Mestrado em Engenharia Informática

Estágio

Relatório Final

Desenvolvimento de sistema OLAP para análise de informação de gestão académica da UC

Inês Valente Domingues
ines@student.dei.uc.pt

Orientador:
Prof. Dr. Bruno Cabral

Data: 1 de Julho de 2014

FCTUC DEPARTAMENTO
DE ENGENHARIA INFORMÁTICA
FACULDADE DE CIÊNCIAS E TECNOLOGIA
UNIVERSIDADE DE COIMBRA

Resumo

Hoje em dia a quantidade de informação armazenada numa base de dados de uma empresa ou instituição tende a crescer exponencialmente. Devido às características e ao formato das bases de dados relacionais é difícil observar a vasta quantidade de informação de modo a obter uma ideia global da situação atual e da sua evolução. Esta lacuna tende a levar à recorrência de produtos de *business intelligence* que permitem a análise de elevadas quantidades de dados produzindo informação relevante sobre a situação da empresa ou instituição, auxiliando assim a tomada de decisões.

A Universidade de Coimbra também possui este problema. A reitoria, a administração, os coordenadores de curso e a comissão pedagógica necessitam de analisar indicadores de desempenho previamente definidos que muitas vezes são difíceis de obter e calcular.

O desenvolvimento da *data warehouse* para a Universidade de Coimbra vem solucionar esse problema, uma vez que os indicadores serão incorporados numa plataforma *web* e irão passar a estar disponíveis de forma simples e intuitiva. Para tornar isso possível, a informação necessária vai ser extraída da base de dados do NÓNIO, vão ser aplicadas transformações por forma a adaptar os dados para o preenchimento de vários cubos *OLAP (On-line Analytical Processing)*. Desse modo irá ser possível analisar a informação através de tabelas e gráficos interativos.

Palavras-Chave

Business Intelligence, Cubo, *Dashboard*, *Data Warehouse*, Indicadores de Performance (*KPI)*, Sucesso Escolar

Agradecimentos

À minha mãe e restante família, por tudo!

Ao Fábio Matos pelo apoio incondicional.

Ao Carlos Cortinhas pela ajuda e amizade.

Aos meus orientadores, Prof. Dr. Bruno Cabral e Eng. Pedro Pinto.

Aos meus colegas de estágio, Beatriz Fragoso e Hugo Costa.

Aos elementos do NÓNIO, em especial ao Eng. Marco Neves.

E a todas as pessoas que direta ou indiretamente contribuíram para este estágio.

Índice

Lista de Figuras

Lista de Tabelas

Glossário

ACID	*Atomicity, Consistency, Isolation, Durability*
DW	*Data Warehouse*
ETL	*Extraction, Transforming and Loading*
FCTUC	Faculdade de Ciências e Tecnlogia da Universidade de Coimbra
KPIs	*Key Performance Indicator*
MDX	*Multidimensional Expressions*
OLAP	*Online Analytical Processing*
OLTP	*Online Transaction Processing*
SO	Sistema Operativo
SQL	*Structured Query Language*
TIC	Tecnologias da Informação e Comunicação
UC	Universidade de Coimbra

1 Introdução

Este documento tem como objetivo apresentar o projeto desenvolvido no âmbito do Estágio/Dissertação do Mestrado em Engenharia Informática da Faculdade de Ciências e Tecnologia da Universidade de Coimbra no ano letivo 2013/2014.

Este trabalho, relativo ao sucesso escolar, enquadra-se no projeto DW-UC que visa desenvolver uma *data warehouse* para a Universidade de Coimbra. Esse projeto está inserido num projeto maior, o *UC em números para o apoio à decisão* que pertencente ao *SAMA* (Serviço de Apoio à Modernização Administrativa) cuja finalidade é a modernização e consecutiva melhoria dos serviços prestados na UC.

1.1 Contextualização

Devido ao constante crescimento da informação nas empresas e instituições, torna-se difícil analisar grandes quantidades de dados para ficar a par da situação atual e do histórico da entidade. Assim, surge a necessidade de soluções que consigam lidar com quantias enormes de dados e que originem informação significativa sobre o estado do negócio e respetiva evolução.

O projeto DW-UC visa a criação de um sistema que permita a monitorização de indicadores de desempenho (*KPIs*), muitos deles definidos no Plano Estratégico 2011-2015 da UC. A análise desses indicadores tem como finalidade avaliar a performance da universidade e, ao transmitir informação relevante, auxiliam a tomada de decisões.

No início deste estágio o projeto DW-UC já decorria há cerca de 9 meses e já se encontrava implementado um protótipo *online* com os custos e as receitas alusivas à investigação. Pretendia-se completar a plataforma com a adição de mais módulos entre os quais, os custos com o ensino, o sucesso escolar, a receita e os recursos humanos. Assim, foi formada uma equipa de trabalho composta por quatro elementos e cada um ficou responsável pelo desenvolvimento de um módulo adicional. A equipa foi orientada pelo Prof. Dr. Bruno Cabral e pelo Eng. Pedro Pinto.

1.2 Objetivos

Para alcançar o sucesso escolar a reitoria, a administração da UC, os coordenadores de cursos e a comissão pedagógica necessitam de indicadores muitas vezes difíceis de obter e de calcular. Assim, o objetivo deste estágio consiste no levantamento de alguns indicadores de desempenho (*KPIs*), na preparação da informação para responder aos mesmos e na sua disponibilização aos utilizadores finais de forma simples e imediata, através do acesso a uma plataforma *web* com tabelas e gráficos interativos.

Para criar a plataforma *web*, é importante estudar a área relativa ao sucesso escolar na universidade onde serão obtidos os indicadores de desempenho que devem constar na plataforma. Nesse estudo serão tidas em conta algumas das questões que esta área pretende responder, sendo essas questões as seguintes:

- Quais são as unidades curriculares com maior taxa de aprovação? E quais as unidades curriculares com menor taxa de sucesso escolar?

- A média de entrada influência o sucesso escolar?
- Quanto tempo leva um aluno, em média, a concluir um curso? E quantos conseguiram terminar no tempo estipulado?
- Em que cursos predomina o abandono escolar? E que tipo de abandono é mais frequente?
- Como é que as questões anteriores têm vindo a evoluir nos últimos anos letivos?

Após o levantamento dos *KPIs* e das questões, é necessário extrair a informação necessária para responder aos mesmos, efetuar um tratamento aos dados para os armazenar num sistema que permita a consulta e análise de grandes quantidades de informação com boa performance, ou seja, uma *data warehouse*. Por último, é essencial a criação de gráficos e tabelas com base nos dados armazenados na *data warehouse* e disponibilizar esses *dashboards* na plataforma *web*.

1.3 Noções gerais

Durante este relatório vão ser usados bastantes termos relacionados com *Data Warehouses* e *OLAP* (*On-Line Analytical Processing*). Serve este subcapítulo para explicar de forma breve o significado do vocabulário mais comum. Uma *data warehouse* é uma base de dados capaz de organizar grandes quantidades de informação de modo a facilitar o seu acesso e análise, viabilizando a tomada de decisões. O *ETL* (*Extract Transform Load*) que, como o nome indica, significa a extração dos dados da sua origem, a sua transformação através de várias operações e o seu carregamento para um destino onde voltarão a ser armazenados. *OLAP* é um tipo de aplicação que permite aceder à informação guardada na *data warehouse* para a visualizar e/ou analisar. No modelo *OLAP* a informação encontra-se organizada por cubos, uma vez que permite a existência de várias dimensões e granularidade.

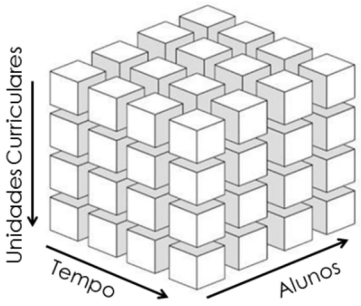

Figura 1 – Exemplo de um cubo de OLAP.

Na Figura 1, está representado um cubo com a informação dos alunos inscritos nas unidades curriculares, nos vários anos letivos. E, cada cubo pequeno representa uma classificação obtida por um aluno, numa unidade curricular, num determinado ano letivo.

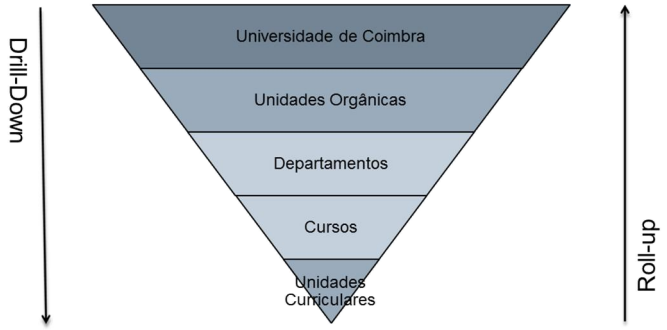

Figura 2 – Granularidade do módulo do sucesso escolar.

A granularidade indica o nível de detalhe dos dados, quanto maior for a granularidade mais genérica será a informação. Neste trabalho são usadas algumas formas de navegar na informação, o *drill-down* que significa descer na granularidade dos dados, ou seja, aumentar a especificação do dados. E o *roll-up* que significa subir níveis na granularidade, voltar atrás ou tornar os dados menos específicos. A granularidade usada neste projeto e a representação das formas de navegação encontram-se ilustradas na Figura 2.

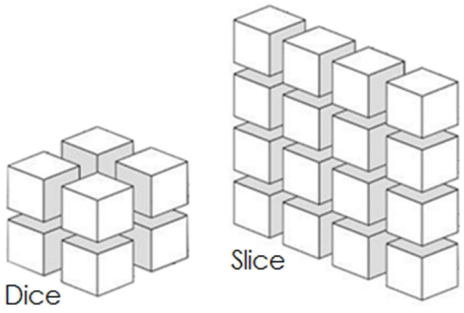

Figura 3 – Representação das operações *slice and dice*.

Também é possível fazer *slice and dice* aos cubos *OLAP* que significa partir a informação em partes mais pequenas para a analisar. O *slice*, como está representado no segundo conjunto de cubos da Figura 3, parte o cubo original num pedaço mais pequeno considerando apenas uma dimensão. Já o *dice*, o primeiro conjunto de cubos da Figura 3, divide o cubo num pedaço mais pequeno considerando várias dimensões.

1.4 Estrutura do relatório

Este relatório encontra-se dividido em dez capítulos. O primeiro, como é usual, trata-se da introdução ao trabalho a ser desenvolvido.

O capítulo seguinte introduz o projeto NÓNIO, a fonte dos dados e o que já se encontra implementado.

No terceiro capítulo são apresentados os requisitos funcionais e não funcionais para o desenvolvimento da ferramenta, a interface e o protótipo que foi criado.

O quarto capítulo é dedicado às questões relacionadas com a arquitetura do sistema, é apresentado um esquema geral, são analisadas as tecnologias e fundamentada as suas seleções, também será explicado o plano *ETL*, a origem da informação e o modelo de dados da *data warehouse*.

No quinto capítulo são descritos os passos mais importantes da implementação deste módulo do projeto.

O sexto capítulo possui a informação referente à validação da ferramenta e dos dados.

No sétimo capítulo é apresentada a planificação das tarefas que foram realizadas ao longo do ano letivo, separada por semestres.

E termina com a conclusão, os anexos e as referências.

2 Enquadramento

Nesta secção é dada a conhecer a fonte de onde serão retirados todos os dados, a base de dados do projeto NÓNIO. Também será analisado o que já se encontra desenvolvido pelo NONIO no separador Qualidade, quais desses indicadores serão recalculados e que mais-valias trarão ao que já existe implementado.

2.1 Projeto *NÓNIO*

O projeto NÓNIO surgiu em 2006 com o intuito de desenvolver uma aplicação para gerir os processos académicos e adequá-los ao processo de Bolonha. Entrou em funcionamento no ano letivo 2007/2008 apenas para a Faculdade de Ciências e Tecnologia (*FCTUC*) e atualmente já cobre toda a universidade, o que fez com que o sistema anterior, o WoC (*Web on Campus*) caísse em desuso. Em 2010/2011 evoluiu bastante com o desenvolvimento de funcionalidades de apoio pedagógico disponíveis para os docentes. Contínua em crescimento, com novas funcionalidades a serem desenvolvidas constantemente.

O projeto NÓNIO consiste numa plataforma *web* constituída por três vistas distintas, a Inforgestão para os serviços académicos, o Infordocente para os docentes e responsáveis de unidades orgânicas e o Inforestudante destinada aos alunos e candidatos da universidade.

O Inforgestão possui funcionalidades como a gestão e consulta de informações dos alunos, a gestão de pautas, emissão de certidões e declarações, gestão de pagamentos, gestão da rede UC, etc. O Infodocente permite a gestão de unidades curriculares, a consulta de dados pessoais e curriculares dos alunos, entre outras. O Inforestudante possibilita a realização de candidaturas, matrículas e inscrições, a consulta de informação relacionada com as propinas, a consulta de informação curricular, funcionalidades da rede UC, etc.

2.2 Separador "Qualidade"

Atualmente, nas plataformas do NÓNIO, Infordocente e Inforgestão, já existe um separador intitulado Qualidade que permite consultar numa tabela alguns índices de qualidade relativos ao sucesso escolar nas unidades curriculares. Ainda não existe forma de os observar graficamente nem para outros níveis de detalhe, por exemplo, não existe uma forma imediata de saber qual é a taxa de sucesso escolar de um curso. A imagem que se segue, mostra toda a informação que pode ser consultada nesse separador.

Figura 4 – Índices de qualidade que podem ser visualizadas numa tabela.

Todos os valores anteriores encontram-se pré calculados e armazenados na base de dados do NÓNIO. Podem ser recalculados a qualquer momento, por qualquer utilizador com acesso, embora essa operação possa vir a demorar vários minutos. Essa operação é lenta porque serão realizados vários cálculos, com uma elevada quantidade de dados, até obter os valores atualizados para os índices da Figura 4.

Neste trabalho, serão recalculados alguns desses índices de qualidade e serão representados em *dashboards* interativos compostos por gráficos e tabelas. Esses índices encontram-se listados na tabela seguinte e poderão ser consultados em segundos na plataforma *web*.

Tipo	Indicadores
Indicadores	Taxa de avaliados
	Taxa de aprovação
	Taxa de sucesso escolar
	Taxa de inscritos repetentes
	Taxa de avaliados repetentes
	Taxa de aprovados repetentes
	Taxa de sucesso escolar dos repetentes
	Taxa de inscritos pela 1ª vez
	Taxa de avaliados pela 1ª vez
	Taxa de aprovados de 1ª inscrição
	Taxa de sucesso escolar da 1ª inscrição
	Taxa de inscritos em melhoria
	Taxa de avaliados em melhoria
	Taxa de aprovados em melhoria
	Taxa de sucesso escolar das melhorias
Estatísticas	Média e desvio padrão global
	Média e desvio padrão dos repetentes
	Média e desvio padrão das 1ªs Inscrições
	Média e desvio padrão das melhorias

Tabela 1 – Índices de qualidade que estarão presentes neste projeto.

Será possível consultá-los segundo vários níveis de detalhe, ou seja, é possível ficar a saber qual é a taxa de sucesso escolar numa unidade orgânica ou a taxa de alunos inscritos em melhoria num curso. Vai ser ainda possível visualizar uma evolução temporal dos índices de qualidade, compará-los entre si e discriminar os resultados obtidos segundo alguns filtros

disponibilizados, como por exemplo, a média global para os alunos do género feminino inscritos num curso ou a taxa de aprovados de 1ª inscrição numa unidade orgânica para os alunos que tenham entrado nos cursos em primeira opção.

Devido à mudança das regras para efetuar melhoria de classificações obtidas nas unidades curriculares, estas passaram a ser registadas de forma diferente. Desde o ano letivo de 2013/2014 um aluno inscrito em melhoria numa unidade curricular tem que se inscrever e frequentar a unidade curricular como um aluno normal. Anteriormente as melhorias eram contabilizadas como inscrições em exames. Outra vantagem de recalcular alguns dos indicadores do separador qualidade é que este ainda não está preparado para calcular os índices de melhoria de acordo com as novas regras, já a plataforma *web* estará. É esse o motivo pelo qual só existem valores para os indicadores de melhoria a partir do ano letivo 2012/2013.

3 Requisitos

Neste capítulo serão apresentados e sucintamente descritos todos os requisitos relativos ao sucesso escolar que fazem parte da plataforma. Para um conhecimento mais aprofundado deverá ser consultado o documento de requisitos que se encontra em anexo (Anexo [3]).

Os requisitos foram levantados através da leitura de documentação e de reuniões presenciais com a administração da *UC*, representada pela Dra. Conceição Costa e com vários coordenadores de cursos, entre os quais a Prof.ª Dr.ª Bernardete Ribeiro, o Prof. Dr. Carlos Fonseca, o Prof. Dr. Amílcar Cardoso, a Prof.ª Dr.ª Teresa Tavares e o Prof. Dr. Rui Gama. À medida que os requisitos iam ficando claros, foi produzido um protótipo rápido que auxiliou na transmissão dos requisitos e das ideias-chave deste projeto. No final, os requisitos foram validados pela Dr.ª Conceição Costa e pela vice-reitora Prof.ª Dr.ª Margarida Mano representada pelo chefe da Divisão de Planeamento, Gestão e Desenvolvimento Filipe Rocha e pelo consultor externo Dr. José Morais.

Um requisito é uma condição ou uma capacidade que um sistema deve possuir. Existem dois tipos de requisitos, os funcionais e os não funcionais. Os requisitos funcionais indicam o que o sistema deve fazer. Os requisitos não funcionais referem as qualidades do sistema e não as suas funções.

Tanto este capítulo como o documento de requisitos foram criados tendo em conta o modelo "*FURPS+*" que é um dos modelos mais usados para representar requisitos funcionais e não funcionais do *software*. O acrónimo "*FURPS*" engloba as funcionalidades, a usabilidade, a confiabilidade (*reliability* em inglês), performance e suporte. O carácter '+' possibilita acrescentar necessidades adicionais como restrições de *design* e requisitos de implementação, de *design* e físicos.

Foram separados os requisitos funcionais dos não funcionais e identificados da seguinte forma:

Identificação	Descrição
RF_XX_00	RF indica que é um requisito funcional, o XX refere-se à secção onde o requisito se enquadra, pode ser geral (GE), relativo ao sucesso nas avaliações (SA), às conclusões de curso (CC), ao abandono escolar (AE) ou às agregações (AG). E o 00 representa a identificação numérica do requisito.
RNF_XX_00	RNF designa um requisito não funcional. XX representa a categoria do requisito não funcional que poderá ser: US de usabilidade, CO de confiabilidade, PE de performance, SU de suporte e OU de outros. O 00, mais uma vez, é a identificação numérica do requisito.

Tabela 2 – Forma de identificação dos requisitos.

A prioridade representa a importância de um requisito na elaboração de um projeto. É um aspeto importante aquando o seu desenvolvimento uma vez que dá a conhecer os aspetos essenciais e que deverão ser implementados primeiro. Neste caso foram adotadas as seguintes denotações: elevada, média e baixa.

Prioridade	Definição
Elevada	É um requisito imprescindível no projeto que deverá ser obrigatoriamente implementado. Devem ser os primeiros a ser

	desenvolvidos, dada a sua grande importância.
Média	Não é um requisito fundamental ao sistema. A sua inexistência deixa o projeto funcional, de forma satisfatória, embora que incompleto.
Baixa	Requisitos que sem os quais o bom desempenho do projeto não é afetado. São requisitos que acrescentam mais-valias ao sistema e caso não exista tempo para os implementar, podem servir de recomendação para versões posteriores.

Tabela 3 – Prioridades que podem ser atribuídas aos requisitos.

3.1 Requisitos funcionais

Os requisitos funcionais estão divididos em várias categorias, os gerais que se referem a aspetos comuns da plataforma. Os de sucesso nas avaliações, conclusões de curso e abandono escolar que apresentam os requisitos para cada uma das secções. E, por último, os de agregação que mostram os tipos de discriminações que podem ser aplicadas nos dados.

Gerais

Funcionalidades ou características que a plataforma deve conter, independente do subtema do módulo. Ou seja, estes requisitos são comuns a todos os módulos (sucesso escolar, recursos humanos, económico-financeiros ou de investigação).

Identificação	Requisito	Prioridade	Descrição
RF_GE_01	Autenticação	Elevada	Para aceder à plataforma deverá ser efetuado um *login* com as credenciais da conta da universidade. Devem existir autenticações por módulo, ou seja, será definido um grupo de pessoas que terá acesso a cada um dos módulos.
RF_GE_02	Fechar sessão	Elevada	Ao estar autenticado na plataforma, o utilizador pode efetuar o *logout*.
RF_GE_03	Término de sessão	Elevado	Por razões de segurança, a sessão será terminada automaticamente ao fim de algum tempo.
RF_GE_04	Navegação entre os módulos	Média	Deverá ser possível navegar entre os diversos módulos que constituem a plataforma (Investigação, Académicos, Recursos Humanos, e Económico-Financeiros).
RF_GE_05	Navegação interna	Elevada	Deverá ser possível navegar na granularidade, ou seja, fazer o *drill down* e o *roll up* ao longo dos vários níveis. Descer na granularidade deve ser interativo e ao fazê-lo deverá ser deixado um rasto para que o utilizador possa facilmente regressar a um nível anterior para consultar dados que já passaram sem ter que recomeçar do início. Os níveis de granularidade são: Universidade de Coimbra; Unidades Orgânicas; Departamentos; Cursos e Unidades Curriculares.
RF_GE_06	Parâmetros	Elevada	Os parâmetros gerais modificam os

	gerais		indicadores que são apresentados para análise. Esses parâmetros manipulam os dados representados nos gráficos, podendo simplificar a análise ou torná-la mais completa.
RF_GE_07	Parâmetros de tempo	Elevada	Modificam o intervalo de tempo dos indicadores representados, podem ser alternados entre anos letivos, semestres ou trimestres.
RF_GE_08	Esconder parâmetros	Baixa	Deve ser permitido ao utilizador esconder a barra lateral onde estão os parâmetros gerais e os de tempo.
RF_GE_09	Secção de ajuda	Elevada	É importante que exista uma secção na plataforma que o utilizador possa consultar sempre que tiver alguma dúvida sobre o funcionamento da mesma. Esta área tem como funcionalidade explicar o funcionamento do sistema bem como esclarecer todas as dúvidas comuns que possam surgir.
RF_GE_10	Informação auxiliar	Elevada	É permitido ao utilizador consultar mais informação sobre os dados que estão a ser visualizados.
RF_GE_11	Visualização: Gráfico ↔ Tabela	Elevada	Inicialmente, os dados aprecem sobre a forma de gráficos mas através de um botão é possível visualiza-los em forma de tabela.
RF_GE_12	Exportar informação da tabela	Baixa	A informação representada nas tabelas pode ser armazenada num ficheiro.
RF_GE_13	Ordenação e filtragem	Média	Deve ser possível ordenar os dados por um indicador de forma ascendente ou descendente e aplicar uma filtragem para mostrar 3, 5, 10 ou todos os valores. Esta funcionalidade só estará disponível no módulo do sucesso escolar.

Tabela 4 – Requisitos gerais que são comuns a todas as áreas da plataforma.

Sucesso nas avaliações

A área do sucesso nas avaliações vai representar através de gráficos e tabelas as taxas, médias e desvios padrões que se encontram no separador *Qualidade* do Nónio. Esses requisitos encontram-se especificados na Tabela 5. Para cada um dos seguintes requisitos é possível navegar na granularidade que, neste caso começa na Universidade de Coimbra sendo a mais elevada e vai descendo até unidades curriculares, como explicado no requisito *RF_GE_05*.

Identificação	Requisito	Prioridade	Descrição
RF_SA_01	Taxa de avaliados	Elevada	Percentagem de alunos que estando inscritos numa unidade curricular participaram na avaliação da mesma. Esta taxa é dada através do quociente do número de alunos avaliados pelo número de alunos inscritos.

RF_SA_02	Taxa de aprovação	Elevada	Percentagem de alunos que estando inscritos numa unidade curricular, obtiveram aprovação na mesma. Este valor é calculado através do quociente dos alunos aprovados pelos alunos inscritos.
RF_SA_03	Taxa de sucesso escolar	Elevada	Percentagem de alunos que ao terem participado na avaliação de uma determinada unidade curricular obtiveram aprovação na mesma. Esta taxa é calculada pelo quociente dos alunos aprovados pelos alunos avaliados.
RF_ SA _04	Taxa de avaliados repetentes	Elevada	Igual à taxa de avaliados mas tendo em conta apenas os alunos repetentes. Deve ser calculada pelo quociente do total de alunos repetentes avaliados pelo total de alunos repetentes inscritos na unidade curricular.
RF_ SA _05	Taxa de aprovação dos repetentes	Elevada	Percentagem de alunos repentes que obtiveram aprovação na unidade curricular. Mais uma vez, o cálculo faz-se da mesma forma que a taxa de aprovação global mas tendo em conta os alunos repetentes.
RF_ SA _06	Taxa de sucesso escolar dos repetentes	Elevada	Igual à taxa de sucesso escolar mas considerando apenas os alunos que possuem duas ou mais inscrições na mesma unidade curricular. Ou seja, é calculada através do quociente do total de alunos repetentes aprovados pelo total de alunos repetentes avaliados.
RF_ SA _07	Taxa de inscritos repetentes	Elevada	Percentagem de alunos repetentes que estão a frequentar uma unidade curricular. Este valor é dado pelo quociente do número de alunos repetentes inscritos a dividir pelo total de alunos inscritos.
RF_ SA _08	Taxa de avaliados das 1ªs inscrições	Elevada	Percentagem de alunos, inscritos pela primeira, que foram avaliados. É obtido pelo quociente do total de avaliados em 1ª inscrição pelo total de inscritos pela primeira vez.
RF_ SA _09	Taxa de aprovação das 1ªs inscrições	Elevada	Igual à taxa de aprovação mas tendo em conta apenas os alunos inscritos pela primeira vez. É calculada pelo quociente do total de alunos de 1ª inscrição aprovados pelo total de alunos inscritos pela 1ª vez.
RF_ SA _10	Taxa de sucesso escolar das 1ªs inscrições	Elevada	Igual à taxa de sucesso escolar mas tendo em conta alunos que estão inscritos pela primeira vez. É determinada pelo quociente do número de alunos aprovados que se tenham inscrito pela primeira vez pelos alunos avaliados inscritos pela 1ª vez.
RF_ SA _11	Taxa de inscritos de 1ª inscrição	Elevada	Percentagem de alunos que estão inscritos pela primeira vez numa unidade curricular, consiste na divisão do número de alunos

			inscritos pela 1ª vez pelo total de alunos inscritos.
RF_ SA _12	Taxa de avaliados em melhoria	Elevada	As taxas de melhoria também se calculam da mesma forma que as anteriores, mas tendo em consideração apenas os alunos que se inscreveram para fazer melhoria numa determinada unidade curricular. Este valor é dado pelo quociente dos alunos inscritos em melhoria que foram avaliados pelo total de alunos em melhoria inscritos na unidade curricular.
RF_ SA _13	Taxa de aprovados em melhoria	Elevada	Valor relativo obtido através do resultado da divisão do total de alunos aprovados inscritos em melhoria pelo total de alunos inscritos em melhoria.
RF_ SA _14	Taxa de sucesso escolar das melhorias	Elevada	A taxa de sucesso das melhorias é obtida pelo quociente do número de alunos aprovados inscritos em melhoria pelo número de alunos avaliados em melhoria numa unidade curricular.
RF_SA_15	Taxa de inscritos em melhoria	Elevada	Calcula o valor relativo de alunos que se inscreveram para fazer melhoria de uma nota. É a divisão do número de alunos inscritos em melhoria pelo total de alunos inscritos.
RF_ SA _16	Média e desvio padrão globais	Elevada	Média e desvio padrão obtidos na(s) unidade(s) curricular(es).
RF_ SA _17	Média e desvio padrão dos repetentes	Elevada	O mesmo que a média e o desvio padrão globais, mas considerando apenas as classificações obtidas pelos alunos repetentes.
RF_ SA _18	Média e desvio padrão das 1ªs inscrições	Elevada	Média e desvio padrão considerando apenas as notas obtidas pelos alunos que possuem apenas uma inscrição na unidade orgânica.
RF_ SA _19	Média e desvio padrão das melhorias	Elevada	Média e desvio padrão das melhorias efetuadas.

Tabela 5 – Requisitos funcionais do sucesso nas avaliações.

Conclusões de curso

Esta área analisa as questões relacionadas com as conclusões de curso na universidade. A granularidade para cada um dos requisitos apresentados na tabela seguinte vai desde Universidade de Coimbra, Unidades Orgânicas, podendo passar por departamentos caso existam e termina no nível dos cursos. Neste caso não faz sentido existir o nível das unidades curriculares.

Identificação	Requisito	Prioridade	Descrição
RF_CC_01	Conclusões de curso no	Elevada	Valor relativo que representa os alunos que concluíram o curso na duração prevista do

	tempo estipulado		mesmo.
RF_CC_02	Média de acesso	Elevada	Representa a média de acesso ao curso ou a média da formação anterior do aluno. Se estivermos a consultar uma licenciatura ou mestrado integrado, tem em conta a média das notas de entrada na universidade pelo concurso nacional de acesso ao ensino superior (alunos DGES).
RF_CC_03	Média de conclusão	Elevada	Média das notas de conclusão de um curso.
RF_CC_04	Estudantes finalistas	Elevada	Valor relativo que representa os alunos que possuem a situação especial de aluno finalista num determinado ano letivo.
RF_CC_05	Estudantes que concluíram o curso	Elevada	Para um ano letivo, é calculado a percentagem de alunos que concluíram o curso, tendo por base todos os alunos que se encontravam inscritos, de modo a comparar com a percentagem de alunos finalistas.
RF_CC_06	Número de anos de conclusão	Elevada	Deve ser visível a percentagem de alunos que demoraram mais do que o tempo estipulado para a conclusão de um curso, repartida em intervalos de n+1, n+2 e mais de 2 anos.

Tabela 6 – Requisitos funcionais das conclusões de curso.

Abandono escolar

Na secção relativa ao abandono escolar devem ser representados os requisitos descritos na Tabela 7. No caso do abandono a granularidade vai apenas de Universidade de Coimbra a cursos pois, no âmbito deste trabalho, não tem lógica representar o abandono escolar nas unidades curriculares.

Identificação	Requisito	Prioridade	Descrição
RF_AE_01	Abandono efetivo	Elevada	Deve ser possível visualizar o valor relativo de alunos que abandonaram a universidade, ou seja, aqueles que estiveram inscritos no ano letivo anterior e tendo o curso incompleto não se voltaram a inscrever na universidade.
RF_AE_02	Abandono interno	Elevada	Será possível observar a percentagem de alunos que mudaram de curso, ou seja, o valor relativo de alunos que estiveram inscritos num curso num ano letivo e tendo esse curso incompleto num ano letivo posterior inscreveram-se noutro curso da mesma universidade.
RF_AE_03	Abandono total	Elevada	A fórmula do abandono total em todos os níveis de granularidade com a exceção do último é a seguinte: Total de alunos inscritos no ano letivo n – total de alunos inscritos no ano letivo (n+1) + total de novos alunos no

			ano letivo (n+1) – o total de conclusões no ano letivo n. No nível dos cursos essa fórmula passa a representar o abandono efetivo e nesse nível o abandono total é a soma dos dois abandonos.
RF_AE_04	Estudantes que regressaram	Média	Também é interessante que seja possível saber o valor absoluto dos alunos que apesar de terem abandonado a universidade voltaram num ano letivo posterior.
RF_AE_05	Duração média da interrupção	Média	Para os alunos que interromperam a matricula num ano mas acabaram por voltar a inscrever-se na universidade anos letivos mais tarde, deverá ser ainda representado a duração média da interrupção, ou seja, a quantidade média de anos que um aluno esteve sem se inscrever na faculdade.

Tabela 7 – Requisitos funcionais do abandono escolar.

Agregações

É possível agrupar qualquer um dos indicadores expostos anteriormente em intervalos de dados previamente definidos. Ao serem aplicadas, o gráfico é divido em várias barras e cada uma das novas barras corresponde a uma das partições da agregação. Se os dados estiverem a ser apresentados em tabelas, são adicionadas novas colunas com os valores correspondentes a esses intervalos.

Identificação	Requisito	Prioridade	Descrição
RF_AG_01	Idades	Elevada	Discrimina os valores em quatro intervalos de idades, sendo esses intervalos os seguintes: até 20 anos, 20-23 anos, 24-27 anos e mais de 28 anos.
RF_AG_02	Género	Elevada	Permite distinguir entre os dois géneros.
RF_AG_03	Área de estudos	Elevada	Permite a discriminação dos indicadores pela área de estudos que o aluno frequentou anteriormente.
RF_AG_04	Nacionalidade	Elevada	Deve ser possível saber qual o país de origem dos alunos.
RF_AG_05	Escolaridade dos pais	Elevada	Agrupa os dados de acordo com a escolaridade dos pais do aluno que poderão ser as seguintes: superior, secundário, básico 3, básico 2, básico 1 ou analfabeto.
RF_AG_06	Situação profissional dos pais	Elevada	Une os valores dos alunos cuja situação profissional dos pais seja a mesma. Existem as seguintes opções: empregados, desempregados, reformados, outra situação ou desconhecida.
RF_AG_07	Tipo de matrícula	Elevada	Faz distinção entre os alunos de mobilidade e os restantes.
RF_AG_08	Opção de procura do curso	Elevada	Distingue entre os alunos que colocaram o curso atual em primeira opção dos restantes.

RF_AG_09	Situação especial	Elevada	Discrimina os dados de acordo com as situações especiais atribuídas ao aluno.
RF_AG_10	Modo de frequência	Elevada	Diferencia os alunos que estão inscritos em modo integral dos que estão em modo parcial.
RF_AG_11	Ciclo	Elevada	Faz distinção entre o ciclo do curso e pode possuir as seguintes categorias: 1º ciclo, 2º ciclo, 3º ciclo e cursos não conferentes de grau.
RF_AG_12	Grau	Elevada	Esta agregação discrimina os dados pelos graus dos cursos, como por exemplo, Licenciatura, Mestrado (Continuidade), etc.
RF_AG_13	Ano curricular	Média	O utilizador pode agrupar por anos curriculares, ou seja, 1º ano, 2º ano, 4º ano, 5º ano de um determinado curso ou formação.
RF_AG_14	Período letivo	Elevada	Distingue entre os vários períodos letivos. Esta agregação só está disponível no sucesso nas avaliações.

Tabela 8- Requisitos funcionais das agregações.

3.2 Requisitos não funcionais

Os requisitos não funcionais estão inseridos na última parte do acrónimo "*FURPS*" e significam Usabilidade (U), Confiabilidade (R de *reliability* em inglês), Performance (P) e Suporte (S). A Usabilidade tem em conta os fatores humanos, as características estéticas, a consistência na interface, a documentação para o utilizador e os materiais de treino. A fiabilidade tem em consideração características como a disponibilidade do sistema e a capacidade do sistema recuperar de falhas. A performance preocupa-se com o desempenho, o *throughput*, o tempo de resposta, o tempo de recuperação, o tempo de inicialização e o tempo de encerramento. O Suporte engloba características como a manutenção, a compatibilidade, a configurabilidade, a instabilidade, a escalabilidade, etc.

Os tópicos seguintes apresentam e descrevem sucintamente requisitos para algumas dessas categorias de requisitos não funcionais.

Suporte

Identificação	Requisito	Descrição
RNF_AG_01	Atualização dos dados	O processo de *ETL* e o carregamento dos dados para a *Data Warehouse* e para o cubo *OLAP* devem ser automáticos.
RNF_AG_02	Compatibilidade do *browser*	A aplicação é compatível com as seguintes versões dos *browsers* ou superiores: *Internet Explorer 9, Firefox 20, Chrome 35 e Safari 6*.
RNF_AG_03	Compatibilidade do *SO*	É compatível com *Windows* 7 e com as distribuições de *Linux*.
RNF_AG_04	Licenças	Todo o *software* usado é gratuito.

Tabela 9 – Requisitos não funcionais de suporte.

Outros

Outros tipos de requisitos não funcionais que não se enquadram nas categorias anteriores.

Identificação	Requisito	Descrição
RNF_AG_01	*Hardware*	A máquina deverá ter as características mínimas: 4*Gb* de *RAM*, *250Gb* de espaço em disco e um processador dual core. Não é necessário que seja um ambiente de 64 bits.

Tabela 10 – Outros requisitos não funcionais.

3.3 Outros requisitos

Durante o período inicial de reuniões e leitura de documentação foram encontrados outros indicadores que faziam sentido pertencerem à vertente do sucesso escolar. Essas ideias tiveram que ser postas de parte pois não iria haver tempo suficiente para implementar um trabalho tão extenso e complexo durante o ano letivo. No entanto, ficam as ideias para futuras atualizações da plataforma.

● **Prescrições:** A ideia consiste em analisar as prescrições ao longo dos anos letivos tendo em conta as regras em vigor nos mesmos. Pretendia-se com esta análise descobrir quais as unidades orgânicas e os cursos em que existiam mais prescrições, as características que tinham em comum os alunos que prescreviam e em que anos de escolaridade havia mais tendência a prescrever.

● **Assiduidade:** Com os dados de assiduidade que são introduzidos no nónio, era suposto fazer uma secção onde fosse possível acompanhar as presenças nas aulas das várias unidades curriculares. Permitindo assim uma melhor análise sobre a assiduidade, descobrindo em tempo-real quais as unidades curriculares a que os alunos faltam mais, quais as aulas que tinham mais assistência até ao final, quais as alturas em que os alunos deixam de frequentar as aulas e as suas causas. A não comparência dos alunos nas aulas poderá indicar problemas, descontentamento ou sobrecarga de trabalho.

● **Satisfação dos alunos:** Com os resultados obtidos nos inquéritos pedagógicos realizados no fim de cada período de tempo pretendia-se analisar a opinião dos alunos de modo a combater a insatisfação e o insucesso escolar. Embora já existam gráficos que representam as respostas obtidos nos inquéritos pedagógicos, a intenção seria fazer uma análise mais detalhada.

● **Reconhecimento internacional:** Este *KPI* pode passar pela análise dos alunos *Erasmus*, dos investigadores e dos professores estrangeiros ligados à Universidade de Coimbra.

● **Empregabilidade:** Uma das formas de avaliar o sucesso escolar da universidade pode passar por analisar a situação profissional ou académica do aluno após a conclusão de um curso. Sendo assim, um ano após a conclusão de um curso é pedido ao aluno a elaboração de um inquérito sobre empregabilidade. Seria interessante a análise das respostas dadas aos inquéritos de empregabilidade por forma a tirar conclusões e a comprar com a situação económica atual.

3.4 Interface

Como foi explicado anteriormente, o "+" no acrónimo "*FURPS"* possibilita acrescentar necessidades adicionais relacionadas com o aspeto do sistema. Foram criados dois

documentos de suporte à interface, um documento de *design* e um documento de especificação do protótipo, ambos em anexo (Anexos [2] e [4]). O primeiro especifica os aspetos de *design* comuns a todos os módulos da plataforma e o último explica com elevado detalhe o aspeto e as funcionalidades do protótipo rápido que foi elaborado.

Foi elaborado um protótipo rápido com o cuidado de ficar o mais próximo possível do resultado final e ao mesmo tempo coerente com os protótipos dos restantes elementos da equipa. Esse protótipo é interativo e possibilita a navegação nos vários módulos e ao longo de todos os requisitos funcionais. Foi uma ferramenta fulcral para a validação dos requisitos, uma vez que permitiu às entidades competentes ficarem com uma ideia exata de como iria ficar a plataforma *web* no final do projeto.

Figura 5 – *Dashboard* do sucesso nas avaliações.

A Figura 5 mostra o ecrã do sucesso nas avaliações para as três unidades orgânicas. É possível observar as taxas de avaliados, de aprovação e de sucesso escolar no primeiro gráfico da direita e as médias e desvios padrões globais no último gráfico. À esquerda encontram-se gráficos de evolução temporal para os mesmos indicadores, nos últimos três anos letivos. Os restantes ecrãs do protótipo e as suas funcionalidades podem ser consultadas no documento de especificação do protótipo que se encontra em anexo.

4 Arquitetura

A elaboração da arquitetura de um *software* é uma etapa crucial para o sucesso de um projeto uma vez que permite poupar tempo e recursos durante a implementação. Nesta secção será estudado o diagrama com a arquitetura geral do sistema, analisadas e selecionadas as ferramentas, elaborado o plano *ETL*, mencionado de onde provêm os dados e, por fim, explicado o modelo de dados.

4.1 Arquitetura geral

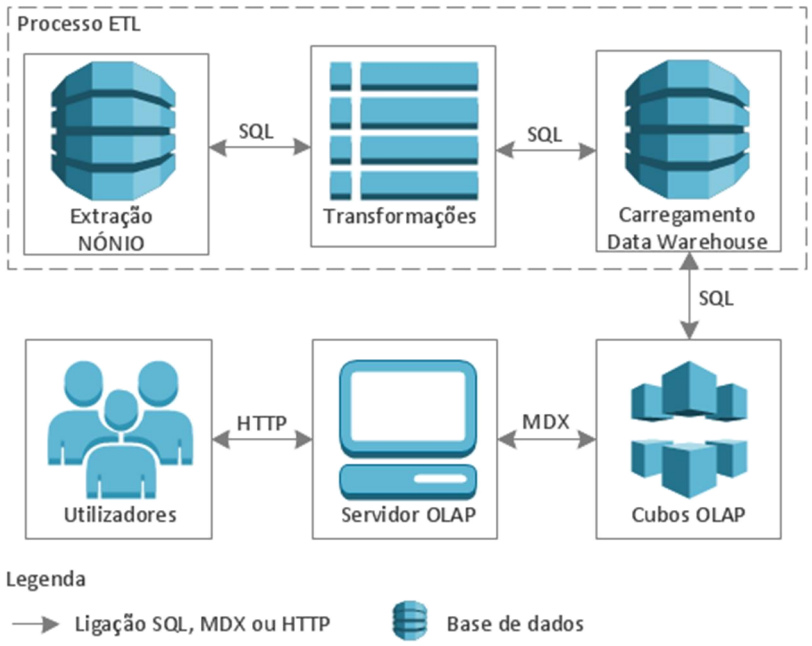

Figura 6 – Arquitetura geral do projeto.

A Figura 6 mostra a arquitetura do sistema que irá ser desenvolvido. A primeira fase é a do processo *ETL*, ou seja a extração, o tratamento e o carregamento dos dados para a *data warehouse*. Segue-se a criação de vários cubos *OLAP* e a elaboração de *dashboards* num servidor *OLAP* que serão acedidos pelos utilizadores finais através de uma plataforma *web*.

Durante o processo de *ETL*, todos os dados são extraídos da base de dados do NÓNIO. Foram criadas vistas de dados com toda a informação que irá ser necessária para integrar na plataforma e esses dados serão extraídos para a área temporária e armazenados em tabelas de modo a que a informação fique pronta para ser tratada. Na área temporária sofre vários processos que se encontram especificados no capítulo 4.3. Após as transformações, os dados são carregados para a *data warehouse*.

No fim do preenchimento da *DW*, são criados vários cubos *OLAP*. O cubo é uma estrutura de dados multidimensional que armazena informação de várias categorias e por vários níveis, facilitando o acesso aos dados através dos conceitos de *drill-down*, *roll-up* e *slice and dice*, explicados nas noções gerais da introdução. A utilização de um cubo também torna o acesso aos dados mais rápido.

Por último, são criados vários *dashboards* com tabelas e gráficos interativos que serão disponibilizados *online* para os utilizadores finais.

Como já foi referido atrás, o módulo do sucesso escolar faz parte de um projeto maior, composto por vários módulos. Embora as etapas da arquitetura geral sejam iguais para todos, cada módulo possui um tema diferente e por isso o autor de cada módulo é responsável pela elaboração de todas as componentes do seu projeto.

4.2 Análise das Tecnologias

As tecnologias mais adequadas para a elaboração do projeto já se encontravam escolhidas, no entanto foi efetuado um levantamento e análise das ferramentas existentes no mercado. Esse estudo deu a conhecer não só as características e a forma de funcionamento das ferramentas selecionadas mas também das suas principais concorrentes.

Os critérios de seleção consistiam em escolher as ferramentas mais adequadas às nossas necessidades e, se possível, dar preferência às *open-source*. Neste subcapítulo é apenas efetuado um breve resumo das conclusões obtidas aquando a análise das ferramentas, no entanto o estudo completo encontra-se no Anexo [1].

4.2.1 Base de dados

A base de dados possui um papel muito importante neste projeto uma vez que será o local onde serão armazenados todos os dados para serem tratados e analisados. Por esse motivo foram estudados os servidores de base de dados mais usados no mercado, o *MySQL*, o *Oracle*, o *SQL Server* e o *PostgreSQL*. Para além dos limites máximos que as bases de dados suportavam foi verificado também se possuíam características como *ACID*, o suporte de vistas materializadas, o suporte de *triggers*, a escalabilidade, etc. Nesta análise o *Oracle* ficou bem classificado, no entanto o *PostgresSQL* também se destacou e tinha a grande vantagem de ser gratuito.

4.2.2 *ETL* (Extração, transformação e carregamento)

As ferramentas de *ETL* são uma ajuda valiosa no tratamento e limpeza de dados provenientes de várias fontes. A escolha da ferramenta deve ser cuidadosa pois esta deverá poupar imenso tempo na uniformização da informação que será guardada na base de dados.

Foram estudadas as seguintes ferramentas: *Adeptia, CloverETL, IBM DataStage, Informatica PowerCenter, Oracle, Pentaho Kettle* e *Talend*. E foram analisadas características como o suporte de vários formatos de entrada dos dados, a existência de transformações complexas, a possibilidade de programação de novos componentes, a sua usabilidade, entre outras. Com essa pesquisa percebeu-se que as ferramentas eram todas equivalentes e que possuíam um funcionamento muito semelhante, à base do arrastar e interligar componentes que executam transformações.

4.2.3 *OLAP* (*Online Analytical Processing*)

O servidor *OLAP* permite o acesso e a análise dos dados auxiliando a gestão e a tomada de decisões. Foram analisadas as ferramentas mais conhecidas como o *Pentaho*, o *IBM Cognos*, o *Oracle* e o *icCube* quanto a características como a usabilidade, a criação de *dashboards*, a possibilidade de navegação na granularidade, a exportação de relatórios, entre outras. Nesse estudo o *Pentaho BI Server* foi claramente o melhor classificado, especialmente porque possuía a vantagem de bastante completo e gratuito.

4.3 Escolha das tecnologias

Após a análise das tecnologias encontram-se reunidas as condições para passar à seleção das mesmas. A figura seguinte mostra todas as ferramentas envolvidas no projeto.

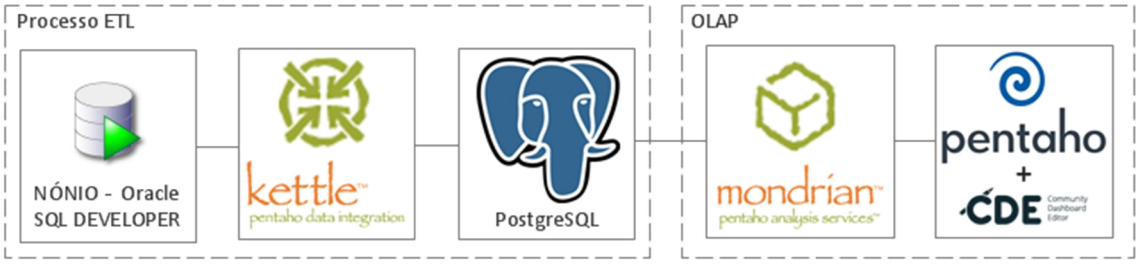

Figura 7 – Tecnologias usadas no projeto DW-UC.

Como foi referido anteriormente, a informação é extraída das vistas materializadas criadas pelo NÓNIO no ambiente *Oracle SQL DEVELOPER*. Esta ferramenta não foi escolhida para ser usada no projeto, trata-se do servidor de base de dados usado pelo NÓNIO e é apenas mencionada neste capitulo porque a fase de *ETL* interage com a mesma.

Para a fase de *ETL* foi escolhido o *Pentaho Data Integration* (*Kettle*) que obteve uma boa avaliação na análise anterior e preenche os requisitos de ser completo, fácil de utilizar e gratuito.

Para a base de dados foi escolhido o *PostgreSQL* uma vez que é a base de dados *open-source* mais completa e personalizável que se encontra disponível atualmente. O *PostgreSQL* irá armazenar a informação da área temporária e toda a *data warehouse*.

Quanto à fase *OLAP*, serão usadas também ferramentas do *Pentaho*. Foi escolhido o *Mondrian* que permite a criação de cubos, o que otimiza as consultas aos dados. Já os *dashboards* da componente *web* serão desenvolvidos no *Pentaho BI Server* com o recurso ao *plugin CDE* (*Community Dashboard Editor*). O *CDE* permite a criação de *dashboards* e suporta tecnologias *web* como *HTML*, *CSS*, *JavaScript*, etc. As ferramentas da *Pentaho* para além de completas, tem a grande vantagem de também serem *open-source*.

4.4 Plano ETL

Embora a parte tangível da *data warehouse* seja a informação que se encontra armazenada nos cubos, uma *data warehouse* é muito mais do que apenas dados representados em gráficos, tabelas, relatórios ou *dashboards*. Segundo Kimball[2], 70% do esforço de criação de uma *data warehouse* pertence ao processo de *ETL*. Para que a informação chegue com qualidade ao utilizador necessita de passar por vários processos onde sofre várias transformações. Esses processos embora não sejam visíveis pelo utilizador final não devem ser menosprezados pois são de extrema importância uma vez que melhoram a qualidade e consistência dos dados.

Esta etapa merece especial cuidado devido aos inúmeros problemas que podem surgir. Os dados tendem a vir sempre pior do que o que é esperado, são despendidos dias no seu tratamento e a qualidade pode ficar longe da idealizada. Muitos dos dados podem conter lacunas, campos e informações importantes podem vir por preencher o que condena o resultado final. Para além dos erros que podem ocorrer durante a criação e a execução do processo de *ETL*, a qualidade final dos dados não é garantida.

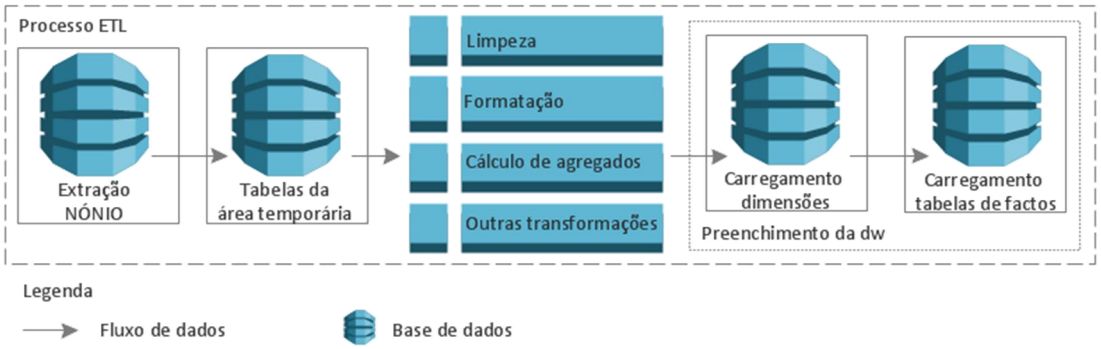

Figura 8 – Plano de ETL do projeto.

Na Figura 8 está representado um esquema que descreve os vários passos do plano de *ETL* deste projeto. Uma vez extraída a informação da base de dados, neste caso do sistema NÓNIO, os dados passam para a denominada área temporária. Existem duas formas de armazenar os dados na área temporária, estes podem permanecer em memória ou podem ser guardados num lugar físico. Ambos possuem as suas vantagens e desvantagens, trabalhar com os dados em memória torna mais rápido o processo *ETL*. Porém, guardar os dados em disco é mais seguro no caso de ocorrência de falhas. Neste projeto os dados são guardados em tabelas e o modelo de dados da área temporária encontra-se explicado no subcapítulo seguinte.

Dentro da área temporária são aplicadas várias transformações aos dados. É efetuada uma limpeza dos dados por forma a eliminar duplicados, corrigir a ortografia, valores sem sentido, dados contraditórios, etc. Na fase posterior os dados são formatados. São tratados os campos que não vêm preenchidos, são efetuadas concatenações de duas ou mais colunas, por exemplo, a junção dos nomes com as siglas das unidades orgânicas. Os valores originais são modificados ou inseridos numa categoria previamente definida, como é o caso dos intervalos de idade dos alunos. Tendo os dados limpos e formatados, a sua qualidade aumentou e já podem ser aplicadas as operações para calcular os factos que constituem o modelo de dados. É nessa fase que são calculados os agregados, como por exemplo, o total de inscrições num curso para descobrir o tempo que um aluno o demorou a concluir.

No fim do tratamento dos dados é efetuado o seu carregamento para a *data warehouse*. Primeiro são preenchidas as dimensões e só depois as tabelas de factos, uma vez que estas possuem ligações (chaves estrangeiras) para as dimensões. Para encontrar essas chaves são efetuadas várias consultas às dimensões de modo a encontrar os registos pretendidos. Só no fim de existirem todos as chaves para os registos das dimensões no fluxo de dados é que são preenchidas as tabelas de facto.

Inicialmente serão tratados e carregados para a *data warehouse* todos os dados desde 2008/2009 até ao momento, mais precisamente, até ao ano letivo de 2012/2013. O objetivo inicial era carregar toda a informação para a *data warehouse* contudo percebeu-se que como o NÓNIO só passou a ser usado por volta desse ano letivo nas várias faculdades, os dados anteriores não possuíam qualidade suficiente para serem tratados. Além de serem em pouca quantidade estavam bastante incompletos, condenando a sua análise. Os dados relativos aos anos letivos posteriores a 2012/2013 serão recolhidos, modificados e adicionados à *data warehouse* de forma automática, no final de cada ano letivo através do agendamento de transformações *ETL*.

4.5 Origem dos dados

Uma vez que o sistema NÓNIO suporta todos os processos de gestão académica da Universidade de Coimbra e os dados do sistema antigo (*WoC*) foram importados para a base de dados do NÓNIO, este possui toda a informação imprescindível para o cálculo dos indicadores do módulo do sucesso escolar.

De modo a obter toda a informação necessária da base de dados, foi elaborado um documento com todos os campos que iam ser precisos. Com base nesse documento foram criadas sete vistas materializadas. Essas vistas encontram-se brevemente descritas na tabela seguinte e explicadas detalhadamente num documento em anexo (Anexo [5]).

Vista materializada	Descrição
MVIEW_CURSOS_UO_A	Vista com a informação dos cursos existentes na universidade e o ano letivo em que funcionaram.
MVIEW_DEMOGRAFIA_MATRICULA	Vista com os dados pessoais e a demografia dos alunos.
MVIEW_ESTADOS_MATRICULAS	Vista com as matrículas dos alunos num curso e o seu respetivo estado.
MVIEW_INSCRICOES_CURSO	Vista com as inscrições dos alunos nos cursos.
MVIEW_INSCRICOES_DISCIPLINAS	Vista com as inscrições dos alunos nas unidades curriculares.
MVIEW_SITUACOES_ESPECIAIS	Vista com as situações especiais atribuídas aos alunos.
MVIEW_UNIDADES_ORGANICA	Vista com todas as unidades orgânicas da universidade.

Tabela 11 – Descrição das vistas materializadas.

As vistas anteriores foram criadas no sistema de desenvolvimento do NÓNIO e são atualizadas automaticamente sempre que surgirem novos dados ou alterações aos existentes.

Embora os dados usados no módulo do sucesso escolar sejam todos extraídos da base de dados do NÓNIO, a *data warehouse* apenas é atualizada anualmente, no final de cada ano letivo porque só assim é possível obter todos os dados do mesmo.

4.6 Modelo de dados

Para além do modelo multidimensional da *data warehouse* pode existir também um modelo relacional e normalizado para a área temporária. Contudo, Kimball[2] refere no seu livro que muitos dos projetos de *data warehouse* falham nesta etapa uma vez que as equipas tendem a gastar muitos recursos e energia na elaboração de modelos relacionais para a área temporária e esquecem-se que o mais importante é o modelo multidimensional. Tendo em conta que o *ETL* já é a fase mais complicada e demorada, não será usado um modelo de dados complexo para a área temporária, esse será construído de acordo com as vistas de dados que forem recebidas, sem grandes preocupações com as relações entre tabelas. O importante na área temporária é o pré-processamento dos dados e o seu posterior carregamento para a *data warehouse*.

4.6.1 Área temporária

O modelo de dados da área temporária é composto por oito tabelas. Estas tabelas guardam os dados retirados das vistas materializadas para serem alvo de transformações e posteriormente carregados para a *data warehouse*.

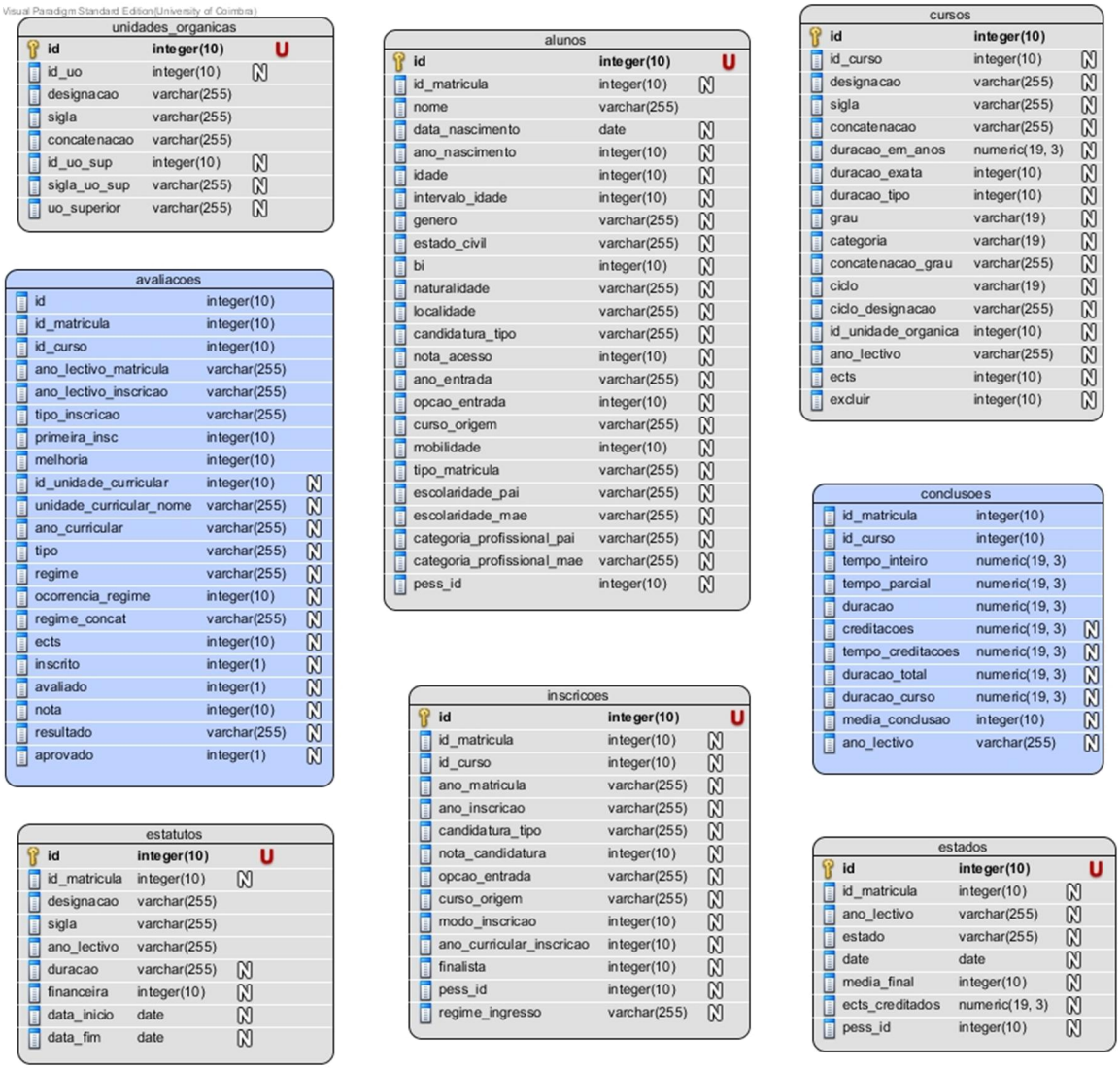

Figura 9 – Modelo de dados da área temporária.

A tabela alunos guarda toda a informação relativa aos estudantes, incluindo os dados demográficos dos mesmos. Os dados desta tabela são retirados da vista *MVIEW_DEMOGRAFIA_MATRICULA*.

A tabela estados armazena apenas o último estado de cada matrícula. Sempre que é criada uma nova matrícula é adicionada uma entrada com o estado ativo. Esse estado pode ser modificado para concluído, anulado, interrompido, entre outros. Um aluno que tenha a matrícula interrompida, se voltar para a frequentar a universidade, é criada uma nova matrícula com o estado ativo. A informação desta tabela vem da vista *MVIEW_ESTADOS_MATRICULAS*.

A tabela inscrições possui todas as inscrições que um aluno faça num curso. Estes dados são retirados da vista *MVIEW_INSCRICOES_CURSO*.

A tabela avaliações armazena os dados relativos às avaliações das unidades curriculares. Todos os dados são extraídos da vista *MVIEW_INSCRICOES_DISCIPLINAS*.

A tabela conclusões serve para calcular o tempo que um aluno demorou a concluir um curso. Esta tabela não recebe dados vindos diretamente das vistas materializadas, os seus dados provém de várias tabelas da área temporária, como a *temp.inscricoes*, a *temp.estados*, etc. Nesta tabela são somadas as inscrições integrais com as parciais, aplicada a fórmula das creditações e assim obtêm-se o tempo, em anos, que o aluno demorou a acabar o curso.

A tabela estatutos guarda as situações especiais atribuídas a cada aluno. A informação desta vista provém da tabela *MVIEW_SITUACOES_ESPECIAIS*.

A tabela cursos guarda informação relativa aos cursos da universidade. Os dados são extraídos da vista *MVIEW_CURSOS_UO_A*.

E, por fim, a tabela unidades orgânicas possui todas as unidades orgânicas da universidade que são retiradas da tabela *MVIEW_UNIDADES_ORGANICAS*.

O código *SQL* para gerar as tabelas e a descrição de cada campo bem como alguns exemplos dos valores que cada coluna pode possuir encontram-se no Anexo [6].

4.6.2 Data Warehouse

Durante a construção de uma *data warehouse* o modelo de dados mais usado é o esquema em estrela. Um esquema em estrela é composto por tabelas de factos com várias tabelas de dimensões associadas. Uma tabela de factos representa os eventos ocorridos. É composta por chaves estrangeiras para as dimensões e por atributos numéricos ou aditivos. Não possui chave primária, a sua chave é composta pelas suas chaves estrangeiras. As dimensões contêm as características dos eventos, ou seja, complementam os factos com mais informação sobre os mesmos. São compostas por uma chave primária e atributos específicos que podem ser temporais, geográficos, relacionados com pessoas, objetos, etc.

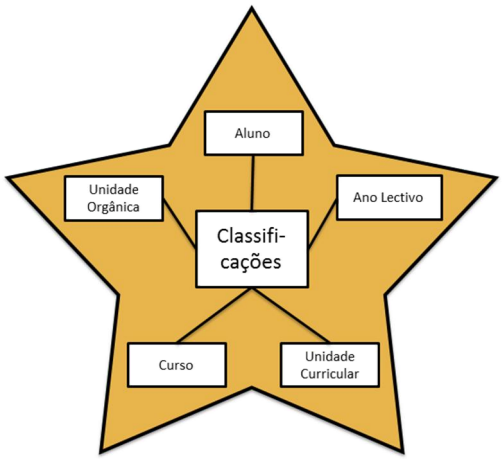

Figura 10 – Figura ilustrativa de um esquema em estrela.

Na Figura 10 encontra-se um exemplo de um esquema em estrela, a tabela de factos classificações possui as notas obtidas pelos alunos, nas unidades curriculares dos vários cursos que pertencem às unidades orgânicas, nos vários anos letivos. Ao redor da tabela de factos encontram-se as dimensões que guardam a informação complementar à tabela de factos.

O modelo de dados é composto por nove tabelas de factos representadas a azul e por sete dimensões representadas a cinzento. Para os indicadores pertencentes ao sucesso nas avaliações são usadas as seguintes tabelas: *se_f_inscricao* e *se_f_classificacao*. Nos separadores

relativos às conclusões de curso são utilizadas as tabelas *se_f_finalista* e *se_f_conclusao*. Para a área do abandono escolar é usada a tabela *se_f_abandono*.

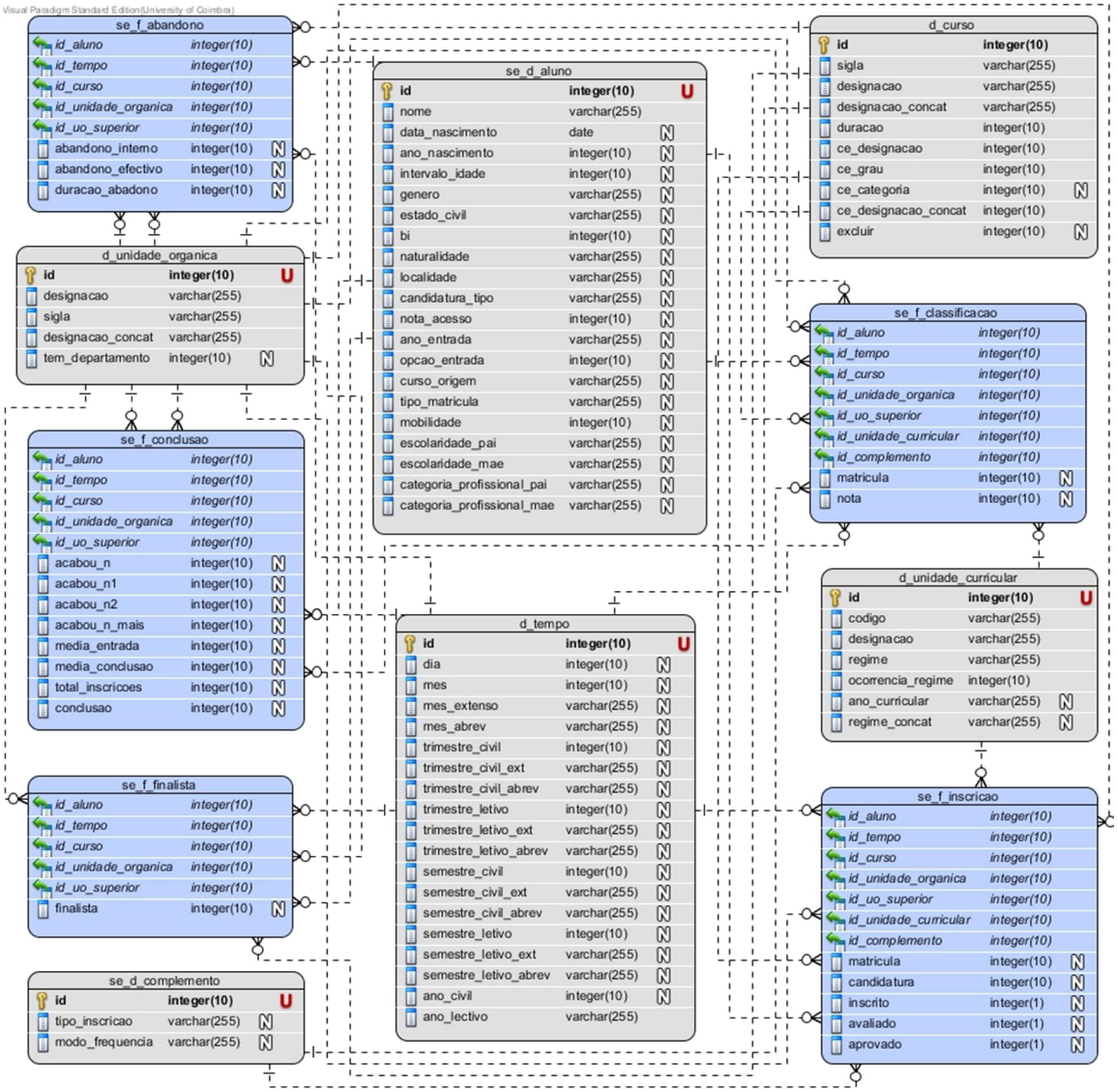

Figura 11 – Parte do modelo de dados da *data warehouse*.

Para uma melhor compreensão do modelo de dados multidimensional, são analisadas as sete dimensões existentes no modelo de dados e posteriormente cada uma das tabelas de factos.

Dimensões	Descrição
d_tempo	Dimensão temporal que possibilita o histórico na base de dados. Possui informação como o ano e os períodos letivos.
se_d_aluno	Dimensão que contém os dados demográficos e de acesso ao ensino superior relativos a cada aluno inscrito na universidade. Sempre que um dos dados do aluno mudar, é introduzido um novo registo com a alteração efetuada para que exista histórico na *data warehouse*. Este tipo de dimensões cujos atributos podem mudar ao longo do tempo denominam-se *slowly changing dimensions*.
d_unidade_organica	Guarda todas as unidades orgânicas pertencentes à universidade. Possui o nome, a abreviatura e a junção dos dois (exemplo: Departamento de Engenharia Informática (DEI)). Os

	departamentos também são unidades orgânicas que pertencem a uma unidade orgânica superior, por isso esta dimensão possui uma coluna denominada *tem_departamento* que indica que uma determinada faculdade possui departamentos.
d_curso	Armazena a informação relativa a todos os cursos da universidade, o nome, a abreviatura, a concatenação de ambos e a duração estabelecida do curso. Também possui informação relativa ao ciclo de estudos a que o curso pertence (licenciatura, mestrado, etc.), a sua categoria (exemplo: mestrado integrado) e a concatenação das duas. A coluna *excluir* indica se esse curso deve ser excluído aquando a análise do abandono escolar.
d_unidade_curricular	Dimensão onde estão todas as unidades curriculares existentes na Universidade de Coimbra. Possui o nome, o ano curricular em que é leccionada (exemplo: 2º ano) e o período curricular a que pertence (exemplo: 2º semestre).
se_d_complemento	Tabela que serve para completar os dados do aluno, indica se o aluno está inscrito pela primeira vez, se é repetente ou se está a fazer melhoria numa unidade curricular e se o aluno está inscrito a tempo parcial ou integral na universidade.
d_situacao_especial	Possui informação sobre as situações especiais que podem ser atribuídas ao aluno, por exemplo, se é finalista, bolseiro de investigação, trabalhador-estudante, etc.

Tabela 12 – Análise das dimensões do modelo de dados.

Tabela de factos das inscrições (*se_f_inscricao*):

A tabela *se_f_inscricao* auxilia o cálculo das taxas de inscritos, de avaliação, de aprovação e de sucesso escolar. Sempre que um aluno se inscreve numa unidade curricular é inserido um novo registo nesta tabela de factos. Esse registo também possui informação sobre o desempenho do aluno na unidade curricular, ou seja, se participou na avaliação da mesmo e se obteve aprovação.

A granularidade destas tabelas de factos vai desde Universidade de Coimbra até unidades curriculares e pode ser efetuado o *drill-down* e o *roll-up* pelos seguintes níveis: Universidade de Coimbra, Unidades Orgânicas, Departamentos (se existirem), Cursos e Unidades Curriculares.

Atributo	Descrição
id_aluno	Identificador que representa o aluno.
id_tempo	Identificador do ano letivo em que foi efetuada inscrição na unidade curricular.
id_curso	Identificador do curso a que compete a unidade curricular em que o aluno se inscreveu.
id_unidade_organica	Identificador da unidade orgânica (departamento) a que pertence a unidade curricular em que o aluno participou. Caso a faculdade não possua departamentos, este campo não se encontrará preenchido.
id_uo_superior	Identificador da unidade orgânica superior (faculdade) a que pertence a unidade curricular que o aluno frequentou.
id_unidade_curricular	Identificado que representa a unidade curricular em questão.
id_complemento	Distingue se a inscrição na unidade curricular foi efetuada pela

	primeira vez, por um aluno repetente ou se foi uma melhoria. E o modo de frequência do aluno, se este estava inscrito a tempo parcial ou integral.
matricula	Este facto possui o valor do identificador único da matrícula do aluno na base de dados para permitir contagens por alunos distintos. Este facto serve para contar o total de alunos inscritos.
candidatura	Sempre que um aluno se inscreveu em unidades curriculares no mesmo ano letivo em que efetuou a candidatura é guardado neste campo o identificador único da matrícula do aluno. Este facto serve para contar o total de novos alunos inscritos.
inscrito	Campo inteiro com o valor sempre igual a um para contar o número de alunos inscritos nas unidades curriculares.
avaliado	Campo inteiro que indica se o aluno participou na avaliação da unidade curricular a que se inscreveu.
aprovado	Campo inteiro que indica se o aluno obteve aprovação na unidade curricular a que se inscreveu.

Tabela 13 – Descrição das tabelas de factos Inscritos.

Embora os factos *matricula* e *inscrito* sirvam para contabilizar o total de alunos inscritos, ambos possuem granularidades diferentes. O facto *matricula* serve para contar, por exemplo no abandono, o total de alunos inscritos no curso, pois este consegue fazer uma contagem por identificadores de matrículas distintos, embora o mesmo identificador surja várias vezes, consoante o número de unidades curriculares que esse aluno se inscreveu (normalmente 6 unidades curriculares por ano). Já o facto inscrito dá o total de alunos, mas é preciso ter em atenção que o mesmo aluno pode vir duplicado mais do que uma vez e no máximo aparece duplicado quantas vezes o número de inscrições em unidades curriculares.

Tabela de factos das classificações (*se_f_classificacoes*):

A tabela *se_f_classificacao* tem como finalidade auxiliar a computação da média e do desvio padrão das notas obtidas pelos alunos. Esta tabela de factos só armazena as classificações positivas obtidas nas unidades curriculares, por isso, cada entrada representa a aprovação de um aluno a uma unidade curricular. Esta tabela de factos como também pertence ao sucesso nas avaliações, possui a mesma granularidade que a tabelas de factos anterior, ou seja, vai desde Universidade de Coimbra até unidades curriculares.

Atributo	Descrição
id_aluno	Identificador que representa o aluno que foi aprovado numa unidade curricular.
id_ano_lectivo	Ano letivo em que a unidade curricular foi realizada.
id_curso	Identificado que representa o curso.
id_unidade_organica	Identificador do departamento, se este existir.
id_uo_superior	Identificador da unidade orgânica.
id_unidade_curricular	Identificador da unidade curricular em que o aluno foi aprovado.
id_complemento	Diferencia entre alunos inscritos pela 1ª vez, repetentes e melhorias. E o modo de inscrição: parcial ou integral.
matricula	Facto com o indentificador da matrícula do aluno na base de dados para possibilitar a contagem do total de alunos e auxiliar as operações de cálculo da média e desvio padrão.

nota	Nota obtida pelo aluno aquando a aprovação na unidade curricular.

Tabela 14 – Descrição da tabela de factos Classificações.

Tabela de factos dos finalistas (*se_f_finalista*):

A única finalidade da tabela *se_f_finalista* é efetuar o cálculo do total de alunos finalistas que se encontram inscritos na universidade. Sempre que um aluno reúne condições para concluir o curso em que se encontra inscrito é-lhe atribuído a situação especial de finalista. Esta tabela de factos guarda o registo de todos os alunos que tinham o estatuto de finalista num determinado ano letivo. Possui as mesmas dimensões que as restantes à exceção da dimensão "Complemento" que uma vez que estamos a analisar conclusões de curso não faz sentido. E o único facto que esta tabela possui é um atributo inteiro, com o valor sempre a um que serve para facilitar a contagem de alunos finalistas na *data warehouse*. A Granularidade desta tabela vai apenas até aos cursos, ou seja, é possível fazer o *drill-down* e o *roll-up* desde a Universidade de Coimbra passando por Unidades Orgânicas, Departamentos (se existirem) e termina no nível dos Cursos.

Tabela de factos das conclusões (*se_f_conclusoes*):

A tabela de factos *se_f_conclusao* serve para calcular todos os restantes indicadores dessa área, as conclusões de curso e respetivas durações bem como as comparações entre as médias de entradas e as médias de saída. Esta tabela de factos armazena os registos dos cursos concluídos na Universidade de Coimbra. As dimensões são as mesmas que as restantes tabelas de factos, mas mais uma vez, sem a ligação à dimensão *se_d_complemento*. Para evitar uma repetição exaustiva de chaves para as dimensões, na tabela seguinte apenas constam os factos da tabela conclusões. E a granularidade não contempla unidades curriculares uma vez que não faz sentido, o que significa que o nível mais específico é o nível dos cursos.

Atributo	Descrição
acabou_n	Campo que indica se o aluno terminou o curso na duração definida para o mesmo.
acabou_n1	Campo que indica se o aluno precisou de mais um ano para concluir o curso.
acabou_n2	Campo que indica se o aluno precisou de dois anos extra para concluir o curso.
acabou_n_mais	Campo que indica se o aluno precisou de mais de dois anos extra para concluir o curso em que se matriculou.
media_entrada	Indica a média de entrada do aluno no curso ou a média da formação anterior do aluno.
media_conclusao	Média que o aluno obteve no curso.
total_inscricoes	Facto que possui o total de inscrições nesse curso, nesse ano letivo.
conclusao	Facto que possui o valor sempre a um para facilitar a contagem dos alunos que concluíram o curso.

Tabela 15 – Descrição da tabela de factos Conclusões.

Abandono (*se_f_abandono*):

A tabela *se_f_abandono* regista o tipo de abandono e a sua duração. Terminado o prazo de inscrições na universidade o abandono escolar será calculado. Como foi explicado no

capítulo de requisitos, existe uma fórmula para o cálculo do abandono. No entanto, para efeito de validação interna dos resultados foi calculado o abandono interno e efetivo analisando os estados de matrículas. O abandono interno consiste na contagem das mudanças de curso dentro da universidade, então foram analisadas todas as matrículas com estado interrompido e foi verificado se cada um desses alunos possuía outra inscrição na universidade mas noutro curso diferente. Se possuísse era inserido um registo na tabela de factos com o tipo de abandono interno.

Já o abandono efetivo consiste no total de alunos que, tendo o curso incompleto, não se voltaram a inscrever na universidade. Para o cálculo deste tipo de abandono foram verificados, mais uma vez, todas as matrículas com o estado interrompido e, para esses alunos foi confirmado se não possuíam mais nenhuma matrícula na universidade até ao presente ano letivo. Se possuíssem uma matrícula num ano letivo mais à frente eram contabilizados o total de anos que esteve sem se inscrever e inserido na tabela de factos no campo *duracao_abandono*.

A granularidade no abandono escolar também vai desde a universidade aos cursos. Esta tabela de factos também não possui uma ligação à dimensão "Complemento" porque não faz sentido. As diferenças em relação às restantes encontram esclarecidas na tabela seguinte:

Atributo	Descrição
abandono_interno	Facto que possui valor um caso o aluno em questão tenha mudado de curso.
abandono_efectivo	Facto que possui o valor a um caso o aluno em questão tenha abandonado a universidade.
duração_abandono	Facto que guarda o valor da duração do abandono para os alunos que interromperam a matrícula mas voltaram mais tarde para a universidade.

Tabela 16 – Descrição dos factos da tabela de factos Abandono.

Ainda sobre o cálculo do abandono, foi pedido para excluir os casos que possam indicar falsos abandonos. São esses os alunos de mobilidade, os cursos em associação e a seguinte lista de situações especiais: "*CsF – Programa Ciência sem Fronteiras*", "*PLI-Programa Internacional de Licenciaturas*", "*Doutoramento em Regime de Cotutela*", "*Programa Inter-Universitário de Doutoramento em Psicologia*", "*Doutorando deslocado em instituição estrangeira*", "*Programa Ciência sem Fronteiras*", "*Protocolos*", "*Protocolo_Universidade Federal da Bahia e Universidade de Coimbra_Lic Direito_Dupla titulação*".

Muitos dos alunos nas situações anteriores, como tiram o curso em várias universidades, pode parecer que abandonaram o curso quando na realidade podem apenas estar a frequentar o mesmo curso noutra universidade e haver problemas de comunicações de inscrições ou estados de matrículas.

A Figura 11 – Parte do modelo de dados da *data warehouse*.Figura 11 representa apenas uma parte do modelo de dados, no entanto, para as tabelas *se_f_inscricao*, *se_f_classificacao*, *se_f_conlcusao* e *se_f_abandono* foram criados duplicados das tabelas de factos com a única diferença de que possuem uma ligação à dimensão que guarda as situações especiais. Uma vez que um aluno pode possuir várias situações especiais no decorrer do ano letivo, o registo do mesmo aluno terá que ser introduzido na tabela várias vezes, tantas quanto as situações especiais que conter no ano letivo. Ao introduzir várias vezes o mesmo registo variando apenas o valor da dimensão das situações especiais deixa-se de poder obter valores fidedignos nas operações sobre os factos, daí haver a necessidade de criar tabelas

praticamente duplicadas. A outra parte do modelo de dados encontra-se no Anexo [6], bem como mais informação sobre o mesmo.

4.6.3 Partilha do modelo de dados

Os vários modelos de dados para a *data warehouse* dos restantes elementos do Projeto DW-UC possuíam dados comuns. Então, para efeitos de poupança de espaço ocupado e redundância dos dados foi pensado criar um modelo de dados único e integrar todas as dimensões e tabelas de factos de cada módulo. Existiram algumas reuniões onde percebemos quais os campos que seriam comuns, foram criadas tabelas aptas para serem partilhadas por vários módulos e foi definida uma nomenclatura para identificar o(s) autor(es) das mesmas. Cada tabela não partilhada devia começar pelas iniciais do seu módulo ("se" neste caso), seguir-se por "f" caso fosse uma tabela de factos ou por "d" caso fosse uma dimensão e só no fim é que constava o nome da tabela. As tabelas que não começassem pelas iniciais dos módulos seriam partilhadas. No desenrolar do projeto percebeu-se que a única informação que podia realmente ser partilhada seriam as dimensões. Mais tarde levantaram-se também problemas em partilhar apenas as dimensões entre os módulos. Uma vez que as dimensões são o que ocupa menos espaço numa *data warehouse* acabou por ficar acordado que os modelos de dados não seriam partilhados entre os módulos.

4.6.4 Previsão do espaço ocupado pela *Data Warehouse*

Aquando a criação do modelo de dados da *data warehouse* também é importante calcular o espaço que irá ser ocupado.
Serão guardadas 552 entradas na dimensão anos letivos de modo a conseguir dar respostas aos próximos anos letivos. Pelo relatório de gestão e contas consolidado de 2012[3] existem 23700 alunos inscritos na faculdade, por isso foi especulado que tenham passado cerca de 50000 alunos pela universidade nos anos letivos mais recentes. Sabe-se que existem cerca de 120 unidades orgânicas na universidade e, mais uma vez, pelo relatório anterior[3] existiram em 2012/2013 um total de 273 cursos na universidade, valor esse que foi arredondado para 300. Supôs-se que já existam umas 7000 unidades curriculares, uma vez que não foram encontrados valores aproximados para essa dimensão. Foram combinados todos os valores possíveis para a dimensão *se_d_complemento* e foi obtido um total de 12 entradas. Sabe-se ainda que existem 63 situações especiais diferentes.
Sabendo que cada inteiro ocupa 4 *bytes*, cada carácter ocupa 1 *byte* e cada data ocupa 7 *bytes*, foram usados valores superiores aos esperados e foi realizada a seguinte estimativa.

Dimensão	Nº de registos	Espaço ocupado
d_tempo	552	426696 bytes
se_d_aluno	50000	167450000 bytes
d_unidade_organica	120	92760 bytes
d_curso	300	237900 bytes
d_unidade_curricular	7000	8981000 bytes
se_d_complemento	12	6168 bytes
d_situacao_especial	63	65016 bytes
	Total aproximado:	169 MB

Tabela 17 – Previsão do espaço ocupado pelas dimensões.

Tabelas de Factos	Tamanho de um registo	Espaço para 10^8 registos
se_f_inscricao	48 bytes	4.47 GB
se_f_classificacao	36 bytes	3.35 GB
se_f_finalista	24 bytes	2.24 GB
se_f_conlcusao	52 bytes	4.84 GB
se_f_abandono	32 bytes	2.98 GB
se_f_inscricao_especial	52 bytes	4.84 GB
se_f_classificacao_especial	40 bytes	3.73 GB
se_f_conclusao_especial	56 bytes	5.22 GB
se_f_abandono_especial	36 bytes	3.35 GB
	Total aproximado:	35 GB

Tabela 18 – Estimativa para o espaço ocupado pelas tabelas de factos.

Habitualmente as tabelas de facto ocupam 90% ou mais do espaço total de uma *data warehouse*. Uma vez que não foi possível apurar quantos registos irão ter ao certo cada uma das tabelas de factos, foi escolhido um valor exageradamente grande para simular o pior dos casos e foram efetuados os cálculos.

Com as estimativas concluiu-se que, se na altura da construção, cada tabela de factos chegar aos 100000000 registos (10^8), a *data warehouse* passa a ocupar cerca de *35GB* o que é perfeitamente suportável.

4.6.5 Espaço atual ocupado pela *data warehouse*

A tabela seguinte indica a quantidade de espaço que está a ser usada de momento por cada tabela da *data warehouse*. De notar que na Tabela 19 não foi tido em consideração o espaço ocupado pelos índices, vistas ou agregados. Isto porque à medida que vamos usando a plataforma o *Mondrian* vai gerando os agregados dos cubos e guarda essa informação em memória. Por isso é que quando um *dashboard* é executado pela primeira vez tende a ser mais lento do que quando já existem dados em cache.

Tabelas de Factos	Espaço ocupado	Dimensões	Espaço ocupado
se_f_inscricao	95 MB	*d_tempo*	88 KB
se_f_classificacao	52 MB	*se_d_aluno*	61 MB
se_f_finalista	1440 KB	*d_unidade_organica*	56 KB
se_f_conlcusao	3040 KB	*d_curso*	216 KB
se_f_abandono	744 KB	*d_unidade_curricular*	968 KB
se_f_inscricao_especial	117 MB	*se_d_complemento*	16 KB
se_f_classificacao_especial	31 MB	*d_situacao_especial*	56 KB
se_f_conclusao_especial	1792 KB		
se_f_abandono_especial	512 KB		
Total:	302 MB	Total:	62 MB

Tabela 19 – Espaço ocupado atualmente pela *data warehouse*.

Concluir-se então que o espaço ocupado atualmente pela *data warehouse* ronda os 364 MB.

5 Implementação

Neste capítulo constam os detalhes mais importantes da fase de implementação, a um nível mais aprofundado que no capítulo anterior. De um modo geral a implementação pode ser dividida em fases, a fase inicial do *ETL*, a criação dos cubos e, por fim, a criação dos *dashboards*.

5.1 ETL (extração, transformação e carregamento)

Nesta secção serão introduzidas e explicadas de forma mais detalhada todas as transformações *ETL* e posteriormente os *jobs*. Um *job* é uma forma de executar uma sequência de transformações ou de outros *jobs*. Podem ser agendados e executados de forma automática. É dessa forma que a *data warehouse* é atualizada.

5.1.1 Componentes das transformações

Na maioria das transformações *ETL* os componentes do *Pentaho Data Integration* que foram usados eram quase sempre os mesmos. As funções desses componentes encontram-se brevemente descrita na tabela seguinte, bem como alguns exemplos práticos do seu uso.

Componentes	Descrição
	O primeiro componente carrega os dados guardados numa tabela para o fluxo de dados da transformação. O segundo guarda os dados pretendidos do fluxo de dados da transformação numa tabela da base de dados.
	Este componente adiciona uma sequência numérica, foi bastante usado para gerar identificadores únicos para as tabelas.
	Ambos os componentes atualizam os dados de uma tabela, a única diferença é que o primeiro percorre a tabela à procura dos dados que é suposto inserir e se não os encontrar, insere um novo registo. O segundo só faz atualizações aos dados.
	O componente *Merge Join* permite a junção de dois fluxos de dados num só. Essa junção é feita com base em 4 tipos: *Inner Join*, *Left Outer, Right Outer e Full Outer*. Este componente só funciona se antes da sua execução os dados vierem ordenados por uma ou mais colunas.
	Componente que faz concatenações de duas ou mais colunas, muito usado para juntar os nomes com as siglas nas dimensões.
	Este componente extrai informação de uma *string*. Foi muito usado para extrair o ano pretendido das datas ou dos anos letivos.
	Este componente serve para selecionar valores do fluxo de dados no entanto neste projeto foi usado para mudar o tipo de dados e a precisão dos valores selecionados.
	Componente que verifica se o campo de um registo não está preenchido, se não estiver coloca um valor por defeito. Foi usado para atribuir o valor "Desconhecido" em vez de deixar o campo vazio.
	O *Add constants* adiciona ao fluxo de dados uma coluna com um valor definido. Foi usado, por exemplo, para preencher alguns

	factos com o valor 1 e facilitar a sua contagem.
Calculator	A calculadora efetua várias operações aos dados numéricos, faz arredondamentos, várias operações com datas, etc. Foi bastante usada para calcular idades, aspetos das conclusões de curso e duração do abandono.
Group by	Este componente permite o cálculo de agregados. Agrupa os dados consoante um valor e possibilita o uso de somas, médias, máximos, mínimos, total de valores agrupados, etc.
Unique rows	Componente que elimina as linhas duplicadas, muito útil aquando a limpeza dos dados.
Filter rows Switch / Case	Ambos os componentes efetuam condições sobre os dados e separam-nos por vários fluxos.
Execute SQL script	Executa *queries SQL* à base de dados. Foi usado essencialmente para modificar valores numa coluna através da instrução *update*.
Modified Java Script Value	Este componente permite a programação de *JavaScript*. E foi usado para modificar alguns valores ou introduzir os valores num intervalo definido, por exemplo, o intervalo de idades.
Block this step until steps finish	O *Block this step until steps finish* bloqueia o fluxo de dados até que outro componente, definido, acabe completamente de executar.
Combination lookup/update	Este componente foi bastante usado no preenchimento das dimensões. Procura um registo na tabela, se não existir ou existir com algum campo diferente, insere um novo e mantem o anterior.
Database lookup	Procura um registo com base em alguns campos selecionados e devolve outros campos desse mesmo registo. Foi bastante usado no preenchimento das tabelas de factos para encontrar os identificadores dos registos pretendidos nas dimensões.

Tabela 20 – Explicação dos componentes mais usados do *Pentaho Data Integration*.

5.1.2 Preenchimento da área temporária

As transformações que preenchem as tabelas da área temporária são quase todas muito simples uma vez que consistem na extração dos dados das vistas materializadas do NÓNIO e no seu armazenamento em tabelas temporárias.

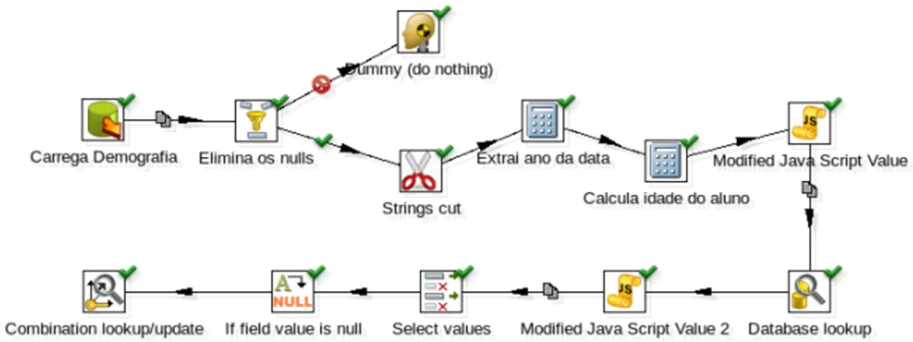

Figura 12 - Preenchimento da tabela da área temporária alunos.

A Figura 12 mostra uma transformação ligeiramente mais complexa, do preenchimento de uma tabela da área temporária. Neste caso os dados com a demografia do aluno são extraídos da vista materializada *MVIEW_DEMOGRAFIA_MATRICULA*, é aplicado um filtro que elimina os registos cujo identificador da matrícula ou o nome não venham preenchidos. Se um registo da demografia do aluno não estiver associado a uma matrícula na universidade, esse registo é inútil uma vez que não traz informação nenhuma que possa ser usada nos indicadores. De seguida é recortado o primeiro ano da *string* que possui o ano

letivo em que foi efetuada a matrícula e é convertido para inteiro na calculadora. Ainda na calculadora é extraído o ano da data de nascimento, e é subtraído ao ano de nascimento do aluno o ano em que este se matriculou na universidade para obter a data que o aluno tinha nessa altura. Através de *JavaScript* é inserida a data num dos intervalos definidos, bem como a escolaridade do pai e da mãe, a situação profissional de ambos os pais, se é aluno de mobilidade ou não, etc. No componente *database lookup* é feita uma consulta à tabela da área temporária onde estão guardadas as inscrições, é procurado o registo pelo identificador de matrícula e devolvida informação sobre a candidatura deste aluno no curso, como por exemplo o tipo de candidatura, o curso de origem, a opção em que o aluno colocou o curso, etc. Mais uma vez, através de *JavaScript* é colocado o curso que o aluno frequentou anteriormente num dos intervalos pré-definidos. É convertida a opção em que o aluno colocou o curso em que entrou para o tipo de dados inteiro. É verificado se existem campos vazios, nomeadamente na situação profissional e escolaridade dos pais, na nacionalidade do aluno, no estado civil, etc. Se estiverem vazios, são preenchidos com a palavra "Desconhecido(a)", uma vez que não existe informação sobre esses parâmetros. E, por fim, a tabela de alunos é preenchida. Ao contrário das outras transformações para o preenchimento da área temporária em que as tabelas são truncadas e recarregadas novamente, tendo em que conta que existem bastantes alunos a abordagem adotada foi atualizar a tabela, introduzindo apenas os novos alunos.

Outra tabela da área temporária cujo preenchimento também é diferente das restantes é a tabela onde ficam guardadas as conclusões de curso. Essa tabela não é preenchida com dados extraídos diretamente do NÓNIO mas sim, com dados já armazenados nas tabelas da área temporária. Cada registo dessa tabela representa uma conclusão de um curso por um aluno e é nessa tabela que é calculado o total de anos que esse aluno demorou a concluir o curso com base no número de inscrições a tempo integral, no número de inscrições a tempo parcial e nas creditações que lhe foram atribuídas aquando a candidatura. Dado que o preenchimento dessa tabela consiste na execução de cinco ficheiros com transformações, foi optado por não ser adicionada ao relatório, no entanto pode ser consultado no Anexo [7] onde constam todas as transformações *ETL*.

5.1.3 Preenchimento das dimensões

Após o preenchimento das tabelas da área temporária e da preparação dos dados para o carregamento da *data warehouse* pode passar-se ao preenchimento das dimensões. Há dimensões que não precisarão de ser atualizadas, por exemplo a dimensão tempo já possui registos para os próximos anos letivos. A dimensão complemento também não precisa ser atualizada uma vez que já possui todas as combinações de registos possíveis. Quanto às restantes o preenchimento também é simples uma vez que consiste no carregamento dos dados da tabela da área temporária e na sua inserção na dimensão através da transformação *Combination lookup/update* que foi desenvolvida para o preenchimento de dimensões na *data warehouse*.

Figura 13 – Preenchimento da dimensão das situações especiais.

No caso da figura anterior, que preenche a dimensão com as situações especiais que podem ser atribuídas aos alunos, como existem grupos definidos com as várias situações especiais, antes de introduzir o registo da situação especial na dimensão, é criado, através de *JavaScript* uma coluna com o grupo a que pertence essa situação especial.

5.1.4 Preenchimento das tabelas de factos

A fase mais demorada do *ETL* é o preenchimento das tabelas de factos. Isso deve-se em parte à vasta quantidade de consultas que precisam de ser feitas às várias tabelas da área temporária e às dimensões antes de possuir os dados todos prontos para introduzir esse registo.

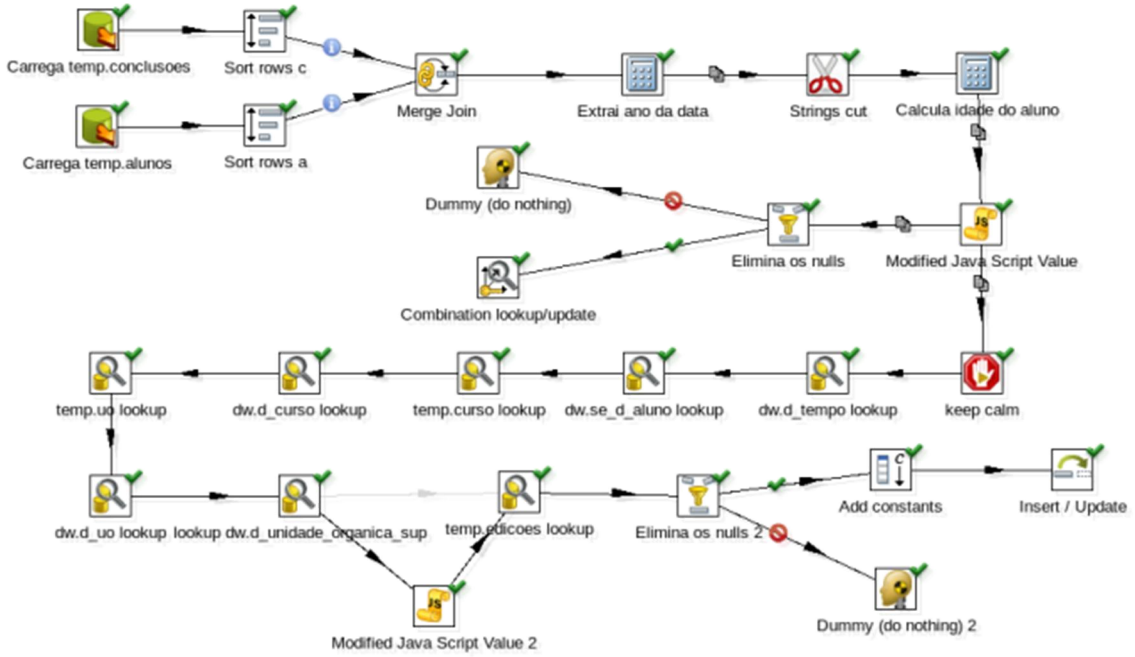

Figura 14 – Exemplo de uma transformação que preenche uma tabela de factos.

A Figura 14 mostra uma das transformações mais simples que preenche as tabelas de factos. Para preencher a tabela de factos com as conclusões de curso, são carregados os dados da tabela temporária conclusões e são unidos com os dados carregados da tabela temporária alunos com base no identificador de matrícula que é comum a ambas as tabelas. São efetuados os componentes para extrair o ano de nascimento do aluno da data de nascimento e é calculada a idade que o aluno tinha quando terminou o curso para posteriormente ser inserido num intervalo de idades e, caso o registo do aluno com esse intervalo de idade ainda não exista na dimensão aluno, é introduzido. A transformação intitulada "*keep calm*" espera que a atualização da dimensão aluno seja efetuada e só no fim é que são executadas várias consultas às dimensões ligadas a esta tabela de factos de modo a encontrar os identificadores dos registos. Isto é, a tabela conclusões guarda uma conclusão de um curso num determinado ano letivo, numa determinada unidade orgânica. Então, é preciso encontrar o registo com o ano letivo igual ao ano letivo em que o aluno terminou o curso e guardar o seu *id* para no final introduzir na tabela de factos. O mesmo para a dimensão dos cursos, para a dimensão das unidades orgânicas, etc. Quando os registos que estão no fluxo de dados da transformação não tem informação suficiente, como é o caso da informação relativa ao curso e à unidade orgânica, são efetuadas consultas às tabelas da área temporária de modo a obter a informação que é preciso para procurar os registos nas dimensões. Estas consultas são feitas sucessivamente até se possuir todos os *ids* e todos os dados precisos para preencher a tabela de factos.

As transformações de inserção dos dados nas tabelas de factos não podem terminar com um componente *combination lookup/update*, à semelhança das transformações anteriores uma vez que, se existirem alterações numa ocorrência, essa transformação insere um novo registo com a alteração. No caso dos registos das tabelas de factos, o que é pretendido é que se existirem alterações nas ocorrências, que estas sejam atualizadas, então a transformação

usada tem de ser um *Insert/Update*. Este componente percorre a tabela até encontrar a combinação de valores pretendidos e, se o registo não existir é inserido, se existir e tiver campos diferentes, é atualizado, senão não faz nada, uma vez que já existe o registo com os valores corretos.

5.1.5 Componentes dos *jobs*

Os principais componentes do *Pentaho Data Integration* que podem ser usados na construção de *jobs* encontram-se explicados na tabela seguinte.

Componentes	Descrição
	Inicia o *job*.
	Permite a execução de consultas SQL aquando a execução do *job*.
	Componente que executa uma transformação ETL guardada num ficheiro.
	Componente que executa outro *job* guardado num ficheiro.
	Termina a execução do *job*.

Tabela 21 – Explicação dos componentes mais usados na criação de *jobs*.

5.1.6 Jobs

Neste projeto foi criado um *job* principal que fica agendado e é responsável pela atualização da *data warehouse*. O que esse *job* faz é chamar os três *jobs* responsáveis pela atualização de dados de cada uma das fases deste projeto.

Figura 15 – Job responsável por atualizar a *data warehouse*.

Quando este *job* começa a executar, chama o *job* que foi criado para atualizar todas as tabelas da área temporária. Após a área temporária ficar atualizada, é chamado um outro *job* que executa todas as transformações necessárias para atualizar as dimensões e, por fim, é executado um *job* que atualiza as tabelas de factos chamando todas as transformações para esse efeito. Mais uma vez, os restantes *jobs* podem ser consultados em anexo (Anexo [7]).

De modo a obter uma ideia de quanto tempo demora a atualização da *data warehouse*, foram carregados os registos todos de um ano letivo e foi tida em consideração o tempo que os *jobs* demoraram a executar.

Job	Tempo de execução
Atualiza área temporária	4h 22min.
Atualiza dimensões	1h 47 min.
Atualiza tabela de factos	20h 53min.

Tabela 22 – Tabela com os tempos de execução.

A máquina em que foram executadas as transformações possui 4 GB de RAM e um processador dual core. Os tempos podem parecer elevados mas há tabelas, como é o caso dos alunos ou das avaliações que possuem muitos registos e por isso demoram bastante tempo a carregar. De relembrar também que mesmo que não sejam inseridos muitos dados, por exemplo nas dimensões, a transformação *ETL* tem que carregar os dados e percorrer a tabela à procura desses registos, o que ainda demora um certo tempo.

5.2 Cubos *OLAP* (*Online analytical processing*)

A criação de um cubo podia ser feita manualmente, escrevendo linhas de *XML* ou podia ser feita através duma *GUI*, o *Schema Workbench*.

Para cada tabela de factos foi criado um cubo. Um cubo é composto por dimensões e *measures*. Uma *measures* é uma quantidade ou o valor de algo que queremos obter. Por exemplo, o total de alunos inscritos ou a média de conclusão de um curso. Cada *measure* possui um nome, uma coluna na tabela de factos à qual está associado e uma função de agregação. Essas funções de agregação podem ser somas, contagens, contagens de valores distintos, máximos, mínimos e médias. A tabela seguinte possui o nome dos cubos criados e a que tabela de factos estão associados.

Nome do cubo	Tabela de factos	Nome do cubo	Tabela de factos
Conclusoes	*se_f_conlcusao*	ConclusoesEspeciais	*se_f_conclusao_especial*
Finalistas	*se_f_finalista*	InscricoesEspeciais	*se_f_inscricao_especial*
Inscricoes	*se_f_inscricao*	ClassificacoesEspeciais	*se_f_classificacao_especial*
Classificacoes	*se_f_classificacao*	AbandonosEspeciais	*se_f_abandono_especial*
Abandonos	*se_f_abandono*		

Tabela 23 – Cubos usados no módulo do Sucesso Escolar.

Para conseguir usar dois ou mais cubos em simultâneo, o *Mondrian* obrigava à criação de cubos virtuais. A tabela seguinte possui os dois cubos virtuais criados e as tabelas de factos que estão associadas aos mesmos.

Nome do cubo vitual	Tabelas de factos
Conclusoes_Finalistas_Inscricoes	*se_f_conlcusao, se_f_finalista e se_f_inscricao.*
Inscricoes_Conclusoes_Abandono_Estatutos	*dw.se_f_inscricao, dw.se_f_inscricao_especial, dw.se_f_conclusao_especial, dw.se_f_inscricao e dw.se_f_inscricao_especial.*

Tabela 24 – Cubos virtuais usados no projeto.

5.2.1 *MDX* (*Multidimensional Expressions*)

A linguagem para aceder à informação armazenada em modelos multidimensionais, como é o caso dos cubos de dados *OLAP*, é o *MDX*. Durante o processo de aprendizagem de *MDX* foi usado o *Saiku Analytics* para desenvolver as *queries* através da sua interface gráfica. No entanto rapidamente se percebeu as suas limitações e as *queries* tiveram que ser criadas manualmente.

A sintaxe do *MDX* é um pouco similar à linguagem *SQL* que é usada em consultas a bases de dados relacionais. A sua estrutura base é a seguinte:

```
SELECT
NON EMPTY {[Measures].[nota]} ON COLUMNS,
NON EMPTY {Hierarchize({[d_curso.hie_d_curso].[sigla].Members})} ON ROWS
FROM [Classificacoes]
WHERE ([d_uo_superior.hie_d_uo_superior].[Faculdade de Economia],
[d_tempo.hie_d_tempo].[2012/2013])
```

Tabela 25 – Exemplo de uma *query MDX*.

A *query* anterior é usada nos *dashboards* do indicador que mostra as médias do sucesso nas avaliações, no nível de granularidade correspondente aos cursos e sem que nenhum filtro seja aplicado. O que a *query* faz é selecionar uma *Measure* do cubo *Classificacoes* que possui a média de todas as classificações obtidas pelos alunos e coloca-a nas colunas. As linhas ficam preenchidas com as siglas dos cursos. Na cláusula *where* pede para mostrar apenas os cursos da Faculdade de Economia e dados relativos apenas ao ano letivo de 2012/2013.

```
with member [Measures].[pp_finalistas] as '([Measures].[finalista] / [Measures].[matricula])'
  member [Measures].[p_finalistas] as '([Measures].[pp_finalistas] * 100)'
  member [Measures].[pp_conclusoes] as '([Measures].[conclusao] / [Measures].[matricula])'
  member [Measures].[p_conclusoes] as '([Measures].[pp_conclusoes] * 100)'
  member [Measures].[ppp_finalistas] as 'IIf((IsEmpty([Measures].[finalista]) AND
([Measures].[conclusao] > 0)), 0, [Measures].[p_finalistas])'
  member [Measures].[ppp_conclusoes] as 'IIf(([Measures].[conclusao] < 0), 0,
[Measures].[p_conclusoes])'
select non empty {[Measures].[ppp_finalistas], [Measures].[ppp_conclusoes]} on columns,
  non empty
Order(Crossjoin(Hierarchize({[d_unidade_organica.hie_d_unidade_organica].[sigla].Member
s}), [se_d_aluno.hie_idade].[intervalo_idade].Members), [Measures].[p_conclusoes], BASC)
on rows
from [Conclusoes_Finalistas_Inscricoes]
where ([d_tempo.hie_d_tempo].[2012/2013],
[d_uo_superior.hie_d_uo_superior].[Faculdade de Ciências e Tecnologia].[FCTUC])
```

Tabela 26 – Outro exemplo de uma *query MDX*.

A *query MDX* anterior faz parte do *dashboard* com o indicador que compara a percentagem de alunos finalistas com a percentagem de conclusões desse ano letivo. Esta *query* começa por criar membros que consistem em criar novas *measures* para usar na *query* através da aplicação de fórmulas sobre os *measures* já disponíveis. Nos primeiros membros são efetuados vários cálculos de modo a encontrar a percentagem de alunos finalistas e a percentagem de alunos que concluíram o curso. Os últimos dois membros criados são proteções para o caso dos valores no cubo estarem mal e surgirem erros, como por exemplo, divisões por zero ou percentagens com valor vazio. Então são aplicadas condições que verificam se existe finalistas ou conclusões e se não existirem força o valor a zero.

A percentagem de finalistas e a percentagem de conclusões é colocada nas colunas. Nas linhas são mostrados os departamentos da *FCTUC* ordenados pela percentagem de conclusões por ordem ascendente e discriminados por intervalo de idade do aluno. Observando ainda a cláusula *where* percebemos que os dados serão referentes ao ano letivo de 2012/2013.

5.3 Servidor *OLAP*

No *Pentaho BI Server*, como já foi mencionado anteriormente, foi usado o *plugin CDE (Community Dashboard Editor)* que permite a criação de *dashboards* interativos e funciona à base de 3 componentes, o *layout*, os *data sources* e os componentes.

No *layout* é definido o *template* do *dashboard* onde podem ser selecionados *templates* já existentes ou criados novos uma vez que suporta *HTML*, *CSS* e *JavaScript*. Para o *layout* dos *dashboards* do projeto foi criado um novo com recurso à *framework Bootstrap*. Também foi nesta área que foi adicionada a *framework* dos gráficos, da ferramenta que exporta as tabelas para um ficheiro, etc. Para a criação dos gráficos foi usada a *API Highcharts*.

Nos *data sources* são criadas as ligações aos dados, neste caso foram usadas *queries MDX* para aceder aos mesmos. A execução de uma *query MDX* devolve resultados que precisam de ser arranjados antes de serem expostos nos *dashboards*.

Nos componentes, como o nome indica, são criados todos os componentes que constituem os *dashboards*, desde gráficos, tabelas, caixas de seleção, scripts, etc. Os gráficos recebem os dados que surgem como resultado de uma *query*. Consoante o formato do resultado da *query*, é executado código *JavaScript* que interpreta esse resultado e que o deixa preparado para ser armazenado em séries, prontas a serem expostas nos gráficos. Esta parte do projeto deu bastante trabalho uma vez que podem ser obtidos resultados muito diferentes na execução das *queries*. Para além dos gráficos todos os outros componentes que fazem parte dos *dashboards* também foram customizados com código *JavaScript*. A construção das *queries MDX* também foi feita de forma dinâmica, eram verificados que filtros se encontravam selecionados nos *dashboards* e consoante esses parâmetros eram adicionadas ou removidas condições às *queries MDX* através de código *JavaScript*.

5.4 Otimização

Após a criação dos *dashboards* percebeu-se que alguns demoravam algum tempo até conseguirem apresentar os resultados, especialmente os *dashboards* do sucesso nas avaliações dado a elevada quantidade de dados relativos às inscrições nas várias unidades curriculares. O uso de cubos virtuais, ou seja, duas ou mais tabelas de factos em simultâneo, também tornava os *dashboards* mais lentos. Partiu-se então para a otimização mais prática, a colocação de índices. Os índices devem ser inseridos em chaves primárias, chaves forasteiras, colunas com bastantes valores diferentes, etc. Foram criados três índices do tipo *btree* nas várias tabelas de factos, nos campos *id_aluno*, *id_curso* e *id_unidade_curricular*. Esses índices aceleraram o tempo de obtenção de resultados, no entanto ainda não era suficiente. De modo a conseguir tornar os *dashboards* ainda mais rápidos foi também otimizado o código *JavaScript* que interpreta o resultado da *query* e que coloca esse resultado em séries para ser representado nos gráficos.

5.5 Produto final

O produto final resultou em oito *dashboards*, quatro para as conclusões de curso, dois para o sucesso nas avaliações e dois também para o abandono escolar. Como no capítulo 3.4 foi mostrada uma imagem do protótipo do sucesso nas avaliações, neste capítulo também foi exposta uma imagem dos *dashboards* com as taxas de sucesso nas avaliações. O *dashboard* do protótipo teve que ser separado em dois de modo a ficar coerente com os restantes módulos da plataforma. Outra vantagem que a separação trouxe também foi a diminuição do tempo de carregamento porque passaram a ser carregados menos componentes na execução do

dashboard. A Figura 16 mostra o *dashboard* com as taxas de avaliados, de aprovados e de sucesso escolar para a Universidade de Coimbra no ano letivo de 2013/2014. No gráfico de evolução temporal (à esquerda), é possível comprar os valores para essas mesmas taxas nos últimos três anos letivos.

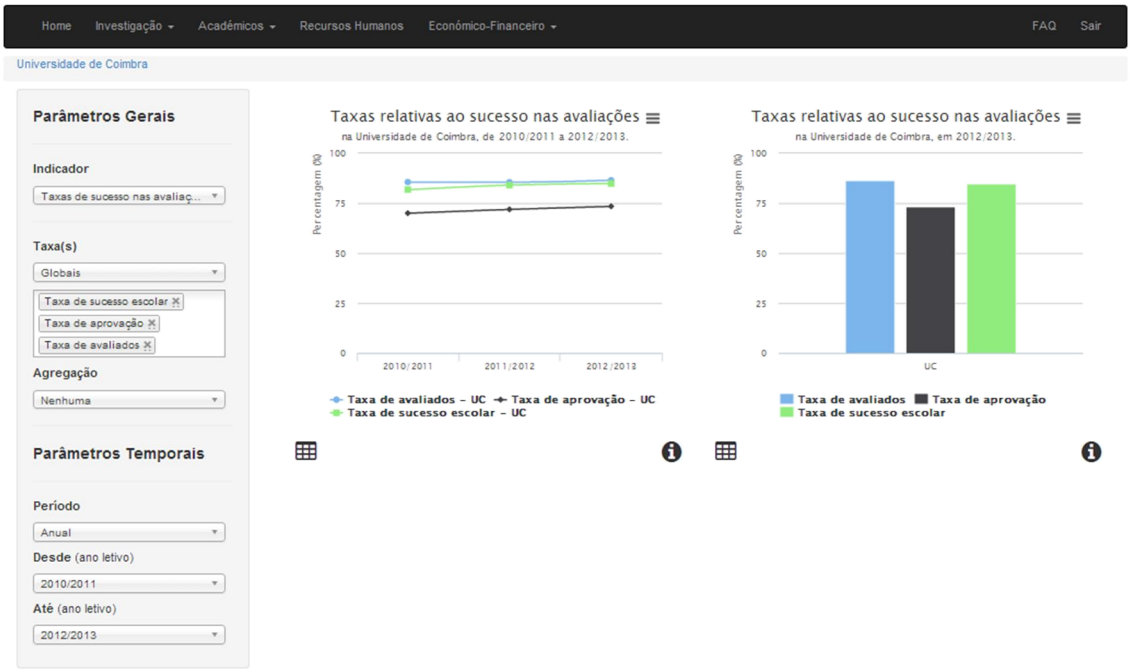

Figura 16 – *Dashboard* com alguns indicadores do sucesso nas avaliações.

A Figura 17 mostra um dos *dashboards* relativos às conclusões de curso. Nesse *dashboard* é possível visualizar a percentagem de alunos que não concluíram o curso no tempo estipulado (gráfico da esquerda), para as várias unidades orgânicas nos últimos três anos letivos. O gráfico da direita mostra os alunos que precisaram de mais do que a duração prevista para a conclusão do curso, discriminado pela quantidade de anos extra que foram necessários. A informação desse gráfico é referente às várias unidades orgânicas no ano letivo de 2012/2013.

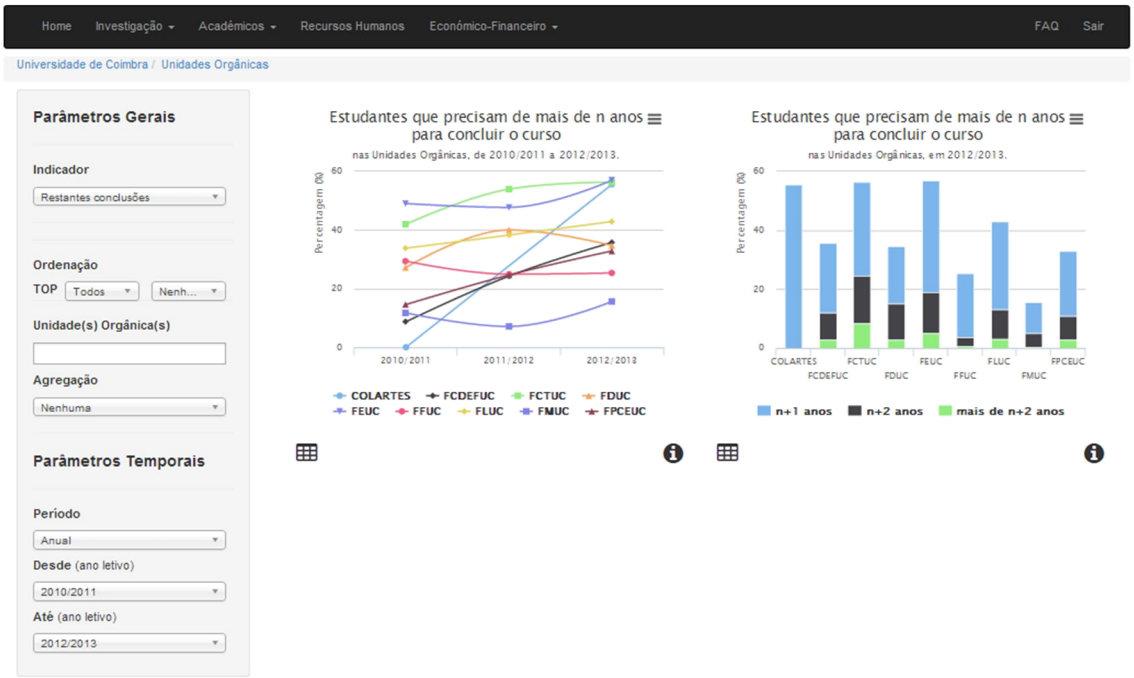

Figura 17 – *Dashboard* com as conclusões de curso.

6 Validação

A validação é um aspeto muito importante não só porque permite verificar se o que estava planeado nos requisitos foi implementado mas também se resolve o problema que havia sido identificado. Não adianta criar uma solução perfeitamente funcional se esta não for intuitiva e apelativa para o utilizador, fácil de usar e capaz de satisfazer as necessidades dos mesmos.

Neste capítulo é detalhado o processo de validação do módulo do sucesso escolar, são descritas as várias etapas, as pessoas envolvidas, os testes e a documentação que foi produzida para o efeito.

6.1 Validação dos resultados

Aquando a realização dos *dashboards* os resultados obtidos iam sendo validados com a informação disponível nas plataformas criadas pelo NÓNIO.

Antes de disponibilizar os *dashboards* foram efetuados vários testes pelos restantes elementos da equipa de modo a encontrar erros de programação.

Posteriormente os *dashboards* foram enviados ao Eng. Marco Neves do Projeto NÓNIO para que este fosse testando à medida que os estes eram produzidos.

Antes da elaboração dos *dashboards* relativos ao abandono escolar a vice-reitora Dra. Madalena Alarcão demonstrou preocupação em como seriam calculados os dados relativos a essa mesma área. Foi então agendada uma reunião para 29 de Abril com a vice-reitora onde esteve presente a Helena Galante da Divisão de Avaliação e Melhoria Contínua. Após esclarecidas as questões sobre o cálculo do abandono foram mostrados os *dashboards* relativos às conclusões de curso. Todas as alterações sugeridas no decorrer da reunião foram tidas em consideração e foram efetuadas.

No mês seguinte foi agendada mais uma reunião com a Helena Galante no Palácio dos Grilos onde foram mostradas as alterações efetuadas e o trabalho realizado até ao momento. Foi obtido mais *feedback* sobre os *dashboards* e foram efetuadas novas alterações aos mesmos.

Após a conclusão de todos os *dashboards* foi marcada uma reunião com os responsáveis pela validação do módulo do sucesso escolar, nomeadamente, com a Helena Galante e o Filipe Rocha da Divisão de Planeamento, Gestão e Desenvolvimento. Foi elaborado um documento de suporte à validação (ANEXO [8]) e que foi enviado por *email* algum tempo após o agendamento da mesma. A reunião realizou-se a 20 de Junho e foi obtido mais *feedback* sobre a plataforma. Até ao momento de escrita deste relatório ainda não foi possível obter o documento de validação preenchido devido a problemas de disponibilidade dos responsáveis pela validação. No entanto todos os aspectos sugeridos pelos mesmos já foram alterados.

6.2 Testes

Para a validação dos requisitos implementados foi elaborada uma tabela onde constam todos os requisitos e uma coluna que indica se foram ou não executados. Após a tabela são justificados os motivos pelos quais alguns requisitos foram ou não cumpridos. O documento

que foi usado para efetuar a validação dos dados (anexo [8]) possui uma tabela com testes funcionais que comprovam que a grande maioria dos requisitos funcionais foram implementados.

Id.	Requisito	Ex.	Id.		Ex.
RF_GE_01	Autenticação	✔	**RF_ SA _19**	Média e desvio padrão das melhorias	✔
RF_GE_02	Fechar sessão	✔	**RF_CC_01**	Conclusões de curso no tempo estipulado	✔
RF_GE_03	Término de sessão	✔	**RF_CC_02**	Média de acesso	✔
RF_GE_04	Navegação entre os módulos	✔	**RF_CC_03**	Média de conclusão	✔
RF_GE_05	Navegação interna	✔	**RF_CC_04**	Estudantes finalistas	✔
RF_GE_06	Parâmetros gerais	✔	**RF_CC_05**	Estudantes que concluíram o curso	✔
RF_GE_07	Parâmetros de tempo	✔	**RF_CC_06**	Número de anos de conclusão	✔
RF_GE_08	Esconder parâmetros	×	**RF_AE_01**	Abandono efetivo	✔
RF_GE_09	Secção de ajuda	✔	**RF_AE_02**	Abandono interno	✔
RF_GE_10	Informação auxiliar	✔	**RF_AE_03**	Abandono total	✔
RF_GE_11	Visualização: Gráfico ↔ Tabela	✔	**RF_AE_04**	Estudantes que regressaram	✔
RF_GE_12	Exportar informação da tabela	✔	**RF_AE_05**	Duração média da interrupção	✔
RF_GE_13	Ordenação e filtragem	✔	**RF_AG_01**	Idades	✔
RF_SA_01	Taxa de avaliados	✔	**RF_AG_02**	Género	✔
RF_SA_02	Taxa de aprovação	✔	**RF_AG_03**	Área de estudos	✔
RF_SA_03	Taxa de sucesso escolar	✔	**RF_AG_04**	Nacionalidade	✔
RF_ SA _04	Taxa de avaliados repetentes	✔	**RF_AG_05**	Escolaridade dos pais	✔
RF_ SA _05	Taxa de aprovação dos repetentes	✔	**RF_AG_06**	Situação profissional dos pais	✔
RF_ SA _06	Taxa de sucesso escolar dos repetentes	✔	**RF_AG_07**	Tipo de matrícula	✔
RF_ SA _07	Taxa de inscritos repetentes	✔	**RF_AG_08**	Opção de procura do curso	✔
RF_ SA _08	Taxa de avaliados das 1ªs inscrições	✔	**RF_AG_09**	Situação especial	✔
RF_ SA _09	Taxa de aprovação das 1ªs inscrições	✔	**RF_AG_10**	Modo de frequência	✔
RF_ SA _10	Taxa de sucesso escolar das 1ªs	✔	**RF_AG_11**	Ciclo	✔

	inscrições				
RF_ SA _11	Taxa de inscritos de 1ª inscrição	✔	**RF_AG_12**	Grau	✔
RF_ SA _12	Taxa de avaliados em melhoria	✔	**RF_AG_13**	Ano curricular	✕
RF_ SA _13	Taxa de aprovados em melhoria	✔	**RF_AG_14**	Período letivo	✔
RF_ SA _14	Taxa de sucesso escolar das melhorias	✔	**RNF_AG_01**	Atualização dos dados	✔
RF_SA_15	Taxa de inscritos em melhoria	✔	**RNF_AG_02**	Compatibilidade do *browser*	✔
RF_ SA _16	Média e desvio padrão globais	✔	**RNF_AG_03**	Compatibilidade do *SO*	✔
RF_ SA _17	Média e desvio padrão dos repetentes	✔	**RNF_AG_04**	Licenças	✔
RF_ SA _18	Média e desvio padrão das 1ªs inscrições	✔	**RNF_AG_01**	*Hardware*	✔

Tabela 27 – Tabela com todos os requisitos do projeto.

Legenda: ✔ – Requisito implementado. ✕ – Requisito não implementado.

O requisito *RF_GE_08* não foi implementado, tratava-se de um requisito de prioridade baixa cuja única vantagem seria deixar mais espaço para a visualização dos gráficos. Uma vez que os gráficos ficam bastante perceptíveis sem que a barra lateral seja escondida, este requisito foi deixado para segundo plano e acabou por não ser implementado por nenhum módulo. O *RF_AG_13* também não foi implementado, isto porque não foi conseguido identificar o ano curricular a que pertenciam algumas unidades curriculares. Há unidades curriculares em que os alunos se podiam inscrever em qualquer ano curricular, outras cujo ano curricular variava consoante a área ou o curso do aluno. Todos os restantes requisitos funcionais foram implementados e encontram-se disponíveis nos *dashboards*.

Quanto aos requisitos não funcionais também foram todos verificados. Para o *RNF_AG_01* foram deixados *jobs* agendados para atualizar a *data warehouse* automaticamente. Para comprovar o requisito *RNF_AG_02* foi testada a plataforma *web* em vários *browsers* e funcionou bem, embora tenha apresentado um pequeno defeito nas caixas de seleção no *Firefox*, defeito esse que é perfeitamente suportável. Para o desenvolvimento do projeto foi adotada uma distribuição *Linux* por isso o requisito foi validado em parte. O *software* é de facto gratuito e as características de hardware são as que foram usadas na máquina de desenvolvimento.

7 Planeamento

Neste capítulo será apresentado o planeamento para o estágio, divido por semestres. Serão também analisadas todas as tarefas realizadas durante os mesmos. Como ferramenta de gestão de projeto, foram criados diagramas de *Gantt*. O estágio teve início a 16 de Setembro de 2013 e terminará a 30 de Junho de 2014.

7.1 Primeiro semestre

Durante o primeiro semestre o método de trabalho da equipa foi à base de atribuições de tarefas que eram validadas semanalmente pelos orientadores. Praticamente todas as terças-feiras pelas nove horas da manhã ocorriam reuniões onde era mostrado e analisado o progresso até ao momento, esclarecidas eventuais questões que surgiam e realizado o planeamento para a semana seguinte.

Tendo em conta que este módulo está inserido num projeto maior, também foram realizadas várias reuniões com a equipa por forma a discutir e a definir o que era comum a todos. Foi criada documentação e partilhada por todos recorrendo ao uso da *Dropbox*.

O diagrama de *Gantt* referente ao primeiro semestre pode ser visualizado no Anexo [9]. De seguida são descritas todas as tarefas realizadas.

7.1.1 Tarefas realizadas

● **Integração no projeto:** Os primeiros dias, mais precisamente, de 16 a 23 de Setembro, foram dedicados à integração no projeto. Houve uma reunião inicial, a 19 de Setembro onde foram definidas as áreas de trabalho de cada um dos estagiários e os grandes objetivos do trabalho. Foi lida documentação relativa ao que já se encontrava implementado, nomeadamente a área de custos e receitas com a investigação, documentos criados pela Universidade e artigos teóricos relacionados com o âmbito do projeto.

● **Levantamento dos *KPIs*:** A definição dos indicadores de desempenho para a área do sucesso escolar foi uma tarefa demorada e cuidada, devido à sua elevada importância. Foram agendadas várias reuniões com a administração da UC e coordenadores de vários cursos e faculdades de modo a averiguar com que informações lidavam diariamente e quais os dados que sendo imprescindíveis, eram difíceis de obter. Durante o período de tempo compreendido entre 26 de Setembro a 23 de Outubro foi reunido com a Doutora Conceição Costa do Departamento de Avaliação e Melhoria Continua, com a Doutora Bernardete Ribeiro, coordenadora da Licenciatura em Engenharia Informática, com o Doutor Carlos Fonseca, coordenador do Doutoramento em Engenharia Informática, com o Doutor Amílcar Cardoso, com a Doutora Teresa Tavares e com o Doutor Rui Gama, atuais subdiretores da Universidade de Letras.

Durante esta fase foram tidas em consideração todas as sugestões obtidas durante as reuniões, foram estudados os documentos cedidos pelos mesmos e foram obtidos os indicadores de desempenho. Foi ainda pensado nas questões a que o sistema pretendia responder, na granularidade, nos parâmetros gerais e nos filtros a aplicar a cada *KPI*. Foram elaborados *mockups* para servir de apoio às reuniões e à elaboração do protótipo.

● **Elaboração da análise das tecnologias:** Em simultâneo com o levantamento dos indicadores de desempenho e com a elaboração do protótipo, foi escrito a análise das tecnologias que se encontra no capítulo de arquitetura e em anexo.

● **Elaboração do protótipo rápido:** Após definidos os indicadores de desempenho procedeu-se à elaboração do protótipo. Esta tarefa foi bastante demorada uma vez que muitas questões relativas ao protótipo tiveram que ser discutidas em grupo, com os outros estagiários, para que todas as vertentes do projeto ficassem coerentes. Também se realizaram várias reuniões internas de avaliação do protótipo, de modo a refinar os pormenores do mesmo. O protótipo foi elaborado com a ferramenta *Justinmind Prototyper* e os gráficos com o *Highcharts*.

● **Documento de Requisitos:** De 1 a 14 de dezembro foi elaborado um documento de requisitos. Visto que todas as áreas da plataforma devem ficar coerentes, os requisitos não funcionais foram discutidos em grupo.

● **Validação dos protótipos:** A validação dos protótipos ocorreu de 10 a 31 de dezembro. Uma vez que esta dependia da disponibilidade de terceiros, não pode ser efetuada anteriormente. Foi efetuada pela Dra. Conceição Costa e pela reitora. Após ambas as reuniões foram efetuadas as alterações acordadas nos protótipos.

● **Documento de especificação dos protótipos:** Aquando a conclusão dos protótipos, foi elaborado um documento de especificação onde são explicados os pressupostos de *design* e analisados os detalhes e as funcionalidades do mesmo.

● **Escrita do relatório:** De 1 a 28 de janeiro foi dedicada à elaboração do relatório intermédio de estágio. Foi revisto a análise das tecnologias e foram elaborados os restantes capítulos.

● **Modelo de dados:** De 7 a 21 de janeiro foi elaborado o modelo de dados. Após ter sido elaborado um esboço do modelo de dados, este foi analisado e discutido pela equipa do projeto.

● **Plano de ETL:** De 7 a 14 de janeiro foi elaborado o plano de ETL que iria ser implementado no segundo semestre.

7.2 Segundo semestre

O planeamento para o segundo semestre encontra-se dividido em dois subcapítulos, no planeamento que foi previsto e no planeamento que foi executado. Durante o primeiro semestre foi criado um planeamento que rapidamente se percebeu que seria impossível de cumprir. Para além de não abranger possíveis imprevistos, não foi tido em conta o tempo adequado para a aprendizagem das ferramentas usadas. Por esse motivo foi reformulado um novo planeamento que foi cumprido.

7.2.1 Previsão

O planeamento do segundo semestre vai passar por implementar, testar e produzir documentação. O método de trabalho continuará semelhante ao primeiro semestre, sendo mantidas as atribuições e validações de tarefas durante as reuniões semanais.

A realização do trabalho foi dividida por áreas de acordo com os requisitos, ou seja, primeiro será implementada a área relativa ao sucesso nas avaliações e serão efetuados testes de integração uma vez que existem no projeto muitos aspetos semelhantes aos dos outros elementos da equipa. Os testes de integração passam por juntar os aspetos comuns nos projetos, quer a nível do aspeto da plataforma final, quer ao nível do modelo de dados.

Consequentemente serão elaboradas as conclusões de curso e o abandono escolar e efetuados testes no final de cada um. Dentro das áreas existem tarefas que passam por elaborar o processo ETL, preencher o modelo de dados e desenvolver os *dashboards* correspondentes.

No final do trabalho serão executados testes de validação e corrigidos eventuais bugs em simultâneo com a escrita do relatório final.

O diagrama de *Gantt* com o planeamento para o segundo semestre encontra-se em anexo (Anexo [9]).

7.2.2 Tarefas realizadas

O Anexo [9] também mostra o diagrama de *Gantt* do trabalho realizado no segundo semestre. Abaixo encontram-se as tarefas detalhadas.

● **Preparação do ambiente de trabalho:** De 10 de fevereiro a 11 de fevereiro foram efetuadas várias tarefas de modo a deixar o ambiente de trabalho configurado e pronto para trabalhar. Foram configuradas as ferramentas e criados utilizadores de acesso às mesmas. Foi criada a base de dados da área temporária, a ligação à base de dados do *NÓNIO*, etc.

● **Aprendizagem do *Pentaho Data Integration*:** De 12 de fevereiro a 20 de fevereiro foi lida a documentação do *Pentaho Data Integration* e o livro *Pentaho Data Integration Cookbook* da autoria de *Alex Meadows, Adrián Sergio Pulvirenti* e *María Carina Roldán*. Também foram realizados alguns tutoriais.

● ***ETL* das conclusões de curso:** O carregamento dos dados relativos às conclusões de curso começou a ser efetuado para a área temporária a 21 de fevereiro. Ao contrário do que estava previsto foi começado pelos indicadores das conclusões de curso uma vez que a vista das classificações que é imprescindível para os indicadores do sucesso escolar ainda não estava completa. Todas as transformações destes indicadores ficaram prontas a 22 de março.

● **Aprendizagem dos cubos:** Foi aprendido a trabalhar com cubos *OLAP* de 23 a 25 de março.

● **Criação dos cubos das conclusões de curso:** Foram criados de 26 a 27 de março.

● **Aprendizagem do *CDE*:** Ocorreu de 28 de março a 4 de abril. Esta aprendizagem foi bastante demorada uma vez que não existia muita informação na internet acerca da programação e customização em *JavaScript* dos vários componentes do *CDE*, entre outros aspetos deste *plugin* que pecam por falta de informação *online*.

● ***Dashboards* das conclusões de curso:** O primeiro *dashboard* demorou imenso tempo a ser criado por imperícia e também porque à medida que iam sendo mostrados apareciam sempre sugestões novas para melhorar o aspecto estético e a sua usabilidade. Os restantes *dashboards* foram criados em menos tempo. Esta tarefa decorreu entre 31 de Março a 26 de Abril.

● ***ETL* do sucesso nas avaliações:** Após a elaboração do *ETL* para as conclusões de curso a elaboração deste foi bastante rápida. No entanto a tabela das avaliações possuía muito mais que um milhão de registos o que fez com que fossem gastos vários dias a carregar para a *data warehouse* todos esses registos. Esta fase ocorreu de 27 de abril a 11 de maio.

● **Criação dos cubos do sucesso nas avaliações:** Bastou o dia 12 de maio para criar todos os cubos. A criação dos cubos era bastante rápida no entanto, sempre que eram criados cubos, era usado o *Saiku* para visualizar os dados e assim conseguir fazer uma validação com a informação disponível no NÓNIO.

● ***Dashboards* do sucesso nas avaliações:** De 13 a 27 de maio foram elaborados os *dashboards* dos indicadores relativos ao sucesso nas avaliações.

● **ETL do abandono escolar:** Decorreu de 28 de maio a 4 de junho. Demorou 1 semana porque a fórmula foi aplicada erradamente da primeira vez.

● **Criação dos cubos do abandono escolar:** Foram elaborados a 5 de junho.

● ***Dashboards* do abandono escolar:** De 6 a 13 de junho foram criados os *dashboards* com os indicadores do abandono escolar.

● **Testes e correção de erros:** No período de 13 de maio a 30 de junho foram testados os vários *dashboards* e sempre que era detetado um erro este era corrigido. Sempre que era sugeridas novas alterações aos *dashboards*, estas também eram executadas. Por isso é que o período de testes e correção de erros foi bastante extenso.

● **Escrita de documentação:** À medida que o projeto ia sendo elaborado também ia sendo criada documentação em paralelo, como foi o caso do documento de transformações *ETL*, do documento com o modelo de dados, etc. Este período decorreu entre 6 de março a 16 de junho.

● **Escrita do relatório de estágio:** Foi reservado para a escrita do relatório as semanas compreendidas entre 16 a 30 de junho.

● **Validação do módulo:** Decorreu entre os dias 20 e 30 de junho.

8 Conclusões

Neste documento foram apresentadas as principais tarefas desenvolvidas durante a elaboração do módulo do sucesso escolar do projeto DW-UC. Uma vez que a área coincidia com aspetos já criados pelo projeto NÓNIO, foi exposto o que já se encontrava realizado, o que iria ser usado e implementado e as mais-valias que traria.

Foi apresentado o que se pretendia desenvolver através da análise de requisitos e como se pretendia desenvolver através da arquitetura. Os requisitos para o desenvolvimento da *data warehouse* relativa ao sucesso escolar foram levantados e demonstrados através da realização de um protótipo interativo. Foi elaborada a arquitetura do sistema onde foi incluída uma visão geral, as tecnologias que iriam ser usadas e as concorrentes, o plano *ETL* e o modelo de dados.

Definidos os requisitos e a arquitetura, passou-se à fase da implementação. Na implementação foram inseridos os detalhes da realização do projeto. Detalhes esses que foram introduzidos no capítulo da arquitetura e dando bastante enfâse à parte mais trabalhosa do projeto, o processo *ETL*.

Antes de finalizar este documento foi descrita a validação, embora ainda se encontre a decorrer. No entanto, dada a sua importância teria mesmo que ser mencionada. Foram ainda listadas todas as tarefas desenvolvidas ao longo deste ano letivo.

Terminado o ano letivo, considero que o balanço geral é positivo e que foram alcançados todos os objetivos pretendidos.

Ao nível das competências, este estágio contribuiu para a minha formação, permitindo a obtenção e aprofundamento de conhecimentos na área de *business intelligence*, área da qual gosto bastante. Também contribuiu a nível profissional e pessoal, tornando-me mais experiente em reuniões profissionais e com clientes.

9 Anexos

Documentos enviados em anexo:

[1] Estudo das tecnologias, documento_analise_das_tecnologias_2014-06-27.pdf

[2] Especificação do protótipo, documento_especificacao_prototipo_sucesso_escolar_2014-06-13.pdf

[3] Documento de requisitos, documento_requisitos_sucesso_escolar_2014-06-27.pdf

[4] Documento de especificação do *design*, doc_especificacao_design_02-12-2013.pdf

[5] Documentação das vistas materializadas, documento_das_vistas_materializadas_2014-06-13.pdf

[6] Documento com o modelo de dados, documento_modelo_de_dados_2014-06-24.pdf

[7] Documento com as transformações *ETL*, documento_etl_sucesso_escolar_2014-06-25.pdf

[8] Documento de validação, documento_validacao_sucesso_escolar_2014-06-18.pdf

[9] Diagramas de *Gantt*, documento_diagramas_gantt_2014-06-27.pdf

10 Referências

[1] KIMBALL, R., ROSS, M. 2002. The Data Warehouse Toolkit – A Complete Guide to Dimensional Modeling (Seconde Edition). John Wiley and Sons, Inc., New York, Chichester, Weinheim, Brisbane, Singapore, Toronto.

[2] KIMBALL, R., CASERTA, J. 2004. The Data Warehouse ETL Toolkit – Practical Techniques for Extracting, Cleaning, Conforming, and Delivering Data. Wiley Publishing, Inc., Indianapolis.

[3] Relatório de gestão e contas consolidado de 2012.
http://www.uc.pt/dpgd/doc_gestao/relatorio_gestao_contas_consolidado_UC_2012.pdf