistema.