



**INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DO
TOCANTINS**

**CAMPUS PORTO NACIONAL
LICENCIATURA EM COMPUTAÇÃO**

NILSON ALVES DOS SANTOS

**DESENVOLVIMENTO E VALIDAÇÃO DE SOFTWARE DE AUXÍLIO À GESTÃO
DE INSTITUIÇÕES EVANGÉLICAS: Uma prática a luz da Igreja Salvação em
Cristo de Porto Nacional - TO**

PORTO NACIONAL

2018

NILSON ALVES DOS SANTOS

**DESENVOLVIMENTO E VALIDAÇÃO DE SOFTWARE DE AUXÍLIO À GESTÃO
DE INSTITUIÇÕES EVANGÉLICAS: Uma prática a luz da Igreja Salvação em
Cristo de Porto Nacional - TO**

Monografia apresentada a Coordenação do Curso Superior de Licenciatura em Computação do *Campus* Porto Nacional, do Instituto Federal do Tocantins, como exigência à obtenção do grau de Licenciado em Computação.

Orientador: Prof. Me. Rafael Miranda Correia

PORTO NACIONAL

2018

Santos, Nilson Alves.

DESENVOLVIMENTO E VALIDAÇÃO DE SOFTWARE DE AUXÍLIO A GESTÃO DE INSTITUIÇÕES EVANGÉLICAS: Uma prática a luz da Igreja Salvação em Cristo de Porto Nacional - TO. / Nilson Alves dos Santos. – Porto Nacional: Instituto Federal de Educação, Ciência e Tecnologia, 2018. 140 f.

Trabalho de Conclusão de Curso (Graduação) – Instituto Federal de Educação, Ciência e Tecnologia, Curso de Licenciatura em Computação, Porto Nacional, TO, 2018.

Orientador: Prof. Me. Rafael Miranda Correia.

1. Desenvolvimento Web. 2. Educação. 3. Gestão. 4. Igreja. 2. Software. I. DESENVOLVIMENTO E VALIDAÇÃO DE SOFTWARE DE AUXÍLIO A GESTÃO DE INSTITUIÇÕES EVANGÉLICAS: Uma prática a luz da Igreja Salvação em Cristo de Porto Nacional - TO.

NILSON ALVES DOS SANTOS

**DESENVOLVIMENTO E VALIDAÇÃO DE SOFTWARE DE AUXÍLIO A GESTÃO
DE INSTITUIÇÕES EVANGÉLICAS: Uma prática a luz da Igreja Salvação em
Cristo de Porto Nacional - TO**

Monografia apresentada a Coordenação do Curso Superior de Licenciatura em Computação do *Campus* Porto Nacional do Instituto Federal do Tocantins, como exigência à obtenção do grau de Licenciado em Computação.

Orientador: Prof. Me. Rafael Miranda Correia

Aprovado em 17/12/2018.

BANCA EXAMINADORA

Professor Orientador Me. Rafael Miranda Correia
Instituto Federal de Educação, Ciência e Tecnologia do Tocantins

Professora Convidada Me. Shirley Alves Viana Vanderlei
Instituto Federal de Educação, Ciência e Tecnologia do Tocantins

Professor Convidado Tiago Soares dos Reis
Instituto Federal de Educação, Ciência e Tecnologia do Tocantins

CONCEITO FINAL: 10,0

À minha família: meus pais Deuselio Alves dos santos e Ivanildes Alves dos Santos, meus irmãos e irmãs, minha esposa: Ihanka Jorrane Pereira e filhos: Gleison Alves Borges, Beatriz Cerqueira Alves, Jaqueline Alves da Silva e Mariana Pereira Alves.

AGRADECIMENTOS

A Deus por ter me dado saúde e força para superar as dificuldades.

Ao Instituto Federal do Tocantins, seu corpo docente, direção e administração que oportunizaram a janela do vislumbro, um horizonte superior eivado pela acendrada confiança no mérito e ética presente.

Ao meu orientador, Prof. Me. Rafael Miranda Correia, pelo suporte no tempo que lhe coube, pelas suas correções e incentivos que culminaram na conclusão deste projeto.

Aos meus familiares, pelo amor, incentivo e apoio incondicional.

E a todos que, direta ou indiretamente, fizeram parte da minha formação, o meu muito obrigado.

“Meus filhos terão computadores, sim, mas antes terão livros. Sem livros, sem leitura, os nossos filhos serão incapazes de escrever - inclusive a sua própria história.”

(Bill Gates)

RESUMO

O presente trabalho aborda o desenvolvimento e validação de *software* de auxílio à gestão de instituições evangélicas: Uma prática a luz da Igreja Salvação em Cristo de Porto Nacional - TO. Objetivando utilizá-lo como ferramenta de auxílio da gestão e desfrutar dos benefícios tecnológicos que aprimorará a mesma. Desta forma, a problemática foi evidenciada pela falta de praticidade na gestão percebida pelos líderes da Igreja. Assim, levantou-se a hipótese de que o desenvolvimento do *software* auxiliará na gestão e garantirá os benefícios. Todas as etapas aqui descritas sobre o desenvolvimento do *software* são frutos de um trabalho minucioso apresentado em formato de monografia. Nesse sentido, os resultados da pesquisa que avaliaram a qualidade do *software* desenvolvido, bem como os benefícios da aplicação pós-testes, frutificaram resultados satisfatórios, medidos através do grau de sua aceitabilidade. Com tudo, o presente estudo é visado na construção de um *software* e, posteriormente, na apreciação por líderes e membros selecionados através de critérios técnicos e científicos sobre o assunto pesquisado. Desta forma, o processo de desenvolvimento foi dividido em 3 (três) etapas que foram realizadas criteriosamente: a primeira, definiu os requisitos essenciais para a construção do *Software* e a preparação do ambiente de desenvolvimento; a segunda, foi o desenvolvimento e implantação e a terceira, foi a pesquisa com o intuito de avaliar a aceitabilidade do *software* implantado. Para a realização deste estudo foram utilizadas pesquisas bibliográficas através de livros, artigos científicos e sites para obtenção de informação e a pesquisa experimental utilizada para coletar os dados com a avaliação do referido *software*. O método de pesquisa utilizado empregou-se uma pesquisa qualitativa com enfoque quantitativo. A pesquisa foi realizada com o universo de 72 participantes, sendo 1 pastor presidente, 2 pastores locais, 1 secretário, 1 tesoureiro 37 membros, 16 congregados e 14 classificados como outros. Como instrumento de coleta dos dados, foi utilizado um questionário aplicado via *Web*. Os resultados obtidos mostram que o sistema desenvolvido atende as necessidades e melhora a gestão da instituição religiosa. 99% dos pesquisados responderam que indicariam o sistema para implantação na instituição religiosa, mostrando com isso a relevância deste trabalho.

PALAVRAS CHAVE: Desenvolvimento Web. Educação. Gestão. Igreja. Software.

ABSTRACT

The present work deals with the development and validation of software to aid the management of evangelical institutions: A practice in the light of the Salvation in Christ Church in Porto Nacional - TO. Aiming to use it as a management aid tool and enjoy the technological benefits that will improve it. In this way, the problem was evidenced by the lack of practicality in the management perceived by the leaders of the Church. Thus, the hypothesis was raised that the development of the software will aid in the management and will guarantee the benefits. All the steps described here about the development of the software are fruit of a detailed work presented in monograph format. In this sense, the results of the research that evaluated the quality of the developed software, as well as the benefits of the posttest application, yielded satisfactory results, measured by the degree of its acceptability. However, the present study is aimed at the construction of software and, later, on the appreciation of selected leaders and members through technical and scientific criteria on the subject researched. In this way, the development process was divided in 3 (three) steps that were carefully performed: the first, defined the essential requirements for the construction of the Software and the preparation of the development environment; the second one was the development and deployment, and the third was the research with the purpose of evaluating the acceptability of the implanted software. For the accomplishment of this study were used bibliographical research through books, scientific articles and sites to obtain information and the experimental research used to collect the data with the evaluation of said software. The research method used was qualitative research with a quantitative approach. The research was carried out with a total of 72 participants, including 1 pastor, 2 local pastors, 1 secretary, 1 treasurer 37 members, 16 congregated and 14 classified as others. As a data collection instrument, a web-based questionnaire was used. The results show that the developed system meets the needs and improves the management of the religious institution. 99% of the respondents answered that they would indicate the system for implantation in the religious institution, thus showing the relevance of this work.

Keywords: Web development. Education. Management. Churches. Software.

LISTA DE FIGURAS

Figura 1- Arquitetura de aplicações Web usando o padrão MVC.....	32
Figura 2 - Diagrama de caso de uso	49
Figura 3 - Diagrama de Classe.....	50
Figura 4 - Diagrama de sequência, representando o login.....	51
Figura 5 - Diagrama de Sequência representando cadastro de campo	52
Figura 6 - Diagrama de sequência de edição de dados do campus.....	53
Figura 7 - Diagrama de sequência de deleção de dados do campus.....	54
Figura 8 - Diagrama de sequência de lista de dados do campus	55
Figura 9 - Diagrama banco de dados	56
Figura 10 - Estrutura da Organização	58
Figura 11 - Classe de Conexão com o Banco de Dados.....	59
Figura 12 - Método estático de conexão com o banco de dados	60
Figura 13 - Inclusão do Banco na Classe	60
Figura 14 - Método da Classe	61
Figura 15 - Portal de Navegação nos Eventos	62
Figura 16 - Tela de cursos.....	62
Figura 17 - Tela de detalhes de curso	63
Figura 18 - Tela de login e cadastro	64
Figura 19 - Tela de área do aluno	64
Figura 20 - Tela de estudo e prova.....	65
Figura 21 - Tela de resultados.....	66
Figura 22 - Tela dados do certificado	66
Figura 23 - Tela de certificado	67
Figura 24 - Tela de login SGE	68
Figura 25 - Tela painel principal	68
Figura 26 - Tela painel secretaria.....	69
Figura 27 - Tela painel campus	69
Figura 28 - Tela cadastro de campus e tela de relatório	70
Figura 29 - Tela igreja	70
Figura 30 - Tela cadastro de igreja.....	71
Figura 31 - Tela lista de membros	71
Figura 32 - Tela cadastro de membros.....	72

Figura 33 - Tela tesouraria	73
Figura 34 - Tela cadastro de movimento e categoria	73
Figura 35 - Tela de usuários.....	74
Figura 36 - Tela cadastro de usuário.....	74
Figura 37 - Vinculo com a Igreja.....	78
Figura 38 - Indutividade da interface do sistema	79
Figura 39 - Grau de facilidade do entendimento do sistema	80
Figura 40 - Grau de Agilidade.....	80
Figura 41 - Grau de produtividade do Sistema	81
Figura 42 - Grau de assertividade das informações	82
Figura 43 - Grau de Responsividade do sistema.....	83
Figura 44 - Grau de acessibilidade ao sistema.....	83
Figura 45 - Desempenho do Sistema	84
Figura 46 - Implantação do sistema	85
Figura 47 - Módulo EaD do sistema atende o ensino/aprendizagem	85

LISTA DE TABELAS

Tabela 1 – Atividades para processo de software	27
Tabela 2 - Característica de componentes	29
Tabela 3 - Requisitos Funcionais.....	47
Tabela 4 – Requisitos Não Funcionais	48
Tabela 5 - Ferramentas e as tecnologias utilizadas.....	57
Tabela 6 - Respostas referente ao questionário da pesquisa.....	77

LISTA DE ABREVIATURAS E SIGLAS

CBSE	Component-based Software Engineering
CSS	Cascading Style Sheets
CNPJ	Cadastro Nacional de Pessoas Jurídicas
EAD	Educação a Distância
HTML	Hyper Text Markup Language
HTTP	Protocolo de Transferência de Hipertexto
IFTO	Instituto Federal de Educação, Ciência e Tecnologia do Tocantins
JS	Java Script
MVC	Modelo Visualização e Controle
MySQL	Gerenciador de banco de dados
ONGs	Organizações não governamental
PHP	Hypertext Preprocessor
RF	Requisitos Funcionais
RNF	Requisitos Não Funcionais
SGE	Sistema de Gestão Evangélica
SQL	Structured Query Language
UML	Unified Modeling Language
WEB	World Wide Web

SUMÁRIO

1. INTRODUÇÃO	16
1.1. Problemática da Pesquisa.....	18
1.2. Objetivos	18
1.2.1. Objetivo Geral.....	18
1.2.2. Objetivos Específicos	18
1.3. Hipótese	19
1.4. Justificativa.....	19
1.5. Organização do Trabalho.....	20
2. FUNDAMENTAÇÃO TEÓRICA	22
2.1. Instituição Religiosa e seus desafios de gestão	22
2.1.1. Administração Eclesiástica	23
2.2. Desenvolvimentos de software web	25
2.2.1. Engenharia de Software	26
2.2.1.1. Processos de software	27
2.2.1.2. Processo de desenvolvimento baseado em componentes	28
2.2.1.3. Padrão MVC	31
2.2.2. Ferramentas e Tecnologias utilizadas no desenvolvimento Web	33
2.2.2.1. Wampserver	33
2.2.2.2. MySQL Workbench	33
2.2.2.3. Astha professional	34
2.2.2.4. Sublime Text.....	34
2.2.2.5. UML Unified Modeling Language	34
2.2.2.6. Bootstrap	34
2.2.2.7. Composer	35
2.2.2.8. mPDF	35
2.2.2.9. PHP	36

2.2.2.10. SQL	36
2.3. Utilização de software / heurísticas de usabilidade	37
2.4. Educação a Distância.....	40
2.5. Trabalhos correlacionados	41
2.5.1. Sistema para igrejas evangélicas multicongregadas	41
2.5.2. Sistema de Gerenciamento de Igrejas (SGI)	42
2.6. Comentários Parciais	42
3. PROCEDIMENTOS METODOLÓGICOS.....	43
3.1. Materiais e Métodos	43
3.2. Desenvolvimento do <i>Software</i>	45
3.2.1. Descrição do Sistema.....	45
3.2.2. Análise e levantamento de Requisitos do Sistema	46
3.2.2.1. Requisitos Funcionais	47
3.2.2.2. Requisitos Não Funcionais	48
3.2.3. Análise do sistema SGE	48
3.2.3.1. Diagrama Caso de Uso	49
3.2.3.2. Diagrama de Classe	50
3.2.3.3. Diagrama de sequência.....	51
3.2.3.4. Diagrama de Modelagem Banco de Dados	55
3.2.4. Organização do Sistema - SGE.....	57
3.2.4.1. Implementação	57
3.2.4.2. Módulos e Telas do Sistema	61
3.3. Avaliação do software	75
3.3.1. Confeção do Questionário da Pesquisa.....	75
3.3.2. Submissão do Questionário da Pesquisa	75
3.4. Comentários Parciais	75
4. ANÁLISE E RESULTADOS DOS DADOS	77
4.1. Comentários Parciais	86

5. CONSIDERAÇÕES FINAIS	87
5.1. Trabalhos Futuros	88
REFERÊNCIAS.....	89
APÊNDICES	96
APÊNDICE A - Questionário usado na coleta de dados da pesquisa de avaliação do sistema	96
PARTE I – Explicação da pesquisa	96
PARTE III – Dados da igreja.....	97
PARTE IV – Sobre o sistema	97
APÊNDICE B – Dados coletados com a aplicação do questionário da pesquisa	100
ANEXOS	102
ANEXO A – Certificado de artigo publicado e apresentado JICE / 2016.	102
ANEXO B – Artigo publicado revista Sítio Novo IFTO / 2018	103
ANEXO C – Artigo publicado na Revista Espiral / 2018.	120

1. INTRODUÇÃO

A sociedade muda, evolui, avança de forma inexorável. Os fantásticos avanços da tecnologia e da ciência contemporâneas são incorporados, com rapidez, pelas mais diversas organizações humanas, que para Oliveira (1998), as quais se modernizam a proporção em que passam a dispor de novas ferramentas em seu arsenal de trabalho. As que não o fizerem, correrão o risco de serem tragadas no caudal da competitividade.

O chamado "Efeito Sap" (migrações para novos sistemas), segundo o portal consmaster (2017), preconiza que sistemas telemáticos prontos serão os responsáveis sobre a completa administração dos negócios.

Nesse sentido, Vianna (1997) salienta que *macrosoftwares* mudarão integralmente a burocracia e a mentalidade da gestão vigente. Haverá uma nova forma de comandar organizações, onde será difícil imaginar que um dia já se trabalhou em clima de um jurássico organograma e de um rígido sistema de controle operado, vagarosamente, por pessoas.

Ainda, segundo Oliveira (1998), as igrejas estão inseridas nesse mesmo contexto sócio organizacional. Não poderão fugir de inúmeras exigências da sociedade pós-moderna, aliás, a informatização tornou-se tão real e natural na vida moderna que, muitas vezes, se integra dentro das igrejas de modo despercebido: na energia elétrica, na água, na telefonia, no serviço interno de som, nos meios de transporte. O microcomputador é apenas uma gota d'água nesse lago profundo. Porém, quem desprezá-lo terá grandes dificuldades em manter-se no cenário que se descortina.

Assim, a tecnologia é relevante na automatização dos processos existentes e um facilitador de mudanças organizacionais propiciando ganhos de produtividade. Beltrame e Maçada, (2009) enfatiza que as tecnologias permitem acessar uma enorme quantia de informações ganhando eficiência ao longo do tempo, como mostra Becker e Lunardi (2005) quando observaram os investimentos em Tecnologia da Informação no setor bancário.

Contudo, o presente trabalho objetivou a desenvolver e validar um Software *web* de gestão das instituições evangélicas, utilizá-lo como ferramenta de auxílio da gestão e desfrutar dos benefícios que possibilitam uma melhoria no ambiente, na

gestão e no acesso dos usuários, que segundo Aniceto (2009) aplicação *Web* é um sistema utilizado através de um navegador, na internet ou em redes privadas, de onde é acessado pelos diferentes usuários, que auxiliasse de forma automatizada e segura a gestão de igreja, facilitando para os interessados o controle de suas instituições de qualquer lugar com disponibilidade de internet e, consecutivamente, agilizando o processo de organização das informações cadastradas. A Tecnologia *Web* surgiu com o objetivo de formar um repositório do conhecimento humano (BERNERS-LEE, 1994).

Desta forma, o trabalho aqui apresentado descreve a ferramenta de gerenciamento de igrejas, no que se refere à gestão dos dados cadastrais dos envolvidos como controle de campus, igrejas, membros, financeiro, oferta de cursos bíblicos, escola bíblica dominical, teológicos entre outros, com emissão de certificados, e relatórios, bem como apresentar os passos utilizados na sua elaboração e validação.

Antes de tudo, levantaram-se os aspectos teóricos sobre as áreas envolvidas no presente trabalho, tais como, conceitos de Instituição Religiosa e seus desafios de gestão, desenvolvimento de software web, utilização de software / heurísticas de usabilidade, Educação a Distância - EaD e trabalhos correlacionados.

Desta forma, o processo de desenvolvimento foi dividido em 3 (três) etapas que foram realizadas criteriosamente: a primeira, definiu os requisitos essenciais para a construção do *Software* e a preparação do ambiente de desenvolvimento; a segunda, foi o desenvolvimento e implantação e a terceira foi a pesquisa com o intuito de avaliar a aceitabilidade do *software* implantado.

Em síntese, o método de pesquisa utilizado empregou-se uma pesquisa qualitativa com enfoque quantitativo. A pesquisa foi realizada com o universo de 72 participantes, sendo: 1 pastor presidente, 2 pastores locais, 1 secretário, 1 tesoureiro 37 membros, 16 congregados e 14 classificados como outros. Como instrumento de coleta dos dados, utilizou-se um questionário aplicado via *Web*. Os resultados obtidos mostram que o sistema desenvolvido atende as necessidades e melhora a gestão da instituição religiosa, porque 99% dos pesquisados responderam que indicariam o sistema para implantação na instituição religiosa, mostrando, com isso, a relevância do presente trabalho.

1.1. Problemática da Pesquisa

O desenvolvimento deste estudo iniciou após observar a problemática evidenciada pela falta de praticidade percebida pelos Líderes da Igreja Evangélica Assembleia de Deus Ministério da Salvação em Cristo de Porto Nacional – TO, pois a utilização de fichas impressas, recibos, livros caixa preenchidos de forma manuscrita, estavam colecionando erros, informações desencontradas. Assim, pensou-se na criação de um sistema que auxiliasse a os dirigentes a gerir com mais eficiência a instituição religiosa, contendo no seu escopo uma plataforma de cursos bíblicos e teológicos a distância, a fim de qualificarem seus membros e obreiros com mais comodidade para que exerçam suas funções com excelência.

Pois, percebeu-se que sem um *Software* de gestão para auxiliar na administração da instituição religiosa Salvação em Cristo, levou a igreja colecionar muitas informações sem qualidade, registros desencontrados, com pouca assertividade, obreiros desqualificados e que acarretou decisões equivocadas pela igreja.

Contudo, foram essas observações que motivaram a presente pesquisa, criando a perspectiva de desenvolver uma aplicação *web* que reduzisse os problemas com a gestão de igrejas. Diante disso, elaborou-se o seguinte questionamento: Quais os ganhos produtivos que a igreja Assembleia de Deus, Ministério da Salvação em Cristo de Porto Nacional no Tocantins, terá com a construção e utilização de um Software de Gestão Evangélica?

1.2. Objetivos

Nesta seção estão descritos os objetivos gerais e específicos a serem alcançados através desta pesquisa.

1.2.1. Objetivo Geral

Desenvolver e validar um Software de gestão das instituições evangélicas, utilizá-lo como ferramenta de auxílio da gestão e desfrutar dos benefícios tecnológicos.

1.2.2. Objetivos Específicos

Para alcançarmos o objetivo anterior, especificamos em:

- a) Desenvolver um Software web para Gestão Evangélica;
- b) Utilizar o Software como ferramenta de auxílio da Gestão;
- c) Desfrutar dos benefícios como: facilidade, agilidade, assertividade e maior produtividade.

1.3. Hipótese

A hipótese levantada que solucionará a problemática constitui-se, no desenvolvimento e implantação de um software que auxilie na gestão da igreja Assembleia de Deus Ministério da Salvação em Cristo, que garantirá benefícios como: facilidade; agilidade; assertividade e maior produtividade.

1.4. Justificativa

Nas últimas décadas, houve uma grande transformação da composição da população religiosa no Brasil. Esta mudança se caracteriza pela queda da população católica e pelo simultâneo crescimento da comunidade evangélica. Só na década de 1990, praticamente dobrou, saltando de pouco mais de 13 milhões de fiéis, em 1991, para cerca de 26 milhões em 2000 (NERI, CARVALHÃES, & MONTE, 2011).

O crescimento dos evangélicos no Brasil continua intenso, segundo levantamento realizado em todo o país pelo Instituto Datafolha (2016), o número chega a 29%, sete pontos percentuais a mais do que o Censo 2010 do Instituto Brasileiro de Geografia e Estatística – IBGE. Atualmente 45 milhões de pessoas se consideram evangélica (IBGE, 2010).

Essa imensa multidão procura uma igreja para se congregar que resulta no aumento do número de igrejas evangélicas, mas, também, no crescimento do número de membros nas igrejas tradicionais, aquelas que já existiam. Isso remete a uma certeza, é preciso controlar esse crescimento de forma profissional, sem dúvidas.

Segundo o Portal *InfoMoney* (2016), pesquisa desenvolvida pelo IDG *Research Services* mostrou que empresas e instituições que se utilizam de sistemas de gestão tem um nível de crescimento 35% mais rápido e 10% a mais de eficiência do que aquelas que não tem acesso a mesma ferramenta.

Porém, não adiantam informatizar a desorganização e ter uma coleção de informações sem qualidade, registros desencontrados e com pouca assertividade, podendo levar a decisões erradas que gera um custo, às vezes, irreparável.

Assim, o presente sistema *Web* personalizado com as funcionalidades pertinentes a gestão evangélica, conta com ferramentas tecnológicas que uma simples anotação ou linhas de planilhas não podem contar, interação, metodologias próprias e específicas de entradas de dados que visam à consolidação das informações de forma que elas possam ser aproveitadas na sua totalidade.

As facilidades do uso desta ferramenta permitem aos pastores, dirigentes e gestores uma visão crítica e realista sobre a sua membresia, suas receitas, as despesas e gastos; contribui na tomada de decisões, fortalece o crescimento e o relacionamento com os próprios membros, colaboradores e fornecedores; garante facilidade, agilidade, assertividade e maior produtividade para a instituição religiosa.

Em conformidade com as respostas dos seguintes questionamentos levantados na avaliação do software: “Você indicaria esse sistema para ser implantado na sua igreja? ”, na qual 99% dos pesquisados responderam que SIM; “O Módulo EaD do sistema atende o ensino/aprendizagem da igreja? ”, 99% responderam que SIM”, assim, esta pesquisa ganha ênfase justificando a sua relevância em consonância com alta média apresentada nas respostas.

Da literatura consultada, não foi encontrado nenhum trabalho que apresentasse a mesma proposta levando em consideração a problemática solucionada, nem mesmo os objetivos alcançados. Desta forma, este trabalho ganha ênfase e se justifica para a realização, pois, é almejado diminuir os problemas na gestão de igrejas com um software Web que seja possível prover aos líderes evangélicos dados precisos no processo de gestão de igrejas.

1.5. Organização do Trabalho

Para melhor entendimento, o presente trabalho está organizado em cinco capítulos:

No Capítulo 1, é apresentada a contextualização, o tema, a definição do problema, os objetivos a serem alcançados, a hipótese levantada, a justificativa para realização do presente estudo e, por fim, a presente organização do trabalho. No Capítulo 2, é apresentado o referencial teórico, em que são abordados elementos que proporcionam o suporte conceitual necessário para a elaboração da atual pesquisa. Os elementos abordados são: Instituição Religiosa e seus desafios de gestão, desenvolvimento de software web, Utilização de Software / heurísticas de usabilidade e Educação a Distância – EaD e trabalhos correlacionados. No Capítulo 3, está descrito de forma sistematizada os procedimentos metodológicos, onde estão apresentadas de forma minuciosa, todas as etapas do desenvolvimento deste estudo: são apresentados os materiais e métodos; desenvolvimento e implantação do software e avaliação. No Capítulo 4, são abordadas as discussões, análises e resultados da pesquisa. No Capítulo 5, é apresentado as conclusões as quais se

chegaram após o presente estudo, bem como as perspectivas e propostas para trabalhos futuros.

2. FUNDAMENTAÇÃO TEÓRICA

Este capítulo apresentará os aspectos teóricos sobre as áreas envolvidas no presente trabalho. São apresentados conceitos de Instituição Religiosa e seus desafios de gestão, desenvolvimento de software web, utilização de software / heurísticas de usabilidade, Educação a Distância - EaD e trabalhos correlacionados.

2.1. Instituição Religiosa e seus desafios de gestão

O reconhecimento efetivo da existência das Instituições Religiosas se deu através da Lei nº 10.825, de 22 de dezembro de 2003 pela inclusão do § 1º e do inciso IV do artigo 44 do Código Civil Brasileiro que diz:

§ 1º São livres a criação, a organização, a estruturação interna e o funcionamento das organizações religiosas, sendo vedado ao poder público negar-lhes reconhecimento ou registro dos atos constitutivos e necessários ao seu funcionamento. (CCB, 2003).

Esse reconhecimento das Instituições Religiosas, na visão de Monello (2012), é a legalização como pessoa jurídica de direito privado e, conseqüentemente, no cumprimento das demais exigências legais e fiscais aplicáveis às pessoas jurídicas. Entre as exigências legais, estão previstos o registro de suas atas, de seu estatuto organizativo, associativo, fundacional, regimentos, regulamentos, inscrição no Cadastro Nacional de Contribuintes do Ministério da Fazenda (CNPJ), na prefeitura, cartórios entre outros, bem como o cumprimento de todas as obrigações acessórias exigidas em lei.

Snyder e Runyon (2002), argumentam que a Igreja enquanto organização é estática, cheia de padrões, técnica, que traz ordem a si mesma. A adesão à autoridade formal e à tradição torna-se natural. Ao lado do elemento fé, nasce a doutrina, ao lado da comunhão, a instituição dos sacramentos, ao lado do serviço, os cargos e funções. A Igreja para se preservar apresenta três elementos: doutrina, sacramentos e funções, mas ao reconhecer o papel de cada um, começa a vislumbrar as possibilidades de uma congregação saudável.

A abordagem da escolha racional da religião, em quaisquer de suas vertentes, ainda não mereceu a devida atenção de pesquisadores locais, seja qual for sua área de conhecimento. No contexto das ciências da administração, ela encontra-se, até os dias de hoje, totalmente ausente. Como salientado por Frigerio (2008, p. 17), tal abordagem “tem obtido pouca atenção nos meios acadêmicos

latino-americanos”. Para o autor, poucos são os trabalhos locais que explicitam com precisão seus argumentos. O Brasil não se constitui exceção (Mariano, 2008).

Entretanto, “o campo religioso brasileiro é o da pluralidade” (Passos et al., 2011, p. 694). “Liberdade para as religiões é o que não falta por aqui” (Pierucci, 1996, p. 276). Resulta, daí a elevação da competitividade entre igrejas e o aumento da luta pela sobrevivência (Mariano, 2008, 2013; Pierucci, 1999, 2008). Com efeito, a franca concorrência decorrida do pluralismo acabou provocando o aparecimento de agentes religiosos mais ativos, dinâmicos e bem-dispostos, e em organizações religiosas mais eficientes na mobilização de séquitos, públicos e clientelas (PIERUCCI, 1999, 2006a, 2006b, 2008).

Segundo Neri, Carvalhães e Monte (2011, p. 42), uma “maior ligação entre o espírito empresarial e a organização religiosa seria uma marca dos ramos religiosos emergentes no Brasil”. Isso poderia ser vislumbrado não apenas nas interações entre diferentes igrejas e mundo produtivo, mas também na própria natureza de organizações religiosas, que adquirem um caráter mais empreendedor.

2.1.1. Administração Eclesiástica

Para compreender de maneira concisa o conceito de administração eclesiástica necessitou-se a segmentação ou a separação dos termos a fim de delinear e detalhar seus conceitos dentro do que se encaixa a teoria.

Segundo Stoner (1999) a administração é o processo de planejar, organizar, liderar e controlar os esforços realizados pelos membros da organização e o uso de todos os outros recursos organizacionais para alcançar os objetivos estabelecidos.

Assim, Chiavenato (2000) complementa afirmando que o papel principal da administração é gerenciar os recursos e tomar decisões a fim de conquistar um objetivo definido. A tarefa básica da administração é a de fazer as coisas por meio de pessoas de maneira eficiente e eficaz.

Para Rebrates (2011) a organização de uma sociedade constituída, comporta três setores, a saber: I. Primeiro Setor corresponde à vontade popular pelo voto que confere o poder ao governo; II. Segundo Setor corresponde à iniciativa de atuar no mercado usando o lucro como instrumento principal e III. Terceiro Setor corresponde às instituições com preocupações e práticas sociais, sem fins lucrativos, que geram bens e serviços de caráter público, tais como ONGs (Organizações não Governamentais), instituições religiosas, clubes de serviços, entidades beneficentes, centros sociais, organizações de voluntariado etc.

Portanto, as instituições religiosas se inserem no terceiro setor, que se definem como instituições sem fins lucrativos, na maioria dos casos regida no âmbito particular exclusivamente na execução de atividades de utilidade pública. Drucker (1995, p. 9) declara:

“As organizações sem fins lucrativos não têm “lucros”. Elas têm a tendência de considerar tudo aquilo que fazem como justo, moral e a serviço de uma causa; assim, não se mostram dispostas a dizer, caso alguma coisa não produza resultados, que seus recursos devem ser redirecionados. Talvez as organizações sem fins lucrativos necessitem mais da disciplina do abandono organizado do que de uma empresa. Elas precisam enfrentar opções críticas.”

Conceição (2013), afirma que a administração moderna possibilita a inserção de novas terminologias aos estudos, basicamente com o intuito de apurar o conhecimento quanto as suas práticas nas mais diversas áreas do conhecimento. Assim, ao agregar o termo “eclesiástico” ao termo administração direciona-se o estudo dessas atividades a de uma Igreja ou instituição religiosa que possa ser gerenciada.

No que tange a administração como ciência, esta pode ser considerada uma ciência social segundo Damke, Walter e Silva (2010), visto que seu objeto de estudo se constitui de fenômenos de ordem social, ou seja, do estudo e da melhoria da coordenação e do controle de atividades humanas associadas. Além disso, a administração também pode ser classificada como uma ciência aplicada, ou seja, uma ciência social com direção específica.

Ainda na perspectiva de Conceição (2013), o termo eclesiástico está intimamente ligado a questões religiosas, relativo ou concernente à igreja, ao clero e/ou sacerdócio, ou seja, o corpo ministerial de obreiros, líderes religiosos e administradores, portanto se refere à igreja e termos religiosos.

Porém, a administração das organizações religiosas segue os mesmos princípios da administração de qualquer organização, regendo-se pelas mesmas leis, apenas com as vistas voltadas para outros objetivos (MONELLO, 2012).

Segundo Kessler e Câmara (1987) administração eclesiástica é o estudo dos diversos assuntos ligados ao trabalho do pastor/padre no que tange a sua função de líder ou administrador principal da igreja a que serve.

As organizações que tem como fundamento de conduta a Palavra de Deus, ou seja, a Bíblia Sagrada, a transparência e clareza de suas movimentações precisam estar alinhadas a ordem, para que os processos administrativos não

venham prejudicar os objetivos maiores, que neste caso é propagação do Evangelho, alcançando os indivíduos presentes na sociedade onde se inserem (CONCEIÇÃO, 2013, p. 8 e 9).

Muzio (2011) afirma que o amadorismo administrativo e a desorganização legal afetaram o crescimento de igrejas na América latina na tentativa de planejar com excelência e estabelecer alvos, limitando o gerenciamento eclesiástico. Continua ressaltando que é necessário que a igreja atual deve estabelecer um novo diálogo entre o modelo de liderança da igreja com os novos paradigmas do século XXI.

“Hoje é notório que a instituição cristã tem sido bem mais que casa de oração, ou assembleia. A igreja do século XXI tem sido espaço para lazer, entretenimento, espaço para o crescimento profissional e social” (SANTOS, 2010).

Santos (2010), ainda argumenta que toda igreja possui de alguma forma um banco de dados, por muita das vezes em arquivo impresso e de preenchimento manual. Para um controle interno administrativo efetivo, pode-se estabelecer um banco de dados virtual onde cadastros de membros, atas, escalas internas, controle de entrada e saída de caixa entre outros, podem ser bem manipulados por um bom sistema. Através desta ferramenta, a fidelidade do gerenciamento de qualquer informação inserida, torna-se mais segura e precisa.

2.2. Desenvolvimentos de software web

Sommerville (2011) conceitua *Softwares*, como sendo programas de computador com documentação associada. Produtos de *software* podem ser desenvolvidos para um cliente específico ou para o mercado em geral.

Na ótica de Pressman (2011), *Software* é o produto que profissionais desta área constroem e, depois, mantêm ao longo do tempo. Esse produto abrange programas executáveis, código-fonte, documentação na forma impressa ou virtual, entre outros.

Por outro lado, Seibel (2008) conceitua *Software web* como sistemas que executam em ambientes distribuídos, onde as partes do sistema podem executar em máquinas diferentes comunicando-se via protocolo HTTP - Protocolo de Transferência de Hipertexto ou HTTPS (Protocolo Seguro), onde a interface com o usuário é realizada pelos navegadores.

Assim, as aplicações desenvolvidas utilizando o paradigma cliente-servidor já vivenciam esta experiência, onde a parte relativa à interface com o usuário reside na

estação do cliente e a parte de acesso aos dados no servidor de banco de dados. Seibel (2008) conclui que a lógica da aplicação pode ficar dividida entre o cliente e o servidor. Esta é uma arquitetura em dois níveis.

Portanto, o surgimento da Internet trouxe uma mudança significativa na maneira como o software corporativo é organizado (SOMMERVILLE 2011). Antes da Internet, aplicações corporativas eram, na maioria das vezes, monolíticas, programas isolados executando em computadores isolados ou em *clusters* de computadores. Para Pressman (2006, 2011), um software é altamente distribuído, às vezes pelo mundo todo. As aplicações corporativas não são programadas do zero, de fato, elas envolvem reuso extensivo de componentes e programas.

2.2.1. Engenharia de Software

Na visão de Rezende (2005), a engenharia de software é uma metodologia que desenvolve soluções profissionais utilizando-se dos recursos de *software*, observando os padrões requeridos de qualidade, produtividade e efetividade.

Para Carvalho e Chiossi (2001), a Engenharia pode ser compreendida como uma disciplina que reúne metodologias, métodos e ferramentas a serem utilizados, desde a percepção do problema até o momento que o sistema desenvolvido deixa de ser operacional, visando resolver problemas inerentes ao processo de desenvolvimento e ao produto de software.

Do mesmo modo, Sommerville (2011) define Engenharia de software como sendo uma disciplina de engenharia cujo foco está em todos os aspectos da produção de *software*, desde os estágios iniciais da especificação do sistema até sua manutenção, quando o sistema já está sendo usado.

Em geral, os engenheiros de software adotam uma abordagem sistemática e organizada para seu trabalho, pois essa costuma ser a maneira mais eficiente de produzir software de alta qualidade conforme pontua (SOMMERVILLE, 2011, p. 5).

“No entanto, engenharia tem tudo a ver com selecionar o método mais adequado para um conjunto de circunstâncias, então uma abordagem mais criativa e menos formal pode ser eficiente em algumas circunstâncias. Desenvolvimento menos formal é particularmente adequado para o desenvolvimento de sistemas Web, que requerem uma mistura de habilidades de software e de projeto”.

Com base nesses conceitos, percebe-se que, quando se trabalha no desenvolvimento de um produto ou sistema, é importante seguir uma série de passos previsíveis, um roteiro que ajude a criar um resultado de alta qualidade e

dentro do prazo estabelecido. Esse roteiro dentro da engenharia, segundo Pressman (2011), é denominado processo de software.

2.2.1.1. Processos de software

Fuggetta (2000), Pressman (2006, 2011) e Osterweil (1987) afirmam que a qualidade do processo de desenvolvimento impacta diretamente na qualidade do produto a ser desenvolvido. O processo de software pode ser definido como um conjunto de políticas, estruturas organizacionais, tecnologias, procedimentos e artefatos necessários para conceber, desenvolver, implantar e manter um produto de software (Fuggetta, 2000).

Segundo Greer e Conradi (2009), a necessidade de um processo definido tem sido reconhecida como uma estratégia de redução de riscos para a gerência de projetos. Bertollo e Falbo (2003) acrescentam que a principal causa dos problemas no desenvolvimento de software é a falta de um processo de desenvolvimento de software claramente definido e efetivo.

Uma das abordagens propostas para melhorar a qualidade dos processos de software é prover a reutilização de processos, a partir de processos definidos previamente, disseminando conhecimentos e boas práticas entre os projetos de uma organização (KELLNER, 1996).

Por certo, um processo de software é um conjunto de atividades relacionadas que levam à produção de um produto de software.

“Essas atividades podem envolver o desenvolvimento de software a partir do zero em uma linguagem padrão de programação como Java ou C., no entanto, aplicações de negócios não são necessariamente desenvolvidas dessa forma. Atualmente, novos softwares de negócios são desenvolvidos por meio da extensão e modificação de sistemas existentes ou por meio da configuração e integração de prateleira ou componentes do sistema (SOMMEVILLE, 2011, p. 18).”

Existem muitos processos de *software* diferentes, mas todos devem incluir quatro atividades fundamentais para a engenharia de *software* (SOMMEVILLE 2011). Conforme descrito na Tabela 1.

Tabela 1 – Atividades para processo de software

Atividades	Descrição
1. Especificação de software.	A funcionalidade do software e as restrições a seu funcionamento devem ser definidas.

2. Projeto e implementação de software.	O software deve ser produzido para atender às especificações.
3. Validação de software.	O software deve ser validado para garantir que atenda às demandas do cliente.
4. Evolução de software.	O software deve evoluir para atender às necessidades de mudança dos clientes.

FONTE: Sommerville, 2011, adaptada pelo autor.

De alguma forma, essas atividades fazem parte de todos os processos de software. Na prática, são atividades complexas em si mesmas, que incluem subatividades como validação de requisitos, projeto de arquitetura, testes unitários etc. (PRESSMAN 2006). Existem também as atividades que dão apoio ao processo, como documentação e gerenciamento de configuração de *software*.

2.2.1.2. Processo de desenvolvimento baseado em componentes

Conforme preconiza Pressman (2011), quando uma empresa não pode usar um sistema de prateleira, porque eles não atendem a seus requisitos, o software de que necessitam precisa ser especialmente desenvolvido. Brown (2000) diz que para o desenvolvimento de softwares customizados, a engenharia de software baseada em componentes é uma forma eficaz, orientada ao reuso.

De acordo com Sommeville (2011), engenharia de software baseada em componentes (CBSE, do inglês *component-based software engineering*) surgiu na década de 1990 como uma abordagem para softwares de desenvolvimento de sistemas com base no reuso de componentes de softwares.

Inúmeras são as definições de componentes reusáveis são encontradas na literatura. Em Sametinger (1997) é apresentado que os componentes podem ser vistos como alguma parte do sistema de software que é identificável e reusável, ou como o estado seguinte de abstração depois de funções, módulos e classes.

Pressman (2011) e Sommeville (2011) concordam entre si, que os componentes são abstrações de nível mais alto do que objetos e são definidos por suas interfaces. Geralmente, eles são maiores que objetos individuais e todos os detalhes de implementação são escondidos de outros componentes. CBSE é o processo de definir, implementar, integrar ou compor componentes independentes, pouco acoplados em sistemas.

Na comunidade CBSE, não existe consenso sobre um componente ser uma unidade independente de software que pode ser composta com outros componentes para criar um sistema de software. Além disso, os autores têm definições diferentes a respeito de um componente de software. Councill e Heineman (2001) definem um componente como “Um elemento de software que está de acordo com um modelo de componente padrão e pode ser independentemente implantado e composto, sem modificações, de acordo com um padrão de composição”.

Essa definição é essencialmente baseada em padrões, assim, uma unidade de software que esteja em conformidade com esses padrões é um componente. No entanto, Szyperski (2002) não menciona padrões em sua definição de um componente, mas, em vez disso, concentra-se nas principais características dos componentes:

“Um componente de software é uma unidade de composição com interfaces contratualmente especificadas e apenas dependências de contexto explícitas. Um componente de software pode ser implantado de forma independente e está sujeito a ser composto por parte de terceiros. ”

O que as definições anteriormente mencionadas têm em comum é o fato de concordarem que os componentes são independentes e que, em um sistema, eles são a unidade fundamental de composição. Na opinião de Sommeville (2011), uma melhor definição de um componente pode ser obtida combinando-se essas propostas.

A Tabela 2 mostra o que Sommeville (2011), considera como características essenciais de um componente como na CBSE.

Tabela 2 - Característica de componentes

Característica do componente	Descrição
Padronizado	A padronização de componentes significa que um componente usado em um processo CBSE precisa obedecer a um modelo de componentes padrão. Esse modelo pode definir as interfaces de componentes, metadados de componente, documentação, composição e implantação.
Passível de composição	Um componente deve ser independente. Deve ser possível compor e implantá-lo sem precisar usar outros componentes específicos. Nessas situações em que os componentes precisam dos serviços prestados externamente, estes devem ser explicitamente definidos em uma especificação de interface ‘requires’.
Implantável	Para ser implantável, um componente deve ser autocontido. Deve ser capaz de operar como uma

	entidade autônoma em uma plataforma de componentes que forneça uma implementação do modelo de componentes, o que geralmente significa que o componente é binário e não tem como ser compilado antes de ser implantado. Se um componente é implantado como um serviço, ele não precisa ser implantado por um usuário de um componente. Pelo contrário, é implantado pelo prestador do serviço.
Documentado	Os componentes devem ser completamente documentados para que os potenciais usuários possam decidir se satisfazem a suas necessidades. A sintaxe e, idealmente, a semântica de todas as interfaces de componentes devem ser especificadas.

FONTE: Sommerville (2011).

Atualmente, as empresas operam em um ambiente global, com mudanças rápidas. Assim, precisam responder a novas oportunidades e novos mercados, a mudanças nas condições econômicas e ao surgimento de produtos e serviços concorrentes.

“Softwares fazem parte de quase todas as operações de negócios, assim, novos softwares são desenvolvidos rapidamente para obterem proveito de novas oportunidades e responder às pressões competitivas. O desenvolvimento e entrega rápidos são, portanto, o requisito mais crítico para o desenvolvimento de sistemas de software. Na verdade, muitas empresas estão dispostas a optar por uma implantação mais rápida do software de que necessitam (SOMMEVILLE, 2011, p. 18).”

Dessa forma, o desenvolvimento Baseado em Componentes surge como uma nova perspectiva de desenvolvimento de software caracterizada pela composição de partes já existentes. Conforme Szyperski (2002), construir novas soluções pela combinação de componentes desenvolvidos aumenta a qualidade e dá suporte ao rápido desenvolvimento, levando à diminuição do tempo de entrega do produto ao mercado.

Ainda em Szyperski (1999) vamos encontrar o seguinte esclarecimento, sistemas definidos através da composição de componentes permitem que sejam adicionadas, removidas e substituídas partes do sistema sem a necessidade de sua completa substituição. Com isso, CBSE auxilia na manutenção dos sistemas de software, por permitir que os sistemas sejam atualizados através da integração de novos componentes e/ou atualização dos componentes já existentes.

Entretanto, para que essa integração aconteça, surge à necessidade de uma estrutura que venha integrar/unir esses componentes, ou seja, uma arquitetura padrão que dão suporte ao reuso dos componentes. Para o desenvolvimento do

sistema SGE, utilizou-se para esse fim o padrão MVC (modelo-visão-controlador, do inglês *Model-View-Controller*), (PRESSMAN, 2011). Esse padrão é a base para gerenciamento da interação em muitos sistemas baseados em Web.

2.2.1.3. Padrão MVC

A ideia de padrões como uma forma de apresentar, compartilhar e reusar o conhecimento sobre sistemas de software é, hoje, amplamente utilizada. O gatilho para isso foi a publicação de um livro sobre os padrões de projeto orientado a objetos (GAMMA et al., 1995), o que levou ao desenvolvimento de outros tipos de padrões, como padrões para o projeto organizacional (COPLIEN e HARRISON, 2004), padrões de usabilidade e interação (MARTIN e SOMMERVILLE, 2004), gerenciamento de configuração e assim por diante.

Do mesmo modo, o portal macoratti.net esclarece que a arquitetura MVC - (Modelo Visualização e Controle) fornece uma maneira de dividir a funcionalidade envolvida na manutenção e apresentação dos dados de uma aplicação, dividindo as responsabilidades nas seguintes camadas:

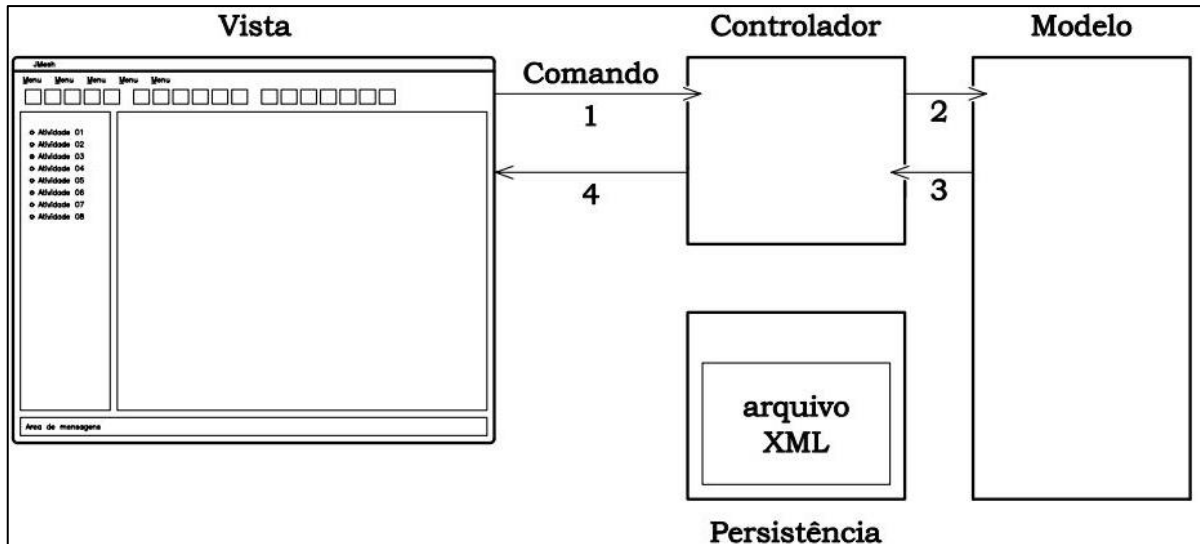
- a) **Model (modelo)**, a camada que gerencia os modelos de dados da aplicação, persistência de informações e o acesso aos dados em si;
- b) **View (visualização)**, que são as interfaces gráficas, as telas da aplicação propriamente ditas, sendo a camada de comunicação com o usuário, manipulação e exibição dos dados;
- c) **Controller (controlador)**, a camada que realiza a comunicação da view com o sistema, recebendo e redirecionando requisições, decidindo quais views exibir, e realizando requisições para a camada de modelo;

Pressman (2004, 2011) afirma que o padrão MVC, separa os elementos de um sistema, permitindo mudá-los de forma independente. Por exemplo, pode-se adicionar uma nova visão ou alterar uma exibição existente sem quaisquer alterações nos dados subjacentes do modelo.

Outro ensinamento é o de Sommerville (2011), em que o padrão de arquitetura em camadas é outra maneira de conseguir a separação e independência. Onde, a funcionalidade do sistema é organizada em camadas separadas, e cada camada só depende dos recursos e serviços oferecidos pela camada imediatamente abaixo dela.

Como pode ser visto na figura 1, o fluxo de informações para realização de cada tarefa ocorre em quatro etapas segundo Booch et al. (2000).

Figura 1- Arquitetura de aplicações Web usando o padrão MVC



FONTE: Booch et al. (2000).

Na primeira etapa, o objeto Comando, responsável pela tarefa, aciona o Controlador ativo informando a requisição. A seguir (2), o Controlador cria o objeto correspondente à entidade e o adiciona ao Modelo pertinente. Na etapa 3, o Controlador cria objetos representativos dos objetos do Modelo. Finalmente, na etapa 4, os objetos pertencentes ao Controlador são apresentados na área da vista.

Em síntese, todas as requisições feitas pelo usuário são enviadas ao *Controller*. Este manipula os dados usando o *Model* e invoca a *View* correta, de acordo com a ação executada ou com os resultados vindos do *Model* (BOOCH E TAL. (2000).

A grande vantagem de se utilizar o padrão MVC, na visão de Minetto (2000), é a separação de lógica e apresentação, sendo que isso favorece o trabalho em equipe. Um designer poderia trabalhar na apresentação, definindo o HTML, CSS, JAVASCRIPT, enquanto um (*Data Base Administrator* DBA), administrador de banco de dados, poderia trabalhar com o modelo e outro programador poderia se concentrar nas regras de negócio inseridas no controlador. Dessa forma, qualquer mudança, por exemplo, na apresentação, teria pouco ou nenhum impacto nas demais camadas da aplicação.

2.2.2. Ferramentas e Tecnologias utilizadas no desenvolvimento Web

No desenvolvimento Web faz-se necessário o uso de recursos para criação de website e sistemas. Este tópico aborda sobre os conceitos das ferramentas e tecnologias que foram utilizadas no desenvolvimento do sistema SGE – Gestão de igrejas, todas as ferramentas utilizadas são livres. As ferramentas utilizadas foram Wampserver, MySQL Workbench, Astha professional, Sublime Text e as tecnologias ou linguagens usadas no desenvolvimento foram UML, Bootstrap, Composer, mPDF, PHP e SQL.

2.2.2.1. Wampserver

Segundo a documentação desta ferramenta, quando desenvolve site /sistema há necessidade de testar e validar códigos precisa fazer vários testes para ter a certeza que tudo está em ordem, e para isso, é preciso fazer diversos upload dos arquivos, e se esses testes fossem feitos por um servidor web, seria mais demorado, pois necessitaria de usar uma conexão via internet. Já o servidor local (Localhost) não necessita de acesso à internet, por tanto esse processo é mais rápido e prático.

Nesse sentido, escolheu-se o *WampServer* para o desenvolvimento desta aplicação segundo a documentação do *WampServer*, este é uma plataforma de desenvolvimento web baseada em Windows para aplicações web dinâmicas usando o servidor Apache2, linguagem de script PHP e banco de dados MySQL. Ele também tem o *PHPMYAdmin* para gerenciar bancos de dados com mais facilidade.

2.2.2.2. MySQL Workbench

Na documentação da ferramenta define O MySQL Workbench como sendo uma ferramenta visual unificada para arquitetos de banco de dados, desenvolvedores e DBAs. A palavra origina do inglês e significa *Database Administrator* (Administrador de Bancos de Dados). O MySQL Workbench fornece modelagem de dados, desenvolvimento de SQL (*Structured Query Language*) é a linguagem padrão universal para manipular bancos de dados relacionais e ferramentas abrangentes de administração para configuração de servidores, administração de usuários e backup.

Ainda na documentação o MySQL Workbench permite que um DBA, um desenvolvedor ou um arquiteto de dados projete, modele, gere e gerencie visualmente bancos de dados. Ele inclui tudo o que um modelador de dados precisa para criar modelos complexos de ER – Entidade Relacional, engenharia direta e reversa e também fornece os principais recursos para realizar tarefas difíceis de

gerenciamento de alterações e documentação que normalmente exigem muito tempo e esforço.

2.2.2.3. Astah professional

De acordo com o portal techtudo, Astah professional é um software para modelagem UML (*Unified Modeling Language* – Linguagem de Modelagem Unificada) com suporte a UML 2, desenvolvido pela Change Vision, Inc e disponível para sistemas operacionais Windows 64 bits. Anteriormente conhecido por JUDE, um acrônimo de Java and UML Developers Environment (Ambiente para Desenvolvedores UML e Java).

Astah professional disponibiliza para desenvolvimento, os diagramas de Classes, Casos de Uso, Sequência, Comunicação, Máquina de Estados, Atividade, Componentes, Implantação e Diagrama de Estrutura Composta.

2.2.2.4. Sublime Text

Para Pinto (2013), o sublime text é um editor de texto é essencial em qualquer sistema operativo. Além das funcionalidades básicas de edição de texto, este tipo de ferramenta integra hoje muitas funções direcionadas para a programação.

Sublime Text é um editor de texto e código-fonte multiplataforma, escrito em linguagem C++ que foi inicialmente pensado para ser uma extensão do vim. Este editor oferece recursos extraordinários e um desempenho simplesmente surpreendente.

2.2.2.5. UML Unified Modeling Language

UML é um acrônimo para a expressão *Unified Modeling Language*. Pela definição do nome, segundo Ribeiro (2012) do portal Devmidia, a UML é uma linguagem que define uma série de artefatos que ajuda na tarefa de modelar e documentar os sistemas orientados a objetos.

A UML permite montar e visualizar um escopo do projeto em desenvolvimento. Segundo Fowler; Scott (2000, p.19) “UML (Unified Modeling Language) é a sucessora da onda de métodos de análise e projeto orientado a objetos (OOA& D) que surgiu no final dos anos oitenta e no início dos anos noventa”. Dessa forma o desenvolvedor pode visualizar seu produto em diagramas personalizados.

2.2.2.6. Bootstrap

Bootstrap é o mais popular framework JavaScript, HTML e CSS para desenvolvimento de sites e aplicações web responsivas e alinhadas com a filosofia

mobile first. Torna o desenvolvimento *front-end* muito mais rápido e fácil. Indicado para desenvolvedores de todos os níveis de conhecimento, dispositivos de todos os tipos e projetos de todos os tamanhos. O site do *Bootstrap* o define como um poderoso, elegante e intuitivo *framework front-end* que possibilita um desenvolvimento web de modo ágil e fácil (SILVA, 2015).

O *Bootstrap* foi criado para agilizar o desenvolvimento *front-end*, proporcionando agilidade e qualidade no desenvolvimento de interfaces. Esse framework oferece componentes de interfaces prontos em que o desenvolvedor não precisará ficar perdendo tempo na criação de componentes de interfaces, fazendo que o tempo de desenvolvimento da aplicação seja otimizado.

2.2.2.7. Composer

No portal onde se encontra a documentação descreve que o Composer é uma ferramenta para o gerenciamento de dependências em PHP. Ele permite que o desenvolvedor declare as bibliotecas que o projeto necessita, ele gerencia, instala e atualiza essas dependências.

As dependências são todos os artefatos bibliotecas e pacotes de software ou de sistemas de terceiros que o projeto utiliza para funcionar. O principal objetivo do Composer, segundo o blog da *kinghost*, é retirar a responsabilidade dos desenvolvedores de instalar e atualizar bibliotecas. O Composer utiliza um arquivo chamado *composer.json*, nele são declaradas as bibliotecas que o projeto depende e com comandos apropriados, os conteúdos são baixados e instalados.

2.2.2.8. mPDF

PHP não possui uma IDE para geração de relatórios ainda, por isso escolheu-se o mPDF, que conforme a documentação da ferramenta, é uma biblioteca PHP que gera arquivos PDF a partir do HTML codificado em UTF-8. A biblioteca mPDF trabalha de maneira bem simples, passa-se o código HTML contendo o layout do documento e a folha de estilo CSS para estilização, depois é o método para renderizar o arquivo PDF e abrir a página de leitura, atende bem as necessidades quando precisa gerar qualquer tipo de arquivo no formato PDF com PHP.

Para o portal Devwilliam, essa biblioteca trabalha com a estrutura do relatório em HTML e a estilização com folhas de estilo CSS. Com ela tem-se maior poder de customização no relatório, basicamente escreve um código HTML, exemplo uma tabela, estiliza ela com CSS e passa esses conteúdos para o objeto instanciado a

partir da biblioteca mPDF, usando os métodos adequados tem-se um PDF sendo exibido no navegador e com a opção de ser gravado na máquina do usuário.

2.2.2.9. PHP

O manual do PHP descreve o com sendo um acrônimo recursivo para PHP: (*Hypertext Preprocessor*) é uma linguagem de script *open source* de uso geral, muito utilizada, e especialmente adequada para o desenvolvimento web e que pode ser embutida dentro do HTML.

O manual enfatiza que o que distingue o PHP de algo como o *JavaScript* no lado do cliente é o código que é executado no servidor, gerando o HTML e, assim, enviado para o navegador. O navegador recebe os resultados da execução desse script, mas não sabe qual era o código fonte. Você pode inclusive configurar seu servidor web para processar todos os seus arquivos HTML com o PHP, e então não há como os usuários dizerem o que você tem na sua manga. Uma melhor opção para usar o PHP é que ele é extremamente simples para um iniciante, mas oferece muitos recursos avançados para um programador profissional.

O uso do PHP combinado com o HTML permite a utilização de diversos recursos nas páginas *Web*, tornando-se fundamental para o desenvolvimento do sistema. Essa combinação entre HTML e PHP é muito útil (Niederauer, 2008), possibilitando exibir dados no navegador de forma mais dinâmica e precisa.

2.2.2.10. SQL

Para a manipulação dos dados armazenados no banco de dados utiliza-se a linguagem SQL, por meio de instruções SQL. Setzer e Silva (2005) define SQL – Structured Query Language, não é uma linguagem de programação e sim de declaração e acesso a tabelas. Essas instruções permitem realizar consultas, alterar, deletar e inserir informações como registros no Banco de dados criado também pela própria linguagem. Uma característica fundamental é a consulta em diversas tabelas do banco apresentando melhores resultados nas buscas possibilitando consultas poderosas, outro ponto importante é a fácil interligação com aplicações Web permitindo uma maior interação entre os serviços cliente-servidor, nesse processo de interação o cliente solicita as consultas e o servidor realiza as instruções no banco.

Os comandos mais comuns para manipulação de dados são: (SELECT, INSERT, DELETE e UPDATE), que pode ser integrado as cláusulas (FROM) e de filtro (WHERE). Seguidos da ordenação, (ORDER BY) ou agrupamento (GROUP

BY), através dos comandos citados o administrador do banco pode realizar manutenção no banco de dados. Para Beighley (2010, p.6) “um banco de dados é o contêiner que guarda todas as tabelas e outras estruturas SQL relacionadas àquelas tabelas”. Os comandos SQL podem manipular objetos em diferentes classes ou funções: Linguagem de Definição de Dados (DDL); Linguagem de Manipulação de Dados (DML) e outras (MACHADO E ABREU, 2009).

Existem diversos sistemas de gestão de base de dados que utilizam a SQL, ainda conforme Beighley (2010, p.22) que cita “Standard SQL, MYSQL, Oracle, MS SQL SERVER, PostgreSQL, DB2 e mais uma porção de sistemas[...]”, o autor mostra que a linguagem SQL possui suporte por diversos sistemas de gerenciamento de BD, desde ferramentas gratuitas até as pagas, possibilitando aos administradores uma quantia variada de opções de gerenciadores.

2.3. Utilização de software / heurísticas de usabilidade

Várias posições surgem em torno do conceito de usabilidade. Por exemplo, Shackell (1986) considera quatro parâmetros para medir a usabilidade: eficiência, aprendizagem, flexibilidade e atitude do utilizador. Para Hix e Hartson (1993), a usabilidade está relacionada com a eficácia e a eficiência da interface e com a reação do utilizador à interface.

Como descrito pela Revistabw (2015), usabilidade define a facilidade com que um usuário (humano ou não) pode utilizar uma determinada ferramenta ou objeto. Dentro da disciplina Interface Homem-Máquina, o termo designa a capacidade de um usuário utilizar, de forma satisfatória e eficiente, uma interface de hardware ou software para atingir seus objetivos inicialmente propostos.

Segundo Cybis (2007), uma avaliação heurística representa um julgamento de valor sobre as qualidades ergonômicas das Interfaces Humano Computador. Essa avaliação é realizada por especialistas em ergonomia, com base em sua experiência e competência no assunto. Eles examinam o sistema interativo e diagnosticam os problemas ou as barreiras que os usuários provavelmente encontrarão durante a interação.

Em Revistabw (2015), vamos encontrar o seguinte esclarecimento:

“A usabilidade é, em geral, desprezada pelos criadores de software em nome da rapidez de desenvolvimento ou de um menosprezo pelos usuários “menos capacitados” para utilizar um sistema. No entanto, um sistema que não é amigável ao usuário poderá gerar aumento do custo de operação e de treinamento de usuários, por exemplo.” (REVISTABW, 2015).

Existem diversas recomendações propostas para garantir a usabilidade de um sistema. Nielsen (1993), no livro *Engenharia de Usabilidade* do inglês *Usability Engineering* propôs 10 heurísticas de usabilidade que um sistema deve garantir. As heurísticas de Nielsen são largamente utilizadas por organizações em todo o mundo, principalmente para construção de interfaces de software e websites.

Assim sendo, a baixo será apresentada e descrita as 10 heurística proposta por Nielsen (1993).

- a) **Visibilidade de Status do Sistema:** Esta heurística estará sendo atendida, quando o usuário deverá estar livre para ir e voltar dentro das “telas” do sistema sem ser “travado” pelo mesmo. O sistema deverá, ativamente, exibir mensagens (textuais, audíveis, caixas de mensagens, etc.) para constantemente informar o usuário o que está ocorrendo, não deixando que este se sinta preso ou perca o foco do que está fazendo.
- b) **Relacionamento entre a interface do sistema e o mundo real.** Esta heurística estará sendo atendida se o sistema não utilizar linguagens exageradamente técnicas, mas, sim, conversar com o usuário em uma linguagem que permita seu entendimento de modo confortável. O sistema deve agir como um educado e bom amigo!
- c) **Liberdade e controle do usuário:** Esta heurística estará sendo atendida, caso o usuário tenha liberdade para cancelar uma tarefa sempre que quiser. Assim, em casos de processamentos que não possam ser cancelados até que sejam finalizados por terem chegado ao seu final, o sistema deverá explicar o porquê desta impossibilidade temporária de cancelamento.
- d) **Consistência:** Esta heurística estará sendo atendida se o sistema apresentar uniformidade na exibição de padrões de menus, ícones, textos e os botões e outros comandos do sistema estejam sempre localizados no mesmo local e sempre executem da mesma forma. O *Look and feel* (aparência das janelas, esquema de cores, formato e cores dos botões e demais controles visuais) da aplicação deverá seguir o mesmo padrão do sistema operacional onde a aplicação está sendo executada.
- e) **Prevenção de erros:** Esta heurística estará sendo atendida, se o sistema conseguir evitar que erros básicos sejam executados pelo usuário. Um erro, aqui, é uma ação que não colabora para a execução da função intencionada na mente do usuário. Ela pensa que está fazendo X, quando, na realidade, faz Y,

dentro de um determinado sistema. O fato de o usuário executar Y poderá, não raro, colocar o sistema em um estado não adequado e mesmo inconsistente.

- f) **Reconhecimento ao invés de lembrança:** Esta heurística estará sendo atendida, caso o sistema faça uso de elementos e controles interativos (avisos, gráficos, textos e sons comuns dos paradigmas internacionais atuais) que permitam ao usuário deduzir os comandos que serão realizados por meio deles, o que remove uma sobrecarga na memória do usuário.
- g) **Flexibilidade e eficiência de uso:** Esta heurística estará sendo atendida, caso um sistema proveja mais de uma maneira de se realizar as suas funcionalidades, ou seja, o usuário comum poderá proceder no caminho padrão para realização de uma tarefa (Ex: Para imprimir um documento do Microsoft Word, Clique em Arquivo depois Clique em Imprimir, Clicar em IMPRIMIR) ou outro caminho alternativo (Ex: Tecla de Atalho CTRL + P, Clicar em IMPRIMIR).
- h) **Estética e design minimalista:** Esta heurística estará sendo atendida, caso determinado sistema, em sua interface, exiba elementos gráficos e textuais simples o suficiente para alcançar os objetivos pretendidos, nada além nem aquém disso.
- i) **Ajude os usuários a reconhecer, diagnosticar e sanar erros:** Esta heurística estará sendo atendida, caso uma aplicação não exiba mensagens de erros cujos textos sejam verdadeiras sentenças amedrontadoras, mas, sim, que sejam explicações sobre o que poderá ter ocorrido e como o usuário poderá proceder para resolver o erro em questão.
- j) **Ajuda e documentação:** Esta heurística estará sendo atendida, caso um sistema possua interface tão eficiente e de fácil utilização que permita ao usuário realizar várias ações sem ampla necessidade de acessar manuais e ajudas externas. Ao mesmo, tempo, embora o sistema tenha estas características, ele deverá possuir boa documentação, manuais de instruções de uso para, caso o usuário tenha necessidade de consultar, esteja à sua disposição.

Revistabw (2015) esclarece, que existem outras recomendações de outros autores, heurísticas de outros especialistas, afinal, “não só de Nielsen que viverá o Engenheiro de Usabilidade. ”

SHNEIDERMAN e PLAISANT (2004, *Apud* Cybis, 2007), descrevem suas heurísticas perseguir a consistência, Fornecer Atalhos, fornecer, feedback informativo, marcar o final dos diálogos, fornece prevenção e manipulação simples de erros, permitir o cancelamento das ações, fornece controle e iniciativa ao usuário, reduzir a carga de memória de trabalho.

A Norma ISO 9241:10 (1998, *Apud* Cybis, 2007), lista os seguintes princípios de avaliação de interfaces, conhecidos como “Princípios de Diálogo”: Adaptação à tarefa; Autodescrição (feedback); Controle ao usuário; Conformidade às expectativas do usuário; Tolerância aos erros; Facilidade de individualização; Facilidade de aprendizagem.

2.4. Educação a Distância

Esta seção apresenta uma abordagem geral sobre Educação a Distância EaD, o Portal do Ministério da Educação afirma que a Educação a Distância é a modalidade educacional na qual alunos e professores estão separados, física ou temporalmente e, por isso, faz-se necessária a utilização de meios e tecnologias de informação e comunicação.

Essa modalidade é regulada por uma legislação específica e pode ser implantada na educação básica (educação de jovens e adultos, educação profissional técnica de nível médio) e na educação superior.

Antes da modalidade Reis, Almeida e Correia (2018, p. 421), argumenta que “a educação se restringia somente ao ambiente da sala de aula, somente com a utilização dos recursos do quadro-negro e giz”.

Já na ótica de MUGNOL (2009), a Educação a Distância ao longo de sua história traçou uma trajetória de avanços e retrocessos. No início do século XX, tornou-se uma modalidade de ensino capaz de atender a todos os níveis, incluindo programas formais de ensino, aqueles que oferecem diplomas ou certificados e programas de caráter não formais, cujo objetivo é oferecer capacitação para a melhoria no desenvolvimento das atividades profissionais.

Segundo Paulsen (2002), a modalidade de ensino à distância, baseada na Web, caracteriza-se pela separação física entre o docente e o discente; utilização de uma rede computadorizada para apresentar os conteúdos; e comunicação bidirecional para que os discentes se comuniquem entre si e com o docente.

Nessa perspectiva, Maciel (2018) descreve que na modalidade de EaD o distanciamento geográfico e o isolamento relativo do aprendiz – características da

modalidade – podem ampliar a distância entre os professores e aprendizes e gerar a tão frequente evasão. Por outro lado, estudos sinalizam que o ambiente virtual pode gerar aproximações e uma convivência tão próxima quanto a presencial, conforme afirma Maraschi (2000) que as tecnologias na realidade têm a função de eliminar a distância ou construir outras interações diferentes da presencial. Ao possibilitarem a suspensão das distâncias espaciais permitem a convivência e a diversidade identitária.

Morer (2008) defende que, na verdade, a EaD bem-sucedida reside no equilíbrio do incremento à qualidade na educação e no *design* instrucional: ou seja, a direção da educação versus a direção da tecnologia. Não é possível considerar um e desdenhar o outro. A inovação em EaD a ser questionada conduz ao fato de que o sistema deve ser aberto, bem desenhado em relação ao design, adaptativo e flexível, com menos barreiras e mais inclusão,

Nessa perspectiva Ehlers (2008). Enfatiza como promover acesso a todos, melhorar o uso de habilidades e promover qualidade na usabilidade? Uma sociedade que promove inovações e desenvolve-se por meio de estudo deve ter todos juntos. Não perder as mentes criativas daqueles que não têm acesso à educação, à tecnologia, ao estudo e a infraestrutura do conhecimento. *E-learning* para todos requer romper, derrubar as barreiras para aqueles que não têm acesso às novas infraestruturas do conhecimento. E abrir a aproximação à inclusão envolve abrir um convite a todos os grupos (*stakeholder groups*) em direção ao diálogo de como as barreiras da motivação, tecnologia, da pedagogia e de acesso podem ser vencidas.

Assim, as plataformas de ambientes virtuais a distância adquire real importância, pois o sucesso da EaD que elas buscam contemplar está visceralmente ligado à sua construção. Neste processo é importante que o aluno alcance a produção de conhecimento significativo, onde o conhecimento se incorpore em seu mundo intelectual e vivencial.

2.5. Trabalhos correlacionados

Esta seção contém uma síntese de sistemas que realizam gerenciamento de igrejas.

2.5.1. Sistema para igrejas evangélicas multicongregadas

Rodrigues (2015), desenvolveu um trabalho que teve como objetivo modelar um sistema para gerenciamento de igrejas evangélicas multicongregadas,

abrangendo as rotinas executadas nesse tipo de entidade. Em complemento à modelagem, algumas telas de cadastros do módulo "Secretaria" foram desenvolvidas. Esses controles realizam as operações básicas (inclusão, atualização, exclusão e consulta) sobre as entidades: cidades, regionais congregações e pessoas presentes.

Este trabalho não houve pesquisa a fim de avaliar a usabilidade do software e aceitação do público alvo.

2.5.2. Sistema de Gerenciamento de Igrejas (SGI)

O Sistema de Gerenciamento de Igrejas (SGI), desenvolvido por Silva (2011), teve como objetivo a contribuição no processo de agilizar as tarefas diárias para o controle das Igrejas do Evangelho Quadrangular de Borá – SP. Para atender ao objetivo estabelecido o sistema deve controlar o cadastro de Igrejas, Membros, Cargos, entradas de Dízimos e Ofertas e Contas à Pagar. Além desses controles, o SGI também emitirá relatórios gerenciais com o intuito de auxiliar os gestores na tomada de decisão.

O SGI foi desenvolvido para funcionar em ambiente Desktop, não contempla módulo de escola bíblicas e também não houve pesquisa a fim de avaliar a usabilidade do software e aceitação do público alvo.

2.6. Comentários Parciais

Este capítulo discutiu informações sobre elementos que proporcionaram o suporte conceitual necessário para a elaboração do trabalho, os elementos abordados foram: Instituição Religiosa e seus desafios de gestão, desenvolvimento de software web, Utilização de Software / heurísticas de usabilidade e Educação a Distância.

O estudo realizado nesse capítulo deu embasamento teórico para o desenvolvimento do software para Web proposto, este *software* é apresentado no capítulo seguinte. No final do estudo foi realizada uma busca por trabalhos correlacionados, que realizam gerenciamento de igrejas por meio da *Web*. Com base nessa busca foi possível verificar qual o diferencial que o presente software deve oferecer e as ferramentas e tecnologias a serem utilizadas na criação do sistema.

3. PROCEDIMENTOS METODOLÓGICOS

Este capítulo apresenta a metodologia científica adotada neste trabalho, de forma a sistematizar a revisão bibliográfica para descrever sobre o desenvolvimento de software web, gestão de igrejas e Ensino a Distância, seguida de como foi realizada a pesquisa de avaliação do produto gerado com esse projeto. E por fim, o modelo de engenharia de *software* utilizado para o desenvolvimento e documentação deste projeto, descrevendo como ocorreu todo o desenvolvimento e execução do projeto; o modelo de desenvolvimento, a análise de requisitos, UML Linguagem de Modelagem Unificada (do inglês, UML - Unified Modeling Language), banco e outros.

3.1. Materiais e Métodos

O levantamento teórico foi realizado por meio de uma pesquisa bibliográfica, uma vez que o pesquisador busca embasamento na literatura existente, é feita a partir do levantamento de referências teóricas já analisadas, e publicadas por meios escritos e eletrônicos, como livros, artigos científicos, páginas de web sites. Na opinião de Fonseca, (2002, p. 32). “Qualquer trabalho científico inicia-se com uma pesquisa bibliográfica, que permite ao pesquisador conhecer o que já se estudou sobre o assunto”. Existem, porém, pesquisas científicas que se baseiam unicamente na pesquisa bibliográfica, procurando referências teóricas publicadas com o objetivo de recolher informações ou conhecimentos prévios sobre o problema a respeito do qual se procura a resposta.

Gil (2007), afirma que os exemplos mais característicos desse tipo de pesquisa são sobre investigações sobre ideologias ou aquelas que se propõem à análise das diversas posições acerca de um problema.

Em Salomon (2004), é esclarecido que a pesquisa bibliográfica se fundamenta em conhecimentos proporcionados pela Biblioteconomia e Documentação, entre outras ciências e técnicas empregadas de forma metódica envolvendo a identificação, localização e obtenção da informação, fichamento e redação do trabalho científico. Esse processo solicita uma busca planejada de informações bibliográficas para elaborar e documentar um trabalho de pesquisa científica.

Para a consecução do presente trabalho foi adotado uma classificação de pesquisa de campo, conforme Fonseca (2002), caracteriza-se pelas investigações em que, além da pesquisa bibliográfica e/ou documental, se realiza coleta de dados junto a pessoas, com o recurso de diferentes tipos de pesquisa (pesquisa *ex-post-facto*, pesquisa-ação, pesquisa participante, etc.)

Este estudo apresenta o enfoque do método de abordagem qualitativa, que segundo Marconi e Lakatos (1993, p8).

“Método é o conjunto das atividades sistemáticas e racionais que com maior segurança e economia, permite alcançar o objetivo – conhecimento válido e verdadeiro – traçando o caminho a ser seguido, detectando erros e auxiliando as decisões do cientista” (MARCONI E LAKATOS, 1993, P8).

Embora a pesquisa utilize o método qualitativo para analisar os dados obtidos com a avaliação do *software*, vale ressaltar que “mesmo pesquisas com dados qualitativos podem ter um tratamento matemático, especialmente usando-se de análises estatísticas” (ABRANTES et al, 2007, p.13). Desta forma, fez-se o uso dos métodos estatísticos, tendo em vista a necessidade de tornar quantificável algumas informações do questionário.

A escolha do universo da pesquisa em que o estudo foi desenvolvido está definido com 72 participantes, deste total, 52 são integrantes da Igreja Assembleia de Deus, Ministério da Salvação em Cristo, sendo: 3 pastores, 1 secretário, 1 tesoureiro, 37 membros e 16 congregados, o restante, definido como outros no formulário de pesquisa, são 14 técnicos com conhecimento em desenvolvimento de *software*. O instrumento de coleta de dados escolhido foi a aplicação de um questionário confeccionado para ser aplicados via web aos entrevistados, como descrito no subitem 3.3.1 – Confeção do Questionário da Pesquisa.

O questionário é um instrumento ou programa de coleta de dados que segundo Carnevalli e Miguel, (2001), pode ser enviado via correio, fax, Internet, etc., o questionário pode ser estruturado não disfarçado: o respondente sabe qual é o objetivo da pesquisa, e o questionário é padronizado, usando principalmente questões fechadas. A confecção é feita pelo pesquisador; o preenchimento é realizado pelo informante. A linguagem utilizada no questionário deve ser simples e direta para que o interrogado compreenda com clareza o que está sendo perguntado (KAUARK, MANHÃES E MEDEIROS, 2010).

O questionário foi confeccionado utilizando a plataforma do google, formulário *Google Form* e utilizado como instrumento de coleta de dados, os links foram

enviados via e-mail, seguido de uma nota explicativa, onde, informava a natureza da pesquisa e de cada instrumento aplicado. Pois como afirma Marconi e Lakatos (2003, p. 201):

“Junto com o questionário deve-se enviar uma nota ou carta explicando a natureza da pesquisa, sua importância e a necessidade de obter respostas, tentando despertar o interesse do recebedor, no sentido de que ele preencha e devolva o questionário dentro de um prazo razoável” (MARCONI E LAKATOS, 1993, P8).

Contudo, explicar os objetivos da pesquisa na nota, solicitando colaboração aumenta a influência no retorno dos questionários, pois conforme as autoras anteriormente citadas, deve se despertar o interesse do pesquisado para que ele preencha e devolva o formulário do mesmo modo que foi enviado.

Em primeiro momento as coletas dos dados foram feitas em planilhas eletrônicas do pacote *Google Docs*, no segundo momento os dados foram salvos em planilhas do Microsoft Excel para facilitar a análise e tabulação dos dados, em terceiro momento foi aplicado os métodos estatísticos para a tabulação dos dados. E por fim, os resultados foram impressos em gráficos gerados pela ferramenta utilizada, para serem apresentados no capítulo 4 Análises e Resultados dos Dados.

3.2. Desenvolvimento do Software

Neste tópico está descrito as etapas do desenvolvimento do sistema SGE – Sistema de Gestão Evangélica, definindo os modelos de engenharia de software que garantem a consistência e qualidade no desenvolvimento da aplicação. Após a definição dos modelos de engenharia, foi possível elaborar as análises para a especificação dos requisitos e a criação dos diagramas, possibilitando a implementação dos códigos, tornando reais as funcionalidades do sistema.

3.2.1. Descrição do Sistema

O sistema computacional implementado como resultado deste trabalho é uma plataforma para gerenciamento de igrejas em ambiente *cloud* que permite ao usuário utilizar toda sua estrutura através de uma interface *web* amigável, para controle de membros, relatórios, lançamentos financeiros, controle de Escola Bíblica dominical, cursos teológicos dentre outros. As vantagens oferecidas são: Facilidade, agilidade, assertividade e maior produtividade; Substituição dos materiais impressos manuais por informações digitais; Melhor organização das informações cadastradas.

A plataforma possui um site inicial com informações sobre a igreja, já o sistema SGE possui três níveis de acesso Administrador, secretário e tesoureiro,

possibilitará o cadastro de campus, igrejas, membros, permite ainda o lançamento do financeiro entrada e saída por categoria, controle de usuários e emissão de diversos tipos de certificados e relatórios para impressão em arquivo com extensão PDF. O sistema está disponível no seguinte endereço: <http://www.adesc.dx.am/>

O processo de desenvolvimento do sistema foi realizado em 4 etapas que são descritas a seguir:

Início: A concepção do sistema se deu junto aos líderes da igreja Assembleia de Deus Ministério da Salvação em Cristo, levantando os requisitos baseados nas finalidades do *software* proposto, e posteriormente analisadas junto ao professor orientador. Nesta fase foram produzidos os diagramas que deram base no desenvolvimento da aplicação.

Elaboração: Esta fase realizou-se a seleção das tecnologias, e recursos a serem utilizados no desenvolvimento do *software*, aprimorando as documentações produzidas na fase anterior por meio das revisões das regras de negócios aplicadas a gestão de igrejas e EaD.

Construção: com os requisitos criados, as regras de negócios definidas, nesta fase foram implementados o código fonte do sistema, realizados alguns testes com o intuito de garantir a qualidade do produto e realizar correções necessárias.

Avaliação: a avaliação foi desenvolvida por meio de testes das funcionalidades, os testes foram realizados pelo público do universo da pesquisa desenvolvida neste projeto. Para possibilitar a realização da pesquisa o *software* foi implantando em servidores na *Web* utilizando domínios gratuitos e para finalizar foi realizada uma revisão final do sistema com a base nos resultados da pesquisa.

3.2.2. Análise e levantamento de Requisitos do Sistema

A análise dos requisitos está ligada diretamente com a fase inicial deste projeto, através da identificação dos requisitos envolvidos podemos elaborar os diagramas do sistema relacionando as funcionalidades e os relacionamentos entre os usuários, ou seja, definir quais as funcionalidades do sistema e quais restrições o sistema deverá possuir. Assim definiu-se o ambiente *Web*, a engenharia a ser aplicada, a linguagem e ferramentas de desenvolvimento e qual gerenciador de banco de dados a ser utilizado.

A apresentação dos requisitos levantados permite entender o sistema, especificar as funcionalidades que se espera que o sistema tenha (CAMPOS, 2018). O sistema SGE conta com 3 níveis de acesso e o ou módulos do sistema como

apresentado no subtópico 3.2.4.2. Módulos e telas do Sistema, essa divisão foi necessária para um gerenciamento mais eficiente das informações. A seguir é apresentado os requisitos funcionais e respectivamente os requisitos não funcionais.

3.2.2.1. Requisitos Funcionais

A tabela 3 apresenta os requisitos funcionais previstos para o sistema, ou seja, vinculação com os casos de uso associados.

Tabela 3 - Requisitos Funcionais

ID	NOME	DESCRIÇÃO	PRIORIDADE
RF01	Cadastro	O sistema deve permitir cadastrar, alterar, deletar e visualizar: campos, igrejas, membros, categoria, movimento, alunos e 4 tipos de usuário (administrador, secretário, tesoureiro e aluno).	Essencial
RF02	Relatório	O sistema deve permitir emissão de relatórios de campos, igrejas, membros e financeiros.	Importante
RF03	Curso	O sistema deve permitir cadastro de cursos	Importante
RF04	Matrícula	O sistema deve permitir matrículas de alunos em cursos	Essencial
RF05	Prova	O sistema deve gerar prova	Importante
RF06	Resultado	O sistema deve computar e gerar o resultado percentual da prova	Essencial
RF07	Certificado	O sistema deve permitir a emissão e impressão de certificado	Importante
RF08	Login	O sistema deve permitir usuários cadastrados realizar o login	Essencial
RF09	Senha	O sistema deve permitir o envio de senha para o e-mail do usuário	Importante

Fonte: Santos (2018), produção próprio autor.

3.2.2.2. Requisitos Não Funcionais

A tabela 4 apresenta os requisitos não funcionais previstos para o sistema.

Tabela 4 – Requisitos Não Funcionais

ID	NOME	DESCRIÇÃO	PRIORIDADE
NF01	Interface	O sistema deverá ter uma interface amigável, responsiva, de fácil acesso e aprendizagem;	Importante
NF02	Portabilidade	O sistema deverá ser operacionalizado em qualquer sistema operacional e funcionar em qualquer navegador <i>Web</i> .	Essencial
NF03	Segurança	3.1-Usuários logado em uma área não pode em hipótese alguma acessar outra área da qual não tenha permissão; 3.2 - A matrículas nos cursos só será com usuário autenticado; 3.3 - A base de dados deve ser protegida para acesso apenas de usuários autorizados.	Essencial
NF04	Confiabilidade	4.1- O usuário não poderá se inscrever duas vezes no mesmo curso; 4.2- Uma igreja não pode pertencer a mais de um campo; 4.3- Um membro não pode pertencer a mais de uma igreja.	Essencial
NF05	Desempenho	5.2 - O tempo de resposta do sistema não deve ultrapassar 30 segundos.	Importante
NF06	Reusabilidade	6.1 Os módulos do sistema devem ser reusados	Importante
NF07	Logica	O sistema deve ser implementado em PHP com banco de dados SQL.	importante

Fonte: Santos (2018), produção próprio autor.

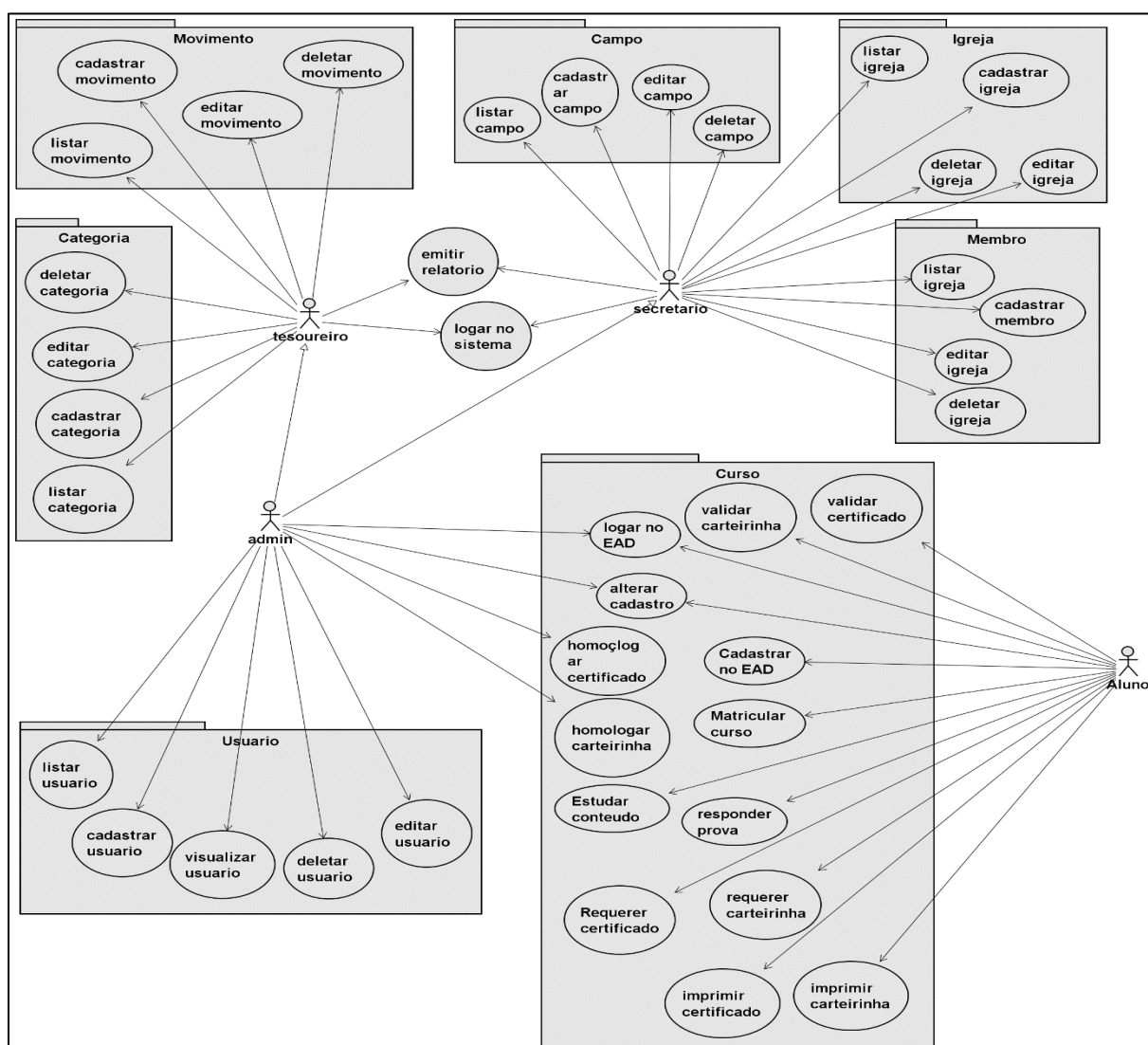
3.2.3. Análise do sistema SGE

Nessa etapa do desenvolvimento foram modeladas as UML (*Unified Modeling Language*), Oliveira (2018) afirma que esta é uma linguagem de propósito geral para modelagem de software ou os diagramas de interação do sistema, ou diagramas de interação do sistema, a elaboração dos diagramas foi realizada com base na análise dos requisitos levantados na etapa anterior. Os diagramas criados foram o de Caso de Uso, o diagrama de Classe, diagrama de Sequência e o diagrama de banco. A elaboração da UML permite que desenvolvedores padronizem e visualizem seus projetos, permitindo apresentar o sistema de forma clara antes da sua codificação.

3.2.3.1. Diagrama Caso de Uso

No diagrama de caso de uso que é um diagrama UML que representa as funcionalidades do sistema, contendo as funcionalidades essenciais realizadas pelos seus atores (GUEDES, 2018). Esses atores são: os alunos, o administrador, o secretário e o tesoureiro, o administrador é o responsável por cadastrar e gerenciar novos usuários, tendo acesso a todas funcionalidades do sistema; o secretário é o responsável por cadastrar e gerenciar campos, igrejas e membros; o tesoureiro gerencia toda tesouraria e por último os alunos irão se auto cadastrar, matricular em cursos realizar provas, requerer emissão de certificados, carteirinha e validá-los. Na Figura 2 tem-se o diagrama de caso de uso do *software*.

Figura 2 - Diagrama de caso de uso



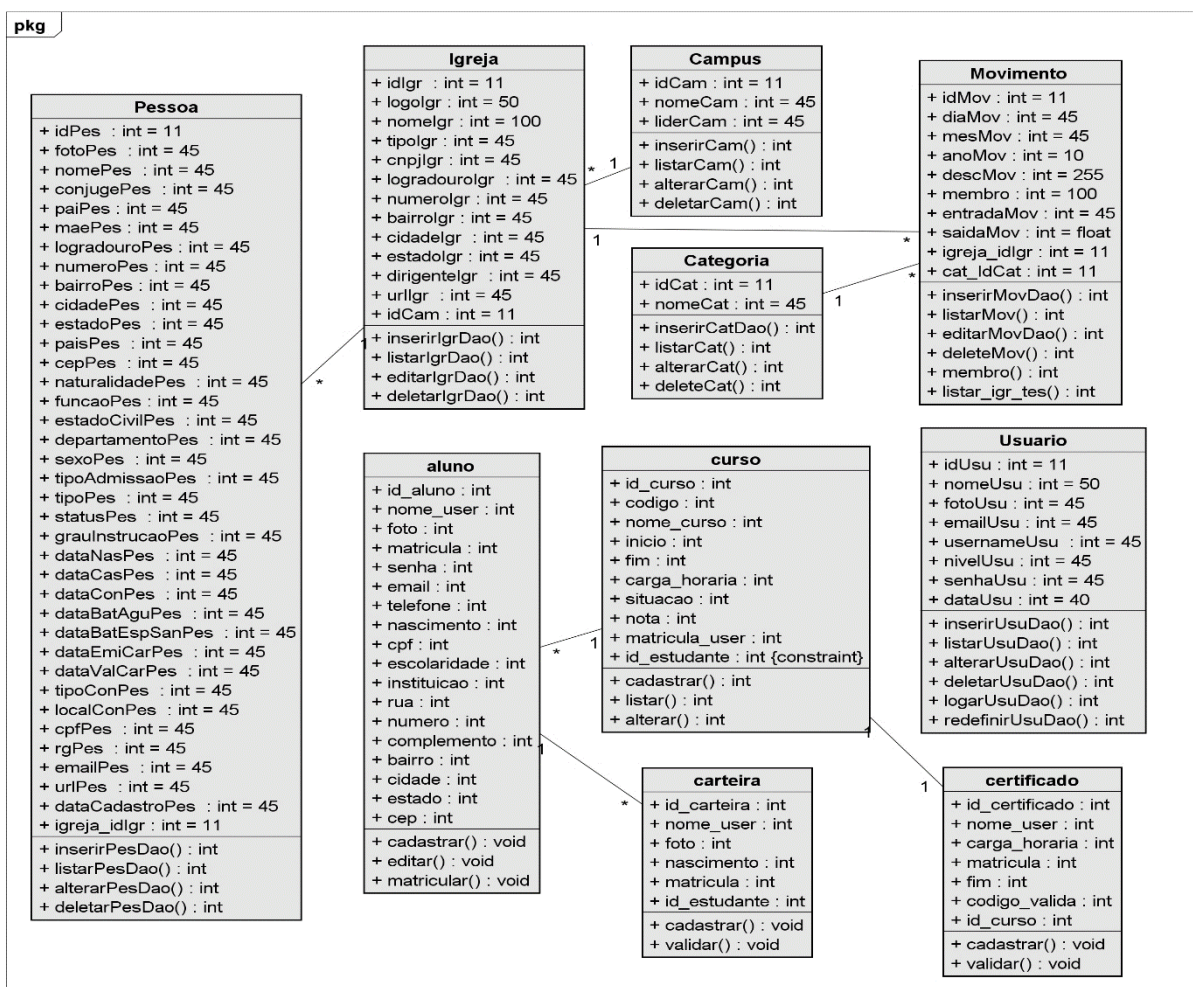
FONTE: Santos (2018), produção próprio autor

3.2.3.2. Diagrama de Classe

O diagrama de classe segundo Guedes (2018) é um dos mais importantes na elaboração das UML do software, pois representam os objetos do sistema, interações, comportamentos e relacionamentos. O diagrama de classe representa as interações de cada operação e métodos de cada objeto através do seu comportamento.

O diagrama ajuda a equipe envolvida entender o sistema, e possibilita outros profissionais que não estão envolvidos com o desenvolvimento do sistema, entender a estrutura do *software* e suas funcionalidades. Partindo da análise de requisitos foi criado o seguinte modelo de diagrama descrito na Figura 3.

Figura 3 - Diagrama de Classe

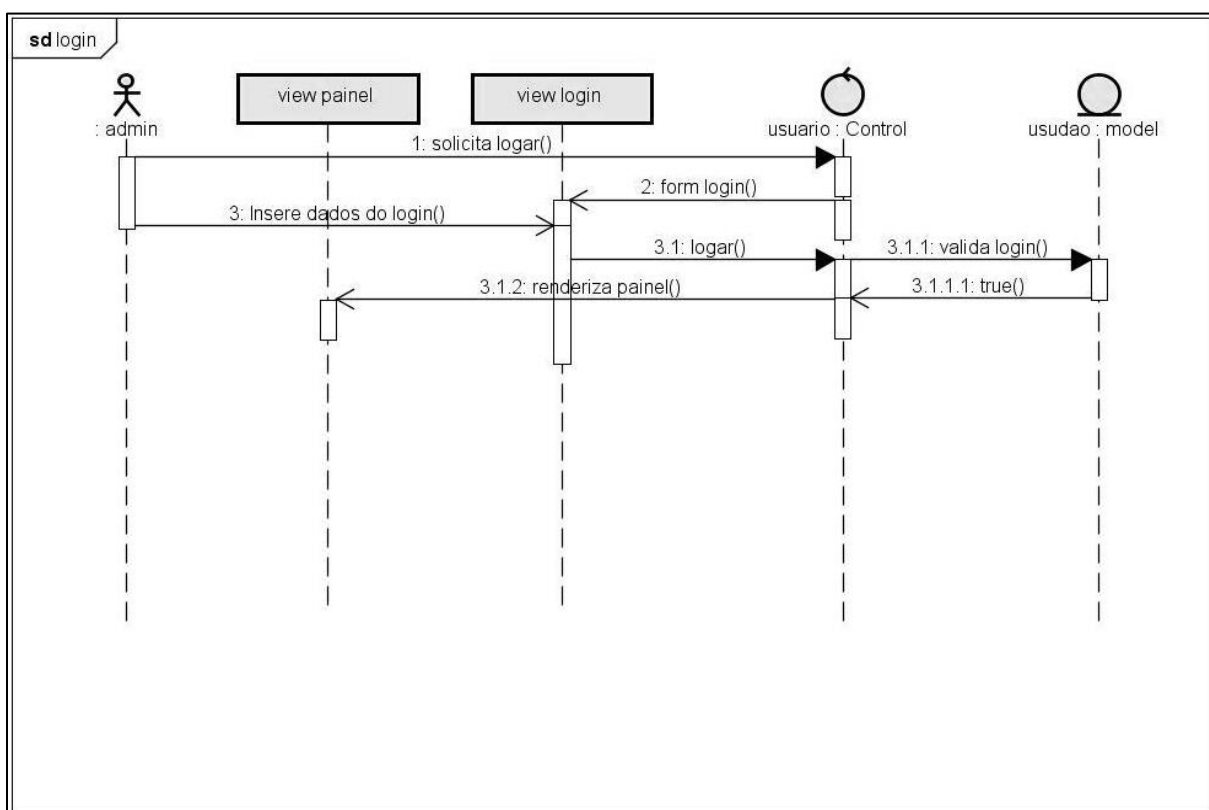


FONTE: Santos (2018), produção próprio autor

3.2.3.3. Diagrama de sequência

O diagrama de sequência segundo Guedes (2018) representa a comunicação entre os objetos do sistema, ou seja, a troca de mensagens enviadas de um objeto para o outro, os objetos são organizados um ao lado do outro, ligados entre-se por linhas que representam as mensagens de comunicação entre os objetos. A Figura 4 observa-se o diagrama de sequência ilustrando como é feito o login no *software*.

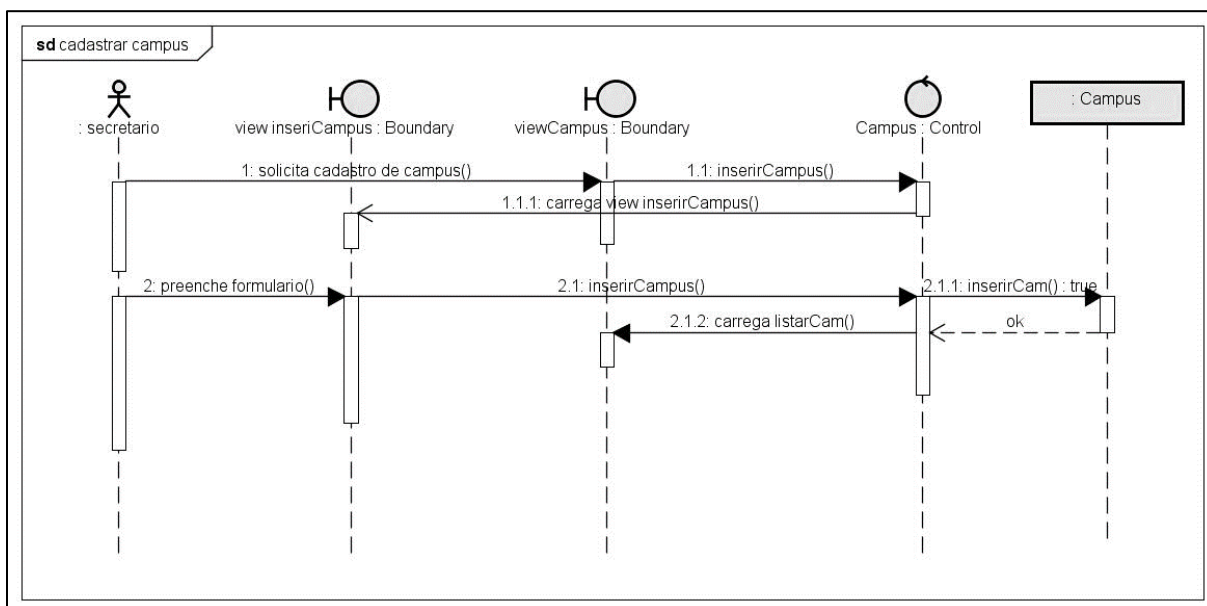
Figura 4 - Diagrama de sequência, representando o login



FONTE: Santos (2018), produção próprio autor.

O diagrama de sequência para efetuar o login pode ser lido da seguinte forma: 1) O usuário solicita o acesso ao sistema; 2) O usuário informa o nome de usuário e senha na página de login; 3) O sistema verifica na base de dados se os dados informados existem; 4) O usuário recebe acesso ao sistema de acordo sua permissão. No próximo diagrama, figura 5 é descrito uma situação que o secretário realiza o cadastro de campo.

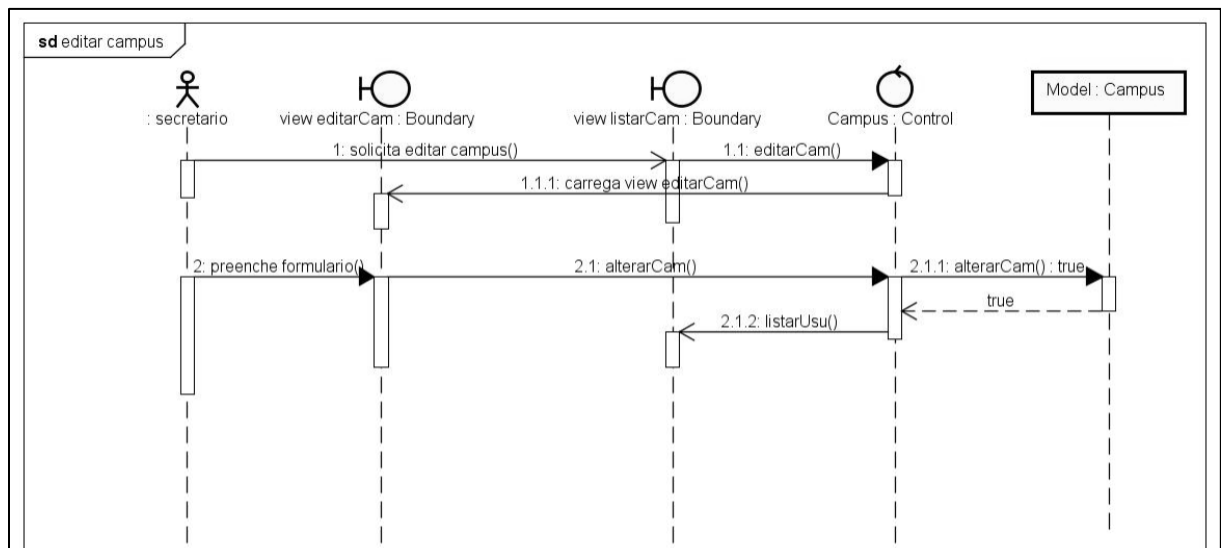
Figura 5 - Diagrama de Sequência representando cadastro de campo



FONTE: Santos (2018), produção próprio autor.

O diagrama de sequência para efetuar o cadastro de campo pode ser lido da seguinte forma: 1) o usuário secretário solicita cadastro de campus; 1.1) Aciona o controller Campus onde contém o método inserirCampus; 1.1.1) Renderiza a view inseriCampus, nesta contém o formulário de cadastro; 2) São preenchido todos os dados no formulário, após é realizada a submissão do formulário; 2.1) Os dados são recebidos pelo controller Campus, através do método inserirCampus onde a model CampusDao é instanciada; 2.1.1) No método inserirCam da model CampusDao é realizada a persistência dos dados no banco; 2.1.2) O controller Campus é acionado novamente através do método que renderiza a view listarCam. A seguir observa-se o diagrama de sequência que representa a edição de dados do campus na figura 6.

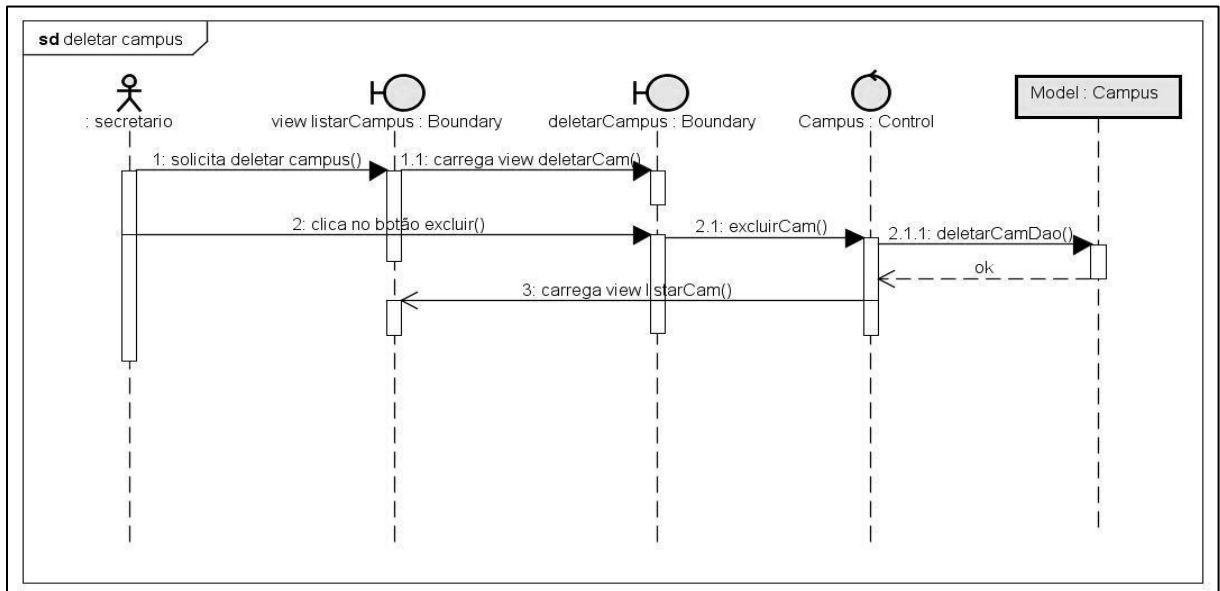
Figura 6 - Diagrama de sequência de edição de dados do campus



FONTE: Santos (2018), produção próprio autor

O diagrama de sequência para efetuar a edição de campus pode ser lido da seguinte forma: 1) o usuário secretário solicita a alteração de dados do campus; 1.1) Aciona método editarCam do controller Campus; 1.1.1) Renderiza a view editarCam; nesta contém o formulário de edição; 2) São preenchidos os dados a serem alterados, após é realizada a submissão do formulário; 2.1) Os dados são recebidos pelo controller Campus, através do método editarCam, onde a model CampusDao é instanciada e apontado para o método desta; 2.1.1) O método alterarCam realiza a persistência dos dados no banco; 2.1.2) O controller Campus é acionado novamente e através do método editarCam a view listarCam é renderizada. A seguir observa-se o diagrama de sequência que representa a exclusão de dados do campus, figura 7.

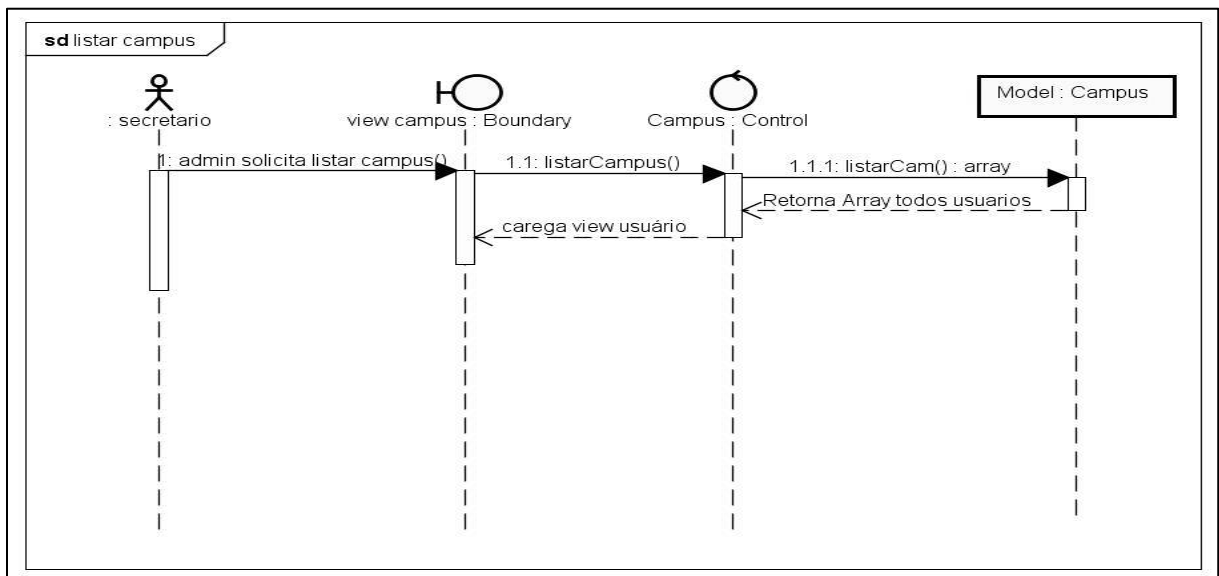
Figura 7 - Diagrama de sequência de deleção de dados do campus



FONTE: Santos (2018), produção próprio autor

O diagrama de sequência para efetuar a edição de campus pode ser lido da seguinte forma: 1) O usuário secretário solicita exclusão dos dados do campus; 1.1) O controller Campus que contém o método editarCam renderiza a view deletarCam; 2) Aciona-se botão referente ao campus a ser excluído; 2.1) O método excluirCam é acionado, a model CampusDao é instanciada; 2.1.1) O método deletarCam realiza a exclusão dos dados do campus do banco; 3) O controller Campus é acionado novamente e através do método listarCam a view listarCam é renderizada. A seguir observa-se o diagrama de sequência que representa a visualização de dados do campus, figura 8.

Figura 8 - Diagrama de sequência de lista de dados do campus



FONTE: Santos (2018), produção próprio autor

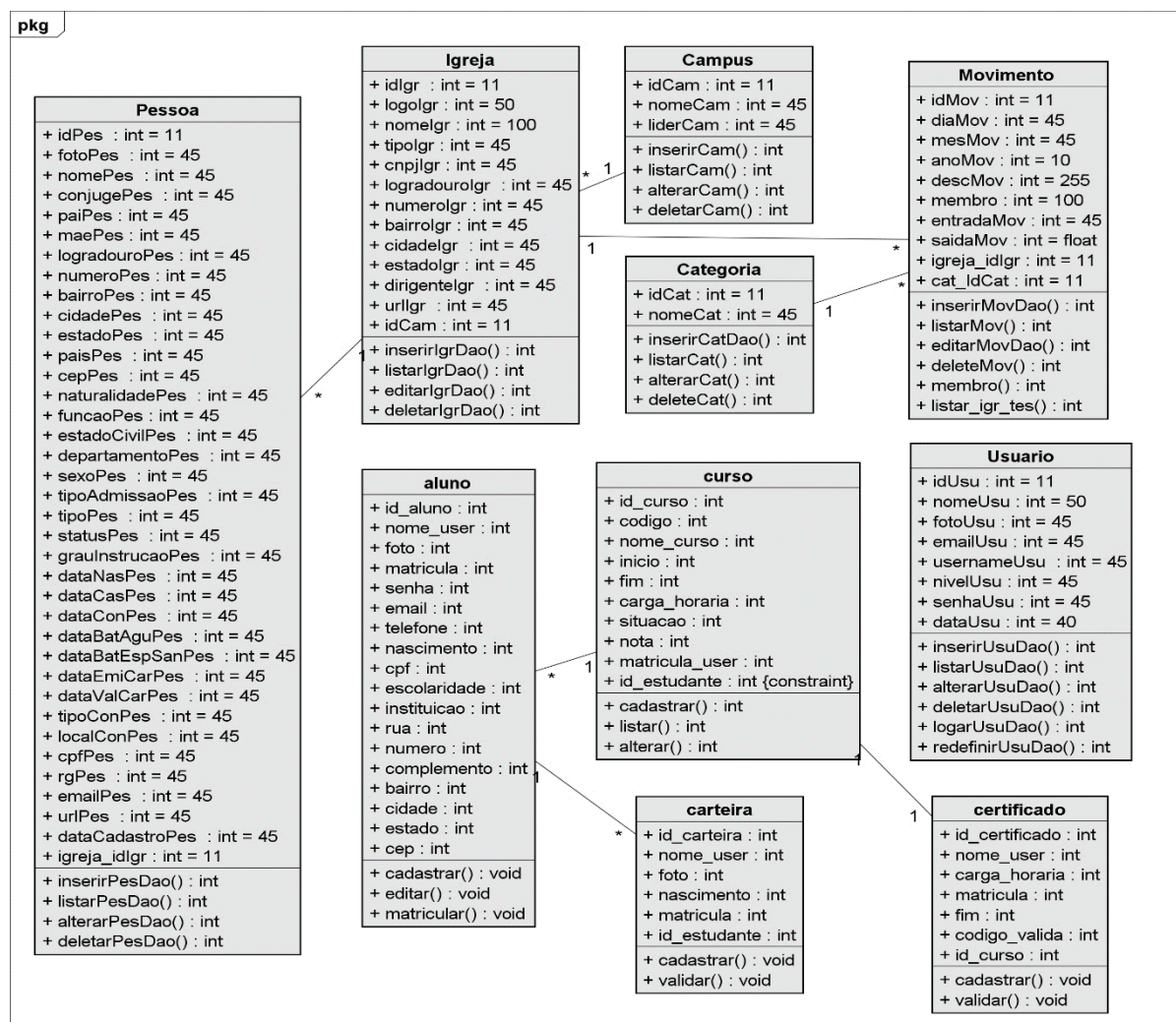
O diagrama de sequência para efetuar a edição de campus pode ser lido da seguinte forma: 1) O usuário solicita a lista de campus cadastrados, 1.1) O método listarCampus do controler Campus é acionado, a classe CampusDao é instanciada e apontado para o método listarCam; 1.1.1) Na classe CampusDao o método listarCam é responsável pela realização da consulta no banco este retorna um array de todos os campus do banco, devolve para o método listarCampus no controller Campus, que renderiza a view campus.

Os diagramas de sequência de inserir, cadastrar, alterar e listar dos módulos de igreja, membro, categoria, movimento, curso, alunos e usuários seguem a mesma lógica dos descritos anteriores. A seguir no tópico 3.2.3.4 observa-se o diagrama de banco de dados.

3.2.3.4. Diagrama de Modelagem Banco de Dados

A especificação lógica do banco de dados foi criada com auxílio do diagrama entidade-relacionamento. Na figura 11 a seguir, encontra-se o modelo lógico utilizado para a criação da base de dados do SGE.

Figura 9 - Diagrama banco de dados



FONTE: Santos (2018), produção próprio autor.

A função de cada tabela está descrita a seguir:

- Tabela usuário - possui os atributos referentes aos usuários do sistema;
- Tabela pessoa - possui os atributos referentes aos membros inseridos no sistema;
- Tabela igreja - possui os atributos referentes as igrejas cadastradas no sistema;
- Tabela campus - possui os atributos referentes ao cadastro de campus;
- Tabela categoria - possui os atributos referentes a categorias de movimento financeiro;
- Tabela movimento - possui os atributos referentes as entrada e saídas financeiras de cada igreja;

- g) Tabela aluno - possui os atributos referentes aos alunos do sistema EaD;
- h) Tabela curso - possui os atributos referentes aos cursos cadastrados no sistema EaD;
- i) Tabela carteira - possui os atributos referentes as carteirinhas solicitadas pelos alunos;
- j) Tabela certificado - que possui os atributos referentes aos certificados emitidos;

3.2.4. Organização do Sistema - SGE

Esta seção demonstra os detalhes da implementação do sistema, apresentando módulos e telas do sistema, bem como descreve as linguagens, ferramentas e tecnologias utilizadas na implementação.

3.2.4.1. Implementação

Neste tópico desta seção descreve o levantamento, definição das ferramentas, linguagem de programação e frameworks utilizados. Esta etapa se faz necessária para garantir a consistência e qualidade no desenvolvimento da aplicação. Na tabela 5 a seguir, apresenta as ferramentas e as tecnologias que foram utilizadas para modelar e implementar o sistema - SGE.

Tabela 5 - Ferramentas e as tecnologias utilizadas

Ferramenta / tecnologia	Versão	Referência	Finalidade
Wampserver	3.1.0	http://www.wampserver.com/en/	Servidor web local
Apache	2.4.27		Servidor HTTP
PHP	7.1.9		Linguagem de programação
MySQL	5.7.19	https://www.mysql.com/products/workbench/	Gerenciador de banco de dados
SQL	5.7.19		Linguagem de banco de dados
MySQL Workbench	6.3		Administrador de banco de dados
Bootstrap	4.1	https://getbootstrap.com/	Framework de desenvolvimento com HTML, CSS e JS.
Sublime Text	3.0	https://www.sublimetext.com/	Editor de código-fonte
Composer	1.7.2	https://getcomposer.org/	Gerenciador dependência para PHP
mPDF	7.0	https://github.com/mpdf/mpdf	Biblioteca PHP gerador arquivo PDF de HTML
Astha professional	7.2.0	http://astah.net/download	Ferramenta de modelagem UML

FONTE: Santos (2018), produção próprio autor.

Toda a codificação utilizada no desenvolvimento da aplicação foi salva em arquivos para melhor organização, os arquivos foram agrupados por pastas, classificadas por funcionalidades. A estrutura da organização pode ser visualizada na figura 10.

Figura 10 - Estrutura da Organização



FONTE: Santos (2018), produção próprio autor

A aplicação foi desenvolvida obedecendo o padrão de arquitetura de software *MVC*, separando o sistema em 3 camadas. A camada de interação do usuário (*view*), a camada de manipulação dos dados (*model*) e a camada de controle (*controller*).

A *model* é responsável pela leitura e escrita dos dados, e também de suas validações, a *view* é a camada de interação com o usuário, ela apenas faz a exibição dos dados por meio de arquivos com extensão *phtml*, já o *controller* é o responsável por receber todas as requisições do usuário, seus métodos chamados *actions* são responsáveis por uma página, controlando qual *model* usar e qual *view* será mostrado ao usuário.

Segundo a documentação da linguagem PHP, uma API (o termo, em inglês, é um acrônimo para *Application Programming Interface*, ou Interface para Programação de Aplicativos) tem a função de definir “classes, métodos, funções e variáveis que sua aplicação necessitará chamar para realizar a tarefa desejada”. Em PHP, as APIs para comunicação com bancos de dados são providenciadas, em

geral, por meio de extensões, ou seja, código cuja função é estender as funcionalidades programadas no núcleo da linguagem (OVERVIEW, 2014).

Para conexão com banco de dados MySQL, a linguagem oferece três APIs: as extensões MySQL, mysqli e PDO.

Para conexão de banco desse *software* utilizou-se a classe nativa do PHP a PDO, cujo nome é um acrônimo para *PHP Data Objects* (em português, objetos de dados da PHP), é uma extensão um pouco diferente. Ela provê uma camada de abstração ao acesso aos dados, ou seja, as mesmas funções são usadas de maneira independente do sistema de gerência de banco de dados utilizado. Isso quer dizer que, além de prover acesso a bancos MySQL, a PDO pode trabalhar com outros também, necessitando um *software (driver)* específico para cada tipo de banco (MANUAL DO PHP). A codificação dessa conexão pode ser visualizada nas Figuras 11 e 12.

Figura 11 - Classe de Conexão com o Banco de Dados



```
1 <?php
2 namespace App\Model;
3 class ConexaoDb
4 {
5     # Variável que guarda a conexão PDO.
6     protected static $db;
7     # Private construct - garante que a classe só possa ser instanciada internamente.
8     private function __construct()
9     {
10         # Informações sobre o banco de dados:
11         $db_host = "localhost";
12         $db_nome = "sgi";
13         $db_usuario = "root";
14         $db_senha = "";
15         $db_driver = "mysql";
16         # Informações sobre o sistema:
17         $sistema_titulo = "SGI";
18         $sistema_email = "nalves512@gmail.com";
19         try
20         {
21             self::$db = new PDO("$db_driver:host=$db_host; dbname=$db_nome", $db_usuario, $db_senha);
22             # Garante que o PDO lance exceções durante erros.
23             self::$db->setAttribute(PDO::ATTR_ERRMODE, PDO::ERRMODE_EXCEPTION);
24             # Garante que os dados sejam armazenados com codificação UTF-8.
25             self::$db->exec('SET NAMES utf8');
26         }
27         catch (\PDOException $e)
28         {
29             # Envia um e-mail para o e-mail oficial do sistema, em caso de erro de conexão.
30             mail($sistema_email, "PDOException em $sistema_titulo", $e->getMessage());
31             # Então não carrega nada mais da página.
32             die("Connection Error: " . $e->getMessage());
33         }
34     }
35 }
```

FONTE: Santos (2018), produção próprio autor

A conexão com o banco de dados na API PDO, instancia-se um objeto PDO, passando ao construtor, como parâmetro, os dados da conexão e, opcionalmente, usuário e senha. Um exemplo desse procedimento pode ser observado na linha 21 da figura 11. Observa-se na figura 12 o método estático da classe ConexaoDb.php que retorna a conexão,

Figura 12 - Método estático de conexão com o banco de dados

```
35  # Método estático - acessível sem instanciação.
36  public static function conexao()
37  {
38      # Garante uma única instância. Se não existe uma conexão, criamos
39      # uma nova.
40      if (!self::$db)
41      {
42          new ConexaoDb();
43      }
44      # Retorna a conexão.
45      return self::$db;
46  }
```

FONTE: Santos (2018), produção próprio autor.

O método estático da figura 12 é acessível sem instanciação, que garante uma única instância e se não existe uma conexão, será criada uma nova e retorna a variável que contém a conexão. Na Figura 13 representa o uso da classe de conexão nas classes models que são responsáveis pela manipulação do banco.

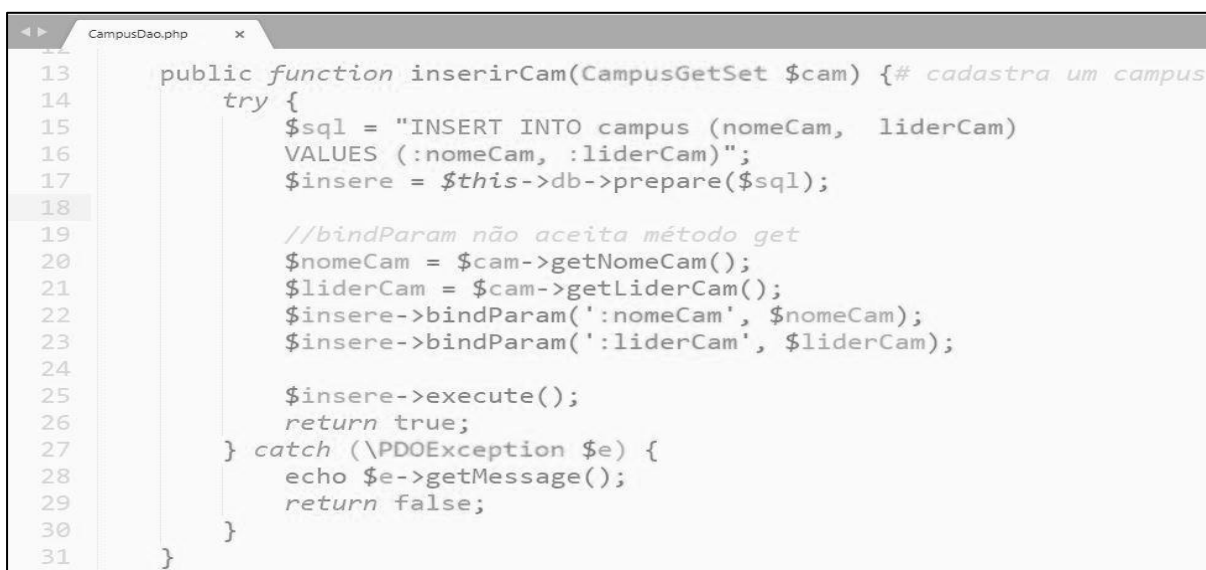
Figura 13 - Inclusão do Banco na Classe

```
1  <?php
2  namespace App\Model;
3  use App\Model\ConexaoDb;
4
5  class CampusDao {
6
7      public $db;
8
9      function __construct() {
10         $this->db = ConexaoDb::conexao();
11     }
```

FONTE: Santos (2018), produção próprio autor

Foi utilizada a linguagem SQL para manipulação dos dados no banco, quando é necessário fazer a inserção, alteração, visualização ou exclusão de informações do banco, devem-se criar as funções de manipulação. A figura 14 exemplifica uma função e seus métodos de manipulação utilizando a PDO.

Figura 14 - Método da Classe



```
13 public function insereirCam(CampusGetSet $cam) {# cadastra um campus
14     try {
15         $sql = "INSERT INTO campus (nomeCam, liderCam)
16             VALUES (:nomeCam, :liderCam)";
17         $insere = $this->db->prepare($sql);
18
19         //bindParam não aceita método get
20         $nomeCam = $cam->getNomeCam();
21         $liderCam = $cam->getLiderCam();
22         $insere->bindParam(':nomeCam', $nomeCam);
23         $insere->bindParam(':liderCam', $liderCam);
24
25         $insere->execute();
26         return true;
27     } catch (\PDOException $e) {
28         echo $e->getMessage();
29         return false;
30     }
31 }
```

FONTE: Santos (2018), produção próprio autor

3.2.4.2. Módulos e Telas do Sistema

Este item da seção será descrito as funcionalidades de cada módulo do software, definindo quais são e quem são seus usuários. O sistema foi organizado em 3 módulos para garantir a segurança das informações inseridas durante a utilização do sistema, cada módulo possui um nível de acesso, partindo de aberto até as áreas restritas, estes módulos são: a) página inicial; b) EaD; c) SGE.

O sistema apresenta no seu primeiro módulo uma página inicial que é de acesso aberto, com menu superior que permite acesso a sessões dentro da própria página bem como a outras funcionalidades do sistema. A página inicial pode ser observada na figura 15.

Figura 15 - Portal de Navegação nos Eventos



FONTE: Santos (2018), produção próprio autor

A tela de cursos também é de caráter público, sem restrição. Nesta tela o usuário poderá escolher um dos cursos disponíveis para realizar sua matrícula, verificar autenticidade de certificados e carteirinhas de estudantes emitidos pelo sistema. Na Figura 16 encontra-se listados todos os cursos e área de validação de certificados e carteirinha de estudantes.

Figura 16 - Tela de cursos

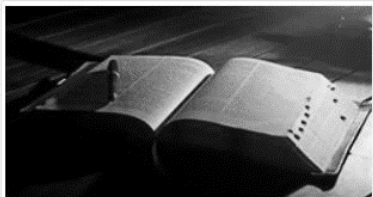


FONTE: Santos (2018), produção próprio autor

Após usuário escolher um curso para matricula, este será direcionado para tela de detalhes do curso, poderá confirmar matricula no curso e ainda confirmar validar carteirinha e certificado como apresentado na Figura 17.

Figura 17 - Tela de detalhes de curso

CURSO ONLINE GRÁTIS DE INTRODUÇÃO À TEOLOGIA CRISTÃ



Categoria do curso: Teologia

Area de aplicação: Aplica-se à área de ensino e pregação da palavra de Deus

Carga horária: de 280 horas

MATRICULAR!

A Escola de Cursos Online da **ADESC** oferece o curso de **Introdução à Teologia Cristã** **grátis** com certificado.

No curso gratuito de **Introdução à Teologia Cristã** o(a) aluno(a) terá acesso a um vasto material, disponível para download, em sua "Área do aluno" para estudar e se qualificar quando e onde quiser, disponível 24 horas por dia, 7 dias por semana em nossa plataforma de Ensino a Distância (EAD).

O curso online grátis de Introdução à Teologia Cristã possui **carga horária variável de 280 horas**. A carga horária final desse curso dependerá exclusivamente da necessidade do(a) aluno(a).

Para concluir esse curso é necessário realizar uma **avaliação** por meio de um questionário com 10 perguntas compostas de 4 respostas de múltipla escolha, no qual o(a) aluno(a) deverá obter uma média igual ou superior a 60,0 pontos para que esteja habilitado(a) a solicitar seu certificado.

Suporte ao aluno

Para maiores informações você pode solicitar ajuda de um de nossos atendentes pelo suporte ao aluno por **telefone:** (63) 3363-3786, **e-mail:** mscadesc@gmail.com ou **WhatsApp:** (63) 99291 1507.

FONTE: Santos (2018), produção próprio autor.

Assim que o usuário acionar o botão de matrícula, será direcionado para a tela cadastro e login, caso este não estiver autenticado no sistema. Como mostra a figura 18.

Figura 18 - Tela de login e cadastro

FONTE: Santos (2018), produção próprio autor

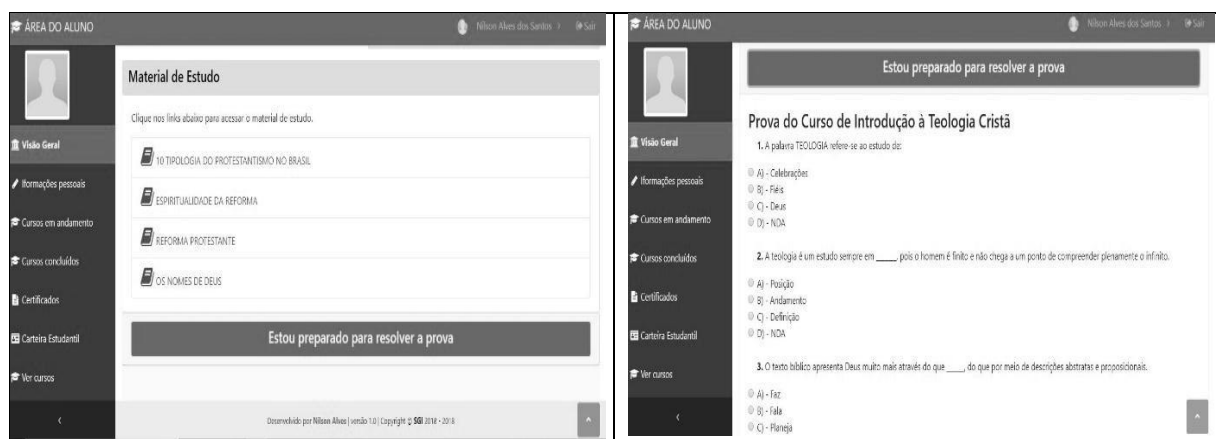
Na tela de cadastro e login o usuário tem a possibilidade de realizar seu cadastro, caso ainda não seja cadastrado, se sim pode realizar sua autenticação no sistema, podendo ainda nesta tela realizar a validação de certificados e carteirinhas, caso o usuário resolva logar no sistema será direcionado para área do aluno, de conforme a figura 19.

Figura 19 - Tela de área do aluno

FONTE: Santos (2018), produção próprio autor

A tela área do aluno é composta por uma barra superior, nela contém o nome, foto do usuário *logado* e o botão sair, na tela do aluno contém um menu lateral que possui acesso a todas informações pessoais do aluno, cursos em andamento e concluídos, certificados e carteirinhas estudantis, ainda na tela contém dois botões onde o primeiro é a confirmação de matrícula do curso escolhido e o segundo retorna a área de cursos, contém um menu que lista os certificados solicitados, cursos em andamento, concluídos e matriculados, nesse possui link para área de estudo e prova do curso conforme a Figura 20.

Figura 20 - Tela de estudo e prova



FONTE: Santos (2018), produção próprio autor

Na tela de estudo e prova representada pela figura 18 o aluno usuário poderá realizar o download do material de estudo e resolver as 10 questões de múltipla escolha da prova, após prova resolvida o aluno solicitará a correção imediata, o sistema calcula e disponibiliza o resultado na tela de resultados conforme figura 21.

Figura 21 - Tela de resultados



FONTE: Santos (2018), produção próprio autor

Na tela de resultado o sistema informa o percentual de acerto através de um número percentual e uma barra de progresso, emite uma mensagem de acordo com o índice de acerto, caso o aluno acerte a cima de 60% das questões da prova será aprovado, caso contrário será reprovado e terá a opção de estudar novamente e refazer a prova, uma vez aprovado, terá a opção de concluir o curso, daí será direcionado para tela de confirmação de dados para emissão do certificado. Conforme visto na figura 22.

Figura 22 - Tela dados do certificado



FONTE: Santos (201), produção próprio autor

Na tela de dados do certificado o aluno visualizará os dados a serem impressos no certificado e confirmará a impressão do mesmo. A figura 23 permite observar o certificado pronto para impressão em extensão PDF.

Figura 23 - Tela de certificado

IGREJA ASSEMBLÉIA DE DEUS		MINISTÉRIO DA SALVAÇÃO EM CRISTO
CERTIFICADO		
Certificamos que Nilson Alves dos Santos, concluiu com excelência o curso de Introdução à Teologia Cristã, em 06/10/2018, com duração de 280 horas, tendo um aproveitamento de 100 %.		
Código de validação deste Certificado: f00ad9f6df176cb31494ee7c3afe9eb1		
Acesse o site www.sge.com.br		
SECRETÁRIO		PASTOR

FONTE: Santos (2018), produção próprio autor

No módulo SGE – Gestão Evangélica, todas as telas são de acesso restrito, após solicitar acesso ao sistema, na página inicial, o usuário será direcionado para tela de login onde poderá se autenticar de acordo com o seu nível de permissão. Na figura 24 apresenta-se a tela de login.

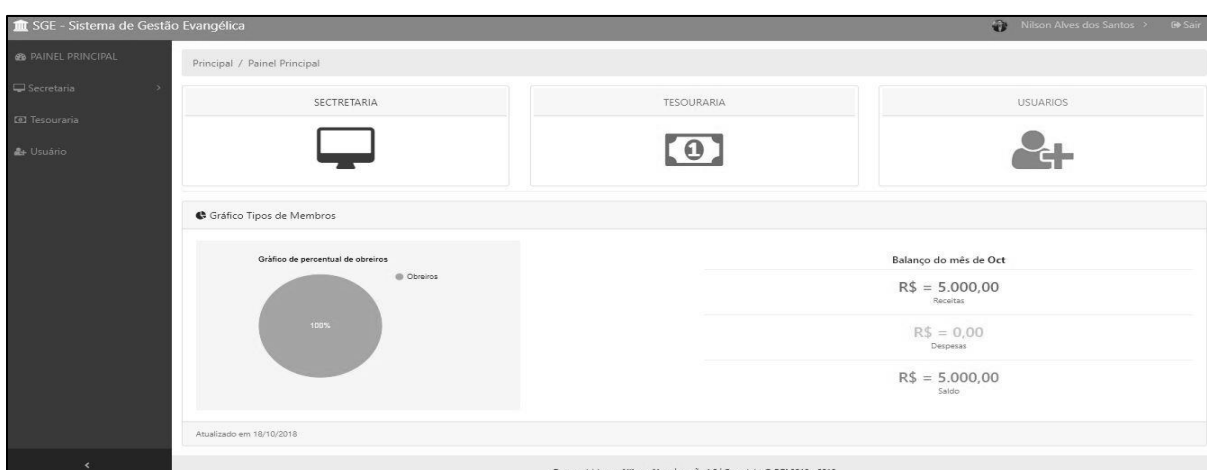
Figura 24 - Tela de login SGE



FONTE: Santos (2018), produção próprio autor

Após login no sistema o usuário administrador terá acesso ao painel principal do sistema conforme demonstrado na figura 25.

Figura 25 - Tela painel principal



Fonte: Santos (2018), produção próprio autor.

Na tela de painel principal o administrador pode ter acesso a qualquer parte do sistema detalhadas nas figuras a seguir.

Figura 26 - Tela painel secretaria



FONTE: Santos (2018), produção próprio autor

O painel secretaria é de acesso exclusivo do secretário nesta, ele poderá acessar as páginas de campus, igrejas e membros, conforme descritos abaixo.

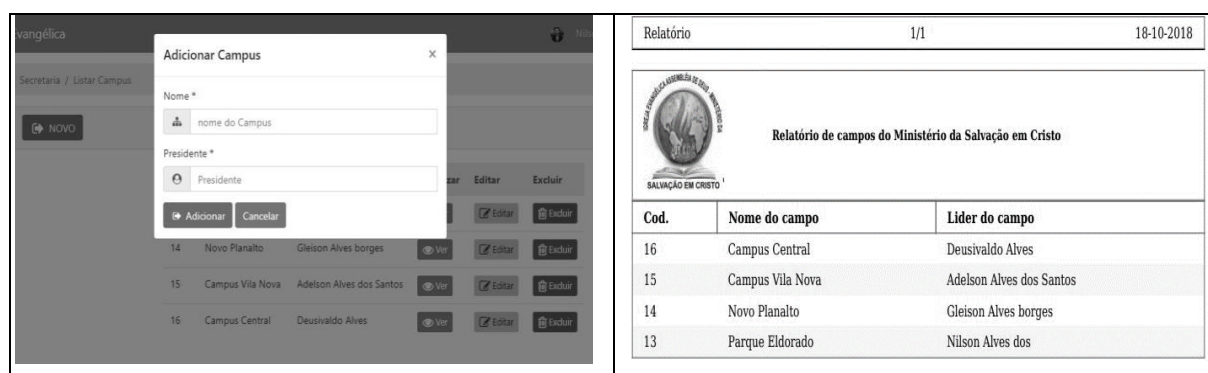
Figura 27 - Tela painel campus



FONTE: Santos (2018), produção próprio autor.

Nesta tela, lista os *campus* cadastrados, contém o botão para impressão de relatório dos campus com extensão PDF, botão para visualizar detalhes de campus, botão de edição, exclusão e um o botão que aciona a janela modal de cadastro de novos campus conforme figura 28.

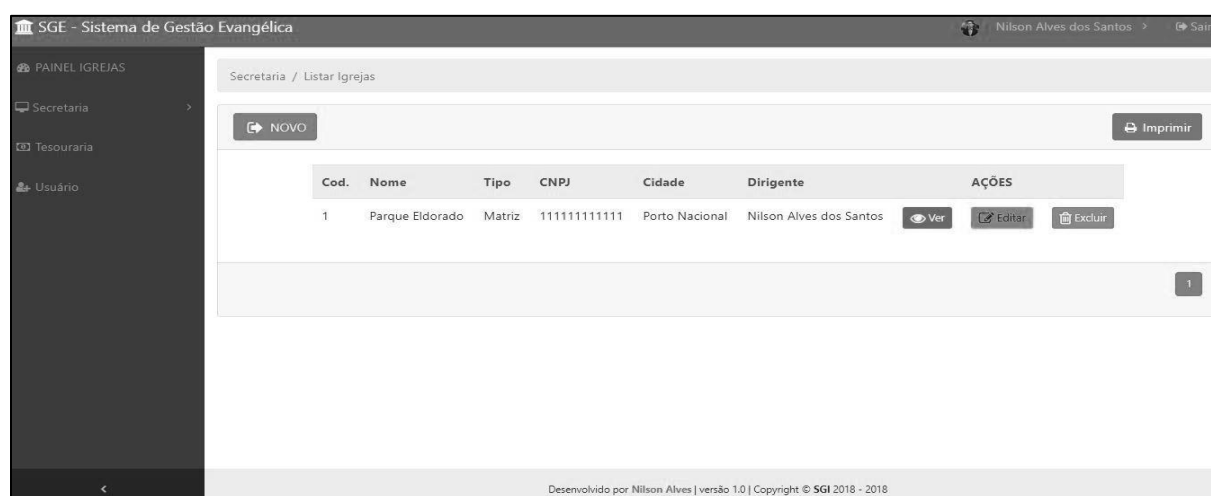
Figura 28 - Tela cadastro de campus e tela de relatório



FONTE: Santos (2018), produção próprio autor

Na tela de painel igrejas o secretário poderá manipular todas as informações referentes a igrejas, emitir relatório, visualizar, editar e excluir bem como cadastrar novas igrejas, conforme figura 29.

Figura 29 - Tela igreja



FONTE: Santos (2018), produção próprio autor

Na tela de cadastro de igreja o secretário preencherá todas informações da igreja e submeterá o formulário, de acordo com a figura 30.

Figura 30 - Tela cadastro de igreja

SGE - Sistema de Gestão Evangélica

Nilson Alves dos Santos > Sair

INserir IGREJAS

Secretaria

Tesouraria

Usuário

Igrejas / Inserir Igrejas

Ver Igrejas

OBS:
Somente Imagens ('gif', 'jpeg', 'jpg', 'png').
Escolher ficheiro Nenhum ficheiro selecionado

Igreja*

Nome da igreja

Dirigente*

Dirigente

Campus*

Escolha um Campus

Estado*

Selecione o Estado

Rua*

Av. Rua...

Pagina Web

www.url.com

Adicionar

Tipo*

Selecione o tipo da igreja

CNPJ

CNPJ

Cidade*

Cidade

Bairro*

Bairro

Numero

Numero

FONTE: Santos (2018), produção próprio autor.

Na tela de cadastro de igreja o secretário poderá voltar a tela anterior e de lá entrar na tela de manipulação de membros, conforme tabela 31.

Figura 31 - Tela lista de membros

SGE - Sistema de Gestão Evangélica

Nilson Alves dos Santos > Sair

PAINEL MEMBROS

Secretaria

Tesouraria

Usuário

Secretaria / Listar Membros

NOVO

Cod.	Nome	Funcao	Sexo	Tipo	Status	E-mail	Igreja	Ação
1	Nilson Alves dos Santos	Diacono	M	Obreiro	Ativo	nalves512@gmail.com	1	Ver Editar Excluir

1

Desenvolvido por Nilson Alves | versão 1.0 | Copyright © SGI 2018 - 2018

FONTE: Santos (2018), produção próprio autor

Na tela de painel de membros o secretário poderá manipular todas as informações referentes a membros, emitir relatório, visualizar, editar e excluir bem como cadastrar novos membros, conforme figura 32.

Figura 32 - Tela cadastro de membros

SGE - Sistema de Gestão Evangélica

Nilson Alves dos Santos Sair

PAINEL INserir MEMBROS

Ver Membros

Imagem (gif, jpeg, jpg, png)
Escolher arquivo Nenhum ficheiro selecionado

Sexo*
Selecione o Sexo

Número
Número

Estado
Selecione o Estado

Naturalidade
Naturalidade

Departamento
Departamento

Nome*
Nome do Membro

Pai
Pai

Conjuge
Conjuge

Rua
Av. Rua...

Bairro
Bairro

País
País

Função
Função

Tipo de Admissão
Selecione Tipo de Admissão

Mãe*
Mãe

Igreja
Selecione uma igreja

Cidade
Cidade

CEP
CEP

Estado Civil
Selecione Estado Civil

Tipo de Membro
Selecione Tipo Membro

FONTE: Santos (2018), produção próprio autor.

A figura 33 demonstra o acesso exclusivo do tesoureiro, nela pode observar lista de entrada e saída de valores, balanço financeiro mensal e geral.

Figura 33 - Tela tesouraria

FONTE: Santos (2018), produção próprio autor

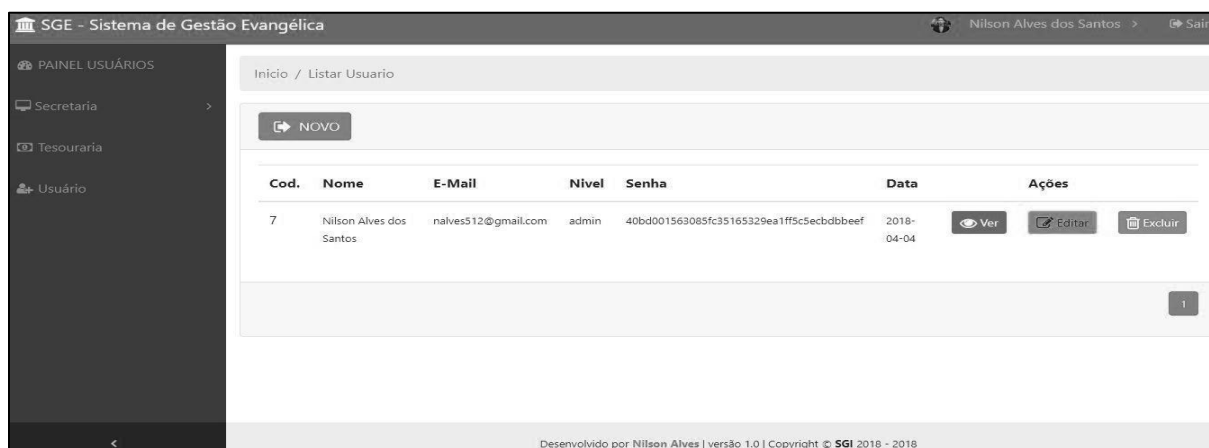
Na tela de tesouraria o tesoureiro poderá realizar o cadastro de movimento financeiro, e cadastro de categorias do movimento, conforme descrito na figura 34.

Figura 34 - Tela cadastro de movimento e categoria

FONTE: Santos (2018), produção próprio autor.

A figura 35, tela de usuários a seguir é de acesso exclusivo do usuário administrador, esta dispõe a lista de todos os usuários cadastrados.

Figura 35 - Tela de usuários



FONTE: Santos (2018), produção próprio autor.

Nesta tela o administrador poderá visualizar, editar e excluir usuários, além de realizar novo cadastro através do botão novo, conforme figura 36.

Figura 36 - Tela cadastro de usuário

A imagem mostra a interface de usuário do SGE - Sistema de Gestão Evangélica para o cadastro de um novo usuário. No topo, há uma barra de navegação com o nome de usuário 'Nilson Alves dos Santos' e um botão 'Sair'. À esquerda, um menu lateral contém 'PAINEL INSERIR USUÁRIOS', 'Secretaria', 'Tesouraria' e 'Usuário'. O conteúdo principal exibe o caminho 'Início / Inserir Usuario' e um botão 'Ver usuários'. Abaixo, há um formulário com campos para: 'Email*' (com ícone de envelope), 'Nível*' (menu suspenso), 'Data*' (formato dd/mm/aaaa), 'Nome*' (campo de texto), 'Usuário*' (campo de texto), 'Senha*' (campo de texto com ícone de cadeado) e 'Repetir Senha*' (campo de texto com ícone de cadeado). Há também um campo para upload de imagem com uma observação: 'OBS: Somente imagem gif, jpeg, jpg, png.' e um botão 'Escolher ficheiro'. No rodapé, consta 'Desenvolvido por Nilson Alves | versão 1.0 | Copyright © SGI 2018 - 2018'.

FONTE: Santos (2018), produção próprio autor.

Nessa tela o administrador insere todas as informações do usuário, submete o formulário e pode retornar a tela da tesouraria e posteriormente acessar qualquer módulo do sistema novamente.

Dessa forma conclui-se a descrição do desenvolvimento do *software*. Em seguida no tópico 3.3 descreve-se a avaliação do sistema.

3.3. Avaliação do software

Para avaliar a qualidade do sistema foi desenvolvida uma pesquisa. Para tanto, confeccionou-se um questionário de pesquisa com objetivo de avaliar a usabilidade do *software*. Essa etapa da pesquisa foi realizada após o desenvolvimento do *software*, para facilitar a elaboração dos instrumentos de pesquisa e coleta dos dados em conformidade com o sistema, o questionário foi definido em reunião junto ao professor orientador que norteou a confecção do mesmo, assim todas as edições do questionário foi acompanhada pelo orientador.

A edição do questionário foi realizada em alguns passos: o primeiro foi pesquisa sobre como confecciona-los, o segundo passo foi a criação dos escopos para tomada de base na edição, o terceiro passo foi as correções dos escopos, o quarto passo foi a confecção do questionário no Google Docs, ferramenta gratuita para criação de formulários, sendo fundamental para a realização da pesquisa por meio da internet. A seguir encontra-se o questionário e como foi realizada a confecção.

3.3.1. Confecção do Questionário da Pesquisa

A confecção do questionário da pesquisa se deu com base na necessidade de avaliar o software que foi desenvolvido como produto final desse trabalho, assim a sua elaboração se deu com base no método avaliador baseado na escala de Likert. Instrumento de mensuração das atitudes ou opiniões (Likert, 1932), as chamadas escalas de atitude, como descritos a seguir: 1- Pouquíssima interativa, 2- Pouco interativa, 3- Interativa, 4- Muito interativa, 5- Muitíssima interativa. Tal escala garante uma maior precisão na análise e tabulação dos dados obtidos. O questionário de pesquisa construído nesta etapa está disposto no apêndice A.

3.3.2. Submissão do Questionário da Pesquisa

Após a confecção foi realizada a submissão do questionário de pesquisa, com o objetivo de coletar os dados para avaliação da utilização do sistema, o questionário de pesquisa foi submetido por meio de um formulário Google Docs que permite aos participantes responderem via internet, outra vantagem deste recurso é enviar o link do formulário via e-mail, possibilitando que o mesmo seja respondido com tempo pelo participante e de qualquer lugar que disponha de internet.

3.4. Comentários Parciais

Neste capítulo foram apresentados os procedimentos metodológicos traçados desde a criação dos formulários até sua aplicação e os métodos utilizados para

coleta e análise dos dados utilizados nesta pesquisa. O mesmo foi dividido em duas etapas, a primeira sendo a definição dos procedimentos metodológicos da pesquisa e a segunda sendo a confecção do método de coleta de dados.

Dessa forma, foram apresentados na primeira etapa os embasamentos teóricos para realização da referida pesquisa, foi realizada a definição do universo, o método a ser utilizado e a escolha da forma da aplicação do questionário de pesquisa.

Na segunda etapa, foi selecionado o instrumento de coleta de dados, seguindo com a sua confecção que no primeiro momento foi elaborado um escopo do questionário da pesquisa, no segundo momento foi confeccionado o questionário da pesquisa, que só então foi aplicado para a coleta dos dados.

Todas as atividades realizadas neste capítulo foram importantes para alcançar os objetivos desta pesquisa que é desenvolver o *software*, utilizá-lo e desfrutar dos benefícios proposto neste trabalho, sendo válidos através dos dados apresentados nesta pesquisa, que comprovam a qualidade e importância da aplicação desenvolvida.

4. ANÁLISE E RESULTADOS DOS DADOS

Neste item são apresentados os resultados e análises obtidas com a pesquisa que avalia a aceitação do software SGE. Com vista avaliar a aceitação do *software* proposto, foram levantados os seguintes questionamentos: cidade, nome da instituição de vínculo, qual o vínculo com a instituição, grau de indução da interface, grau de facilidade do entendimento do sistema, grau de Agilidade, grau de Produtividade, grau de Assertividade das informações, grau de respostas, grau de acessibilidade e se atende à necessidade.

A média dos resultados obtidos com a submissão do questionário da pesquisa encontram se dispostos na Tabela 6, conforme a escala de 5 pontos de Likert. A tabela contendo as respostas completas estão dispostas no apêndice B.

Tabela 6 - Respostas referente ao questionário da pesquisa

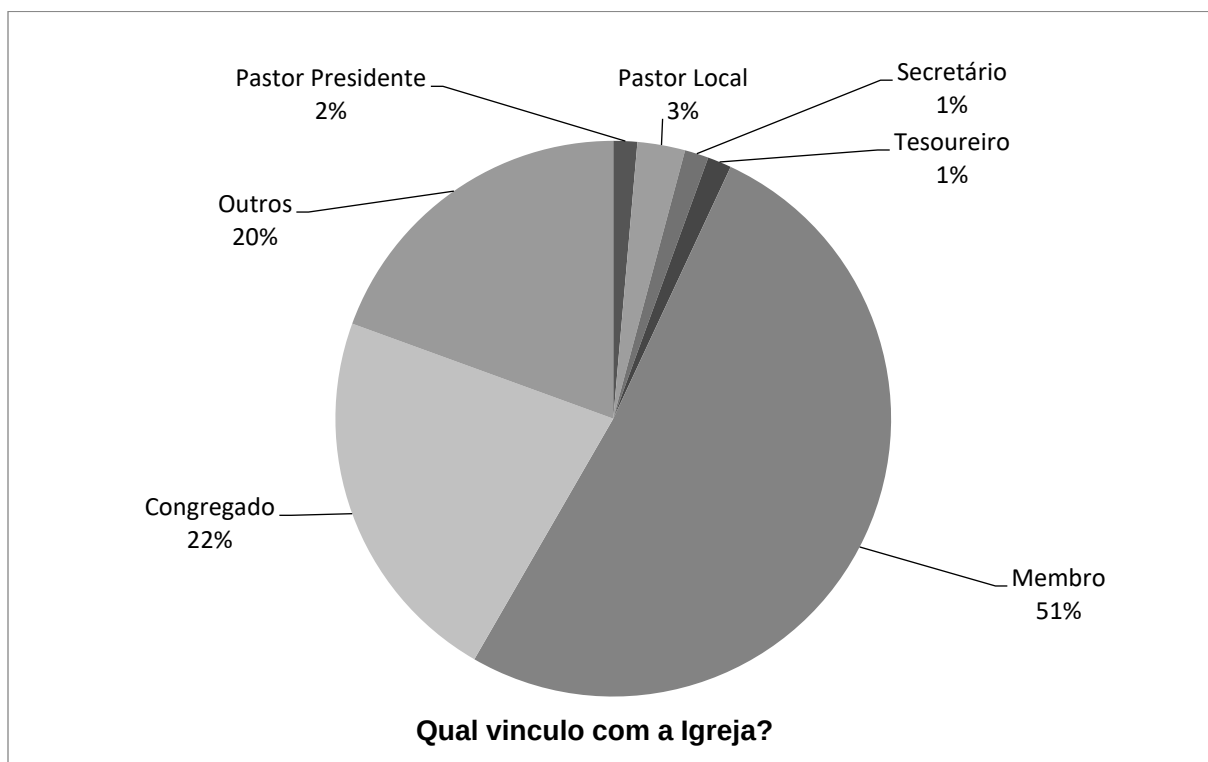
1 - A interface do sistema é indutiva? RESPOSTAS 1 A 5	2 - Com base na utilização do sistema, como você classifica o grau de facilidade do entendimento do sistema? RESPOSTAS 1 A 5	3 - Com base na utilização do sistema, como você classifica o grau de Agilidade? RESPOSTAS 1 A 5	4 - Ainda com base na utilização do sistema, como você classifica o grau de produtividade? RESPOSTAS 1 A 5	5 - Continuando com a utilização do sistema, como você classifica o grau de assertividade das informações? RESPOSTAS 1 A 5	6 - Um sistema responsivo é quando o site automaticamente se encaixa no dispositivo do usuário como você avalia esse comportamento do sistema? RESPOSTAS 1 A 5	7 - Qual o grau de acessibilidade do sistema? RESPOSTAS 1 A 5	8 - O desempenho do sistema atende as necessidades do serviço local? RESPOSTAS 1 A 5	9 - Você indicaria esse sistema para ser implantado na sua igreja? RESPOSTAS 1 A 3	10 - O Módulo EaD do sistema atende o ensino/aprendizagem da igreja? RESPOSTAS 1 A 3
4,541	4,388	4,375	4,500	4,375	4,791	4,486	4,625	2,986	2,986

FONTE: Santos (2018), produção próprio autor

Do universo da pesquisa de 72 participantes, 1 pastor presidente, 2 pastores locais, 1 secretário, 1 tesoureiro 37 membros, 16 congregados e 14 classificados como outros. Desse total de 72, todos responderam ao questionário da pesquisa.

Foi perguntado qual o vínculo dos pesquisados com a instituição evangélica Assembleia de Deus Salvação em Cristo 51% declaram membros, 22% congregados, 20% outros, 2% pastor presidente 3% pastor local, 1% secretário e 1% tesoureiro, conforme mostra a figura 37.

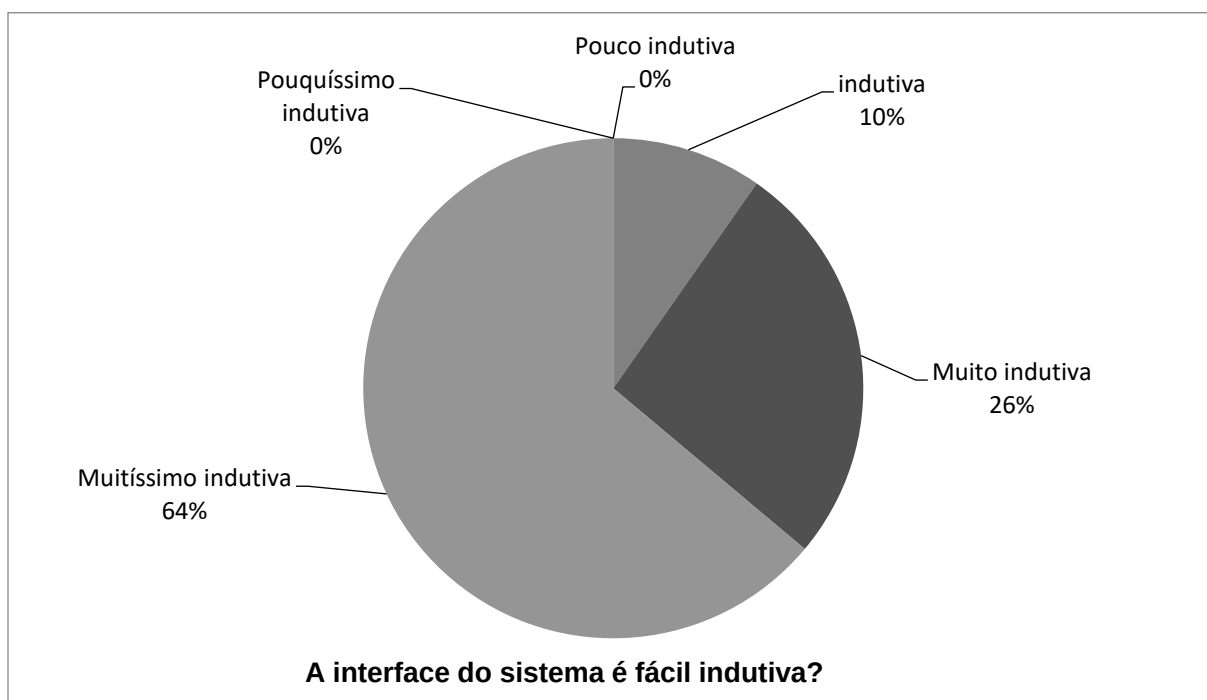
Figura 37 - Vínculo com a Igreja



FONTE: Santos (2018), produção próprio autor

Nas próximas figuras, representa a avaliação do software pois, a figura apresentada anteriormente refere-se aos participantes da pesquisa. A figura 38 aponta a avaliação da interface do sistema, constando que 64% dos participantes responderam que a interface é muitíssima indutiva, 26% responderam que o sistema é muito indutivo e 10% responderam que é indutivo, pouco indutiva e pouquíssima indutiva não pontuaram.

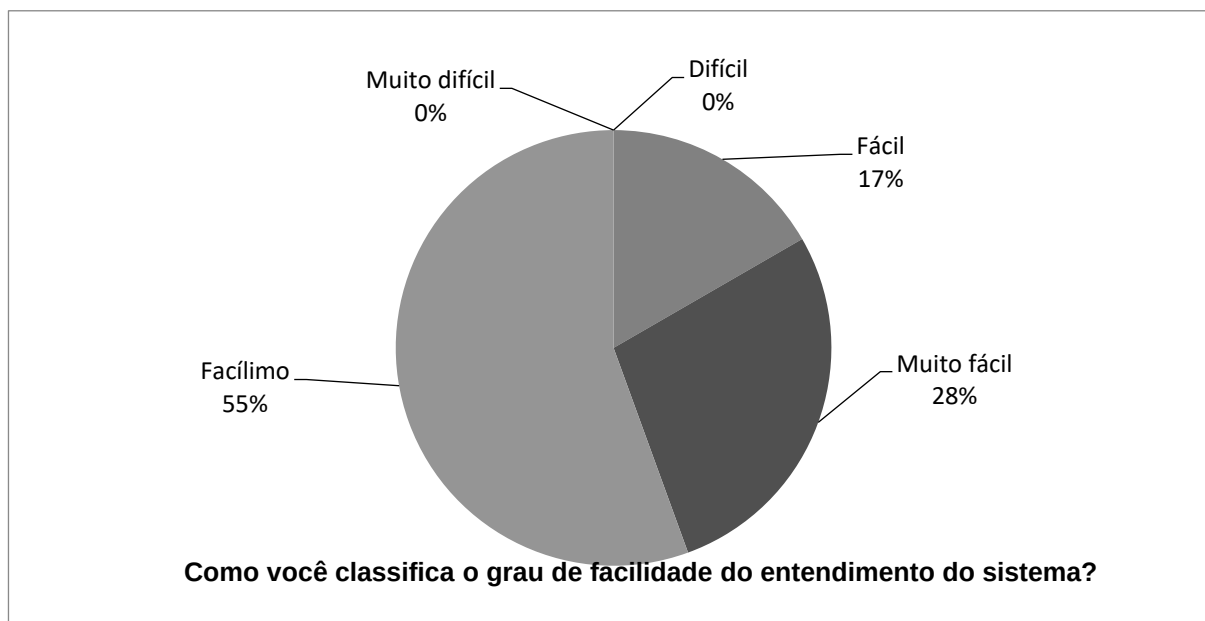
Figura 38 - Indutividade da interface do sistema



FONTE: Santos (2018), produção próprio autor

No próximo Gráfico da figura 39 apresenta o resultado referente a pergunta: com base na utilização do sistema, como você classifica o grau de facilidade do entendimento do sistema? 55% dos pesquisados responderam que o sistema é fácil, 28% responderam que o software é muito fácil e 17% responderam que o sistema é muito fácil, já a classificação difícil e muito difícil não pontuou.

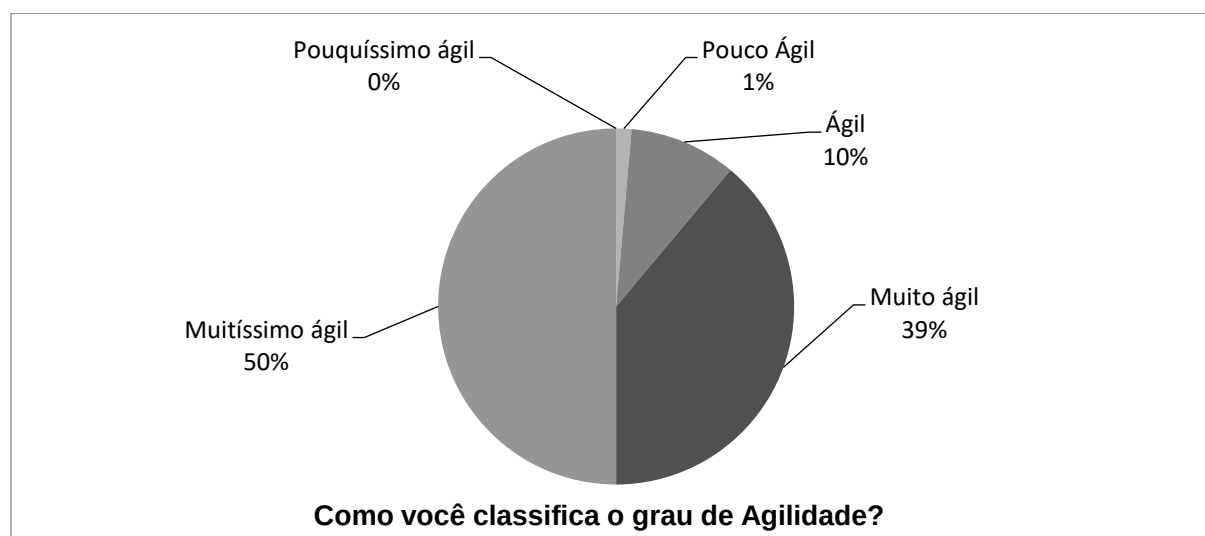
Figura 39 - Grau de facilidade do entendimento do sistema



FONTE: Santos (2018), produção próprio autor

Quando perguntado sobre a o grau de agilidade do sistema 50% dos participantes afirmaram que o sistema é muitíssimo ágil, 39% afirmaram que o sistema é muito ágil, 10% afirmaram que o sistema é ágil e 1% responderam que o sistema é pouco ágil, a classificação pouquíssima ágil não pontuou.

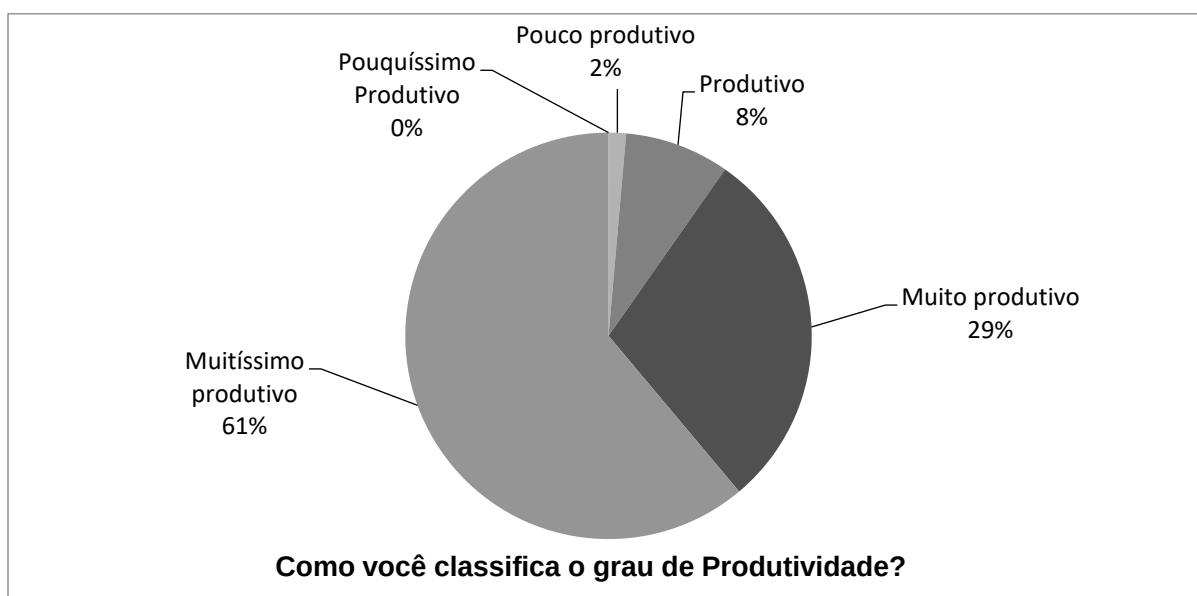
Figura 40 - Grau de Agilidade



FONTE: Santos (2018), produção próprio autor

A figura 41 comprova que o sistema é produtivo, pois apenas 2% responderam que o sistema era pouco produtivo, assim, 61% responderam que é muitíssimo produtivo, 29% que o sistema é muito produtivo, 8% produtivo e nenhum respondeu que o sistema é poucoíssimo produtivo.

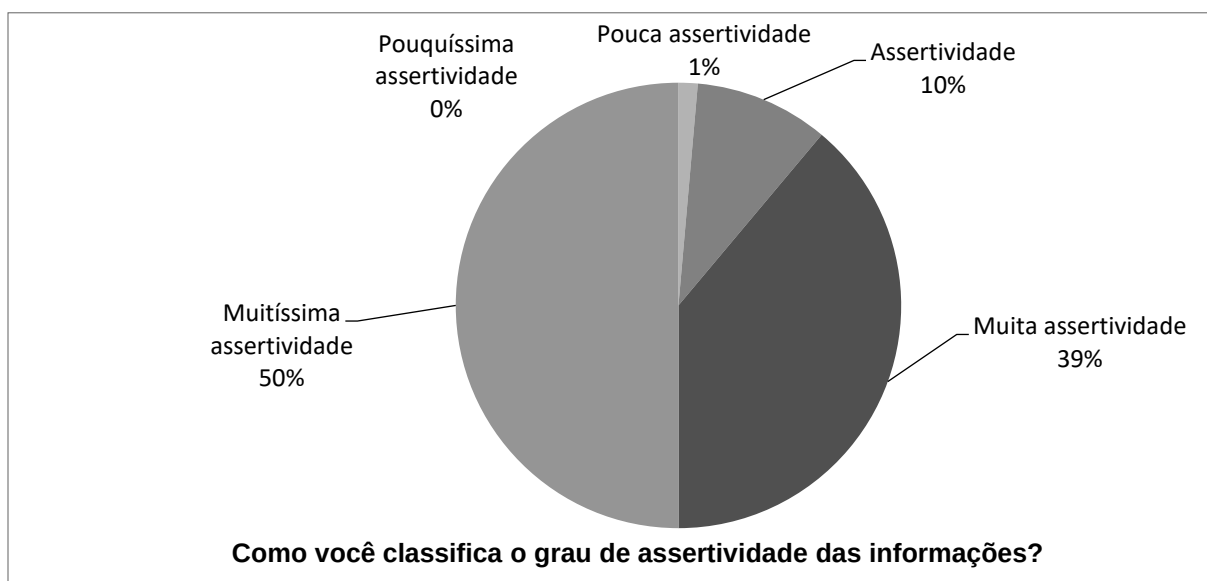
Figura 41 - Grau de produtividade do Sistema



FONTE: Santos (2018), produção próprio autor

A figura 42 a seguir demonstra que 50% dos participantes informam que o sistema tem muitíssima assertividade, 39% disseram que tem muita assertividade, 10% informaram que o sistema tem assertividade, apenas 1% disseram que tem pouca assertividade e nenhum responderam pouquíssima assertividade.

Figura 42 - Grau de assertividade das informações



FONTE: Santos (2018), produção próprio autor

Um sistema responsivo é quando o site automaticamente se encaixa no dispositivo do usuário (PC, celular, *tablet*, etc.), quando perguntado sobre a o grau de responsividade do sistema 83% dos participantes afirmaram que o sistema é muitíssimo responsivo, 13% afirmaram que o sistema é muito responsivo, 4% afirmaram que o sistema é responsivo as classificações pouco e pouquíssimo responsivo não pontuaram, conforme disposto no gráfico da figura 43 a seguir.

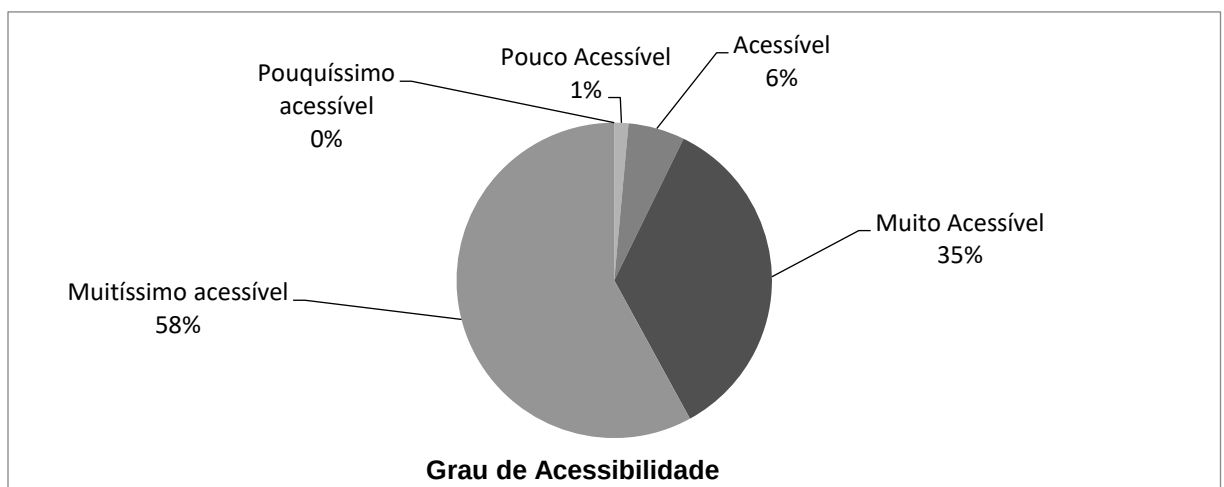
Figura 43 - Grau de Responsividade do sistema



FONTE: Santos (2018), produção próprio autor.

A figura 44, avalia a acessibilidade do sistema, assim 58% responderam que o sistema é muitíssimo, 35% responderam que é muito acessível, 6% que é acessível, apenas 1% respondeu que é pouco acessível e ninguém respondeu a opção pouquíssimo acessível.

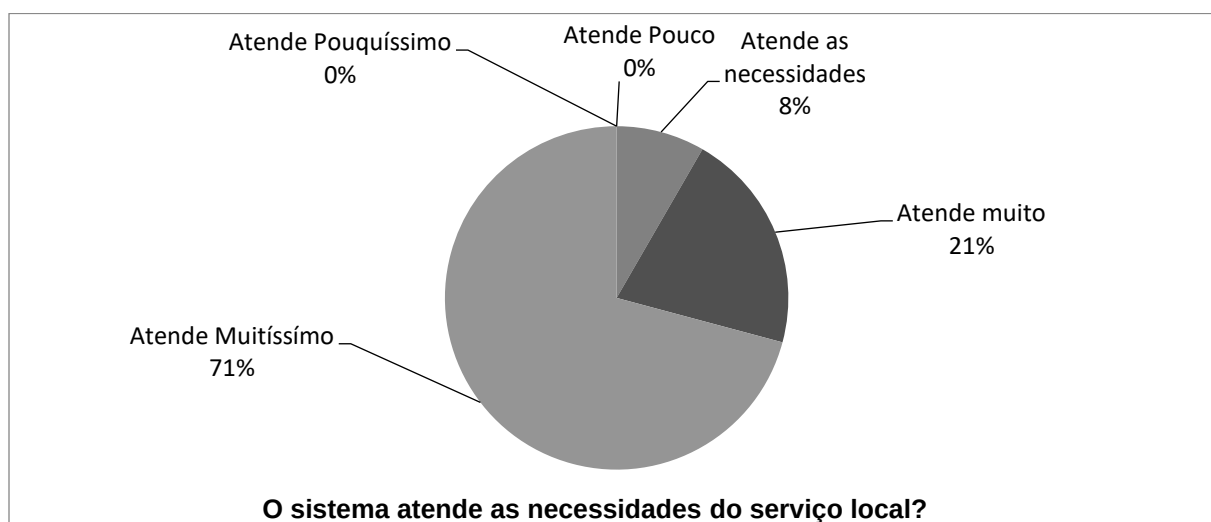
Figura 44 - Grau de acessibilidade ao sistema



FONTE: Santos (2018), produção próprio autor.

As figuras a seguir representam a opinião dos participantes sobre o desempenho do sistema, se atende as necessidades e se indicariam o software para implantação na igreja. Os dados apresentados revelam que o sistema SGE está atendendo aos objetivos da pesquisa, pois conforme apresentado, 71% dos pesquisados afirmaram que o sistema atende muitíssimo as necessidades, 21% disseram que atende muito as necessidades e 8% disseram que atende as necessidades, conforme figura 45.

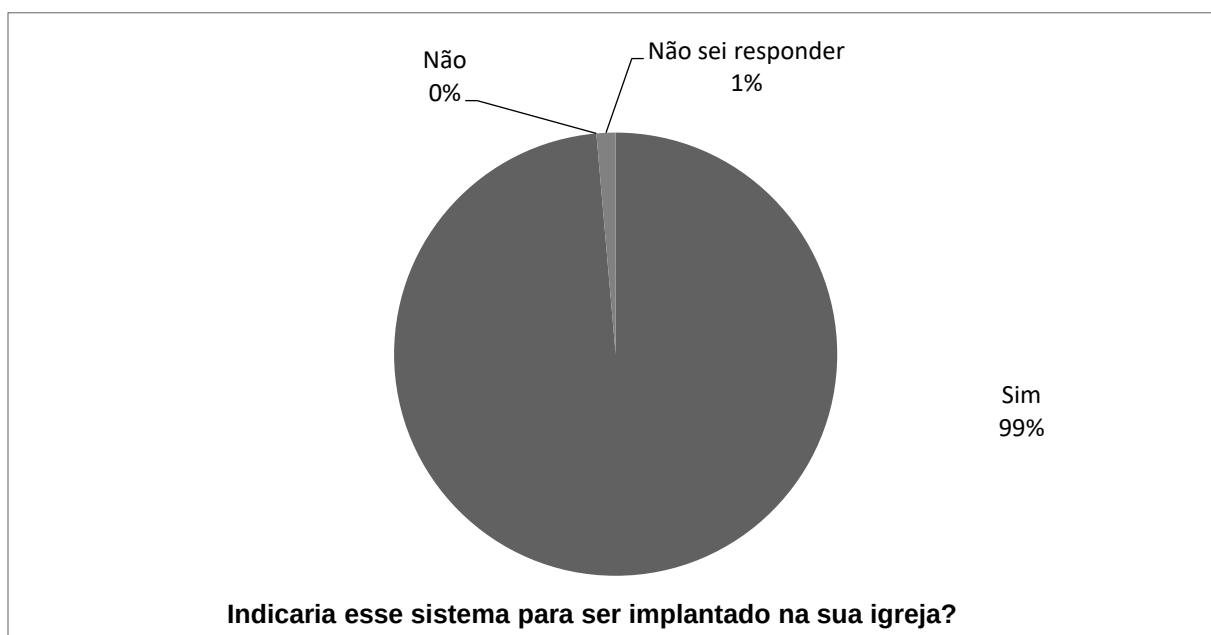
Figura 45 - Desempenho do Sistema



FONTE: Santos (2018), produção próprio autor

A figura 46 demonstra que 99% dos participantes indicariam o sistema para ser implantado na instituição só 1% não souberam opinar.

Figura 46 - Implantação do sistema



FONTE: Santos (2018), produção próprio autor

Por fim, perguntou-se se o sistema SGE no módulo de EaD atende as necessidades do serviço da igreja, 99% dos pesquisado responderam que sim, ao passo que apenas 1% não souberam responder, conforme mostra a figura 47.

Figura 47 - Módulo EaD do sistema atende o ensino/aprendizagem



FONTE: Santos (2018), produção próprio autor.

4.1. Comentários Parciais

Neste capítulo está sendo apresentados os resultados obtidos na avaliação do software junto aos membros envolvidos na instituição religiosa, a análise destes resultados mostra a excelente aceitação do sistema desenvolvido, atendendo ao objetivo específico b.

Os dados obtidos com a pesquisa permitiram avaliar a aceitação do sistema, suas funcionalidades, o desempenho, a interatividade, a facilidade de uso e outros. Esses dados foram coletados com base nas repostas dos participantes da pesquisa que acessaram o *software* na *Web* e realizaram a avaliação, percebe-se ao final que 99% dos pesquisados indicariam o *software* para implantação na instituição.

5. CONSIDERAÇÕES FINAIS

Diante dos resultados obtidos com a realização do referido estudo, nota-se que o *software* foi avaliado positivamente pelos pesquisados, dessa forma, esse trabalho se propôs a desenvolver um *software* denominado Sistema de Gestão Evangélica – SGE, utilizá-lo como ferramenta de auxílio da Gestão e desfrutar dos benefícios, que reduzisse os problemas com a gestão de igrejas. Daí a importância de usufruir de um sistema de gestão, no âmbito da Igreja Evangélica Assembleia de Deus, Ministério da Salvação em Cristo, *Campus* Porto Nacional – TO.

Portanto, seguir padrões, modelos, métodos e processos da engenharia de *software* foram fundamentais para cumprir um dos objetivos deste trabalho, que é o desenvolvimento de um sistema de informação para processamento de dados, inserindo e resgatando informações do banco de dados. Neste processo o sistema auxiliará na gestão de igrejas evangélicas, promovendo significativamente facilidade, agilidade, assertividade e maior produtividade no processo de gerenciamento.

As ferramentas e os ambientes utilizados para o desenvolvimento deste sistema mostraram-se ideais com base no Referencial teórico deste trabalho, facilitando a tomada de decisão durante a seleção dos recursos para o desenvolvimento do sistema, aplicando conceitos que abrangem do primeiro ao último semestre do curso. Através dos recursos e técnicas utilizados no levantamento de requisitos, e no que diz a literatura, abstraiu-se uma boa percepção do problema e, a partir daí, foi possível desenvolver toda a codificação do *software*.

A aplicação apresentada foi validada junto aos membros e líderes da Igreja Evangélica Assembleia de Deus, Ministério da Salvação em Cristo, – *Campus* Porto Nacional – TO. onde confirmou-se que o referido *software* solucionou o problema inicialmente proposto. Pois, o que antes era feito de forma manual agora poderá ser feito de forma computadorizada, além do que, outras igrejas de mesma ideologia que tenham interesse em utilizar o sistema poderão se beneficiar também deste *software*, devido seu fácil manuseio e suas respectivas funcionalidades. Desta forma os objetivos foram alcançados.

De fato, o *software* apresentado como sistema de gestão evangélica melhora a gestão de igrejas, no que se refere a facilidade, agilidade, assertividade e maior produtividade. Assim, o sistema atende as atuais necessidades, alcançando os

objetivos e a hipótese levantada foi validada. Pois, de acordo com os dados da pesquisa, 71% dos pesquisados afirmaram que o sistema atende muitíssimo as necessidades, 21% disseram que atende muito as necessidades e 8% disseram que atende as necessidades, esses resultados podem contribuir diretamente, de forma positiva, no processo de avaliação do sistema.

Por fim, este trabalho como um todo propiciou grande experiência ao autor, a principal razão é o envolvimento e foco em todas as etapas de desenvolvimento e escrita do projeto. Com o foco no desenvolvimento, foi possível colocar em prática todos os conceitos levantados durante a pesquisa bibliográfica bem como as habilidades adquiridas nas disciplinas no decorrer do curso.

5.1. Trabalhos Futuros

Para trabalhos futuros, propõe-se realizar atualização no módulo de cadastro de membros para que estes realizem seus próprios cadastros, o administrador terá o trabalho apenas de moderar cada cadastro, tornando assim, o sistema mais autônomo. Seria interessante implementar um módulo para cobrança das emissões de certificados do módulo EaD, ainda neste módulo propõe-se criar um painel administrativo onde proporcionará o administrador a realizar cadastro de cursos novos no sistema, pois na atual versão só é permitido a inclusão de novos cursos diretamente no código.

Propõe-se realizar atualizações no sistema de login, automatizando a recuperação de senhas, sendo estas enviadas via e-mail do usuário cadastrado no sistema. Além do mais poderiam ser desenvolvidos novos módulos que surgissem com as necessidades e fazer incorporações a outros sistemas. E, por fim, recomenda-se que sempre incentive o desenvolvimento de *software* que preencham necessidades reais das instituições religiosas e interessados, e também incentive o aprimoramento de seus conhecimentos adquiridos durante a realização do curso.

REFERÊNCIAS

AGÊNCIA NOTÍCIA IBGE. **Censo 2010: número de católicos cai e aumenta o de evangélicos, espíritas e sem religião.** Disponível em: <<https://agenciadenoticias.ibge.gov.br/agencia-noticias/2013-agencia-de-noticias/releases/14244-asi-censo-2010-numero-de-catolicos-cai-e-aumenta-o-de-evangelicos-espíritas-e-sem-religiao.html>>. Acesso em: 06 de maio 2018.

ANICETO, Jefferson. **Aplicações Web. Apostila ASP.net.** Escola Técnica da Univale (ETEIT). 2009.

BEIGHLEY, Lynn. **Use a cabeça SQL.** São Paulo: Alta Books Editora, 2010.

BELTRAME, M. M.; MAÇADA, A. C. G. **Validação de um Instrumento para medir o valor da tecnologia da informação (TI) para as organizações.** Disponível em: <<http://www.spell.org.br/documentos/download/5930>>. Acesso em: 13 de abr. 2018.

BERNERS-LEE, T. **The World Wide Web — past, present and future.** Journal of Digital Information, [S.R.], v.1, n.1, Aug. 1996. ISSN: 1368-7506. Disponível em: <<https://journals.tdl.org/jodi/index.php/jodi/article/view/3/3>>. Acesso em: 27 mar. 2018.

BERTOLLO, G. e FALBO, R. A. (2003) **“Apoio Automatizado à Definição de Processos de Software em Níveis”.** In: II Simpósio Brasileiro de Qualidade de Software (SBQS), Fortaleza, Brasil.

BLOG DA KINGHOST. **Introdução ao Composer: o que é, como instalar, como usar.** Disponível em: <<https://king.host/blog/2018/02/composer-o-que-e-como-instalar-como-usar/>>. Acesso em 22 out. 2018.

BOOCH, G., Rumbaugh, J., & Jacobson, I., 2000. **UML - Guia do Usuário.** Editora BRASIL. LEI Nº 10.825, DE 22 DE DEZEMBRO DE 2003. **Dá nova redação aos arts. 44 e 2.031 da Lei nº 10.406, de 10 de janeiro de 2002, que institui o Código Civil.** Disponível em: < http://www.planalto.gov.br/ccivil_03/leis/2003/L10.825.htm >. Acesso em: 08 maio 2018.

BROWN, Alan W. **Large-scale component-based development.** Upper Saddle River, NJ: Prentice Hall, 2000. 286p.Campus.

CAMPOS, Jorge Murilo Raniero; LEAL, Gislaine Camila Lapasiani. **PROPOSTA DE PROCESSO PARA GERENCIAMENTO DE REQUISITOS NO DESENVOLVIMENTO DE SOFTWARE.** Trabalhos de Conclusão de Curso do DEP, v. 13, n. 1, 2018.

CARVALHO, A.; CHIOSSI, T. **Introdução à Engenharia de Software.** São Paulo: Unicamp, 2001.

CHIAVENATO, Idalberto. **Introdução à teoria geral da administração.** 6 ed. Rio de Janeiro: Campus, 2000. Coleção formação a distância & e-Learning, Inofor, 2002.

CONCEIÇÃO, Frederico do Nascimento. **Critérios fundamentais que determinam a construção de uma liderança cristã em uma entidade religiosa do Distrito Federal.** UniCEUB – Centro universitário de Brasília. 2013.

CONSMATER, **Empresas de todos os tamanhos e segmentos podem desfrutar das inovações da plataforma SAP.** Disponível em: <<http://consmaster.com.br/2017/01/>>. Acesso em 29 out. 2018.

COPLIEN, J. H.; HARRISON, N. B. **Organizational Patterns of Agile Software Development.** Englewood Cliffs, NJ: Prentice Hall, 2004.

COUNCILL, W. T.; HEINEMAN, G. T. **Definition of a Software Component and its Elements.** In: Heineman, G. T.; Councill, W. T. (Orgs). Component-based Software Engineering. Boston: Addison-Wesley, 2001, p. 5-20.

CYBIS, W. A; BETIOL, A. H.; FAUST, R. **Ergonomia e Usabilidade: conhecimentos, métodos e aplicações.** Novatec, 2007.

DAMKE, Elói. J; WALTER, S. Anita; SILVA, E. Damião. **A Administração é uma Ciência? Reflexões Epistemológicas acerca de sua Cientificidade.** Revista de Ciências da Administração, v. 12, n. 28, p. 125-144, set. /dez.2010.

DE BARROS, Zucule; JENNIFER, Martina. **Engenharia de Software.** 2018.

DEVMIDIA. **Atividades básicas ao processo de desenvolvimento de Software.** Disponível em: <<https://www.devmedia.com.br/atividades-basicas-ao-processo-de-desenvolvimento-de-software/5413>>. Acesso em: 22 nov. 2018.

DOCUMENTAÇÃO DO PHP. **Introdução.** Disponível em: <<http://www.php.net/manual/en/intro.pdo.php>>. Acesso em: 22 out. 2018.

DOCUMENTAÇÃO DO WAMPSEVER. **Wampserver.** Disponível em: <<http://www.http://www.wampserver.com/>>. Acesso em: 22 out. 2018.

EHLERS, U.D. **Linking strategic directions for European eLearning beyond 2010.** In: eLearningLisboa07 Conference Proceedings. Coordenação: Isabel Vilhena. Fundação para a Divulgação das Tecnologias de Informação, Tipografia Guerra, Viseu, Lisboa: 2008.

FOWLER, M.; SCOTT, K. **UML essencial um breve guia para a linguagem – padrão de modelagem de objetos.** 2. ed. Bookman, Porto Alegre, 2000.

FRIGERIO, A. (2008). **O paradigma da escolha racional: mercado regulado e pluralismo religioso.** Tempo Social, Revista de Sociologia da USP, 20(2), 17-39. Disponível em: <<http://www.revistas.usp.br/ts/article/view/12577>>. <http://dx.doi.org/10.1590/S0103-20702008000200002>>. Acesso em 6 maio 2018.

FUGGETTA, A. (2000) **“Software Process: A Roadmap”.** In: 22nd International Conference on Software Engineering (ICSE), Proceedings of the Conference on The Future of Software Engineering, pp. 25-34., New York: ACM Press.

GAMMA, E.; HELM, R.; JOHNSON, R.; VLISSIDES, J. **Design Patterns: Elements of Reusable Object-Oriented Software**. Reading, Mass.: Addison-Wesley, 1995.

GUEDES, Gilleanes TA. **UML 2-Uma abordagem prática**. Novatec Editora, 2018.

GREER, D. e CONRADI, R. (2009) “**Software project initiation and planning – an empirical study**”, IET Software, 3(5), p. 356–368

INSTITUTO de Pesquisa Datafolha. **Perfil e opinião dos evangélicos no Brasil – total da amostra**. Disponível em: <<http://datafolha.folha.uol.com.br/opiniaopublica/2016/12/1845231-44-dos-evangelicos-sao-ex-catolicos.shtml>>. Acesso em: 06 de maio 2018.

INTERNATIONAL STANDARD ORGANISATION (ISO). **Ergonomic requirements for office work with visual display terminals (VDTs)**. Part 10: Dialogue principles. Genève,1996. (ISO 9241- 10:1998).

KELLNER, M. I. (1996) "Connecting Reusable Software Process Elements and Components". In: 10th International Software Process Workshop, pp. 8-11, Dijon, France.

KESSLER, Nemuel; Câmara, Samuel. **Administração eclesiástica**. Rio de Janeiro: Casa Publicadora das Assembleias de Deus, 1987.

MAÇADA, A. C. G.; BECKER, J.; LUNARDI, G. L. **Efetividade de Conversão dos Investimentos em TI na Eficiência dos Bancos Brasileiros**. RAC. Revista de Administração Contemporânea, Curitiba-PR, v. 9, n. 1, p. 9-33, 2005.

MACHADO, Felipe Nery Rodrigues; ABREU, Mauricio Pereira de. **Projeto de Banco de Dados: uma visão prática**. 16 ed. São Paulo: Érica, 2009.

MACIEL, Ira Maria. Educação a distância. **Ambiente virtual: construindo significados**. Boletim Técnico do SENAC, v. 28, n. 3, p. 38-45, 2018.

MANUAL DO MPDF. **Documentação**. Disponível em: <<https://mpdf.github.io/>>. Acesso em 22 out. 2018.

MARASCHIN, Cleci. **A sociedade do conhecimento e a educação a distância**. CAPISANI, Dulcimira. Educação e arte no mundo digital. Campo Grande, MS: EAD/UFMS, 2000.

MARIANO R. (2008). **Usos e limites da teoria da escolha racional da religião**. **Tempo Social**, Revista de Sociologia da USP, 20(2), 41-66. Disponível em: <http://www.scielo.br/scielo.php?pid=S0103-20702008000200003&script=sci_arttext. <http://dx.doi.org/10.1590/S0103-20702008000200003>>. Acesso em 6 maio 2018.

MARTIN, D.; SOMMERVILLE, I. **Patterns of interaction: Linking ethnomethodology and design**. ACM Trans. on Computer-Human Interaction, v. 11, n. 1, p. 59-89, 2004.

MINETTO, Elton Luís. **Frameworks para Desenvolvimento em PHP - 2000**. Disponível em:

<<http://www.martinsfontespaulista.com.br/anexos/produtos/capitulos/243418.pdf>>. Acesso em 27 maio/2018.

MORER, Albert Sangrà. **E-learning value**. In: eLearningLisboa07 Conference Proceedings. Coordenação: Isabel Vilhena. Fundação para a Divulgação das Tecnologias de Informação, Tipografia Guerra, Viseu, Lisboa: 2008, p. 136.

MUGNOL, Marcio. **A educação a distância no brasil: Conceitos e fundamentos**. Revista Diálogo Educacional, [S.l.], v. 9, n. 27, p. 335-349, jul. 2009. ISSN 1981-416X. Disponível em: <<https://periodicos.pucpr.br/index.php/dialogoeducacional/article/view/3589/3505>>. Acesso em: 09 out. 2018.

MUZIO, Rubens. **O dna da liderança cristã**. Arapongas: Aleluias, 2011.
Neri, M. C., Carvalhaes, L., & Monte, S. R. S. (2011). **Novo mapa das religiões (Relatório de Pesquisa/2011)**, São Paulo, SP, Centro de Políticas Sociais, Fundação Getúlio Vargas. Recuperado de <http://www.cps.fgv.br/cps/religiao/>. Acesso em 06 maio 2018.

NIEDERAUER, Juliano. **PHP para quem conhece PHP: Recursos avançados para criação de websites dinâmicos**. São Paulo: Novatec Editora, 2008.

NIELSEN, Jakob. **Usability Engineering**. San Francisco, CA, USA: Morgan Kaufmann

OLIVEIRA Gilson. **Informatização de Igrejas** 1998. Disponível em: <<http://www.geocities.ws/prgilson/GVirtual/gvtxt06.htm>>. Acesso em: 01 de maio 2018.

OSTERWEIL, L. (1987) **"Software Process Are Software Too"**. In: 9th International Conference on Software Engineering (ICSE), Monterey, Estados Unidos, p. 2-13.

PASSOS, M., ZORZIN, P. L. G., & Rocha, D. (2011). **O que (não) dizem os números - para além das estatísticas sobre o "Novo Mapa das Religiões Brasileiro"**. Horizonte, 9(23), 690-714. Disponível em: <<http://periodicos.pucminas.br/index.php/horizonte/article/view/P.21755841.2011v9n23p690>>. Acesso em 6 maio 2018.

PAULSEN, M. **E-Learning: o papel dos sistemas de gestão da aprendizagem na Europa**.

PIERUCCI, A. F. (1996). **Liberdade de cultos na sociedade de serviços**. In A. F. Pierucci & R. Prandi (Eds.), A realidade social das religiões no Brasil (Cap. 12, pp. 275-286). São Paulo: Hucitec.

PIERUCCI, A. F. (1999, dezembro 26). **Fim da união estado-igreja amplou oferta de religiões**. Folha de São Paulo, Caderno Especial. Disponível em: <<http://www1.folha.uol.com.br/fsp/especial/fe/fe16.htm>>. Acesso em 6 maio 2018.

PIERUCCI, A. F. (2006a). **Ciências sociais e religião: a religião como ruptura**. In F. Teixeira & R. Menezes (Orgs.). As religiões no Brasil: continuidades e rupturas (Cap. 1, pp. 17-34). Petrópolis: Vozes.

PIERUCCI, A. F. (2006b). **Religião como solvente - uma aula**. Novos Estudos – CEBRAP, (75), 111-127. Disponível em: <<http://www.scielo.br/pdf/nec/n75/a08n75.pdf>>. Acesso em 6 maio 2018.

PIERUCCI, A. F. (2008). **De olho na modernidade religiosa**. Tempo Social. Revista de Sociologia da USP, 20(2), 9-16. Disponível em: <http://www.scielo.br/pdf/ts/v20n2/01.pdf>. Acesso em 06 maio 2018.

PINTO, Pedro. **Sublime Text – É este o melhor editor de texto do mundo?** Disponível em: <<https://pplware.sapo.pt/software/sublime-text-e-este-o-melhor-editor-de-texto-do-mundo/>>. Acesso em 22 out.2018.

PORTAL ADVOCACIA SERGIO MONELLO. **A Organização Religiosa, Seu Estatuto e Suas obrigações Legais**. Disponível em: <http://www.advocaciasergiomonello.com.br/sitesterceiros/adv_sergio_monello2/index.php/noticias/59-a-organizacao-religiosa-seu-estatuto-e-suas-obrigacoes-legais>. Acesso em 08 de maio de 2018.

PORTAL COMPOSER. **Introdução**. Disponível em: <<https://getcomposer.org/doc/00-intro.md>>. Acesso em 22 out. 2018.

PORTAL DEWILLIAM. **Biblioteca mpdf gerar relatórios em php**. Disponível em: <<http://www.devwilliam.com.br/php/biblioteca-mpdf-gerar-relatorios-em-php>>. Acesso em 22 out. 2018.

Portal Infomoney. **Acelere o crescimento da sua empresa em até 35% com uma plataforma ERP**. Disponível em: <<http://www.infomoney.com.br/conteudo-patrocinado/noticia/5206438/accelere-crescimento-sua-empresa-ate-com-uma-plataforma-erp>>. Acesso em: 06 de maio 2018.

PORTAL MACORATTI.NET, **Padrões de Projeto: O modelo MVC - Model View Controller**. Disponível em: <http://www.macoratti.net/vbn_mvc.htm>. Acesso em 22 out. 2018.

PORTAL MYSQL.COM. **Documentação**. Disponível em: <<https://www.mysql.com/products/workbench/>>. Acesso em 22 out. 2018.

PORTAL TECHTUDO. **Modele softwares com Astah**. Disponível em: <<https://www.techtudo.com.br/tudo-sobre/astah-community.html>>. Acesso em out. 2018.

PRESSMAN, R. **Engenharia de Software: Uma Abordagem Profissional**. 7ª Ed., São Paulo: McGraw-Hill, 2011.

PRESSMAN, R. S. (2006) **“Software Engineering: A Practitioner's Approach”**, 6th Edition, New York: McGraw-Hill, 2006. Publishers Inc, 1993.

REBRATES, Rede Brasileira do Terceiro Setor. **O Terceiro Setor**. Disponível em: <http://www.terceirosetor.org.br/quemsomos/index.cfm?page=terceiro>. Acesso em 25 de Maio 2018.

REIS, T. S; ALMEIDA, B. M. B. M. CORREIA, R. M. **Softwares de auxílio à docência no processo de Ensino-aprendizagem de Língua Portuguesa**. In: FUCHS, C; SCHWENGBER, I. L; SCHUTZ, J. A. Educação, Escola e Contemporaneidade: Questões e Diálogos. São Carlos. Pedro & João Editores. 2018. 421.

REVISTABW. **10 heurísticas de usabilidade de Nielsen**. Revista Brasileira de Web: Concursos. Disponível em: <<http://www.revistabw.com.br/concursos/10-heuristicas-de-usabilidade-de-nielsen/>>. Criado em: 21/08/2013. Última atualização: 24/07/2015. Visitado em: 28/05/2018.

REZENDE, D. **Engenharia de Software e Sistemas de Informação**. 3ª Ed., Rio de Janeiro: Brasport, 2005.

RIBEIRO, Leandro. Portal Devmidia. **O que é UML** Disponível em: <<https://www.devmedia.com.br/o-que-e-uml-e-diagramas-de-caso-de-uso-introducao-pratica-a-uml/23408>>. Acesso em 22 out. 2018.

RIVAS, Andre Ronaldo; COSTA, Ivanir; SALVETTI, Nilson. **Gestão do conhecimento aplicada à engenharia de requisitos de software: estudo de caso em uma operadora de telecomunicações**. Ciência da Informação, 2018, 47.2.

ROCHA, José Gladistone. **Arquitetura em Camadas com uso do Paradigma MVC e Processo Unificado na Programação de Software Orientado a Objetos**. TECNOLOGIAS EM PROJEÇÃO, v. 9, n. 1, p. 31-49, 2018.

RODRIGUES de Oliveira, Lucas et al. **Desenvolvimento e Avaliação de um Perfil UML para Modelagem de Jogos Educacionais Digitais**. Revista Brasileira de Informática na Educação, v. 26, n. 2, 2018.

SAMETINGER, Johannes. **Software Engineering with Reusable Components**. New York: Springer, 1997. 271p.

SANTOS, Geraldo Majela Cardoso. **Informatização de Igrejas**, 2010. Disponível em:< <https://christiansi.wordpress.com/2010/08/07/informatizacaodeigrejas/> >. Acesso em 24 maio/2018.

SEIBEL, Tiago. **O Que São Aplicações WEB?** - 2008. Disponível em: <<http://www.tiagoseibel.com.br/2008/02/o-que-so-aplicaes-web.html>>. Acesso em 26 maio/2018.

SETZER, Valdemar W; SILVA, Flávio Soares Corrêa da. **Banco de Dados: Aprenda o que são melhore seu conhecimento construa os seus**.1 ed. São Paulo: Edgard Blucher, 2005.

SHNEIDERMAN, Ben; PLAISANT, Catherine. **Designing the user interface: Strategies for effective Human-Computer Interaction**. 4th edition. Berkeley, California: Addison Wesley Longman, Inc., 2004.

SILVA, M. S. (2015). **Bootstrap 3.3.5**. Novatec Editora, Sao Paulo, SP.

SNYDER, Howard A. e RUNYON, Daniel. **Decoding the Church**, 2002 p. 61

SOMMERVILLE, Ian. **Engenharia de Software**. Tradução Ivan Bosnic e Kalinka G. de O. Gonçalves ; revisão técnica Kechi Hiramã. — 9. ed. — São Paulo: Pearson Prentice Hall, 2011.

STONER, James A. F.; FREEMAN, R. Edward. **Administração**. 5 ed. Rio de Janeiro: Prentice Hall do Brasil, 1999.

SZYPERSKI, C. **Component Software: Beyond Object-oriented Programming**, 2a ed. Harlow, Reino Unido: AddisonWesley, 2002.

SZYPERSKI, Clemens. **Component Software: beyond object-oriented programming**. Harlow: Addison-Wesley, 1999. 411p.

VIANNA, M.L. de S. **O desperdício de energia na irrigação do arroz**, Lavoura Arrozeira, Porto Alegre, v.50, n.432, p.9-11, jul. / ago. 1997.

APÊNDICES

APÊNDICE A - Questionário usado na coleta de dados da pesquisa de avaliação do sistema

PARTE I – Explicação da pesquisa

Este questionário é parte integrante de uma pesquisa da elaboração da monografia para obtenção do título de licenciado em computação, junto ao Instituto Federal de Educação, Ciência e Tecnologia do Tocantins, sob a orientação do Professor Me. Rafael Miranda Correia.

Desta forma, esta pesquisa objetiva avaliar o *software* SGE – Gestão Evangélica, desenvolvido para auxílio da gestão e desfrute dos benefícios após a implantação do sistema.

Assim, esta ferramenta de coleta de dados foi construída com base na escala de Likert de 5 ponto, como descritos a seguir: 1- Pouquíssima interativa, 2- Pouco interativa, 3- Interativa, 4- Muito interativa, 5- Muitíssima interativa.

Assim, garantimos que as informações a serem inseridas serão utilizadas exclusivamente para fins desta pesquisa, não existindo a obrigatoriedade do preenchimento da parte II deste questionário onde são encontradas as perguntas sobre a identificação dos respondentes.

*Obrigatório

PARTE II – Dados dos entrevistados

Nome: -

Telefone:

Cidade:

PARTE III – Dados da igreja

Nome da Igreja: *

—

Vínculo com a Igreja: *

- ☐ Pastor Presidente
- ☐ Pastor Local
- ☐ Secretário
- ☐ Tesoureiro
- ☐ Membro
- ☐ Congregado
- ☐ Outros

PARTE IV – Sobre o sistema

1 - A interface do sistema é fácil indutiva? *

- ☐ Pouquíssimo indutiva
- ☐ Pouco indutiva
- ☐ Indutiva
- ☐ Muito indutiva
- ☐ Muitíssimo indutiva

2 - Com base na utilização do sistema, como você classifica o grau de facilidade do entendimento do sistema? *

- ☐ Muito difícil
- ☐ Difícil
- ☐ Fácil
- ☐ Muito fácil
- ☐ Facilímo

3 - Com base na utilização do sistema, como você classifica o grau de Agilidade? *

- ☐ Pouquíssimo ágil
- ☐ Pouco ágil
- ☐ Ágil
- ☐ Muito ágil

☐ MUITÍSSIMO ágil

4 - Ainda com base na utilização do sistema, como você classifica o grau de Produtividade? *

☐ Pouquíssimo produtivo

☐ Pouco produtivo

☐ Produtivo

☐ Muito produtivo

☐ MUITÍSSIMO produtivo

5 - Continuando com a utilização do sistema, como você classifica o grau de Assertividade das informações? *

☐ Pouquíssima assertividade

☐ Pouca assertividade

☐ Assertividade

☐ Muita Assertividade

☐ MUITÍSSIMA assertividade

6 - Um sistema responsivo é quando o site automaticamente se encaixa no dispositivo do usuário (PC, celular, tablet, etc). Em uma escala de 1 a 5, como você avalia esse comportamento do sistema? *

☐ Pouquíssimo responsivo

☐ Pouco responsivo

☐ Responsivo

☐ Muito responsivo

☐ MUITÍSSIMO responsivo

7 - Qual o grau de acessibilidade do sistema? *

☐ Pouquíssimo acessível

☐ Pouco acessível

☐ Acessível

☐ Muito acessível

☐ MUITÍSSIMO acessível

8 - O desempenho do sistema atende as necessidades do serviço local? *

☐ Atende pouquíssimo

☐ Atende Pouco

☐ Atende as necessidades

☐ Atende muito

☐ Atende muitíssimo

9 - Você indicaria esse sistema para ser implantado na sua igreja? *

☐ Sim

☐ Não

☐ Não sei responder

Observação, caso marca a opção não sei responder, justificar a resposta.

10 - O Módulo EaD do sistema atende o ensino/aprendizagem da igreja? *

☐ Sim

☐ Não

☐ Não sei responder

APÊNDICE B – Dados coletados com a aplicação do questionário da pesquisa

1 - A interface do sistema é intuitiva? RESPOSTAS 1 A 5	2 - Com base na utilização do sistema, como você classifica o grau de facilidade do entendimento do sistema? RESPOSTAS 1 A 5	3 - Com base na utilização do sistema, como você classifica o grau de Agilidade? RESPOSTAS 1 A 5	4 - Ainda com base na utilização do sistema, como você classifica o grau de produtividade? RESPOSTAS 1 A 5	5 - Continuando com a utilização do sistema, como você classifica o grau de assertividade das informações? RESPOSTAS 1 A 5	6 - Um sistema responsivo é quando o site automaticamente se encaixa no dispositivo do usuário como você avalia esse comportamento do sistema? RESPOSTAS 1 A 5	7 - Qual o grau de acessibilidade do sistema? RESPOSTAS 1 A 5	8 - O desempenho do sistema atende as necessidades do serviço local? RESPOSTAS 1 A 5	9 - Você indicaria esse sistema para ser implantado na sua igreja? RESPOSTAS 1 A 3	10 - O Módulo EaD do sistema atende o ensino/aprendizagem da igreja? RESPOSTAS 1 A 3
5	3	4	5	4	5	4	5	3	3
5	5	5	4	4	4	4	4	3	3
5	4	4	5	5	5	5	5	3	3
3	3	3	3	3	3	4	4	3	3
4	3	4	4	3	5	4	4	3	3
3	3	4	5	4	4	4	5	3	3
5	5	5	5	5	5	5	5	3	3
4	5	4	5	4	5	4	5	3	3
5	4	4	5	5	5	5	5	3	3
5	5	5	5	5	5	5	5	3	3
3	3	3	2	2	3	3	3	3	3
5	3	3	3	4	5	4	3	3	3
3	3	3	3	3	3	3	3	3	3
5	4	3	4	5	4	3	3	3	3
3	4	5	3	3	5	5	5	3	3
5	5	5	5	5	5	5	5	3	3
4	3	4	4	4	4	4	4	3	3
5	5	4	4	4	5	4	4	3	3
5	3	5	5	5	5	5	5	3	3
4	4	5	4	3	5	4	4	3	3
4	4	2	3	5	5	5	3	3	3
3	4	5	4	3	5	3	5	3	3
5	5	5	5	5	5	5	5	3	3
5	5	5	5	5	5	5	5	2	3
5	4	5	5	5	5	4	4	3	3

5	4	5	5	5	5	5	5	3	3
5	5	4	5	5	5	5	5	3	3
5	5	5	5	4	5	5	4	3	3
5	5	5	5	4	4	5	5	3	3
4	4	4	5	5	5	4	5	3	3
5	5	4	5	4	5	5	5	3	3
5	5	5	4	4	5	4	5	3	3
5	5	5	5	5	5	5	5	3	3
4	4	4	4	4	5	4	5	3	3
5	5	4	5	4	5	4	5	3	3
5	5	4	5	4	5	4	4	3	3
4	4	4	5	5	5	4	5	3	3
4	4	4	5	5	5	5	5	3	3
4	5	5	4	4	5	5	5	3	3
5	5	5	5	5	5	5	5	3	3
3	3	4	4	4	5	4	5	3	3
5	5	5	4	4	5	4	5	3	3
4	4	4	4	4	5	4	4	3	3
5	5	4	5	4	5	4	5	3	3
5	5	4	5	5	5	5	4	3	3
5	5	5	5	5	5	5	5	3	3
4	5	4	5	5	5	4	5	3	3
4	4	5	5	5	5	4	5	3	3
5	4	4	4	5	5	4	5	3	3
5	4	4	5	5	5	5	5	3	3
5	5	5	5	5	5	5	5	3	3
5	5	5	5	4	5	5	4	3	3
5	5	5	5	5	5	5	5	3	3
4	5	5	4	4	5	4	5	3	3
5	5	5	5	5	5	5	5	3	3
5	5	5	5	5	5	5	5	3	3
5	5	5	5	5	5	5	5	3	3
5	5	4	4	4	5	4	5	3	3
4	4	4	4	5	5	5	5	3	3
4	5	5	5	5	5	5	5	3	3
5	5	4	5	5	4	5	5	3	3
5	5	5	4	4	5	5	5	3	3
5	4	4	4	4	5	5	4	3	3
4	4	5	5	4	5	5	4	3	3
5	3	3	5	4	5	5	4	3	3
4	5	5	5	5	4	5	5	3	3
5	5	5	5	5	4	5	5	3	3
4	5	5	5	5	5	5	5	3	3
5	5	4	4	4	5	5	5	3	3
5	5	5	5	5	5	5	5	3	3
5	5	5	5	5	5	5	5	3	3
5	3	3	3	3	4	2	3	3	2
5	5	5	5	5	5	5	5	3	3
4,541	4,388	4,375	4,500	4,375	4,791	4,486	4,625	2,986	2,986

ANEXOS

Trabalhos realizados no decorrer do curso

ANEXO A – Certificado de artigo publicado e apresentado JICE / 2016.



ANEXO B – Artigo publicado revista Sítio Novo IFTO / 2018

Desenvolvimento e validação de software de auxílio à gestão de instituições evangélicas¹

¹ Extraído da Monografia intitulada: DESENVOLVIMENTO E VALIDAÇÃO DE SOFTWARE DE AUXÍLIO À GESTÃO DE INSTITUIÇÕES EVANGÉLICAS: Uma prática a luz da Igreja Salvação em Cristo de Porto Nacional – TO., apresentada a Coordenação do Curso Superior de Licenciatura em Computação do *Campus* Porto Nacional, do Instituto Federal do Tocantins-IFTTO, como exigência à obtenção do grau de Licenciado em Computação.

Resumo – O presente trabalho aborda o desenvolvimento e validação de *software* de auxílio à gestão de instituições evangélicas: Uma prática a luz da Igreja Salvação em Cristo de Porto Nacional - TO. O software foi denominado Sistema de Gestão Evangélica – SGE, objetivando utilizá-lo como ferramenta de auxílio da gestão e desfrutar dos benefícios tecnológicos. A fim de alcançar o objetivo anterior, especificamos em: i) Desenvolver um Software web para Gestão Evangélica; ii) Utilizar o Software como ferramenta de auxílio da Gestão; iii) Desfrutar dos benefícios como: facilidade, agilidade, assertividade e maior produtividade. Desta forma, a problemática foi evidenciada pela falta de praticidade na gestão percebida pelos líderes da Igreja. Assim, levantou-se a hipótese de que o desenvolvimento do software auxiliará na gestão, garantirá os benefícios como: facilidade, agilidade, assertividade e maior produtividade. O processo de desenvolvimento foi dividido em 3 (três) etapas, a primeira, definiu os requisitos essenciais para a construção do *Software*; a segunda, foi o desenvolvimento e implantação e a terceira, foi a pesquisa com o intuito de avaliar a aceitabilidade do *software* implantado. Empregou-se uma pesquisa qualitativa com enfoque quantitativo, no universo de 72 participantes. Como instrumento de coleta dos dados, foi utilizado um questionário aplicado via *Web*. Os resultados obtidos mostram que o sistema desenvolvido atende as necessidades e melhora a gestão da instituição religiosa, pois, 99% dos pesquisados responderam que indicariam o sistema para implantação na instituição religiosa, mostrando com isso a relevância deste trabalho, pois a hipótese foi validada e os objetivos foram alcançados.

PALAVRAS CHAVES: desenvolvimento *web*, ead, gestão, igrejas, *software*

Development and validation of software to support the management of evangelical institutions

Abstract – The present work deals with the development and validation of software to aid the management of evangelical institutions: A practice in the light of the Salvation in Christ Church in Porto Nacional - TO. The software was called the Evangelical Management System - SGE, aiming to use it as a tool to help management and enjoy the technological benefits. In order to achieve the above objective, we specify in: ii) Develop a Web Software for Evangelical Management; ii) Using the Software as a Management Assistance Tool; ii) Enjoy the benefits like: ease, agility, assertiveness and increased productivity. In this way, the problem was evidenced by

²Licenciando em Computação pelo Instituto Federal do Tocantins-IFTO. Av. Tocantínia, 566 - Jardim América, Porto Nacional - TO, 77500-000, Brasil. *nalves512@gmail.com

³Professor Mestre, do Instituto Federal do Tocantins-IFTO. Coordenador Curso de Licenciatura em Computação. Av. Tocantínia, 566 - Jardim América, Porto Nacional - TO, 77500-000, Brasil. *rafaelmiranda@ifto.edu.com.br

the lack of practicality in the management perceived by the leaders of the Church. Thus, the hypothesis was raised that the development of the software will aid in the management, will guarantee the benefits as: ease, agility, assertiveness and increased productivity. The development process was divided in 3 (three) stages, the first, defined the essential requirements for the construction of the Software; the second one was the development and deployment, and the third was the research with the purpose of evaluating the acceptability of the implanted software. A quantitative qualitative research was used, in the universe of 72 participants. As a data collection instrument, a questionnaire was used via the Web. The results show that the developed system meets the needs and improves the management of the religious institution, 99% of the respondents answered that they would indicate the system for implantation in the religious institution, thus showing the relevance of this work, since the hypothesis was validated and the objectives were reached.

KEYWORDS: web development, ead, management, churches, software

Introdução

Ao analisar os problemas que acarretaram a Igreja Assembleia de Deus, Ministério da Salvação em Cristo de Porto Nacional Tocantins, a tomada de decisões equivocadas, a coleccionar informações sem qualidade, registros desencontrados, com pouca assertividade, levou ao seguinte questionamento: Quais os ganhos produtivos que a igreja terá com a construção e utilização de um Software de Gestão Evangélica?

Assim, a hipótese levantada que solucionará a problemática constitui-se, no desenvolvimento e implantação de um *software* que auxilie na gestão da ADESC, que garantirá benefícios como: facilidade; agilidade; assertividade e maior produtividade.

A partir deste momento, fez-se uma consulta na literatura a fim de conhecer os principais conceitos utilizados nesta pesquisa.

Partindo dessas primícias, *Softwares*, segundo Rivas et al, (2018), refere-se a programas de computador com documentação associada, é o produto que profissionais desta área constroem e, depois, mantêm ao longo do tempo.

A engenharia de software é uma metodologia que desenvolve soluções profissionais utilizando-se dos recursos de *software*, observando os padrões requeridos de qualidade, produtividade e efetividade conforme preconiza (BARROS e JENNIFER, 2018).

Sendo assim, o padrão escolhido para o desenvolvimento do sistema SGE foi o Modelo Visão e Controlador - MVC. Rocha (2018), diz que todas as requisições feitas pelo usuário são enviadas ao Controlador, este manipula os dados usando o

SANTOS E CORREIA (2018)

Modelo e invoca a Visão correta, de acordo com a ação executada ou com os resultados vindos do Modelo.

Nessa perspectiva, elaborou-se o seguinte Objetivo Geral: Desenvolver e validar um Software de gestão das instituições evangélicas, utilizá-lo como ferramenta de auxílio da gestão e desfrutar dos benefícios tecnológicos.

Para alcançarmos o objetivo anterior, especificamos em: i) Desenvolver um Software web para Gestão Evangélica; ii) Utilizar o Software como ferramenta de auxílio da Gestão; iii) Desfrutar dos benefícios como: facilidade, agilidade, assertividade e maior produtividade.

Em conformidade com a resposta do seguinte questionamento levantados na avaliação do software: “Você indicaria esse sistema para ser implantado na sua igreja? ”, na qual 99% dos pesquisados responderam que SIM. Assim, esta pesquisa ganha ênfase justificando a sua relevância em consonância com alta média apresentada nas respostas.

Para melhor entendimento, o presente trabalho está organizado em cinco seções: Na sessão 1 é apresentada a contextualização, os principais conceitos utilizados nesta pesquisa, a definição do problema, a hipótese levantada, os objetivos a serem alcançados, a justificativa e, por fim, a presente organização do trabalho. Na seção 2, está descrito de forma sistematizada os procedimentos metodológicos, todas as etapas do desenvolvimento deste estudo. Na seção 3, são abordadas as discussões, análises e resultados da pesquisa. Na seção 5, é apresentado as conclusões bem como as perspectivas e propostas para trabalhos futuros.

Materiais e métodos

Ferramentas e tecnologias utilizadas

Tabela 1 - Ferramentas e as tecnologias utilizadas

Ferramenta / tecnologia	Versão	Referência	Finalidade
Wampserver	1.0	http://www.wampserver.com/en/	Servidor web local
Apache	4.27		Servidor HTTP
PHP	1.9		Linguagem de programação

MySQL	M	.7.19		Gerenciador de banco de dados
QL	S	.7.19		Linguagem de banco de dados
MySQL Workbench	M	.3	https://www.mysql.com/products/workbench/	Administrador de banco de dados
Bootstrap	B	.1	https://getbootstrap.com/	Framework de desenvolvimento com HTML, CSS e JS.
Sublime Text	S	.0	https://www.sublimetext.com/	Editor de código-fonte
Composer	C	.7.2	https://getcomposer.org/	Gerenciador de dependência para PHP
MPDF	m	.0	https://github.com/mpdf/mpdf	Biblioteca PHP gerador arquivos PDF de HTML
Astah Professional	A	.2.0	http://astah.net/download	Ferramenta de modelagem UML

FONTE: Santos (2018), produção próprio autor.

Análise e levantamento de Requisitos do Sistema

A apresentação dos requisitos levantados permite entender o sistema, quais as funcionalidades, restrições e desempenho que o *software* deve fornecer (CAMPOS, 2018).

Requisitos Funcionais

A tabela 3 apresenta os requisitos funcionais previstos para o sistema, ou seja, vinculação com os casos de uso associados.

Tabela 2 - Requisitos Funcionais

D	NO	ME	DESCRIÇÃO	PR
				IORIDADE
F01	astro	Cad	O sistema deve permitir cadastrar, alterar, deletar e visualizar: campos, igrejas, membros, categoria, movimento, 3 tipos de usuário (administrador, secretário e tesoureiro).	Essencial

F02	Relatório	Relatórios de campos, igrejas, membros e financeiros.	O sistema deve permitir emissão de	Importante
F03	Curso	Cursos	O sistema deve permitir cadastro de	Importante
F04	Login	Log	O sistema deve permitir usuários cadastrados realizar o login	Essencial
F05	Senha	Senha para o e-mail do usuário	O sistema deve permitir o envio de	Importante

FONTE: Santos (2018), produção próprio autor.

Requisitos Não Funcionais

A tabela - 3 apresenta os requisitos não funcionais previstos para o sistema.

Tabela 3 – Requisitos Não Funcionais

D	NO	ME	DESCRIÇÃO	PR
				IORIDADE
F01	Interface	Inte	O sistema deverá ter uma interface	Importante
			amigável, responsiva, de fácil acesso e aprendizagem;	
F02	Portabilidade	Port	O sistema deverá ser operacionalizado	Essencial
			em qualquer sistema operacional e funcionar em qualquer navegador Web.	
F03	Segurança	Seg	3.1-Usuários logado em uma área não pode em hipótese alguma acessar outra área da qual não tenha permissão;	Essencial
			3.1 - A base de dados deve ser protegida para acesso apenas de usuários autorizados.	
F04	Confiabilidade	Con	4.1- Uma igreja não pode pertencer a mais de um campo;	Essencial
			4.2- Um membro não pode pertencer a mais de uma igreja.	
F05	Desempenho	Des	O tempo de resposta do sistema não deve ultrapassar 30 segundos.	Importante

	Reu	Os módulos do sistema deve ser	Imp
F06	sabilidade reusado		ortante
	Log	O sistema deve ser implementado em	imp
F07	ica	PHP com banco de dados SQL.	ortante

Fonte: Santos (2018), produção próprio autor.

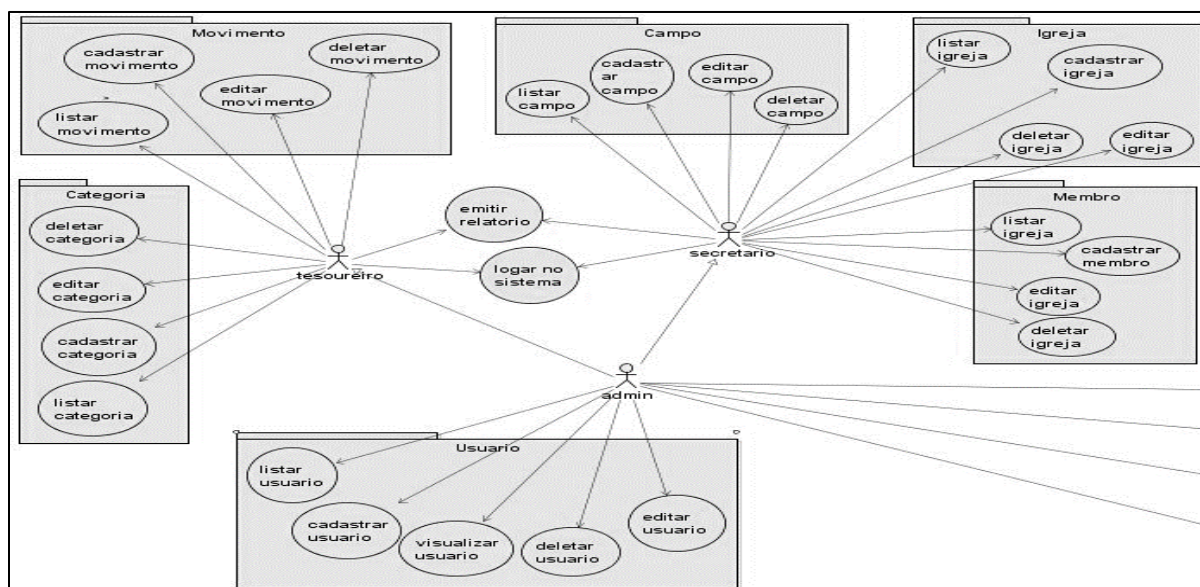
Análise do sistema SGE

Nessa etapa do desenvolvimento foram modeladas as UML (*Unified Modeling Language*). Oliveira (2018) afirma que, esta é uma linguagem de propósito geral para modelagem de software ou os diagramas de interação do sistema. A elaboração dos diagramas foi realizada com base na análise dos requisitos levantados na etapa anterior.

Diagrama Caso de Uso

No diagrama de caso de uso que é um diagrama UML que representa as funcionalidades do sistema, contendo as funcionalidades essenciais realizadas pelos seus atores, (GUEDES, 2018). Na Figura 2 tem-se o diagrama de caso de uso do *software*.

Figura 1 – Diagrama de caso de uso

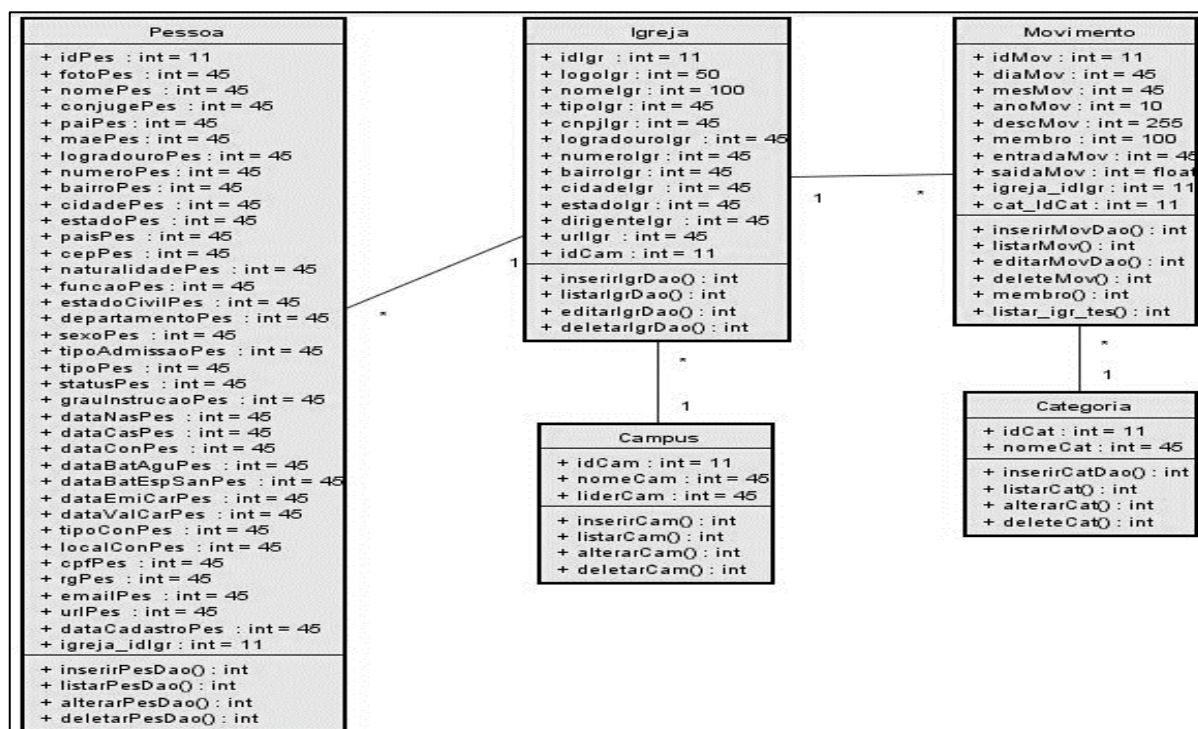


Fonte: Santos (2018), produção próprio autor.

Diagrama de Classe

Partindo da análise de requisitos foi criado o seguinte modelo de diagrama de classes, conforme afirma Guedes (2018), esse diagrama é um dos mais importantes na UML, seu principal enfoque está em permitir a visualização das classes que comporão o sistema com seus respectivos atributos e métodos. Descrito na Figura 3.

Figura 2 - Diagrama de Classe

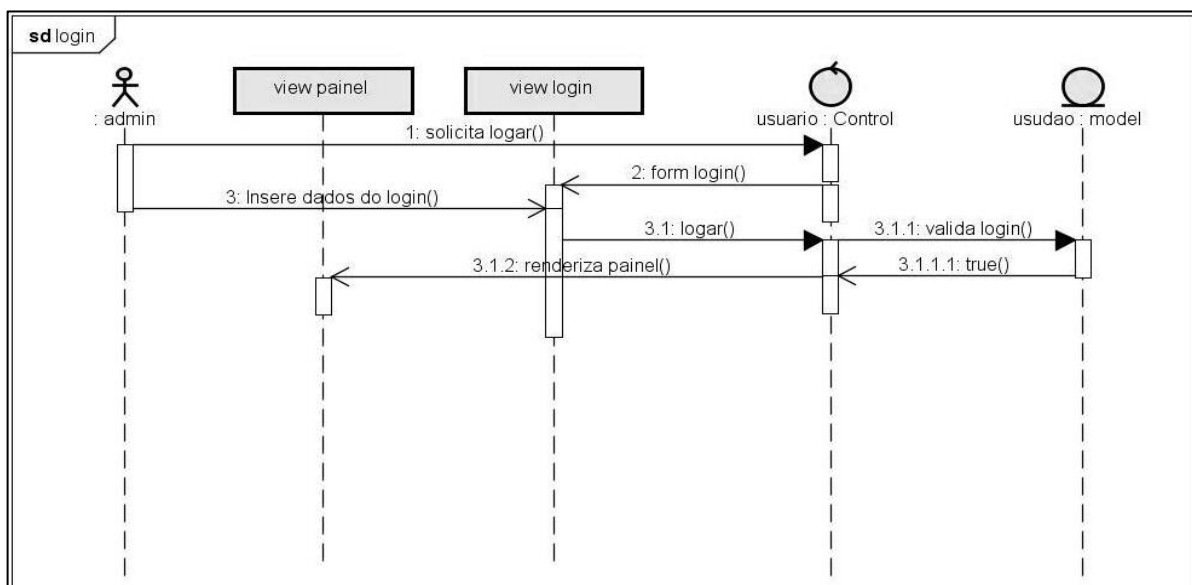


FONTE: Santos (2018), produção próprio autor

Diagrama de sequência

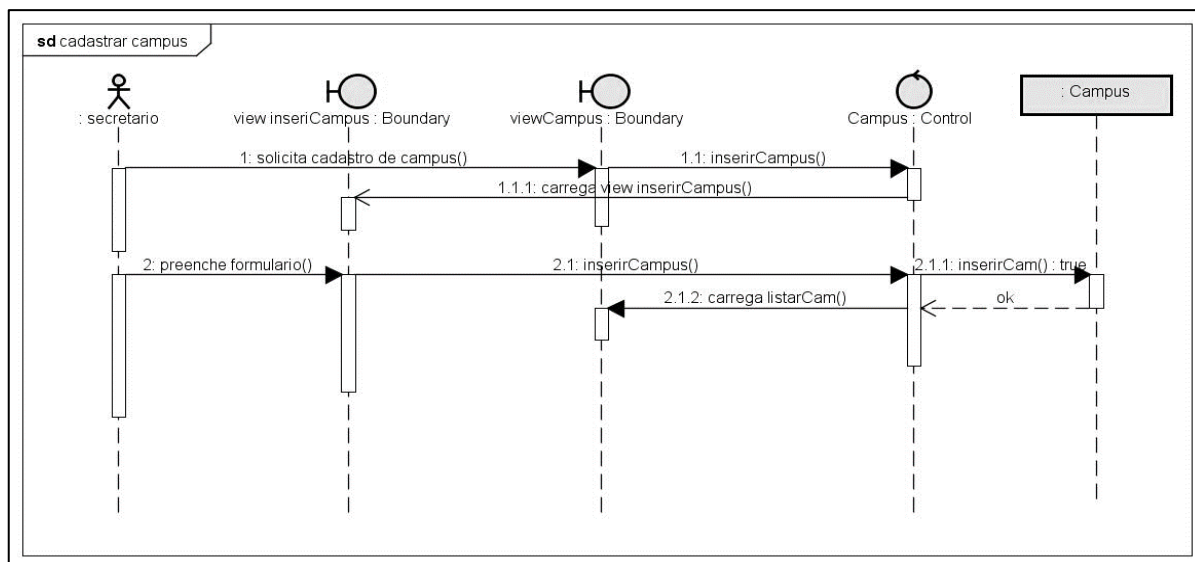
O diagrama de sequência representa a comunicação entre os objetos do sistema, (GUEDES, 2018). Nas figuras de 4 a 8 observa-se os diagramas de sequências ilustrando como é feito o login, inserir, cadastrar, alterar e listar no *software*.

Figura 3 - Diagrama de sequência, representando o login



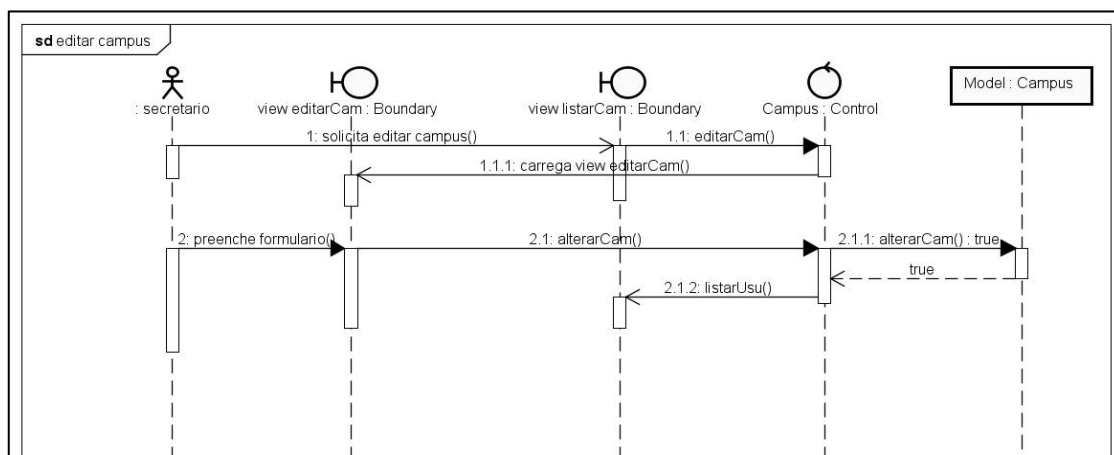
FONTE: Santos (2018), produção próprio autor.

Figura 4 - Diagrama de Sequência representando cadastro de campo



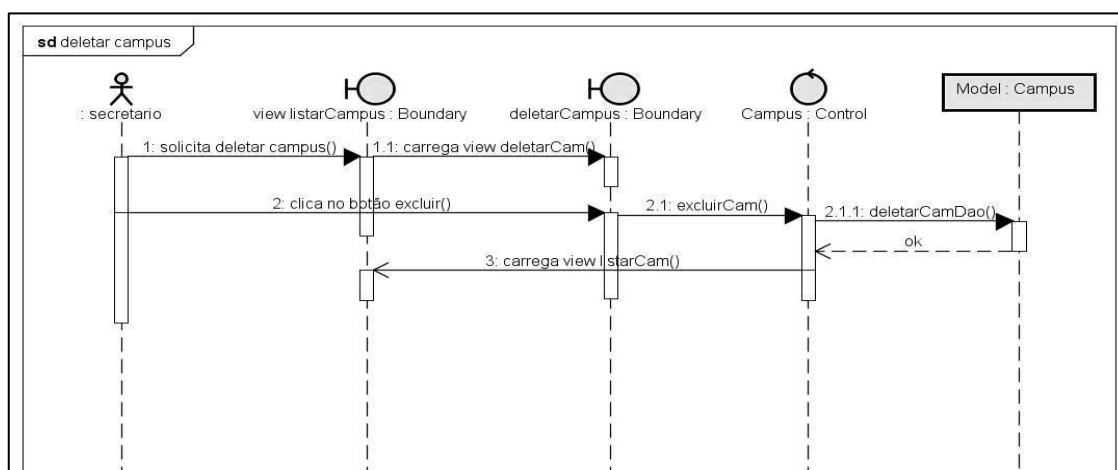
FONTE: Santos (2018), produção próprio autor.

Figura 5 - Diagrama de sequência de edição de dados do campus



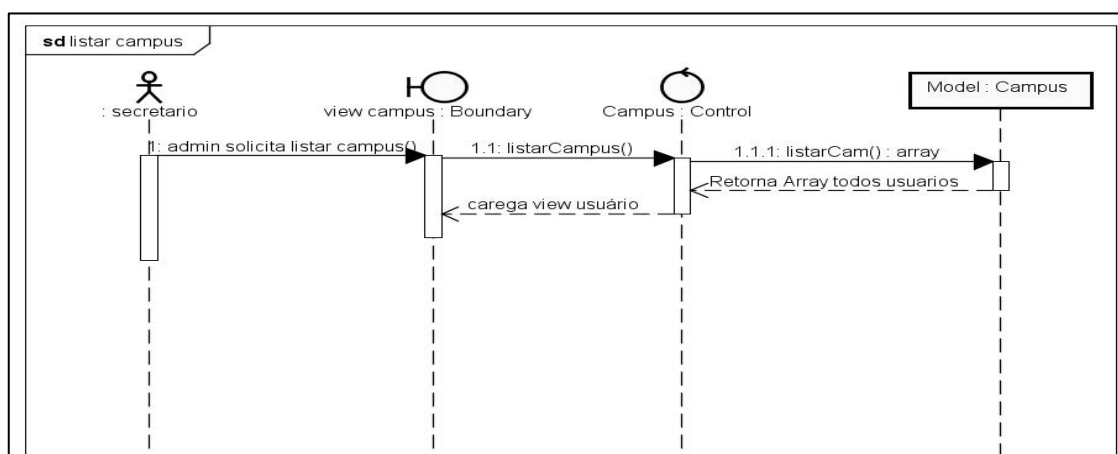
FONTE: Santos (2018), produção próprio autor

Figura 6 - Diagrama de sequência de deleção de dados do campus



FONTE: Santos (2018), produção próprio autor

Figura 7 - Diagrama de sequência de lista de dados do campus



FONTE: Santos (2018), produção próprio autor

Implementação

Implementação de software, segundo Portal Devmedia (2007) é um conjunto de atividades organizadas, usadas para definir, desenvolver, testar e manter um software.

Toda a codificação utilizada no desenvolvimento da aplicação foi salva em arquivos para melhor organização, os arquivos foram agrupados por pastas, classificadas por funcionalidades. A estrutura da organização pode ser visualizada na figura 9.

Figura 8 - Estrutura da Organização



Fonte: Santos (2018), produção próprio autor

Telas do Sistema

Nas figuras de 9 a 12 a seguir apresentam as telas do *software* SGE

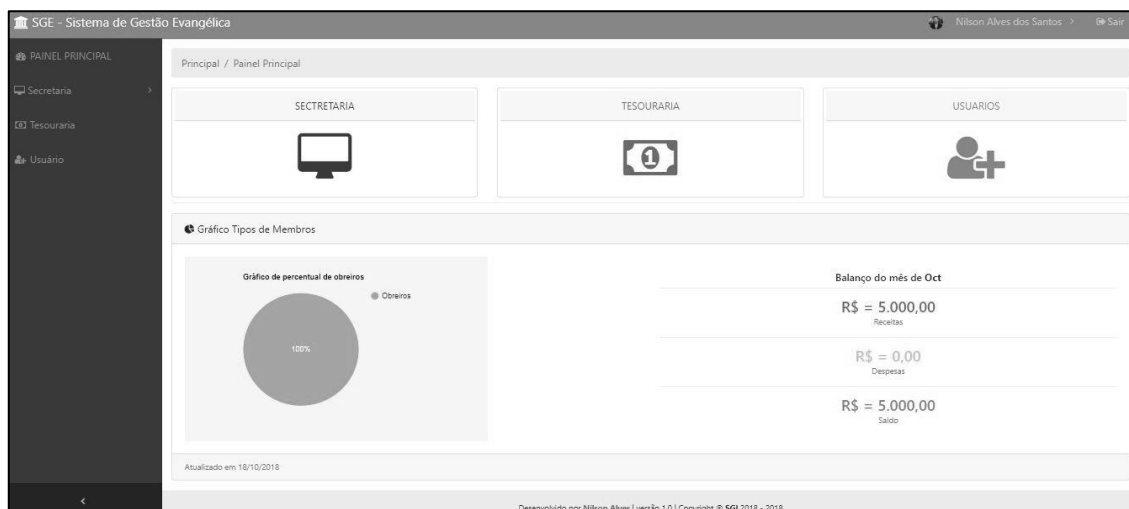
Figura 9 - Tela de login SGE



SANTOS E CORREIA (2018)

FONTE: Santos (2018), produção próprio autor

Figura 10 - Tela painel principal



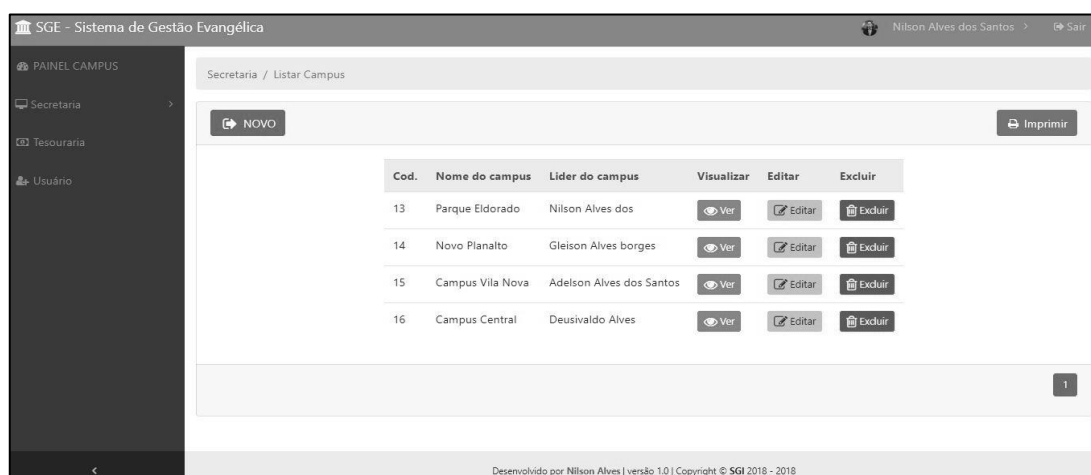
Fonte: Santos (2018), produção próprio autor.

Figura 11 - Tela painel secretaria



FONTE: Santos (2018), produção próprio autor

Figura 12 - Tela painel campus



FONTE: Santos (2018), produção próprio autor.

Revista Sítio Novo – Vol. 2 – Ano 2018

Avaliação do software

Para avaliar a qualidade do sistema foi desenvolvida uma pesquisa. Para tanto, confeccionou-se um formulário utilizando a plataforma do google, formulário *Google Form* e utilizado como instrumento de coleta de dados, os links foram enviados via e-mail, seguido de uma nota explicativa, onde, informava a natureza da pesquisa.

Em primeiro momento as coletas dos dados foram feitas em planilhas eletrônicas do pacote *Google Docs*, no segundo momento os dados foram salvos em planilhas do Microsoft Excel para facilitar a análise e tabulação dos dados, em terceiro momento foi aplicado os métodos estatísticos para a tabulação dos dados. E por fim, os resultados foram impressos em gráficos gerados pela ferramenta utilizada, conforme apresentados nos resultados e discursões.

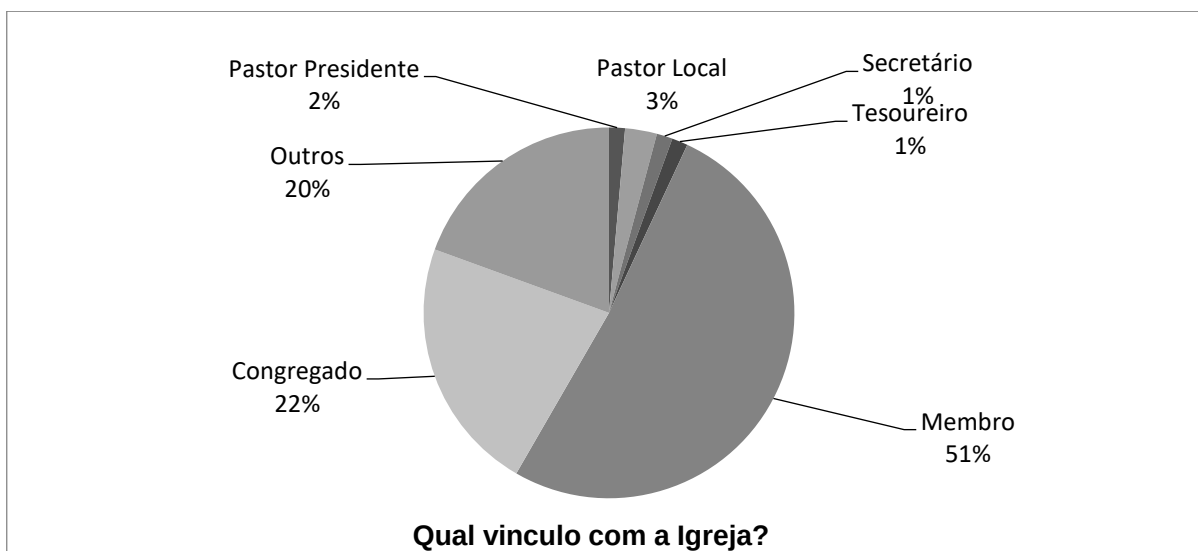
Resultados e discussão

Neste item são apresentados os resultados e análises obtidas com a pesquisa que avalia a aceitação do software SGE.

Do universo da pesquisa de 72 participantes, 1 pastor presidente, 2 pastores locais, 1 secretário, 1 tesoureiro 37 membros, 16 congregados e 14 classificados como outros. Desse total de 72, todos responderam ao questionário da pesquisa.

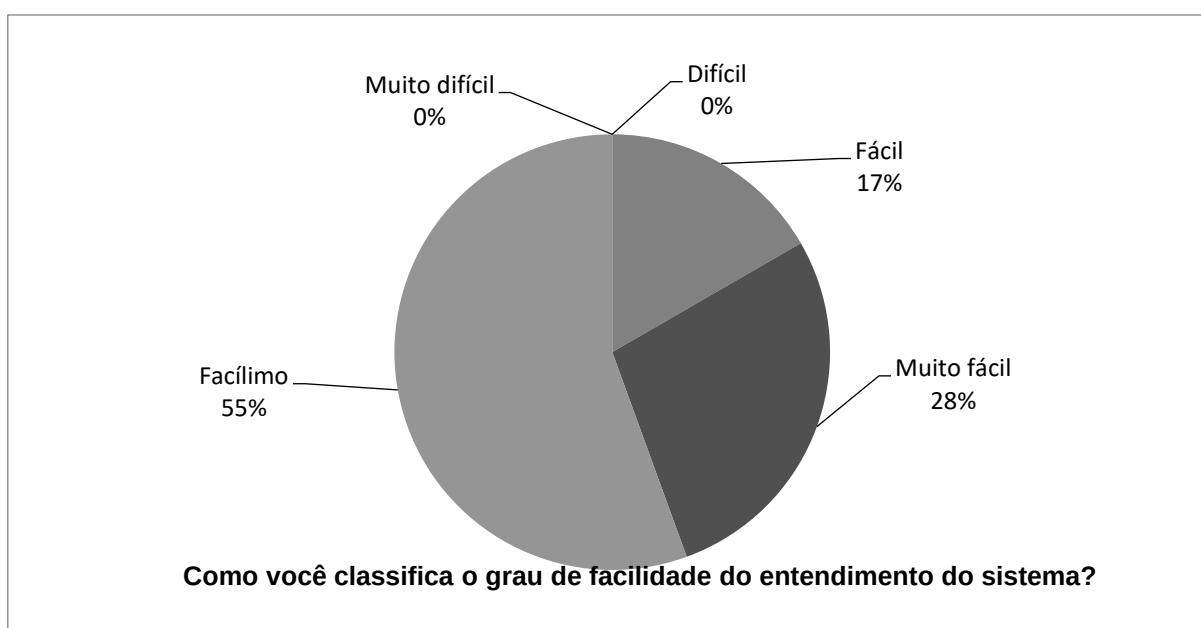
A partir da figura 16 a 21 apresenta o através dos gráficos tipo pizza os resultados da pesquisa aplicada.

Figura 13 - Vínculo com a Igreja



FONTE: Santos (2018), produção próprio autor

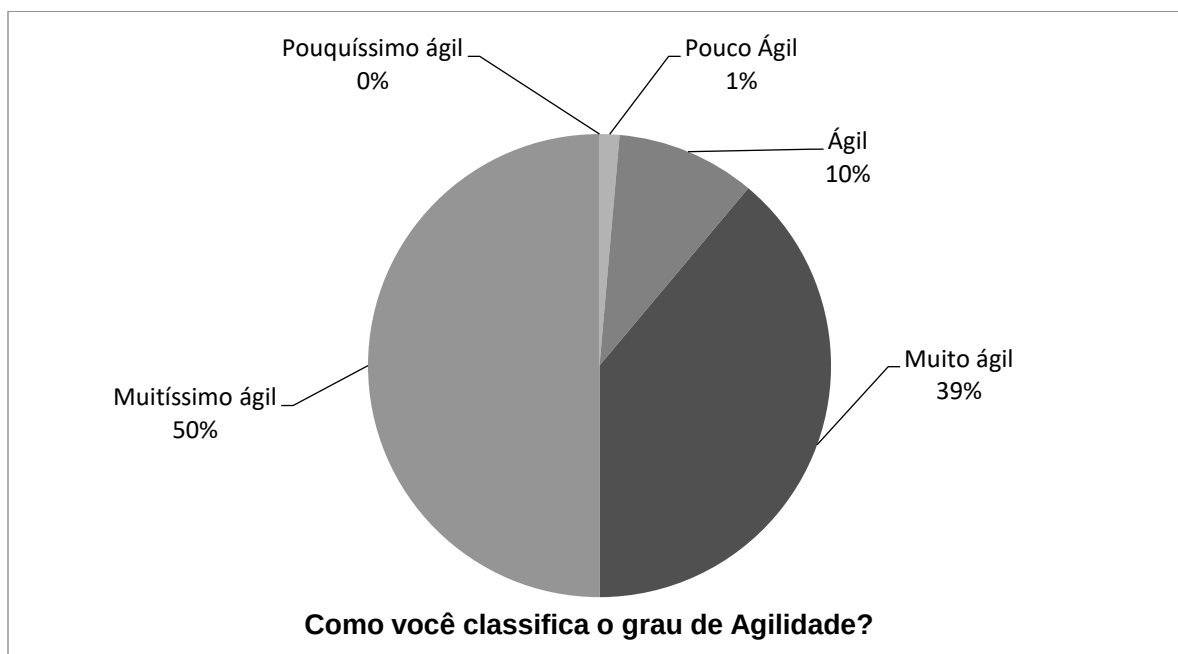
Figura 14 - Grau de facilidade do entendimento do sistema



FONTE: Santos (2018), produção próprio autor

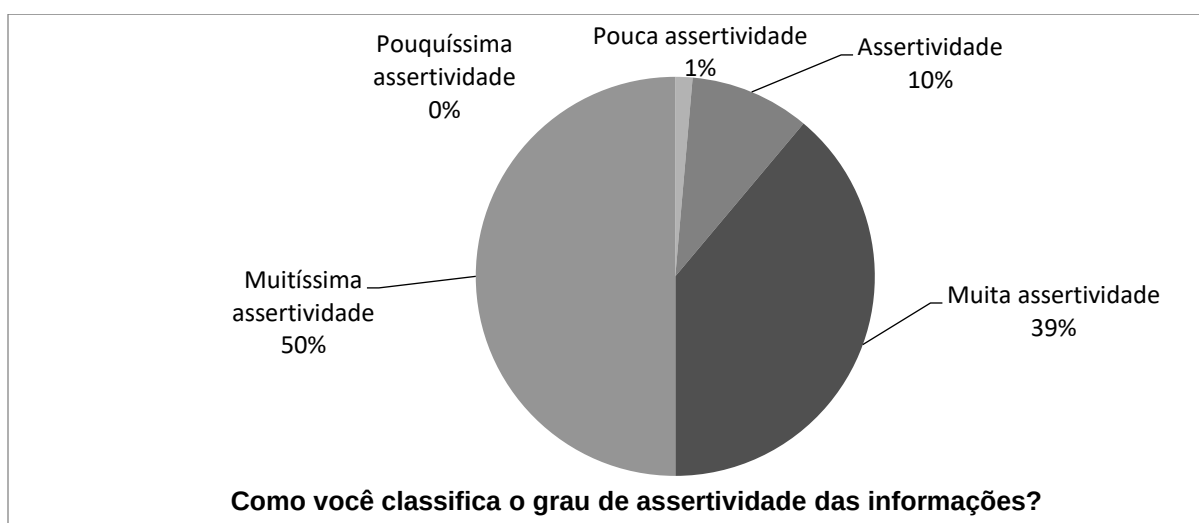
SANTOS E CORREIA (2018)

Figura 15 - Grau de Agilidade



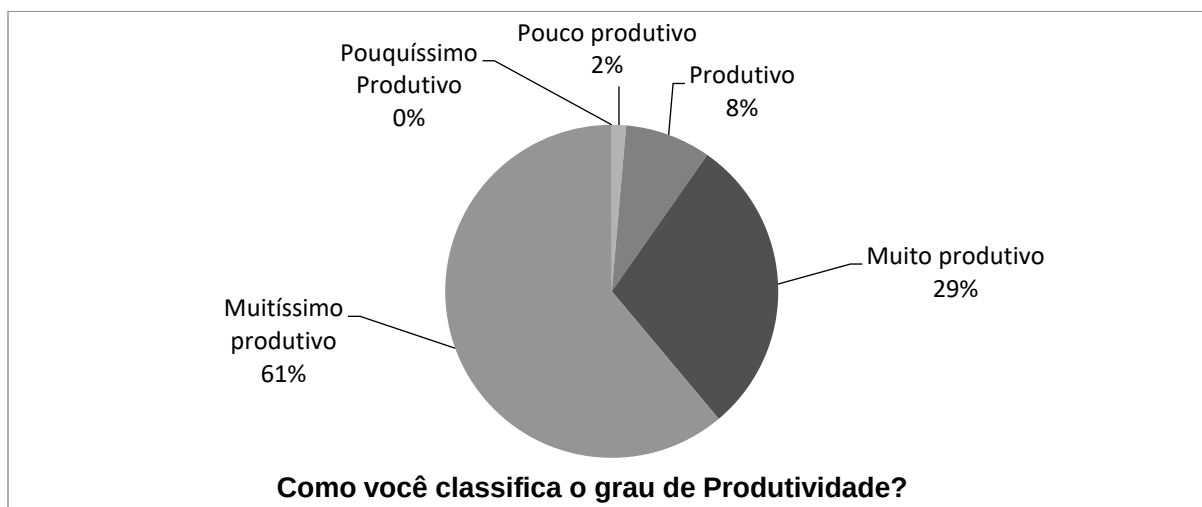
FONTE: Santos (2018), produção próprio autor

Figura 16 - Grau de assertividade das informações



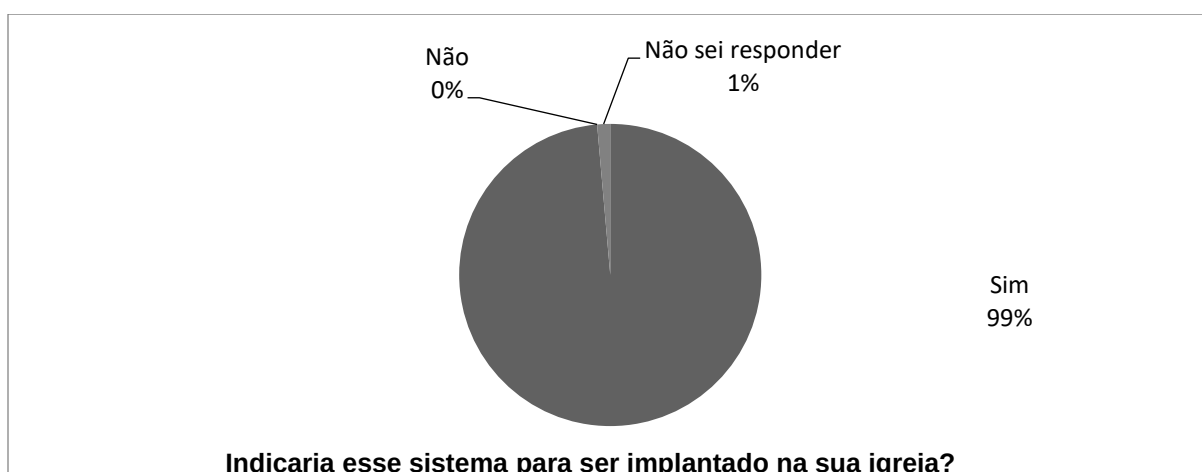
FONTE: Santos (2018), produção próprio autor

Figura 17 - Grau de produtividade do Sistema



FONTE: Santos (2018), produção próprio autor

Figura 18 - Implantação do sistema



FONTE: Santos (2018), produção próprio autor

Conclusões

Diante dos resultados obtidos com a realização deste estudo, nota-se que o *software* foi avaliado positivamente pelos pesquisados, desta forma, este trabalho se propôs a desenvolver um *software* denominado Sistema de Gestão Evangélica – SGE, utilizá-lo como ferramenta de auxílio da Gestão Igreja Evangélica Assembleia de Deus, Ministério da Salvação em Cristo, *Campus* de Porto Nacional – TO., e desfrutar dos benefícios facilidade, agilidade, assertividade e maior produtividade.

Assim, o sistema atende as atuais necessidades, alcançando os objetivos e a hipótese levantada foi validada. Pois, de acordo com os dados da pesquisa, 99%

SANTOS E CORREIA (2018)

dos pesquisados afirmaram que implantariam o sistema, esses resultados podem contribuir diretamente, de forma positiva, no processo de avaliação do sistema.

Como trabalhos Futuros propõe-se realizar a recuperação de senhas, sendo estas enviadas via e-mail do usuário cadastrado no sistema. Além do mais poderiam ser desenvolvidos novos módulos que surgissem com as necessidades e fazer incorporações a outros sistemas.

Referências

CAMPOS, Jorge Murilo Raniero; LEAL, Gislaine Camila Lapasiani. **PROPOSTA DE PROCESSO PARA GERENCIAMENTO DE REQUISITOS NO DESENVOLVIMENTO DE SOFTWARE**. Trabalhos de Conclusão de Curso do DEP, v. 13, n. 1, 2018.

DE BARROS, Zucule; JENNIFER, Martina. **Engenharia de Software**. 2018.

DEVMIDIA. **Atividades básicas ao processo de desenvolvimento de Software**. Disponível em: <<https://www.devmedia.com.br/atividades-basicas-ao-processo-de-desenvolvimento-de-software/5413>>. Acesso em: 22 nov. 2018.

GUEDES, Gilleanes TA. **UML 2-Uma abordagem prática**. Novatec Editora, 2018.

PRESSMAN, R. **Engenharia de Software: Uma Abordagem Profissional**. 7ª Ed., São Paulo: McGraw-Hill, 2011.

PRESSMAN, R. S. (2006) **“Software Engineering: A Practitioner's Approach”**, 6th Edition, New York: McGraw-Hill, 2006. Publishers Inc, 1993.

RIVAS, Andre Ronaldo; COSTA, Ivanir; SALVETTI, Nilson. **Gestão do conhecimento aplicada à engenharia de requisitos de software: estudo de caso em uma operadora de telecomunicações**. Ciência da Informação, 2018, 47.2.

ROCHA, José Gladistone. **Arquitetura em Camadas com uso do Paradigma MVC e Processo Unificado na Programação de Software Orientado a Objetos**. TECNOLOGIAS EM PROJEÇÃO, v. 9, n. 1, p. 31-49, 2018.

RODRIGUES de Oliveira, Lucas et al. **Desenvolvimento e Avaliação de um Perfil UML para Modelagem de Jogos Educacionais Digitais**. Revista Brasileira de Informática na Educação, v. 26, n. 2, 2018.

SOMMERVILLE, Ian. **Engenharia de Software**. Tradução Ivan Bosnic e Kalinka G. de O. Gonçalves ; revisão técnica Kechi Hirma. — 9. ed. — São Paulo: Pearson Prentice Hall, 2011.



ANEXO C – Artigo publicado na Revista Espiral / 2018.

**DESENVOLVIMENTO E VALIDAÇÃO DE SOFTWARE DE QUALIFICAÇÃO
OBREIROS DE INSTITUIÇÕES EVANGÉLICAS: Uma prática a luz da Igreja
Salvação em Cristo de Porto Nacional – TO**

***DEVELOPMENT AND VALIDATION OF QUALIFICATION SOFTWARE WORKERS
OF EVANGELICAL INSTITUTIONS: A practice in the light of the Salvation in
Christ Church in Porto Nacional - TO***

Nilson Alves dos Santos¹, Rafael Correia Miranda²

¹Acadêmico do Curso de Licenciatura em Computação – IFTO. Campus de Porto Nacional. E-mail: nalves512@gmail.com

² Professor Me. do Curso de Licenciatura em Computação – IFTO. Campus de Porto Nacional. E-mail: rafaelmiranda@ifto.edu.com.br

RESUMO

O presente trabalho aborda o desenvolvimento e validação de *software* de qualificação de obreiros de instituições evangélicas: Uma prática a luz da Igreja Salvação em Cristo de Porto Nacional - TO. O *software* foi denominado Sistema de Qualificação de Obreiros – SQO, objetivando utilizá-lo como ferramenta qualificação de obreiros a distância. Desta forma, a problemática foi evidenciada pela falta de obreiros qualificados para consagração, percebida pelos líderes da Igreja. Assim, levantou-se a hipótese de que o desenvolvimento de uma *portal web* de cursos a distância, garantirá obreiros qualificados para consagração. O processo de desenvolvimento foi dividido em 3 (três) etapas, a primeira, definiu os requisitos essenciais para a construção do *Software*; a segunda, foi o desenvolvimento e implantação e a terceira, foi a pesquisa com o intuito de avaliar a aceitabilidade do *software* implantado. Empregou-se uma pesquisa qualitativa com enfoque quantitativo, no universo de 72 participantes. Como instrumento de coleta dos dados, foi utilizado um questionário aplicado via *Web*. Os resultados obtidos mostram que o sistema desenvolvido atende as necessidades e melhora a gestão da instituição religiosa, pois, 99% dos pesquisados responderam que indicariam o sistema para implantação na instituição religiosa, mostrando com isso a relevância deste trabalho.

PALAVRAS CHAVES: Desenvolvimento *web*. *Software*. EaD. Obreiro. Igrejas.



ABSTRACT

The present work proposes to develop and validate software to support evangelical programs: A practice in the light of the Salvation in Christ Church in Porto Nacional - TO. The software was denominated System of Qualification of Workers - SQO, aiming the use of a tool of evaluation of workers of long distance. In this way, a problem was evidenced by the lack of objections to the consolidation, by the leaders of the Church. Thus, a hypothesis has emerged that the development of a web portal of long-term courses, guarantors of workers have to be consecrated. The development process was divided into 3 (three) phases, the first defining the essential requirements for the construction of the Software; The second was the development and deployment, and the third was a survey to evaluate the acceptability of the implanted software. A quantitative qualitative research was used, with no universe of 72 participants. As a tool for data collection, Web access was evaluated. The results of the managers are those of the developed system and the needs of the management of the religious institution, since, 99% of the respondents answered that indicate the system for the implantation of the organization religious, showing the importance of this work.

Keywords: *Web development. Software. EaD. Worker. Churches.*

INTRODUÇÃO

A sociedade muda, evolui, avança de forma inexorável. Os fantásticos avanços da tecnologia e da ciência contemporâneas são incorporados, com rapidez, pelas mais diversas organizações humanas, que para Oliveira (1998), as quais se modernizam a proporção em que passam a dispor de novas ferramentas em seu arsenal de trabalho. As que não o fizerem, correrão o risco de serem tragadas no caudal da competitividade.

Ainda, segundo Oliveira (1998), as igrejas estão inseridas nesse mesmo contexto sócio organizacional. O microcomputador é apenas uma gota d'água nesse lago profundo. Porém, quem desprezá-lo terá grandes dificuldades em manter-se no cenário que se descortina.

Desta forma, o trabalho aqui apresentado descreve a ferramenta de qualificação de obreiros, no que se refere à gestão dos dados cadastrais dos envolvidos como cadastro de cursos, alunos, materiais didáticos, provas, emissão de certificados de conclusão de cursos e carteirinhas de estudantes bem como apresentar os passos utilizados na sua elaboração e validação.



Devido a percepção dos líderes, que sem uma plataforma de cursos a distância, não seria possível qualificarem seus obreiros para consagração aos cargos disponíveis na Igreja Assembleia de Deus Salvação em Cristo – ADESC.

Diante disso, elaborou-se o seguinte questionamento: A construção e utilização de um portal de ensino a distância atenderá o ensino e aprendizagem dos obreiros e membros da Igreja Assembleia de Deus Salvação em Cristo – ADESC?

Nessa perspectiva, elaborou-se o seguinte Objetivo Geral: Desenvolver e validar um portal *web* de ensino a distância, utilizá-lo como ferramenta qualificação de obreiros.

Para alcançarmos o objetivo anterior, especificamos em: i). Desenvolver um portal de curso a distância; ii). Implantar o sistema via *web*; iii). Utilizar a plataforma para qualificação de obreiros.

Assim, a hipótese levantada que solucionará a problemática constitui-se, no desenvolvimento e implantação de um software que auxilie processo de ensino e aprendizagem da ADESC.

Esse trabalho se justifica devido informações do portal *InfoMoney* (2016), diz, que pesquisa desenvolvida pelo IDG *Research Services* mostrou que empresas e instituições que se utilizam de sistemas de gestão qualificação de seus funcionários / associados tem um nível de crescimento 35% mais rápido e 10% a mais de eficiência do que aquelas que não tem acesso a mesma ferramenta.

Devido as facilidades do uso desta ferramenta permitem aos líderes uma visão crítica e realista sobre a perfil e qualificação de seus membros e obreiros.

Em conformidade com as respostas dos seguintes questionamentos levantados na avaliação do software: Perguntou-se se o sistema EaD atende o ensino e aprendizagem da igreja, 99% dos pesquisado responderam que sim, “Você indicaria esse sistema para ser implantado na sua igreja? ”, na qual 99% dos pesquisados responderam que SIM. Assim, esta pesquisa ganha ênfase justificando a sua relevância em consonância com alta média apresentada nas respostas.

Para melhor entendimento, o presente trabalho está organizado em cinco seções: Na sessão 1 é apresentada a contextualização, o tema, a definição do problema, os objetivos a serem alcançados, a hipótese levantada, a justificativa para realização do presente estudo e, por fim, a presente organização do trabalho. Na



seção 2, os principais conceitos utilizados nesta pesquisa. Na seção 3, está descrito de forma sistematizada os procedimentos metodológicos, todas as etapas do desenvolvimento deste estudo. Na seção 4, são abordadas as discussões, análises e resultados da pesquisa. Na seção 5, é apresentado as conclusões as quais se chegaram após o presente estudo, bem como as perspectivas e propostas para trabalhos futuros.

DESENVOLVIMENTOS DO SOFTWARE WEB

Softwares, segundo Sommerville (2011), refere-se a programas de computador com documentação associada.

Na ótica de Pressman (2011), *Software* é o produto que profissionais desta área constroem e, depois, mantêm ao longo do tempo.

Engenharia de Software

Na visão de Rezende (2005), a engenharia de software é uma metodologia que desenvolve soluções profissionais utilizando-se dos recursos de *software*, observando os padrões requeridos de qualidade, produtividade e efetividade.

Para Carvalho e Chiossi (2001), a Engenharia pode ser compreendida como uma disciplina que reúne metodologias, métodos e ferramentas a serem utilizados, desde a percepção do problema até o momento que o sistema desenvolvido deixa de ser operacional, visando resolver problemas inerentes ao processo de desenvolvimento e ao produto de software.

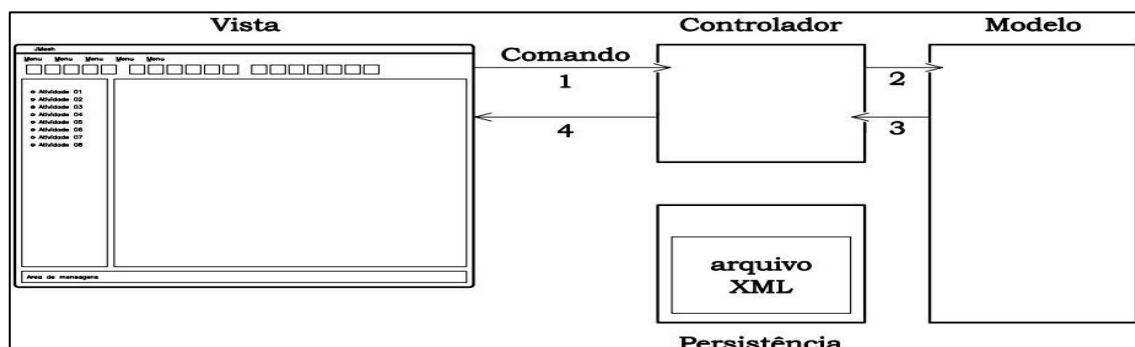
Ainda em Szyperski (1999) vamos encontrar o seguinte esclarecimento, sistemas definidos através da composição de componentes permitem que sejam adicionadas, removidas e substituídas partes do sistema sem a necessidade de sua completa substituição. Com isso, CBSE auxilia na manutenção dos sistemas de software, por permitir que os sistemas sejam atualizados através da integração de novos componentes e/ou atualização dos componentes já existentes.

Padrão MVC

Já o padrão escolhido para o desenvolvimento da ferramenta SGE foi padrão MVC. Como pode ser visto na figura 1, nesse padrão, o fluxo das informações para realização de cada tarefa ocorre em quatro etapas segundo Booch et al. (2000).



Figura 1 - Arquitetura de aplicações Web usando o padrão MVC



FONTE: Booch et al. (2000).

Em síntese, todas as requisições feitas pelo usuário são enviadas ao *Controller*. Este manipula os dados usando o *Model* e invoca a *View* correta, de acordo com a ação executada ou com os resultados vindos do *Model* (BOOCH E TAL. (2000).

Ferramentas e tecnologias utilizadas

Tabela 1 - Ferramentas e as tecnologias utilizadas

Ferramenta / tecnologia	Versão	Referência	Finalidade
Wampserver	3.1.0	http://www.wampserver.com/en/	Servidor web local
Apache	2.4.27		Servidor HTTP
PHP	7.1.9		Linguagem de programação
MySQL	5.7.19		Gerenciador de banco de dados
SQL	5.7.19		Linguagem de banco de dados
MySQL Workbench	6.3	https://www.mysql.com/products/workbench/	Administrador de banco de dados
Bootstrap	4.1	https://getbootstrap.com/	Framework de desenvolvimento com HTML, CSS e JS.
Sublime Text	3.0	https://www.sublimetext.com/	Editor de código-fonte
Composer	1.7.2	https://getcomposer.org/	Gerenciador dependência para PHP
mPDF	7.0	https://github.com/mpdf/mpdf	Biblioteca PHP gerador arquivos PDF de HTML
Astha professional	7.2.0	http://astah.net/download	Ferramenta de modelagem UML

FONTE: Santos (2018), produção próprio autor.

EDUCAÇÃO A DISTÂNCIA -

Esta seção apresenta uma abordagem geral sobre Educação a Distância EaD, no entanto, o Portal do Ministério da Educação afirma que a Educação a Distância – EaD, é a modalidade educacional na qual alunos e professores estão



separados, física ou temporalmente e, por isso, faz-se necessária a utilização de meios e tecnologias de informação e comunicação.

Essa modalidade é regulada por uma legislação específica e pode ser implantada na educação básica (educação de jovens e adultos, educação profissional técnica de nível médio) e na educação superior.

Já na ótica de MUGNOL (2009), a Educação a Distância ao longo de sua história traçou uma trajetória de avanços e retrocessos. No início do século XX, tornou-se uma modalidade de ensino capaz de atender a todos os níveis, incluindo programas formais de ensino, aqueles que oferecem diplomas ou certificados e programas de caráter não formais, cujo objetivo é oferecer capacitação para a melhoria no desenvolvimento das atividades profissionais.

Segundo Paulsen (2002), a modalidade de ensino à distância, baseada na Web, caracteriza-se pela separação física entre o docente e o discente; utilização de uma rede computadorizada para apresentar os conteúdos; e comunicação bidirecional para que os discentes se comuniquem entre si e com o docente.

Nessa perspectiva, Maciel (2018) descreve que na modalidade de EaD o distanciamento geográfico e o isolamento relativo do aprendiz – característicos da modalidade – podem ampliar a distância entre os professores e aprendizes e gerar a tão frequente evasão. Por outro lado, estudos sinalizam que o ambiente virtual pode gerar aproximações e uma convivência tão próxima quanto a presencial, conforme afirma Maraschi (2000) que as tecnologias na realidade têm a função de eliminar a distância ou construir outras interações diferentes da presencial. Ao possibilitarem a suspensão das distâncias espaciais permitem a convivência e a diversidade identitária.

Morer (2008) defende que, na verdade, a EaD bem-sucedida reside no equilíbrio do incremento à qualidade na educação e no *design* instrucional: ou seja, a direção da educação versus a direção da tecnologia. Não é possível considerar um e desdenhar o outro. A inovação em EaD a ser questionada conduz ao fato de que o sistema deve ser aberto, bem desenhado em relação ao design, adaptativo e flexível, com menos barreiras e mais inclusão, como enfatiza Ehlers (2008 p. 17). “Podemos promover acesso a todos, melhorar o uso de habilidades e promover qualidade na usabilidade”.



Assim, as plataformas de ambientes virtuais a distância adquire real importância, pois o sucesso da EaD que elas buscam contemplar está visceralmente ligado à sua construção. Neste processo é importante que o aluno alcance a produção de conhecimento significativo, onde o conhecimento se incorpore em seu mundo intelectual e vivencial.

PROCEDIMENTOS METODOLÓGICOS

Análise e levantamento de Requisitos do Sistema

A apresentação dos requisitos levantados permite entender o sistema, especificar as funcionalidades que se espera que o sistema tenha.

Requisitos Funcionais

A tabela 3 apresenta os requisitos funcionais previstos para o sistema, ou seja, vinculação com os casos de uso associados.

Tabela 2 - Requisitos Funcionais

ID	NOME	DESCRIÇÃO	PRIORIDADE
RF01	Cadastro	O sistema deve permitir cadastrar, alterar, deletar e visualizar: alunos	Essencial
RF03	Curso	O sistema deve permitir cadastro de cursos	Importante
RF04	Matrícula	O sistema deve permitir matrículas de alunos em cursos	Essencial
RF05	Prova	O sistema deve gerar prova	Importante
RF06	Resultado	O sistema deve computar e gerar o resultado percentual da prova	Essencial
RF07	Certificado	O sistema deve permitir a emissão e impressão de certificado	Importante
RF08	Login	O sistema deve permitir usuários cadastrados realizar o login	Essencial

Fonte: Santos (2018), produção próprio autor.



Requisitos Não Funcionais

A **tabela - 3** apresenta os requisitos não funcionais previstos para o sistema.

Tabela 3 – Requisitos Não Funcionais

ID	NOME	DESCRIÇÃO	PRIORIDADE
NF01	Interface	O sistema deverá ter uma interface amigável, responsiva, de fácil acesso e aprendizagem;	Importante
NF02	Portabilidade	O sistema deverá ser operacionalizado em qualquer sistema operacional e funcionar em qualquer navegador <i>Web</i> .	Essencial
NF03	Segurança	3.1-Usuários logado em uma área não pode em hipótese alguma acessar outra área da qual não tenha permissão; 3.2 - A matrículas nos cursos só será com usuário autenticado; 3.3 - A base de dados deve ser protegida para acesso apenas de usuários autorizados.	Essencial
NF04	Confiabilidade	4.1- O usuário não poderá se inscrever duas vezes no mesmo curso;	Essencial
NF05	Desempenho	5.2 - O tempo de resposta do sistema não deve ultrapassar 30 segundos.	Importante
NF06	Reusabilidade	6.1 Os módulos do sistema deve ser reusado	Importante
NF07	Logica	O sistema deve ser implementado em PHP com banco de dados SQL.	Importante

Fonte: Santos (2018), produção próprio autor.

Análise do sistema SGE

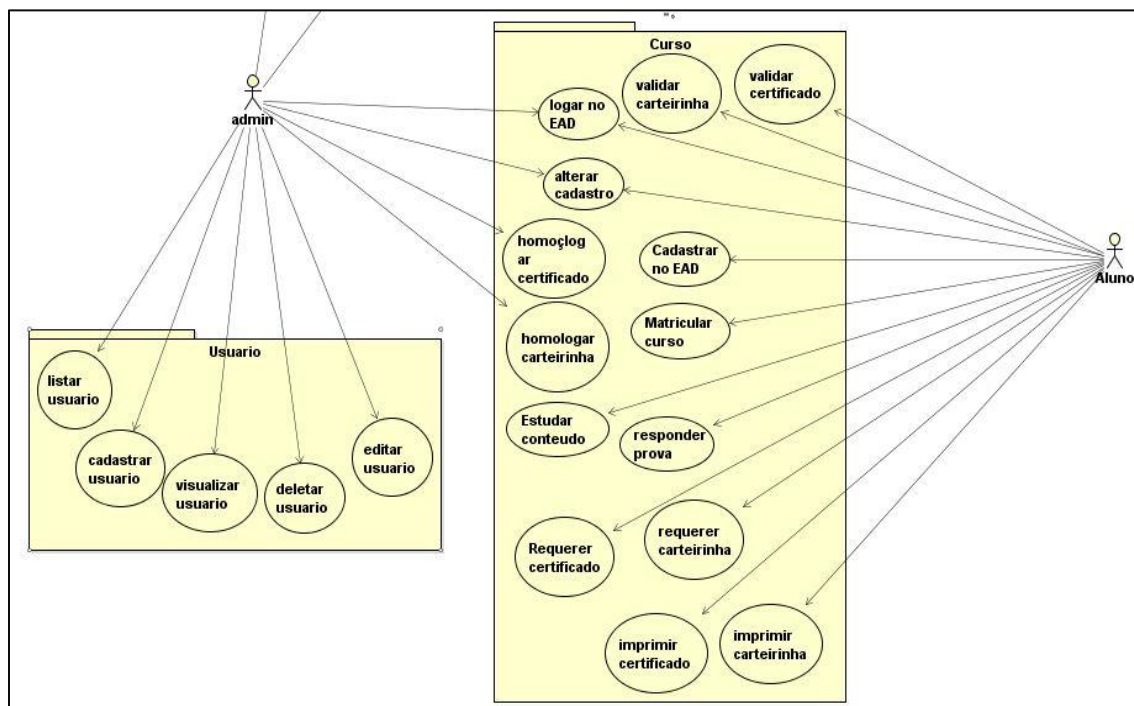
Nessa etapa do desenvolvimento foram modeladas as UML ou os diagramas de interação do sistema, a elaboração dos diagramas foi realizada com base na análise dos requisitos levantados na etapa anterior.

Diagrama Caso de Uso

No diagrama de caso de uso que é um diagrama UML que representa as funcionalidades do sistema, contendo as funcionalidades essenciais realizadas pelos seus atores. Na Figura 2 tem-se o diagrama de caso de uso do *software*.



Tabela 2 – Diagrama de caso de uso

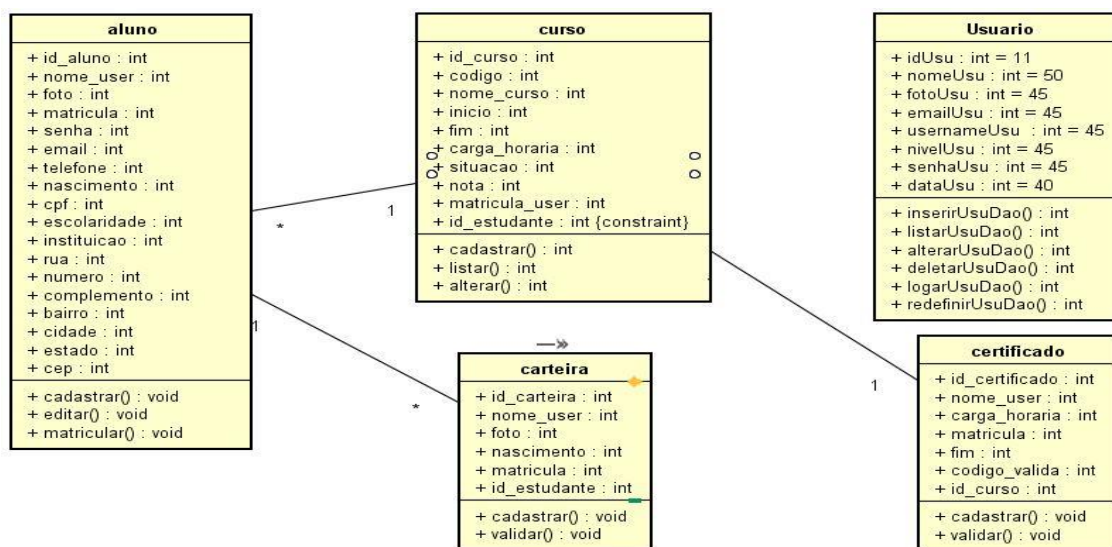


Fonte: Santos (2018), produção próprio autor.

Diagrama de Classe

Partindo da análise de requisitos foi criado o seguinte modelo de diagrama descrito na Figura 3.

Figura 3 - Diagrama de Classe



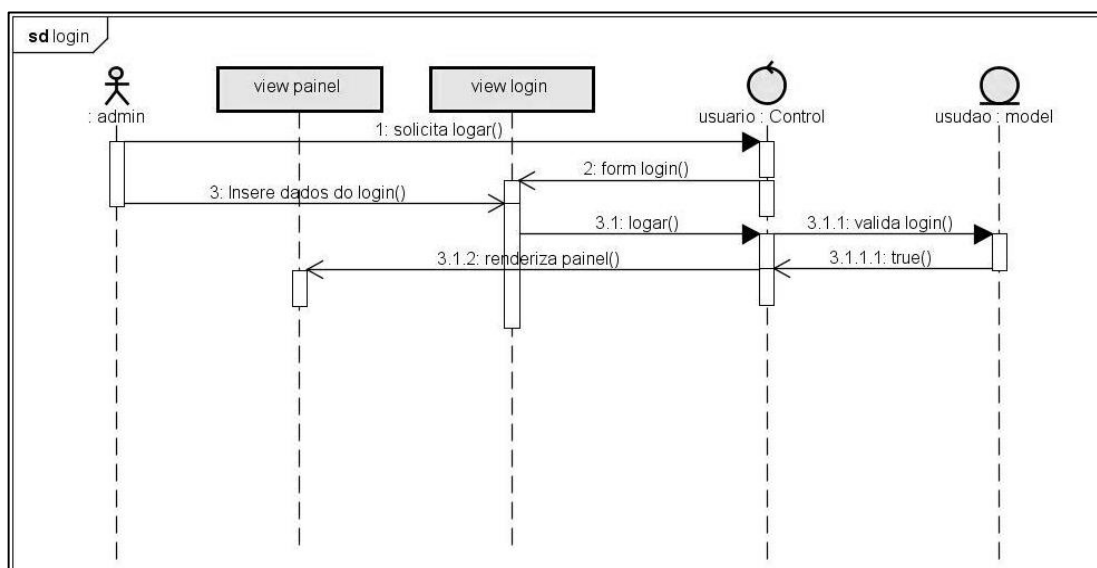
FONTE: Santos (2018), produção próprio autor



Diagrama de sequência

O diagrama de sequência representa a comunicação entre os objetos do sistema. Nas figuras de 4 a 8 observa-se os diagramas de sequências ilustrando como é feito o login, inserir, cadastrar, alterar e listar no *software*.

Figura 4 - Diagrama de sequência, representando o login



FONTE: Santos (2018), produção próprio autor.

Implementação

Toda a codificação utilizada no desenvolvimento da aplicação foi salva em arquivos para melhor organização, os arquivos foram agrupados por pastas, classificadas por funcionalidades. A estrutura da organização pode ser visualizada na figura 9.

Figura 5 - Estrutura da Organização





FONTE: Santos (2018), produção próprio autor

Telas do Sistema

Nas figuras de 10 a 15 a seguir apresentam as telas do *software* SGE

Figura 10 - Portal de Navegação nos Eventos



FONTE: Santos (2018), produção próprio autor

Figura 11 - Tela de login do SQO

ACESSO À ÁREA DO ALUNO

Desejo me Cadastrar

Se você ainda não é cadastrado, informe seus dados corretamente abaixo e poderá acessar a área do aluno e se matricular em nossos cursos.

Nome completo:

DDD + Telefone (apenas números):

Informe seu e-mail (letras minúsculas):

Criar senha de acesso ao site:

CADASTRAR

Já sou Cadastrado

Insera seu e-mail e senha abaixo e clique no botão "Entrar".

Digite seu email:

Digite sua Senha:

ENTRAR

[Esqueceu a senha?](#)

VALIDE SEU CERTIFICADO

Digite o Código do seu certificado

Validar

VALIDE SUA CARTEIRINHA

Digite o Código da sua carteirinha

Validar

 Desenvolvido por **Nilson Alves** | versão 1.0 | Copyright © SGI 2018 - 2018



FONTE: Santos (2018), produção próprio autor

Figura 12 - Tela painel principal

Fonte: Santos (2018), produção próprio autor.

Figura 13 – Tela descrição de cursos

A Escola de Cursos Online da **ADESC** oferece o curso de **Introdução à Teologia Cristã grátis** com certificado.

No curso gratuito de **Introdução à Teologia Cristã** o(a) aluno(a) terá acesso a um vasto material, disponível para download, em sua "Área do aluno" para estudar e se qualificar quando e onde quiser, disponível 24 horas por dia, 7 dias por semana em nossa plataforma de Ensino a Distância (EAD).

O curso online grátis de Introdução à Teologia Cristã possui **carga horária variável de 280 horas**. A carga horária final desse curso dependerá exclusivamente da necessidade do(a) aluno(a).

Para concluir esse curso é necessário realizar uma **avaliação** por meio de um questionário com 10 perguntas compostas de 4 respostas de múltipla escolha, no qual o(a) aluno(a) deverá obter uma média igual ou superior a 60,0 pontos para que esteja habilitado(a) a solicitar seu certificado.

Suporte ao aluno

Para maiores informações você pode solicitar ajuda de um de nossos atendentes pelo suporte ao aluno por **telefone:** (63) 3363-3786, **e-mail:** mscadesc@gmail.com ou **WhatsApp:** (63) 99291 1507.

FONTE: Santos (2018), produção próprio autor



Figura 14 - Tela painel do aluno

FONTE: Santos (2018), produção próprio autor.

Figura 15 - Tela conteúdo

FONTE: Santos (2018), produção próprio autor

Figura 15 - Tela prova

FONTE: Santos (2018), produção próprio autor



Figura 15 - Tela certificado

IGREJA ASSEMBLÉIA DE DEUS

MINISTÉRIO DA SALVAÇÃO EM CRISTO

CERTIFICADO

Certificamos que **Nilson Alves dos Santos**, concluiu com excelência o curso de **Introdução à Teologia Cristã**, em 06/10/2018, com duração de 280 horas, tendo um aproveitamento de 100 %.

Código de validação deste Certificado: **f00ad9f6df176cb31494ee7c3afe9eb1**
 Acesse o site www.sge.com.br

SECRETÁRIO

PASTOR

FONTE: Santos (2018), produção próprio autor

Avaliação do software

Para avaliar a qualidade do sistema, confeccionou-se um questionário o questionário de pesquisa foi submetido por meio de um formulário Google Docs que permitiu aos participantes responderem via internet.

ANÁLISE E RESULTADOS DOS DADOS

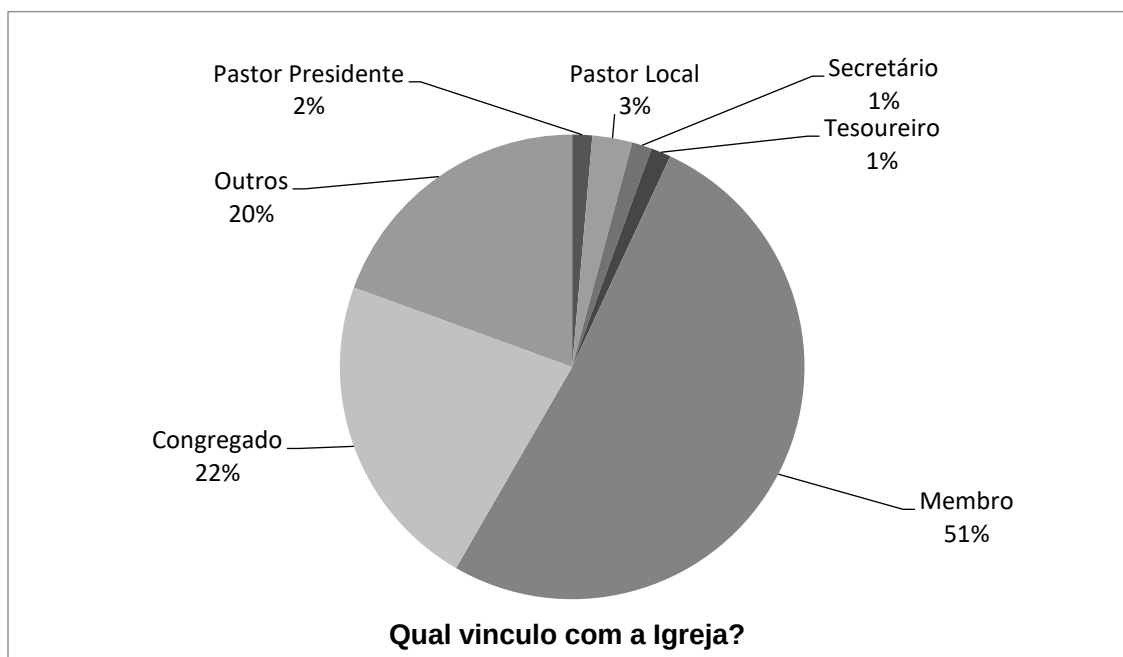
Neste item são apresentados os resultados e análises obtidas com a pesquisa que avalia a aceitação do software SQO.

Do universo da pesquisa de 72 participantes, responderam 1 pastor presidente, 2 pastores locais, 1 secretário, 1 tesoureiro 37 membros, 16 congregados e 14 classificados como outros. Desse total de 72, todos responderam ao questionário da pesquisa.

A partir da figura 16 a 21 apresenta o através dos gráficos tipo pizza os resultados da pesquisa aplicada.

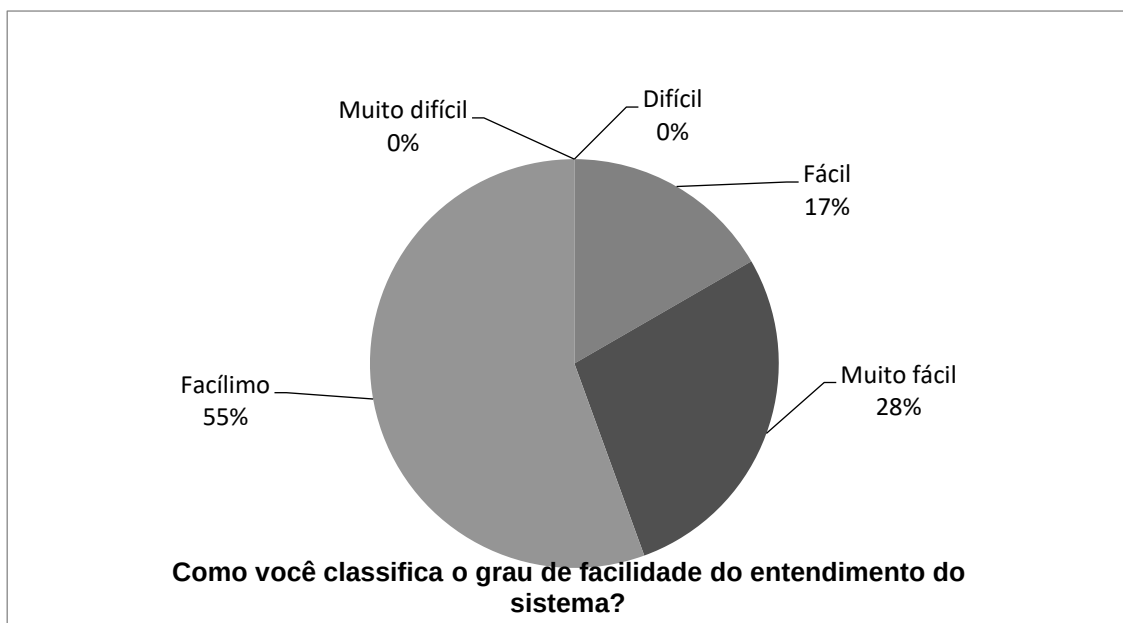


Figura 16 - Vinculo com a Igreja



FONTE: Santos (2018), produção próprio autor

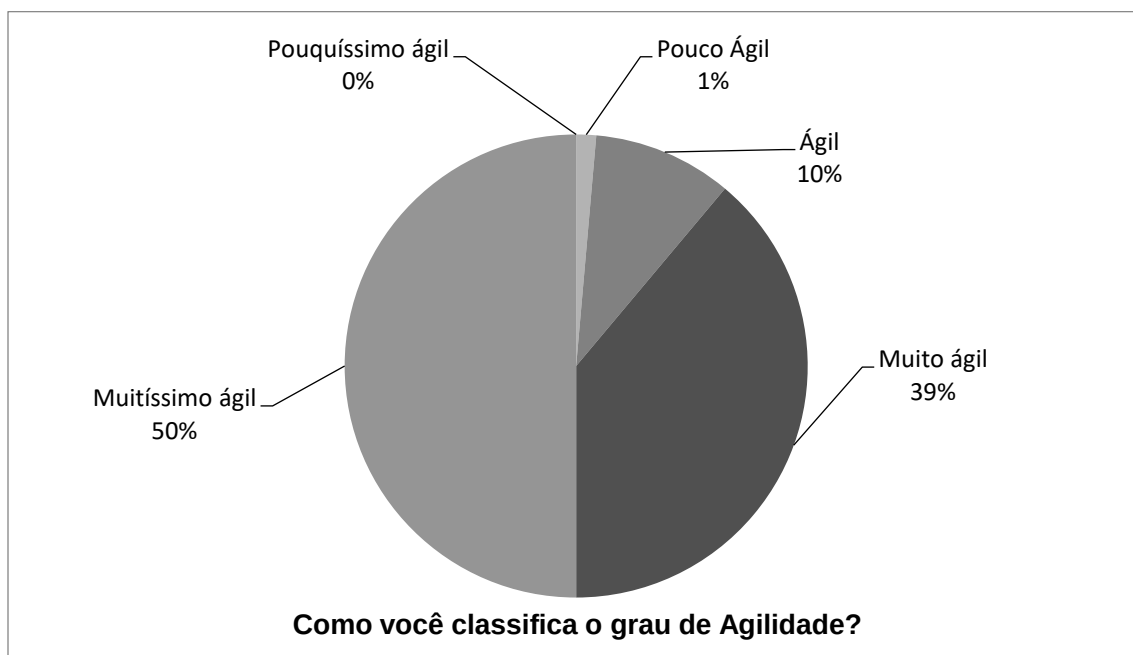
Figura 17 - Grau de facilidade do entendimento do sistema



FONTE: Santos (2018), produção próprio autor

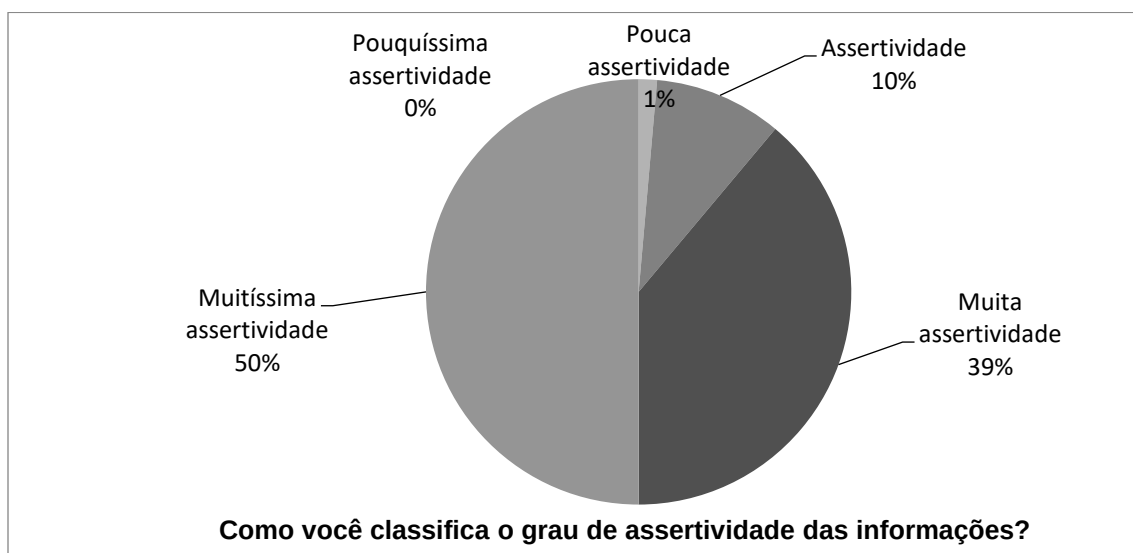


Figura 18 - Grau de Agilidade



FONTE: Santos (2018), produção próprio autor

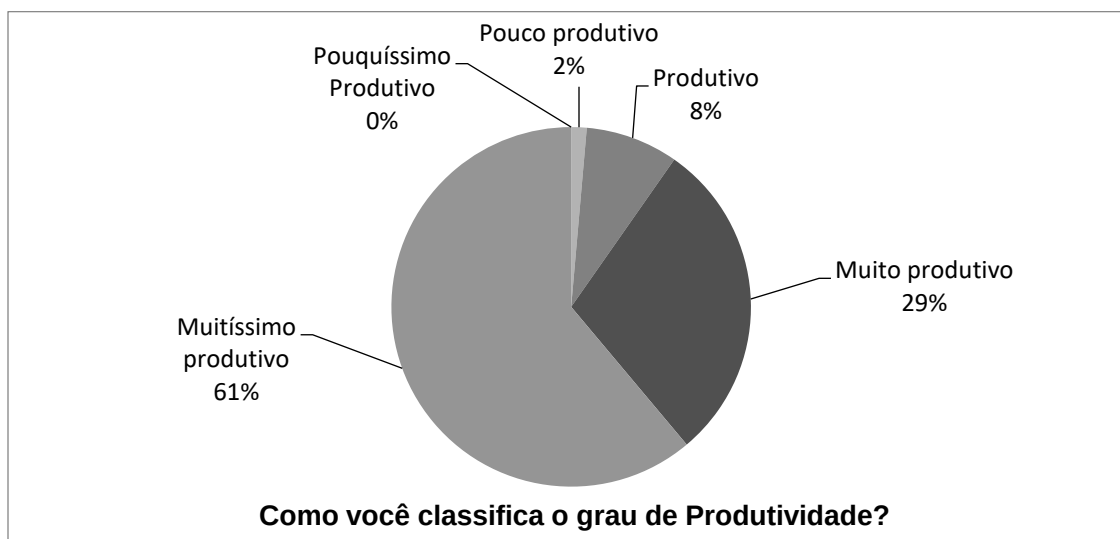
Figura 19 - Grau de assertividade das informações



FONTE: Santos (2018), produção próprio autor



Figura 20 - Grau de produtividade do Sistema



FONTE: Santos (2018), produção próprio autor

Figura 21 - Implantação do sistema



FONTE: Santos (2018), produção próprio autor



Figura 22 - Módulo EaD do sistema atende o ensino/aprendizagem



FONTE: Santos (2018), produção próprio autor

CONSIDERAÇÕES FINAIS

Diante dos resultados obtidos com a realização deste estudo, nota-se que o *software* foi avaliado positivamente pelos pesquisados, desta forma, este trabalho se propôs a Desenvolver e validar um portal *web* de ensino a distância, utilizá-lo como ferramenta qualificação de obreiros da Igreja Evangélica Assembleia de Deus, Ministério da Salvação em Cristo, *Campus* de Porto Nacional – TO.

Assim, o sistema atende as atuais necessidades, alcançando os objetivos e a hipótese levantada foi validada. Pois, de acordo com os dados da pesquisa, 99% dos pesquisados afirmaram que implantariam o sistema, esses resultados podem contribuir diretamente, de forma positiva, no processo de avaliação do sistema.

Como trabalhos Futuros propõe-se realizar a recuperação de senhas, sendo estas enviadas via e-mail do usuário cadastrado no sistema. Além do mais poderiam ser desenvolvidos novos módulos que surgissem com as necessidades e fazer incorporações a outros sistemas.

REFERÊNCIAS

- BOOCH, G., Rumbaugh, J., & Jacobson, I., 2000. **UML - Guia do Usuário**. Editora
- CARVALHO, A.; CHIOSSI, T. **Introdução à Engenharia de Software**. São Paulo: Unicamp, 2001.



EHLERS, U.D. **Linking strategic directions for European eLearning beyond 2010**. In: eLearningLisboa07 Conference Proceedings. Coordenação: Isabel Vilhena. Fundação para a Divulgação das Tecnologias de Informação, Tipografia Guerra, Viseu, Lisboa: 2008.

MACIEL, Ira Maria. Educação a distância. **Ambiente virtual: construindo significados**. Boletim Técnico do SENAC, v. 28, n. 3, p. 38-45, 2018.

MANUAL DO MPDF. **Documentação**. Disponível em: <<https://mpdf.github.io/>>. Acesso em 22 out. 2018.

MORER, Albert Sangrà. **E-learning value**. In: eLearningLisboa07 Conference Proceedings. Coordenação: Isabel Vilhena. Fundação para a Divulgação das Tecnologias de Informação, Tipografia Guerra, Viseu, Lisboa: 2008, p. 136.

MUGNOL, Marcio. **A educação a distância no brasil: Conceitos e fundamentos**. Revista Diálogo Educacional, [S.l.], v. 9, n. 27, p. 335-349, jul. 2009. ISSN 1981-416X. Disponível em:

<<https://periodicos.pucpr.br/index.php/dialogoeducacional/article/view/3589/3505>>. Acesso em: 09 out. 2018.

OLIVEIRA Gilson. **Informatização de Igrejas** 1998. Disponível em: <<http://www.geocities.ws/prgilson/GVirtual/gvtxt06.htm>>. Acesso em: 01 de maio 2018.

PAULSEN, M. **E-Learning: o papel dos sistemas de gestão da aprendizagem na Europa**.

Portal Infomoney. **Acelere o crescimento da sua empresa em até 35% com uma plataforma ERP**. Disponível em: <<http://www.infomoney.com.br/conteudo-patrocinado/noticia/5206438/accelere-crescimento-sua-empresa-ate-com-uma-plataforma-erp>>. Acesso em: 06 de maio 2018.

PRESSMAN, R. **Engenharia de Software: Uma Abordagem Profissional**. 7ª Ed., São Paulo: McGraw-Hill, 2011.

REIS, T. S; ALMEIDA, B. M. B. M. CORREIA, R. M. **Softwares de auxílio à docência no processo de Ensino-aprendizagem de Língua Portuguesa**. In: FUCHS, C; SCHWENGBER, I. L; SCHUTZ, J. A. Educação, Escola e Contemporaneidade: Questões e Diálogos. São Carlos. Pedro & João Editores. 2018. 421.

REZENDE, D. **Engenharia de Software e Sistemas de Informação**. 3ª Ed., Rio de Janeiro: Brasport, 2005.

SOMMERVILLE, Ian. **Engenharia de Software**. Tradução Ivan Bosnic e Kalinka G. de O. Gonçalves ; revisão técnica Kechi Hiramã. — 9. ed. — São Paulo: Pearson Prentice Hall, 2011.



SZYPERSKI, C. **Component Software: Beyond Object-oriented Programming**, 2a ed. Harlow, Reino Unido: AddisonWesley, 2002.

SZYPERSKI, Clemens. **Component Software: beyond object-oriented programming**. Harlow: Addison-Wesley, 1999.