

UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ
CÂMPUS CORNÉLIO PROCÓPIO
DIRETORIA DE GRADUAÇÃO E EDUCAÇÃO PROFISSIONAL
DEPARTAMENTO DE COMPUTAÇÃO
CURSO SUPERIOR DE TECNOLOGIA EM ANÁLISE E
DESENVOLVIMENTO DE SISTEMAS

RODRIGO FERNANDES PAULINO

**DESENVOLVIMENTO DE WEBSITE RESPONSIVO PARA A
PARÓQUIA CRISTO REI**

TRABALHO DE CONCLUSÃO DE CURSO

CORNÉLIO PROCÓPIO

2016

RODRIGO FERNANDES PAULINO

**DESENVOLVIMENTO DE WEBSITE RESPONSIVO PARA A
PARÓQUIA CRISTO REI**

Trabalho de Conclusão de Curso apresentada como requisito parcial para obtenção do grau de Tecnólogo em Análise e Desenvolvimento de Sistemas da Universidade Tecnológica Federal do Paraná - Câmpus Cornélio Procópio.

Orientador: Prof. Dr. Alexandre L'Erario

CORNÉLIO PROCÓPIO

2016



TERMO DE APROVAÇÃO

DESENVOLVIMENTO DE WEBSITE RESPONSIVO PARA A PARÓQUIA CRISTO REI

por

Rodrigo Fernandes Paulino

Este Trabalho de Conclusão de Curso foi julgado adequado para obtenção do Título de “Tecnólogo em Análise e Desenvolvimento de Sistemas” e aprovado em sua forma final pelo Programa de Graduação em Análise e Desenvolvimento de Sistemas da Universidade Tecnológica Federal do Paraná.

Cornélio Procópio, 28/06/2016

Prof. Titulação, Nome Professor Orientador
Universidade Tecnológica Federal do Paraná

Prof. Titulação, Nome professor membro da
banca
Universidade Tecnológica Federal do Paraná

Prof. Titulação, Nome professor membro da
banca
Universidade Tecnológica Federal do Paraná

RESUMO

PAULINO, Rodrigo. DESENVOLVIMENTO DE WEBSITE RESPONSIVO PARA A PARÓQUIA CRISTO REI. 68 f. Trabalho de Conclusão de Curso – Curso superior de tecnologia em análise e desenvolvimento de sistemas, Universidade Tecnológica Federal do Paraná. Cornélio Procópio, 2016.

Esta monografia tem como objetivo a narrativa da criação de um *website* responsivo para a Paróquia Cristo Rei de Cornélio Procópio, que por sua vez visa aprimorar o relacionamento entre Paróquia e fiéis. A justificativa para este projeto é a mudança no paradigma de comunicação, devido à facilidade de acesso a internet atualmente. O *website* por sua vez possui diversos recursos e estratégias para aproximar a comunidade cristã da igreja, facilitando a comunicação entre ambas as partes. É possível ao usuário solicitar documentos, agendar visitas com um dos padres da paróquia, pré-agendar um casamento e visualizar as liturgias diariamente, tudo de forma rápida e prática. A monografia descreve com mais detalhes as tecnologias e o método adotado para auxiliar o desenvolvimento do projeto, mostrando também os artefatos gerados ao longo do desenvolvimento.

Palavras-chave: Liturgia diária. Gerenciamento de Usuário. Agendar Visita. Agendar Casamento. Solicitar Documentos.

ABSTRACT

PAULINO, Rodrigo. DEVELOPMENT OF RESPONSIVE WEBSITE FOR THE PARISH CRISTO REI. 68 f. Trabalho de Conclusão de Curso – Curso superior de tecnologia em análise e desenvolvimento de sistemas, Universidade Tecnológica Federal do Paraná. Cornélio Procópio, 2016.

This monograph aims to describe the creation of a responsive website for the Parish Cristo Rei of Cornélio Procópio, which aims to improve the relationship between parish and christian community. The justification for this project is the paradigm change in the communication, due the easy way to access the internet currently. The website has many features and strategies to improve the relationship between the christian community and the church, facilitating communication between both parties. It is possible to the user to make a request of a document, schedule a visit with a parish priest, schedule a marriage and view daily liturgies. The monograph describes in more detail the technologies and methods adopted to assist the development of the project, showing all the generated artifacts over development.

Keywords: Daily Liturgy. User Management. Schedule Visit. Schedule Marriage. Documents Request.

LISTA DE FIGURAS

| | | |
|-----------|--|----|
| FIGURA 1 | – Hierarquia das Igrejas Católicas | 10 |
| FIGURA 2 | – Descrição visual do padrão MVC | 14 |
| FIGURA 3 | – Descrição visual para o padrão de arquitetura utilizado no projeto | 15 |
| FIGURA 4 | – Ciclo de vida do Processo WAP | 19 |
| FIGURA 5 | – Parte 1 - Modelo de Requisitos | 23 |
| FIGURA 6 | – Parte 2 - Modelo de Requisitos | 24 |
| FIGURA 7 | – Parte 3 - Modelo de Requisitos | 25 |
| FIGURA 8 | – Parte 4 - Modelo de Requisitos | 26 |
| FIGURA 9 | – Diagrama de caso de uso | 27 |
| FIGURA 10 | – Modelo Entidade-Relacionamento | 29 |
| FIGURA 11 | – Página de login da parte administrativa | 30 |
| FIGURA 12 | – Página de Calendário de Liturgias | 30 |
| FIGURA 13 | – Página de Liturgia do dia | 31 |
| FIGURA 14 | – Codificação da regra de negócio - Classe LiturgiaBO - Método Salvar .. | 32 |
| FIGURA 15 | – Codificação da regra de negócio - Classe LiturgiaBO - Método Inserir .. | 32 |
| FIGURA 16 | – Codificação da regra de negócio - Classe LiturgiaBO - Método Alterar .. | 33 |
| FIGURA 17 | – Codificação da regra de negócio - Classe LiturgiaBO - Método Excluir .. | 33 |
| FIGURA 18 | – Codificação da regra de negócio - Classe LiturgiaBO - Método ConsultarPorId | 34 |
| FIGURA 19 | – Codificação da regra de negócio - Classe LiturgiaBO - Método ConsultarPorData | 34 |
| FIGURA 20 | – Codificação da regra de negócio - Classe LiturgiaBO - Método ConsultarPorTodos | 34 |
| FIGURA 21 | – Parte 1 - Descrição do Caso de Uso Autenticar Usuário | 38 |
| FIGURA 22 | – Parte 2 - Descrição do Caso de Uso Autenticar Usuário | 39 |
| FIGURA 23 | – Parte 3 - Descrição do Caso de Uso Autenticar Usuário | 40 |
| FIGURA 24 | – Parte 4 - Descrição do Caso de Uso Autenticar Usuário | 41 |
| FIGURA 25 | – Parte 1 - Descrição do Caso de Uso Gerenciar Usuário | 42 |
| FIGURA 26 | – Parte 2 - Descrição do Caso de Uso Gerenciar Usuário | 43 |
| FIGURA 27 | – Parte 3 - Descrição do Caso de Uso Gerenciar Usuário | 44 |
| FIGURA 28 | – Parte 4 - Descrição do Caso de Uso Gerenciar Usuário | 45 |
| FIGURA 29 | – Parte 5 - Descrição do Caso de Uso Gerenciar Usuário | 46 |
| FIGURA 30 | – Parte 6 - Descrição do Caso de Uso Gerenciar Usuário | 47 |
| FIGURA 31 | – Parte 7 - Descrição do Caso de Uso Gerenciar Usuário | 48 |
| FIGURA 32 | – Parte 1 - Descrição do Caso de Uso Solicitar um Documento | 49 |
| FIGURA 33 | – Parte 2 - Descrição do Caso de Uso Solicitar um Documento | 49 |
| FIGURA 34 | – Parte 3 - Descrição do Caso de Uso Solicitar um Documento | 50 |
| FIGURA 35 | – Parte 1 - Descrição do Caso de Uso Agendar Atendimento com o Padre | 51 |
| FIGURA 36 | – Parte 2 - Descrição do Caso de Uso Agendar Atendimento com o Padre | 51 |
| FIGURA 37 | – Parte 3 - Descrição do Caso de Uso Agendar Atendimento com o Padre | 52 |
| FIGURA 38 | – Parte 1 - Descrição do Caso de Uso Pré-agendar Casamento | 53 |
| FIGURA 39 | – Parte 2 - Descrição do Caso de Uso Pré-agendar Casamento | 53 |

| | | |
|-----------|--|----|
| FIGURA 40 | – Parte 3 - Descrição do Caso de Uso Pré-agendar Casamento | 54 |
| FIGURA 41 | – Parte 1 - Descrição do Caso de Uso Gerenciar Liturgia | 55 |
| FIGURA 42 | – Parte 2 - Descrição do Caso de Uso Gerenciar Liturgia | 56 |
| FIGURA 43 | – Parte 3 - Descrição do Caso de Uso Gerenciar Liturgia | 57 |
| FIGURA 44 | – Parte 4 - Descrição do Caso de Uso Gerenciar Liturgia | 58 |
| FIGURA 45 | – Parte 5 - Descrição do Caso de Uso Gerenciar Liturgia | 59 |
| FIGURA 46 | – Parte 6 - Descrição do Caso de Uso Gerenciar Liturgia | 60 |
| FIGURA 47 | – Parte 1 - Arquitetura Global da Aplicação | 61 |
| FIGURA 48 | – Parte 2 - Arquitetura Global da Aplicação | 62 |
| FIGURA 49 | – Parte 3 - Arquitetura Global da Aplicação | 63 |
| FIGURA 50 | – Parte 4 - Arquitetura Global da Aplicação | 63 |
| FIGURA 51 | – Parte 5 - Arquitetura Global da Aplicação | 64 |
| FIGURA 52 | – Parte 6 - Arquitetura Global da Aplicação | 64 |
| FIGURA 53 | – Parte 7 - Arquitetura Global da Aplicação | 65 |
| FIGURA 54 | – Parte 8 - Arquitetura Global da Aplicação | 66 |
| FIGURA 55 | – Parte 9 - Arquitetura Global da Aplicação | 67 |
| FIGURA 56 | – Modelo Navegacional do contendo as páginas | 68 |

LISTA DE SIGLAS

| | |
|------|---|
| HD | High Definition |
| CSS | Cascading Style Sheets |
| HTML | HyperText Markup Language |
| MVC | Model-View-Controller |
| WUI | Web User Interface |
| BO | Bussiness Object |
| SGBD | Sistema Gerenciador de Banco de Dados |
| DAL | Data Access Layer |
| UML | Unified Modeling Language |
| IDE | Integrated Development Environment |
| WAP | Web Agile Process |
| MSF | Microsoft Solutions Framework |
| RUP | Rational Unified Process |
| WEBE | Web Engineering |
| CMMI | Capability Maturity Model - Integration |

SUMÁRIO

| | | |
|----------|------------------------------|-----------|
| 1 | INTRODUÇÃO | 9 |
| 1.1 | JUSTIFICATIVA | 11 |
| 1.2 | OBJETIVOS | 11 |
| 1.3 | ORGANIZAÇÃO DO TEXTO | 12 |
| 2 | FUNDAMENTAÇÃO TEÓRICA | 13 |
| 2.1 | FRAMEWORK BOOTSTRAP | 13 |
| 2.2 | ARQUITETURA | 14 |
| 3 | MATERIAIS E MÉTODOS | 17 |
| 3.1 | FERRAMENTAS | 17 |
| 3.1.1 | Visual Studio | 17 |
| 3.1.2 | SQL Server | 18 |
| 3.2 | MÉTODOS | 18 |
| 3.2.1 | Processo de Desenvolvimento | 18 |
| 3.2.2 | WAP | 19 |
| 4 | EXECUÇÃO | 22 |
| 4.1 | INICIAÇÃO | 22 |
| 4.2 | ELABORAÇÃO | 27 |
| 4.3 | CONSTRUÇÃO | 30 |
| 5 | CONSIDERAÇÕES FINAIS | 35 |
| 5.1 | IMPLEMENTAÇÃO FUTURA | 35 |
| 5.2 | DIFICULDADES E OBSERVAÇÕES | 35 |
| | REFERÊNCIAS | 37 |
| 6 | ANEXOS | 38 |

1 INTRODUÇÃO

O mundo virtual se tornou real para grande parte da população do Brasil; um pouco mais que 85 milhões de pessoas utilizaram a rede em 2013, segundo pesquisa do IBGE. O acesso à internet ainda não está disponível para todos, mas em cinco anos, o aumento foi maior que 50%, o que torna essa mídia cada vez mais relevante.

Outra pesquisa, aponta que 15% dos brasileiros tem em mãos um *smartphone*, dentre esses, 73% utilizam alguma rede social em seus aparelhos (PEWGLOBAL, 2014). A pesquisa Mobile Report, realizada pela Nielsen Ibope, apontou que no primeiro semestre de 2015 o número de pessoas utilizando *smartphone* para acessar a internet chegou a mais de 68 milhões no Brasil (NIELSEN, 2015).

Diante dessa nova realidade, cresce a cada dia a demanda pelo desenvolvimento de *softwares* e aplicativos para *web* com vistas para as telas de celulares, ou seja, uma mudança de paradigma na forma de planejar e construir soluções computacionais com mais acessibilidade para esses dispositivos.

Essa comunicação virtual atinge todos os níveis da sociedade, até mesmo o religioso. As igrejas valem-se das mídias eletrônicas para se relacionar com seu público. Entre as várias denominações cristãs, a Igreja Católica Apostólica Romana é a que possui a hierarquia mais complexa. Ela é composta por várias dioceses, que por sua vez são formadas por diversas paróquias, comunidade de fiéis congregada em torno de um sacerdote. A figura 1 ilustra como é a hierarquia das igrejas.



Figura 1: Hierarquia das Igrejas Católicas

De acordo com o artigo escrito por Juberto Santos sobre a Hierarquia da Igreja Católica, a Arquidiocese é a província eclesiástica que abrange todas as dioceses de uma região. Quem a governa e a preside é o bispo mais importante: o arcebispo.

A Diocese é a circunscrição eclesiástica dirigida pelo bispo. Ela é também chamada de Bispado. O Código do Direito Canônico, no nº 369, afirma que a diocese é a 'porção do povo de Deus confiada a um bispo'.

Dentro de cada diocese existem um ou mais Vicariatos. Os Vicariatos episcopais são um instrumento evangelizador mais descentralizado. Colaboram para o atendimento às exigências da ação evangelizadora em cada grande área geográfica ou ambiental, organizando melhor o trabalho e as relações pastorais.

A Forania é um grupo determinado de paróquias dentro de um Vicariato. Cada forania é confiada a um vigário forâneo (título dado pelo bispo a um grupo de padres dentro de um Vicariato). Essa união de diversas paróquias mais próximas territorialmente favorece o trabalho pastoral mediante uma ação em comum.

A Paróquia é uma comunidade dentro da Diocese entregue aos cuidados pastorais e administrativos de um presbítero que recebe o título de pároco. Antigamente eram chamadas de 'Freguesias'. Ele deve trabalhar em comunhão com a diocese, as lideranças pastorais e os

demais fiéis batizados.

Por último, a Capela, antigamente chamadas de "ermidas", é uma pequena comunidade numa região administrada por uma Paróquia. Além do Culto a Deus, podem-se realizar casamentos e os demais sacramentos, além das atividades sociais e pastorais (SANTOS, 2002).

A boa comunicação entre os padres e seus fiéis é o que vai garantir com que a Igreja atinja seus objetivos de evangelização. O *website* é um instrumento facilitador do processo de comunicação. Ele otimiza o acesso às diversas informações relacionadas ao cotidiano da paróquia.

Esta monografia descreverá as motivações, objetivos, métodos, processos, arquiteturas usadas para a criação de um sistema *web* para a Paróquia Cristo Rei de Cornélio Procópio, Paraná.

1.1 JUSTIFICATIVA

Uma vez que comunicação virtual é imperativa no mundo contemporâneo, essa monografia visa aprimorar esse meio de comunicação na Paróquia Cristo Rei de Cornélio Procópio através de um *website*, favorecendo a interação dos fiéis com o padre e as estruturas paroquiais.

Além de disponibilizar informações aos interessados sobre as várias atividades da paróquia, este trabalho tem como diferencial otimizar a aproximação dos padres com os fiéis. Tais como agendamento para visitas com o padre da paróquia, recursos para solicitar documentos referentes a vida cristã da comunidade e de pré-agendar um casamento ou curso de noivos.

O sistema proposto será operacionalizado para múltiplas plataformas via *web*, desenvolvido de forma responsiva, aprimorando a experiência do usuário ao acessar em qualquer dispositivo que seja, desde *smartphones* até telas em *full High Definition* (HD).

A responsividade do sistema e a agilidade digital auxiliam na solicitação e emissão de documentos requisitados pelos fiéis, diminuindo a burocracia e a necessidade de deslocamento até a paróquia, favorecendo não só os que moram na região, como também pessoas de outras partes do mundo.

1.2 OBJETIVOS

Este trabalho tem como objetivo narrar o desenvolvimento de uma plataforma *web* para a Paróquia Cristo Rei, visando otimizar o acesso dos fiéis aos serviços da paróquia e auxiliar a

troca de informação entre ambos. Também permite o acesso dos usuários ao *website* por meio de diversos dispositivos distintos e sistemas operacionais de forma responsiva.

1.3 ORGANIZAÇÃO DO TEXTO

A monografia está estruturada da seguinte forma: o capítulo 2 tem como base as tecnologias que foram utilizadas para a implementação; o capítulo 3 apresenta os métodos e os materiais que fizeram parte do projeto; o capítulo 4 mostra a execução do projeto, juntamente com os principais artefatos; no capítulo 5 são apresentadas as considerações finais; por fim, no capítulo 6 são apresentados os anexos a esta monografia.

2 FUNDAMENTAÇÃO TEÓRICA

Esse capítulo apresenta a descrição das tecnologias utilizadas de forma sucinta e objetiva, sendo elas um *framework* que irá cuidar da parte responsiva do *website*, além de uma breve explicação sobre a arquitetura do sistema.

2.1 FRAMEWORK BOOTSTRAP

Como apresentado no capítulo 1, o *website* foi desenvolvido de forma responsiva, será adaptável aos diversos tamanhos de telas e resoluções. Devido ao crescimento de pessoas navegando na internet por meio de *smartphones*, é fundamental que as empresas e desenvolvedores de sistemas computacionais adotem esse novo paradigma na criação de *websites*. Para otimizar o processo e garantir a qualidade no desenvolvimento de sistemas computacionais, independentemente da plataforma, foram criados os *framework*.

Segundo, (GOVONI, 1999) os *framework* são uma coleção abstrata de classes, interfaces e padrões dedicados para resolver uma série de problemas comuns através de uma arquitetura flexível e extensível. Pensando em desenvolvimento de designs responsivos foi criado o *Bootstrap*, um *framework* voltado para a agilidade na criação desse tipo de design, com classes pré-formatadas para adaptar-se a diferentes resoluções. O *Bootstrap* foi criado utilizando classes em *Cascading Style Sheets* (CSS) de modo que para ser utilizado basta fazer o uso das mesmas classes no código *HyperText Markup Language* (HTML).

O *Bootstrap* é formado por diversas classes em CSS, que em seu código reconhecem o dispositivo ao qual está sendo acessado, e a partir do tamanho da resolução da tela, configura as classes para se comportar de maneira que a experiência do usuário não seja prejudicada. Usando as classes desse *framework* dentro das tags HTML da página, o site já estará agindo de forma responsiva (TWITTER,).

2.2 ARQUITETURA

Para fins de organização o *software* conta uma arquitetura que utiliza de boas práticas do padrão *Model-View-Controller* (MVC), que nada mais é do que um padrão de arquitetura de software que separa a codificação em três camadas diferentes, sendo elas a camada de modelo, onde são armazenados os dados da aplicação, a camada é representada pelo banco de dados. A camada de controle, onde é feita a manipulação dos dados da camada de modelo, para serem tratados e traduzidos de forma legível para a camada de visão. A camada de visão irá exibir os dados já tratados na camada de controle para o público. A figura 2 ilustra como a arquitetura MVC se comunica entre si.

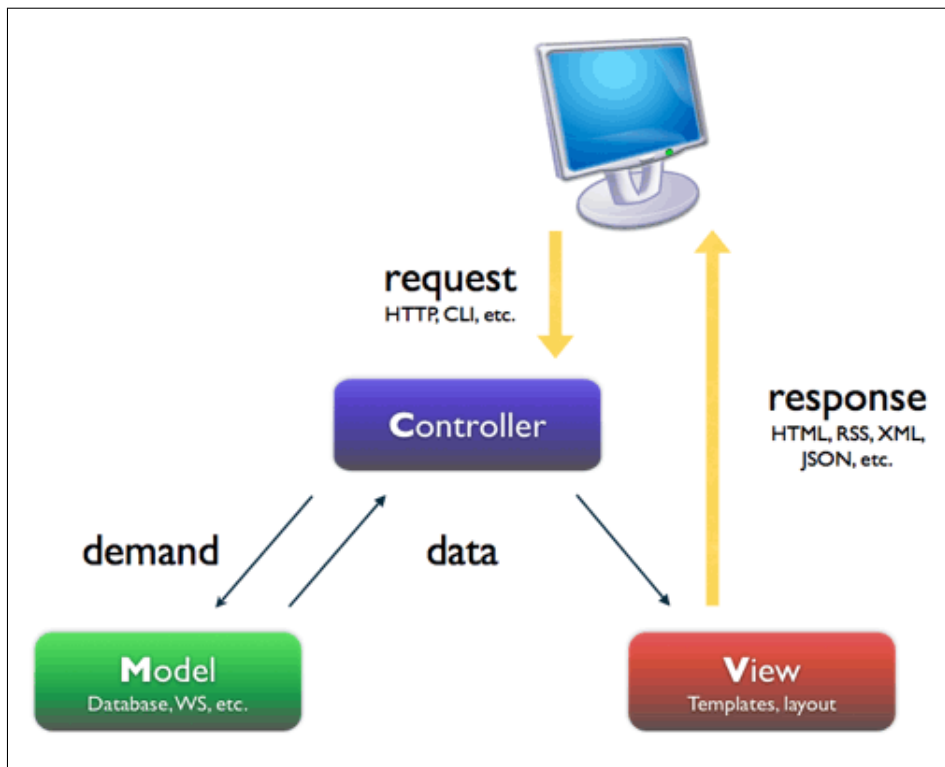


Figura 2: Descrição visual do padrão MVC

Diferente do padrão MVC, a arquitetura utilizada neste projeto separa a camada de controle em duas, uma responsável somente pela comunicação com a camada de banco de dados e outra responsável pelo tratamento dos dados como é mostrado na figura 3.

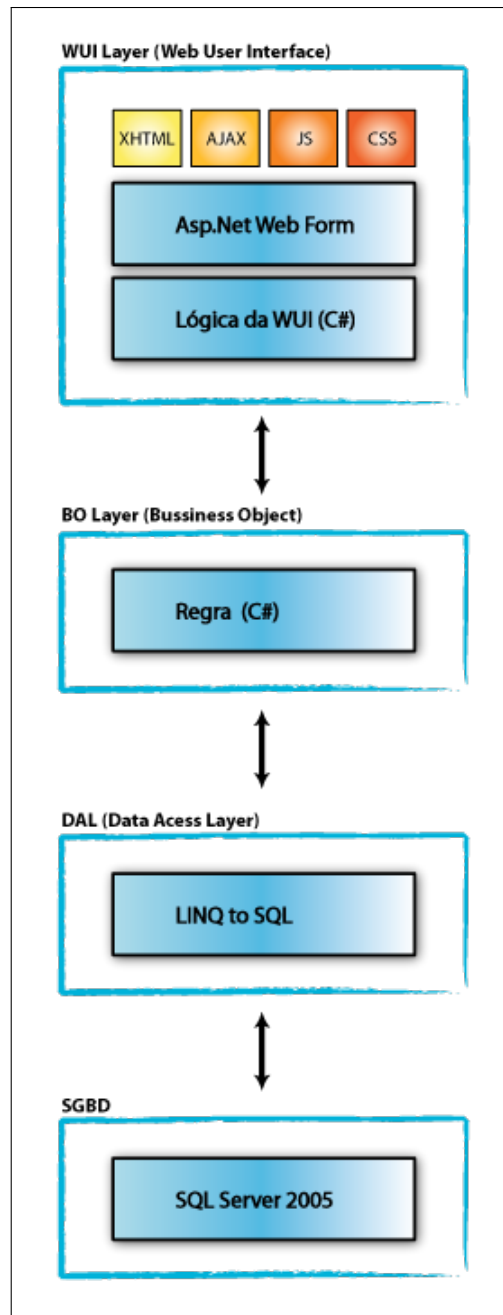


Figura 3: Descrição visual para o padrão de arquitetura utilizado no projeto

- Web User Interface (WUI):* Responsável pela apresentação gráfica ao usuário; Utiliza objetos providos da *Business Object (BO)*;
- BO Layer:* Responsável pela implementação da Regra de Negócios, por meio de classes de negócio; Utiliza os dados do Sistema Gerenciador de Banco de Dados (SGBD) por meio de da tecnologia LINQ to SQL;
- Data Access Layer (DAL):* Camada que realiza o mapeamento dos dados em um modelo de objetos por meio da tecnologia LINQ to SQL e VBML;

- e) SGBD: Responsável pelo gerenciamento das informações persistidas da aplicação; Oferece para a DAL as informações da aplicação.

3 MATERIAIS E MÉTODOS

Este capítulo apresenta os materiais e ferramentas utilizadas para a realização deste trabalho assim como o método de desenvolvimento adotado para melhor organização das atividades.

3.1 FERRAMENTAS

3.1.1 VISUAL STUDIO

O Microsoft Visual Studio é um pacote de programas para desenvolvimento de software que se utiliza do *.NET framework* e de algumas linguagens variadas. Possibilitando assim uma gama de possíveis projetos em áreas distintas, com diversos recursos como aplicações web, aplicações *desktop*, integração com banco de dados, alguns diagramas da *Unified Modeling Language* (UML) (MICROSOFT,).

Para este trabalho, as linguagens ASP.NET e C# serão utilizadas para a elaboração de classes, e as linguagens de marcação HTML e CSS para a criação de páginas web e estilo para as mesmas.

ASP.NET é a plataforma da Microsoft para o desenvolvimento de aplicações Web e esta na base do *framework* .NET herdando todas as suas características, por isso, como qualquer aplicação .NET, as aplicações para essa plataforma podem ser escritas em várias linguagens, como C# (MICROSOFT,).

A escolha dessas linguagens deu-se pela utilização das ferramentas no ambiente de trabalho e pela vasta quantidade de recursos que a *Integrated Development Environment* (IDE) da Microsoft tem a oferecer de forma gratuita.

3.1.2 SQL SERVER

O SQL Server é um sistema gerenciador de banco de dados desenvolvido pela Microsoft. O SGBD vai ser responsável pela camada de modelo, armazenando e recuperando os dados pertinentes ao sistema web.

A escolha desse SGBD deu-se pela compatibilidade com os outros produtos da Microsoft escolhidos para o desenvolvimento do projeto.

3.2 MÉTODOS

”Metodologia é um conjunto de abordagens, técnicas e processos utilizados pela ciência para formular e resolver problemas de aquisição objetiva do conhecimento, de uma maneira sistemática”(RODRIGUES, 2007).

3.2.1 PROCESSO DE DESENVOLVIMENTO

Um processo de desenvolvimento de *software* é um conjunto de atividades, parcialmente ordenadas, com a finalidade de obter um produto de *software*. É estudado dentro da área de Engenharia de Software, sendo considerado um dos principais mecanismos para se obter *software* de qualidade e cumprir corretamente os contratos de desenvolvimento, sendo uma das respostas técnicas adequadas para resolver a crise do *software* (REISSWITZ, 2009).

A ausência de um processo disciplinado e também flexível para aplicações Web pode levar a sérios problemas, uma vez que esse tipo de aplicação é impulsionado pelo conteúdo e está em contínua evolução. Assim, percebe-se a sua característica de desenvolvimento ágil, e fazer rapidamente não significa realizar um trabalho deficiente, em termos de engenharia, e sim disciplinado e flexível o suficiente para atender os requisitos funcionais e de qualidade necessários (NETO, 2008).

O processo utilizado nesse trabalho foi o *Web Agile Process* (WAP), estruturado por (NETO, 2008). A metodologia segue os princípios e boas práticas dos processos *Microsoft Solutions Framework* (MSF), *Rational Unified Process* (RUP), *Web Engineering* (WEBE), amplamente utilizados na indústria de software e, além disso, o WAP adota os padrões do *Capability Maturity Model - Integration* (CMMI) nível de maturidade 2.

3.2.2 WAP

O WAP faz uso da mesma estrutura de fases do RUP (iniciação, elaboração, construção e transição), apresentando diferença nas disciplinas e suas atividades, além dos artefatos a serem produzidos. Na figura 4, é possível compreender a organização das fases e disciplinas do WAP.

| DISCIPLINAS | FASES | | | |
|---------------------------|-----------|------------|------------|-----------|
| | Iniciação | Elaboração | Construção | Transição |
| Requisitos | | | | |
| Modelagem | | | | |
| Implementação | | | | |
| Teste | | | | |
| Implantação | | | | |
| Gerenciamento de Projeto | | | | |
| Gerenciamento Operacional | | | | |

Figura 4: Ciclo de vida do Processo WAP

A escolha dessa processo se deu pela sua riqueza na produção de artefatos e pela sua flexibilidade em relação as atividades em cada iteração, tornando-o objetivo e ágil no desenvolvimento de uma aplicação para *web*.

A cada iteração - espaço de tempo designado para a construção de algum requisito da aplicação - do ciclo de vida do projeto, as atividades de cada disciplina são executadas. As disciplinas a serem utilizadas nesse trabalho são:

- a) Iniciação: Delimitar o escopo da aplicação, seus riscos e requisitos, e estabelecer um ambiente propício para o projeto. Em outras palavras, o marco é o objetivo do ciclo de vida.
- b) Elaboração: Essa fase tem como foco o planejamento para prosseguir com a construção do projeto e a definição de uma linha base. O marco é a arquitetura do ciclo de vida.
- c) Construção: Como o próprio nome já diz, essa fase será marcada pelo desenvolvimento da aplicação em si de forma incremental e com iterações curtas, sempre fornecendo versões beta para alguns testes. O marco é a capacidade operacional da aplicação.

- d) Transição: Trabalhar nos detalhes finais do projeto, implantar no ambiente estabelecido previamente, fornecer auxílio aos usuários acerca da aplicação e avaliar os aspectos gerais. O marco é a liberação da aplicação.

Cada uma dessas fases irá passar por um ciclo de vida com as disciplinas que foram definidas para o escopo do projeto, dentre as disciplinas temos as seguintes descrições:

- a) Requisitos: Levantar as necessidades do projeto, definir o escopo e detalhar ou refinar a aplicação, mantendo todo o time de desenvolvimento em sincronia.
- b) Modelagem: Definir uma arquitetura, mapear os requisitos em diagramas para que os artefatos auxiliem na disciplina de implementação.
- c) Implementação: Transformar os modelos de projeto em códigos funcionais para serem testados e implantados posteriormente.
- d) Teste: Procurar por possíveis erros na aplicação a fim de refiná-la visando uma maior qualidade.
- e) Implantação: Preparar a aplicação e disponibilizá-la para os interessados no projeto, assim como fornecer suporte ao cliente.
- f) Gerenciamento de Projeto: Planejar, monitorar e dar suporte às disciplinas e fases, almejando o andamento do projeto de forma correta, gerenciando recursos e cronogramas.
- g) Gerenciamento Operacional: Fornecer suporte as atividades operacionais como planejamentos, configurações de ambiente e monitoramentos.

Devido a relação complexidade x tempo, o processo seguiu em estilo cascata. Os principais artefatos a serem gerados durante o decorrer do ciclo de vida deste projeto serão:

- Diagrama de Caso de Uso;
- Código de marcação e estilo;
- Código-fonte;
- Instruções de banco de dados;
- Plano de iteração;

- Processo adaptado;

O quadro 1 mostra as atividades selecionadas em cada fase do processo WAP para o desenvolvimento do projeto, mantendo assim uma linha lógica de tarefas a serem executadas.

Tabela 1: Processo Adaptado

| - | Iniciação | Elaboração | Construção | Implantação |
|---------------------------|--|---|--|--|
| Requisitos | -Definir a aplicação e o escopo; -Revisar requisitos; | -Detalhar e refinar a aplicação; -Revisar requisitos; | -Revisar requisitos; | |
| Modelagem | -Definir arquitetura candidata; | -Projetar Arquitetura da Usabilidade; -Projetar Arquitetura da Aplicação; -Projetar Banco de Dados; -Revisar modelagem; | -Projetar Casos de Uso; -Projetar Banco de Dados; -Revisar modelagem; | -Projetar Casos de Uso (Modificações); -Projetar Banco de Dados; -Revisar modelagem; |
| Implementação | | -Construir elementos multimídia e linguagens de marcação; | -Construir elementos multimídia e linguagens de marcação; - Implementar componentes; -Implementar tarefas de banco de dados; | -Construir elementos multimídia e linguagens de marcação; - Implementar componentes; -Implementar tarefas de banco de dados; - Disponibilizar a aplicação; |
| Gerenciamento de Projeto | -Planejar o projeto; -Planejar uma iteração; -Monitorar e guiar o projeto; | -Planejar uma iteração; - Monitorar e guiar o projeto; | -Planejar uma iteração; - Monitorar e guiar o projeto; | -Monitorar e guiar o projeto; |
| Gerenciamento Operacional | -Estabelecer processo para o projeto; - Configurar o ambiente; -Utilizar o ambiente; | -Utilizar o ambiente; | -Utilizar o ambiente; | -Utilizar o ambiente; |

4 EXECUÇÃO

Neste capítulo serão apresentados os principais artefatos gerados no decorrer do desenvolvimento do projeto. Os artefatos apresentados estão de forma sequencial e estão em suas respectivas versões finais de acordo com cada fase apresentada na 1, embora o processo tenha sido feito de modo iterativo e incremental.

O *website* em questão ainda não foi implantado e nem hospedado em nenhum servidor, portanto ainda não houve retorno para a Paróquia Cristo Rei e nenhum artefato da fase de Implantação foi gerado.

Os demais artefatos gerados no processo se encontram no capítulo 6 referente a anexos.

4.1 INICIAÇÃO

As figuras de 5 e 8 são parte do artefato gerado na atividade de definir aplicação e escopo cujo tempo de duração foi de 4 horas

DESCRIÇÃO FUNCIONAL

Benefícios Gerais

| ID | Benefício |
|----|---|
| B1 | Disponibilização de informações de forma prática e acessível a diversos dispositivos. |
| B2 | Aplicação de técnicas de usabilidade para melhor interação do cliente com o website. |

Atores

| ID | Nome | Descrição |
|----|------------|---|
| A1 | Secretária | Responsável por alimentar a aplicação web com o conteúdo a respeito da Paróquia, tais como: eventos, notícias e imagens. |
| A2 | Padre | Responsável por administrar os usuários da aplicação e possui também as demais funcionalidades disponíveis para os outros atores. |
| A3 | Fiel | Utiliza da aplicação e interage com suas funções. |
| A4 | Aplicação | Representa funcionalidades a serem executadas automaticamente pela aplicação, sem a intervenção de um usuário. |

Figura 6: Parte 2 - Modelo de Requisitos

Requisitos Funcionais (Casos de Uso)

Tipos de Prioridades:

- **Alta:** deverão ser desenvolvidas de forma completa na aplicação e como atividades iniciais;
- **Média:** deverão ser desenvolvidas na aplicação, mas podem ser desenvolvidas após as funções de alta prioridades;
- **Baixa:** são funcionalidades que poderão ser desenvolvidas por última, podendo ser opcionais.

| ID | Nome | Ator | Resumo/Necessidades | Prioridades |
|-----|---------------------------------|--------------|---|-------------|
| UC1 | Autenticar usuário | A1, A2, e A4 | Possibilitar ao usuário que possuir login, acessar a aplicação web por meio de suas credenciais (login e senha). | Alta |
| UC2 | Gerenciar usuários | A2 | Disponibilizar mecanismo para inserir, alterar, excluir, listar e consultar os dados de usuários da aplicação. Além disso, atribuir um Tipo de Usuário ao Usuário e disponibilizar uma Página na qual o usuário que está logado possa manipular os seus dados pessoais. | Alta |
| UC3 | Solicitar documentos | A3 e A1 | O usuário poderá solicitar algum documento referente a paróquia usando seus dados, gerando uma chave para fins de validação do documento gerado. | Alta |
| UC4 | Agendar atendimento com o padre | A3 | O usuário poderá agendar uma vista com o padre da paróquia, verificando com antecedência uma data e horário disponível. | Alta |
| UC5 | Pré-agendar casamento | A3 | O usuário poderá verificar as datas disponíveis para casamento e a partir disso fazer um pré-agendamento, reservando a data por um período de tempo. | Alta |
| UC7 | Gerenciar liturgia | A1 e A2 | Disponibilizar mecanismo para inserir, alterar, excluir, listar e consultar a liturgia assim. | Média |

Figura 7: Parte 3 - Modelo de Requisitos

| Requisitos Não-Funcionais (Especificação Suplementar) | | |
|--|--|------------------|
| ID | Requisitos | Categoria |
| NF1 | A aplicação deverá ser manipulada através de uma interface gráfica para a Web. | Usabilidade. |
| NF2 | A aplicação deverá funcionar na Internet. | Ambiente. |
| NF3 | A aplicação deverá utilizar Microsoft SQL Server 2005 ou superior como SGDB. | Padronização. |
| NF4 | A aplicação deverá utilizar o Asp.Net 3.5 ou superior como linguagem de programação. | Padronização. |

Figura 8: Parte 4 - Modelo de Requisitos

4.2 ELABORAÇÃO

A figura 9 é um artefato referente a atividade de revisar requisitos que teve um tempo de duração de 4 horas

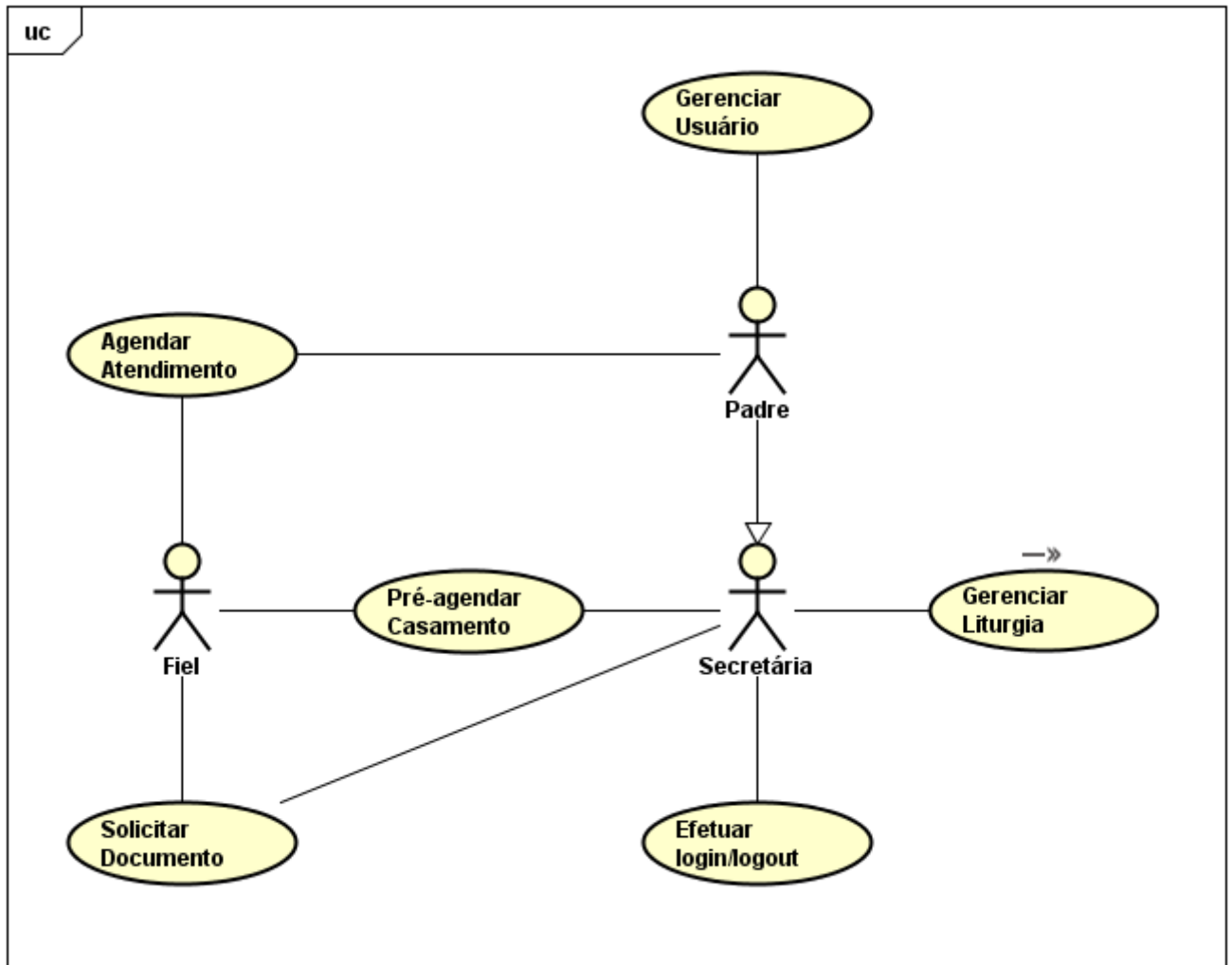


Figura 9: Diagrama de caso de uso

A figura 10 é um artefato referente a atividade de projetar banco de dados que teve um tempo de duração de 16 horas

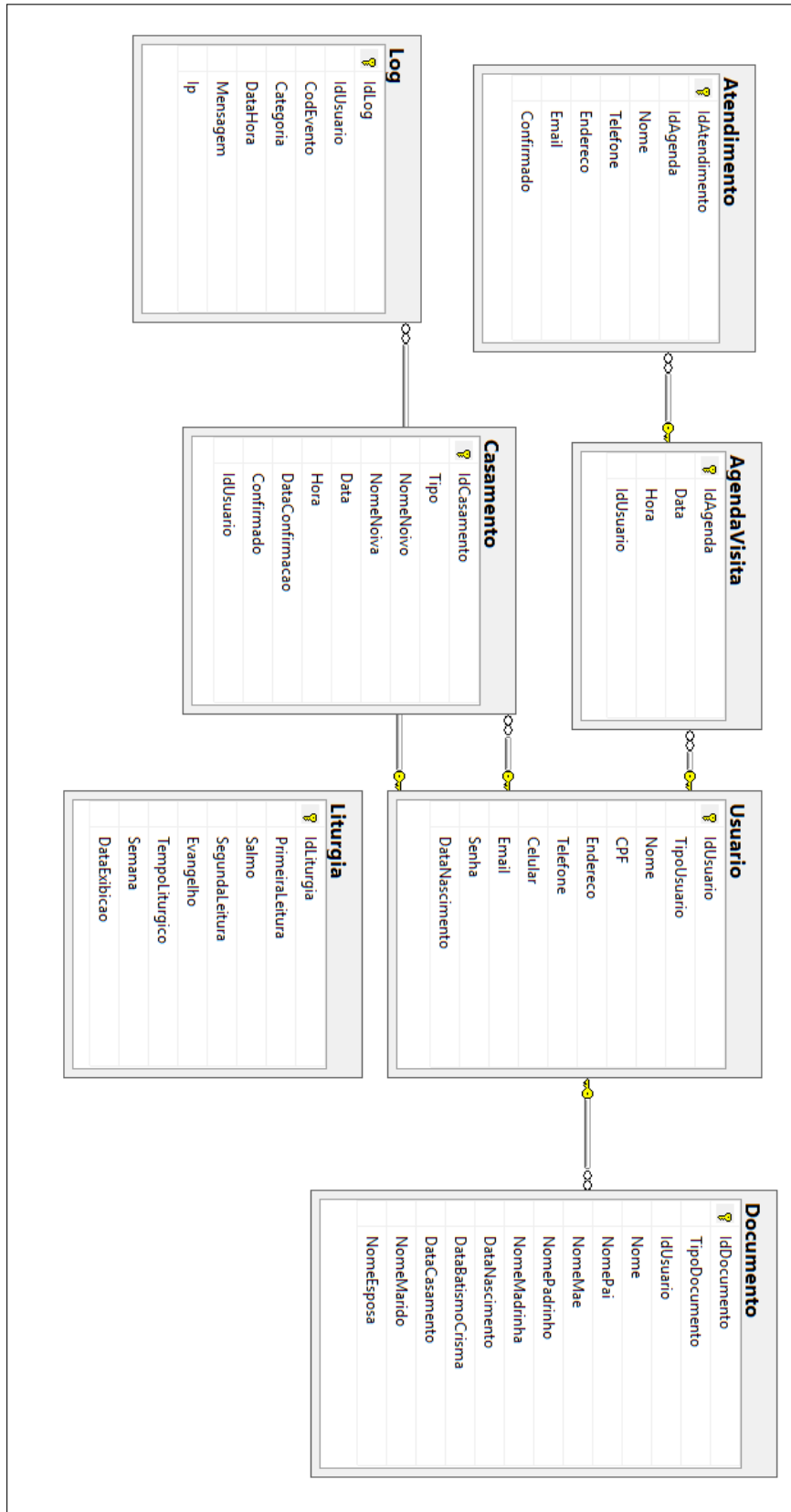


Figura 10: Modelo Entidade-Relacionamento

4.3 CONSTRUÇÃO

A figura 11 até a 13 são artefatos referentes a atividade de construir elementos de multimídia e linguagens de marcação cujo tempo de duração foi de 40 horas

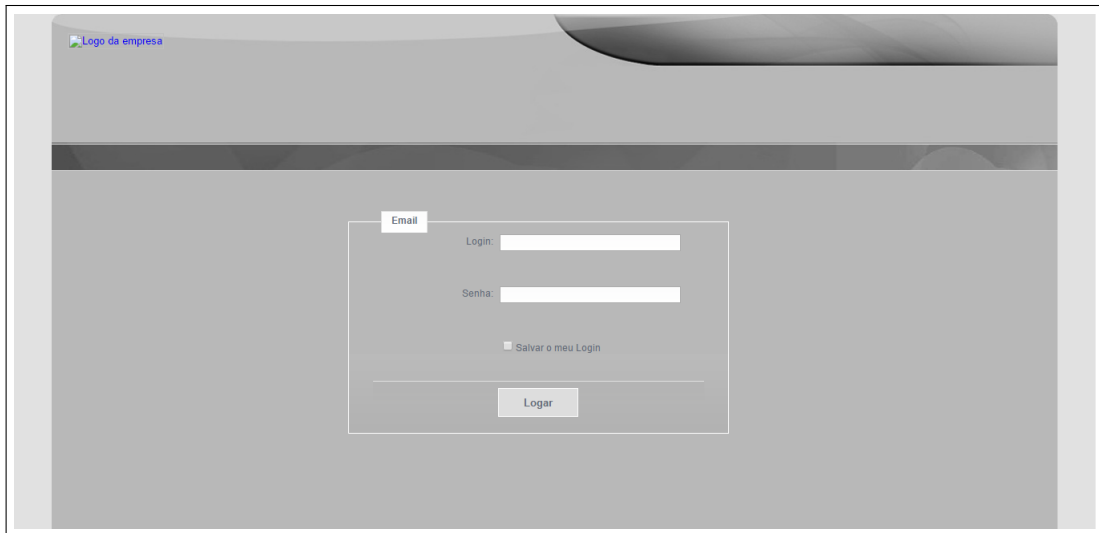


Figura 11: Página de login da parte administrativa



Figura 12: Página de Calendário de Liturgias

Paróquia Cristo Rei Liturgia Solicitações Atendimento Missas

2 Semana tempo

Primeira Leitura
primeira leitura alteração

Salmo
salmo

Segunda Leitura
segunda leitura

Evangelho
evangelho

Figura 13: Página de Liturgia do dia

As figuras de 14 até a 17 são parte do artefato gerado na atividade de implementar componentes cujo tempo de duração foi de 56 horas

```

LiturgiaBO.cs
ParoquiaCristoRei.Regra.LiturgiaBO
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Data.SqlClient;
using System.Data.Sql;
using System.Data.SqlTypes;
using System.Data;
using ParoquiaCristoRei.Regra;
using ParoquiaCristoRei.DAL;
namespace ParoquiaCristoRei.Regra
{
    public class LiturgiaBO
    {
        BancoLINQ<PCR_DALDataContext> Contexto = new BancoLINQ<PCR_DALDataContext>();

        public bool Salvar(Liturgia liturgia, int? idUsuarioLogado, String ipUsuario)
        {
            if (liturgia.IdLiturgia > 0)
                return this.Alterar(liturgia, idUsuarioLogado, ipUsuario);
            else
                return this.Inserir(liturgia, idUsuarioLogado, ipUsuario);
        }
    }
}

```

Figura 14: Codificação da regra de negócio - Classe LiturgiaBO - Método Salvar

```

LiturgiaBO.cs
ParoquiaCristoRei.Regra.LiturgiaBO Contexto
private bool Inserir(Liturgia liturgia, int? idUsuarioLogado, String ipUsuario)
{
    LogBO logBO = new LogBO();
    Log log = new Log();
    string msgLog = "";

    try
    {
        Contexto.BeginTransaction();
        Contexto.DataContext.Liturgia.InsertOnSubmit(liturgia);
        Contexto.DataContext.SubmitChanges();
        Contexto.CommitTransaction();

        msgLog = "A Liturgia [" + liturgia.Semana + " do " + liturgia.TempoLiturgico + "] foi inserida com sucesso [Inserir - LiturgiaBO]";
        log = new Log() { Categoria = (int)CategoriaLog.Informacao, CodEvento = 10007, IdUsuario = idUsuarioLogado, Mensagem = msgLog };
        logBO.Salvar(log);
        return true;
    }
    catch (Exception e)
    {
        Contexto.RollbackTransaction();
        msgLog = "Erro ao inserir Liturgia [Inserir - LiturgiaBO.cs][" + e.Message + "][" + e.Source + "];";
        log = new Log() { Categoria = (int)CategoriaLog.Erro, CodEvento = 30023, IdUsuario = idUsuarioLogado, Mensagem = msgLog };
        logBO.Salvar(log);
        return false;
    }
}
}

```

Figura 15: Codificação da regra de negócio - Classe LiturgiaBO - Método Inserir

```

LiturgiaBO.cs
ParoquiaCristoRei.Regra.LiturgiaBO Contexto
private bool Alterar(Liturgia liturgia, int? idUsuarioLogado, String ip)
{
    LogBO logBO = new LogBO();
    Log log = new Log();
    string msgLog = "";

    try
    {
        Contexto.BeginTransaction();

        Liturgia liturgiaAlterar = this.ConsultarPorId(liturgia.IdLiturgia, idUsuarioLogado);
        liturgiaAlterar.Evangelho = liturgia.Evangelho;
        liturgiaAlterar.PrimeiraLeitura = liturgia.PrimeiraLeitura;
        liturgiaAlterar.SegundaLeitura = liturgia.SegundaLeitura;
        liturgiaAlterar.Salmo = liturgia.Salmo;
        liturgiaAlterar.Semana = liturgia.Semana;
        liturgiaAlterar.TempoLiturgico = liturgia.TempoLiturgico;
        liturgiaAlterar.DataExibicao = liturgia.DataExibicao;

        Contexto.DataContext.SubmitChanges();
        Contexto.CommitTransaction();

        msgLog = "A Liturgia [" + liturgia.Semana + " do " + liturgia.TempoLiturgico + "] foi alterado com sucesso [Alterar - LiturgiaBO.cs]";
        log = new Log() { Categoria = (int)CategoriaLog.Informacao, CodEvento = 10004, IdUsuario = idUsuarioLogado, Mensagem = msgLog };
        logBO.Salvar(log);

        return true;
    }
    catch (Exception e)
    {
        Contexto.RollbackTransaction();

        msgLog = "Erro ao inserir a Liturgia [" + liturgia.Semana + " do " + liturgia.TempoLiturgico + "][Alterar - LiturgiaBO.cs]";
        log = new Log() { Categoria = (int)CategoriaLog.Erro, CodEvento = 30015, IdUsuario = idUsuarioLogado, Mensagem = msgLog };
        logBO.Salvar(log);

        return false;
    }
}

```

Figura 16: Codificação da regra de negócio - Classe LiturgiaBO - Método Alterar

```

LiturgiaBO.cs
ParoquiaCristoRei.Regra.LiturgiaBO Contexto
public bool Excluir(Liturgia liturgia, int? idUsuarioLogado, String ip)
{
    LogBO logBO = new LogBO();
    Log log = new Log();
    string msgLog = "";

    try
    {
        Liturgia liturgiaExcluir = liturgia;

        Contexto.BeginTransaction();

        Contexto.DataContext.Liturgia.DeleteOnSubmit(liturgiaExcluir);
        Contexto.DataContext.SubmitChanges();
        Contexto.CommitTransaction();

        msgLog = "A Liturgia [" + liturgia.Semana + " do " + liturgia.TempoLiturgico + "] foi excluída com sucesso [Excluir - LiturgiaBO.cs]";
        log = new Log() { Categoria = (int)CategoriaLog.Aviso, CodEvento = 20001, IdUsuario = idUsuarioLogado, Mensagem = msgLog };
        logBO.Salvar(log);

        return true;
    }
    catch (Exception e)
    {
        Contexto.RollbackTransaction();

        msgLog = "Erro ao Excluir Liturgia [" + liturgia.Semana + " do " + liturgia.TempoLiturgico + "][Excluir - LiturgiaBO.cs]";
        log = new Log() { Categoria = (int)CategoriaLog.Erro, CodEvento = 30005, IdUsuario = idUsuarioLogado, Mensagem = msgLog };
        logBO.Salvar(log);

        return false;
    }
}

```

Figura 17: Codificação da regra de negócio - Classe LiturgiaBO - Método Excluir

```

LiturgiaBO.cs
ParoquiaCristoRei.Regra.LiturgiaBO Contexto

public Liturgia ConsultarPorId(int id, int? idUsuarioLogado)
{
    LogBO logBo = new LogBO();
    Log log = new Log();
    string msgLog = "";

    try
    {
        return Contexto.DataContext.Liturgia.Single(Liturgia => Liturgia.IdLiturgia == id);
    }
    catch (Exception e)
    {
        msgLog = "Erro ao Consultar Liturgia [ConsultarPorId - LiturgiaBO.cs]" + e.Message + "[" + e.Source + "]";
        log = new Log() { Categoria = (int)CategoriaLog.Erro, CodEvento = 30029, IdUsuario = idUsuarioLogado, Mensagem = msgLog };
        logBo.Salvar(log);

        return null;
    }
}

```

Figura 18: Codificação da regra de negócio - Classe LiturgiaBO - Método ConsultarPorId

```

LiturgiaBO.cs
ParoquiaCristoRei.Regra.LiturgiaBO Contexto

public Liturgia ConsultarPorData(DateTime date, int? idUsuarioLogado)
{
    LogBO logBo = new LogBO();
    Log log = new Log();
    string msgLog = "";

    try
    {
        return Contexto.DataContext.Liturgia.Single(Liturgia => Liturgia.DataExibicao == date);
    }
    catch (Exception e)
    {
        msgLog = "Erro ao Consultar Liturgia [ConsultarPorId - LiturgiaBO.cs]" + e.Message + "[" + e.Source + "]";
        log = new Log() { Categoria = (int)CategoriaLog.Erro, CodEvento = 30029, IdUsuario = idUsuarioLogado, Mensagem = msgLog };
        logBo.Salvar(log);

        return null;
    }
}

```

Figura 19: Codificação da regra de negócio - Classe LiturgiaBO - Método ConsultarPorData

```

LiturgiaBO.cs
ParoquiaCristoRei.Regra.LiturgiaBO Contexto

public List<Liturgia> ConsultarTodos(int? idUsuarioLogado)
{
    LogBO logBo = new LogBO();
    Log log = new Log();
    string msgLog = "";

    try
    {
        List<Liturgia> lista = new List<Liturgia>();
        var consultarLiturgia = from Liturgia in Contexto.DataContext.Liturgia orderby Liturgia.IdLiturgia descending select Liturgia;
        return lista = consultarLiturgia.ToList();
    }
    catch (Exception e)
    {
        msgLog = "Erro ao Consultar todas as Liturgias [ConsultarTodos - LiturgiaBO.cs]" + e.Message + "[" + e.Source + "]";
        log = new Log() { Categoria = (int)CategoriaLog.Erro, CodEvento = 30029, IdUsuario = idUsuarioLogado, Mensagem = msgLog };
        logBo.Salvar(log);

        return null;
    }
}

```

Figura 20: Codificação da regra de negócio - Classe LiturgiaBO - Método ConsultarPorTodos

5 CONSIDERAÇÕES FINAIS

Esta monografia teve como objetivo desenvolver um *website* para a Paróquia Cristo Rei de Cornélio Procópio, de forma que o mesmo atue de forma responsiva.

Para tal desenvolvimento o método escolhido foi o WAP, que foi dividido em 4 fases, sendo elas: iniciação, elaboração, construção e implantação. Cada fase dessa teve uma iteração que percorreram uma série de atividades para guiar o processo de forma concisa e objetiva.

Cada ciclo de iteração se encerrou com o planejamento do próximo, definindo prazos para cada atividade, até que o produto final estivesse completo.

A partir dessas atividades foram gerados artefatos e produtos de trabalho que foram apresentados no capítulo anterior.

Ao final do projeto espera-se que com o *website* a Paróquia consiga manter um relacionamento mais próximo com seus fiéis através de uma comunicação virtual.

5.1 IMPLEMENTAÇÃO FUTURA

Como possível implementação futura, ficou em aberto a parte de gerenciamento do Calendário Paroquial, que listaria os eventos da Paróquia, contendo fotos de eventos passados em um calendário organizado e transparente à comunidade cristã.

Os dias que houveram eventos, ficariam em destaque e ao acessados levariam à uma página com os dados do evento e possíveis fotos tiradas na data em questão.

5.2 DIFICULDADES E OBSERVAÇÕES

Na atividade de implementar código de marcação e estilo houve um atraso, devido a parte de *design* do *website* assim como o tratamento de imagens para o mesmo ficar responsável por outro departamento da empresa na qual o projeto foi desenvolvido, porém esse departamento não entregou o material solicitado para o andamento do projeto.

Ao longo do processo, a descrição dos casos de uso e a estrutura do banco de dados precisaram ser modificadas diversas vezes devido a uma falha de comunicação na definição dos requisitos, consequentemente perdeu-se um tempo médio não previsto para realizar esses ajustes. Após esse contratempo o processo seguiu sem nenhum outro imprevisto.

REFERÊNCIAS

GOVONI, D. **Java application frameworks**. [S.l.]: Wiley, 1999.

MICROSOFT. **ASP.NET**. [S.l.]. Disponível em: <<http://www.asp.net/>>.

NETO, M. F. de A. CustomizaÇÃo de processo para aplicaÇões web com princÍpios das metodologias msf, rup, webe e cmmi. **DEPARTAMENTO DE COMPUTAÇÃO CURSO DE PÓS-GRADUAÇÃO EM ENGENHARIA DE SOFTWARE COM UML**, 2008.

NIELSEN. 68 milhões usam a internet pelo smartphone no brasil. **Mobile Report**, 2015. Disponível em: <<http://www.nielsen.com/br/pt/press-room/2015/68-milhoes-usam-a-internet-pelo-smartphone-no-Brasil.html>>.

PEWGLOBAL. Emerging nations embrace internet mobile technology. 2014. Disponível em: <<http://www.pewglobal.org/2014/02/13/emerging-nations-embrace-internet-mobile-technology/>>.

REISSWITZ, F. **Análise De Sistemas V. 7**. [S.l.: s.n.], 2009.

RODRIGUES, W. C. Metodologia científica. 2007.

SANTOS, J. A hierarquia da igreja católica. **História da Igreja**, 2002. Disponível em: <<http://www.catequisar.com.br/texto/colunas/juberto/30.htm>>.

TWITTER. **Bootstrap**. [S.l.].

6 ANEXOS

- Atividade: detalhar e refinar a aplicação - Tempo de duração: 20 horas

| <h1>Especificação de Caso de Uso</h1> <h2>Autenticar usuário - uc1</h2> | | | |
|---|--------|---------------------|-------------------|
| <h3>Histórico da Revisão</h3> | | | |
| Data | Versão | Descrição | Autor |
| 11/01/2016 | 1.0 | Adições dos cenário | Rodrigo Fernandes |
| | | | |
| | | | |

Figura 21: Parte 1 - Descrição do Caso de Uso Autenticar Usuário

Especificação de Caso de Uso

Autenticar usuário - uc1

Objetivo

O principal objetivo deste documento é especificar o caso de uso Autenticar usuário e seus respectivos cenários.

Figura 22: Parte 2 - Descrição do Caso de Uso Autenticar Usuário

Cenários

Cenário 1 - Login no sistema

Fluxo Normal

| | | |
|--|--|--|
| ID: | uc1c1 | |
| Ator principal: | Padre e Secretária | |
| Ator secundário: | | |
| Resumo: | Passos específicos para se logar no sistema. | |
| Pré-condição: | O ator precisa ter uma conta cadastrada. | |
| Ações do Ator: | Ações do Sistema: | |
| 1. Digita o nome de usuário e a senha em seus respectivos campos e clica em "Login". | 2. Sistema verifica autenticidade do usuário e confere a senha digitada com a cadastrada no sistema e retorna a página inicial do site após confirmação. 2.1 Caso as informações não coincidam, sistema exibe mensagem de erro ao tentar fazer login. | |
| Pós-condição: | | |
| Obs.: | | |
| Documentos relacionados: | | |

Figura 23: Parte 3 - Descrição do Caso de Uso Autenticar Usuário

| Cenário 2 - Logout | |
|-------------------------------------|---|
| Fluxo Normal | |
| ID: | uc1c2 |
| Ator principal: | Secretária, Padre e Usuário |
| Ator secundário: | |
| Resumo: | Passos específicos para efetuar um logout no sistema. |
| Pré-condição: | O ator precisa estar logado no sistema. |
| Ações do Ator: | Ações do Sistema: |
| 1. Usuário clica no botão “logout”. | 2. Sistema destrói a sessão atual e retorna a página inicial. |
| Pós-condição: | |
| Obs.: | |
| Documentos relacionados: | |

Figura 24: Parte 4 - Descrição do Caso de Uso Autenticar Usuário

Especificação de Caso de Uso

Gerenciar Usuários - uc2

Histórico da Revisão

| Data | Versão | Descrição | Autor |
|------------|--------|----------------------|-------------------|
| 11/01/2016 | 1.0 | Adições dos cenários | Rodrigo Fernandes |
| | | | |
| | | | |

Figura 25: Parte 1 - Descrição do Caso de Uso Gerenciar Usuário

Especificação de Caso de Uso

Gerenciar Usuários - uc2

Objetivo

O principal objetivo deste documento é especificar o caso de uso Gerenciar usuários e seus respectivos cenários.

Figura 26: Parte 2 - Descrição do Caso de Uso Gerenciar Usuário

Cenários

Cenário 1 - Buscar usuários

Fluxo Normal

| | | |
|--|---|--|
| ID: | uc2c1 | |
| Ator principal: | Secretária e Padre | |
| Ator secundário: | | |
| Resumo: | Passos específicos para se buscar um ou mais usuários. | |
| Pré-condição: | O ator precisa estar logado no sistema. | |
| Ações do Ator: | Ações do Sistema: | |
| 1. Digita o nome do usuário no campo de busca e clica em "buscar". | 2. Retorna lista com os usuários que possuem o mesmo nome que foi digitado. | |
| Pós-condição: | | |
| Obs.: | | |
| Documentos relacionados: | | |

Figura 27: Parte 3 - Descrição do Caso de Uso Gerenciar Usuário

Cenário 2 - Adicionar usuário

Fluxo Normal

| | | |
|--|--|--|
| ID: | uc2c2 | |
| Ator principal: | Secretária e Padre | |
| Ator secundário: | | |
| Resumo: | Passos específicos para se adicionar um usuário. | |
| Pré-condição: | O ator precisa estar logado no sistema. | |
| Ações do Ator: | Ações do Sistema: | |
| 1. Preenche os campos com as informações pessoais do usuário que deseja adicionar e clica em "Salvar". | 2. Verifica se não há duplicidade nas informações digitadas com as já cadastradas no banco e mostra mensagem de confirmação. 2.1. Exibe mensagem de erro caso haja duplicidade nas informações. | |
| Pós-condição: | | |
| Obs.: | | |
| Documentos relacionados: | | |

Figura 28: Parte 4 - Descrição do Caso de Uso Gerenciar Usuário

| Cenário 3 - Alterar usuário | |
|---|---|
| Fluxo Normal | |
| ID: | uc2c3 |
| Ator principal: | Padre |
| Ator secundário: | |
| Resumo: | Passos específicos para se alterar um usuário. |
| Pré-condição: | O padre precisa estar logado no sistema; uc2c1 precisa ter ocorrido. |
| Ações do Ator: | Ações do Sistema: |
| <ol style="list-style-type: none"> 1. Seleciona um usuário da lista e clica em “Editar”. 3. Altera os campos necessários e clica em “Salvar”. | <ol style="list-style-type: none"> 2. Abre página com as informações do usuário selecionado já preenchidas nos campos. 4. Verifica se todos os campos estão preenchidos e exibe mensagem de confirmação. 4.1 Exibe mensagem de erro caso algum campo não esteja preenchido corretamente. |
| Pós-condição: | |
| Obs.: | |
| Documentos relacionados: | |

Figura 29: Parte 5 - Descrição do Caso de Uso Gerenciar Usuário

| Cenário 4 - Excluir usuário | |
|--|---|
| Fluxo Normal | |
| ID: | uc2c4 |
| Ator principal: | Padre |
| Ator secundário: | |
| Resumo: | Passos específicos para excluir um usuário |
| Pré-condição: | O padre precisa estar logado no sistema; uc2c1 precisa ter ocorrido |
| Ações do Ator: | Ações do Sistema: |
| 1. Seleciona um usuário da lista e clica em "Excluir". | 2. Exclui o usuário e seus dados do sistema. |
| Pós-condição: | |
| Obs.: | |
| Documentos relacionados: | |

Figura 30: Parte 6 - Descrição do Caso de Uso Gerenciar Usuário

Cenário 5 - Alterar informações

Fluxo Normal

| | | |
|--|---|--|
| ID: | uc2c5 | |
| Ator principal: | Padre, secretária | |
| Ator secundário: | | |
| Resumo: | Passos específicos para se alterar informações do ator logado. | |
| Pré-condição: | O ator precisa estar logado no sistema. | |
| Ações do Ator: | Ações do Sistema: | |
| <ol style="list-style-type: none"> 1. Clica em “Editar meus dados”. 3. Altera os campos necessários e clica em “Salvar”. | <ol style="list-style-type: none"> 2. Abre página com as informações do ator logado já preenchidas nos campos. 4. Verifica se todos os campos estão preenchidos e exibe mensagem de confirmação. 4.1 Exibe mensagem de erro caso algum campo não esteja preenchido corretamente. | |
| | | |
| Pós-condição: | | |
| Obs.: | | |
| Documentos relacionados: | | |

Figura 31: Parte 7 - Descrição do Caso de Uso Gerenciar Usuário

| Especificação de Caso de Uso | | | |
|-------------------------------------|---------------|---------------------|-------------------|
| Solicitar documentos- uc3 | | | |
| Histórico da Revisão | | | |
| Data | Versão | Descrição | Autor |
| 11/01/2016 | 1.0 | Adições dos cenário | Rodrigo Fernandes |
| | | | |
| | | | |

Figura 32: Parte 1 - Descrição do Caso de Uso Solicitar um Documento

| Especificação de Caso de Uso | |
|---|--|
| Soliticitar documentos- uc3 | |
| Objetivo | |
| <hr/> <p>O principal objetivo deste documento é especificar o caso de uso Solicitar documentos e seus respectivos cenários.</p> | |

Figura 33: Parte 2 - Descrição do Caso de Uso Solicitar um Documento

| Cenários | |
|--|--|
| Cenário 1 - Solicitar documentos | |
| Fluxo Normal | |
| ID: | uc3c1 |
| Ator principal: | Usuário |
| Ator secundário: | Secretária |
| Resumo: | Passos específicos para se listar eventos. |
| Pré-condição: | |
| Ações do Ator: | Ações do Sistema: |
| <ol style="list-style-type: none"> 1. Usuário escolhe o tipo de documento a ser solicitado. 3. Usuário preenche os campos necessário e clica em “Enviar informações” | <ol style="list-style-type: none"> 2. Retorna formulário com campos necessários de acordo com o tipo de documento. 4. Verifica se todos os campos foram preenchidos corretamente, gera uma inserção no banco com os dados do documento solicitado e envia um email para o atendimento com a solicitação. 4.1. Exibe mensagem alertando que os campos não foram preenchidos. |
| Pós-condição: | |
| Obs.: | |
| Documentos relacionados: | |

Figura 34: Parte 3 - Descrição do Caso de Uso Solicitar um Documento

| Especificação de Caso de Uso | | | |
|---------------------------------------|---------------|---------------------|-------------------|
| Agendar visita com padre - uc4 | | | |
| Histórico da Revisão | | | |
| Data | Versão | Descrição | Autor |
| 11/01/2016 | 1.0 | Adições dos cenário | Rodrigo Fernandes |
| | | | |
| | | | |

Figura 35: Parte 1 - Descrição do Caso de Uso Agendar Atendimento com o Padre

| Especificação de Caso de Uso |
|---|
| Agendar visita com padre - uc4 |
| Objetivo |
| <hr/> <p>O principal objetivo deste documento é especificar o caso de uso Agendar visita com o padre e seus respectivos cenários.</p> |

Figura 36: Parte 2 - Descrição do Caso de Uso Agendar Atendimento com o Padre

Cenários

Cenário 1 - Agendar visita

Fluxo Normal

| | | |
|---|--|--|
| ID: | uc4c1 | |
| Ator principal: | Usuário | |
| Ator secundário: | | |
| Resumo: | Passos específicos para agendar visita com padre. | |
| Pré-condição: | | |
| Ações do Ator: | Ações do Sistema: | |
| 1. Escolher uma data e horário disponível no calendário e clicar em "Agendar visita". | 2. Atualiza o calendário e marca a data e o horários escolhidos como indisponível e exibe mensagem de confirmação. | |
| Pós-condição: | | |
| Obs.: | | |
| Documentos relacionados: | | |

Figura 37: Parte 3 - Descrição do Caso de Uso Agendar Atendimento com o Padre

| Especificação de Caso de Uso | | | |
|-------------------------------------|---------------|---------------------|-------------------|
| Pré-agendar casamento - uc5 | | | |
| Histórico da Revisão | | | |
| Data | Versão | Descrição | Autor |
| 11/01/2016 | 1.0 | Adições dos cenário | Rodrigo Fernandes |
| | | | |
| | | | |

Figura 38: Parte 1 - Descrição do Caso de Uso Pré-agendar Casamento

| Especificação de Caso de Uso |
|--|
| Pré-agendar casamento - uc5 |
| Objetivo |
| <hr/> <p>O principal objetivo deste documento é especificar o caso de uso Pré-agendar casamento e seus respectivos cenários.</p> |

Figura 39: Parte 2 - Descrição do Caso de Uso Pré-agendar Casamento

| Cenários | |
|--|--|
| Cenário 1 - Pré-agendar casamento | |
| Fluxo Normal | |
| ID: | uc5c1 |
| Ator principal: | Usuário |
| Ator secundário: | |
| Resumo: | Passos específicos para pré-agendar casamento. |
| Pré-condição: | |
| Ações do Ator: | Ações do Sistema: |
| 1. Escolher uma data e horário disponível no calendário e clicar em “Pré-agendar”. | 2. Atualiza o calendário e marca a data e o horários escolhidos como indisponível por 30 dias e exibe mensagem avisando que a data voltará a ficar disponível no período de 30 dias, caso não haja confirmação por parte do usuário. |
| Pós-condição: | |
| Obs.: | |
| Documentos relacionados: | |

Figura 40: Parte 3 - Descrição do Caso de Uso Pré-agendar Casamento

Especificação de Caso de Uso

Gerenciar liturgia - uc6

Histórico da Revisão

| Data | Versão | Descrição | Autor |
|------------|--------|---------------------|-------------------|
| 11/01/2016 | 1.0 | Adições dos cenário | Rodrigo Fernandes |
| | | | |
| | | | |

Figura 41: Parte 1 - Descrição do Caso de Uso Gerenciar Liturgia

Especificação de Caso de Uso

Gerenciar liturgia - uc6

Objetivo

O principal objetivo deste documento é especificar o caso de uso Gerenciar liturgia e seus respectivos cenários.

Figura 42: Parte 2 - Descrição do Caso de Uso Gerenciar Liturgia

| Cenários | |
|------------------------------------|---|
| Cenário 1 - Listar liturgia | |
| Fluxo Normal | |
| ID: | uc6c1 |
| Ator principal: | Padre e secretária |
| Ator secundário: | |
| Resumo: | Passos específicos para se listar liturgia. |
| Pré-condição: | O ator precisa estar logado no sistema. |
| Ações do Ator: | Ações do Sistema: |
| 1. Clica em “Listar liturgias”. | 2. Lista todas as liturgias cadastradas. |
| Pós-condição: | |
| Obs.: | |
| Documentos relacionados: | |

Figura 43: Parte 3 - Descrição do Caso de Uso Gerenciar Liturgia

Cenário 2 - Adicionar liturgia

Fluxo Normal

| | | |
|---|---|--|
| ID: | uc6c2 | |
| Ator principal: | Secretária e Padre | |
| Ator secundário: | | |
| Resumo: | Passos específicos para se adicionar uma liturgia. | |
| Pré-condição: | O ator precisa estar logado no sistema. | |
| Ações do Ator: | Ações do Sistema: | |
| 1. Preenche os campos com as informações da liturgia juntamente com a data e clica em “Salvar”. | 2. Verifica se os campos foram preenchidos corretamente e se não há duplicidade nas informações digitadas com as já cadastradas no banco e mostra mensagem de confirmação. 2.1. Exibe mensagem de erro caso haja duplicidade nas informações ou se os campos não foram preenchidos corretamente. | |
| Pós-condição: | | |
| Obs.: | | |
| Documentos relacionados: | | |

Figura 44: Parte 4 - Descrição do Caso de Uso Gerenciar Liturgia

Cenário 3 - Alterar liturgia

Fluxo Normal

| | | |
|---|--|--|
| ID: | uc6c3 | |
| Ator principal: | Padre e secretária | |
| Ator secundário: | | |
| Resumo: | Passos específicos para se alterar uma liturgia. | |
| Pré-condição: | O ator precisa estar logado no sistema; uc6c1 precisa ter ocorrido. | |
| Ações do Ator: | Ações do Sistema: | |
| <ol style="list-style-type: none"> 1. Seleciona uma liturgia da lista e clica em “Editar”. 3. Altera os campos necessários e clica em “Salvar”. | <ol style="list-style-type: none"> 2. Abre página com as informações da liturgia selecionado já preenchidas nos campos. 4. Verifica se todos os campos estão preenchidos e exibe mensagem de confirmação. 4.1 Exibe mensagem de erro caso algum campo não esteja preenchido corretamente. | |
| Pós-condição: | | |
| Obs.: | | |
| Documentos relacionados: | | |

Figura 45: Parte 5 - Descrição do Caso de Uso Gerenciar Liturgia

Cenário 4 - Excluir liturgia

Fluxo Normal

| | | |
|--|--|--|
| ID: | uc6c4 | |
| Ator principal: | Padre e secretária | |
| Ator secundário: | | |
| Resumo: | Passos específicos para excluir uma liturgia | |
| Pré-condição: | O ator precisa estar logado no sistema; uc6c1 precisa ter ocorrido | |
| Ações do Ator: | Ações do Sistema: | |
| 1. Seleciona uma liturgia da lista e clica em "Excluir". | 2. Exclui a liturgia e seus dados do sistema. | |
| | | |
| Pós-condição: | | |
| Obs.: | | |
| Documentos relacionados: | | |

Figura 46: Parte 6 - Descrição do Caso de Uso Gerenciar Liturgia

- Atividade: projetar arquitetura de usabilidade/projetar arquitetura da aplicação - Tempo de duração: 16 horas



Website Paróquia Cristo Rei

Arquitetura Global da Aplicação

Histórico da Revisão

| Data | Versão | Descrição | Autor |
|------------|--------|-----------------------|---------|
| 28/01/2016 | 1.0 | Preenchimento inicial | Rodrigo |
| | | | |
| | | | |
| | | | |
| | | | |

Figura 47: Parte 1 - Arquitetura Global da Aplicação



Figura 48: Parte 2 - Arquitetura Global da Aplicação

3. Metas e Restrições de Arquitetura

Restrições de Segurança

- Todas as páginas administrativas devem verificar se o usuário está logado e se o mesmo possui acesso a página acessada;
- Todos os SQL devem ser armazenados no SGBD por meio de Stored Procedure, caso não seja mapeado no modelo LINQ;

Restrições de Design

- Todas as WUI devem acessar apenas objetos providos da Regra de Negócio;
- As WUI não podem acessar diretamente o banco de dados;

Restrições da WUI

- Utilização de CSS para estilização das páginas;
- Toda WUI deve herdar a Master Page.

Figura 49: Parte 3 - Arquitetura Global da Aplicação

4. Visão de Casos de Uso

| ID | Nome |
|-----|---------------------------------|
| UC1 | Autenticar usuário |
| UC2 | Gerenciar usuários |
| UC3 | Solicitar documentos |
| UC4 | Agendar atendimento com o padre |
| UC5 | Pré-agendar casamento |
| UC7 | Gerenciar liturgia |

Figura 50: Parte 4 - Arquitetura Global da Aplicação

5. Visão Lógica

5.1 Visão Geral

Toda página WUI deve acessar objetos da Regra de Negócio e esses objetos acessam o SGBD, como ilustra o Diagrama de Seqüência.

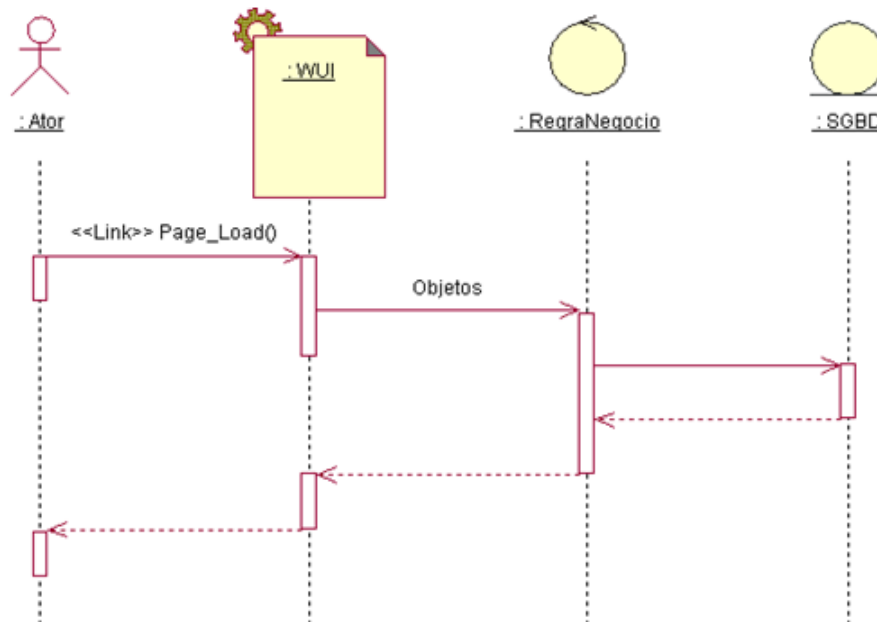


Figura 51: Parte 5 - Arquitetura Global da Aplicação

5.2 Padrões para Implementação

- Toda classe de Regra de Negócio deve, no mínimo, seguir os seguintes:
 - Ter o nome BO (Business Object) em seu nome;
 - Salvar() : método público que retorna um bool. É Responsável por persistir os dados de um objeto e o mesmo deve automatizar o Inserir e Alterar.
 - Inserir(): método privado que retorna um bool. É responsável pela inserção de um novo objeto e é chamado pelo Persistir().
 - Alterar(): método privado que retorna um bool. É responsável pela alteração dos dados de um objeto é chamado pelo Persistir().
 - Excluir() : método público que retorna um bool. É Responsável por excluir um objeto.
 - ConsultarTodos(): método público com retorno de Coleção. Responsável pela consulta de todos os objetos de uma determinada classe da Regra de Negócio. Cada objeto da Coleção deve conter um conjunto mínimo de dados preenchidos.
 - ConsultarPorID(): método publico com retorno de um objeto. Responsável pela consulta de todos os dados de um determinado objeto da Regra de Negócio.

Figura 52: Parte 6 - Arquitetura Global da Aplicação

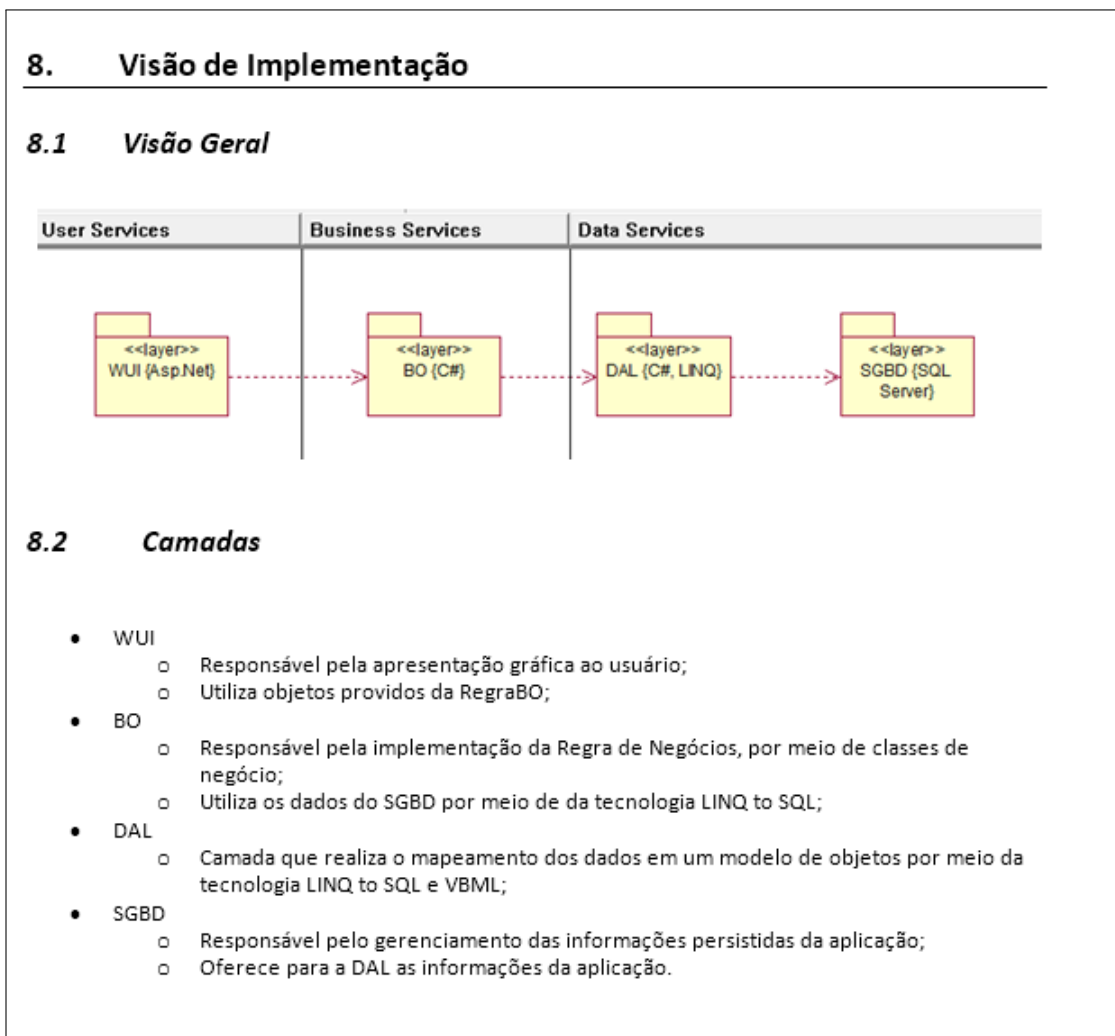


Figura 53: Parte 7 - Arquitetura Global da Aplicação

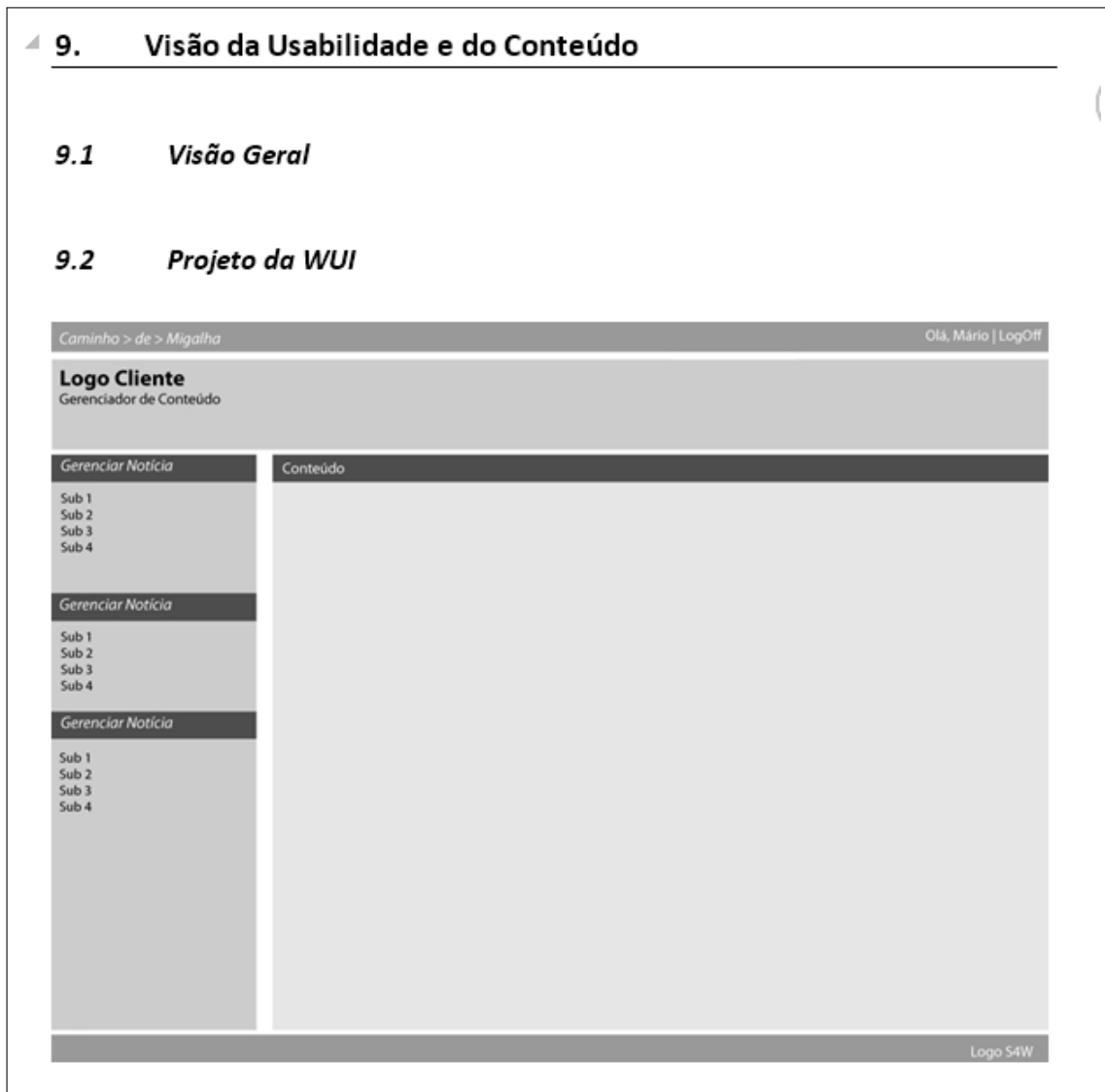


Figura 54: Parte 8 - Arquitetura Global da Aplicação

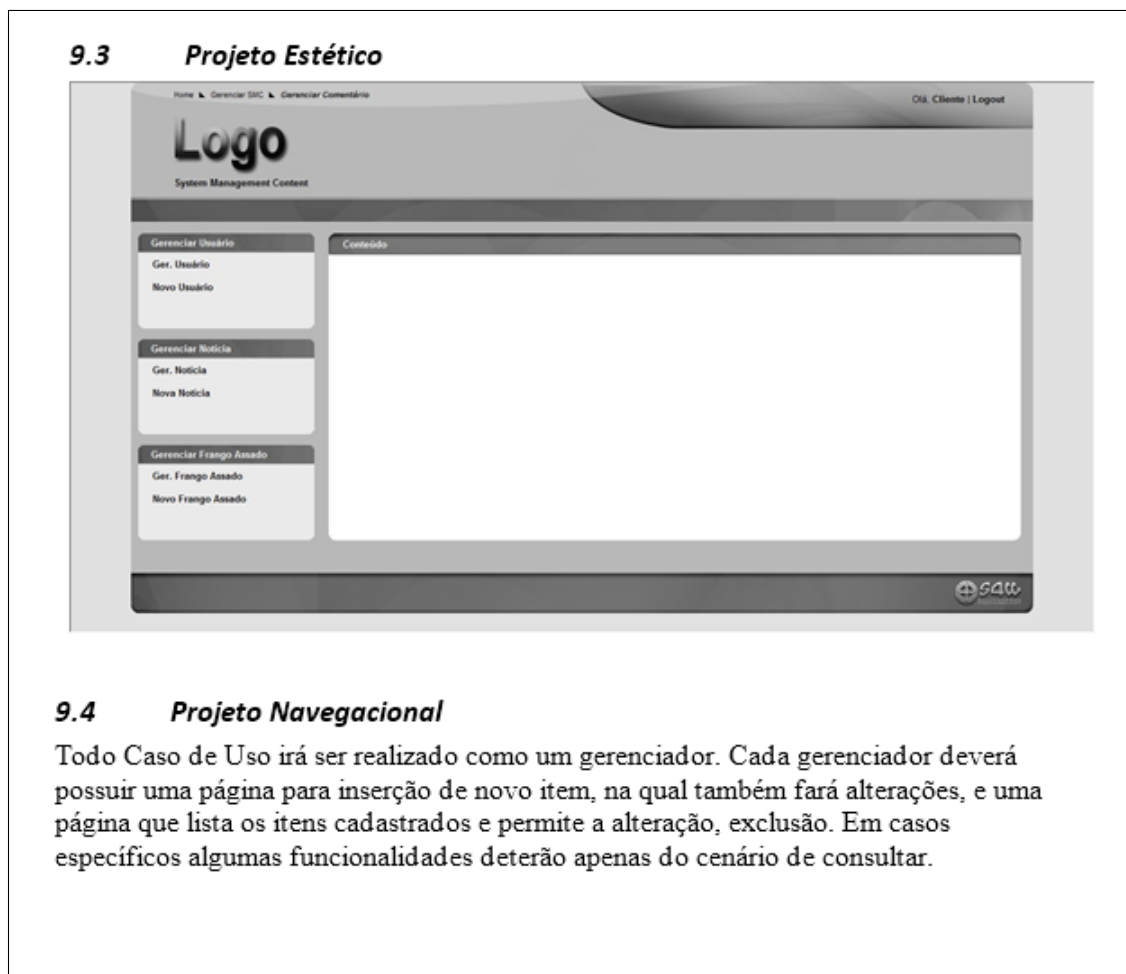


Figura 55: Parte 9 - Arquitetura Global da Aplicação

- Atividade: projetar caso de uso - Tempo de duração: 4 horas

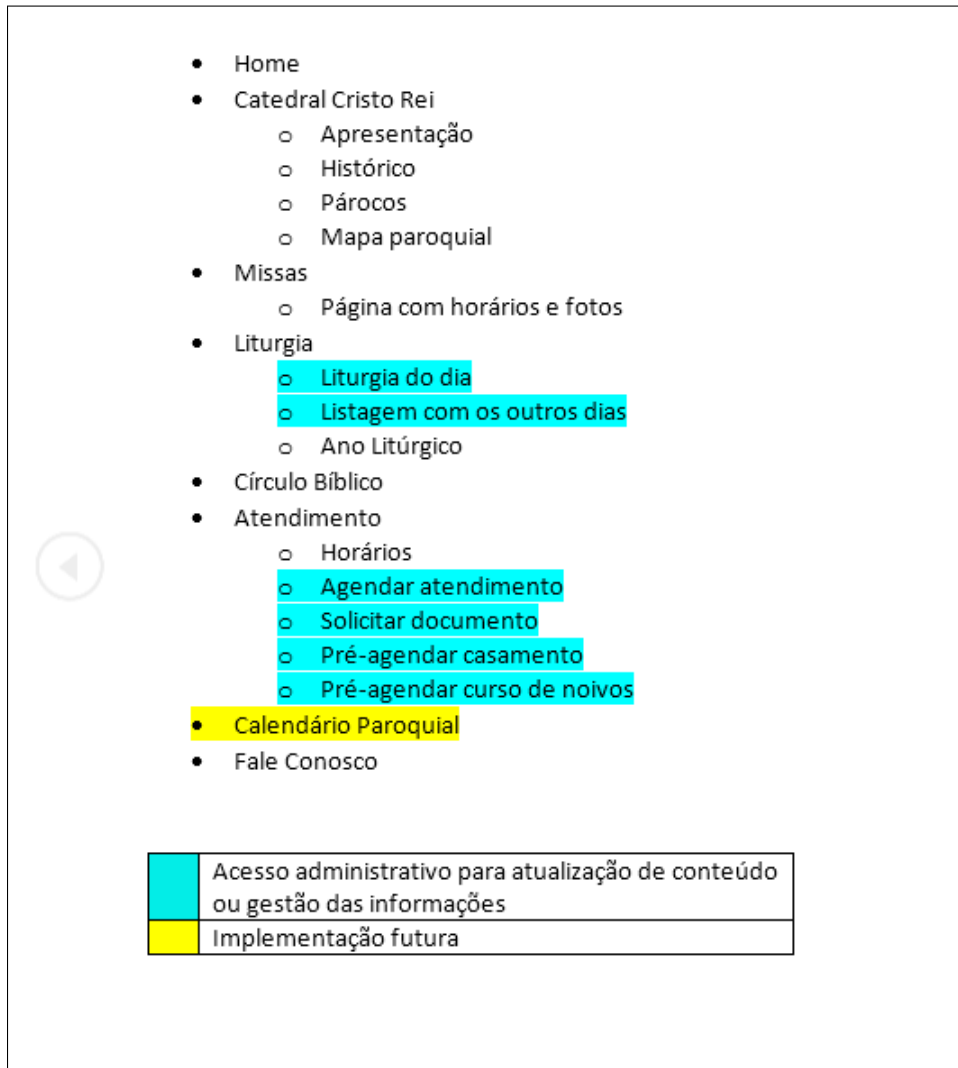


Figura 56: Modelo Navegacional do contendo as páginas