



Universidad de Murcia

Facultad de Informática

Cuadro de Mandos de Rendimiento Académico en Centros

Trabajo Fin de Máster

Moisés Frutos Plaza

Estudiante de Máster en Tecnologías de Análisis de
Datos Masivos: BIG DATA

Murcia, Julio 2017



Departamento de Informática y Sistemas

Facultad de Informática

Universidad de Murcia

Cuadro de Mandos de Rendimiento Académico en Centros

Trabajo de Fin de Máster

Moisés Frutos Plaza

Estudiante de Máster en Tecnologías de Análisis de
Datos Masivos: BIG DATA

Directores: Pedro Miguel Ruiz Martínez
y José Manuel Juárez Herrero

Murcia, Julio 2017

Cuadro de Mandos de Rendimiento Académico en Centros

Trabajo de Fin de Máster
Universidad de Murcia

Murcia, Julio 2017

La composición del texto ha sido realizada usando \LaTeX

Moisés Frutos Plaza
Estudiante de de Máster en Tecnologías de Análisis de Datos Masivos:
BIG DATA

Directores: Pedro Miguel Ruiz Martínez
y José Manuel Juárez Herrero
Doctores en Informática

Facultad de Informática
Departamento de Ingeniería de la Información y las Comunicaciones

Universidad de Murcia
Campus de Espinardo
30100 Murcia

Teléfono: (+34) 868884311
Fax: (+34) 868884151
Correo electrónico: moises.frutos@um.es

MENOS ES MÁS
Mies van der Rohe

ERRARE HUMANUM EST,
SED PERSEVERARE DIABOLICUM
Séneca

Resumen

Este *Trabajo de Fin de Máster*, en adelante *TFM*, se enmarca dentro del dominio de *Inteligencia del Negocio*, en concreto para la ayuda en la toma de decisiones en Centros Académicos Universitarios, principalmente Facultades de la Universidad de Murcia, de ahí que el nombre del proyecto sea *FacultatUM*, del genitivo plural del latín que indica “*de las facultades*” y al mismo tiempo se resalta el sufijo “*UM*” en mayúsculas al ser una herramienta por y para la Universidad de Murcia. Con esta herramienta se pretende asistir, en lo posible, al equipo decanal de cada facultad, a la hora de la toma de decisiones correctivas, que ayuden a conseguir los objetivos fijados en los resultados académicos anuales, o cuanto menos mejorar los anteriores.

Este TFM detalla como se lleva a cabo el análisis de los objetivos prefijados por decanato para este proyecto (ver Apéndice [A](#)), el análisis del dominio del problema donde dicho proyecto tiene lugar, el estado actual del arte en el que se contemplan las distintas herramientas que podrían satisfacer las necesidades de *FacultatUM*, la herramienta finalmente elegida así como la realización del diseño y la implementación del proyecto en la misma. Y, por supuesto, sin olvidarnos de las razones de su elección. Todo ello se irá desarrollando por varios capítulos, que seguirán el orden cronológico del proyecto según se ha ido realizando.

Agradecimientos

En primer lugar, quiero agradecer a mis tutores Pedro Miguel Ruiz Martínez y José Manuel Juárez Herrero su tiempo, sus consejos y apoyo en el planteamiento del presente TFM, pero sobre todo al final de la entrega del mismo. Gracias por todos vuestros ánimos.

En segundo lugar, a la doctora en Psiquiatría Doña Marina Sánchez por haberme permitido continuar hasta aquí pese a la grave enfermedad que padezco.

En tercer lugar, quiero darles las gracias, de corazón, a todos aquellos, ya sean PDI, PAS, alumnos ingresados y egresados, que, de corazón, con su trabajo e ilusión diarias hacen de este centro una facultad mejor. Porque al final, somos todos nosotros quienes creamos y formamos FIUM, y las relaciones humanas también son importantes. Todos cometemos errores, yo el primero, pero si éstos han sido debidos al ánimo de mejorar, bienvenidos sean.

En último lugar, pero no por ello menos importante, quiero dar las gracias a mis padres por todo, pero en especial por haberme cuidado en este año tan particularmente malo y complicado para mí.

Moisés Frutos Plaza

Índice general

Resumen	V
Agradecimientos	VII
Índice de figuras	XI
Notaciones	XIII
1. Introduccion	1
2. Análisis de Objetivos, Subobjetivos y Requisitos Previos	3
2.1. Objetivo Principal Propuesto	3
2.2. Sub-Objetivos Propuestos	4
2.3. Requisitos Previos	4
2.4. Fuera del alcance del proyecto	4
2.5. Análisis	5
3. Análisis del Dominio del Problema	7
3.1. Dominio del Problema	7
3.2. Conclusión	11
4. Elección de Indicadores Clave y Métricas	13
4.1. Indicadores Clave de Resultado (Key Result Indicators o KRIs)	13
4.2. Indicadores Secundarios y Métricas Técnicas tanto de Proceso como de Resultado	14
4.3. Indicadores Clave de Proceso (Key Performance Indicators o KPIs)	16
4.4. Conclusión	16
5. Análisis de Herramientas	17
5.1. Pentaho 5.4 Community	17
5.2. Qlik Sense	19
5.3. Python 2.7 + Matplotlib + Jupyter Notebook + Dashboard Jupyter	20
5.4. Pentaho 7.1 Community	21
5.5. Elección de las herramientas y sus razones	23

6. Implementación del Proyecto	25
6.1. Planificación y Decisiones de Diseño del Proyecto	25
6.2. Desarrollo y Despliegue del Proyecto	28
7. Conclusiones Finales y Vías Futuras	35
7.1. Conclusiones Finales	35
7.2. Vías futuras	36
A. Objetivos Prefijados por Decanato y Estimaciones para el Máster de Big Data	39
B. Ficheros del Trabajo	43
C. Datos de los equipos empleados	45
C.1. Equipo donde se ha desarrollado	45
C.2. Equipo donde se ha desplegado	45
Bibliografía	47

Índice de figuras

1.1. Cuadro de Mandos Directivo	2
5.1. Pentaho Community 5.4	18
5.2. Qlik Sense	19
5.3. Python Jupyter Notebook con extensión para desplegar en Dashboard Server	21
5.4. Pentaho Community 7.1	23
6.1. Diagrama Multidimensional para el Data Warehouse	26
6.2. Esquema del Cubo OLAP Mondrian	27
6.3. Pentaho Server Back-End desde donde desarrollar Cuadros de Mandos o Ejecutar Consultas	30
6.4. Consulta MDX hecha en Saiku en modo Visual	31
6.5. Misma Consulta MDX que la anterior hecha en Saiku pero en modo Manual	32
6.6. Cuadro de Mandos Directivo o de Nivel 1	33
6.7. Cuadro de Mandos Técnico o de Nivel 2	33
A.1. Objetivos y Requisitos Prefijados por Decanato	40
A.2. Estimaciones Esperadas para las tasas del Máster de Big Data	41

Notaciones

Inteligencia del Negocio

- *PI*: Proviene de las siglas en inglés Process Indicator, y como su propio nombre indica es un Indicador de Proceso relativo al funcionamiento de una organización o centro durante un tiempo concreto.
- *RI*: Proviene de las siglas en inglés Result Indicator, y como su propio nombre indica es un Indicador de Resultado relativo a los hitos obtenidos tras el funcionamiento de una organización o centro después de un determinado período de tiempo.
- *KPI*: Proviene de las siglas en inglés Key Process Indicator, y hace referencia a aquel indicador de proceso que es **clave** para el correcto funcionamiento de una organización o centro. Por ejemplo, si debido a la capacidad de producción de una organización se espera una tasa de producción diaria de 2 millones de tapas, y al final del día sólo se han producido 1 millón, algo en el proceso productivo no está yendo bien. Por lo que no es necesario esperar al final del ejercicio fiscal del año 2017 para tomar alguna medida correctiva, sino al día siguiente mismo. Por tanto permite tomar decisiones y medidas *durante* el proceso.
- *KRI*: Proviene de las siglas en inglés Key Result Indicator, y hace referencia a aquel indicador de resultado que es **clave** para determinar si los hitos obtenidos, tras el funcionamiento de una organización o centro después de un determinado período de tiempo, se ajusta o no a las expectativas esperadas. Por ejemplo, un KRI sería el beneficio neto obtenido tras el ejercicio fiscal de 2017, si éste fuese negativo probablemente no se ajustaría a las expectativas esperadas. Este indicador suele consultarse *después* del proceso como resultado del mismo.
- *Cubo OLAP*: Un cubo OLAP es una estructura de datos que permite un análisis rápido de datos de acuerdo con las múltiples Dimensiones que definen un problema empresarial (o institucional, como es nuestro caso). Un cubo multidimensional para un informe de ventas podría estar compuesto, por ejemplo, por 7 Dimensiones: Vendedor, Cantidad de Ventas, Región, Producto, Día, Mes, Año [1].

- *Mondrian*: Es un servidor de procesamiento analítico en línea (OLAP) de código abierto escrito en JAVA, el sistema responde a las consultas lo suficientemente rápido como para permitir una exploración interactiva de los datos, incluso si tienen millones de registros que ocupan varios gigabytes. Mondrian lleva el análisis multidimensional a las masas, permitiendo que los usuarios examinen datos del negocio perforando (dentro del cubo) y haciendo tabulación cruzada de la información (que hay dentro de él) [2].
- *Consulta MDX*: MDX viene de las siglas en inglés MultiDimensional eXpressions (Expresiones Multidimensionales). En una consulta MDX, la instrucción SELECT especifica un conjunto de resultados que contiene un subconjunto de datos multidimensionales que se han devuelto desde un cubo [3].

Programación Web

- *Cliente*: Un equipo cliente es el que suele enviar peticiones a un equipo servidor y esperar la respuesta de éste para mostrar la información por pantalla. Por ejemplo, esto es lo que hace un navegador web.
- *Servidor*: Es el equipo encargado de recibir y gestionar las peticiones de los equipos cliente, y responderles con la información adecuada. Por ejemplo, servidor web [Google](#).
- *Back-End*: Capa para el Desarrollador de una aplicación web como, por ejemplo, Pentaho, Joomla, Wordpress, etc...
- *Front-End*: Capa visible para el Usuario Final de la aplicación web, en la que quedan ocultas todas las herramientas de desarrollo. Por ejemplo, una página web dinámica con formularios, blogs, artículos, gráficos, vídeos y otros contenidos multimedia al servicio del usuario.

Capítulo 1

Introduccion

El presente Trabajo Fin de Máster (TFM) se enmarca dentro del dominio de la Inteligencia del Negocio, pero con una particularidad, que no es para un negocio, o al menos no para una organización con ánimo de lucro, sino para una institución pública con ánimo de mejorar sus resultados académicos. Por lo que este trabajo pretende llevar a cabo una extrapolación del ámbito del negocio, al ámbito académico utilizando las mismas técnicas.

A priori, el proyecto, no es un proyecto demasiado ambicioso ni pretencioso, sólo busca aprovechar la Inteligencia del Negocio en el ámbito académico con el ánimo de mejorar. No obstante, eso no significa, aunque lo parezca, que esto es algo sencillo. Primero porque una institución pública, a diferencia de una organización privada, está sujeta a una serie de leyes y normativas que se han de contemplar y tener en cuenta. Y, segundo, porque a medida que se avanza en el proyecto, uno se da cuenta de que éste puede alcanzar la envergadura que uno desee, lo que significa una ganancia en complejidad y extensión, que debido al tiempo finito de este TFM no se puede acometer, pero sí que se puede poner una primera piedra para que dicho proyecto sea extensible en el futuro. Es decir, que más adelante se pueda extender y personalizar con las necesidades de cada centro (facultad).



Figura 1.1: Cuadro de Mandos Directivo

Por tanto, éste es el propósito de este TFM. Realizar unos cuadros de mandos básicos para el rendimiento en centros académicos que sienta la base para algo que puede ser mucho mayor en el futuro. En los capítulos sucesivos, iremos viendo de forma cronológica como se ha ido analizando, diseñando, e implementando el proyecto; es decir, todo lo que es el Proceso de Desarrollo del Software. También veremos qué herramientas teníamos a nuestra disposición a día de hoy y cuales hemos elegido finalmente, y porqué, analizando sus ventajas e inconvenientes. Y, por último, extraeremos una serie de conclusiones en cuanto a lo que se ha podido conseguir con la realización del proyecto *FacultatUM*, y las posibilidades que quedan abiertas para su futuro.

Capítulo

Análisis de Objetivos, Subobjetivos y Requisitos Previos

En este capítulo, se analizarán los objetivos propuestos para este TFM por el decanato de la Facultad de Informática de la Universidad de Murcia, con el fin de realizar un proyecto de Cuadro de Mandos basado en Inteligencia del Negocio para visualizar el Rendimiento Académico del centro y servir de ayuda para la toma de decisiones al equipo decanal. Se ha acordado en denominar a este proyecto *FacultatUM*, ya que el nombre lleva intrínsecamente la palabra latina *facultatum* ("de las facultades") y al mismo tiempo el sufijo -UM indicativo de la Universidad de Murcia.

El proyecto se ha pensado en un principio teniendo en cuenta tanto la ley vigente *Real Decreto 1393/2007* [4] como los propios *Estatutos de la Universidad de Murcia* [5], así como los *Indicadores de Calidad de Referencia de la propia UM* [4]; no obstante, la elección final de los indicadores clave ha seguido como criterio principal la granularidad temporal en dos fases: Convocatoria y Curso Académico. Ya que hay decisiones correctivas que pueden tomarse durante el curso académico, y decisiones que se toman al final del mismo. De todo ello se hablará en profundidad más adelante.

2.1. Objetivo Principal Propuesto

El objetivo principal es solamente uno, pero uno que engloba a todos los demás. Literalmente dice así:

- *"Construcción de herramienta de apoyo a la toma de decisiones en el*

ámbito de un centro universitario a partir de resultados académicos.”

2.2. Sub-Objetivos Propuestos

Aparte del objetivo principal se proponen los siguientes objetivos secundarios:

- Identificar KPIs, KRIs, y otros Indicadores Secundarios o Métricas Técnicas.
- Preparación de los datos (proceso ETL).
- Desarrollo cuadro de mandos integral.
- Evaluación de la herramienta.

2.3. Requisitos Previos

El Decanato establece una serie de requisitos previos antes de la realización del proyecto y son los siguientes:

- El TFM se centrará en el desarrollo de una herramienta, principalmente visual, aunque otros aspectos podrán ser tenidos en cuenta.
- El volcado de datos es muy puntual, por tanto la preparación de datos será offline.
- El cuadro de mandos podrá ser una aplicación de escritorio o web.
- El proceso de preparación de datos y el cuadro de mandos deberán ser multiplataforma (OSX, Windows, Linux).
- Utilización de herramientas gratuitas, de licencia libre, o licencia académica.
- El sistema deberá ser ligero y no deberá requerir de un despliegue complejo, pudiendo ser ejecutable en un ordenador de capacidad limitada.

2.4. Fuera del alcance del proyecto

A su vez, el Decanato, limita el alcance del proyecto y deja fuera del mismo, y, por tanto, del presente TFM, lo siguiente:

- Análisis de datos de rendimiento académico (no hay datos reales disponibles).
- La validación de la herramienta en el centro.

No obstante, estas cosas podrían contemplarse dentro de las vías futuras.

2.5. Análisis

A priori parece que se pide un proyecto sencillo dividido en dos grandes pasos, por un lado la preparación de los datos mediante un proceso ETL y por el otro la creación de un Cuadro de Mandos. No obstante, más adelante conforme el proyecto se vaya desarrollando se podrá comprobar que no todo es tan sencillo como se pinta al principio.

Por ejemplo, sólo con el proceso de ETL ya van surgiendo dificultades que, lo que a priori parecía sencillo, ya no es tan sencillo, puesto que la preparación de los datos exige un rigor científico y técnico, y una visión en profundidad, para que los datos guardados finalmente en el DW, aunque no sean reales, por lo menos sean verosímiles y compatibles. Por suerte, desde Decanato se asegura que con los datos reales no tendremos ningún tipo de problemas, ya que los datos ya se encuentran en una base de datos real. No obstante, tomar estas precauciones en pos de la consistencia del proyecto nunca está demás, ya que las fuentes de datos pueden presentar registros erróneos debido a errores humanos, ya que nadie está exento de introducir sin querer mal los datos de un registro, y siempre es bueno que haya algún tipo de log o visor que nos informe del porqué no se han cargado los datos, a ser posible identificando todos los errores (esto como veremos más adelante lo hemos conseguido mediante transformaciones Kettle).

De todos modos, tampoco hay que ser excisivamente meticulosos en el proceso de ETL, ya que el tiempo para la realización del proyecto es finito, y en los requisitos previos se deja claro que la herramienta será principalmente visual. Por lo que la parte del Cuadro de Mandos si cabe es más importante aún que la del proceso de ETL. Por lo que es necesario hallar una solución de compromiso para que, dentro del poco tiempo que se dispone, darle cierta consistencia a los datos del proyecto toda vez que nos centramos en la correcta Visualización de los Datos dentro del Cuadro de Mandos.

Una vez analizados el objetivo principal, los subobjetivos y los requisitos previos, y siendo siempre conscientes del tiempo que disponemos, el siguiente paso es situar el proyecto dentro del contexto del Dominio del Problema. Y justamente eso es lo que veremos en el capítulo siguiente.

Capítulo 3

Análisis del Dominio del Problema

En este capítulo se hará un análisis del dominio del problema para situar el proyecto dentro de su contexto.

3.1. Dominio del Problema

El dominio del problema trata sobre el Rendimiento Académico dentro de un Centro perteneciente a la Universidad de Murcia, específicamente dentro de lo que es una Facultad.

Dentro de una facultad, el Decano y el Equipo Decanal son los encargados de gestionar, entre otras cosas, dicho rendimiento académico. Pero esa gestión no se puede hacer de forma arbitraria, ya que al ser la Universidad de Murcia una Institución Pública Oficial del Estado, que está sujeta al Ministerio de Educación, Ciencia y Deporte, se debe de tener en cuenta en la gestión el Real Decreto [4] que existe al respecto de los resultados académicos, que se esperan tras el curso académico, y también los propios estatutos internos de la Universidad de Murcia [5], que establece tanto las funciones de los centros como las del decano y el equipo decanal, y, por ende, el margen de maniobra que éstos disponen para aplicar o tomar decisiones correctivas para mejorar el rendimiento académico dentro de la normativa.

Además, con el fin de mejorar el rendimiento académico, la Unidad para la Calidad de la propia Universidad de Murcia establece un procedimiento [6] que establece, a su vez, una serie de indicadores con los resultados académicos que la propia institución espera, y que, por supuesto, también han de contemplarse.

El análisis del dominio se centrará en ver de forma más detallada cada uno de los aspectos antes mencionados y como nuestro proyecto podría encajar en los mismos.

3.1.1. Análisis del Real Decreto

En el Real Decreto 1393/2007 [4], de 29 de octubre, se establece la ordenación de las enseñanzas universitarias oficiales. En dicha ordenación se establecen también los Resultados Previstos:

- Estimación de valores cuantitativos para los indicadores
 1. *Tasa de graduación*: porcentaje de estudiantes que finalizan la enseñanza en el tiempo previsto en el plan de estudios o en un año académico más en relación a su cohorte de entrada.
 2. *Tasa de abandono*: relación porcentual entre el número total de estudiantes de una cohorte de nuevo ingreso que debieron obtener el título el año académico anterior y que no se han matriculado ni en ese año académico ni en el anterior.
 3. *Tasa de eficiencia*: relación porcentual entre el número total de créditos del plan de estudios a los que debieron haberse matriculado a lo largo de sus estudios el conjunto de graduados de un determinado año académico y el número total de créditos en los que realmente han tenido que matricularse.
- Procedimiento general de la Universidad para valorar el progreso y los resultados de aprendizaje de los estudiantes en términos de las competencias
 1. Resultados de Pruebas Externas.
 2. Trabajo de Fin de Grado.
 3. Trabajo de Fin de Máster.

El Estado solicita principalmente los porcentajes de Tasa de Graduación, Tasa de Abandono, y Tasa de Eficiencia al final del curso académico, sin especificar cual debería de ser la cantidad recomendada de cada una de ellas, aunque es fácilmente deducible que cuanto *mayores* sean las tasas de Graduación y Eficiencia, y cuanto *menor* sea la de Abandono *mejor*. Lo contrario sería cuanto menos *preocupante*.

Por otra parte, deja en manos de cada una de las universidades un procedimiento general o marco, para evaluar el rendimiento de los alumnos en términos de las competencias adquiridas, que normalmente se suelen realizar

mediante pruebas externas, trabajos de fin de grado y de máster cuyos criterios de evaluación son muy subjetivos y varían, no sólo entre universidades, sino entre facultades e incluso miembros de un tribunal de una misma facultad. Debido a la imposibilidad de cuantificar los distintos criterios subjetivos de cada tribunal y/o sus miembros, el proyecto FacultatUM no va a entrar en esta última parte, y se va a centrar exclusivamente en la primera parte (estimación de los valores cuantitativos para los indicadores).

3.1.2. Análisis de los Estatutos de la Universidad de Murcia

En los Estatutos de la propia Universidad de Murcia [5], en su capítulo IV: De los Centros, Departamentos, Institutos y otras estructuras, Sección 1ª, podemos extraer las funciones que deben desempeñar tanto los centros como el decano y su equipo decanal. Esto es necesario analizarlo para conocer las posibles decisiones a las que el proyecto facultatUM puede asistir, para que éstas siempre estén dentro de los estatutos de la institución.

■ Artículo 52. Naturaleza y funciones de los Centros

1. Las Facultades, Escuelas Técnicas o Politécnicas Superiores y Escuelas Universitarias o Escuelas Universitarias Politécnicas, son los centros encargados de la organización de las enseñanzas y de los procesos académicos, administrativos y de gestión conducentes a la obtención de títulos de carácter oficial y validez en todo el territorio nacional, así como de aquellas otras funciones que determinen los presentes Estatutos y las normas que los desarrollen. Estos Centros estarán constituidos por el personal docente e investigador, estudiantes y personal de administración y servicios que se les adscriba. El Consejo de Gobierno regulará el procedimiento de adscripción a los Centros.
2. Son otras funciones de los Centros:
 - a) Elaborar los planes de estudio que conduzcan a la obtención de los títulos impartidos en el Centro.
 - b) Organizar y coordinar las actividades docentes así como gestionar los servicios propios de apoyo a la enseñanza y, en su caso, a la investigación.
 - c) Promover actividades de formación continua y extensión universitaria.
 - d) Proponer la plantilla necesaria para llevar a cabo el plan docente, la administración y los servicios del Centro.

- e) Gestionar el presupuesto del Centro y supervisar su correcta aplicación.

■ Artículo 58. Decanos y Directores de Centros

1. Los Decanos de Facultad y Directores de Escuela ejercen las funciones de representación, dirección y gestión ordinaria de su Centro. Son nombrados por el Rector, a propuesta de la Junta de Facultad o Escuela.
2. Son otras funciones del Decano o Director:
 - a) Ejecutar los acuerdos de la Junta de Centro.
 - b) Proponer al Rector el nombramiento y cese de los Vicedecanos o Subdirectores y del Secretario del Centro.
 - c) Informar a la Junta de Centro de los nombramientos y ceses de Vicedecanos o Subdirectores y del Secretario del Centro.
 - d) Coordinar la actividad de los Vicedecanos o Subdirectores y del Secretario del Centro.
 - e) Velar por el cumplimiento de las obligaciones académicas y laborales de los miembros del Centro y por la aplicación, en su caso, del régimen disciplinario.
 - f) Cualquier otra función que le sea conferida en los presentes Estatutos y normas que los desarrollen así como las referidas a todos los demás asuntos propios del Centro que no hayan sido expresamente atribuidas a otros órganos.
3. El Decano o Director del Centro podrá ser eximido parcialmente por el Rector de sus obligaciones académicas, sin detrimento de su régimen de dedicación.

Conociendo la normativa tanto sobre facultades como sobre el decanato de las mismas, podemos orientar la creación del Cuadro de Mandos del proyecto con el fin de que asista de la mejor manera posible a las posibles decisiones correctivas sobre las funciones aquí contempladas.

3.1.3. Análisis del Procedimiento para los Resultados Académicos de la Unidad para la Calidad

Por último debemos de analizar los Resultados Académicos esperados por la propia Unidad para la Calidad de la Universidad de Murcia [6], ya que de por sí nos aportan indicadores y métricas de Rendimiento Académico, que podemos usar en el proyecto, y que resultan de mucha utilidad tanto para la facultad como para decanato.

1. IN01-PC05 Ficha Indicador Tasa de rendimiento

2. IN02-PC05 Ficha Indicador Tasa éxito
3. IN03.1-PC05 Ficha Indicador Tasa de graduación en la duración del plan de estudios, n
4. IN03.2-PC05 Ficha Indicador Tasa de graduación (n+1)RD 1393-2007
5. IN041-PC05 Ficha Indicador Tasa de Abandono RD
6. IN04.2-PC05 Ficha Indicador Tasa de Abandono REACU
7. IN04.3-PC05 Tasa de abandono en el curso siguiente al de ingreso
8. IN05-PC05 Ficha Indicador tasa eficiencia
9. IN06-PC05 Ficha Indicador duración media estudios
10. IN08-PC05 Ficha Indicador n^o de estudiantes matriculados

Entre las tasas que hemos visto, podemos destacar las de graduación, abandono y eficiencia, ya que estas aparecen también contempladas en el Real Decreto, pero aparte hay otras métricas interesantes como las tasas de rendimiento, éxito, abandono en el curso siguiente al ingreso, o duración media de los estudios, de las que se puede extraer información adicional importante a la hora de la toma de decisiones en el futuro.

3.2. Conclusión

Conociendo el dominio del problema y los resultados que se esperan, es el momento de identificar cuales son los indicadores clave, tanto de resultado como de proceso, así como los indicadores secundarios y métricas técnicas que nos puedan ayudar a conseguir los objetivos fijados dentro del marco del dominio del problema. Esto justo es lo que veremos en capítulo siguiente.

Capítulo 4

Elección de Indicadores Clave y Métricas

En este capítulo se seleccionarán aquellos indicadores que consideramos clave, por una parte durante el proceso del curso académico (KPI), y, por otra parte, aquellos una vez ha finalizado el curso académico (KRI). Además seleccionaremos otros indicadores secundarios y/o métricas técnicas que nos puedan aportar información adicional y detallada. Para ello se va a seguir, en proporcionalidad, la *Regla del 10/80/10 de Kaplan y Norton* (10 KRIs/80 RIs y PIs/10 KPIs) [7].

4.1. Indicadores Clave de Resultado (Key Result Indicators o KRIs)

En esta sección es donde tiene lugar el primer "10" de la Regla de Kaplan y Norton, y es aquí donde se seleccionan los indicadores clave de resultado para, a la hora de hacer balance, saber *qué ha pasado*, en nuestro caso cuales han sido los resultados del curso académico y a partir de ahí hacer una valoración sobre el rendimiento de los mismos (si ha sido positivo, negativo, o no ha cambiado respecto al anterior balance).

1. **Tasa de graduación:** porcentaje de estudiantes que finalizan la enseñanza en el tiempo previsto en el plan de estudios o en un año académico más en relación a su cohorte de entrada.
2. **Tasa de abandono:** relación porcentual entre el número total de estudiantes de una cohorte de nuevo ingreso que debieron obtener el título el año académico anterior y que no se han matriculado ni en ese año académico ni en el anterior.

3. **Tasa de eficiencia:** relación porcentual entre el número total de créditos del plan de estudios a los que debieron haberse matriculado a lo largo de sus estudios el conjunto de graduados de un determinado año académico y el número total de créditos en los que realmente han tenido que matricularse.

4.2. Indicadores Secundarios y Métricas Técnicas tanto de Proceso como de Resultado

En esta sección es donde tiene lugar el "80" de la Regla de Kaplan y Norton, y justo aquí es donde se indica todos esos indicadores y medidas secundarias que nos pueden aportar información adicional, pero que no son claves ni en el proceso ni en el resultado. Por tanto, es aquí donde tienen lugar lo que denominamos *métricas técnicas*, las cuales dan detalles técnicos de los indicadores clave, que tal vez no sean especialmente relevantes para la dirección de una organización, pero sí para el departamento técnico encargado de dar una explicación del porqué uno o varios indicadores clave no están yendo como tienen que ir. En Inteligencia de Negocio a estos indicadores secundarios con métricas técnicas se les cataloga como indicadores de nivel 2 o nivel técnico (el nivel directivo sería el nivel 1, y éste suele centrarse en los indicadores clave).

No obstante, para nuestro proyecto FacultatUM también son muy importantes estos indicadores, ya que no sólo nos interesa saber *qué esta pasando* sino *porqué está pasando*. O lo que es lo mismo, el decanato del centro no sólo precisa de los indicadores de nivel 1 (directivos) sino también de los indicadores de nivel 2 (técnicos), ya que el decano se ocuparía del cuadro de mandos directivo, y, en el caso de encontrar que hay problemas, delegar en un vicedecano u otro miembro del equipo decanal, el encontrar los porqués de esos problemas detectados, usando estos últimos para ello la trazabilidad que les permitiría el cuadro de mandos técnico. Mientras que el decano seguiría con el cumplimiento de sus otras obligaciones y funciones.

Los indicadores secundarios que hemos encontrado interesantes son:

1. Indicador por Medio de Acceso
2. Indicador por Centro de Procedencia
3. Indicador por Código Postal
4. Indicador de Tasa de Aprobados por Grupo
5. Indicador de Nota Media por Grupo
6. Indicador de Nota Numérica Media por Grupo

7. Indicador de Número de Convocatorias Presentadas hasta Aprobar por Grupo
8. Indicador de Tasa de Aprobados en Alumnos de Nuevo Ingreso por Grupo
9. Indicador de Tasa de Aprobados en Alumnos Repetidores por Grupo
10. Indicador de Tasa de Aprobados por Asignatura
11. Indicador de Nota Media por Asignatura
12. Indicador de Nota Numérica Media por Asignatura
13. Indicador de Número de Convocatorias Presentadas hasta Aprobar por Asignatura
14. Indicador de Tasa de Aprobados en Alumnos de Nuevo Ingreso por Asignatura
15. Indicador de Tasa de Aprobados en Alumnos Repetidores por Asignatura
16. Indicador de Tasa de Aprobados por Profesor
17. Indicador de Nota Media por Profesor
18. Indicador de Nota Numérica Media por Profesor
19. Indicador de Número de Convocatorias Presentadas hasta Aprobar por Profesor
20. Indicador de Tasa de Aprobados en Alumnos de Nuevo Ingreso por Profesor
21. Indicador de Tasa de Aprobados en Alumnos Repetidores por Profesor
22. Indicador de Tasa de Aprobados por Medio de Acceso (Selectividad, Rama, Mayores de 25 años, etc...)
23. Indicador de Nota Media por Medio de Acceso (Selectividad, Rama, Mayores de 25 años, etc...)
24. Indicador de Nota Numérica Media por Medio de Acceso (Selectividad, Rama, Mayores de 25 años, etc...)
25. Indicador de Número de Convocatorias Presentadas hasta Aprobar por Medio de Acceso (Selectividad, Rama, Mayores de 25 años, etc...)
26. Indicador de Tasa de Aprobados en Alumnos de Nuevo Ingreso por Medio de Acceso (Selectividad, Rama, Mayores de 25 años, etc...)

27. Indicador de Tasa de Aprobados en Alumnos Repetidores por Medio de Acceso (Selectividad, Rama, Mayores de 25 años, etc...)

4.3. Indicadores Clave de Proceso (Key Performance Indicators o KPIs)

Esta sección es la más importante de todas, ya que es donde tiene lugar el último "10" de la Regla de Kaplan y Norton, y es aquí donde se seleccionan los indicadores clave **durante** el proceso, para precisamente tomar las decisiones correctivas a tiempo; es decir, antes de que termine el curso académico en nuestro caso. Esta sección se centra por tanto en saber *qué está pasando*, detectar un problema si lo hubiere y entonces tomar las medidas que se consideren necesarias por parte del equipo directivo, en nuestro caso el decanato.

Hemos detectado dos indicadores de proceso claves que podemos ver en cada una de las convocatorias:

1. Tasa de Presentados/Matriculados
2. Tasa de Aprobados/Presentados

Si, por ejemplo, se matriculan 8 alumnos en el presente máster, y sólo se presentan 4 de media en la primera convocatoria, ésto es claramente indicativo de que hay un problema. Lo mismo sucede si de esos 4 que se presentan de media sólo 2 aprueban las asignaturas. En una organización el equipo directivo detectaría que existe uno o más problemas (nivel 1) y es cuando preguntaría al departamento técnico porqué están ocurriendo estos problemas (nivel 2). Como ya se ha comentado antes, el decanato asume los dos roles de modo que aparte de detectar los problemas también quieren encargarse de saber el porqué suceden. Y eso es lo que va a tratar de abordar el proyecto *Facultatum*.

4.4. Conclusión

Después de realizar el análisis del dominio del problema y teniendo siempre en cuenta los objetivos fijados al principio, hemos creído conveniente la elección de los KPIs, KRIs, PIs y RIs citados en las secciones anteriores del presente capítulo. Lo siguiente sería diseñar la aplicación, pero antes tenemos que conocer como está el Estado del Arte en la actualidad, y eso es justo lo que vamos a ver en el capítulo siguiente.

Capítulo 5

Análisis de Herramientas

En este capítulo se van a mencionar algunas de las herramientas más destacables enfocadas a la realización de un Cuadro de Mandos y a la Visualización de Datos del mismo. Finalmente se determinará la elección de la herramienta que hemos visto más interesante, y expondremos las razones del porqué.

5.1. Pentaho 5.4 Community

En realidad Pentaho 5.4 Community es una plataforma que incluye varias herramientas para Inteligencia de Negocio. Por una parte incorpora un servidor de Inteligencia de Negocio, por otro Spoon para Integración de Datos, Schema Workbench para el diseño de cubos OLAP en Mondrian, y el Community Dashboard Editor para la creación de cuadros de mandos. Además en el Marketplace hay una extensión del mismo denominada *Saiku* [8] que es muy útil para la construcción de consultas MDX en su licencia gratuita Community. Estas han sido las herramientas, y versiones, vistas en el presente curso en la asignatura de Inteligencia de Negocio. A continuación veremos sus ventajas e inconvenientes:

- Ventajas
 - Las versiones Community son totalmente Open Source y Libres.
 - Versatilidad a la hora de realizar y personalizar un Cuadro de Mandos basado en, por ejemplo, consultas MDX a un cubo tal y como se ha visto en este máster con Saiku.
 - Herramientas vistas y utilizadas en la asignatura del propio máster de Big Data: Inteligencia del Negocio.

- Aplicación web multiplataforma que permite gestión de usuarios y permisos. Lo cual, además de dar cierta seguridad a la hora de acceder a los datos, permite también separar el Back-End del Front-End.
- Inconvenientes
- La aplicación web trabaja internamente con Apache Tomcat y Mondrian para el despliegue de la aplicación web y la traducción de un cubo de datos respectivamente. Esto conlleva una serie de limitaciones técnicas, como que no se pueda asignar más memoria dinámica una vez la máquina virtual de Java ya se ha desplegado, y en el caso de Mondrian que no soporte las dimensiones Bridge dentro de un cubo (por suerte en el caso de nuestro cubo no hay ninguna dimensión Bridge, pero nunca está demás conocer esta limitación).
 - La versión de Pentaho no deja de ser Community, el único soporte es el de la comunidad; es decir, no hay un soporte profesional o Business por parte de Hitachi para esta versión. Tal vez para el presente TFM esto no suponga un problema, pero para un proyecto profesional de mayor envergadura para una gran organización sí que podría serlo. Además la comunidad ya no está dando tanto soporte a la versión 5.4, ya que se considera desfasada.
 - Aunque Saiku es un proyecto Open Source, requiere de licencia Community para su uso, el cual es limitado sólo a consultas MDX.
 - Pentaho 5.4 está hecho para trabajar con Java JDK 1.7, el cual ya está actualmente bastante desfasado, y **no** funciona con Java JDK 1.8.



Figura 5.1: Pentaho Community 5.4

5.2. Qlik Sense

Qlik Sense [9] es una herramienta orientada principalmente a la realización de aplicaciones de visualización de datos de forma moderna, profesional, interactiva y muy amigable con el usuario. La licencia gratuita ilimitada es sólo para la versión de escritorio, la cual no es multiplataforma, y la versión multiplataforma es en la nube con una licencia gratuita muy básica. A continuación se detallan sus ventajas e inconvenientes:

■ Ventajas

- Muy versátil y también trabaja con PostgreSQL.
- Más moderno, amigable e interactivo con el usuario.
- Licencia gratuita ilimitada para aplicación de escritorio y licencia gratuita básica para aplicación web en la nube.

■ Inconvenientes

- La aplicación de escritorio no se puede instalar en Linux. Por lo que pierde el sentido de multiplataforma que buscamos. La aplicación web en la nube sí es multiplataforma, pero la licencia básica sólo permite compartirla con hasta 5 usuarios y guardar hasta 100000 registros, y esto se puede quedar bastante corto para el cuadro de mandos de un centro.
- Es Privativo y, por tanto, no es Open Source, y esto podría derivar en problemas con las actualizaciones de versiones futuras en las que el centro debería de estar supeditado al soporte de la compañía del software.

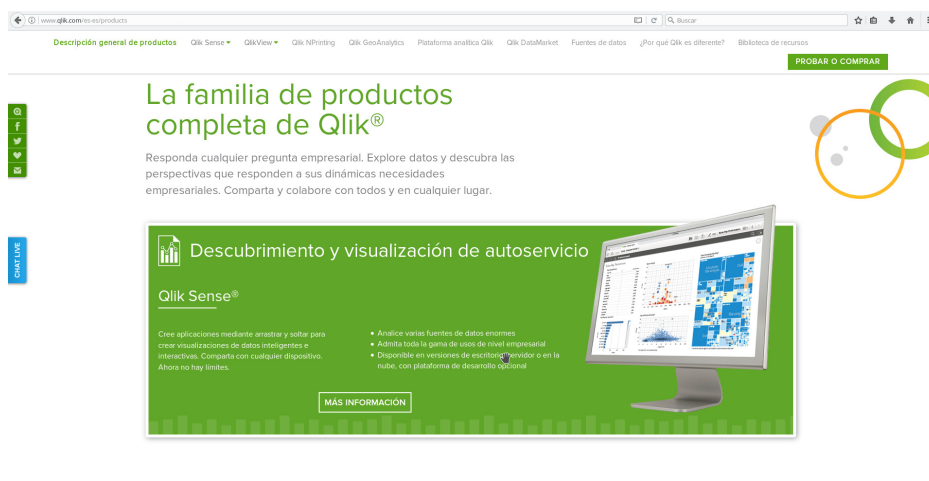


Figura 5.2: Qlik Sense

5.3. Python 2.7 + Matplotlib + Jupyter Notebook + Dashboard Jupyter

Ésta ha sido una de las opciones que hemos barajado desde el principio, pero las múltiples dependencias, las cuales dificultan una fácil configuración—parte de los objetivos prefijados por decanato—, y la no separación entre el Front-End (capa de usuario) y el Back-End (capa del desarrollador) nos hizo replantearnos otras alternativas. Aún así, qué duda cabe, son herramientas más que dignas de mención dentro de este capítulo de Análisis de Herramientas, ya que también ofrecen un abanico de grandes posibilidades y todo ello siendo software libre. Veamos tanto las ventajas como los inconvenientes a continuación:

■ Ventajas

- Entorno de desarrollo propio para construir cualquier tipo de aplicación a medida integrando cualquier tipo de fuente de datos. Por lo que serviría tanto para la creación del Proceso ETL como para la posterior creación del Cuadro de Mandos.
- Todas las herramientas empleadas son Open Source, lo cual de ser necesario pueden ser modificadas a conveniencia por nosotros mismos, añadiendo, por ejemplo, nuevas funcionalidades que necesitemos tanto en el presente como en el futuro (extensibilidad).
- Permite el uso de visualización de datos en gráficas tanto estáticas como animadas, e incluso interactivas con el usuario al estilo Matlab con Matplotlib.
- Aplicación web en sí misma, por lo que se puede acceder desde cualquier navegador web, lo que asegura el objetivo de que sea multiplataforma.

■ Inconvenientes

- No separa el Back-End del Front-End. Es decir, código Python y gráficas aparecen entremezcladas en la misma web. La herramienta Dashboard Jupyter permite hacer esa separación haciendo un despliegue del Cuadro de Mandos pero para un servidor Node.js [10], independiente de Jupyter. Por lo que ya tendríamos que desplegar hacia otra dependencia, que encima nada tiene que ver con Python ya que es Javascript. Esto cuanto menos es poco ortodoxo y escapa al requisito de sencillez de despliegue de la aplicación fijado para el presente TFM 2.3.
- Debido a lo anterior no hay gestión de usuarios predefinida, ni permisos que asignarles a estos, salvo el poner una contraseña a Jupyter Notebook con lo que sólo el administrador podría acceder,

pero no el resto de usuarios. Esto es muy peligroso ya que podría dejar abierta una brecha de seguridad en el que cualquiera podría modificar el código y acceder a datos sensibles protegidos por la Ley Orgánica 15/1999, de 13 de diciembre, de Protección de Datos de Carácter Personal [11].

- Es Privativo y, por tanto, no es Open Source, y esto podría derivar en problemas con las actualizaciones de versiones futuras en las que el centro debería de estar supeditado al soporte de la compañía del software.

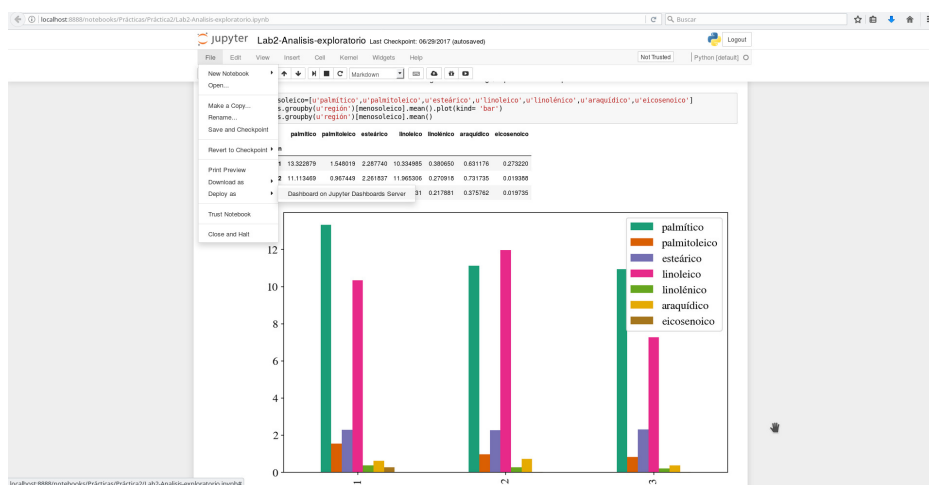


Figura 5.3: Python Jupyter Notebook con extensión para desplegar en Dashboard Server

5.4. Pentaho 7.1 Community

Pentaho 7.1 Community [12] es la última versión de Pentaho, y a diferencia de la vista en clase de la asignatura de Inteligencia de Negocio, esta versión sí que trabaja ya al menos con el JDK 1.8, lo cual es, ya de por sí, un avance bastante significativo con respecto a la versión 5.4 que sólo funcionaba con JDK 1.7. Por otra parte, la Comunidad ha puesto mucho esfuerzo en modernizar la extensión para la realización de Cuadros de Mando *Dashboard CDE* [13], la cual ya viene integrada dentro del propio servidor de Inteligencia de Negocio. Dicha extensión permite crear y personalizar cualquier tipo de Cuadro de Mandos a partir de varias fuentes de datos (Bases de Datos, Cubos Mondrian, o ficheros XML entre otros), tal y como se puede ver en esta [demostración](#).

Antes de ver las ventajas e inconvenientes cabe destacar que gracias al servidor de Inteligencia de Negocio no sólo se consigue la separación de capas

Back-End y Front-End sino el problema de la gestión de Usuarios y Permisos, que teníamos con la opción anterior. Y es por ésto que finalmente nos decantamos por el uso de estas herramientas para el desarrollo del proyecto *Facultatum*. A continuación podemos ver las ventajas e inconvenientes:

■ Ventajas

- Las versiones Community son totalmente Open Source y Libres.
- Aunque sean versiones más modernas, son las mismas herramientas que se han visto en la asignatura de Inteligencia de Negocio de este máster.
- Versatilidad a la hora de realizar y personalizar un Cuadro de Mandos basado en, por ejemplo, consultas MDX a un cubo tal y como se ha visto en este máster con Saiku.
- Versión más moderna y mejorada de la herramienta CDE para la creación de Cuadros de Mandos, la cual ya viene integrada en el servidor.
- Aplicación web multiplataforma que permite gestión de usuarios y permisos. Lo cual, además de dar cierta seguridad a la hora de acceder a los datos, permite también separar el Back-End del Front-End.
- Mayor soporte de la comunidad en el momento de realizar el presente TFM. De hecho algunas de las demostraciones abiertas hechas con esta herramienta son de este año 2017 (La demostración T-Wars es de Abril).

■ Inconvenientes

- La aplicación web trabaja internamente con Apache Tomcat y Mondrian para el despliegue de la aplicación web y la traducción de un cubo de datos respectivamente. Esto conlleva una serie de limitaciones técnicas, como que no se pueda asignar más memoria dinámica una vez la máquina virtual de Java ya se ha desplegado, y en el caso de Mondrian que no soporte las dimensiones Bridge dentro de un cubo (por suerte en el caso de nuestro cubo no hay ninguna dimensión Bridge, pero nunca está demás conocer esta limitación).
- La versión de Pentaho no deja de ser Community, el único soporte es el de la comunidad; es decir, no hay un soporte profesional o Bussiness por parte de Hitachi para esta versión. Tal vez para el presente TFM esto no suponga un problema, pero para un proyecto profesional de mayor envergadura para una gran organización sí que podría serlo.

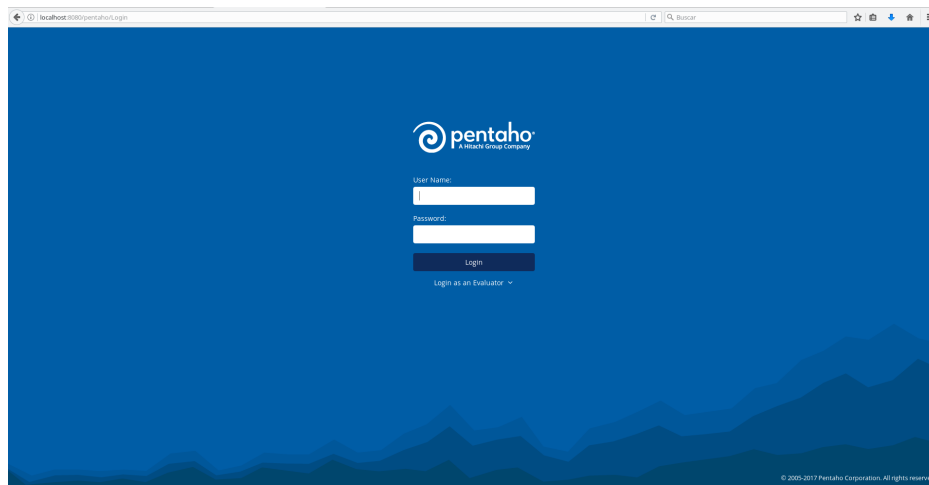


Figura 5.4: Pentaho Community 7.1

5.5. Elección de las herramientas y sus razones

Tras comprobar las Herramientas existentes y los Objetivos Fijados del TFM, creemos que la mejor solución de compromiso es el uso de *Pentaho 7.1 Community*. Las razones son que las herramientas que incorpora nos aportan versatilidad, modernidad, software libre con soporte de la comunidad, seguridad en el acceso al cuadro de mandos y a los datos, y una diferenciación necesaria entre la capa de usuario y la capa de desarrollo.

Una vez visto este análisis y elegidas las herramientas, el próximo paso es diseñar en torno a las mismas nuestro Cuadro de Mandos, justamente esto es lo que veremos en el siguiente capítulo.

Capítulo 6

Implementación del Proyecto

En este capítulo veremos como se planificó y se tomaron las decisiones de diseño del proyecto *Facultatum* en base a los objetivos prefijados por decanato y teniendo en cuenta también las herramientas finalmente elegidas para su posterior implementación, en la cual veremos como se desarrolló y finalmente se desplegó el proyecto en detalle.

6.1. Planificación y Decisiones de Diseño del Proyecto

Lo primero y fundamental de todo era diseñar el Data Warehouse (DW), que como su propio nombre en inglés indica, es el encargado del Almacenamiento de los Datos, y, por tanto, es una parte vital del diseño del proyecto.

Aunque físicamente los datos puedan estar almacenados en una Base de Datos SQL estándar, la lógica del diseño para organizar los datos se abstrae de ello, y ya no piensa en Entidad-Relación o Tercera Forma Normal como se acostumbraba a pensar anteriormente. Sino que ahora, en Inteligencia del Negocio por lo menos, se habla más bien de dimensiones, de hechos, y des-normalización de los datos de modo que permitan dos cosas fundamentales:

1. Acceso más rápido a los datos a la hora de realizar una consulta.
2. Recoger los cambios de estado de una entidad, por ejemplo, un alumno, un profesor, o una asignatura.

De ahí que como primer paso en el diseño se decidiese hacer un diagrama multidimensional para construir nuestro futuro DW (ver Figura 6.1). La herramienta elegida para su diseño fue *draw.io* [14] ya que es una herramienta

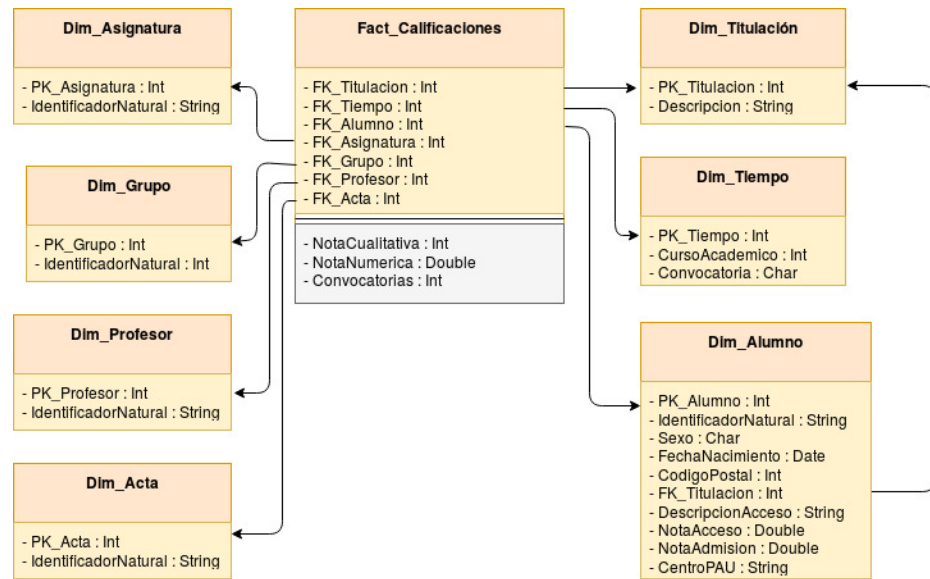


Figura 6.1: Diagrama Multidimensional para el Data Warehouse

fantástica que permite la construcción de cualquier tipo de diagrama de forma totalmente gratuita. El resultado final del diseño del DW se puede ver en la siguiente figura 6.1.

Una vez diseñado el diagrama multidimensional, el siguiente paso era elegir una base de datos que fuese lo suficientemente robusta toda vez que fuera software libre. En ese paso del diseño elegimos PostgreSQL debido a su robusted y potencial para almacenar los más de 120000 registros con los que contábamos de forma completamente segura, fiable y rápida. Otra de las razones que pesaron es que fue precisamente la base de datos que vimos en la asignatura de Inteligencia de Negocio de este máster. A partir de ahí diseñamos con el diagrama multidimensional el esquema de la base de datos, donde los hechos era una tabla central cuya clave primaria eran todas las claves foráneas hacia cada una de las claves primarias de las tablas dimensiones. En las tablas de las dimensiones registraríamos los datos relativos a las mismas, en muchas de ellas, tenemos un *Identificador Natural* para recoger los cambios de estado en varios registros de ser necesario (alumno que termina el grado en 2015, y se matricula para el máster de Big Data en 2016, mientras que su DNI, su Identificador Natural, sigue siendo el mismo, pero la titulación que cursa ya no, por tanto se crea nuevo registro, y se da a lugar a una desnormalización). Una vez diseñado el esquema de la base de datos para que albergue el Data Warehouse, se realiza un script en SQL con la base de datos aún vacía, para, que en el caso de que fuera necesario, cambiar algo, borrarla y crearla de nuevo desde cero.

El siguiente paso en el diseño era ver como íbamos a hacer el proceso

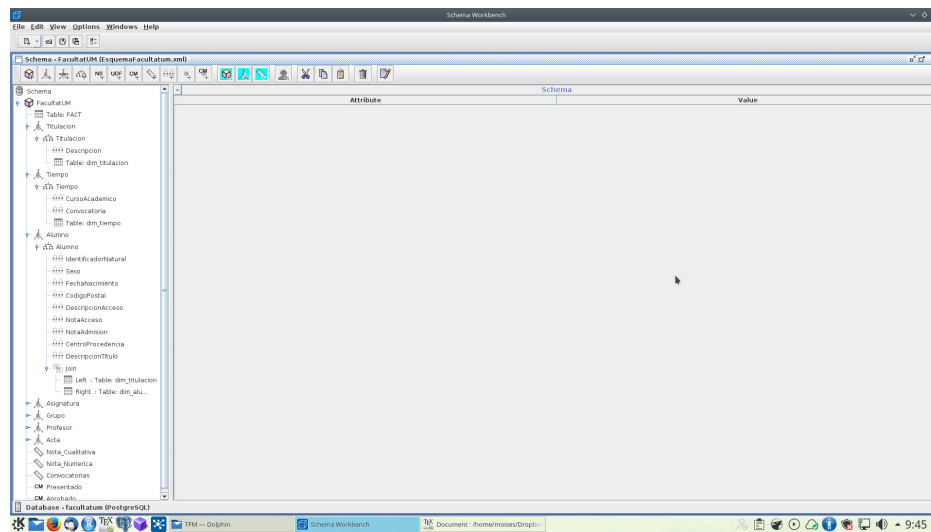


Figura 6.2: Esquema del Cubo OLAP Mondrian

ETL de carga de los datos desde los Excels proporcionados a las tablas del DW. La decisión de diseño estribaba entre hacerla con una metodología más programática (Python), o menos programática pero a la vez más práctica (Transformaciones y Trabajos Kettle con Spoon). Al final nos decantamos por la segunda opción, porque Spoon ya venía integrado en la plataforma Pentaho como Data Integration, que era en donde al final íbamos a implantar el proyecto. Además, los pasos y los logs quedaban mucho más claros en Kettle, por lo que su depuración era más exacta y sencilla.

Por último, nos quedaba tomar otra decisión de diseño importante. El modo, o los modos, cómo íbamos a acceder a los datos. Pentaho permite las consultas SQL al DW, pero también la construcción de un cubo OLAP Mondrian en el que se pueden realizar consultas MDX ¿Hasta que punto eran convenientes unas u otras? Al final decidimos que puesto que la plataforma es versátil y permite ambas opciones ¿porqué no poder usar ambas según el caso? De modo que decidimos también diseñar el esquema de cubo OLAP Mondrian para posteriormente subirlo a Pentaho y usarlo. De hecho el propio Pentaho incluye dentro de su plataforma la aplicación Schema Workbench para la construcción de dichos cubos (ver Figura 6.2).

6.1.1. Conclusión Diseño

Ya teníamos el diseño del proyecto completo, sabíamos lo que teníamos que hacer, y también sabíamos como hacerlo. De modo que ya finalmente sólo faltaba llevar a la práctica todo y proceder a su implementación. Justamente esto es lo que contaremos en la siguiente sección.

6.2. Desarrollo y Despliegue del Proyecto

6.2.1. Preparación del Entorno de Trabajo

Como en todo proyecto, lo primero era preparar el entorno de trabajo. Esto requería de descargar, instalar y configurar todas las herramientas elegidas para el desarrollo del proyecto. Y justo eso fue lo que hicimos como detallamos a continuación:

1. Java JDK 1.8

- Descarga: Descarga del fichero `.tar.gz` directamente desde su web.
- Instalación: Descomprimir el `.tar.gz` en una carpeta.
- Configuración: Crear o modificar la variable de entorno `JAVA_HOME` indicando la ruta hacia la carpeta dentro del fichero `.bashrc` en el home de usuario con permisos de superusuario (para que no tengamos que volver a hacerlo cada vez que iniciamos sesión, en este caso en Linux).

2. PostgreSQL 9.5

a) Sistema Gestor de Base de Datos

- Descarga e Instalación: Descarga e instala desde el propio repositorio de Ubuntu con `sudo apt-get install postgresql postgresql-contrib`.
- Configuración: Ejecutar `sudo -i -u postgres`, crear el nuevo rol: `fiumrol` mediante `createuser -i`, y por último crear la base de datos `facultatum` y asignarla a `fiumrol` mediante `createdb -O fiumrol facultatum`.

b) Herramienta Visual de Administración - pgAdmin III

- Descarga e Instalación: Descarga e instala desde el propio repositorio de Ubuntu con `sudo apt-get install pgadmin3`.
- Configuración: Ejecutar `pgadmin3`, conectar con el servidor de Base de Datos PostgreSQL en `localhost`, al puerto por defecto `5432`, con el usuario `fiumrol`. Una vez podamos acceder ya podemos trabajar visualmente con la base de datos.

3. Plataforma Pentaho Community 7.1

a) Pentaho Server - Business Analytics Platform

- Descarga: Descarga del fichero `.zip` directamente desde su web.
- Instalación: Descomprimir el `.zip` en una carpeta.

- Configuración: Ejecutar `./start-pentaho.sh`, añadir una nueva fuente de datos JNDI siguiendo los mismos pasos que para pgAdmin III, y configurar las cuentas de usuario y sus permisos de Pentaho desde el Back-End.
- b) Data Integration - Kettle - Spoon
- Descarga: Descarga del fichero `.zip` directamente desde su web.
 - Instalación: Descomprimir el `.zip` en una carpeta.
 - Configuración: Ejecutar `./spoon.sh` y configurar la conexión a la Base de Datos PostgreSQL compartiendo dicha conexión con todas las transformaciones y trabajos Kettle.
- c) Design Tools - Schema Workbench
- Descarga: Descarga del fichero `.zip` directamente desde su web.
 - Instalación: Descomprimir el `.zip` en una carpeta.
 - Configuración: Ejecutar `./schema-workbench.sh` y configurar la conexión a la Base de Datos PostgreSQL ya que esto nos ayudará bastante durante el diseño del cubo OLAP. Luego a la hora de publicar el cubo en la plataforma de Pentaho hay que acceder al servidor de Pentaho cuya dirección por defecto es `http://localhost:8080/pentaho`, introducir nuestras credenciales de usuario, e indicar el nombre del JNDI que hemos creado antes.

Con todo esto ya tendríamos listo el entorno de trabajo para el desarrollo del proyecto, al menos en Linux que es donde se ha desarrollado (ver Apéndice C).

6.2.2. Desarrollo y despliegue

Como ya comentamos en el capítulo anterior, como parte del diseño primero desarrollamos el esquema de la base de datos mediante un script SQL (ver Apéndice B) para adecuarlo al DW, que teníamos representado en el diagrama multidimensional. Tras ello usamos el Schema Workbench de Pentaho para diseñar nuestro cubo OLAP en Mondrian y poder publicarlo, o desplegarlo, en nuestro servidor de Pentaho.

El siguiente paso a desarrollar era uno de los grandes pasos de este TFM, que es el *proceso ETL* donde teníamos que cargar todos los datos de los Excels al DW. Para ello nos servimos de Spoon, para hacer primero todas las transformaciones Kettle por cada Dimensión y Tabla de Hechos, y finalmente definir un trabajo Kettle por encima que hiciese una por una, y en orden, todas las transformaciones. Lo bueno de esta herramienta visual, es que durante el proceso podíamos ver paso por paso el log por pantalla y ver si estaba

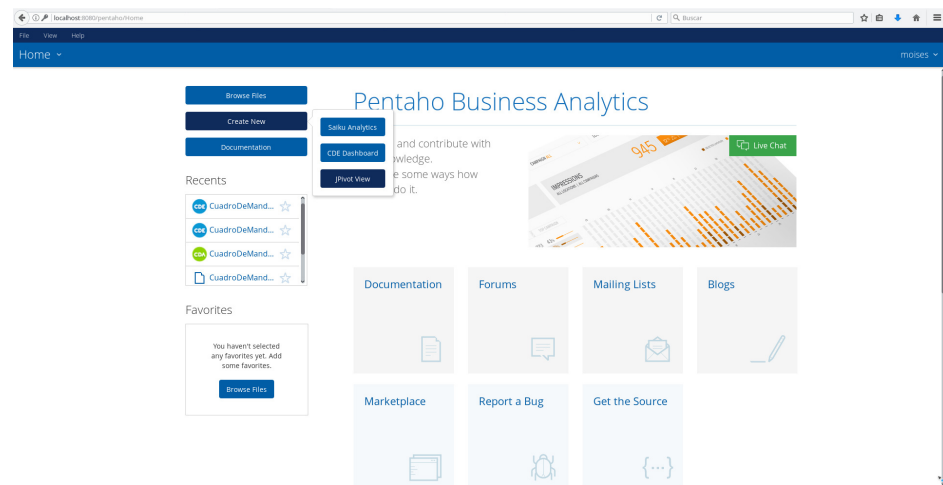


Figura 6.3: Pentaho Server Back-End desde donde desarrollar Cuadros de Mandos o Ejecutar Consultas

yendo bien las transformaciones y cargas o no. Lo cual, para nosotros, era bastante importante controlar.

Una vez todos los datos estaban cargados dentro del Data Warehouse, el último paso, y otro de los grandes pasos importantes del proyecto, era crear los cuadros de mandos directivo (nivel 1) y técnico (nivel 2) dentro de la plataforma de Pentaho. Para ello íbamos a utilizar los dos KPIs que habíamos descrito en el capítulo 4, y dada la versatilidad de la herramienta CDE (Community Dashboard Editor) íbamos a utilizar primero una consulta SQL al JNDI para el primer KPI, y una segunda consulta MDX para el cubo OLAP de Mondrian, precisamente para poder corroborar que podíamos acceder a los mismos datos del DW con distintos lenguajes de consulta.

No obstante, debido a la complejidad del Mondrian de Pentaho, que no permite realizar todas las consultas MDX (por ejemplo no se pueden hacer comprobaciones de IS NULL, y pasamos una noche entera intentado ejecutar consultas con agregados sin que no nos diese errores), nos hemos tenido que valer de otra herramienta para comprobar si las consultas MDX en el Mondrian de Pentaho eran válidas, y esa herramienta ha sido *Saiku* [8] para Pentaho, la cual se puede adquirir gratuitamente desde el Marketplace de Pentaho, y su instalación es tan sencilla como descomprimir la carpeta dentro de los plugins del servidor de Pentaho. Sin embargo, aún era necesario solicitar una licencia Community gratuita al desarrollador, al igual que pasaba con la edición 5.1 de Pentaho, lo cual tampoco fue un paso demasiado traumático, ya que se hizo en apenas 10 minutos ¿Porqué Saiku y no JPivot que es la herramienta que viene por defecto en la plataforma para los cubos en Pentaho? Pues porque JPivot se queda muy corto, a nivel visual no permite incluir medidas, ni miembros calculados, no puedes hacer cross-

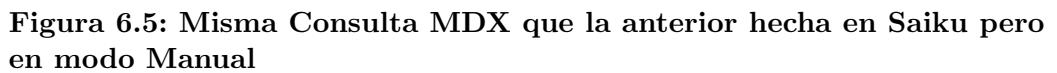
Curso	Presentado	Aprobado	Presentado	Aprobado	Presentado	Aprobado
Des001	1	1	1	1	1	1
Des002	1	1	1	1	1	1
Des003	1	1	1	1	1	1
Des004	1	1	1	1	1	1
Des005	1	1	1	1	1	1
Des006	1	1	1	1	1	1
Des007	1	1	1	1	1	1
Des008	1	1	1	1	1	1
Des009	1	1	1	1	1	1
Des010	1	1	1	1	1	1
Des011	1	1	1	1	1	1
Des012	1	1	1	1	1	1
Des013	1	1	1	1	1	1
Des014	1	1	1	1	1	1
Des015	1	1	1	1	1	1
Des016	1	1	1	1	1	1
Des017	1	1	1	1	1	1

Figura 6.4: Consulta MDX hecha en Saiku en modo Visual

join en las columnas, y prácticamente una consulta MDX bien hecha había que hacerla casi totalmente a mano. Y esto no era uno de los requisitos del presente TFM: *“El TFM se centrará en el desarrollo de una herramienta, principalmente visual”* (Ver las Figuras 6.4 y 6.5).

No queríamos que el proyecto *facultatUM* se quedara en la mera visualización de unos cuadros de mandos, sino que también fuera una plataforma extensible, de modo que todo aquello que se pudiera hacer de forma visual era importante por este motivo. Es decir, que cualquiera a nivel técnico con los suficientes conocimientos en Bases de Datos y entornos Back-End, pero no en programación, pudiera construir su propio cuadro de mandos a voluntad de ser necesario ¿Porqué? Porque cada facultad, y cada departamento dentro de ella puede que algunos indicadores secundarios y métricas les resulten útiles y otros no, o precisen más nivel de detalle o menos. De ahí el deseo de que el proyecto fuera extensible y personalizable, toda vez que fuera lo más visual posible.

Una vez, teníamos la herramienta de Saiku instalada, decidimos crear los dos cuadros de mandos de nivel 1 y nivel 2 (ver Figuras 6.6 y 6.7) a modo de ejemplos ilustrativos de lo que el proyecto dentro de la plataforma es capaz de hacer. Pero sobre todo quisimos centrarnos en la utilidad **visual** del primero para que con pocos elementos en pantalla “dijera” mucho sólo con mirarlo. A nivel directivo, ya sea el decano, o el gerente de una organización con ánimo de lucro, normalmente tiene muchas ocupaciones y muy poco tiempo. Por lo que lo normal es que el vea el cuadro y sólo verlo le baste para detectar si hay o no hay algún problema, y en el caso de haberlo delegar en el departamento técnico encargado la trazabilidad, o las indagaciones pertinentes, para saber porqué ocurre ese problema, o quien o quienes lo han casuado y si se pueden



Finalmente para desplegar la aplicación web, sólo bastaría con ejecutar `./start-pentaho.sh` dentro de la carpeta del servidor de Pentaho y después con un explorador web acceder a <http://localhost:8080> si se está ejecutando en la propia máquina. Si se accede desde un lugar remoto habría que sustituir “localhost” por la ip del equipo y el puerto podría variar en función del administrador de sistemas encargado del servidor donde finalmente se aloje la aplicación web.

La principal conclusión que podemos extraer de este capítulo, es que, aunque a veces parezca que dedicar tanto tiempo al análisis y al diseño de una aplicación, luego eso se nota a la hora de la implementación de la misma, ya que con lo único que hemos tenido que lidiar aquí prácticamente han sido con detalles puramente técnicos, pero ya teníamos claro qué teníamos que hacer y cómo hacerlo, y lo hemos hecho sin tener que retroceder a la etapa de diseño o de análisis para corregir cosas. Eso al final, en el ciclo de desarrollo de cualquier software ahorra mucho tiempo, ya que enmendar los errores de diseño, y va no digamos los de análisis, luego cuestan mucho más



Figura 6.6: Cuadro de Mandos Directivo o de Nivel 1

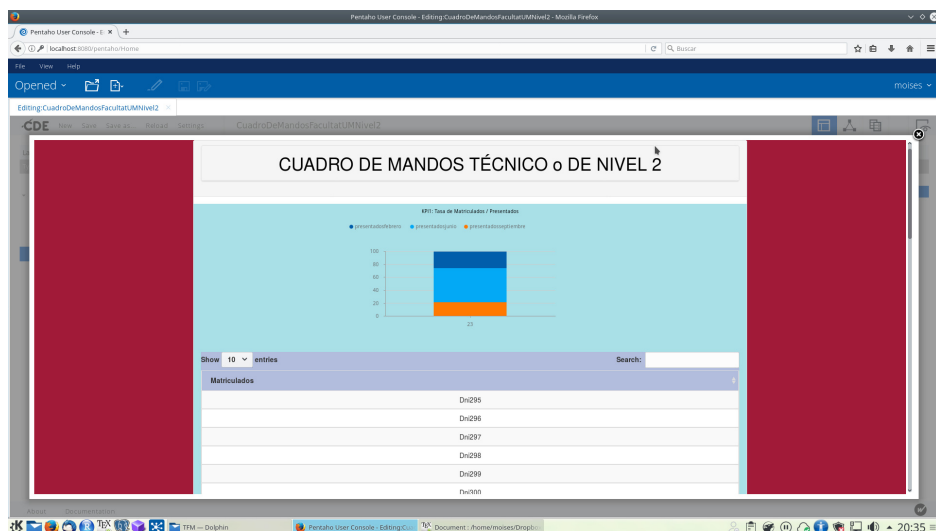


Figura 6.7: Cuadro de Mandos Técnico o de Nivel 2

tiempo que enmendar los errores de implementación.

Por lo que salvo algunos casos de configuración de las herramientas, errores en las consultas MDX con el Mondrian de Pentaho, y algunos que otros ajustes en el cuadro de mandos con Bootstrap para hacerlo Responsive con todos los dispositivos, apenas nos hemos encontrado con muchas más dificultades, pero una vez más, insistimos, eso es debido a la inversión de tiempo y esfuerzo en las etapas previas.

Capítulo

7

Conclusiones Finales y Vías Futuras

7.1. Conclusiones Finales

El objetivo de este proyecto es el estudio de los componentes elementales para el desarrollo de un cuadro de mandos de un centro universitario (Facultad). En particular, este trabajo se ha centrado en el análisis, diseño e implementación de un cuadro de mandos básico de resultados académicos. En primer lugar, en este TFM se han identificado y analizado un conjunto de indicadores clave analizando los reales decretos y la propia normativa de la Universidad de Murcia. En segundo lugar se ha diseñado un Diagrama Multidimensional y se ha realizado un Data Warehouse en base al mismo dentro de una base de datos. En tercer lugar se ha hecho un cubo OLAP y se ha utilizado una consulta MDX sobre el mismo para uno de los KPIs. Finalmente, se ha implementado una versión básica de un cuadro de mandos de nivel 1 (directivo) y de nivel 2 (técnico) donde también se ha puesto de manifiesto que se puede obtener datos directamente del DW haciendo también consultas SQL demostrando su versatilidad.

Otra conclusión que podemos extraer es como el hecho de haber invertido más tiempo en el análisis del problema y el diseño del proyecto puede luego dar a lugar a no sólo una implementación más sencilla, sino el que tengamos que volver hacia atrás para corregir errores y aprovechar mejor ese tiempo finito del que antes estábamos hablando. Tal vez el Proceso de Desarrollo del Software no sea algo evaluable en este máster específico, aunque sí lo sea en el grado, pero pensamos que cualquier software que se haga, y más si es un proyecto para la Universidad de Murcia, tiene que regirse siempre por el código de las buenas prácticas que aprendimos en PDS. Por lo que haber hecho este proyecto siguiendo ese código de buenas prácticas ha resultado

ser algo enriquecedor tanto para el propio proyecto en sí como para los que hemos participado en su desarrollo.

Como última conclusión quería destacar una de las frases que puse en el prefacio del presente TFM parafraseando la célebre frase del arquitecto *Mies van der Rohe*: “*Menos es más*”. Creemos que esto es fundamental en visualización, y así lo hemos querido manifestar en el primer cuadro de mandos, donde lo importante era con poco decir mucho, pero diciéndolo de forma visual ¿qué era lo que queríamos decir? Lo importante, si había o no había problemas. Si se detectan luego ya se trazarán, pero lo principal es detectar los problemas y si puede ser en un sólo golpe de vista mejor. Ese era desde el principio uno de los grandes objetivos que perseguíamos, y como conclusión creo que hemos sabido interpretar la esencia de esas grandes palabras.

7.2. Vías futuras

Los cuadros de mandos se han hecho a modo ilustrativos sobre una titulación en concreto, el presente máster de Big Data. Una vía futura sería añadir un combobox en el cuadro de mandos y poder elegir entre las diferentes titulaciones disponibles, y si queremos, incluso tener un histórico con todos los cursos académicos y sus respectivas convocatorias también en sus respectivos comboboxes. En definitiva, cualquier idea que sirva para extender y personalizar no sólo los cuadros de mandos, sino la propia plataforma incluyendo, por ejemplo, una herramienta para informes en los que al final de un curso académico pongamos en ellos los KRIs. También sería una muy buena idea y una vía por la cual seguir.

Apéndice

Apéndice



Objetivos Prefijados por Decanato y Estimaciones para el Máster de Big Data

En este apéndice se muestran tanto los Objetivos Prefijados por Decanato como las Estimaciones Esperadas [15] para las tasas del presente Máster de Big Data.

TFM: "Cuadro de mandos de rendimiento académico en centros" (tentativo)

Curso: 2016-2017

Alumno: Moises Frutos Plaza.

Tutores: Pedro Miguel Ruiz Martínez y José Manuel Juárez Herrero.

Objetivo principal:

Construcción de herramienta de apoyo a la toma de decisiones en el ámbito de un centro universitario a partir de resultados académicos.

Sub-objetivos:

- Identificar KPI.
- Preparación de datos (limpieza, ETL)
- Desarrollo cuadro de mandos integral.
- Evaluación de la herramienta.

Aspectos a estudiar:

- KPI de proceso y de salida: sólo factibles aquellas información disponible
- Estudio comparativo de herramientas necesarias (ver requisitos previos)
- Necesidad de cubos o no.
- Cuadros de mandos: criterios de diseño y elaboración a partir de KPI.

Fuentes de datos/información:

- Estructura de las bases de datos de alumno, profesor, titulación, asignatura y calificaciones. Los ejemplos proporcionados no se corresponden a datos reales, aunque siguen el tipo de registros y volumen referente a de 3 años (aproximadamente).
- Indicadores de Resultados Académicos de SGIC Unidad Calidad UMU:
<http://www.um.es/web/unica/contenido/titulaciones/grados-masteres/sgc/manual-procedimientos-sgc>
- Resultados Previstos (Anexo I secc. 8) Real Decreto 1393/2007
<https://www.boe.es/buscar/act.php?id=BOE-A-2007-18770>
- Información sobre posibles indicadores (informales) de la Facultad de Informática.

Requisitos Previos:

- El TFM se centrará en el desarrollo de una herramienta, principalmente visual, aunque otros aspectos podrán ser tenidos en cuenta.
- El volcado de datos es muy puntual, por tanto la preparación de datos será offline.
- El cuadro de mandos podrá ser una aplicación de escritorio o web.
- El proceso de preparación de dato y el cuadro de mandos deberán ser multiplataforma (OSX, Windows, Linux).
- Utilización de herramientas gratuitas, de licencia libre o licencia académica.
- El sistema deberá ser ligero y no deberá requerir de un despliegue complejo, pudiendo ser ejecutable en un ordenador de capacidad limitada.

Queda fuera del alcance del proyecto:

- Análisis de datos de rendimiento académico (no hay datos reales disponibles).
- La validación de la herramienta en el centro.

Figura A.1: Objetivos y Requisitos Prefijados por Decanato



8. RESULTADOS PREVISTOS

8.1. Valores cuantitativos estimados para los indicadores y su justificación.

Justificación de los indicadores:

Los Sistemas de Garantía de Calidad de los Centros que impulsan esta propuesta utilizan una serie de indicadores de resultados que se definen en el RD 1393/2007 como:

Tasa de graduación: porcentaje de estudiantes que finalizan la enseñanza en el tiempo previsto en el plan de estudios o en un año académico más en relación a su cohorte de entrada.

Tasa de abandono: relación porcentual entre el número total de estudiantes de una cohorte de nuevo ingreso que debieron obtener el título el año académico anterior y que no se han matriculado ni en ese año académico ni en el anterior.

Tasa de eficiencia: relación porcentual entre el número total de créditos del plan de estudios a los que debieron haberse matriculado a lo largo de sus estudios el conjunto de titulados de un determinado año académico y el número total de créditos en los que realmente han tenido que matricularse.

Tasa de rendimiento: relación porcentual entre el número de créditos superados (excluidos los adaptados, convalidados y reconocidos) por el alumnado y el número total de créditos en los que se ha matriculado.

Atendiendo al perfil de ingreso recomendado, los objetivos planteados, la metodología de enseñanza y otros elementos indicados a lo largo de la presente memoria así como a los valores anteriores sobre los másteres actuales en USC y UMU, las estimaciones propuestas son las siguientes:

Indicador	Estimación
Tasa de Graduación	85%
Tasa de Abandono	5%
Tasa de Eficiencia	95%
Tasa de Rendimiento	95%



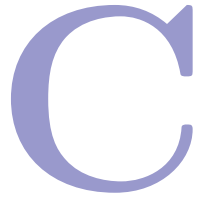
Figura A.2: Estimaciones Esperadas para las tasas del Máster de Big Data

Apéndice B

Ficheros del Trabajo

Todos los ficheros relativos a este Trabajo de Fin de Máster (cubo OLAP Mondrian, Trabajo y Transformaciones Kettle, Script SQL, Consultas SQL y MDX, etc...) pueden consultarse en el siguiente repositorio:

<https://github.com/Zollkron/tfm>



Apéndice

Datos de los equipos empleados

C.1. Equipo donde se ha desarrollado

- Procesador: AMD FX-8370 Eight-Core Processor
- Memoria Principal: 16 GB RAM DDR3 a 2400 MHz
- Tarjeta Vídeo: AMD Radeon RX 580
- S.O.: Ubuntu 16.04.2 LTS 64-bit
- Java: Oracle JDK 1.8

C.2. Equipo donde se ha desplegado

- Procesador: Intel(R) Core(TM) i5-2450M CPU @ 2.50GHz 64-bit
- Memoria Principal: 6 GB RAM DDR3 a 2400 MHz
- Tarjeta Vídeo: AMD Radeon HD 7550M
- S.O.: Ubuntu 16.04.2 LTS 64-bit
- Java: Oracle JDK 1.8

Bibliografía

- [1] Olap.com team and PARIS Tech. Olap cube, 2017. <http://olap.com/learn-bi-olap/olap-bi-definitions/olap-cube/>.
- [2] Hitachi and Pentaho Community. Mondrian, 2017. <http://community.pentaho.com/projects/mondrian/>.
- [3] Microsoft. Consulta de mdx básica (mdx), 2016. [https://msdn.microsoft.com/es-es/library/ms144785\(v=sql.120\).aspx](https://msdn.microsoft.com/es-es/library/ms144785(v=sql.120).aspx).
- [4] Gobierno de España. Real decreto 1393/2007, anexo i, sección 8: Resultados previstos, 2007. <https://www.boe.es/buscar/act.php?id=BOE-A-2007-18770>.
- [5] Universidad de Murcia. Estatutos de la universidad de murcia, 2004. http://www.um.es/csif/pdfs/estatutos_um.pdf.
- [6] Universidad de Murcia. Manual de procedimientos SGIC, 2010. <http://www.um.es/web/unica/contenido/titulaciones/grados-masteres/sgc/manual-procedimientos-sgc>.
- [7] David Parmenter. *Key Performance Indicators: developing, implementing and using KPIs*. Wiley, 2015. páginas 3-23.
- [8] Meteorite.bi. Saiku business analytics ce 3.15, 2017. <http://www.meteorite.bi/products/saiku>.
- [9] Qlik. Qlik Sense, 2017. <http://www.qlik.com/es-es/products/qlik-sense>.
- [10] Parente. Jupyter dashboards server, 2017. https://github.com/jupyter-incubator/dashboards_server.
- [11] Gobierno de España. Ley orgánica 15/1999, de 13 de diciembre, de protección de datos de carácter personal, 1999. <https://www.boe.es/buscar/doc.php?id=BOE-A-1999-23750>.
- [12] Hitachi and Pentaho Community. Pentaho Community 7.1 Business Analytics Platform (Server), 2017. <http://community.pentaho.com/>.

- [13] Hitachi and Pentaho Community. Community Dashboard Editor (CDE), 2016. <http://community.pentaho.com/ctools/cde/>.
- [14] JGraph Ltd. Draw.io v7.3.5, 2017. <https://www.draw.io/>.
- [15] Facultad de Informática, Universidad de Murcia, and Universidad de Santiago de Compostela. Estimaciones propuestas para los resultados previstos, 2017. <http://www.um.es/informatica/index.php?recurso=26561>.