

Vânia Rosaete Fernandes Pereira - 2899



Implementação do Sistema de Gestão de Informação de AAC

RELATÓRIO FINAL DO ESTÁGIO

Instituição do Estágio:

Núcleo Operacional para Sociedade de Informação

Orientador no NOSI: Engenheira Esir Fernandes

Esir.fernandes@nosi.cv

Telf:9979379

Universidade de Cabo Verde

Licenciatura em Tecnologias de Informação e Comunicação.

Setembro/2009

Resumo

O presente trabalho descreve o contexto que envolveu a execução do projecto do estágio curricular intitulado “*Sistema de Gestão de Informação da AAC*” que é um sistema responsável para efectuar o registo de todas as actividades da Agência da Aviação Civil (AAC).

Foi realizado no NOSI (Núcleo Operacional para Sociedade de Informação) durante o ano lectivo de 2008/2009, no âmbito do curso de Licenciatura em Tecnologias de Informação e Comunicação na UNICV.

A concepção do “*Sistema de Gestão de Informação da AAC*” teve por base a plataforma de desenvolvimento assente no Sql Navigator 6.1, plataforma esta que é partilhada por toda equipa de desenvolvimento do NOSI.

É importante dizer que a aplicação “SGI-AAC” foi desenvolvida para resolver um problema real da AAC e a sua eficácia foi já comprovada pelos seus primeiros utilizadores, visto que visto que já houve encontro com os mesmos onde pudemos mostrar as funcionalidades da aplicação.

Agradecimentos

Quero agradecer a minha orientadora de estágio Ezir e colegas do NOSI Joelma e Amândio pelo apoio prestado.

Aos meus colegas do curso de Licenciatura em Tecnologias de Informação e Comunicação em especial Dilson Semedo, Genoveva Teixeira e Anise Mendes, pelas trocas de experiências e por toda ajuda prestada ao longo da realização do presente trabalho.

Finalmente quero também agradecer a todas as pessoas que de uma forma ou de outra contribuíram para a realização do meu trabalho.

Instituição do Estágio:	I
1. Introdução.....	7
1.1 Enquadramento	7
1.2 Organização do Relatório	8
1.3 Apresentação da Instituição de acolhimento.....	8
1.4 Apresentação do Projecto Estágio	9
1.5 Contributos deste trabalho	14
2. Procedimentos Técnicos.....	17
2.1 Modelo de Dados	17
2.1.1 Modelo Entidade - Relacionamento	17
2.1.2 Criação de Tabelas na Base de dados	18
2.2 Interface Web.....	21
2.3 Implementação do SI-AAC.....	29
2.3.1 Análise dos Procedimentos e Funções mais Relevantes	29
3. Conclusão	44
4. Trabalhos futuro	44
Bibliografia.....	46

Índice de Figuras

Figure 1 - Organograma NOSI.....	9
Figura 2 - Integração entre os módulos do projecto.....	10
Figura 3 - Sistema integrado	15
Figura 4 - Modelo de dados do módulo Teste	18
Figura 5 – Tabela AAC_perguntas.....	19
Figura 6 – Tabela AAC_resposta.....	19
Figura 7 – Tabela AAC_requisição	19
Figura 8 – Tabela AAC_aluno_teste.....	20
Figura 9 – Tabela AAC_teste	20
Figura 10 – Tabela AAC_resultado_aluno.....	20
Figura 11 – Pagina inicial da aplicação	21
Figura 12 – Formulário de inserir perguntas.....	22
Figura 13 – Tela de inserção efectuada com sucesso.....	23
Figura 14 – Formulário de inserção de Respostas.....	23
Figura 15 – Formulário de requisição de testes	24
Figura 16 – Lista de categorias de testes	24
Figura 17 - Pagina pesquisa teste	25
Figura 18 – Tela de inscrição de aluno ao teste	25
Figura 19 – Teste com o resultado	26
Figura 20 – Formulário de inserção das respostas escolhidas pelo aluno	27
Figura 21 – Lista dos alunos inscritos num teste	27
Figura 22 - Formulario de elaboração de Teste	28
Figura 23 - Package aac_tipos	30
Figura 24 - Tela de criação de uma View	40
Figura 25 - Report Builder - vista de paper layout	41
Figura 26 - Report Builder - vista Data Modeler	42
Figura 27 - Report de teste com e sem correcção.....	43

Lista de Abreviaturas

NOSI	– Núcleo Operacional para Sociedade de Informação
AAC	– Agencia de Aviação Civil
PL/SQL	– Procedural Language/Structured Query Language
BD	– Base de Dados
SQL	– Structured Query Language
HTML	– Hyper Text Markup Language
PDF	– Portable Data Format
XML	– eXtensible Markup Language - <i>Linguagem de marcação de dados</i>
OJT	– On Job Training
SRS	– Surveillance and Reporting System
DSO	– Divisão de Segurança Operacional
PAN	– Pessoal Aeronautico Nacional
RAN	– Registo Aeronave Nacional

1. Introdução

Este estágio, presente no último ano da Licenciatura de Tecnologias de Informação e Comunicação na Universidade de Cabo Verde, tem como função primordial exercitar todos os conhecimentos adquiridos ao longo do ciclo académico e conseqüentemente, a experiência profissional fundamental para o mundo do trabalho.

Neste relatório está descrito de forma sucinta todo o trabalho desenvolvido ao longo do estágio desenvolvida no Núcleo Operacional para Sociedade de Informação, no departamento Análise e Desenvolvimento de Aplicações Web de 12 de Maio de 2009 a 21 de Agosto de 2009.

É facultada uma perspectiva geral sobre o projecto e são dadas a conhecer sucintamente as principais tarefas executadas durante o estágio.

1.1 Enquadramento

Actualmente as organizações dependem dos seus Sistemas de Informação.

A utilização de Bases de Dados é essencial na tomada de decisão e aquisição de vantagens competitivas por parte das organizações, surgindo como uma necessidade de importância crescente.

A AAC dando valor a essa necessidade, adquiriu uma Base de dados Access para resolver os seus problemas. Devido a grande quantidade de informações geridas pela AAC, a aplicação mostrou-se insuficiente para os desafios diários, visto que essa BD, apresentava algumas deficiências. Não havia integração entre os módulos, não havia segurança dos dados, dados duplicados nas diferentes base de dados, não havia controlo de alterações de informações, etc.

Neste contexto, foi proposto a AAC um sistema aplicacional que integra todos os módulos anteriormente utilizados pela AAC, eliminado o risco de duplicação de informação, erros humanos e aumentando o nível de protecção do sistema informático e mais segurança das informações armazenadas.

1.2 Organização do Relatório

O presente relatório de estágio possui quatro capítulos. No primeiro capítulo é feita a introdução de trabalho sendo igualmente apresentado o enquadramento e os principais objectivos do mesmo.

O segundo capítulo descreve os procedimentos técnicos seguidos para o desenvolvimento da aplicação SGI – AAC.

No terceiro capítulo serão feitas a apresentação dos resultados do trabalho e as respectivas conclusões.

Finalmente no último capítulo deste relatório serão feitas um conjunto de sugestões para trabalhos futuros.

1.3 Apresentação da Instituição de acolhimento

O NOSi é uma Unidade de Execução da Comissão Interministerial para a Inovação e Sociedade de Informação (CIISI). Tem como atribuições e competências a promoção e execução de medidas de política que mobilizem a sociedade, o sector privado e o sector público para o advento da sociedade de informação e a implementação das medidas que visem a modernização da estrutura organizacional da administração pública, rumo à governação electrónica.

O NOSi tem feito um trabalho de integração de sistemas de informação para evitar que o cidadão tenha que se deslocar a várias repartições para obter um único serviço.

A melhoria da qualidade dos serviços prestados passa pela rapidez, redução de custos e, sobretudo, por novos padrões de atendimento. De entre estes principais serviços de integração, implementados pelo NOSi, destaca-se a Casa do Cidadão, que veio introduzir um novo paradigma de relacionamento entre o Estado e os cidadãos.

Organograma da Instituição

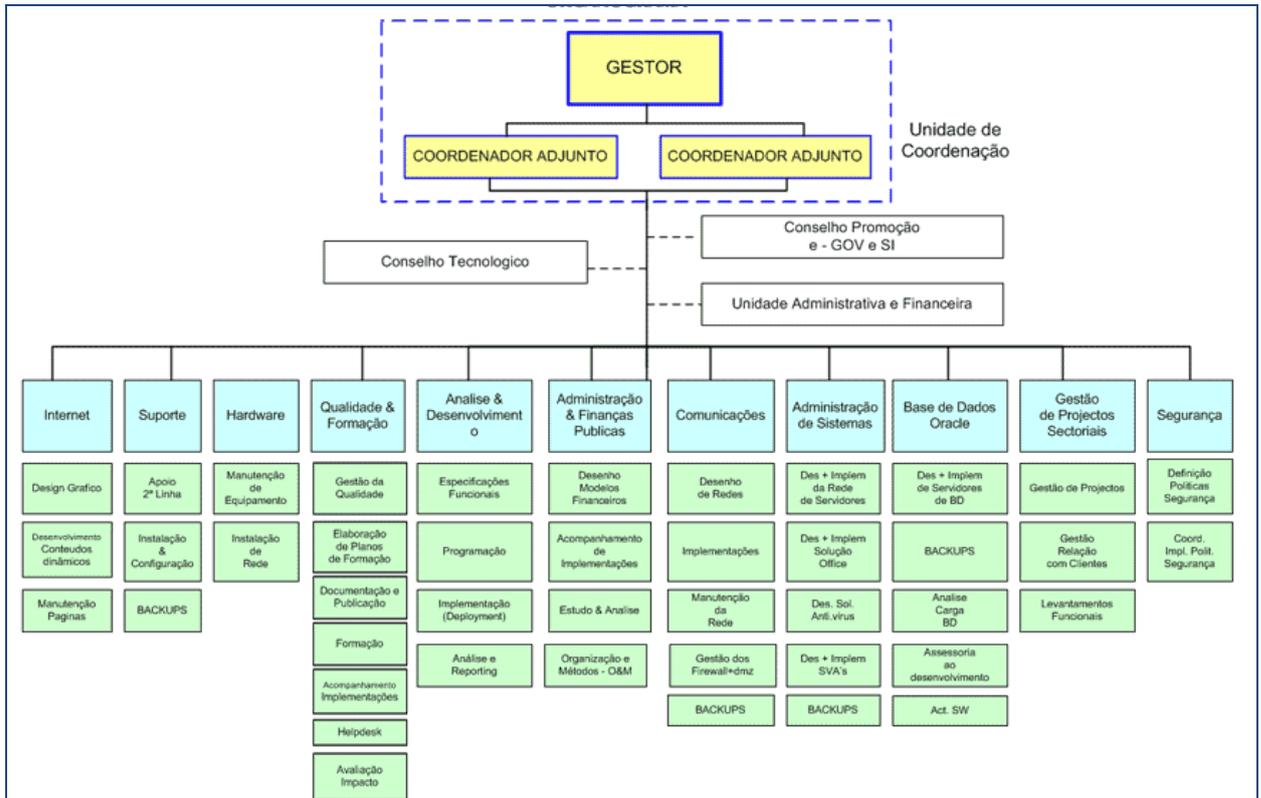


Figure 1 - Organograma NOSI

1.4 Apresentação do Projecto Estágio

A AAC é uma agência reguladora que desempenha actividades administrativas de regulação técnica e económica, supervisão e regulamentação do sector da aviação civil.

Um dos principais objectivos da AAC é a estruturação e a consolidação do sector da aviação civil no que diz respeito à segurança das actividades da aviação civil e à promoção da eficiência e regularidade do transporte aéreo, factores importantes para o desenvolvimento económico e social de Cabo Verde.

Atingir os objectivos acima referidos requer que a AAC tenha a seu dispor, meios e tecnologias que lhes permitam não só gerir o seu negócio a nível interno como também a nível externo.

Para ajudar nessa tarefa, é do interesse da AAC o desenvolvimento de uma BD que lhe permita fazer essa gestão.

O sistema desenvolvido deve permitir a AAC efectuar o registo de todas as suas actividades diárias de gestão. Actividades do tipo:

- Gestão de Aeronave;
- Gestão do pessoal aeronáutico;
- Gestão de testes;
- Etc;

Todas essas actividades devem ser geridas de uma forma segura e confiável, introduzindo mecanismos de seguranças de acesso ao sistema.

Módulos do projecto

A aplicação está SGI-AAC está dividida em vários módulos, como mostra a figura abaixo. Esses módulos estão todos integrados entre si e são apresentados de seguida.



Figura 2 - Integração entre os módulos do projecto

Módulo SRS: A AAC executa actividades de Regulamentação, de Certificação, de Licenciamento e de Supervisão. Este módulo permite a gestão das actividades planeadas e outras actividades que tenham sido executadas por diferentes departamentos da AAC, assim como a gestão das não conformidades (findings) encontradas durante as certificações e supervisões.

O presente módulo permite ainda a emissão de relatórios básicos sobre as actividades registadas.

Módulo PAN: Este módulo permite a gestão de todo o Pessoal Aeronáutico Nacional, nomeadamente o registo dos dados pessoais e de qualificação e licenciamento do pessoal aeronáutico (Pilotos, Assistentes de bordo, etc.).

Módulo RAN: Este módulo tem as seguintes funções:

- a) **Registo Aeronáutico Nacional:** registo das aeronaves registadas em Cabo Verde: inclui todos os dados das aeronaves, proprietário, operador, matrícula data de emissão etc.
- b) **Dados de Certificação:** Aeronaves (dados para emissão de certificados navegabilidade, autorização especial de voo, etc), Operadores (dados dos operadores certificados, nome, número de certificação, autorizações), organizações de Manutenção (dados dos operadores certificados, nome, número de certificação, autorizações).

Módulo DSO: no módulo da **Divisão de Segurança Operacional**, são registadas informações relativas aos Operadores Aéreos, às Organizações de Manutenção Autorizadas (OMA), às Aeronaves e onde são emitidas as autorizações especiais de voo e os certificados de navegabilidade.

Módulo OJT: Neste módulo são registados os treinos que permitem ao funcionário obter novos códigos de actividades para o seu trabalho.

Módulo TESTE: Este módulo permite a gestão dos testes que posteriormente serão utilizados para a avaliação do pessoal.

Actividades Planeadas

Para o desenvolvimento do projecto foram planeadas as seguintes tarefas:

1. Levantamento dos dados – o apesar de ter sido planeada por todos, foi efectuado apenas pelo responsável do projecto;
2. Análise dos dados – também efectuado pelo responsável do projecto;
3. Desenho de Modelo Entidade Relacionamento;
4. Implementação do Sistema;
5. Teste;
6. Apresentação do projecto à Instituição;
7. Modificações
8. Entrega do sistema à Instituição.

Desenvolvimento dos Módulos

Para o desenvolvimento do projecto, foi destacado seis técnicos entre os quais a estagiária e uma responsável do projecto.

A responsável do projecto efectuou todo o levantamento dos requisitos e modelagem do sistema.

Cada módulo era da responsabilidade de um dos técnicos, mas podia sempre contar com a ajuda de outros técnicos caso for necessário ou caso o nível de tarefa exige tal.

O módulo de Parametrização foi desenvolvido por todos os técnicos de acordo com a necessidade de cada um e serviu como base de treino para o arranque do desenvolvimento dos restantes módulos. O módulo em que esteve mais envolvida, é o **Módulo TESTE**, que é responsável para gerar e realizar testes efectuados ao pessoal.

Tecnologias Utilizadas

SQL Navigator: A ferramenta que se utiliza na Instituição no desenvolvimento de aplicações no qual fui vinculado é o SQL Navigator 5.5 com o uso da linguagem PL/SQL, Javascript e HTML. O SQL Navigator é um ambiente de desenvolvimento completo para desenvolvimento e administração do lado de servidor para Oracle, provê um ambiente integrado para desenvolvimento e teste de stored programs, procedimentos, schemas, objetos, scripts de SQL etc. É uma ferramenta poderosa para o desenvolvimento de aplicativos pois fornece uma interface de usuário amigável o que facilita uma aprendizagem rápida dos seus vários componentes. Possui praticamente tudo que um técnico necessita para a modelagem de uma base de dados e sua exploração SQL.

PL/SQL: Essa linguagem da Oracle, uma extensão procedural ao SQL, é muito usada para implementar módulos de programação lógica para aplicativos. Ela pode ser usada para criar procedimentos armazenados e gatilhos (triggers), controles em laço (looping controls), consultas condicionais e tratamento de erros. Pode-se compilar e armazenar procedimentos PL/SQL na base de dados. Pode-se também executar blocos PL/SQL usando o SQL*Plus, uma ferramenta interativa incluída em todas as versões do Oracle. As unidades de programas PL/SQL podem ser pré-compiladas.

Reports Builder: Reports Builder é o componente de report-building de Oracle Reports Developer do SGBD Oracle (parte do Oracle Developer Suite), uma poderosa ferramenta que permite desenvolver relatórios em qualquer fonte de dados. Permite publicar relatórios em uma variedade de formatos (inclusive HTML, XML, PDF, planilha eletrônica, texto delimitado e Pós-escrito) para qualquer destino (inclusive e-mail, browser de Web, Portal de OracleAS, e sistema de arquivo) de forma eficiente, qualquer destino (inclusive e-mail, browser de Web, Portal de OracleAS, e sistema de arquivo) de forma eficiente.

HTML: linguagem com que se escrevem as páginas web. As páginas web podem ser vistas pelo usuário mediante um tipo de aplicação chamada navegador (browser). Podemos dizer portanto, que o HTML é a linguagem usada pelos navegadores para mostrar as páginas webs ao usuário, sendo hoje em dia a interface mais extensa na rede

Javascript: Usada para tornar as páginas mais interactivas.

É uma linguagem de programação utilizada para criar pequenos programinhas encarregados de realizar ações dentro do âmbito de uma página web. Com Javascript podemos criar efeitos especiais nas páginas e definir interatividades com o usuário. O navegador do cliente é o encarregado de interpretar as instruções Javascript e executá-las para realizar estes efeitos e interatividades, de modo que o maior recurso, e talvez o único, com que conta esta linguagem é o próprio navegador.

Oracle Database 10g: A Oracle Database 10g é a primeira base de dados desenhada para o paradigma de Grid Computing, é a forma mais flexível e a baixo custo para gerir a informação de uma empresa. Reduz significativamente os custos garantindo grande qualidade de serviço e alta disponibilidade. Para além das melhorias a nível de performance e disponibilidade, a base de dados Oracle 10g reduz significativamente os custos de administração dos sistemas de informação, com uma instalação, configuração e manutenção simplificados e com capacidade de automatização a nível de diagnóstico e tuning de SQL. A capacidade de auto gestão, permite ao DBA e aos developers maior produtividade e eficácia nas suas tarefas.

1.5 Contributos deste trabalho

O presente trabalho vai trazer grandes benefícios para a Agência de Aviação Civil, visto que neste momento o sistema de informação da AAC possui três bases de dados (DSO, PAN, SRS) que não estão integrados entre si.

A não integração da referida base de dados permitia que houvessem:

- Uma grande redundância de dados na base de dados
- Informações duplicadas
- Falta de controlo nas alterações dos dados pelos utilizadores

Com a nova BD, esses problemas vão ser resolvidos, visto que as três BD foram juntadas (integradas) numa única base de dados. Veja-se a figura 2.

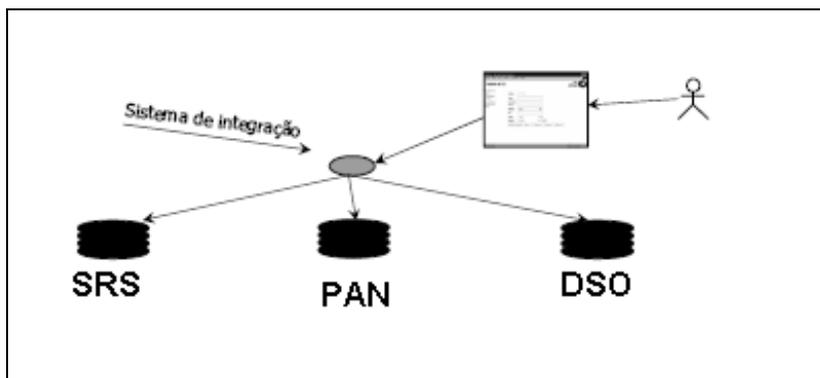


Figura 3 - Sistema integrado

Vantagens de uma BD integrada:

A integração das bases de dados DSO, PAN, SRS possui um conjunto de vantagens que a seguir se destacam:

- **Resposta rápida aos pedidos de informação.** Como os dados estão integrados numa única estrutura a resposta as questões complexas processam-se mais rapidamente.
- **Acesso múltiplo.** O software de gestão de base de dados permite que os dados sejam acedidos de diversas maneiras. Nomeadamente, os dados podem ser visualizados através de pesquisas sobre qualquer um dos campos da tabela.
- **Flexibilidade.** Em consequência da independência entre dados e programas, qualquer alteração num desses elementos não implica modificações drásticas noutras.

- **Integridade da informação.** Dada a absoluta exigência de não permitir a redundância, as modificações de dados são feitas num só sítio, evitando-se assim possíveis conflitos entre diferentes versões da mesma informação.
- **Melhor gestão da informação.** Em consequência da localização central dos dados, sabe-se sempre como e onde está a informação.

O controlo de acesso e alteração dos dados, é feito automaticamente pelo sistema. Para aceder a BD, como tinha dito anteriormente, a pessoa tem de estar cadastrado na BD e obter permissões que são atribuídos pelos agentes da AAC. O sistema desenvolvido tem a capacidade de registar automaticamente o nome do utilizador que acedeu ao sistema, data e hora do acesso, assim como qualquer alteração ou eliminação efectuado nos dados da base de dados.

2. Procedimentos Técnicos

Como tinha dito anteriormente o projecto foi desenvolvido por módulos e cada módulo é como se fosse da responsabilidade de um dos técnicos. O que não quer dizer, que não houve participação do mesmo técnico em outros módulos. Dependendo do nível de trabalho de cada módulo um técnico poderia receber a ajuda de outro técnico.

2.1 Modelo de Dados

O modelo de dados é um subconjunto do modelo de implementação que descreve a representação lógica e física dos dados persistentes no sistema. Também abrange qualquer comportamento definido no banco de dados, como procedimentos armazenados, triggers, restrições etc.

2.1.1 Modelo Entidade - Relacionamento

Na construção do modelo de dados foi usada a ferramenta Microsoft Office Visio, tendo o modelo sofrido várias alterações ao longo da execução do trabalho, de maneira a melhor se adaptar à informação que irá armazenar. A figura 3 mostra o modelo de dados do módulo teste.

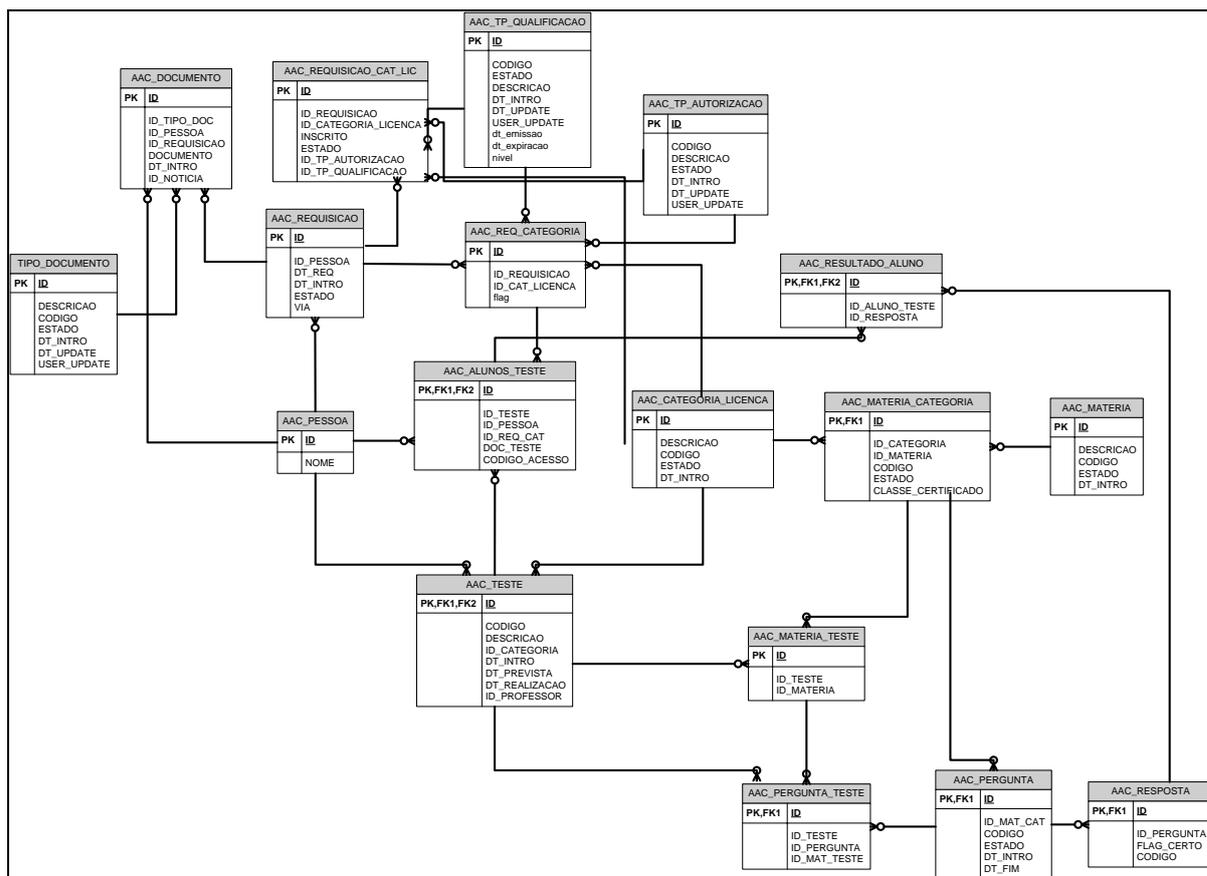


Figura 4 - Modelo de dados do módulo Teste

2.1.2 Criação de Tabelas na Base de dados

Com base no modelo relacional acima identificado, todas as tabelas foram criadas manualmente, utilizando Microsoft Visio 2007

Nesta secção é feita uma breve análise da estrutura das tabelas mais relevantes, indicando as colunas, o tipo de dados, o tamanho, etc.

Tabela AAC_perguntas

Nesta tabela é armazenada todas as possíveis perguntas utilizadas para a elaboração de testes automáticos. Veja-se a figura 4.

Table: AAC_PERGUNTA									
Columns									
Name	Type	Length	Scale	Not Null	Default	Comment	Encrypt	Salt	
ID	NUMBER			<input checked="" type="checkbox"/>			<input type="checkbox"/>	<input type="checkbox"/>	
PERGUNTA	VARCHAR2	2000		<input type="checkbox"/>			<input type="checkbox"/>	<input type="checkbox"/>	
CODIGO	VARCHAR2	20		<input type="checkbox"/>			<input type="checkbox"/>	<input type="checkbox"/>	
ESTADO	VARCHAR2	20		<input type="checkbox"/>			<input type="checkbox"/>	<input type="checkbox"/>	
DT_INTRO	DATE	7		<input type="checkbox"/>			<input type="checkbox"/>	<input type="checkbox"/>	
DT_UPDATE	DATE	7		<input type="checkbox"/>			<input type="checkbox"/>	<input type="checkbox"/>	
USER_UPDATE	VARCHAR2	20		<input type="checkbox"/>			<input type="checkbox"/>	<input type="checkbox"/>	
ID_MAT_CAT	NUMBER			<input type="checkbox"/>			<input type="checkbox"/>	<input type="checkbox"/>	

Figura 5 – Tabela AAC_perguntas

Tabela AAC_resposta

Nesta tabela é armazenada todas as respostas. Para cada resposta, é indicado o ID da tabela pergunta e o flag_certo (se é a resposta correcta ou errada). Veja-se a figura seguinte.

Table: AAC_RESPOSTA									
Columns									
Name	Type	Length	Scale	Not Null	Default	Comment	Encrypt	Salt	
ID	NUMBER			<input checked="" type="checkbox"/>			<input type="checkbox"/>	<input type="checkbox"/>	
ID_PERGUNTA	NUMBER			<input checked="" type="checkbox"/>			<input type="checkbox"/>	<input type="checkbox"/>	
CODIGO	VARCHAR2	20		<input checked="" type="checkbox"/>			<input type="checkbox"/>	<input type="checkbox"/>	
FLAG_CERTO	VARCHAR2	2		<input checked="" type="checkbox"/>		'C' - CERTO; 'E' = ERRADO	<input type="checkbox"/>	<input type="checkbox"/>	
RESPOSTA	VARCHAR2	400		<input type="checkbox"/>			<input type="checkbox"/>	<input type="checkbox"/>	

Figura 6 – Tabela AAC_resposta

Tabela AAC_requisição

Nesta tabela, FIGURA 6, fica armazenada toda a informação acerca das requisições efectuadas. Apenas as requisições. Ainda não estão associadas e nenhum teste.

Table: AAC_REQUISICAO									
Columns									
Name	Type	Length	Scale	Not Null	Default	Comment	Encrypt	Salt	
ID	NUMBER			<input checked="" type="checkbox"/>			<input type="checkbox"/>	<input type="checkbox"/>	
ID_PESSOA	NUMBER			<input checked="" type="checkbox"/>			<input type="checkbox"/>	<input type="checkbox"/>	
DT_INTRO	DATE	7		<input type="checkbox"/>			<input type="checkbox"/>	<input type="checkbox"/>	
DT_REQUISICAO	DATE	7		<input type="checkbox"/>			<input type="checkbox"/>	<input type="checkbox"/>	
ESTADO	VARCHAR2	1		<input type="checkbox"/>	A		<input type="checkbox"/>	<input type="checkbox"/>	
VIA	VARCHAR2	1		<input type="checkbox"/>			<input type="checkbox"/>	<input type="checkbox"/>	

Figura 7 – Tabela AAC_requisição

Tabela AAC_aluno_teste

Como mostra a figura 7, nesta tabela fica guardada as informações das associações do aluno_teste. Isto é da associação de uma requisição, que já vem com a identificação da pessoa com o teste a efectuar.

Name	Type	Length	Scale	Not Null	Default	Comment	Encrypt	Salt
ID	NUMBER			<input checked="" type="checkbox"/>			<input type="checkbox"/>	<input type="checkbox"/>
ID_TESTE	NUMBER			<input checked="" type="checkbox"/>			<input type="checkbox"/>	<input type="checkbox"/>
ID_PESSOA	NUMBER			<input checked="" type="checkbox"/>			<input type="checkbox"/>	<input type="checkbox"/>
ID_REQ_CAT_LIC	NUMBER			<input checked="" type="checkbox"/>			<input type="checkbox"/>	<input type="checkbox"/>
APROVACAO	VARCHAR2	2		<input type="checkbox"/>		S e N - Sim e Não	<input type="checkbox"/>	<input type="checkbox"/>
CODIGO_ACESSO	VARCHAR2	20		<input type="checkbox"/>			<input type="checkbox"/>	<input type="checkbox"/>

Figura 8 – Tabela AAC_aluno_teste

Tabela AAC_testes

Nesta tabela fica armazenada as informações acerca da elaboração. Para cada teste é criado um registo com os dados apresentados na tabela.

Name	Type	Length	Scale	Not Null	Default	Comment	Encrypt	Salt
ID	NUMBER			<input checked="" type="checkbox"/>			<input type="checkbox"/>	<input type="checkbox"/>
CODIGO	VARCHAR2	20		<input type="checkbox"/>			<input type="checkbox"/>	<input type="checkbox"/>
DESCRICAO	VARCHAR2	1000		<input type="checkbox"/>			<input type="checkbox"/>	<input type="checkbox"/>
ID_CATEGORIA	NUMBER			<input type="checkbox"/>			<input type="checkbox"/>	<input type="checkbox"/>
DT_INTRO	DATE	7		<input type="checkbox"/>			<input type="checkbox"/>	<input type="checkbox"/>
DT_PREVISTA	DATE	7		<input type="checkbox"/>			<input type="checkbox"/>	<input type="checkbox"/>
DT_REALIZACAO	DATE	7		<input type="checkbox"/>			<input type="checkbox"/>	<input type="checkbox"/>
DT_UPDATE	DATE	7		<input type="checkbox"/>			<input type="checkbox"/>	<input type="checkbox"/>
USER_UPDATE	VARCHAR2	100		<input type="checkbox"/>			<input type="checkbox"/>	<input type="checkbox"/>
ID_PROFESSOR	NUMBER			<input checked="" type="checkbox"/>		Pessoa que elaborou o teste	<input type="checkbox"/>	<input type="checkbox"/>
TEMPO	VARCHAR2	50		<input type="checkbox"/>		Tempo de Execução em minutos	<input type="checkbox"/>	<input type="checkbox"/>
ESTADO	VARCHAR2	2		<input type="checkbox"/>			<input type="checkbox"/>	<input type="checkbox"/>
FLAG_REALIZADO	VARCHAR2	2		<input type="checkbox"/>		Sim e Não	<input type="checkbox"/>	<input type="checkbox"/>

Figura 9 – Tabela AAC_teste

Tabela AAC_resultado_aluno

Nesta tabela fica armazenada as respostas dos alunos pelo teste efectuado. Esta tabela possui apenas 2 campos de ligação (id_aluno_teste), que permite identificar o aluno e o teste e (id_resposta), que permite identificar as perguntas, a partir da resposta, sendo que para cada resposta, há uma pergunta, ou seja um id da pergunta.

Name	Type	Length	Scale	Not Null	Default	Comment	Encrypt	Salt
ID	NUMBER			<input checked="" type="checkbox"/>			<input type="checkbox"/>	<input type="checkbox"/>
ID_ALUNO_TESTE	NUMBER			<input checked="" type="checkbox"/>			<input type="checkbox"/>	<input type="checkbox"/>
ID_RESPOSTA	NUMBER			<input checked="" type="checkbox"/>			<input type="checkbox"/>	<input type="checkbox"/>

Figura 10 – Tabela AAC_resultado_aluno

2.2 Interface Web

Neste capítulo é analisada a interface web desenvolvida para a Base de Dados SGI - AAC.

Esta interface está dividida em dois grupos, a interface de Parametrização e a interfaces de utilização.

Para ter acesso a essas interfaces, o utilizador tem que se autenticar introduzindo o seu código e password no formulário de controlo de acesso, usado em praticamente todas aplicações desenvolvidas pelo NOSI.

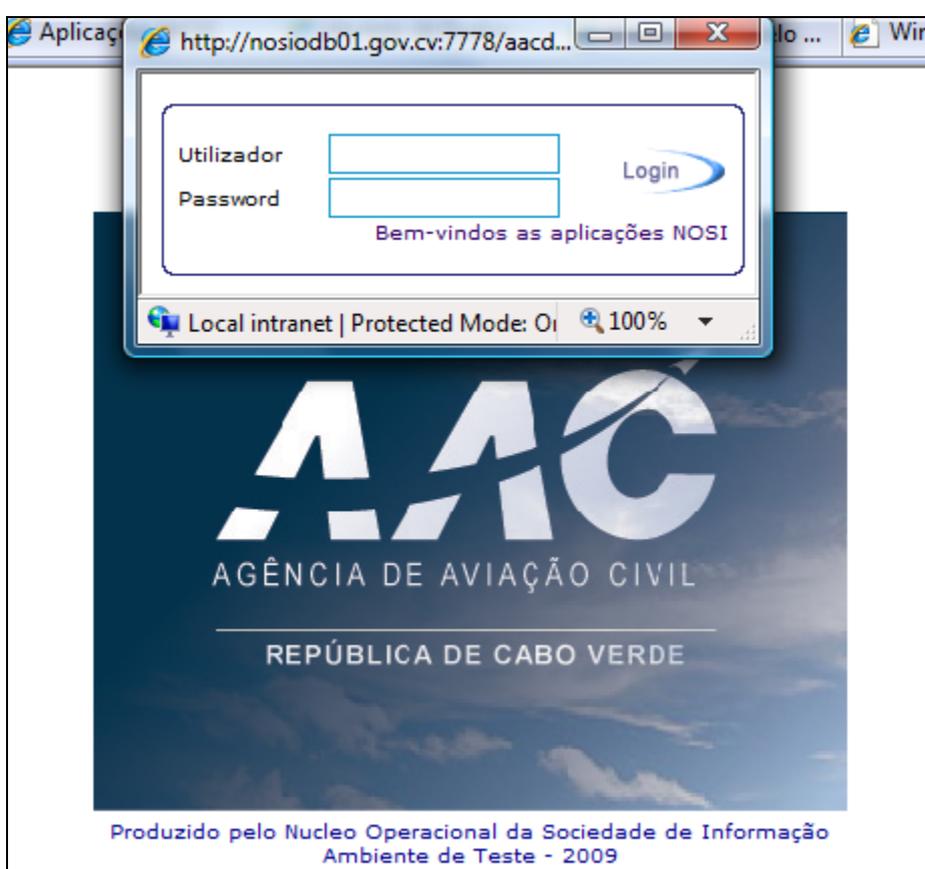


Figura 11 – Pagina inicial da aplicação

Interface de Parametrização

Depois de se autenticar, se tiver um user com permissão de administrador terá acesso a todas as funcionalidades da aplicação (Parametrização e Módulos), podendo aceder às várias páginas muito facilmente através dos menus disponíveis.

Criação das Perguntas

O formulário para criação das perguntas, é bastante fácil, assim como todos os outros demais. Ver a figura a seguir.

The image shows a web form titled "Pergunta". At the top, there is a navigation bar with several menu items: TESTES, PAN, SRS, RAN, DSO, LICENCA, TREINO, and OUTROS. Below this, the form is divided into a "Dados" section. It contains four input fields: "Categoria:" with a dropdown menu showing "Licença de Piloto Privado"; "Matéria:" with a dropdown menu showing "geografia"; "Código:" with a text input field containing "001"; and "Pergunta:" with a text input field containing a series of "x" characters. At the bottom of the form, there are two buttons: "Confirmar" and "Voltar".

Figura 12 – Formulário de inserir perguntas

O utilizador, apenas terá de escolher nos respectivos combos a Categoria e a Matéria que estarão previamente parametrizados pelo administrador. De seguida, colocar o código e descrição da pergunta.

Para finalizar fazer o clique em confirmar. O sistema vai efectuar o **tratamento** da informação. O tratamento é um procedimento que faz a validação da informação. Indica se está a faltar alguma informação ou não. Se estiver tudo correcto, a informação é inserida na BD e o utilizador recebe a mensagem da confirmação do Sucesso, como mostra a figura abaixo.

Código	Pergunta	Estado	Data de Registo
002	Mmmmmmmmmmmmmmmmmmm	A	13-OCT-09

Novo

OK - OPERAÇÃO EFECTUADA COM SUCESSO

Figura 13 – Tela de inserção efectuada com sucesso

Criação das Respostas

No menu Parâmetros Gerais, após escolher a opção Testes e de seguida Respostas, obterá um campo com filtros que ajudam nas pesquisas das respostas.

Para criar uma nova resposta, o utilizador, terá de clicar em NOVO.

Surge o seguinte formulário que permite criar as respostas.

Resposta

TESTES ▾ PAN ▾ SRS ▾ RAN ▾ DSO ▾ LICENCA ▾ TREINO ▾ OUTROS ▾

Dados

Categoria: Licença de Piloto Privado Materia: matematica

Pergunta: Pergunta Tosca deveras HEM

Resposta: XXXXXXXXXXXXXXXX

Código: 123

Resultado: certo errado

Confirmar **Lista**

Figura 14 – Formulário de inserção de Respostas

De acordo com a Categoria e a Matéria seleccionada, é disponibilizada as perguntas em relação aquela Categoria e Matéria.

É necessário inserir a descrição da resposta e se é certa ou errada. Para cada pergunta, podemos ter n respostas, sendo que apenas uma pode ser marcada como certa.

Requisição

Para fazer a requisição de um teste, é necessário que o requisitante, esteja registado no Módulo PAN. Para cada registo de PAN, existem diversos tabs, sendo um deles TESTES.

Como mostra a figura, neste tab, é possível fazer as requisições de testes para determinadas categorias

Dados Pessoais Dados Médicos Licença Qualificações Autorizações Testes						
Pessoa: 124 Contador: 6						
Teste	Data de requisição	Categoria Licença	Inscrito	Via	Resultado	
Teste 1	12-AUG-09	Licença de Piloto Privado	S	Internet	484	
Teste 1	12-AUG-09	Licença de Piloto Privado	S	Internet	485	
Teste 1	12-AUG-09	Licença de Piloto Privado	S	Internet	486	
Teste 1	12-AUG-09	Licença de Piloto Privado	S	Internet	487	
Teste 3	18-AUG-09	Licença de Piloto de Linha Aérea	S	Nenhum		
Vamos Fazer Teste Bora La	12-AUG-09	Licença de Piloto Privado	S	Internet		
Total:6						
<input type="button" value="Gravar"/> <input type="button" value="Nova R."/>						

Figura 15 – Formulário de requisição de testes

Após clicar no botão Nova R. (Nova requisição), surge a lista de Categorias de testes disponíveis.

É necessário escolher a categoria e clicar em confirmar.

Fica automaticamente registado na BD a requisição efectuada.

Dados Pessoais Dados Médicos Licença Qualificações Autorizações Testes		
Informação de Requerimento		
Categoria Licença <input type="checkbox"/> <ul style="list-style-type: none"> <input type="checkbox"/> Licença de Piloto Privado <input type="checkbox"/> Licença de Piloto Comercial34 <input type="checkbox"/> Licença de Piloto de Linha Aérea <input type="checkbox"/> Licença de Técnico de Manutenção de Aeronave <input type="checkbox"/> Licença de Controlador de Trafego Aéreo <input type="checkbox"/> Licença Oficial de Operações de Voo <input type="checkbox"/> Licença <input type="checkbox"/> Licença de Tripulação de Cabina <input type="checkbox"/> Licença de Operador de Estação Aeronáutica <input type="checkbox"/> Licença de Engenheiro de VOO <input type="checkbox"/> Licença de Navegador <input type="checkbox"/> Fwefwefwef <input type="checkbox"/> rr432r34t <input type="checkbox"/> Hgrtherher <input type="checkbox"/> Teste Numero_altera <input type="checkbox"/> Vania <input type="checkbox"/> Teste 3 <input type="checkbox"/> Van <input type="checkbox"/> Teste Hoje2 <input type="checkbox"/> Teste <input type="checkbox"/> Teste Hoje 	Qualificacao <input type="checkbox"/> <ul style="list-style-type: none"> <input type="checkbox"/> Sdfgsf <input type="checkbox"/> Dfsdfs <input type="checkbox"/> Sds <input type="checkbox"/> Dsfsdf <input type="checkbox"/> Dsfsdf <input type="checkbox"/> Dsgsg <input type="checkbox"/> Jhdsgfsgfjsgdfjhsgd <input type="checkbox"/> Dfjsjsgfhgsdfghfshdkdhfdsjs <input type="checkbox"/> Sfdfgdfgdg <input type="checkbox"/> Ysfdgfdgsfjsgfsgdshgfhgsd <input type="checkbox"/> Teste Escola 1 <input type="checkbox"/> Teste Escola 2 <input type="checkbox"/> Atr <input type="checkbox"/> Y <input type="checkbox"/> Sdas <input type="checkbox"/> Sdfsdfsaf <input type="checkbox"/> Zcdvvcvzcxbcx <input type="checkbox"/> Teste 1 	Autorizacao <input type="checkbox"/> <ul style="list-style-type: none"> <input type="checkbox"/> Aut. Abastecimento <input type="checkbox"/> Aut <input type="checkbox"/> Autorização de Voo <input type="checkbox"/> Autorização de aterragem Nac. <input type="checkbox"/> Autorização de decolagem <input type="checkbox"/> Autorização de aterragem Int. <input type="checkbox"/> Teste 1 <input type="checkbox"/> Teste 2

Figura 16 – Lista de categorias de testes

Associar aluno ao teste

Após ter efectuado a requisição, é necessário que um dos inspectores da AAC, faça a atribuição de teste ao Aluno.

Para o efeito, é necessário ir no menu Pesquisa teste, onde poderemos obter a lista de todos os testes elaborados que estão activos. Ver a figura 16.

Sobre o teste, clicar em detalhes. Com esta acção poderemos ver a lista dos alunos associados a este teste.

Filtro

Código: Descrição:

Data Realização: Data Prevista:

Categoria: Licença

Lista

Código	Descrição	Categoria	Data real	Professor
AAAAAA	QWFQEFQEFQ	Licença de Piloto Privado		Zeguy7
EEE	Rgsrgerer	Licença de Piloto Privado		Zeguy7
DSVCF	Sagwcbvs	Licença de Piloto Privado		Zeguy7
DVGS	Sbvdgvdntb	Licença de Piloto Privado		Zeguy7
ASVDV	Svcnfjhgbvfdcs	Licença de Piloto Privado		Zeguy7
TESTERTW	Tese Teste Ver Hoje	Licença de Piloto Privado		Zeguy7
TESTEW	Tese Teste Ver Hoje	Licença de Piloto Privado		Zeguy7
T1	Teste 1	Licença de Piloto Privado	10-SEP-09	Zeguy7
T2	Teste	Licença de Piloto Comercial34		Joelma Tavares2
T3	Teste	Licença de Piloto de Linha Aérea	08-SEP-09	Joelma Sofia Dos Santos Tavares

Total:25

Figura 17 - Pagina pesquisa teste

Dados

Código: T1 Descrição: Teste 1

Data Realização: 10-SEP-09 Data Prevista: 08-AUG-09

Categoria: Licença de Piloto Privado

Imprimir Teste:

Lista Inscritos

ID	Nome	Data Requ.	Resultado	Eliminar
124	Joelma Sofia Dos Santos Tavares	12-AUG-09	Ver	✘

Total:1

Figura 18 – Tela de inscrição de aluno ao teste

A figura 17, mostra a tela de inscrição do Aluno. Nesta tela poderá ser efectuado as seguintes accções:

Eliminar a atribuição – utilizando a imagem (✖). De realçar que o sistema não pemrite que se elimine uma atribuição de um aluno com resultado já inserido.

Inserir e ver o resultado do teste em relação ao aluno – Se estiver com resultado inserido, na coluna Resultado, aparece o texto VER. Para visualizar, clique sobre o texto.

É disponibilizado do seguinte Pop-up, como mostra a figura 18, com o resultado obtido. As respostas com cor vermelha são respostas que o aluno escolheu e que estão erradas e verdes as correctas. No final é indicado se houve Aprovação (mais do que 70% das respostas correctas) ou reprovado.

The image shows a pop-up window with a white background and a thin black border. It contains several sections, each starting with a question label in a rounded rectangular box. Below each question are several lines of text representing answers. The text is as follows:

- Pergunta 3**
dede
testeresposta
teste pergunta 2
teste222
Resposta 1 Pergunta 3
Resposta 2 Pergunta 3 (highlighted in red)
- Pergunta 4**
iwertuetiyrtoiyuipulo
- Pergunta 5**
ieryturtiouypuip
- Pergunta 7**
teste graf1 (highlighted in green)
teste graf2
teste graf3
- Pergunta 6**
Reprovado

At the bottom of the window, there are two dark blue buttons with white text: "Fechar" and "Limpar".

Figura 19 – Teste com o resultado

PERGUNTAS / RESPOSTAS

Pergunta 1

vvvvvvvvvvvvvvvv

aaaaaaaaaaaaaaab

dddddddddddddd

Resposta 1 Pergunta 1

jehrhtyuty5+ypé

rrrrrrrrrrrrrrrrrr

7777777777777777

Pergunta 2

Resposta 1 Pergunta 2

Resposta 2 Pergunta 2

Resposta 2 Pergunta 1

Pergunta 3

Figura 20 – Formulário de inserção das respostas escolhidas pelo aluno

Nova associação de Aluno ao teste - Para associar novo aluno ao teste, apenas clicar no Botão Nova Inscrição.

Surge a lista dos alunos que se inscreveram para a mesma categoria do teste. Ver figura 20.

Dados

Codigo: Descricao:

Data Realização: Data Prevista:

Categoria:

Imprimir Teste:

Lista Inscritos

ID	Nome	Data Requ.	Check
148	JERSON CORREIA FIGUEIREDO	11-AUG-09	<input type="checkbox"/>
195	Hélio Africano Monteiro Querido Varela	17-SEP-09	<input type="checkbox"/>
148	JERSON CORREIA FIGUEIREDO	11-AUG-09	<input type="checkbox"/>

Total:3

Figura 21 – Lista dos alunos inscritos num teste

Elaboração de Teste

Para elaborar um teste, o Inspector terá de aceder ao menu Módulos testes e escolher a opção Elaborar Teste. Surge a tela identificado como figura 21.

Elaborar Teste

Categoria

Categoria: Licença de Piloto Privado Código: 234 Descrição:

Teste para categoria de piloto privado Data Prevista: 23/10/2009

Materia

Fefefe
geografia
geografia
Blublubu2
Blublubu2
vv
Joelmaaaaaaaaaaaaaaaaaaaaa

Materia escolhida	Total	Total Escolhida
✖ matematica	15	15
✖ teste_upd	6	6
✖ Terra teste	2	2
✖ geografia	10	10

Gerar

Total de Perguntas: NaN

Numero de Perguntas: 33

Figura 22 - Formulário de elaboração de Teste

É necessário escolher a categoria do teste e alguns outros dados como indica a figura acima.

De acordo com a categoria, na coluna Materia, é disponibilizada a lista das matérias associadas.

Ao clicar sobre a matéria, da coluna ao lado, surge a descrição da matéria com duas colunas enumerada. Uma com o Total das perguntas da Materia e outra para indicar o total de perguntas daquela matéria que pretende atribuir ao teste.

Para finalizar clique em Confirmar. Automaticamente o sistema gera o teste, com as suas perguntas e respostas

2.3 Implementação do SI-AAC

2.3.1 Análise dos Procedimentos e Funções mais Relevantes

Nesta secção é feita uma breve análise a alguns dos procedimentos, referindo as principais funcionalidades que implementam. De realçar que para a fase de implementação seguiu-se a seguinte metodologia de trabalho.

Para cada tabela foi criado duas packages (conjunto de procedimentos e funções relacionados entre si) e sua estrutura

- Package BD + nome da tabela (AAC_BD_REQUISICAO)
- Package Páginas + nome da tabela (AAC_PAGINAS_REQUISICAO)

A package **BD** é utilizada para criar os procedimentos que fazem ligação com a base de dados. Leitura e escrita nas tabelas.

A package **Páginas** é utilizada para criar os formulários de inserção, consulta e outros. Trata da parte visual que o utilizador tem acesso.

A **estrutura**, permite criar variáveis de memórias temporárias.

Quando for necessário fazer uma consulta, não trabalhamos directamente na tabela, mas sim utilizamos a tabela de memória **col_t_...** Ex: **col_t_teste**

A variável da memória pode ser do tipo da tabela ou do tipo de uma **View** (consiste num comando SELECT realizado sobre tabelas ou outras views. No entanto as views não existem fisicamente).

Também é criado mais duas variáveis, **v_t_.....** e **v_col_...**, sendo que **v_t_.....** pode ser do tipo da tabela ou view, conforme declararmos o tipo da variável da memória **col_t_....** e **v_col_....** do tipo **col_t_....**. Ver a figura 22.

```

50 ----- Tabela AAC_VIEW_TESTE_RSP -----
51 type col_t_teste_rpt IS table of aac_v_teste_rpt%rowtype index by binary_integer;
52
53 v_t_teste_rpt aac_v_teste_rpt%rowtype ;
54
55 v_col_teste_rpt col_t_teste_rpt;
56
57
58 ----- Tabela AAC_TESTE com view-----
59 type col_t_teste IS table of aac_v_teste%rowtype index by binary_integer;
60
61 v_t_teste aac_v_teste%rowtype ;
62
63 v_col_teste col_t_teste;
64
65 ----- Tabela AAC_TESTE com tabela-----
66 type col_t_teste1 IS table of aac_teste%rowtype index by binary_integer;
67
68 v_t_teste1 aac_teste%rowtype ;
69
70 v_col_teste1 col_t_teste1;
71
72
73
74 ----- Tabela AAC_PERGUNTA -----
75 type col_t_PERGUNTA IS table of aac_pergunta%rowtype index by binary_integer;
76
77 v_t_PERGUNTA aac_PERGUNTA%rowtype ;
78
79 v_col_PERGUNTA col_t_PERGUNTA;
80

```

Figura 23 - Package aac_tipos

2.3.1.1 – Análise de alguns procedimentos mais relevantes da package BD

Procedimento le_bd

```

PROCEDURE le
IS
BEGIN
    SELECT *
    BULK COLLECT INTO aac_tipos.v_col_teste
    FROM aac_v_teste order by descricao
    END;

```

Este procedimento é responsável para fazer a leitura da base de dados e colocar os mesmos dados numa tabela de memória v_col_teste (**BULK COLLECT INTO** aac_tipos.v_col_teste).

Normalmente utilizamos esse procedimento quando queremos criar uma listagem de dados da tabela. Neste exemplo, o procedimento vai ler todos os dados que estão na View Teste e armazenar na tabela de memória. Quando chamamos esse procedimento obteremos uma listagem, nesse exemplo específico, todos os testes, ordenado por descrição.

Procedimento GRAVA

```
PROCEDURE grava (p_accao varchar2)
IS

BEGIN

IF p_accao IN ('INSERIR')
THEN
    INSERT INTO aac_teste
        (CODIGO,
        DESCRICAO,
        ID_CATEGORIA,
        DT_INTRO,
        DT_PREVISTA,
        ESTADO,
        ID_PROFESSOR
        )
    VALUES (aac_tipos.v_t_teste.CODIGO,
        aac_tipos.v_t_teste.DESCRICAO,
        aac_tipos.v_t_teste.ID_CATEGORIA,
        SYSDATE,
        aac_tipos.v_t_teste.DT_PREVISTA,
        'A',
        aac_tipos.v_t_teste.ID_PROFESSOR
        )
    RETURNING ID
        INTO aac_tipos.v_t_teste.ID;
    COMMIT;

-----

ELSIF upper(p_accao) = 'ALTERAR'
THEN
    UPDATE aac_teste
    SET codigo          = xml.get ('P_COD'),
        descricao       = xml.get ('P_DESC'),
        id_categoria    = aac_tipos.v_t_teste.id_categoria,
        DT_REALIZACAO  = to_date(xml.get
        ('p_dt_real'), 'dd/mm/yyyy'),
```

```
DT_prevista      = to_date(xml.get
('P_DT_PREV'), 'dd/mm/yyyy'),
dt_update        = sysdate

WHERE ID = xml.get ('p_id');
COMMIT;

-----

ELSIF p_accao = 'ELIMINAR'
THEN
update aac_teste
set estado = 'E'
WHERE ID = xml.get ('p_id');
COMMIT;
END IF;
END;
```

Este procedimento, existem em todas as packages BD. Permite inserir, actualizar e eliminar registos da tabela, dependendo do p_accao. O p_accao é um parâmetro que temos de enviar no momento em que estamos a chamar o procedimento GRAVAR. A p_accao deve ser INSERIR, ACTUALIZAR OU ELIMINAR. Dependendo dos casos, o if do procedimento grava é utilizado pelo sistema.

Ex: aac_bd_teste.grava ('ELIMINAR')

Nesse exemplo, é chamado primeiramente a package aac_bd_teste e de seguida o procedimento grava, em que a p_accao enviada é eliminar. O sistema vai fazer a actualização do estado de teste de A (activo) para E (eliminado), como está definido o procedimento :

```
ELSIF p_accao = 'ELIMINAR'
THEN
update aac_teste
set estado = 'E'
WHERE ID = xml.get ('p_id');
COMMIT;
END IF;
```

Procedimentos da package HTML

Nesta package, fica localizado todos os procedimentos que criam utilitários de formatação e formulários. Utilitários esses que depois serão utilizados por qualquer técnico do projecto SGI-AAC para criar e formatar formulários (Combos, coluna, Linhas, Lista de selecção, butom radio, tcheck box, botão etc).

No momento de desenhar o formulário, apenas temos de chamar o nome do procedimento e mandar os parametros necessários.

Procedimento mete_botao

Este procedimento é responsavél para criar um botão de submit

```
PROCEDURE mete_botao (  
  p_valor          VARCHAR2 DEFAULT 'Enviar',  
  p_type           VARCHAR2 DEFAULT 'submit',  
  p_atributos     VARCHAR2 DEFAULT NULL,  
  p_id            VARCHAR2 DEFAULT 'submit',  
  p_class         VARCHAR2 DEFAULT 'button'  
)  
IS  
BEGIN  
  HTP.p ( ' <input type="'  
    || p_type  
    || ' " name="submit" id="'  
    || p_id  
    || ' " class="'  
    || p_class  
    || ' " value="'  
    || p_valor  
    || ' "'  
    || p_atributos  
    || ' />' )  
  );  
END;
```

No momento de desenhar o botão, apenas temos de inserir o nome do procedimento e o nome que queremos dar ao botão

```
http.p('<table border=0 align="center">');
```

```
    http.p('<tr>');
    http.p('<TD align="Center" >');
    → aac_html.mete_botao('Confirmar', p_atributos=>'
        onclick="javascript:window.close();" ');
    → aac_html.mete_botao(Lista);
    http.p('</td>');
    http.p('</table>');
    aac_html.fecha_coluna;

aac_html.fecha_linha;
```

<input type="button" value="Confirmar"/>	<input type="button" value="Lista"/>
--	--------------------------------------

Procedure abre_linha e abre_coluna'

Procedimentos que permite, colunas e linhas de uma tabela para um formulário.

```
PROCEDURE abre_linha
```

```
IS
```

```
BEGIN
```

```
    HTP.p ('<tr>');
```

```
END;
```

```
PROCEDURE abre_coluna (
```

```
    p_nr_coll    NUMBER DEFAULT 1,
```

```
    p_align      VARCHAR2 DEFAULT 'left'
```

```
)
```

```
IS
```

```
BEGIN
```

```
    HTP.p ('<td align=" ' || p_align || ' " colspan=" ' || p_nr_coll ||
        '>');
```

```
    END;
```

```
PROCEDURE fecha_coluna
```

```
IS
```

```
BEGIN
```

```
    HTP.p ('</td>');
```

```
END;
```

Procedimento para validar dados.

```
PROCEDURE valida_dados_ecran
IS
BEGIN

    IF xml.get ('p_accao') <> 'ELIMINAR'
    THEN
        IF aac_tipos.v_t_materia.codigo IS NULL
        THEN
            xml.ADD ('p_msg', 'Tem que preencher oCodigo');
        END IF;

        IF aac_tipos.v_t_materia.descricao IS NULL
        THEN
            xml.ADD ('p_msg', 'Tem que preencher o Descricao');
        END IF;

        IF aac_tipos.v_t_materia.ID IS NULL
        THEN
            xml.ADD ('p_accao', 'INSERIR');
        END IF;
    END IF;

    IF xml.get ('p_accao') = 'ELIMINAR' OR xml.get ('p_accao') =
'ALTERAR'
    THEN
        IF aac_tipos.v_t_materia.ID IS NULL
        THEN
            xml.ADD ('p_msg', 'Tem que ter o ID');
        END IF;
    END IF;

    IF xml.get ('p_msg') IS NOT NULL
    THEN
        RAISE NO_DATA_FOUND;
    END IF;
END;
```

Este procedimento permite validar a informação constante na tabela temporária. Nesse caso, se ao gravar a informação, estiver a faltar dados em algum dos campos indicados no procedimento (código, descrição, ID), o utilizador vai receber uma mensagem de erro, indicando a informação que falta ser preenchida.

Procedimento para desenhar formulário

Esse é uma parte do **procedure show_page**, que é responsável para efectuar o desenho do formulário para inserção da matéria.

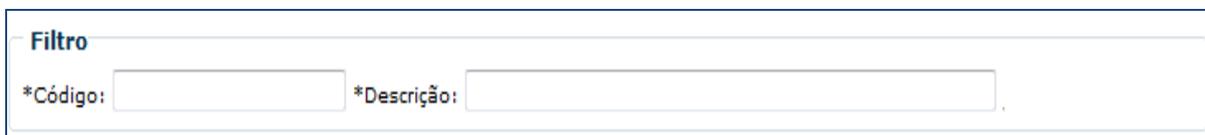
```
aac_html.abre_fieldset (v_fieldset);
aac_html.abre_tabela;
aac_html.abre_linha;
aac_html.abre_coluna (2);

aac_html.mete_input (p_tipo           => 'text',
                    p_nome_campo     => 'p_cod',
                    p_label          => 'Código',
                    p_val_required   => TRUE,
                    p_size           => 10,
                    p_read           => v_read,
                    p_valor_efeito    =>
aac_tipos.v_t_materia.codigo
                    );
aac_html.mete_input (p_tipo           => 'text',
                    p_nome_campo     => 'p_desc',
                    p_label          => 'Descrição',
                    p_val_required   => TRUE,
                    p_size           => 50,
                    p_read           => v_read,
                    p_valor_efeito    =>
aac_tipos.v_t_materia.descricao
                    );
```

Esse procedimento, vai primeiramente desenhar um fieldset - que é como se fosse uma tabela ou border de uma página. De seguida criar uma tabela com uma linha e duas colunas.

Na primeira coluna - é chamado o procedimento – aac_html.mete_input – que cria uma caixa de texto, que vai receber a descrição de Código e valor codigo que será inserido na tabela.

Na segunda coluna - é chamado o procedimento – aac_html.mete_input – que criar uma caixa de texto, que vai receber a descrição de Descrição e valor descricao que será inserido na tabela. A imagem a seguir mostra o resultado dessa acção.



Filtro

*Código: *Descrição:

3.3.1.2 – Análise de algumas Funções mais relevantes

Função tem resultado

```
FUNCTION tem_resultado (p_id_aluno_teste aac_resultado_aluno.id%TYPE)  
RETURN BOOLEAN IS
```

```
v_id_aluno_teste aac_resultado_aluno.id%TYPE;
```

```
BEGIN
```

```
SELECT COUNT (id) INTO v_id_aluno_teste  
FROM aac_resultado_aluno a  
WHERE a.id_aluno_teste = p_id_aluno_teste;
```

```
IF v_id_aluno_teste > 0 THEN
```

```
    RETURN TRUE;
```

```
ELSE
```

```
    RETURN FALSE;
```

```
END IF;
```

```
END;
```

Essa é uma função que retorna um valor Verdadeiro ou falso. Nesse caso, a função recebe um id (p_id_aluno_teste), e vai procurar na tabela (aac_resultado_aluno), se existe algum id igual. Se existir retorna um valor verdadeiro e se não existir valor falso. Essa função foi utilizada para fazer o seguinte cenário :



ID	Nome	Data Requ.	Resultado	Eliminar
124	Joelma Sofia Dos Santos Tavares	12-AUG-09	Ins	✘

Nova Inscrição
Total: 1

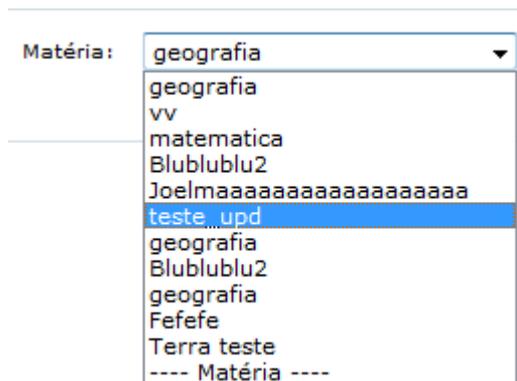
Se for True – na coluna resultado, teremos VER – o que quer dizer que já existe resultado inserido.

Se for FALSE – NA COLUNA resultado INS – que demonstra que não tem resultado. O INS é para inserir resultado porque ainda não tem inserido.

Função Get Materia

```
FUNCTION get_materia
RETURN aac_tipos.tab_record
IS
    v_mat aac_tipos.tab_record;
BEGIN
    v_mat:= aac_tipos.tab_record ();
        SELECT DISTINCT                a.descricao, a.id
            BULK COLLECT INTO          v_mat
        FROM                            aac_materia a
        WHERE                            a.estado ='A'
        ORDER BY a.descricao ASC;
    v_mat.EXTEND();
    v_mat(v_mat.last).opcao := '-- Matéria --';
    RETURN v_mat;
EXCEPTION
    WHEN NO_DATA_FOUND
    THEN NULL;
    WHEN OTHERS
    THEN NULL;
END;
```

Essa função faz o select de um conjunto de valores e vai guardar esses valores numa variável de retorno. Nesse exemplo específico, vai procurar na tabela aac_materia, o id e a descrição da materia e vai armazenar esses valores na variavel *v_mat* que é a variável de retorno. Quando chamamos essa função, recebemos a variável *v_mat*, com os valores armazenados. Essa função é utilizada sempre que queremos ter a lista de matérias, num combo ou de outra forma qualquer, como mostra a imagem a seguir:



2.3.1.3 – Views

Representação lógica de dados de uma ou mais tabelas.

Uma View funciona de forma semelhante a uma tabela. É utilizada em comandos SELECT, INSERT, UPDATE e DELETE, para recuperação e manipulação de dados (com restrições), porém, não armazena esses dados.

Este objeto tem suas linhas e colunas calculadas dinamicamente através de um SELECT pré-estabelecido, cada vez que solicitamos. Apenas a sua definição é armazenada no dicionário de dados.

Podemos dizer que se trata de uma tabela virtual, pois não possui linhas próprias, mas sim as obtém em tempo de execução e as disponibiliza em memória para acesso por uma query.

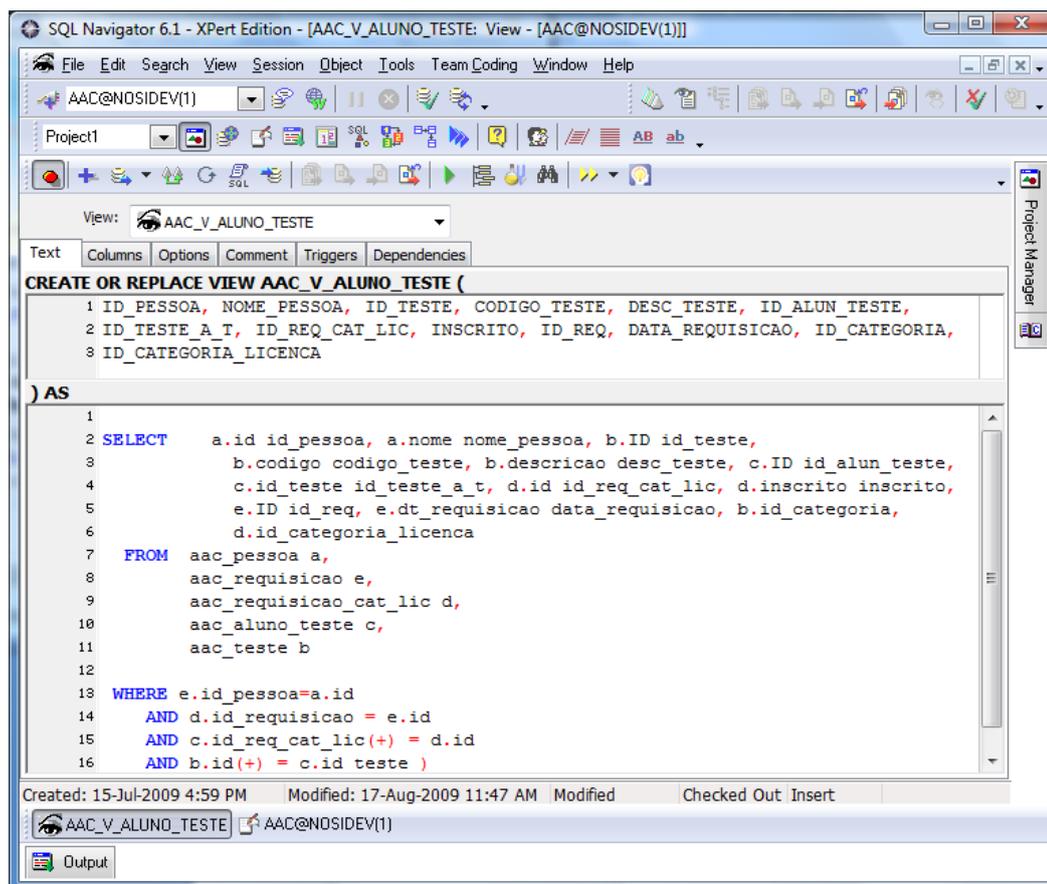


Figura 24 - Tela de criação de uma View

Nesse exemplo temos uma view que junta 5 tabelas, que têm uma interligação entre si. As views, também criamos quando queremos criar listagens com campos e diversas tabelas e também para criar reports.

2.3.1.3 –Reports

No Módulo gestão de teste, desenvolveu-se dois Report:

- 1 – Teste com correção – indicando as respostas correcta e erradas
- 2 – Teste sem correção – indicando apenas perguntas e respostas

A seguinte figura mostra a janela de desenho de formulário. Através do ícone Paper Layout se pode ter acesso a essa janela.

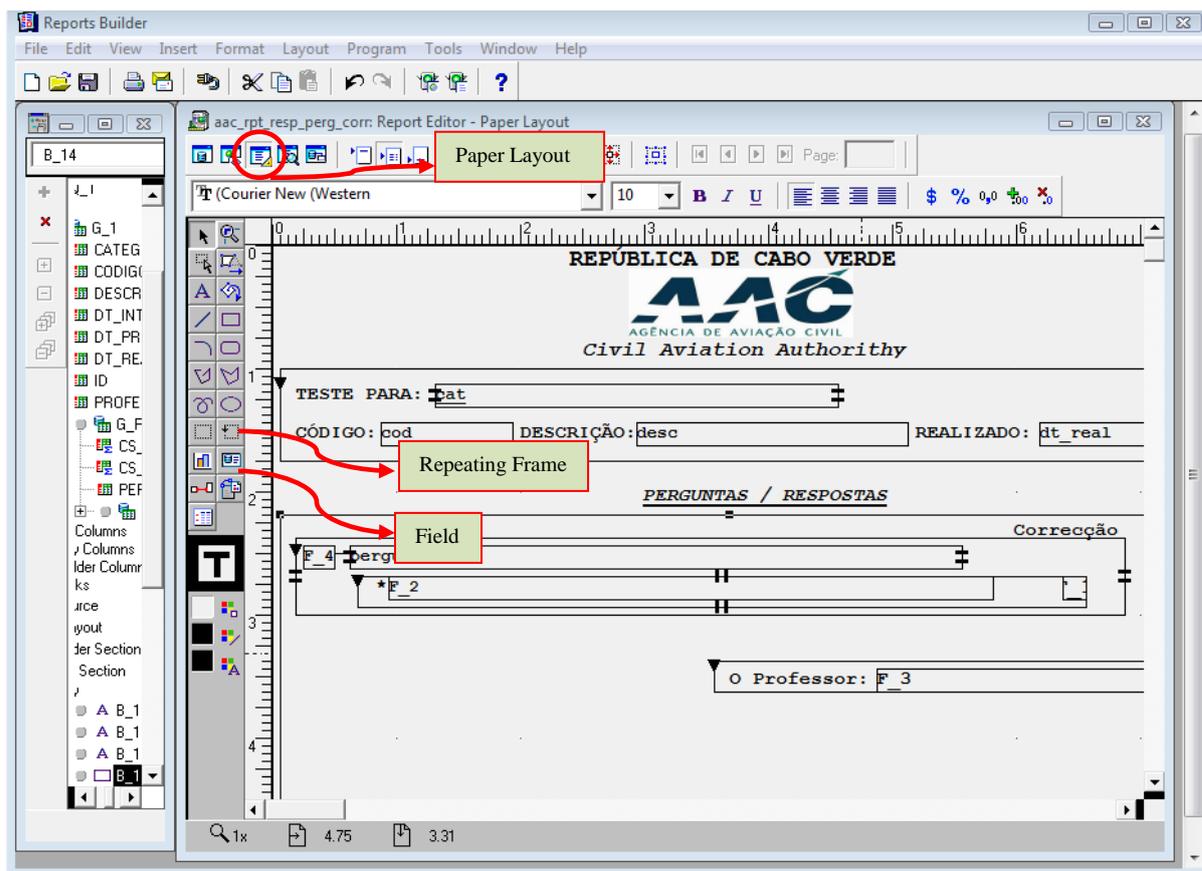


Figura 25 - Report Builder - vista de paper layout

Na figura o ícone **Repeating Frame** permite agrupar o conjunto dos campos que se quer mostrar na tabela, ou seja, define uma área y dentro dela se vai colocando os **Field** e se organizam conforma a figura. Para configurar cada field com o respectivo campo deve-se dar duplo clic sobre o field e abre a janela de configuração onde se especifica o nome o Field e o **Source** corresponde o campo da tabela seleccionada na consulta previamente feita.

Consulta que alimenta o Reporte

Antes do desenho do report, é necessário fazer a consulta onde se especificam o que se quer mostrar no report.

Para isso se deve clicar no ícone Data Model como mostra a seguinte figura

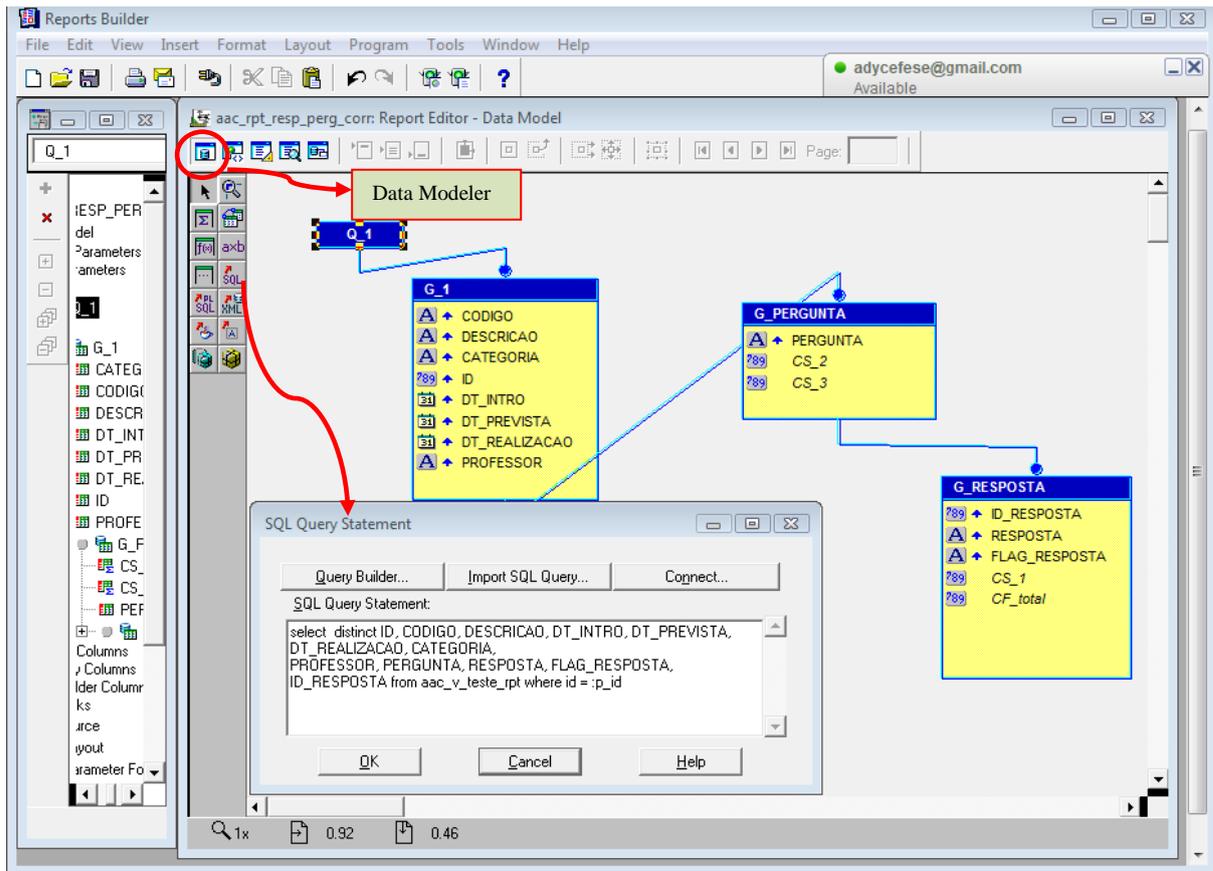


Figura 26 - Report Builder - vista Data Modeler

Quando abre a janela Data Modeler clique sobre o ícone **SQL** e abra a janela **SQL Query Statement** onde se realiza o query.

Ao terminar o report deve ser guardo numa pasta no servidor e depois de faz o mapeamento no para podermos ter acesso a esse reporta na aplicação..

Vincular um Reporte á aplicação

A figura seguinte mostra o código da aplicação que se encarega de explorar o report mostrado na figura anterior.

http.p(HTTP.anchor

```
(
'http://nosiodb01.gov.cv:7778/reports/rwservlet?aac_teste_sem_corr&p_id=' ||
aac_tipos.v_t_teste.id,
    http.img
    ('http://nosiodb01.gov.cv:7778/images/print2.gif',
     cattributes => 'alt= "AAC Teste sem correcção"
width=20 height=20 border=0 valign=absbottom')));
```


3. Conclusão

Os objectivos do estágio foram totalmente cumpridos graças a necessidade de concluir uma grande quantidade de tarefas num curto espaço de tempo, o que me obrigou a aprender depressa, tanto os conceitos de negócio inerentes ao projecto, como as tecnologias utilizadas na área.

A aplicação já foi submetida a algumas apresentações para avaliação do pessoal da AAC. Para os módulos já apresentados, os resultados foram satisfatórios, o que indica que o esforço tanto dos técnicos de NOSI como dos técnicos de AAC que fizeram o acompanhamento, valeram a pena.

Em termos profissionais o estágio foi de extrema utilidade, visto que após o estágio, passei a fazer parte da equipe de desenvolvimento do NOSI. Tive oportunidade de pedir transferência do departamento de Deployment (onde trabalho há 6 anos) para Departamento de Desenvolvimento. O pedido foi aceite graças aos resultados obtidos durante o estágio.

4. Trabalhos futuro

A aplicação, apesar de ter todos os módulos funcionais, existe ainda tarefas a serem cumpridas até a entrega do projecto à Instituição.

De entre as tarefas, as mais relevantes são:

- **Integração dos módulos** – apesar de terem sido desenvolvidas como módulos separados, a integração entre eles é o SRS (Surveillance and Reporting System). Através do SRS, é que devem ser chamadas todos os outros módulos. Ao registar uma aeronave ou elaborar um teste, primeiramente deve ser criado o registo de um SRS, com os seus dados específicos (data, status, actividades, etc) e só depois é que é feita a elaboração dos testes. Entretanto esse registo de SRS tem de ter uma ligação, com a elaboração daquele teste.

- **Requisições de Testes OnLine** - Normalmente são entregues formulários de requisição. Nesta altura se a pessoa ainda não estiver registada na BD da AAC, deverá ser feita o seu registo antes de se registar a requisição propriamente dita.

Deverá ser criado um mecanismo de comunicação on-line com a AAC em que um aluno pode fazer a inscrição de testes on-line. Essa comunicação pode ser feita através do Site da AAC ou através do portal Porton di Nos Ilha, desde que haja uma estrutura que suporta a ligação com a Base de dados.

- **Realização de teste no computador** - Esta funcionalidade visa a concepção de uma aplicação interactiva que permite que os testes sejam realizados num computador o que implica que na altura da inscrição de uma pessoa num teste seja gerado um código de acesso à aplicação de realização de teste. Código esse que permitirá a sua autenticação na aplicação.

Após autenticação a aplicação permite ao utilizador responder interactivamente às perguntas do teste determinado.

A utilização desta aplicação elimina a necessidade de registo dos resultados de cada pessoa, visto que este é feito de forma automática.

Bibliografia

http://eden.dei.uc.pt/~bizarro/files/manual_praticas_bd1_v3.pdf - acessado em 14_ setembro

<http://www.criarweb.com/artigos/230.php> - acessado em 14_ setembro

http://www.oracle.com/global/pt/pmes/documentation/oracledatabase_pt1.pdf - acessado em 15 -setembro

http://www.linuxmall.com.br/_product_files/6/5617/oracle.pdf - acessado em 10 - Setembro

http://www.oracle.com/global/pt/pmes/documentation/oracledatabase_pt1.pdf -

<http://www.camden.rutgers.edu/HELP/Documentation/Oracle/server.815/a67842.pdf> -
acessado em 28 setembro (user guides oracle)

<http://www.sunsoftware.com.br/solucoes/ambiente.desenvolvimento/plsql.pdf> - acessado em 28 setembro

[oracle/333299?s=38e3327e9457ecd1f20307faf5b76bd1&mp](http://www.oracle.com/technology/oracledatabase/333299?s=38e3327e9457ecd1f20307faf5b76bd1&mp); - acessado em 27 Outubro

PL/SQL User's Guide and Reference Release 8.1.5

<http://www.criarweb.com/artigos/10.php> - Introdução ao html – acessado em 27 Outubro

<http://www.criarweb.com/artigos/156.php> - Introdução à Javascript - acessado em 27 Outubro