

Projeto Integrador:

Sistema para Igreja



**1142282439 Enrico Andrade -**

**1142104809 Gabriel Jesus -**

**1142623208 Italo Daniel -**

**1142615686 Kaio Paulo -**

Projeto Integrador:

Sistema para Igreja

Sumário

[**EXEMPLO DE DOCUMENTO DE ESPECIFICAÇÃO DE SOFTWARE 3**](#_heading=h.gjdgxs)

[**1. INTRODUÇÃO 3**](#_heading=h.30j0zll)

[**2. VISÃO GERAL 3**](#_heading=h.1fob9te)

[**2.1. O Projeto e a contribuição à comunidade 3**](#_heading=h.3znysh7)

[**2.2. ESG e as ODS contempladas no projeto 3**](#_heading=h.2et92p0)

[**2.3. Justificativa 3**](#_heading=h.tyjcwt)

[**3. ESCOPO DO PROJETO 4**](#_heading=h.3dy6vkm)

[**3.1. Matriz de papéis e responsabilidades 4**](#_heading=h.1t3h5sf)

[**4. CLIENTE 4**](#_heading=h.4d34og8)

[**5. DESCRIÇÃO DOS REQUISITOS 4**](#_heading=h.2s8eyo1)

[**5.1 REQUISITOS FUNCIONAIS: 4**](#_heading=h.17dp8vu)

[**5.2 REQUISITOS NÃO FUNCIONAIS 5**](#_heading=h.3rdcrjn)

[**5.3 REGRAS DE NEGÓCIO 6**](#_heading=h.26in1rg)

[**6. PRODUCT BACKLOG INICIAL 6**](#_heading=h.lnxbz9)

[**6.1. Divisão de Sprints ou Gantt 7**](#_heading=h.35nkun2)

[**6.2. Histórias de Usuário 7**](#_heading=h.1ksv4uv)

[**7. ARQUITETURA DE SOFTWARE 7**](#_heading=h.44sinio)

[**7.1. Padrão de Projeto e/ou Estilo Arquitetural 7**](#_heading=h.2jxsxqh)

[**8. Design centrado no usuário 7**](#_heading=h.z337ya)

[**8.1. Mapa de Empatia 8**](#_heading=h.3j2qqm3)

[**8.2. PERSONAS / Protopersonas 8**](#_heading=h.1y810tw)

[**9. MODELO DE CASOS DE USO 8**](#_heading=h.4i7ojhp)

[**9.1 IDENTIFICAÇÃO DOS ATORES E SUAS RESPONSABILIDADES 8**](#_heading=h.2xcytpi)

[**9.2 DEFINIÇÃO DE PRIORIDADE DE DESENVOLVIMENTO DOS CASOS DE USO 8**](#_heading=h.1ci93xb)

**9.3 DESCRIÇÃO DETALHADA DO CASO DE USO 8**

# EXEMPLO DE DOCUMENTO DE ESPECIFICAÇÃO DE SOFTWARE

Este projeto será um site para uma igreja com cadastro e controle de usuários, atualização de informações e avisos, marcação de eventos como casamentos e batismos online, com garantia de segurança de dados e acesso controlado pelos membros administrativos da igreja, como aprovação e não aprovação de eventos.

Neste projeto serão contempladas duas ODS, sendo elas:

**ODS N°16. Paz, Justiça e Instituições Eficazes**

“Promover sociedades pacíficas e inclusivas para o desenvolvimento sustentável, proporcionar o acesso à justiça para todos e construir instituições eficazes, responsáveis e inclusivas em todos os níveis”

**ODS N° 12. Consumo e produção responsáveis**

“Garantir padrões de consumo e de produção sustentáveis”

# 1. INTRODUÇÃO

Objetivo deste documento é coletar, analisar e definir as características e necessidades de alto nível da igreja. O foco do trabalho está nas capacidades necessárias para os envolvidos e usuários finais, e porque estas necessidades existem. Os detalhes de como a igreja atende estas necessidades estão detalhados no trabalho abaixo. Este trabalho apresenta o desenvolvimento de um sistema de gestão para uma igreja, focando em facilitar o gerenciamento de membros, agendamentos de eventos e comunicação por meio de avisos. O sistema tem como objetivo principal oferecer uma plataforma eficiente para membros e administradores da igreja, otimizando processos e proporcionando uma melhor experiência para todos os envolvidos. O gerenciamento de uma igreja envolve diversas atividades administrativas e de comunicação que podem ser otimizadas por meio da tecnologia. Neste contexto, o desenvolvimento de um sistema de gestão surge como uma solução viável para simplificar tarefas e melhorar a interação entre membros e líderes religiosos.

# 

# 2. VISÃO GERAL

O objetivo deste trabalho é apresentar o processo de desenvolvimento de um sistema de gestão para uma igreja, abordando desde a identificação das necessidades até a implementação e teste do sistema. Além disso, busca-se fornecer uma visão clara dos benefícios e funcionalidades oferecidos pelo sistema.

**Segurança de dados:** Restrições de segurança devem ser rigorosamente observadas para proteger os dados dos usuários da igreja. Isso inclui criptografia de senhas, controle de acesso e medidas para proteção contra ameaças de segurança.

**Conexão a internet:** O sistema terá constante conexão com a internet, então será necessário que qualquer usuário tenha conexao.

**Escalabilidade:** Embora o sistema seja de pequeno porte, será projetado de forma a permitir futuras expansões e escalabilidade, em caso a igreja cresça ou necessite de mais recursos.

**Manutenção e suporte:** O sistema deve ser documentado adequadamente para permitir sua manutenção contínua e prestação de suporte aos administradores da igreja.

**Interface amigável e intuitiva:** O sistema deverá ter em sua implementação uma interface agradável que não confunda o usuário e que o guie de forma didática durante sua experiencia, contribuindo com a inclusão digital.

# 2.1. O Projeto e a contribuição à comunidade

Através do site as pessoas poderão marcar um casamento ou batizado sem precisar ir até o local. A igreja poderá anunciar seus eventos e avisos com todas as informações importantes para o público de forma rápida e direta, contribuindo com a diminuição do papel em nosso meio ambiente.

A construção do projeto envolveu etapas como levantamento de requisitos, análise de viabilidade, definição de arquitetura e desenvolvimento do sistema. Foram utilizadas tecnologias como banco de dados no mySql, linguagens de programação como HTML, CSS e JavaScript e frameworks para garantir a eficiência e segurança do sistema.

# 2.2. ESG e as ODS contempladas no projeto

**ODS contempladas no projeto:**

Aplicativo

Descrição gerada automaticamente com confiança baixaODS N°16. Paz, Justiça e Instituições Eficazes

“Promover sociedades pacíficas e inclusivas para o desenvolvimento sustentável, proporcionar o acesso à justiça para todos e construir instituições eficazes, responsáveis e inclusivas em todos os níveis”

“16.7 Garantir a tomada de decisão responsiva, inclusiva, participativa e representativa em todos os níveis”

Uma imagem contendo Ícone

Descrição gerada automaticamenteODS N° 12. Consumo e produção responsáveis

“Garantir padrões de consumo e de produção sustentáveis”

“12.5 Até 2030, reduzir substancialmente a geração de resíduos por meio da prevenção, redução, reciclagem e reuso”

ESG contempladas no projeto:

**AMBIENTAL**: Para minimizar o impacto no meio ambiente, a igreja reduzira o uso de papel, como por exemplo em panfletos com informações sobre as próximas missas etc, marcação de eventos como casamentos e batismos serão feitos online, assim diminuindo a produção a produção de resíduos.

**SOCIAL:** Visando um ambiente inclusivo e tratamento justo para todos os colaboradores, sera implementado um sistema interface inituitiva com fonte em alto contraste e legível para todos, junto com um sistema de acessibilidade em libras para deficientes visuais.

**2.3. Justificativa**

Igrejas nem sempre tem um sistema virtual e na maioria dos casos seus colaboradores não possuem conhecimento para criação de tal coisa, fazendo com que seus membros tenham que ir pessoalmente a igreja para busca de informações e marcações de eventos, se limitando sempre ao uso do papel.

Com isso em mente decidimos criar um sistema para auxílio tanto de membros que trabalham na igreja, quanto para colaboradores que buscam informações da igreja sem precisar sair de casa.

O levantamento do problema revelou que a igreja enfrentava dificuldades no gerenciamento de informações dos membros, agendamento de eventos e comunicação interna. Com base nessa pesquisa, foram identificadas as funcionalidades necessárias para solucionar esses problemas, como cadastro de membros, agendamento de eventos e gerenciamento de avisos.

**ODS N°16. Paz, Justiça e Instituições Eficazes**

Atraves de nosso site, sistema será responsivo para que as decisões possam ser tomadas de forma rápida e adaptativa. Teremos uma interface inclusiva e acessível para todos os públicos, com fontes com boas legibilidade e alto contraste com uma ferramenta em libras para deficientes visuais.

**ODS N° 12. Consumo e produção responsáveis**

Com a criação do site a igreja reduzira o consumo de papel de papel em seu dia a dia, já que informações sobre missas e avisos de eventos, marcações de casamentos e batismos serão todos feitos virtualmente.

# 3. ESCOPO DO PROJETO

O escopo deste projeto inclui o desenvolvimento de um sistema de gerenciamento de usuários da igreja e avisos para web. As funcionalidades principais que farão parte do produto são:

* Cadastro de usuários da igreja com informações detalhadas;
* Atualização e exclusão de informações de usuários da igreja;
* Listagem de usuários da igreja para consulta;
* Cadastro de avisos para comunicação interna;
* Atualização e exclusão de avisos;
* Visualização de avisos pelos usuários da igreja;
* Agendamento de eventos pelos usuários;
* Aprovação de agendamento pelo administrador;
* Garantia de segurança de dados e acesso controlado.
* Streaming de uma missa
* Atalhos para redes sociais da igreja

**3.1. Matriz de papéis e responsabilidades**

Documentação: Gabriel Jesus e Kaio Paulo

Desenvolvimento Front-end: Italo Daniel e Kaio Paulo

Desenvolvimento Back-end: Enrico Andrade e Gabriel Jesus

# 4. CLIENTE

Nome: SENAC - SP

Ramo de Atividade: Estabelecimento de ensino em nível médio e superior tanto graduação quanto pós graduação, com cursos em todas as áreas do conhecimento.

# 5. DESCRIÇÃO DOS REQUISITOS

**5.1 REQUISITOS FUNCIONAIS:**

Os seguintes requisitos funcionais foram encontrados após o levantamento de requisitos inicial do Sistema da <Nome da Igreja>

RF01 – Sistema deve implementar criptografia nos campos que se inserem senhas para garantir a segurança dos dados dos usuários.

RF02 – O sistema deve permitir o cadastro de usuários da igreja com informações detalhadas, como: nome, endereço, CPF e informações de contato.

RF03 – O administrador deve conseguir aprovar ou rejeitar os agendamentos feitos pelos membros da igreja.

RF04 – Os administradores da igreja devem conseguir atualizar as informações cadastrais dos membros e fazer a exclusão de usuários da igreja quando necessário.

RF05 – O sistema deve fornecer uma funcionalidade para listar usuários da igreja para consulta, permitindo a busca por identificador.

RF06 – Deve ser possível cadastrar avisos de fácil visualização para os membros da igreja.

RF07 – Deve haver controle de acesso, permitindo diferentes níveis de permissões para membros e administradores.

RF08 – O administrador deve poder aprovar ou rejeitar os cadastros feitos pelos membros da igreja.

RF09 - Os avisos devem ser atualizáveis, a fim de serem feitas modificações e exclusões quando não forem mais relevantes.

**5.2 REQUISITOS NÃO FUNCIONAIS**

RNF01 - **Desempenho**

* O sistema deve ter um tempo de resposta de transação médio inferior a 2 segundos para todas as operações.
* A taxa de transferência mínima deve ser de 10 transações por segundo.

RNF02 – **Sistema Inoperante**

* Em caso de falha de conexão à Internet, o sistema deve permitir operações offline limitadas, como visualização de avisos e consulta de membros.
* Se houver falha no servidor, o sistema deve entrar em modo de leitura, permitindo apenas visualização de dados até que o servidor seja restaurado.

RNF03 – **Necessidade computacional**

* O sistema deve utilizar menos de 500 MB de memória RAM durante a operação normal.
* O espaço em disco necessário para armazenamento de dados e logs não deve exceder 1 GB.

RNF04 – **Treinamento**

* 2 horas para funcionários administrativos (pré-requisito: conhecimentos básicos de informática).
* Ajuda on-line: Será disponibilizado um tutorial básico com as principais funcionalidades do sistema.

**5.3 REGRAS DE NEGÓCIO**

Algumas regras do negócio iniciais também foram identificadas para o sistema. Essas regras são descritas a seguir:

RN01 - **Pré-requisito para acesso às funcionalidades do sistema**

* É necessário estar conectado ao sistema utilizando o usuário cadastro e senha.

RN02 - **Pré-requisito para agendamento de eventos**

* Para agendamento de eventos, o usuário deve ser identificado como membro ou administrador.

RN04 - **Disponibilidade de agendamento de eventos**

* Um agendamento de evento só poderá acontecer se houver horário disponível para a data selecionada.

**6. PRODUCT BACKLOG INICIAL**

O Product Backlog inicial consiste em uma lista de tarefas planejadas para o projeto, sujeita a alterações e evolução ao longo do tempo.

1. **Configuração do Ambiente de Desenvolvimento**

Instalação das ferramentas necessárias para o desenvolvimento do sistema.

1. **Análise de Requisitos**

Revisão detalhada dos requisitos funcionais e não funcionais do sistema.

1. **Estruturação da Arquitetura do Sistema**

Definição da arquitetura em camadas e estilo arquitetural REST.

1. **Implementação do Cadastro de Usuários**

Desenvolvimento do módulo de cadastro de membros da igreja.

1. **Desenvolvimento do Controle de Acesso**

Implementação das funcionalidades de controle de acesso para diferentes níveis de permissões.

1. **Criação do Módulo de Avisos**

Desenvolvimento do sistema de cadastro e visualização de avisos para os membros da igreja.

1. **Integração com Bancos de Dados**

Configuração e integração do sistema com o banco de dados para armazenamento seguro das informações.

1. **Implementação do Agendamento de Eventos**

Desenvolvimento da funcionalidade de agendamento de eventos online.

1. **Implementação da Segurança dos Dados**

Implementação de criptografia nos campos sensíveis para garantir a segurança dos dados dos usuários.

1. **Testes de Unidade e Integração**

Realização de testes para garantir o funcionamento correto das funcionalidades desenvolvidas.

**6.1. Divisão de Sprints ou Gantt**

As tarefas do projeto serão divididas em sprints para um melhor planejamento e acompanhamento do progresso. Abaixo está a divisão das tarefas em duas sprints iniciais:

**Sprint 1 (Semana 1-4)**

Configuração do Ambiente de Desenvolvimento

Análise de Requisitos

Estruturação da Arquitetura do Sistema

**Sprint 2 (Semana 4-8)**

Implementação do Cadastro de Usuários

Desenvolvimento do Controle de Acesso

Criação do Módulo de Avisos

Após a conclusão dessas duas sprints iniciais, novas tarefas serão adicionadas ao backlog de acordo com as necessidades e prioridades identificadas durante o desenvolvimento.

**6.2. Histórias de Usuário**

|  |
| --- |
| 1 Configuração do Ambiente de Desenvolvimento: Como desenvolvedor, eu gostaria de configurar o ambiente de desenvolvimento com as ferramentas necessárias para começar a trabalhar no sistema da igreja, para garantir um fluxo de trabalho eficiente desde o início do projeto. |
|  |
| 2 Análise de Requisitos: Como analista de requisitos, eu gostaria de realizar uma revisão detalhada dos requisitos funcionais e não funcionais do sistema da igreja, para garantir que todas as necessidades dos usuários e do negócio sejam atendidas de forma adequada. |
|  |
| 3 Estruturação da Arquitetura do Sistema: Como arquiteto de software, eu gostaria de definir a arquitetura em camadas e o estilo arquitetural REST para o sistema da igreja, para garantir uma estrutura sólida e escalável que atenda aos requisitos de segurança e desempenho. |
|  |
| 4 Implementação do Cadastro de Usuários: Como administrador, eu gostaria de cadastrar novos membros da igreja com informações detalhadas, como nome, endereço, CPF e informações de contato, para manter um registro atualizado e seguro dos membros. |
|  |
| 5 Desenvolvimento do Controle de Acesso: Como administrador, eu gostaria de implementar um sistema de controle de acesso com diferentes níveis de permissões para membros e administradores, para garantir a segurança e privacidade dos dados dos usuários. |
|  |
| 6 Criação do Módulo de Avisos: Como administrador, eu gostaria de cadastrar avisos de fácil visualização para os membros da igreja, permitindo a atualização e exclusão dos avisos quando necessário, para manter os membros informados sobre eventos e comunicados importantes. |
|  |
| 7 Integração com Bancos de Dados: Como desenvolvedor, eu gostaria de integrar o sistema com um banco de dados para armazenamento seguro e eficiente das informações dos usuários e dos avisos da igreja, garantindo a integridade e disponibilidade dos dados. |
|  |
| 8 Implementação do Agendamento de Eventos: Como membro da igreja, eu gostaria de agendar eventos online, como casamentos e batismos, para facilitar o processo de marcação de eventos e evitar deslocamentos desnecessários até a igreja. |
|  |
| 9 Implementação da Segurança dos Dados: Como desenvolvedor, eu gostaria de implementar criptografia nos campos sensíveis, como senhas, para garantir a segurança e proteção dos dados dos usuários contra acessos não autorizados. |
|  |
| 10 Testes de Unidade e Integração: Como testador, eu gostaria de realizar testes de unidade e integração para garantir o funcionamento correto das funcionalidades desenvolvidas no sistema da igreja, identificando e corrigindo possíveis erros ou falhas. |

**6.3. BDD**

1. **Cadastro de Novo Usuário:**

* **Dado** que o usuário está na página de cadastro.
* **Quando** o usuário preenche corretamente todos os campos obrigatórios.
* **Então** o sistema deve permitir o cadastro do novo usuário.
* **E** exibir uma mensagem de confirmação.

1. **Login de Usuário Registrado:**

* **Dado** que o usuário está na página de login.
* **Quando** o usuário insere suas credenciais corretas (e-mail e senha).
* **Então** o sistema deve permitir o acesso à área restrita.
* **E** redirecionar o usuário para a página principal.

1. **Cadastro de Novo Aviso:**

* **Dado** que o usuário está logado como administrador.
* **Quando** o usuário preenche o formulário de cadastro de aviso.
* **Então** o sistema deve permitir o cadastro do aviso.
* **E** exibir o aviso na página principal para os membros.

1. **Agendamento de Evento:**

* **Dado** que o usuário está logado como membro.
* **Quando** o usuário acessa a página de agendamento de eventos.
* **E** seleciona a data e horário desejados para o evento.
* **Então** o sistema deve confirmar o agendamento do evento.
* **E** exibir o evento na agenda do usuário.

1. **Controle de Acesso Administrativo:**

* **Dado** que o usuário está logado como administrador.
* **Quando** o administrador tenta acessar uma funcionalidade restrita.
* **Então** o sistema deve permitir o acesso sem restrições.
* **E** registrar a ação de acesso no log de atividades.

1. **Atualização de Informações Cadastrais:**

* **Dado** que o usuário está logado em sua conta.
* **Quando** o usuário acessa a página de atualização de informações.
* **E** realiza as alterações necessárias nos campos.
* **Então** o sistema deve atualizar as informações corretamente.
* **E** exibir uma mensagem de confirmação de atualização.

**7. ARQUITETURA DE SOFTWARE**

A arquitetura mais adequada para o sistema da igreja, considerando os requisitos apresentados, é a **Arquitetura em Camadas (Layered Architecture)**.

**Camada de Apresentação (Interface do Usuário):**

Responsável pela interação com o usuário. Inclui interfaces para cadastro, agendamento, listagem, controle de acesso e exibição de avisos. Contempla as funcionalidades visuais e de interação com o sistema.

**Camada de Aplicação:**

Gerencia a lógica de negócios e os processos do sistema. Responsável por validar os dados, controlar acesso, aprovações, exclusões e atualizações. Implementa as regras de negócio (RN01, RN02, RN04).

**Camada de Persistência de Dados:**

Armazena os dados de forma segura e eficiente. Implementa a criptografia para senhas (RF01). Garante integridade e disponibilidade dos dados.

**Camada de Serviços:**

Gerencia a comunicação entre as camadas e outros sistemas externos. Implementa operações offline limitadas (RNF02). Coordena a troca de informações com dispositivos clientes.

**7.1. Padrão de Projeto e/ou Estilo Arquitetural**

**Estilo Arquitetural:**

Dentro da Arquitetura em Camadas, um estilo arquitetural complementar que pode ser adotado é o Estilo Arquitetural REST (Representational State Transfer) para as comunicações entre as camadas e serviços externos.

**Descrição do Estilo Arquitetural REST:**

* Utiliza recursos (endpoints) para representar entidades e operações.
* Comunicação baseada em protocolo HTTP.
* Fácil integração com sistemas web e dispositivos móveis.
* Suporta as operações CRUD (Create, Read, Update, Delete) necessárias para o sistema da igreja.

Essa arquitetura em camadas, combinada com o estilo arquitetural REST, permite uma separação clara de responsabilidades, facilita a manutenção, escalabilidade e integração com outros sistemas, além de atender aos requisitos funcionais, não funcionais e regras de negócio identificados para o sistema da igreja.

**8.** [**Design**](#_heading=h.37m2jsg) **centrado no usuário**

O princípio da UX no projeto visa garantir que a experiência do usuário seja intuitiva, agradável e eficiente em todas as interações com o sistema. Isso inclui a criação de interfaces amigáveis, navegação intuitiva e design responsivo para diferentes dispositivos.

**8.1. Mapa de Empatia**

O mapa de empatia é uma ferramenta que ajuda a compreender melhor as necessidades, desejos, preocupações e motivações do usuário. Vamos criar um mapa de empatia para o projeto da igreja:

* **O que o usuário pensa e sente:** Os usuários desejam um sistema que seja fácil de usar e que os ajude a se conectar com a igreja de forma rápida e eficiente. Eles querem sentir-se seguros ao inserir informações pessoais e ao realizar transações online.
* **O que o usuário vê:** Eles veem a necessidade de um sistema que ofereça informações claras sobre eventos, avisos e serviços da igreja. Esperam uma interface limpa e organizada.
* **O que o usuário ouve:** Os usuários ouvem falar da importância da segurança dos dados online e da facilidade de uso de sistemas similares em outras instituições religiosas.
* **O que o usuário diz e faz:** Eles expressam o desejo de realizar agendamentos online, receber avisos importantes e acessar informações atualizadas sobre a igreja. Eles agem buscando um sistema conveniente e confiável.

**8.2.** [**PERSONAS / Protopersonas**](#_heading=h.1mrcu09)

1. **Ana, a Administradora da Igreja:**

* **Idade**: 45 anos
* **Profissão**: Administradora da Igreja
* **Necessidades**: Aprovar agendamentos, gerenciar usuários, atualizar informações da igreja. Comportamento: Acessa o sistema regularmente para realizar tarefas administrativas.

1. **João, o Membro Ativo da Congregação:**

* **Idade**: 30 anos
* **Profissão**: Advogado
* **Necessidades**: Agendar eventos, receber avisos, acessar informações sobre a igreja. Comportamento: Utiliza o sistema para se manter atualizado sobre eventos e participar das atividades da igreja.

1. **Maria, a Voluntária Engajada:**

* **Idade**: 60 anos
* **Profissão**: Aposentada
* **Necessidades**: Ajudar na organização de eventos, acessar informações sobre voluntariado. Comportamento: Interage com o sistema para colaborar com a igreja em suas atividades voluntárias.

Essas personas representam diferentes perfis de usuários e suas necessidades específicas ao utilizar o sistema da igreja. Isso ajuda a direcionar o design e as funcionalidades do sistema de acordo com as expectativas e comportamentos dos usuários reais.

**9. MODELO DE CASOS DE USO**

Para compreender melhor as necessidades da igreja e o funcionamento de sistemas semelhantes, foram realizados estudos de casos de outras instituições religiosas que utilizam sistemas de gestão. Isso permitiu identificar boas práticas e funcionalidades relevantes a serem incorporadas ao projeto.

**9.1 IDENTIFICAÇÃO DOS ATORES E SUAS RESPONSABILIDADES**

**Membro:**

* Responsabilidades:
* Efetuar o cadastro com suas informações pessoais.
* Solicitar agendamentos de eventos da igreja.
* Acessar informações gerais disponibilizadas pela igreja.

**Membro Administrativo:**

* Responsabilidades:
* Gerenciar as informações dos membros da igreja, incluindo aprovação, atualização e exclusão de registros.
* Gerenciar os agendamentos de eventos da igreja, incluindo adição, atualização e aprovação.
* Gerenciar os avisos da igreja, incluindo adição, atualização e remoção.

**Sistema:**

* Responsabilidades:
* Armazenar e validar as informações dos membros da igreja.
* Armazenar e validar os agendamentos de eventos da igreja.
* Armazenar e validar os avisos da igreja.
* Exibir informações relevantes para os membros.
* Permitir a interação entre membros e membros administrativos.

**9.2 DEFINIÇÃO DE PRIORIDADE DE DESENVOLVIMENTO DOS CASOS DE USO**

Abaixo estão informados os casos de uso identificados e a prioridades para o desenvolvimento de cada um deles:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ***Número*** | ***Nome do caso de uso*** | ***Prioridade*** | ***Justificativa*** | ***Aluno Responsável*** |
| *UC001* | *Manter Membro* | *Alta* | *Gerenciar informações dos membros* | *Enrico, Gabriel* |
| *UC002* | *Manter Agendamento* | *Alta* | *Gerenciar agendamentos* | *Enrico, Gabriel* |
| *UC003* | *Manter Avisos* | *Alta* | *Gerenciar Avisos* | *Italo, Kaio* |

**9.3 Descrição Detalhada dos Casos de Uso:**

1. **Especificação do Caso de Uso: UC001 – Manter Membro**

# Manter membro

## Breve descrição

Este caso de uso permite aos membros administrativos gerenciar as informações dos membros da igreja, incluindo aprovação, atualização e exclusão de registros.

# Atores principais

## Membro

O membro efetua o cadastro com suas informações pessoais.

## Membro Administrativo

Interage com o sistema para aprovar, atualizar ou excluir informações dos membros da igreja.

# Pré-condições

## O membro administrativo está autenticado no sistema.

# Pós-condição

As informações dos membros estão atualizadas ou membros foram removidos do sistema.

# Fluxo de Eventos

## Fluxo Básico

* O membro efetua seu cadastro com suas informações
* O membro administrative aprova as informações do cadastro.
* O sistema valida e armazena as informações no Banco de Dados
* O administrador atualiza informações existentes se necessário.
* O sistema valida e atualiza as informações.
* O administrador remove membros da igreja, confirma a exclusão e o sistema remove o membro.
* Caso de uso encerrado.

## Fluxos Alternativos

### Dados de cadastro incompletos.

Se o membro deixar campos obrigatórios em branco, o sistema exibe uma mensagem de erro e solicita que esses campos sejam preenchidos. O caso de uso retorna ao passo do formulário para correção

### Cancelar o cadastro

Em qualquer momento antes de confirmar o cadastro, o membro pode optar por cancelar a operação.

O sistema interrompe o processo de cadastro e retorna ao menu principal.

1. Especificação do Caso de Uso: UC002 – Manter Agendamento

# Manter membro

## Breve descrição

Este caso de uso permite aos membros administrativos gerenciar os agendamentos de eventos da igreja, incluindo, atualização, e aprovação de agendamentos feitos pelos membros.

# Atores principais

## Membro

O membro efetua a solicitação.

## Membro Administrativo

Interage com o sistema para adicionar, atualizar ou aprovar agendamentos feitos pelos membros.

# Pré-condições

## O membro administrativo está autenticado no sistema.

# Pós-condição

Horários solicitados deverão ficar com status de indisponivel.

# Fluxo de Eventos

## Fluxo Básico

* O membro solicita o agendamento do evento.
* O administrador acessa a funcionalidade de gerenciamento de agendamentos.
* O administrador visualiza os agendamentos pendentes.
* O sistema exibe os detalhes dos agendamentos.
* O administrador aprova ou rejeita os agendamentos.
* Caso de uso encerrado.

## Fluxos Alternativos

### Dados de solicitação incompletos.

Se o membro deixar campos obrigatórios em branco, o sistema exibe uma mensagem de erro e solicita que esses campos sejam preenchidos. O caso de uso retorna ao passo do formulário para correção

### Cancelar solicitação

Em qualquer momento antes de confirmar a solicitação, o membro pode optar por cancelar a operação. O sistema interrompe o processo de cadastro e retorna ao menu principal.

1. Especificação do Caso de Uso: UC003 – Manter Avisos

# Manter avisos

## Breve descrição

Este caso de uso permite aos membros administrativos gerenciar os avisos da igreja, incluindo adição, atualização e remoção de avisos.

# Atores principais

## Membro Administrativo

Interage com o sistema para adicionar, atualizar ou excluir avisos da igreja.

# Pré-condições

## O membro administrativo está autenticado no sistema.

# Pós-condição

Os avisos devem ser atualizados para visualização imediatamente.

# Fluxo de Eventos

## Fluxo Básico

* O administrador acessa a funcionalidade de gerenciamento de avisos.
* O administrador cadastra novos avisos com título e conteúdo.
* O sistema valida e armazena os avisos.
* O administrador atualiza títulos ou conteúdos dos avisos existentes.
* O sistema valida e atualiza os avisos.
* O administrador remove avisos, confirma a exclusão e o sistema remove o aviso.
* Caso de uso encerrado.

## Fluxos Alternativos

### Dados de avisos incompletos.

Se o administrador deixar campos obrigatórios em branco, o sistema exibe uma mensagem de erro e solicita que esses campos sejam preenchidos. O caso de uso retorna ao passo do formulário para correção

### Cancelar o cadastro

Em qualquer momento antes de confirmar o cadastro, o administrador pode optar por cancelar a operação.

O sistema interrompe o processo de cadastro e retorna ao menu principal.

**Projetos Futuros:**

Como projetos futuros, pretende-se implementar funcionalidades adicionais ao sistema, como integração com sistemas de pagamento online para doações e eventos, criação de um aplicativo móvel para facilitar o acesso dos membros e melhorias na interface de usuário.

**Conclusão:**

O desenvolvimento do sistema de gestão para a igreja demonstrou ser uma solução eficaz para otimizar processos administrativos e melhorar a comunicação interna. O sistema contribui significativamente para a organização e eficiência das atividades da igreja, proporcionando uma experiência mais positiva para membros e líderes religiosos.

**Referencias Bibliograficas:**

"The Impact of Technology on Church Administration" - Disponível em bases de dados acadêmicas como JSTOR, PubMed, Scopus, etc.

"Digital Church: Social Media and Technology in Religious Contexts" - Disponível em periódicos de religião e tecnologia.

Lifeway.com - Oferece recursos e artigos sobre gestão de igrejas.

Churchleaders.com - Portal com informações e dicas sobre liderança e administração eclesiástica. E-books e artigos disponíveis em plataformas como Google Scholar, ResearchGate e Academia.edu.