

1. OBJETIVO

O objetivo deste documento é descrever informações essenciais para o projeto, como as ferramentas utilizadas, seus requisitos e suas principais funções.

2. VISÃO GERAL DO CONTEXTO

O controle financeiro é imprescindível para qualquer organização. Tal controle permite que as instituições cumpram com suas responsabilidades financeiras e consigam desenvolver-se no âmbito em que estão inseridas. Por outro lado, percebe-se que algumas instituições não possuem um manejo orçamental adequado, como é o caso de uma vasta quantidade de instituições religiosas atuais, que utilizam planilhas digitais e impressas no seu controle de finanças, configurando um modelo ineficiente, com baixo nível de organização e precisão de dados e com alta demanda de tempo.

3. MAPEAMENTO DOS PROBLEMAS

ID	PROBLEMA	DETALHAMENTO
PR01	Alta demanda de tempo	A contabilização das entradas e saídas, bem como o cálculo do saldo obtido gera uma demanda de tempo aos tesoureiros locais Já no templo sede, o tempo gasto efetua-se majoritariamente na contabilização dos valores de todos os templos locais.
PR02	Dados imprecisos	A utilização de tabelas, ocasiona a visualização dos valores em listas, sem maior detalhamento ou organização. Tal modelo não permite o acompanhamento de métricas por tempo ou igrejas (dentre outras formas) de forma eficiente.
PR03	Baixo nível de organização	O agrupamento de valores em tabelas digitais ou impressas, sem um maior controle, torna este modelo mais suscetível à erros e perdas de dados por possuírem um modelo de armazenamento mais simples.

4. VISÃO GERAL DA SOLUÇÃO PROPOSTA

Com objetivo de resolver os problemas apresentados acima, pretende-se desenvolver um sistema de gestão para instituições religiosas, com foco na unificação, praticidade e clareza na área financeira, tornando-a prática, eficaz e veloz, permitindo maior organização e precisão dos dados. O sistema terá controle de entradas e saídas,
Bruno Reis, Maicon Robert, Paulo César, Victor Fonteles - 3BII

estatísticas financeiras e balanços por diferentes períodos de tempo e funcionalidades, além disso, será dividido entre igrejas e sede. Nas igrejas, o sistema abrangerá o controle de dízimos e ofertas, saídas e outros gastos pertinentes. Já na sede, lidará com o controle geral, separado por mês, trimestre e ano, auxiliando o tesoureiro no monitoramento de todas as igrejas.

5. REQUISITOS FUNCIONAIS

ID	DESCRIÇÃO DO REQUISITO	COMPLEXIDADE	CRITICIDADE	DEPENDÊNCIA
USUÁRIO				
RF001	O sistema deve permitir que os USUÁRIOS <<solicitem cadastro>> por meio de NOME, EMAIL, CPF, SENHA, TELEFONE e IGREJA em que exercem suas atividades. Após preencher o formulário de cadastro, o mesmo é enviado ao TESOUREIRO SEDE para que ele possa avaliar e autorizar ou negar o cadastro.	Média	Alta	
RF002	O sistema deve enviar uma notificação para o USUÁRIO assim que o TESOUREIRO SEDE tomar uma decisão sobre o cadastro, informando se o mesmo foi aprovado ou negado.	Média	Média	RF001, RF020
RF003	O sistema deve permitir que os usuários alterem suas informações de cadastro a qualquer momento, desde que essas alterações sejam novamente autorizadas pelo TESOUREIRO SEDE	Média	Alta	RF001
RF004	O sistema deve exigir que os USUÁRIOS se autenticuem por meio de <<login>> utilizando o EMAIL e a SENHA cadastrados.	Média	Alta	RF001, RF020

RF005	O sistema deve permitir que os usuários façam <<logout>> a qualquer momento, desconectando suas contas do sistema.	Baixa	Alta	RF004
TESOUREIRO				
RF006	O sistema deve permitir ao tesoureiro realizar o <<controle mensal de dízimos>> da igreja que gerencia, com DATA, CONTRIBUINTE, VALOR e a VALIDAÇÃO TESOUREIRO , e também gerar um relatório.	Média	Alta	RF004
RF007	O sistema deve fornecer o histórico de relação mensal dos dizimistas da igreja, incluindo o nome, o valor e a data de cada dízimo.	Média	Alta	RF004, RF006
RF008	O sistema deve permitir ao tesoureiro realizar o <<controle mensal de saídas>> da igreja que gerencia, separadas por categoria e com DATA, DESCRIÇÃO, VALOR e VALIDAÇÃO DO TESOUREIRO .	Média	Alta	RF004
RF009	O sistema deve permitir ao tesoureiro <<gerar um relatório>> de todas as saídas, que contenha o somatório das mesmas e as despesas fixas mensais calculadas.	Alta	Alta	RF004, RF008
RF010	O sistema deve permitir que o tesoureiro local <<gerencie as entradas por culto>> da igreja ao gerar um relatório, permitindo a <i>adição, edição e exclusão</i> . As entradas devem conter informações sobre a DATA, VALIDAÇÃO DO TESOUREIRO, TIPO DE ENTRADA (DÍZIMO OU OFERTA) E TIPO DE CULTO .	Média	Alta	RF004

RF011	O sistema deve permitir ao tesoureiro <i>editar</i> e <i>excluir</i> saídas e entradas que constam em seus respectivos históricos, ou então adicioná-las.	Média	Alta	RF004, RF008, RF010
RF012	O sistema deve permitir que o tesoureiro local visualize e altere as informações da igreja a qual está vinculado, sendo NOME, LOCALIZAÇÃO, DIRIGENTE e TESOUREIRO , desde que essas alterações sejam autorizadas pelo TESOUREIRO SEDE e PASTOR	Baixa	Alta	RF004
RF013	O sistema deve permitir que o tesoureiro local gerencie os membros da sua igreja de forma eficiente, como <i>adicionar</i> novos membros, <i>editar</i> informações de membros existentes, <i>excluir</i> membros, visualizar a lista de membros e filtrá-los por diferentes critérios e gerar relatórios.	Baixa	Alta	RF004
RF014	O sistema deve permitir detalhar cada membro, para ser possível visualizar seus dados e histórico de dízimos.	Média	Média	RF004, RF013
RF015	O sistema deve permitir que o tesoureiro local disponibilize o relatório geral mensal de sua igreja até o último dia útil do mês. O relatório deve incluir informações detalhadas sobre as entradas e saídas da igreja durante o mês, incluindo ofertas e contribuições, despesas e outras transações financeiras, e também o saldo mensal da igreja.	Baixa	Alta	RF004, RF006, RF008, RF010
RF016	Antes de disponibilizar o relatório, o sistema deve permitir que o tesoureiro local revise o mesmo para garantir que está correto e completo e obter a validação do pastor responsável pelo ministério financeiro da igreja.	Baixa	Alta	RF004, RF006, RF008, RF010, RF010, RF004, RF006, RF008, RF015

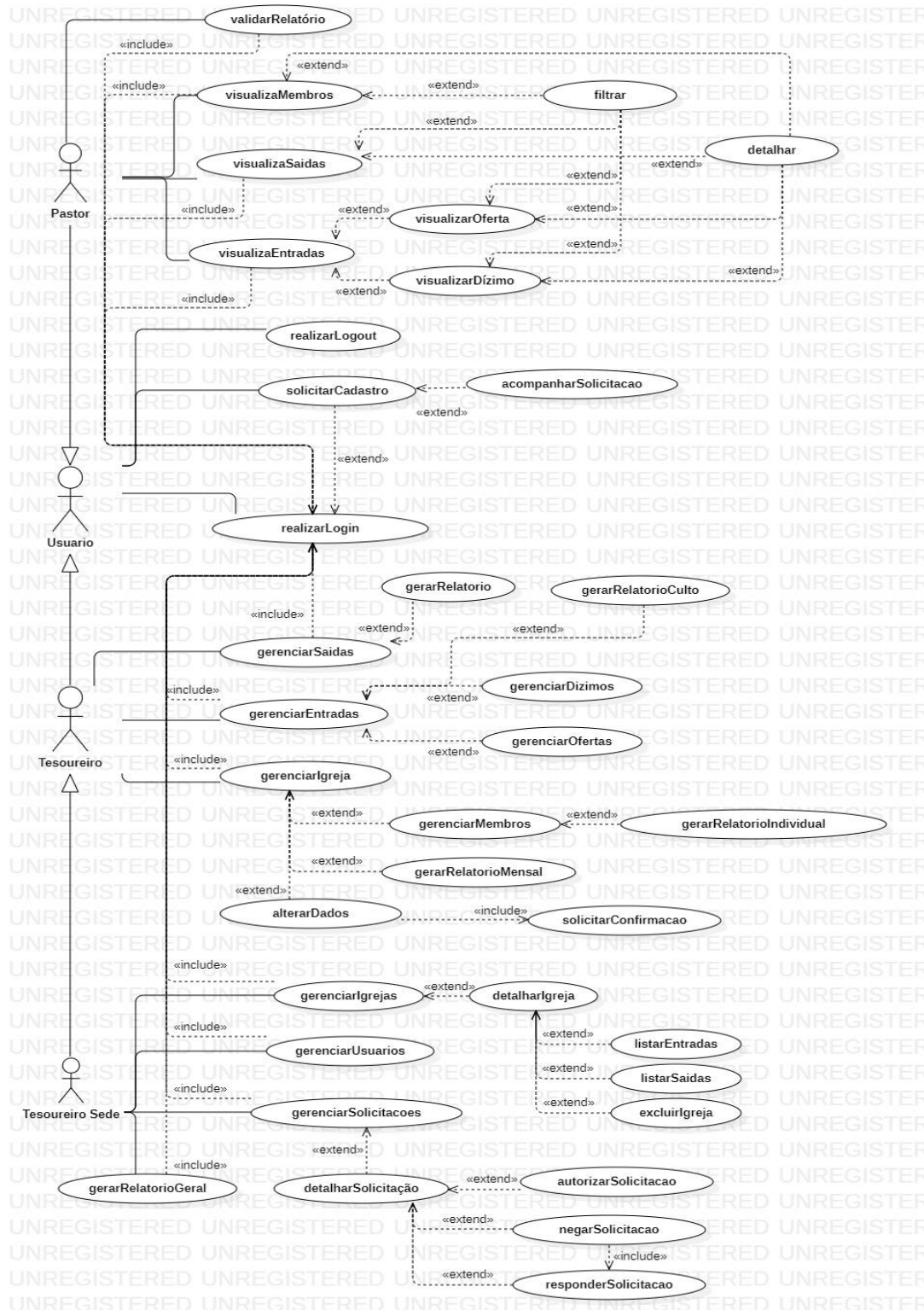
RF017	O sistema deve permitir que o tesoureiro visualize todos os tipos de relatórios gerados, e filtrá-los por tipo, por período de tempo, e no caso do relatório de ofertas, filtrá-los por dia de culto.	Média	Alta	RF004, RF006, RF008, RF010, RF004, RF006, RF008, RF013, RF015
RF018	O sistema deve permitir que o tesoureiro local filtre o histórico de relação de dízimos por nome de dizimista ou por valor de dízimo.	Baixa	Média	RF004, RF006
TESOUREIRO SEDE				
RF019	O sistema deve permitir o cadastro de igrejas, inserindo NOME, LOCALIZAÇÃO, DIRIGENTE e TESOUREIRO . As igrejas possuem também ENTRADAS, SAÍDAS, SALDO E MEMBROS e esses dados são manipulados pelos tesoureiros.	Baixa	Alta	RF004
RF020	O sistema deve permitir que o TESOUREIRO SEDE autorize o cadastro de novos USUÁRIOS , por meio de uma tela de gerenciamento de formulários de cadastro, onde pode visualizar, editar e revogar o acesso de USUÁRIOS já cadastrados.	Baixa	Média	RF004
RF021	O sistema deve permitir que o Tesoureiro sede visualize e filtre os membros de cada igreja cadastrada.	Baixa	Média	RF004, RF019
RF022	O sistema deve permitir que o Tesoureiro sede visualize os relatórios mensais das igrejas, para ter acesso às demais informações financeiras das igrejas cadastradas.	Baixa	Baixa	RF004, RF015, RF016
RF023	O sistema deve fornecer ao TESOUREIRO SEDE um relatório geral com o somatório de	Baixa	Alta	RF004, RF006, RF008 RF010, RF015, RF016

	todas ENTRADAS (DA SEDE, IGREJAS E MISSÕES P/ SEDE) e SAÍDAS (DA SEDE, IGREJAS, PGTO OBREIROS, INSS, ALUGUEL DE OBREIROS, REPAROS/REFORMAS/CONST, ASSISTÊNCIA SOCIAL) de todas as igrejas e fornecer o SALDO FINANCEIRO.			
PASTOR				
RF024	O sistema deve permitir que os pastores validem os relatórios mensais gerados pelos tesoureiros.	Média	Alta	RF004,RF005
RF025	O sistema deve permitir aos pastores a visualização de informações da igreja a qual está vinculado, incluindo ofertas, dízimos, saídas e membros, usando filtros de pesquisa, de forma detalhada para cada item.	Baixa	Média	RF004
RF026	O sistema deve permitir que o pastor visualize e altere as informações da igreja a qual está vinculado, sendo NOME, LOCALIZAÇÃO, DIRIGENTE e TESOUREIRO , desde que essas alterações sejam autorizadas pelo TESOUREIRO SEDE.	Baixa	Média	RF002
GERAIS				
RF027	O sistema deve permitir o download e compartilhamento dos relatórios em formato PDF. Além disso, deve permitir sua impressão, para fins de arquivamento.	Média	Baixa	RF004, RF007, RF009, RF010, RF015
RF028	O sistema deve calcular o valor do FUNDO CONVENCIONAL (5%), OBREIRO(10%), MISSÕES P/ SEDE(10%) com base no total arrecadado pela igreja adicionando este valor às saídas	Alta	Alta	RF004, RF006, RF010

6. REQUISITOS NÃO FUNCIONAIS

REQUISITOS NÃO FUNCIONAIS			
ID	Descrição do Requisito	Complexidade	Criticidade
RNF001	Tanto funcionários com habilidades com computador quanto funcionários com dificuldade, usarão o sistema. Portanto a interface deve ser clara e objetiva.	Baixa	Alta
RNF002	Ter mecanismos de autenticação e controle de acesso dos usuários, proteger dados sensíveis e ser seguro.	Média	Alta
RNF003	O sistema deve ficar disponível para uso a qualquer momento. De preferência das 06h00 às 23:00	Média	Média
RNF004	Considerando que o sistema deve ser ágil e de fácil acesso, o sistema deve ser responsivo, permitindo o uso em diversos dispositivos, principalmente dispositivos móveis.	Média	Média
RNF005	Cada tipo de usuário deve ter acesso apenas às funções designadas para ele.	Média	Alta
RNF006	O sistema deve apresentar mensagem de erro, quando for o caso, de forma clara e objetiva.	Média	Baixa

7. DIAGRAMA DE CASO DE USO

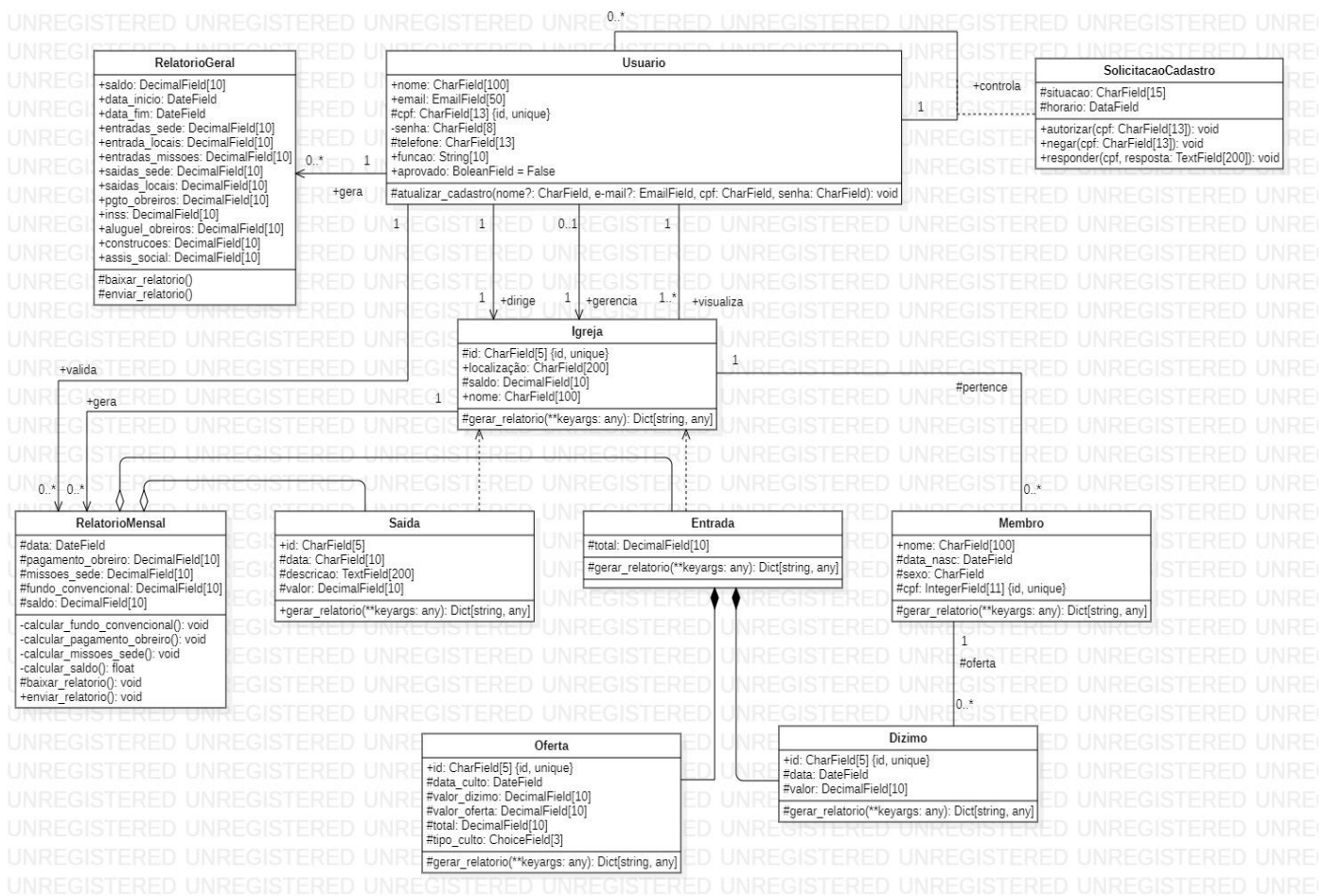


8. DESCRIÇÃO TEXTUAL DOS CASOS DE USO

Disponível em:

<<https://docs.google.com/document/d/1q7cJXL5S49G4lmYa3HJNBh5yJdcw1mD9aNI6Fs6OdMI/edit?usp=sharing>>

9. DIAGRAMA DE CLASSE



10. DEFINIÇÃO DA ARQUITETURA DA SOLUÇÃO

ARQUITETURA DE SOFTWARE

FRONT-END

- *HTML5*

HTML é uma abreviação de Hypertext Markup Language, ou seja, Linguagem de Marcação de Hipertexto. Em suma, o HTML é uma linguagem usada para a publicação de conteúdo (texto, imagens, vídeos, áudio etc.) na web. O HTML é baseado no conceito de hipertexto, que são conjuntos de elementos ligados por conexões, que podem ser palavras, imagens, vídeos, áudio, documentos etc. que quando conectados, formam uma grande rede de informação. A conexão feita em um hipertexto é algo imprevisível que permite a comunicação de dados, organizando conhecimentos e guardando informações relacionadas.

- *CSS3*

O CSS foi desenvolvido para ter tags que ajudam na formatação de sites, uma vez que o HTML foi projetado apenas para escrever a marcação para o site. CSS é a sigla para o termo em inglês Cascading Style Sheets que, traduzido para o português, significa Folha de Estilo em Cascatas. O CSS é utilizado com as linguagens de marcação HTML ou XHTML, servindo para estilizar elementos escritos em uma linguagem de marcação (como o HTML). Essa linguagem separa o conteúdo da representação visual do site.

- *JavaScript*

O JavaScript é uma linguagem de programação desenvolvida em 1995 por Brendan Eich, primeiramente a patente NetScape. Hoje é uma das 10 linguagens de programação mais usadas no mercado, sendo uma linguagem interpretada e estruturada, de script de alto nível e fracamente tipada, sendo muito usada no desenvolvimento WEB junto do HTML e CSS, mas também pode ser usada na parte do servidor com o Node.JS.

- *Bootstrap 5*

O Bootstrap é um dos frameworks mais populares e amplamente utilizados para criar layouts e interfaces de usuário responsivas e atraentes para aplicativos web, dashboards e sites. Ele foi criado pela equipe do Twitter e se destaca por sua facilidade de uso e sua grande variedade de componentes e estilos predefinidos, o que permite aos desenvolvedores criar rapidamente interfaces de usuário elegantes e profissionais sem precisar escrever muito código do zero. Além disso, o Bootstrap é altamente

personalizável, permitindo que os desenvolvedores adaptem facilmente sua aparência e funcionalidade para atender às necessidades específicas de seus projetos.

BACK-END

- Python

Python é uma linguagem de programação criada em 1991 na Holanda por Guido von Rossum. Python foi desenvolvida na linguagem ABC, possui uma sintaxe inspirada em C. Uma linguagem de alto nível, interpretada de script, imperativa, orientada a objetos, funcional, de tipagem dinâmica e forte sendo a linguagem mais popular do ano de 2020.

- Django

O Django é um framework para o desenvolvimento de aplicativos web que adota o padrão model-template-view (MTV) como arquitetura. Com o MTV, a aplicação é dividida em três componentes principais: Model, Template e View, e o framework incentiva o uso do princípio DRY (Don't Repeat Yourself) para reutilizar código existente e evitar repetições desnecessárias. Com essa abordagem, o Django permite que os desenvolvedores criem aplicativos web com mais eficiência e facilidade, além de fornecer uma estrutura sólida e escalável para o desenvolvimento de aplicativos complexos e de alta qualidade.

BANCO DE DADOS:

O Django possui suporte nativo para diversos bancos de dados, incluindo SQLite, MySQL e PostgreSQL. Isso permite aos desenvolvedores escolherem o banco de dados mais adequado para cada aplicação, além de facilitar a integração e o gerenciamento de dados. Além disso, o Django fornece ferramentas para realizar migrações de banco de dados de forma segura e eficiente, evitando erros e perda de dados. A ORM do Django também permite a realização de consultas complexas e a manipulação de dados de forma intuitiva e eficiente.

.

CONTROLE DE VERSÃO:

- Git Hub

O controle de versão é um processo fundamental para qualquer projeto de software, pois permite rastrear as alterações realizadas no código ao longo do tempo e garantir que o projeto esteja sempre atualizado. O projeto utilizará a plataforma Github como ferramenta de versionamento, permitindo que

todos os envolvidos no projeto possam acessar e contribuir com o código de forma colaborativa. Além disso, o Github permite a criação de branches (ramificações) do projeto, permitindo que as alterações possam ser testadas antes de serem integradas à versão principal do código. Isso garante uma maior segurança e estabilidade do projeto, além de facilitar a manutenção e a correção de eventuais bugs.

11. ENVOLVIDOS

Função/Papel	Descrição
Tesoureiro local	Responsável por gerir as finanças localmente, ou seja, em sua igreja. Além disso, realiza a prestação de contas mensalmente ao templo sede
Pastor	Responsável por dirigir a igreja. Autoriza a prestação de contas.
Tesoureiro sede	Atua como tesoureiro local no templo sede e realiza um controle financeiro geral de todas as igrejas
Usuário	Classe generalizadora de qualquer envolvido no sistema que assuma as funções descritas anteriormente.

12. GLOSSÁRIO

Termo	Descrição
Dashboards	Interface gráfica que apresenta informações em tempo real de forma visual e interativa.
Dízimos	Valor doado mensalmente pelos membros, correspondente a 10% dos seus salários.
Layouts	Organização visual de uma página ou aplicativo, incluindo disposição de elementos, cores e tipografia.
Oferta	Contribuição financeira voluntária realizada pelos membros em cultos ou eventos religiosos.

Termo	Descrição
Scripts	Um script é uma série de instruções escritas em código fonte para a execução de uma tarefa específica em um software. É interpretado pelo computador e pode ser modificado ao longo do tempo.
Framework	Um framework é uma estrutura base previamente definida para desenvolvimento de software que acelera o processo e facilita a personalização.
Branches	Branches são versões paralelas do projeto no controle de versão de software.
Tipagem dinâmica	Sistema de tipagem de dados em que o tipo é verificado em tempo de execução, em oposição à tipagem estática, em que o tipo é verificado em tempo de compilação.