



UNIVERSIDADE LUSÓFONA
de Humanidades e Tecnologias
Humani nihil alienum

TFC – Trabalho Final de Curso

Licenciatura de Engenharia Informática

2013-2014



Sistema YOU ARE INVITED

Trabalho realizado por:

Higino Espada, N.º 21006198

Orientador:

Prof. Adriano Couto

ÍNDICE

ÍNDICE DE ILUSTRAÇÕES.....	3
ÍNDICE DE TABELAS	3
ACRÓNIMOS	4
ABSTRACT	5
INTRODUÇÃO.....	5
Propósito e Justificação	6
Motivação	6
Descrição Geral do Problema	7
VISÃO GERAL DO SISTEMA.....	8
O Sistema	8
Stakeholders e Utilizadores	9
Cálculo do Peso de cada Stakeholders	10
ANÁLISE DE REQUISITOS	10
Requisitos Funcionais	10
a) Funcionalidades do Administrador.....	10
b) Funcionalidades dos Noivos.....	11
c) Funcionalidades dos Convidados	12
d) Funcionalidades dos Lojistas	13
e) Funcionalidades Comuns	14
Requisitos Não-Funcionais.....	14
Factores Críticos	15
Cálculo dos factores críticos por Stakeholder.....	17
Processos e Tarefas	18
DESENHO	20
Arquitectura.....	20
Diagrama E/R	22
BPMN	23
UML.....	23
Diagramas de Casos de Uso.....	23
Diagrama de Classes	26
Diagramas de Sequência.....	27
FRAMEWORKS – FERRAMENTAS UTILIZADAS.....	28
Microsoft Visual Studio	28

Classes Externas	29
IMPLMENTAÇÃO - DETALHES	31
Android	31
ASP.Net.....	32
CONCLUSÃO.....	35
GLOSSÁRIO.....	36
BIBLIOGRAFIA	38
ANEXOS.....	39

ÍNDICE DE ILUSTRAÇÕES

Ilustração 1: visão geral do evento	8
Ilustração 2: representação sumária dos convidados.....	12
Ilustração 3: apresentação da lista de prendas	13
Ilustração 4: número de avisos em destaque.....	14
Ilustração 5: arquitectura do sistema YAI	20
Ilustração 6: diagrama E/R	22
Ilustração 7: C2 - diagrama de classes gerado pelo Entity Framework.....	27
Ilustração 8: activities da aplicação Android	32
Ilustração 9: exemplo código reservar uma prenda	33
Ilustração 10: organização das páginas do YAI	34
Ilustração 11: apresentação dos dados agrupados e totais	34

ÍNDICE DE TABELAS

Tabela 1: peso stakeholders	10
Tabela 2: cálculo da importância dos factores críticos de sucesso	17
Tabela 3: casos de uso e actores	24
Tabela 4: CU1 - identificação dos use cases por actor.....	25
Tabela 5: CU2 - identificação dos use cases por actor.....	26

ACRÓNIMOS

YAI	You Are Invited
S1, S2, S3, S4	Stakeholders 1, 2, 3 e 4
U1, U2, U3, U4	Utilizadores 1, 2, 3 e 4
RF01 → RF16	Requisitos Funcionais 1 a 16
RNF01 → RNF07	Requisitos Não-Funcionais 1 a 7
CU1, CU2	Diagramas de Casos de Uso 1 e 2
C1	Diagrama de Classes do Sistema YAI - UML
C2	Diagrama de Classes do Sistema YAI - Entity Framework
S1, S2	Diagramas de Sequência 1 e 2
BD	Base-de-Dados
PC	Computador Pessoal
WEB	World Wide Web
PDF	Portable Document Format
Email	Correio Electrónico
FCS01 → FCS10	Factores Críticos de Sucesso 1 a 10
SI	Sistemas de Informação
OO	Orientada a Objectos

ABSTRACT

This report describes a system, named “You Are Invited” for management wedding ceremonies. Helps the wedding couple to organize the ceremony and interact with the guests.

It is a user-friendly Web System, with a Mobile version (Android), intended to help the distribution of guests among the tables, to manage gift lists and publish information about the event. It will add great value to wedding agencies services.

The project started with a analysis of requirements and user interaction, materialized in UML, BPMN and E/R diagrams. The implementation was based on the .Net Framework (C# and LINQ) and SQL server for data storage.

To richer enrich the interface of Web System was also used other classes, such as Bootstrap and GridViewHelper.

A fully functional version is demonstrable and will be in production soon.

INTRODUÇÃO

Pretendo desenvolver um sistema de gestão de eventos, mais especificamente, eventos dirigidos a cerimónias de casamentos. Este sistema, designado “You Are Invited”, procura facilitar aos noivos a organização do seu casamento e ao mesmo tempo a divulgação e interacção com os seus interessados.

Trata-se de um sistema de Web e Móvel direccionado para vários tipos de utilizadores, tais como, os noivos, que pretendem facilmente gerir e divulgar o seu casamento, os convidados, para interactivamente poderem serem informados sobre o casamento da qual foram convidados, os lojistas com o interesse de colocarem produtos na lista de presentes e as respectivas publicidades e promoções.

Com a média de 30.000 casamentos anuais este sistema tem como principais interessados as agências de casamentos, que podem incluir um preço de utilização ao pacote que os noivos comprem para a realização da cerimónia.

Propósito e Justificação

Devido a várias preocupações e tarefas que os noivos têm que atravessar até a concretização do casamento, tais como, gerir um grande número de convidados, mesas e presentes, vi uma oportunidade de desenvolver um sistema “You are Invited” com o objectivo de facilitar os noivos e aumentar o sucesso e satisfação dos seus casamentos.

Um dos módulos mais prioritários que escolhi foi a gestão da parte dos Noivo com a possibilidade de gerir os seus convidados, mesas, prendas e lojistas na Web, também com a possibilidade de cada utilizador poder usar via móvel, em sistemas Android. Ficando com a possibilidade de aumentar potencialidades no sistema como outras implementações e agregar outros interessados.

Exemplos para aumentar potencialidades para o Sistema e agregar outros interessados:

- Restaurantes (inserir ementa, pratos especiais, vinhos, publicidade)
- Espaços (tendas, salões, quintas, flores, automóvel dos noivos, preços, pacotes, horários)
- Fotógrafos (escolha de fotógrafo, preços, inserir fotos tiradas para os convidados escolherem e serem enviadas e vendidas)
- Agências (lua-de-mel, propostas, pacotes, destinos, preços, promoções, reservas)
- Animação (banda, música, dj's, despedidas de solteiro, preços, horários)
- Igrejas (escolha da igreja, cursos, legislação, papelada, horários, disponibilidade)
- Registo Civil (marcações, padrinhos, certidões)

Motivação

O casamento é um dia único e especial em que os noivos pretendem que a cerimónia corra bem e que seja perfeita. A sua organização envolve muitas tarefas, preocupações e pessoas na qual os noivos acabam por gastar muito tempo e custos ao seleccionarem os convidados, obterem confirmações de presenças, reverem e organizarem os convidados por mesas e disponibilizarem listas de presentes. Por isso, sem dúvidas, que o sistema irá tornar mais prático aos Noivos estas tarefas.

Descrição Geral do Problema

O que pode acontecer numa preparação para o casamento na perspectiva dos Noivos:

- Sem o Sistema YAI

- ✓ Quantos convidados temos?
- ✓ Quem são os convidados?
- ✓ Quantos já confirmaram a presença?
- ✓ Quem é que autorizamos mais convidados?
- ✓ Que convidados estão naquela mesa?
- ✓ Esquecemos de alguém?
- ✓ Será que os convidados sabem onde é o casamento? E a que horas? E o nosso roteiro, eles sabem?
- ✓ Quem é que nos ofereceu aquela prenda? (*após casamento*)
- ✓ *Oh não*, presentes duplicados! (*após casamento*)



- Com o Sistema

- ✓ Pretendo facilitar e solucionar todos estes problemas.
- ✓ Registo de todos os convidados, prendas, mesas.
- ✓ Optimização na organização dos convidados, mesas e reserva de prendas.
- ✓ Automatização de avisos.
- ✓ Melhoria na qualidade da concretização dos eventos.

A complexidade e a dimensão dos problemas da gestão de um casamento estão relacionadas com a dimensão do evento que ele próprio já é, e com a diversidade do público-alvo, de facto, são vários os intervenientes, tais como, noivos, convidados, lojista, “fotógrafos”, “agências”, “restaurantes”, com elevada dispersão etária e

experiências com a tecnologia, nomeadamente, com a navegação na Internet e Smartphones.

Sendo os noivos um dos principais utilizadores, eles têm lidar com vários problemas na realização de um casamento das quais relacionam-se com o número de convidados e ter as confirmações de presença o mais cedo possível, atribuir mesas para cada convidado, procurar lojas e disponibilizar a lista de prendas para que não haja ofertas repetidas.

É importante que o Sistema tenha um meio comunicação eficaz com avisos para os utilizadores já que se torna difícil de arranjar e-mails de todos os convidados e de controlar todas as actividades dos mesmos. Devido a haver uma grande variedade de utilizadores é também fundamental que o Sistema tenha uma interacção simples para atrair e facilitar aos utilizadores todas as funcionalidades disponíveis.

VISÃO GERAL DO SISTEMA

O Sistema

Para se ter uma visão muito geral e ajudar a levantar todos os requisitos do Sistema elaborei a seguinte apresentação gráfica;

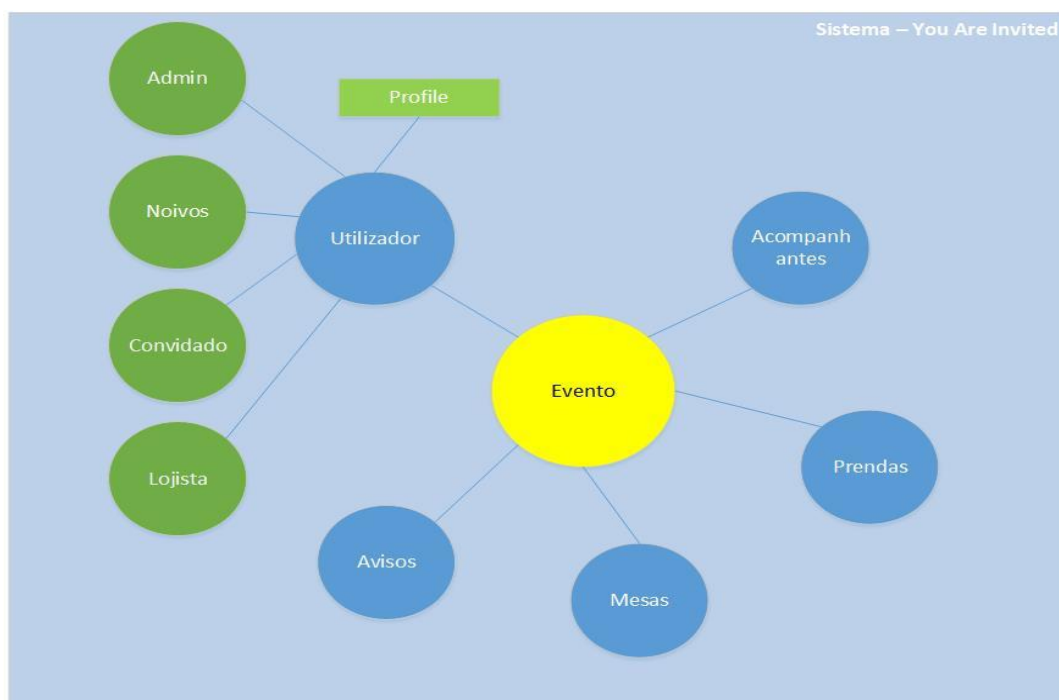


Ilustração 1: visão geral do evento

Para uma compreensão mais profunda de como os dados estão organizados no Sistema está em anexo o desenho do Modelo de Entidade-Relação criado a partido da ADO.Net (ADO.NET explicado no capítulo das FRAMEWORKS).

Quanto a este diagrama, podemos compreender que existe 4 tipo de utilizadores associados a um perfil (Memberships). Cada utilizador tem um comportamento diferente no evento (descrito nos requisitos e ilustrado nos diagramas de UML). O evento tem, para além de um utilizador, acompanhantes, prendas, mesas e avisos.

Stakeholders e Utilizadores

As principais partes interessadas (stakeholders) e utilizadores para este Sistema são:

S1: Agências de Eventos

- É do interesse das Agências para que haja um produto que dê uma imagem moderna, inovadora e útil para os seus clientes. Com isso aumentar mais procura e publicidade às agências.

S2\U1: Elementos das Agências de Eventos

- Visa facilitar-lhe as consultas de eventos a decorrer e efectuados.
- São também utilizadores do Sistema.

S3\U2: Lojas

- É para o interesse das Lojas de divulgar e vender os seus produtos.
- Também são utilizadores do Sistema.

S4\U3: Noivos

- Também é do interesse dos Noivos comprar a utilização do sistema de modo a melhor a organização do seu evento.
- Utilizador principal compra o produto com o interesse de divulgar o seu casamento aos seus convidados e com isso ter uma informação mais organizada.

U4: Convidados

- Estes utilizadores só começam a usar o Sistema a partir do momento em que os Noivos os convidam.

Cálculo do Peso de cada Stakeholders

	S1	S2	S3	S4
S1	X	↗	↗	↗
S2		X	↘	↘
S3			X	↘
S4				X

Tabela 1: peso stakeholders

Como se vê na tabela, quanto aos Stakeholders, o maior interesse para o Sistema são as Agências de Evento e os Noivos, de seguida as Lojas e por fim os elementos das agências de evento.

ANÁLISE DE REQUISITOS

Com o objectivo de limitar as fronteiras e esclarecer todas as funcionalidades separei os requisitos a serem implementados no Sistema em requisitos funcionais, as funções que o sistema deve suportar e em requisitos não-funcionais, tais como o desempenho, usabilidade, confiabilidade, segurança, disponibilidade, manutenção e tecnologia, ou seja, requisitos que o cliente não precisa de saber.

Neste capítulo descrevo os requisitos mais importantes, estando todos os requisitos definidos descritos em anexo (analiseRequistos.PDF).

Também em anexo estão os esboços da Interface para uma melhor percepção dos requisitos e da implementação.

Por fim, no capítulo DESENHO – UML estão definidos as condições, acções e comportamentos do sistema.

Requisitos Funcionais

a) Funcionalidades do Administrador

RF01: O Administrador cria Evento e Regista os Noivos

- ✓ Inserir dados de autenticação (username [único]; password [length > 5]).
- ✓ Inserir dados dos noivos de um dos noivos (Nome; Email; Telefone; Morada).
- ✓ Inserir dados do evento (data de realização do evento).

b) Funcionalidades dos Noivos

RF03: Gerir Convidados

Regras: O sistema deverá alertar aos noivos quando os noivos tentarem registar um convidado já existente.

- ✓ Inserir dados de autenticação para os Convidados (username [único]; password [length > 5]; nome do convidado).
- ✓ Listagem de todos os convidados com os seguintes detalhes (username, password [encriptada]; nome; Data da última actividade dos convidados no Sistema).
- ✓ Editar convidados (alterar passwords; alterar nome).
- ✓ Apagar convidados.

RF04: Gerir Acompanhantes

Regras: O sistema deverá alertar aos noivos quando os noivos tentarem registar um acompanhante já existente. A inserção de um acompanhante só terá sucesso quando associado a um convidado já inserido anteriormente, por isso o Sistema deverá apresentar uma lista dos convidados já registados. Caso já existem mesas inseridas, os noivos podem também associar o acompanhante às mesas correspondentes, não sendo obrigatório.

- ✓ Inserir dados dos acompanhantes (disponibilizar lista dos convidados a associar aos acompanhantes; nome completo do acompanhante; confirmação [vou || não vou || por autorizar || não autorizado]; lista de mesas disponíveis).
- ✓ Listagem de todos os acompanhantes com os seguintes detalhes (convidado associado; nome completo do acompanhante; confirmação; mesa; informação se foi inserido pelos noivos ou convidado).
- ✓ Editar acompanhantes (todos os campos são editáveis).
- ✓ Apagar acompanhantes.

RF10: Sumário do Evento

- ✓ Apresentar um resumo com todos os dados fundamentais para os Noivos (Convidado; prendas por convidados; os acompanhantes por convidado; mesa por cada acompanhante; confirmação por cada acompanhante, total de convidados e confirmações).
- ✓ Possibilidade de exportar em PDF.

Acompanhamento 3	VOU	MESA 2
Convidado 2 - Prenda4		
Acompanhamento 4	VOU	MESA 3
Acompanhamento 5	NÃO VOU	MESA 3
Convidado 3 - Prenda5, Prenda6		
Acompanhamento 6	VOU	MESA 4
Acompanhamento 7	NÃO VOU	MESA 4
Acompanhamento 8	VOU	MESA 4
Acompanhamento 9	VOU	MESA 4
Convidados: 9 - Confirmados: 6		

Ilustração 2: representação sumária dos convidados

c) Funcionalidades dos Convidados**RF11: Minhas Confirmações**

- ✓ Apresentar todos os acompanhantes do convidado, inclusivamente, o próprio convidado. *Nota: O convidado é só uma autorização de se poder autenticar no Sistema e por isso é também considerado como um acompanhante.*
- ✓ Possibilidade de confirmar cada acompanhante durante o prazo (VOU || NÃO VOU)
- ✓ Possibilidade de o convidado adicionar outros acompanhantes que queira levar consigo, sob autorização posterior dos Noivos.
- ✓ Apresentar todas as prendas que o convidado vai oferecer aos noivos.
- ✓ Possibilidade de o convidado adicionar/apagar novas prendas pessoais que vão oferecer aos noivos. (Só como referência para os noivos e para mais tarde saber a quem agradecer).

RF12: Lista de Prendas

Descrição: A lista de prendas são as prendas em que os Noivos inserem no sistema para os convidados depois terem acesso e poderem reservar as prendas que querem oferecer. Assim, evita que não haja oferta de prendas em duplicado e também para os noivos ficam com o registo de quem reservou a prenda para mais tarde os agradecer.

Os lojistas também podem e devem inserir prendas no sistema. Obviamente, essas prendas serão escolhidas pessoalmente pelos Noivos na loja correspondente. Eventualmente, os lojistas podem por outras prendas com descontos e promoções para ganharem alguma publicidade.

Regras: O sistema deverá só permitir aos convidados a reserva de prendas até ao dia do evento. O sistema deverá apresentar só as prendas inseridas pelos noivos e lojistas. O sistema deverá alertar ao convidado quando a reserva não for realizada com sucesso.

- ✓ Apresentar a lista de todas as prendas inseridas pelos Lojistas e pelos Noivos com a possibilidade de reservar as prendas disponíveis.
- ✓ A reserva será um processo sem retorno e por isso deverá haver um aviso aos convidados se realmente quer fazer a reserva.



Nome Oferta	Loja	Acções
TV	FNAC	RESERVADO
Faqueiro	Vista Alegre	RESERVADO
Fogão	WORTEN	RESERVAR

Ilustração 3: apresentação da lista de prendas

d) Funcionalidades dos Lojistas

RF13: Inserir Prendas pelos Lojistas

Regras: O sistema deverá só apresentar ao lojista as prendas da loja correspondente.

- ✓ Inserir dados das Prendas (nome da prenda; valor; outros detalhes da prenda).
- ✓ Listagem de todas as prendas no evento com os seguintes detalhes (nome prenda; valor; loja; morada da loja; detalhes da prenda; estado [livre || reservado]; quem reservou).
- ✓ Editar prendas (alterar nome; alterar morada, anular uma reserva).
- ✓ Apagar prendas inseridas.

e) Funcionalidades Comuns

RF14: Página Inicial

- ✓ Apresentar em destaque 3 imagens dos noivos do evento correspondente.
- ✓ Apresentar em destaque o tempo que falta para o casamento.
- ✓ Apresentar toda informação sobre o casamento.

RF16: Avisos

Regras: O sistema deverá apresentar, aos utilizadores, em destaque no menu o número de avisos por ler.

- ✓ Apresentar em destaque no menu o número de avisos por ler.
- ✓ Apresentar todos os avisos do utilizador.
- ✓ Possibilidade de marcar como lido ou apagar os avisos que recebeu.

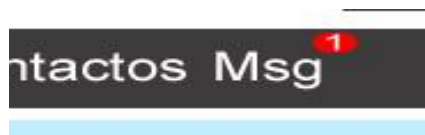


Ilustração 4: número de avisos em destaque

Requisitos Não-Funcionais

RNF01: Aplicação WEB

- ✓ Acesso através de um PC ou SmartPhone com browser WEB em ambiente internet.

RNF02: Facilidade

- ✓ O sistema deve ser bastante simples e perceptível na utilização para todos os utilizadores, principalmente para os convidados.

RNF03: Segurança

- ✓ Aproveitar a ferramenta (Membership do asp.net) para a criação de vários níveis de permissões para cada utilizador (perfis) e autenticações (administradores, noivos, convidados, lojistas).

RNF04: Confidencialidade dos dados

- ✓ Garantir que os administradores não tenham acesso à informação dos convidados, lojistas, mesas, prendas e avisos dos noivos.
- ✓ Garantir que as prendas pessoais que os convidados inseriram sejam só vista pelos noivos.

RNF05: Desempenho e Disponibilidade

- ✓ O sistema deve estar disponível 24 horas.

RNF06: Capacidade

- ✓ O sistema deve permitir bastante capacidade de volume de dados, nomeadamente para armazenar as fotos de capa dos noivos e os dados das centenas de convidados e prendas num evento.

RNF07: Implementação

- ✓ O sistema deve ser implementado em ASP.NET com a linguagem C# para a aplicação WEB, DAL (data access layer) e Web-Services, em JAVA para Android e deve se comunicar com base-de-dados usando SQL Server

Factores Críticos

É importante encontrar os factores críticos pois são os pontos-chave que definem o sucesso ou o fracasso do desenvolvimento deste sistema.

FCS01: Eficiente na apresentação dos dados

É importante que o sistema permite mostrar claramente, segundo os critérios definidos pelos requisitos, todos os dados para que os utilizadores tenham uma boa informação detalhada.

FCS02: Eficiente na introdução de dados

O sistema tem que permitir uma introdução de dados (utilizadores, acompanhantes, mesas, prendas) simples e eficaz para que a sua divulgação seja correcta e clara para cada utilizador.

FCS03: Usabilidade

O sistema com uma boa usabilidade permite uma mais-valia para as agências de eventos e o sistema. Permitindo uma fácil e perceptível uso do sistema para os vários tipos de utilizadores.

FCS04: Reutilização de informação

Para que não se torne repetível os processos, é importante que o sistema perceba que informações podem ser reutilizadas ou que informações já foram introduzidas. Como por exemplo, quando os lojistas inserem as prendas não precisam de estarem sempre a introduzir a morada e nome da loja. Quando os noivos adicionam acompanhantes aos convidados o sistema percebe se esse utilizador já existe para que não haja duplicação de dados.

FCS05: Redução tempo de tarefas

O sistema tem que compensar “bastante” na confirmação dos convidados, na alocação dos convidados nas mesas, nas prendas que cada convidado reservou em comparação ao modo tradicional (sem haver qualquer sistema). Ou seja, basta os noivos irem ao sistema e obter todos esses dados.

FCS06: Estabilidade

É importante que o sistema esteja preparado para um grande volume de convidados, prendas, mesas e lojistas e que a base-de-dados aguente “à vontade” e correctamente todos os dados.

FCS07: Confiabilidade

Como no FCS anterior é importante que o sistema não falhe quando os utilizadores introduzem os dados ou quando os noivos consultem todas as informações desejadas.

FCS08: Segurança e Confidencialidade

O sistema tem que estar protegido de modo que os dados não estejam disponíveis a “terceiros”.

FCS09: Concorrência na reserva de prendas

O sistema tem que estar também protegido quando os convidados fazem reservas de prendas em simultâneo.

FCS10: Detalhes dos Convidados em formato PDF

O sistema tem a capacidade de exportar em PDF o sumário de todos os detalhes por cada convidado para que os noivos possam guardar no seu PC ou imprimir para poderem organizar o seu evento.

Cálculo dos factores críticos por Stakeholder

Cálculo dos factores críticos						Escala	
FCS	Stakeholders				Final	Não intervêm	0
#	S1	S2	S3	S4	100%	Consulta	1
	40%	8%	18%	34%		Participa	2
						Muito	4
						Crítico	8
1	8	4	8	8	1,92		
2	8	4	8	8	1,92		
3	8	2	4	8	1,7		
4	8	1	8	4	1,52		
5	8	2	2	8	1,61		
6	8	2	8	8	1,88		
7	8	4	4	8	1,74		
8	8	4	8	8	1,92		
9	8	0	2	4	1,23		
10	8	0	0	4	1,14		

Tabela 2: cálculo da importância dos factores críticos de sucesso

Através dos cálculos nas tabelas anteriores atribuí um peso ponderando por cada Stakeholder calculado na tabela, *tabela 1: peso stakeholders*, e dei valores de importância a cada factor crítico, *escala*. Com esses cálculos cheguei à conclusão quais os factores críticos de sucesso mais importantes e que merecem mais atenção e cuidados.

O FCS 1, 2, 8 e 6 são os FCS mais importantes que correspondem claramente essencialmente ao objectivo do Sistema.

Processos e Tarefas

É necessário identificar adequadamente as tarefas e descrever os respectivos procedimentos a fim de evitar perda de tempo, repetição de tarefas e diminuição da produtividade.

Os processos são actividades previamente estabelecidas e que tem como objectivo determinar como o trabalho será realizado neste sistema.

Os principais processos que identificados são:

✓ **Processo Criar Novo Evento (admin)**

Este processo é da responsabilidade da agência de eventos em que os noivos se dirigem a uma agência de eventos com o objectivo de adquirirem um pacote de compra para o seu casamento. Os elementos da agência (administrador), no sistema faz o seu login e através de um formulário na WEB solicitam os dados aos noivos (já definidos nos requisitos funcionais) e regista um novo evento. Neste processo os noivos ficam autorizados (login e password) para utilizarem o sistema através na WEB e Android. Neste processo os elementos da agência podem em qualquer momento, se justificar, alterar e/ou apagar dados.

Os processos subdividem-se em:

- Login do admin
- Preenchimento dos dados dos noivos e do evento
- Alterar Evento
- Apagar Evento
- Atribuição de login aos noivos

✓ **Processo Gerir Meu Evento (noivos)**

Este processo é dirigido para os noivos em que eles podem gerir várias funcionalidades sobre o seu evento através na WEB e algumas no Android. Os noivos têm as várias tarefas, tais como, criação/alteração e consulta de dados sobre o seu evento, tais como, convidados, acompanhantes, lojistas, prendas, mesas, avisos, meu evento.

Os processos subdividem-se em:

- Criar “logins” para os convidados e lojistas
- Inserir/editar acompanhantes, mesas, prendas
- Enviar avisos
- Editar meu evento
- Consultar todos os dados relacionados com os convidados.

✓ **Processo Confirmações e Reservas do Convidado**

Este processo é dirigido para os convidados em que eles podem confirmar os acompanhantes e inserir novos acompanhantes, reservar prendas e inserir novas prendas pessoais, ler e apagar avisos e ver informações sobre evento.

Os processos subdividem-se em:

- Confirmar acompanhantes
- Inserir novos acompanhantes
- Reservar Prendas
- Inserir novas Prendas Pessoais

✓ **Processo Inserir Prendas do Lojista**

Este processo é dirigido para os Lojistas em que eles só inserem prendas. Prendas essas escolhidas pelos noivos e disponibilizadas como lista de prendas para todos os convidados poderem reservar.

Os processos subdividem-se em:

- Inserir Prendas
- Editar Prendas
- Apagar Prendas
- Consultar Prendas a Loja

DESENHO

Arquitectura

Na elaboração da arquitectura do Sistema YAI a desenvolver, procurei adaptar a utilização de um estilo “por camadas”, mais especificamente o modelo de 3 camadas e assim representar os 3 grandes blocos que constitui o Sistema: Camada de Dados, Camada de Negócio e Camada de Apresentação.

Os artefactos do sistema estão descritos detalhadamente nos diagramas de E/R, UML e WEB-Service Android.

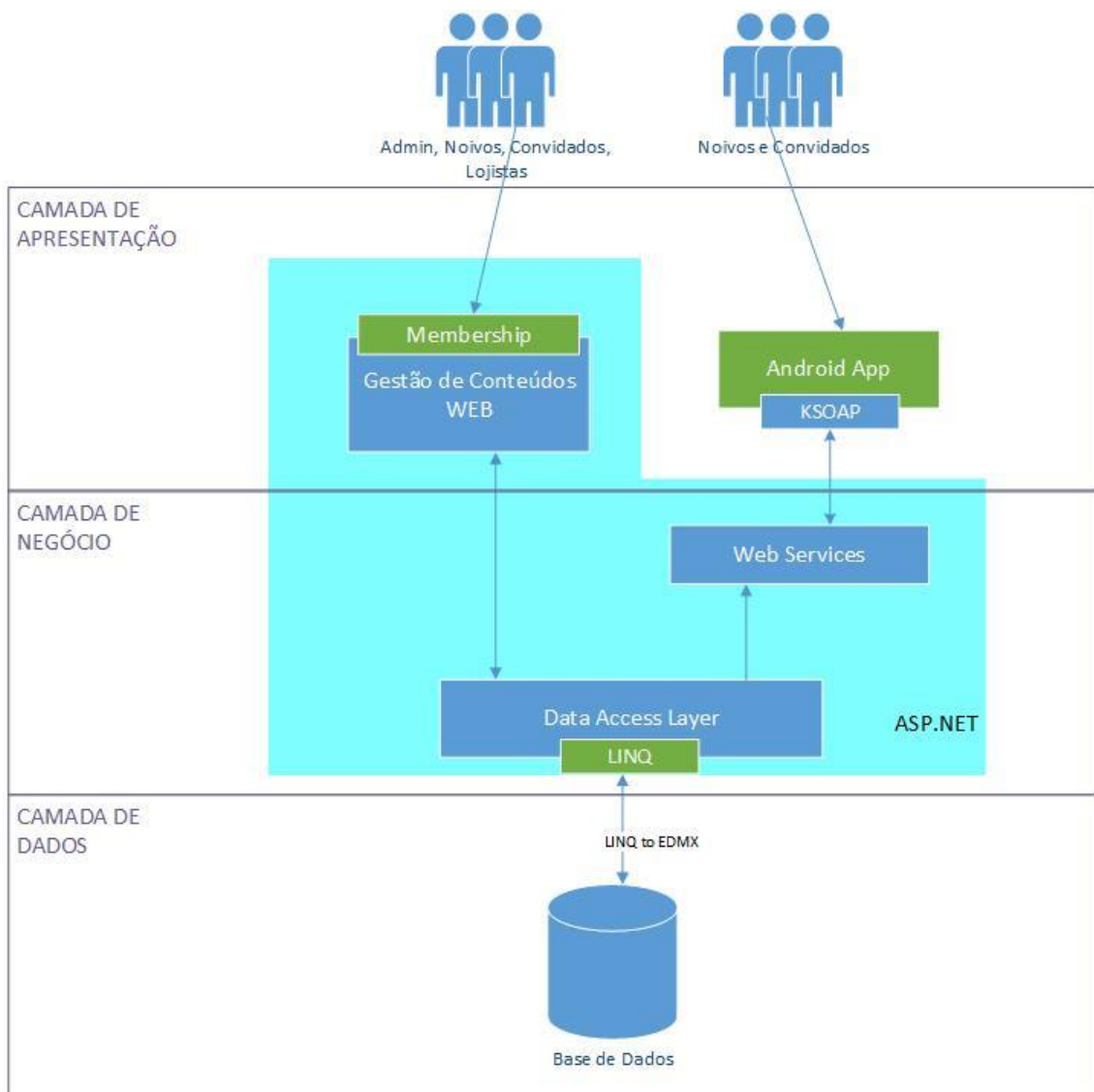


Ilustração 5: arquitectura do sistema YAI

Uma das razões que decidi implementar o Sistema YAI numa arquitectura de 3 camadas foi devido a Heterogeneidade do Sistema. Pois o Sistema está dividido em diferentes plataformas, componentes e linguagens.

No lado do cliente, a aplicação deverá funcionar através da utilização de um browser ou telemóvel Android, sem recurso a quaisquer especificações adicionais de natureza física e/ou lógico.

No lado do servidor, o sistema é suportado por 3 camadas tornando assim o sistema YAI mais escalável e independente.

- **Camada de Apresentação** é uma interface gráfica que interage directamente com o utilizador (GUI). Esta camada tem implementado duas interfaces distintas:
 1. Direccionado para um Browser no PC.
É implementado através da plataforma ASP.Net com a linguagem C# e que vai gerar código HTML para ser mostrado ao cliente.
 2. Direccionado para um Telemóvel Android.
É implementado com a linguagem JAVA (java para Android) para a construção de aplicações e possibilitar ao cliente instalar a aplicação no seu telemóvel e aceder ao SITE do Sistema YAI.
- **Camada de Negócio** é a camada lógica do Sistema e é implementa em ASP.Net na linguagem C#. É nesta camada que se define todas as regras de funcionamento de modo a satisfazer os requisitos e satisfazer os clientes. Nesta camada estão implementados duas componentes:
 1. DAL (Data Access Layer), responsável pela comunicação entre a Camada de Dados e a Camada de Apresentação e detalha todas as funcionalidades do sistema.
 2. Web-Services, disponibiliza todos os “serviços” do DAL para a camada de apresentação, mais especificamente, para a aplicação Android. Só assim é que é possível a aplicação Android se comunicar com o Sistema e “dar” todas as funcionalidades definidas ao cliente.
- **Camada de Dados** é definida como repositório das informações do sistema. Esta camada recebe instruções da camada de negócio e os seus métodos executam na base-de-dados (SQL Server). alterando/consultando os dados.

Diagrama E/R

Tirando as vantagens do ADO .NET criei este diagrama E/R (mais detalhadamente em anexo *ModeloDados_ER.pdf*). Com este diagrama pretendo descrever toda estrutura lógica da base de dados. Identificando assim a partir de um conceito do mundo real como os dados serão armazenados.

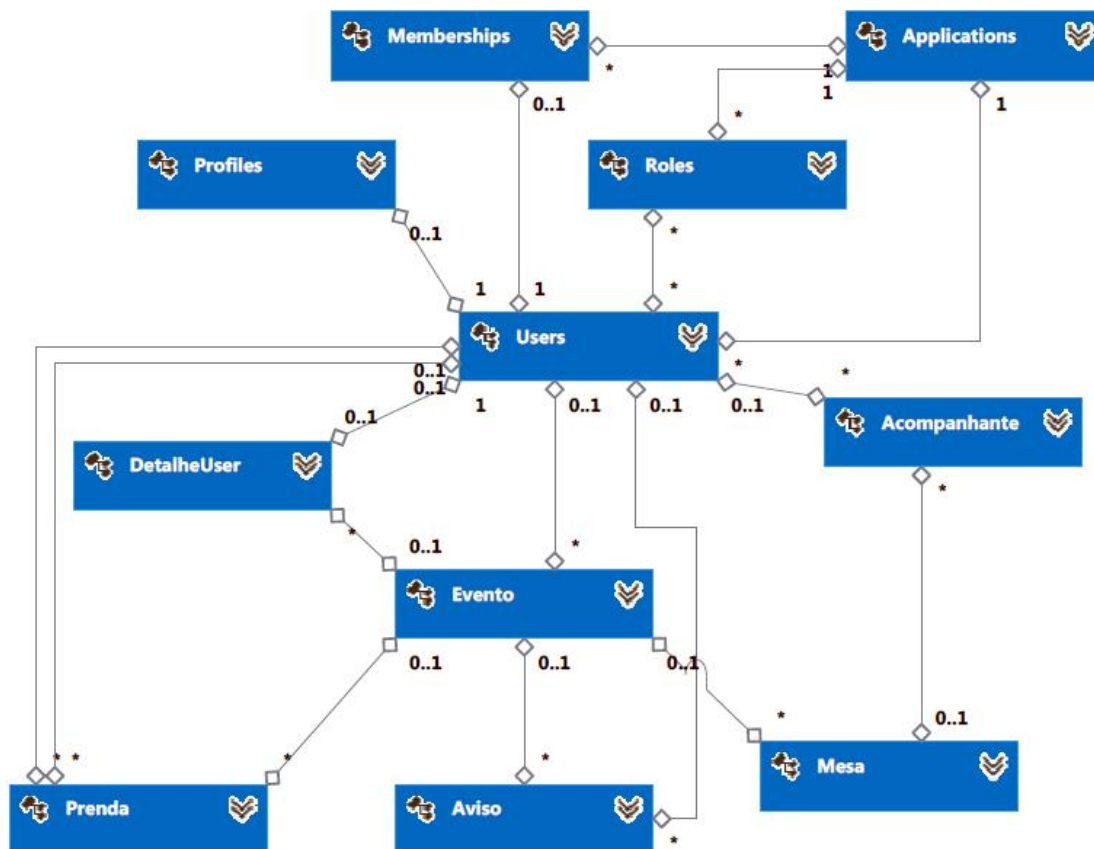


Ilustração 6: diagrama E/R

As tabelas superiores, Profiles, Memberships, Applications, Roles e Users foram criadas automaticamente com os recurso do ASP .NET. Que são essencialmente para a gestão dos utilizadores, nomeadamente, autenticações, autorizações de acessos e perfis.

As restantes tabelas criadas por mim no SQL Server, são todas as entidades e relações essenciais que dão funcionamento ao Sistema YAI.

BPMN

Com o BPMN tentei criar um mecanismo simples para o desenvolvimento dos modelos de processo de negócio, facilitar a compreensão de todas as pessoas envolvidas com o processo e esclarecer melhor a especificação dos requisitos.

Neste capítulo mostro os Processos, Criar Novo Evento dirigido aos elementos da agência de eventos, Gerir Meu Evento aos Noivos, Confirmações e Reservas do Convidado aos Convidados, Inserir Prendas do Lojista.

Os quatro diagramas estão em anexo;

BPMN_processo_ConfirmarReservaConvidado.pdf;

BPMN_processo_CriarNovoEvento.pdf;

BPMN_processo_GerirMeuEvento.pdf;

BPMN_processo_InserirPrendasLojista.pdf;

UML

Para uma melhor descrição de o que fazer, como fazer, quando fazer e porque fiz os diagramas de Casos de Uso, Classes e Sequência. Estes diagramas não demonstram que tipo de trabalho deve ser feito, mas sim em ajudar a obter as visões e aspectos do Sistema YAI.

Diagramas de Casos de Uso

A tabela seguinte encontram-se identificados os principais Actores e casos-de-uso do Sistema YAI.

CASOS DE USO	ACTORES				
	Administrador	Noivos	Convidados	Lojistas	Sistema
Consulta do Site YAI	X	X	X	X	
Criar e Editar Evento	X				
Gerir Evento		X			
Consultar dados do Evento		X			
Criar e Editar Convidados		X			
Adicionar e Editar Acompanhantes		X			
Adicionar Novos Acompanhantes			X		
Confirmar Presenças		X	X		
Adicionar Prendas		X		X	
Adicionar Novas Prendas			X		
Editar Prendas		X		X	
Reservar Prendas		X	X		
Criar e Editar Mesas		X			
Consultar e Apagar Avisos		X	X	X	
Enviar Avisos		X			X

Tabela 3: casos de uso e actores

Diagramas de caso de uso é essencial para representar o comportamento desejado do Sistema YAI independentemente da forma como o Sistema vai ser implementado ou da tecnologia utilizada na construção. Ou seja, dá uma visão geral dos serviços fornecidos pelo sistema e a forma como ele é utilizado pelo cliente. Por isso, é importante conseguir identificar correctamente os actores e os casos de uso do Sistema com base na informação que extraí na fase de levantamento de requisitos e análise.

Apresento dois diagramas de casos de uso dos quatro BPMN analisados, Criar Novo Evento (CU1) e Confirmações e Reservas do Convidado (CU2). Os diagramas encontram-se em anexo.

✓ **CU1 – Criar Novo Evento**

Pré-condições: O administrador tem que estar registado no Sistema.

Pós-condições: Os Noivos ficam registados, recebem um aviso de Boas-Vindas e ficam associados ao seu Evento Novo.

Actor	Use Cases	Extensões	Inclusões
Administrador	Efectuar Login		Validar Login
	Criar Evento		Registar Dados
	Alterar Evento		Registar Dados
	Apagar Evento		Apagar tudo relacionado com o Evento
	Consultar Evento	Se data evento > data actual → Alerta	
Sistema YAI	Validar Login		
	Registar Dados		
	Cria Autenticação para Noivos	Enviar aviso de Boas-Vindas aos Noivos	
	Apagar tudo relacionado com o Evento		

Tabela 4: CU1 - identificação dos use cases por actor

NOTA: Diagrama em anexo (UML_CasoUso_CriarEvento.pdf)

✓ **CU2 – Confirmações e Reservas do Convidado**

Pré-condições: O convidado tem que estar registado. Tem que haver acompanhantes já inseridos associados ao convidado. E existir prendas. A data do evento não pode ter já ultrapassado da data actual.

Pós-condições: Os acompanhantes ficam com a confirmação de presença confirmada. As prendas reservadas ficam associadas ao convidado.

Actor	Use Cases	Extensões	Inclusões
Convidado	Efectuar Login		Validar Login
	Confirma Presenças	Se data evento < data actual – 30 dias → Envia Aviso Noivos	Registar
	Adiciona Novo Acompanhante	Se data evento < data actual – 30 dias → Envia Aviso Noivos	Registar
	Consulta Lista de Prendas		
	Reserva Prendas	Se data evento < data actual → Envia Aviso Noivos e/ou Lojista	Registar
	Adiciona Nova Prenda Pessoal		Registar
Sistema YAI	Validar Login		
	Valida Prenda Reservada		
	Registar Dados		
	Envia avisos para Noivos		
Noivos	Recebe Avisos		
	Autoriza Novo Acompanhante		Envia Aviso

Tabela 5: CU2 - identificação dos use cases por actor

NOTA: Diagrama em anexo (UML_CasoUso_ConfirmacoesReservas.pdf)

Diagrama de Classes

Um diagrama de classes consiste essencialmente num conjunto de classes, caracterizadas por atributos e operações e no seu relacionamento.

É utilizado para representar várias perspectivas do sistema e tem uma relevância fundamental para a modelação estática do sistema como também para a construção posterior de modelos dinâmicos.

Com base do resultado da análise de requisitos criei dois diagramas de classes. O Diagrama de Classes UML (C1) e o diagrama de classes gerado por Entity Framework (C2).

Quanto ao C2, mais uma vez, aproveitei as vantagens do ADO .NET para gerar este diagrama de classes que poderá vê-lo em anexo (*Entity_Classes_SistemaYAI.pdf*). E no Visual Studio poderá ainda ver com mais detalhe as classes com os seus atributos e tipos.

O C1 está em anexo (*UML_Classes_SistemaYAI.pdf*).

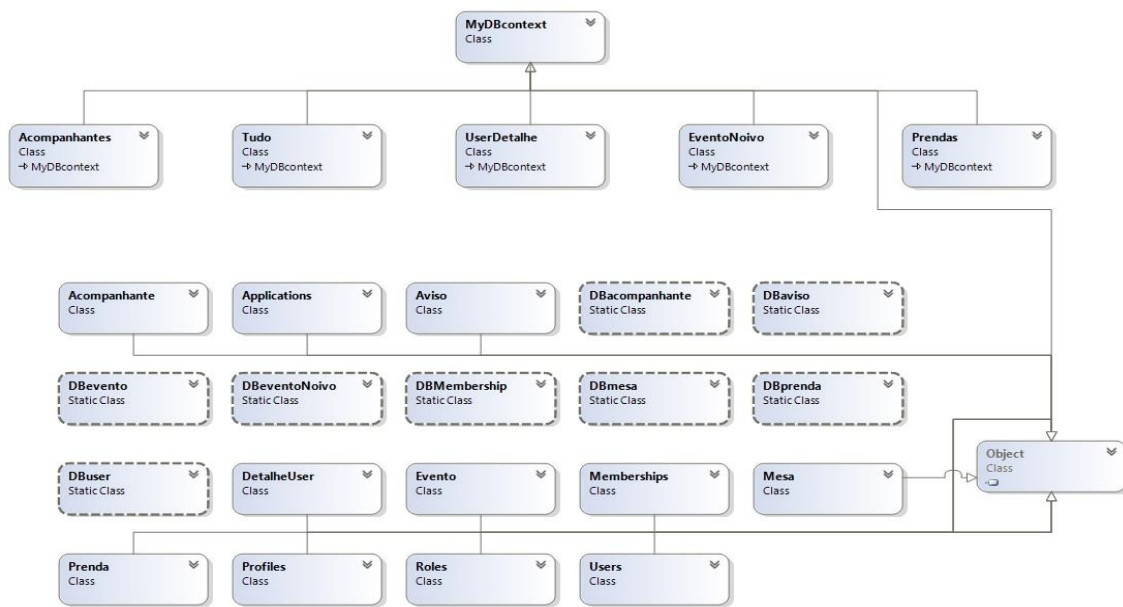


Ilustração 7: C2 - diagrama de classes gerado pelo Entity Framework

Diagramas de Sequência

Estes diagramas servem para representar o comportamento dinâmico do Sistema, ou seja, as interações entre objectos. Com estes diagramas vou especificar os Casos de Uso que desenhei anteriormente.

O diagrama consiste na representação de sequência de chamada de métodos ou troca de mensagens, entre objectos das classes que suportam a aplicação.

Ao fazer estes diagramas estou a tomar decisões ao nível da implementação que dependem da arquitectura e da linguagem de programação a ser utilizada.

Nas páginas seguintes vou apresentar dois diagramas correspondentes a dois processos analisados, Criar Novo Evento (S1) e Confirmações e Reservas do Convidado (S2). Os diagramas encontram-se em anexo.

NOTA: Diagrama em anexo (*UML_Sequencia_CriarEvento.pdf*)

NOTA: Diagrama em anexo (*UML_Sequencia_ConfirmacoesReservas.pdf*)








FRAMEWORKS – FERRAMENTAS UTILIZADAS

Para a construção deste Sistema YAI foram utilizadas várias ferramentas. Dentro destas ferramentas há algumas que merecem mais destaque e da qual irei falar mais das suas vantagens para o desenvolvimento do Sistema.

Microsoft Visual Studio

É um programa da Microsoft que permite desenvolver software dedicado ao .NET Framework e às várias linguagens (C, C++, C#). A vantagem que tiro deste Software é na área de desenvolvimento o que me permite usar várias ferramentas, para a criação de páginas dinâmicas.

As tecnologias que tirei partido do Visual Studio para o desenvolvimento do Sistema YAI são:

-  ASP.NET
-  ADO.NET
-  Entity Framework
-  C#
-  Membership
-  LINQ
-  WEB-Services – WCF

ASP .NET, uma plataforma onde tenho todos os recursos para o desenvolvimento de aplicações WEB. Algumas vantagens e utilidades falei no capítulo IMPLEMENTAÇÃO. Outras vantagens que o ASP .NET tem é na rapidez na compilação em que as páginas são compiladas em um DLL o que torna a 25% mais rápido do que outras plataformas. A possibilidade de usar várias linguagens de programação, tais como C, C++, C#. A separação de uma página em dois arquivos, o layout e o outro em código (Code Behind) o que me possibilita fazer alterações em paralelo no código sem prejudicar os layouts. A disponibilidade de vários controlos o que facilita a comunicação e a publicação com as bases-de-dados e tornar o código mais limpo. Possibilidade de fazer Debug com vários pontos de interrupções e assim perceber melhor o comportamento do código e corrigir erros. Poder usar componentes externas, tais como classes, o que me permite enriquecer e aumentar funcionalidades.

ADO .NET, são classes que integra a plataforma .NET e é utilizada para aceder aos dados numa base-de-dados. O ADO .NET oferece também várias vantagens para esta comunicação com a base-de-dados, tais como, a interoperabilidade através do padrão XML permitindo assim a comunicação de forma transparente com outros sistemas semelhantes ou não. Alto desempenho, neste caso, o acesso de forma desconectado, o que me permite a manipulação de dados de forma mais rápida sem ter que estar ligado a uma base-de-dados. A produtividade, em que tenho um modelo fácil compreensão e que me permite o desenvolvimento rápido de camada para acesso de dados.

Entre as ferramentas disponíveis no ADO .NET o **Entity Framework** foi a ferramenta que utilizei neste trabalho. Segundo a Microsoft, Entity Framework é um conjunto de tecnologias ADO .NET que suporta o desenvolvimento de aplicações orientadas a dados, ou seja, é a maneira como são armazenados os dados de Entidades, atributos e relacionamentos. Com esta ferramenta, primeiro, criei um modelo de entidade-relação o que vai definir as entidades contidas dentro da aplicação. Em anexo, *Modelo Dados ER*, mostro o modelo construído do Sistema YAI. O que é uma grande vantagem desta ferramenta da criação deste modelo em que ele é gerado automaticamente pela minha base-de-dados já existente.

LINQ, uma componente da Microsoft .NET que me adiciona funcionalidades de consultas com base-de-dados em linguagem C#. As vantagens são; a padronização do código o que permite que todos possam entender o código e tornar o código muito mais leve e limpo em que várias instruções possam ser feitas só numa linha de código, como por exemplo expressões Lambda. A possibilidade de relacionar com qualquer fonte de base-de-dados de gerar extensões de classes e usar Stored Procedures.

Membership, um recurso do ASP .NET, o que facilita bastante a criação e gestão de utilizadores e inseri-los em grupos com determinadas permissões, como já falei no capítulo IMPLEMENTAÇÃO. Com este recurso, posso usar controlos do ASP .NET ou programar a gestão dos utilizadores e autenticação. Os dados são gerados automaticamente e criados numa base-de-dados (SQL Server).

Classes Externas


Bootstrap é um framework (front-end e opensource) para desenvolvimento utilizado para criar aplicações WEB para computadores pessoais e móveis. Utiliza HTML(5), CSS, JavaScripts e plugins do JQuery e tem vários componentes, tais como, botões,

campos de entrada, labels, listas, carrossel, tabelas, grids etc. E a estrutura dos arquivos do Bootstrap é organizada por três componentes importantes, CSS, JS e fonts.


A sua utilização tem várias vantagens para o Site YAI, como por exemplo; Layouts Responsivos o que permite uma óptima experiência de visualização, fácil leitura, navegação e adaptável a todo o tipo de monitores (computadores e telemóveis). Vários componentes de edição fácil o que permite adaptarem de forma eficaz ao site YAI. Por fim, funciona em todos os navegadores.

Gridviewhelper é uma class (de *Agrinei Sousa*) que aumenta as funcionalidades de uma Gridview do ASP .NET, tais como, a criação de totais e agrupamentos como se pode ver na *ilustração 10*.

Nesta classe uso duas funções:

 ***helper.RegisterSummary("Convidados", SummaryOperation.Sum);***

Apenas com esta função crio um total (uma linha novo no footer) para a coluna dos Convidados. Esse total que vai ser o número total dos convidados que existe no evento.

 ***helper.RegisterGroup("Convidados", true, true);***

Com esta função consigo agrupar todos os acompanhantes correspondente a um convidado. É criada uma nova linha do convidado agrupado com todos os seus acompanhantes.

Como se vê, com esta classe torna bastante fácil com poucas linhas de código agrupar em grupos os convidados e saber os totais dos convidados. A única desvantagem é que não é possível criar sub-grupos, sendo só possível criar outros grupos em forma sequencial.

IMPLEMENTAÇÃO - DETALHES

Android

Esta aplicação é baseada em programação em Java para telemóveis Android usando as tecnologias Android/Java e SOAP (Web-Services) com a biblioteca KSOAP2.

A aplicação é destinada para os Noivos e Convidados e consiste em implementar as funcionalidades mais essenciais para o acesso ao sistema YAI.

- A primeira funcionalidade da aplicação é o Login, em que o utilizador tem que introduzir o mesmo username e password em que entra no Site YAI.
- As funcionalidades da aplicação no Android para os Noivos são de consultas:
 - Consultas das confirmações de todos os Acompanhantes.
 - Consultas do estado das prendas.
 - Consultas dos avisos.
- As funcionalidades para os Convidados são muito semelhantes às dos noivos mas com a possibilidade de confirmar os seus acompanhantes e reservar prendas:
 - Ver os seus acompanhantes e confirmar presenças.
 - Ver a lista de prendas e reservar prendas.
 - Consultas de avisos.

As restantes funcionalidades do sistema estão disponíveis só pelo SITE YAI. Não se justificando na aplicação Android por serem funcionalidades de “organização e gestão” e serem mais adaptadas para serem trabalhadas com um computador (browser).

Todas as funcionalidades da aplicação são disponibilizadas por Web-Services a partir do Sistema YAI. Com o seguinte diagrama facilita a compreensão dos serviços e *Activities* que a aplicação Android tem:

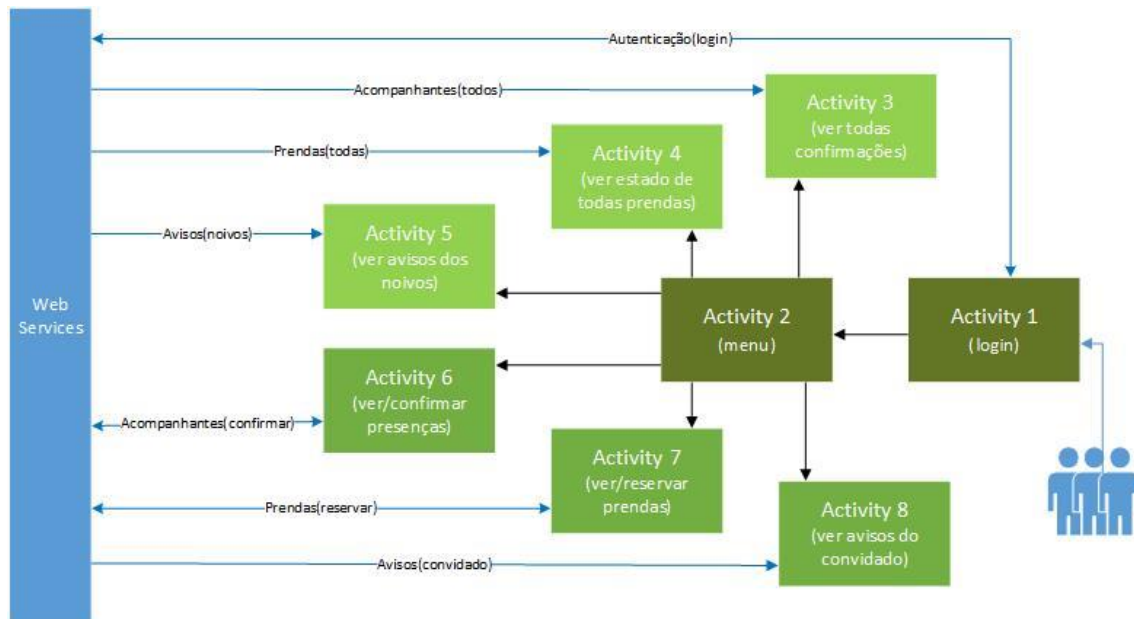


Ilustração 8: activities da aplicação Android

Não se pode deixar de parte em que no arranque do Android há o *AndroidManifest.xml* em que defino as sequências das *Activities*. Também de salientar que os dados (id do user) são transportados entre as activies.

ASP.Net

Durante a implementação do Sistema, um dos requisitos importantes para um bom funcionamento do Sistema é a funcionalidade de Reserva de Prendas pela parte dos convidados ou noivos, a protecção de dados entre vários perfis, apresentação de dados e a introdução de dados.

Para maximizar as funcionalidades do Sistema e atingir os objectivos que foram levantados nos requisitos destaco aqui alguns mecanismos que implementei;

Quanto à **Reserva de Prendas** o sistema tem que ter um mecanismo que proteja ou informe o utilizador quando dois utilizadores reservem a mesma prenda em simultâneo.

Para isso havia várias soluções:

Bloqueio optimista, a actualização não é exclusivo. Bloqueio pessimismo, a actualização é exclusivo.

- Três implementações optimistas que solucionam o problema da concorrência:
 - Datasets – Faz a verificação de valores antigos e novos antes de actualizar.
 - Timestamp – Com um tipo de dados timestamp, durante a actualização é verificado se o timestamp antigo é igual ao novo timestamp.
 - Check old and new value – Retorna os valores, faz as alterações e ao fazer as actualizações finais verifica se o valor antigo e actual são iguais. Se eles não são iguais reverte ou compromete os valores.
- Há várias maneiras possíveis para uma visão pessimista:
 - Shared lock, Exclusive, Update lock, etc...

Como o sistema aceita muitos convidados é possível ocorrer a reserva simultânea da mesma prenda entre convidados diferentes. A ilustração seguinte mostra o código de dois convidados com a possibilidade de reservar a mesma prenda e mostra o código com a uma implementação optimista, Datasets:

The diagram shows two user interfaces, 'Convidado 1' and 'Convidado 2', each with a table for reservation status. Both tables have columns 'Estado' and 'Reservar'. The 'Estado' column has two rows, both with the value 'LIVRE'. The 'Reservar' column has a green button labeled 'Reservar' in each row. Below these is a code snippet for a method named 'ReservarPrenda'.

```
public static bool ReservarPrenda(Prenda prenda, Guid id)
{
    bool res = false;
    try
    {
        if (contexto.Entry(prenda).State == System.Data.Entity.EntityState.Detached)
            prenda = obterCopiaAtualizadaPrenda(prenda);

        if (prenda.Estado == 1)
            return false;

        prenda.Estado = 1;
        prenda.UserReservador = id;

        contexto.Entry(prenda).State = System.Data.Entity.EntityState.Modified;
        res = (contexto.SaveChanges() != 0);
    }
    catch (Exception ex)
    {
        Console.WriteLine(ex.Message);
    }
    return res;
}
```

Ilustração 9: exemplo código reservar uma prenda

O código verifica primeiro na BD qual o estado da prenda. Caso a prenda já esteja reservada (*prenda.Estado == 1*), o código retorna falso, caso contrário actualiza a prenda.

Assim, já é possível de alertar aos outros utilizadores que a prenda “acabou de ser reservada” ficando a reserva realizada com sucesso para o primeiro utilizador.

Quanto à **protecção de dados entre vários perfis**, um dos requisitos importantes, o Sistema usa as Ferramenta Membership e Roles da ASP.Net.

Com esta ferramenta permite facilmente criar e definir regras de autenticação e de acesso como explicado no capítulo “FRAMEWORKS”.

A ilustração ao lado mostra como estão organizados os ficheiros para cada perfil.

Cada perfil tem acesso às páginas que estão na pasta correspondente (Administrador, Convidados, Lojistas, Noivos) garantido assim a protecção de dados de utilizados não autorizados. As outras pastas são de acesso comum a um utilizador registado.

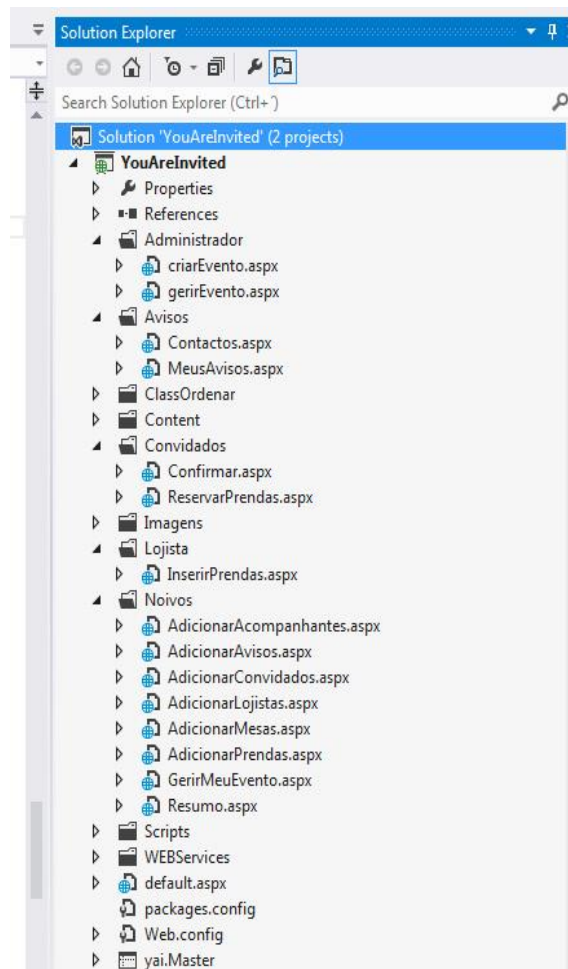


Ilustração 10: organização das páginas do YAI

Outro requisito importante é a **apresentação de dados**, de várias ferramentas usei uma class do autor *Agrinei Sousa* (explicada no capítulo FRAMEWORKS) para simplificar a visualização dos dados para os utilizadores do Site YAI:

Convidados	Confirmações	Mesas
USER: Sr. Conv 1 - PRENDAS: Prenda 1		
Acompanhante 11	Por Confirmar	Mesa não Atribuída
Acompanhante 12	Por Confirmar	Mesa não Atribuída
USER: Sr. Conv 2 - SEM PRENDAS		
Acompanhante 21	Por Confirmar	Mesa não Atribuída
Acompanhante 22	Vou	Mesa não Atribuída
Acompanhante 23	Por Confirmar	Mesa não Atribuída
USER: Sr. Conv 3 - SEM PRENDAS		
Acompanhante 31	Por Confirmar	Mesa não Atribuída
Confirmados: 1 - Total: 6		

Ilustração 11: apresentação dos dados agrupados e totais

Como se vê na ilustração, com esta classe, permite ordenar, organizar os dados por grupo e apresentar totais. Assim torna ao utilizar uma visualização e interpretação dos

dados mais facilmente. *Exemplo: consigo ver claramente os acompanhantes por cada convidado, as prendas por cada convidado, os totais dos convidados.*

Por fim, **a introdução de dados**, especificamente, o upload de imagens pela parte dos noivos, utilizei o método do C#, *GetRandomFileName()*. Este método tem a vantagem de retornar uma string criptograficamente forte, rápida, única e aleatório para a geração de arquivos (.jpg).

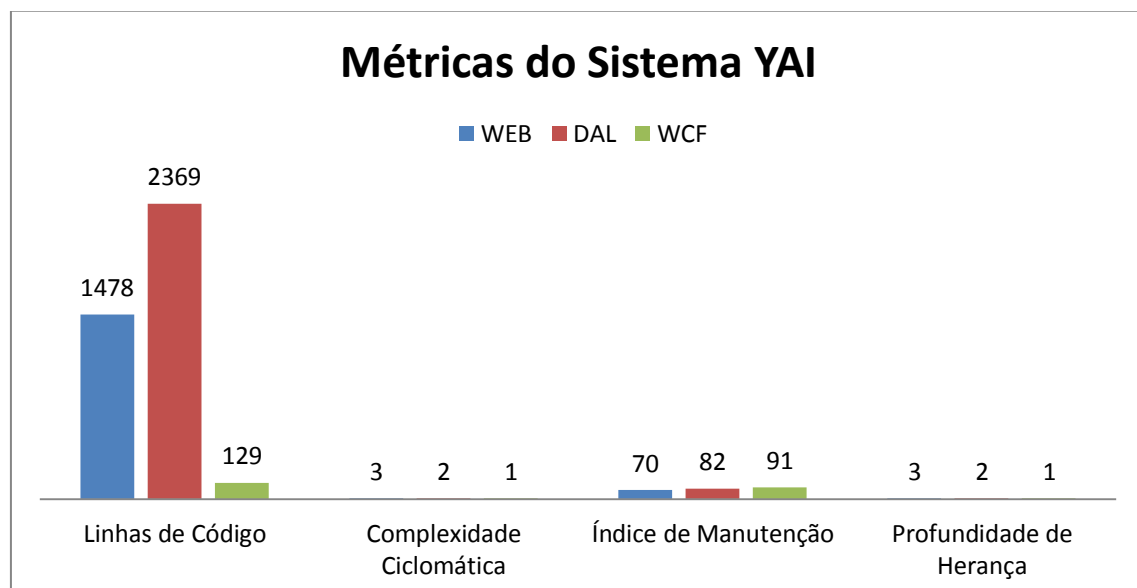
CONCLUSÃO

Fez-se uma implementação completa desde os requisitos até às interfaces, pondo em prática os conhecimentos, análise e concepção de SI, de modelação de BD e programação OO sobre um framework ADO. NET que oferece funcionalidades avançadas para aplicações Web e sobre JAVA para Android o que possibilita fazer aplicações para telemóveis com o Sistema Operativo Android.

Houve oportunidade de lidar com problemas concretos no que concerne à captura de requisitos, ergonomia da Interfaces, os problemas da concorrência e do desempenho em aplicações cliente-servidor.

De seguida, com as ferramentas do Visual Studio, foi possível fazer uma análise métrica quanto à solução desenvolvida. Permitindo assim compreender melhor a complexidade, qualidade e o número de linhas de código que cada classe e função tem o sistema.

De seguida apresento um gráfico com o resumo dos resultados;



Quanto às linhas de código podemos concluir que a Camada de Acesso a Dados apresenta um número maior, pois o Sistema é uma grande parte de consulta, edição e armazenamento dos dados do evento todo, nomeadamente, a gestão de convidados, dos acompanhantes, das prendas e das mesas.

De seguida, verificamos na complexidade ciclomática, que mede a quantidade de caminhos de execução do código, em 1 é o melhor cenário e 8 é considerado crítico.

No Índice de Manutenção, indica a facilidade de manutenção do código. Em que os valores entre 20 e 100 são considerados verde, ou seja, tem uma boa manutenção e poucos pontos problemáticos.

Na profundidade de herança, mede o número de classes que se estendem até à raiz. Ou seja, as classes que são muito profundas são consideradas muito complexas e torna o sistema mais pesado e denso. A média recomendada é inferior a 4.

Com esta análise podemos dizer que o sistema apresenta-se saudável, sem nenhum ponto crítico. Contudo existe um número elevado de linhas de código nas camadas de WEB e DAL o que pode indicar que há procedimentos pesados e que se poderá resolver em dividir esses procedimentos em partes.

Existem também outros aspectos que podem ser melhorados como por exemplo, no que diz respeito ao processo de inserção de convidados, pois por vezes torna-se um trabalho rotineiro quando se trata de muitos convidados. Uma solução possível era os convidados poderem importar uma folha de excel para o sistema e esse faria a inserção automática. Mas este assunto pode ser explorado no futuro, tal como também se pode adicionar novas funcionalidades ao sistema o que trará benefícios para o Sistema, para os Noivos e para os interessados no Sistema.

GLOSSÁRIO

- Sistema YAI** É constituído pelas componentes BD, DAL, Web Services, WEB e Android APP que permite ao Cliente, através da Internet, usar todas as funcionalidades do SITE You Are Invited.
- ASP.NET** Plataforma da Microsoft para desenvolvimento de aplicações Web. É uma componente do IIS (Internet Information Services) que permite através de uma linguagem de programação integrar na .NET Framework criar páginas dinâmicas.
- ADO.NET** Um conjunto de classes integrado na plataforma .NET, que é utilizado para aceder aos dados armazenados numa Base-de-Dados.
- C#** Linguagem de programação orientada a objectos desenvolvida pela Microsoft como parte da plataforma .NET.
- DAL** Camada de Acesso aos Dados é implementada em C# e responsável pela comunicação com a base-de-dados.
- LINQ** Language Integrated Query, componente do Microsoft .Net que adiciona funcionalidades de consultas. Linguagem incorporada à linguagem C# e tem como padrão para comunicar com base-de-dados relacional.
- SQL** Linguagem de acesso e manipulação de dados e implementa conceitos definidos no Modelo Relacional.
- Android** Sistema operacional baseado em Linux para dispositivos móveis liderada pela Google. Permite instalar aplicações feitas em linguagem JAVA.
- JAVA** Linguagem de programação orientada a objectos e é compilada para bytecode e executada por uma máquina virtual.
- Web Services** Integração de sistemas entre aplicações diferentes (YAI → Android) . Com esta tecnologia é possível interagir, disponibilizar serviços e receber/enviar dados em plataformas diferentes na Internet.
- WCF Service** Windows Communication Foundation é uma parte integrada da Framework .NET. Permite utilizar apenas uma framework (unificada) para criar serviços web utilizando diversos protocolos. Ou seja, permite criar o serviço (server side), como também consumir (client side) serviços utilizando diversos protocolos (REST, SOAP, etc...).
- SOAP** Single Object Access Protocol, consiste na invocação remota de

métodos. Estes dados são formatados em XML e enviados em http. Desta forma garante a interoperabilidade e a comunicação do Android com o Sistema YAI.

Activities Interacção de várias janelas com um desenho de interface através de um ecrã de telemóvel com Android.

Modelo de E/R Modelo de Entidade e Relação, é um modelo abstracto que descreve os dados a serem utilizados num sistema.

BPMN Business Process Modeling Notation, é um desenho de processos para que Empresa conheça bem os processos do Sistema com a finalidade de aumentar a qualidade dos serviços, satisfazer os clientes, aumentar a produção e melhorar a qualidade na gestão.

UML Unified Modeling Language, linguagem de modelagem orientada a objectos que visa auxiliar a visualizar o desenho e a comunicação entre objectos.

BIBLIOGRAFIA

- Wikipédia – <http://wikipedia.org>
- Microsoft – Developer Network – <http://msdn.microsoft.com/library>
- Bootstrap - <http://getbootstrap.com>
- GridView de Agrinei Sousa –
http://www.agrinei.com/gridviewhelper/gridviewhelper_en.htm
- Code Project - <http://www.codeproject.com>
- Materiais da Universidade Lusófona - LEI
 1. Acetados do Prof. Luís Alexandre Gomes – Análise e Concepção de Sistemas
 2. Acetados e exercícios da Prof. Inês Oliveira – Sistemas de Informação Multimédia
 3. Exercícios do Prof. Alexandre Pereira – Sistemas Embebidos
- Database Management Systems – Ramakrishnan & Gehrke – McGraw-Hill
- ASP.NET 4.5 – Luís Abreu – FCA
- Professional C# 2012 and .NET 4.5 – Christian Nagel, Bill Evjen, Jay Glynn, Karli Watson, Morgan Skinner – Wrox
- SQL – Luís Damas – FCA

- Exercícios de UML – Henrique O'Neill, Mauro Nunes, Pedro Ramos – FCA
- Fundamental de UML - Henrique O'Neill, Mauro Nunes – FCA
- Android, Introdução ao Desenvolvimento de Aplicações – Ricardo Queirós – FCA

ANEXOS

Devido ao tamanho, qualidade e apresentação, todos os anexos estão também em ficheiros separados no formato do Software original onde foram criados e exportados em PDF.

- Esboços de Interface – Adobe Photoshop e JPEG e PDF
- Arquitectura do Sistema YAI – Visio e PDF
- Web Services para Android – Visio e PDF
- Modelo E/R Detalhado e Resumido – PDF
- Diagramas de BPMN – Visio e PDF
- Diagramas de UML – Visio e PDF
- Análise de Requisitos – PDF
- Projecto ASP.NET do Sistema YAI – Código Fonte
- Projecto JAVA da Aplicação Android – Código Fonte
- Backup da BD do SQL Server