

PONTIFICIA UNIVERSIDAD CATÓLICA DEL PERÚ

FACULTAD DE CIENCIAS E INGENIERÍA



PONTIFICIA
UNIVERSIDAD
CATÓLICA
DEL PERÚ

DESARROLLO DE UN SISTEMA DE GESTIÓN DE EVALUACIONES BASADAS EN RÚBRICAS EN CURSOS DE PROYECTOS UNIVERSITARIOS DE UNA CARRERA ACREDITADA

Tesis para optar el Título de Ingeniera Informática, que presenta el bachiller:

Atenas Marilyn Alan Rodríguez

ASESOR: Mag. Abraham Eliseo Dávila Ramón

Lima, marzo de 2014

RESUMEN DEL PROYECTO FINAL

El proceso de evaluación de cursos universitarios orientados a proyectos, el cual está sujeto al uso de rúbricas, implica un gran esfuerzo de parte de los docentes y sus asistentes de docencia para poder realizar la evaluación de todos los estudiantes durante cada período lectivo. La importancia de los resultados de estas evaluaciones, trascienden del seguimiento que se realizan a los estudiantes en cursos de carrera universitaria, especialmente si la carrera se encuentra substancialmente acreditada o acreditada por algún organismo internacional.

Por ello, el presente proyecto propone el desarrollo de una herramienta informática que facilite el soporte para la recolección y consolidación de las calificaciones de estudiantes basados en rúbricas para la evaluación del cumplimiento de los resultados del programa. Este sistema se basa principalmente en entradas como: fechas del semestre académico, evaluadores (docentes y asistentes de docencia), lista de evaluados (estudiantes) y rúbricas por curso. Con estas entradas, el sistema apoya en las siguientes tareas o actividades: la planificación de las evaluaciones, conformación de equipos de estudiantes, asignación de evaluadores, evaluados y evaluaciones por período lectivo, realización de las evaluaciones y verificación del cumplimiento de la evaluación de todos los criterios de los resultados por estudiante de cada curso. Finalmente para todo este proceso, el sistema genera información de salida como: indicadores de resultados por estudiante y reporte final de las evaluaciones.

El presente trabajo consta de cuatro capítulos. En el primer capítulo se presentan las generalidades del proyecto, definición de la problemática, marco conceptual, revisión del estado del arte, la metodología, el objetivo general del proyecto, los objetivos específicos, los resultados esperados, las herramientas y metodologías a utilizar y todo lo necesario para entender los términos del negocio asociado al proyecto. Por otro lado, en el capítulo 2 se cubre el análisis y el diseño del sistema en los cuales se documentan los requisitos, arquitectura y modelo físico de datos, entre otros. El capítulo 3 se centra en la construcción del software, la selección de herramientas y tecnologías utilizadas en el desarrollo y los estándares de programación utilizados. Asimismo se presenta el plan de pruebas según la metodología Extreme Programming. Finalmente, en el capítulo 4, se presentan las recomendaciones y conclusiones del proyecto, capítulo donde se detallan los logros del proyecto y los futuros trabajos que se pueden realizar, relacionados al tema.

Dedicatoria

A Dios y la Virgen, por motivarme espiritualmente a concluir el presente proyecto.

A mis padres John y Carmen, por ser mi mejor ejemplo a seguir; por su esfuerzo, apoyo y dedicación desde que inicie esta y todas las etapas de mi vida. Por siempre decir si se puede y enseñarme a enfrentar los obstáculos. Por permitir que pueda lograr el sueño de convertirme en ingeniera.

A mi hermana Josselyn, por motivar la culminación de esta etapa y deseándole los mejores deseos para ella, futura ingeniera.

A mi futuro esposo Luiggi, por su apoyo incondicional durante el desarrollo de este proyecto, por su paciencia y comprensión en los momentos que pensé que no lo lograría. Por ser mi gran inspiración y modelo a seguir.

A mi asesor Abraham Dávila, por creer en mí desde un principio, por orientarme y enseñarme en todo el camino del presente proyecto.

A cada uno de ustedes mi total agradecimiento.

Tabla de Contenido

1. CAPÍTULO 1. GENERALIDADES	8
1.1. PROBLEMÁTICA Y SU CONTEXTO	8
1.2. MARCO CONCEPTUAL	10
1.2.1. CRITERIO	10
1.2.2. ESTÁNDAR	10
1.2.3. NIVEL DE VALORACIÓN O ESCALA	10
1.2.4. RÚBRICA	11
1.2.5. EVALUACIÓN CUALITATIVA	14
1.2.6. COMPETENCIA	14
1.2.7. EVALUACIÓN POR COMPETENCIAS	15
1.2.8. ACREDITACIÓN	16
1.2.9. PLAN ESTRATÉGICO INSTITUCIONAL	17
1.3. ESTADO DEL ARTE Y DE LA PRÁCTICA	17
1.3.1. ESTADO DEL ARTE	17
1.3.2. CUADRO COMPARATIVO	25
1.3.3. ESTADO DE LA PRÁCTICA	27
1.4. PROPUESTA DEL PROYECTO	28
1.4.1. OBJETIVO GENERAL	28
1.4.2. OBJETIVOS ESPECÍFICOS	28
1.4.3. RESULTADOS ESPERADOS	28
1.4.4. MÉTODOS, METODOLOGÍAS Y PROCEDIMIENTOS	29
1.4.5. ALCANCES Y LIMITACIONES	36
1.4.6. RIESGOS	37
1.4.7. JUSTIFICACIÓN Y VIABILIDAD DEL PROYECTO	38
1.5. PLANIFICACIÓN DEL PROYECTO	40
1.5.1. ESTRUCTURA DE DESGLOSE DE TRABAJO (EDT)	40
1.5.2. CRONOGRAMA DEL PROYECTO (DIAGRAMA DE GANTT)	42
2. CAPÍTULO 2: DEFINICIÓN DEL SISTEMA	45
2.1. DEFINICIÓN DE REQUISITOS	45
2.1.1. USUARIOS	45
2.1.2. IDENTIFICACIÓN DE REQUISITOS	46
2.1.3. PROTOTIPOS	46
2.1.4. LISTA DE HISTORIAS DE USUARIO	49
2.1.5. HISTORIAS DE USUARIO	51
2.2. ANÁLISIS DE LA SOLUCIÓN	53
2.2.1. REGLAS DE NEGOCIO	53
2.3. DISEÑO DEL SISTEMA	55
2.3.1. ESTRUCTURAS DE DATOS	55
2.3.2. ARQUITECTURA DE LA SOLUCIÓN	57
3. CAPÍTULO 3: CONSTRUCCIÓN Y PRUEBAS	61
3.1. CONSTRUCCIÓN	61
3.1.1. HERRAMIENTAS DE DESARROLLO	61
3.1.2. IMPLEMENTACIÓN DE COMPONENTES	62
3.1.3. BUENAS PRÁCTICAS PARA EL DESARROLLO	69
3.2. PRUEBAS	69

4. CAPÍTULO 4: OBSERVACIONES, CONCLUSIONES Y RECOMENDACIONES	72
4.1. OBSERVACIONES	72
4.2. CONCLUSIONES	72
4.3. RECOMENDACIONES	73
REFERENCIAS BIBLIOGRÁFICAS	74



ÍNDICE DE FIGURAS

Figura 1-1 Ejemplo de uso de hojas de cálculo en la elaboración de rúbricas.	18
Figura 1-2 Editor de rúbrica de Moodle versión 2.2	19
Figura 1-3 Evaluación en el Sistema Avanzado de Calificación basado en rúbricas.	20
Figura 1-4 Fórmula de Cálculo de Calificación Moodle	20
Figura 1-5 Ejemplo de Uso de Fórmula Calificación Moodle 2.2	21
Figura 1-8 Rúbricas de diversas actividades (RubiStar, 2012)	22
Figura 1-9 Registrar información básica de la rúbrica (RubiStar, 2012)	22
Figura 1-10 Edición de la rúbrica, definiendo niveles y criterios (RubiStar, 2012)	22
Figura 1-11 Definición de criterios en ECO (ECO, 2012)	24
Figura 1-12 El sistema Talent Test v3.0 permite evaluar al trabajador para un determinado puesto (Talenttest, 2012).	25
Figura 1-13 Estructura de descomposición del Trabajo (EDT)	41
Figura 1-14 Cronograma del Proyecto- Inicio del Proyecto	42
Figura 1-15 Cronograma del Proyecto –Ejecución y Control	43
Figura 1-16 Cierre del Proyecto	44
Figura 2-1 Prototipo para la creación de una rúbrica general	47
Figura 2-2 Prototipo para la creación de una rúbrica por curso	48
Figura 2-3 Planificación de Evaluaciones	49
Figura 2-6 Diagrama Entidad-Relación para la estructura de datos de las rúbricas.	57
Figura 2-7 Uso del patrón MVC aplicado a la creación de rúbricas	60
Figura 3-1 Estructura Interna de la Solución	62
Figura 3-2 Pantalla inicial de acceso al sistema	63
Figura 3-3 Uso de la variable Session de PHP en el proyecto	64
Figura 3-4 Uso de las variables Session para buscar estudiantes	64
Figura 3-5 Registro Rúbrica General	65
Figura 3-6 Búsqueda de Niveles	66
Figura 3-7 Registro de Rúbrica por Curso	67
Figura 3-8 Planificación de Evaluaciones	68
Figura 3-9 Asignación de equipos evaluados	69

ÍNDICE DE TABLAS

Tabla 1-1 Componentes de Rúbrica (Rogers, 2010)	11
Tabla 1-2 Extracto de Rúbrica del curso INF227 de la Especialidad Ingeniería Informática	13
Tabla 1-3 Cuadro Comparativo del Estado del Arte	26
Tabla 1-4 Mapeo Herramientas, Métodos y Metodologías	30
Tabla 1-5 Riesgos del Presente Proyecto	38
Tabla 1-6 Recursos informáticos	40
Tabla 2-1 Prioridad en historias de usuario	50
Tabla 2-2 Dificultad en desarrollo de las historias de usuario	50
Tabla 2-3 Lista de historias de usuario	51
Tabla 2-4 Administrar Rúbrica General	52
Tabla 2-5 Configurar Tipo de Equipos por Curso	52
Tabla 2-6 Reglas para las rúbricas	54
Tabla 2-7 Reglas de Negocio para los estudiantes	55



CAPÍTULO 1. GENERALIDADES

En este primer capítulo, se presenta la definición de la problemática identificada, el objetivo general del proyecto, los objetivos específicos y sus resultados esperados, el alcance y las limitaciones, los principales principios tomados de las metodologías y procedimientos para el desarrollo del proyecto, la justificación, la viabilidad y finalmente la planificación del proyecto.

1.1. Problemática y su Contexto

La Facultad de Ciencias e Ingeniería (FCI) de la Pontificia Universidad Católica del Perú (PUCP) viene trabajando desde el 2004 en la mejora continua, principalmente enfocada en el proceso de enseñanza-aprendizaje y en la búsqueda de reconocimiento internacional a través de la acreditación del programa. Este proceso implicó el establecimiento de organizaciones como los comités consultivos y equipos de trabajo que buscan propiciar la participación de los docentes en este proceso de mejora continua, capacitar en estos procesos a distintos actores (profesores, administrativos y estudiantes), aprender de casos de éxito a nivel local e internacional, y organizar el trabajo, con apoyo de las unidades de la PUCP como el Instituto para la Calidad (IC-PUCP) y la Dirección Académica de Planeamiento y Evaluación (DAPE), entre otros (Torrealva, 2009).

Posteriormente, en el año 2007, en la nueva edición el Plan Estratégico Institucional (PEI), se formalizó el interés inicial de la Universidad en estos procesos de acreditación. En dicho documento se definieron los procesos, los objetivos estratégicos, y las metas institucionales, los cuales están dirigidos a mejorar el desempeño de la PUCP, en las áreas de formación, investigación y responsabilidad social (Torrealva, 2009).

En el despliegue del PEI, los objetivos estratégicos buscan gestionar y mejorar la calidad de la formación en la PUCP, como el caso de la Acreditación Internacional (DAPE, 2007). Por ello, desde el año 2006, se empezaron diversas actividades con determinados fines, entre los que se encuentran: la definición de los objetivos educacionales, la definición de las competencias y habilidades frente al mundo laboral, la definición de rúbricas de medición para los resultados de cada especialidad, la medición de resultados de la especialidad usando rúbricas, reuniones con docentes y egresados para identificar los aspectos a mejorar, entre otros que siguen surgiendo cada año (Torrealva, 2009).

Con el propósito de realizar la medición del logro de los resultados del programa, fue necesario establecer un conjunto de rúbricas que dirigieran dicha evaluación y que además permitan unificar los criterios de los docentes sobre los resultados. Previo al proceso de acreditación, las evaluaciones se orientaban principalmente a realizar una comprobación del conocimiento alcanzado a través de exámenes, trabajos o lecturas, entre otros; sin embargo, en este caso, las rúbricas se diseñaron para medir ciertas conductas o prácticas que los estudiantes adoptan a lo largo del curso y que implican otra manera de recolectar los resultados.

El uso de las rúbricas, en la evaluación de resultados del programa en los diferentes cursos de las carreras de ingeniería de la PUCP, es un proceso manual, en el mejor de los casos basado en el uso de hojas de cálculo y con apoyo parcial de algunas aplicaciones informáticas como correos electrónicos, entre otros. Para cursos que tienen como responsabilidad evaluar un número reducido (uno o dos) resultados, esta medición representa menor esfuerzo. Para cursos que están al final de la carrera y que usan el modelo de aprendizaje orientado por proyectos (POL, de *Project Oriented Learning*) como es el caso de Ingeniería de Software, Desarrollo de Programas 1 y Desarrollo de Programas 2 (Dávila, 2003) suelen tener la responsabilidad de medir la mayoría de los resultados, siendo un proceso más complejo y engorroso, lo cual demanda mucho tiempo y son fuente potencial de registros incorrectos.

Hasta la fecha, como resultado del sistema de mejora continua, se ha logrado la acreditación internacional de diversas especialidades de ingeniería. En el año 2008, tres especialidades de la FCI fueron reconocidas como substancialmente equivalente por la acreditadora canadiense CEAB (Canadian Engineering for Accreditation Board), reconocimiento que es equivalente a los programas acreditados en universidades de Canadá (Torrealva, 2009). Posteriormente, en el año 2010, cinco especialidades de la FCI aprobaron la acreditación estadounidense ABET (Accreditation Board for Engineering and Technology) (Torrealva, 2011). CEAB y ABET son acreditadoras en Ingeniería que pertenecen al acuerdo de Washington, que agrupa a las más prestigiosas acreditadoras del mundo en el campo de la Ingeniería (Torrealva, 2011). Con la acreditación ABET además se logró la acreditación con el Instituto de Calidad y Acreditación de Carreras Profesionales de Ingeniería y Tecnología (ICACIT), entidad de acreditación peruana. En el futuro, se aspira la continuidad del sistema de mejora, y por ello, sin duda, se requerirá implementar nuevas herramientas que apoyen en cada uno de los procesos del sistema para que en su totalidad siga obteniendo buenos resultados (Torrealva, 2010).

Con el escenario descrito anteriormente, se puede apreciar que las herramientas utilizadas en el proceso de evaluación de cursos de proyectos, el cual está sujeto al uso de rúbricas, implica esfuerzos adicionales por parte del docente y sus asistentes de docencia para poder realizar la evaluación de todos los estudiantes durante cada período lectivo. Por ello, este proyecto propone el desarrollo de una herramienta informática que brinde el soporte para la recolección y consolidación de las calificaciones de estudiantes basados en rúbricas para la evaluación del cumplimiento de los resultados del programa evitando errores y en menor trazo requerido, es decir, contribuir a que el proceso sea eficaz y eficiente.

1.2. Marco Conceptual

Se presentarán a continuación, los principales conceptos relacionados al proyecto de fin de carrera que se consideran necesarios para un adecuado entendimiento.

1.2.1. Criterio

Criterio se refiere a lo que en forma mínima debe contener un producto, ya sea tareas, informes, proyectos, módulos o todo tipo de investigación sujeta a una evaluación con la finalidad de apoyar en la elaboración de las rúbricas (Minte, 2008).

El uso de criterios en una evaluación es establecido por el evaluador y se consideran en el momento de determinar la calidad del trabajo de un estudiante (Reddy, 2010). Es importante destacar que los criterios que se definen en un momento no son los únicos posibles ni tienen un carácter absoluto, pueden estar sujetos a cambios (Minte, 2008). Lo más importante de este concepto es que los criterios deben ser comprensibles tanto para los estudiantes como para los evaluadores (Minte, 2008).

1.2.2. Estándar

Un estándar se define “como tipo, modelo, norma, patrón o referencia” (RAE, 2012).

1.2.3. Nivel de valoración o Escala

El concepto de nivel de valoración describe qué tan bien o mal se ha realizado una determinada actividad; es decir, permite indicar cuál es el objetivo a alcanzar de la evaluación basado en rúbricas. Los términos usados para describir la escala o nivel de rendimiento deben ser discretos pero claros (Stevens, 2005).

1.2.4. Rúbrica

Una rúbrica o matriz de valoración se define como la escala de evaluación donde se establece niveles progresivos que están relacionados con el desempeño de una persona respecto a un proceso determinado (RAE, 2012).

La rúbrica es como un descriptor cualitativo que permite anotar las habilidades para evaluar las competencias. Para su elaboración, se requiere el uso de criterios. El término de “descriptor cualitativo” se refiere en forma explícita de menor a mayor el dominio de una competencia (Simon, 2001).

El uso de rúbricas, como herramienta de evaluación para fines de acreditación internacional, se definen como una forma de establecer las expectativas de rendimiento de los estudiantes, donde las rúbricas proporcionan las características para cada nivel de rendimiento en el desempeño que el estudiante debe ser evaluado (Rogers, 2010).

- **Componentes de una rúbrica:**

Las rúbricas generalmente contienen los siguientes tres componentes (Rogers, 2010):

- **Dimensiones:** se refiere a los criterios de rendimiento.
- **Escalas:** se refiere a nivel de rendimiento.
- **Descriptor.**

Se presenta en la Tabla 1.1 un ejemplo donde se puede visualizar los componentes de una rúbrica

COMMUNICATION SKILLS

	Unsatisfactory 1	Developing 2	Satisfactory 3	Exemplary 4
Criteria				
Criteria	DIMENSIONS		SCALES	
Criteria				
Criteria	DESCRIPTORS			

Tabla 1-1 Componentes de Rúbrica (Rogers, 2010)

- **Tipos de Rúbricas**

Se distinguen dos tipos de rúbricas que interesan en el presente proyecto.

- **Análítica:** En este tipo de rúbrica, cada criterio en relación con el proceso y/o producto evaluado se separa y se evalúa sobre la base de una descriptiva propia; es decir, se enfoca en aspectos importantes de desempeño de los estudiantes. El resultado permite una retroalimentación acerca de las debilidades o fortalezas en el rendimiento del estudiante (Rogers, 2010).

Usualmente, se utiliza este tipo de rúbrica:

- Cuando es necesario observar fortalezas y debilidades.
 - Cuando es necesario evaluar habilidades complicadas o rendimiento.
 - Facilitar la retroalimentación para futuras mejoras.
 - Cuando se desea que los propios estudiantes autoevalúen su rendimiento.
- **Holística:** En este tipo de rúbrica los diferentes criterios son considerados en combinación; es decir de manera general, sobre una única escala descriptiva resultando un juicio más global, con mayor ajuste entre las descripciones indicados en las escalas de evaluación. Cada categoría en la escala describe el rendimiento en varios criterios de rendimiento; al ser genérico se puede utilizar con muchas actividades (Rogers, 2010).

Usualmente, se utiliza este tipo de rúbrica:

- Cuando es necesario una rápida evaluación.
- Cuando solamente una dimensión es necesario para comprender el rendimiento del evaluado.

- **Uso de la rúbrica**

La elaboración de las rúbricas para la evaluación de proyectos de diversas especialidades de la PUCP se encuentra, hasta la fecha, formalizada. Se presenta un ejemplo, de cómo está estructurada una rúbrica en la especialidad de Ingeniería Informática.

DP2	Aspecto	Criterio/Nivel	1	2	3	4
A (Aplica los conocimientos relacionados a las matemáticas, ciencias e ingeniería)						
	Matemáticas	Lógica	Aplicar operaciones lógicas (causa-efecto) en situaciones simples de manera deficiente	Aplicar operaciones lógicas (causa-efecto) en situaciones simples	Aplicar operaciones lógicas (causa-efecto) en situaciones complejas	Establecer soluciones integradas de manera lógica en problemas simples
	Ing. Informática	Algoritmos	Ser capaz de leer código fuente en lenguaje de alto nivel y entender parcialmente el algoritmo	Ser capaz de leer código fuente en lenguaje de alto nivel y entender el algoritmo	Tener la capacidad de modificar un algoritmo	Desarrollar el algoritmo nuevo a partir de una especificación
		Lenguajes de programación	Conocer paradigmas de programación	Aplicar paradigmas de programación	Implementar un algoritmo	Utiliza patrones de programación
B (Diseña y conduce experimentos, así como analiza e interpreta los datos)						
	Diseña	Evolución del experimento	Tener problemas al identificar y definir las variables involucradas	Identificar y definir las variables involucradas	Establecer hipótesis de trabajo	Establecer el grado de confianza del experimento
	Conduce e interpreta resultados	Evolución del experimento	Ejecutar con dificultad un experimento	Ejecutar un experimento	Presentar resultados en formatos organizados	Interpreta los resultados

Tabla 1-2 Extracto de Rúbrica del curso INF227 de la Especialidad Ingeniería Informática

1.2.5. Evaluación cualitativa

La evaluación cualitativa hace hincapié en las experiencias humanas; es decir, se valora más la calidad de cómo se realiza una actividad.

Esta evaluación se caracteriza por:

- Seguir un enfoque inductivo para la recolección de datos, interpretación y presentación de informes.
- Mantener un enfoque global: hallazgos para los resultados de la evaluación.
- Lograr la comprensión de lo subjetivo.
- Considerar las experiencias vividas por los participantes del programa.
- Usar lenguaje natural en toda la evaluación del proceso.
- Recopilar datos detallados.
- Realizar estudios de caso.

La diferencia de la evaluación tradicional con la evaluación cualitativa es que ésta es cuantitativa y se caracteriza en el énfasis de procedimientos de medición que se prestan a representación numéricas (McDavid, 2006).

1.2.6. Competencia

Una competencia se define como “pericia, aptitud, idoneidad para hacer algo o intervenir en un asunto determinado” (RAE 2012). Asimismo, si se enfoca en términos de educación, las competencias son estructuras complejas de comportamientos, es decir son una combinación de recursos que son movilizados para lograr un desempeño. Se clasifican en dos categorías: las competencias genéricas y las competencias específica (Tobón, 2008).

- **Competencia genérica:** son las competencias que son comunes a una rama profesional (por ejemplo, salud, ingeniería, educación) o a todas las profesiones.
- **Competencia específica:** son las competencias que son propias de cada profesión (por ejemplo, competencias específicas del profesional en ingeniería informática o del profesional en psicología).

Las competencias vienen siendo aceptadas en las universidades ya que comienzan a ser objetivo de interés académico, ya que las competencias de educación

superior requieren desarrollar el conocimiento teórico para posteriormente dominar la práctica profesional (Buja, 2011).

1.2.7. Evaluación por competencias

En los últimos años, la evaluación ha pasado por varios cambios debido a la innovación curricular a todas las nuevas estrategias didácticas que se han incorporado en la formación de los estudiantes (Tobón, 2008). La evaluación tradicional, básicamente cuantitativa, es la forma más común y antigua utilizada por los docentes a nivel mundial, tanto en la educación superior como en otros niveles de educación. Sin embargo, debido a la necesidad de evaluar elementos distintos que han surgido a causa de los cambios ocurridos recientemente, se pensó en una evaluación que además de enfocarse en la evaluación del contenido de los programas de estudio con el fin de obtener resultados cuantitativos (por ejemplo: promedios), también se pueda evaluar los momentos y así reconocer debilidades o fortalezas que sean relevantes en los procesos de aprendizaje en diversos contextos. Surge así, lo denominado evaluación por competencias (Buja, 2011).

La evaluación por competencias trata de una evaluación cualitativa la cual busca determinar los logros de los estudiantes y el nivel de dominio de una competencia en base a criterios, los cuales se adquieren de forma creciente, sea dinámico o evolutivo. Es necesario, para este tipo de evaluación, contar con un proceso diferente de analizar las competencias; por ello, se requiere del uso de estándares y rúbricas (Minte, 2008).

A continuación se presentan las formas más utilizadas para evaluar por competencias (Minte, 2008):

- Evaluación de pares
- Evaluación del trabajo en grupo o en equipo
- Informes
- Entrevistas en distintas formas como el caso de la entrevista dirigida.
- Trabajos prácticos
- Creación de un producto
- Cuestionarios
- Trabajos de investigación
- Desarrollo de proyectos de diversas índoles (grupales o individuales)
- Autoevaluación
- Portafolios

- Evaluaciones electrónicas
- Actividades de prácticas pre-profesional y profesional

1.2.8. **Acreditación**

El término acreditación se define como “documento que acredita la condición de una persona y su facultad para desempeñar determinada actividad o cargo” (RAE, 2012).

➤ **Acreditación Internacional**

En este caso, cuando una organización, como la PUCP, desea realizar un plan de mejora continua del servicio educativo que brinda, busca un reconocimiento público de la calidad de una carrera universitaria, y para ello debe pasar por un proceso de requisitos de acreditación, el cual implica que se deba cumplir un conjunto de estándares. Esta acreditación es otorgada por una organización externa a dicha institución (DAPE, 2007).

Los estándares internacionales que se requieren para la acreditación a través de los años han ido en aumento y son más exigentes. Se presentan los criterios de acreditación de uno de las prestigiosas acreditadoras CEAB.

- **Canadian Engineering Accreditation Board (CEAB):** El Consejo de Acreditación de Ingeniería de Canadá, son los encargados de acreditar los programas canadienses de ingeniería de pregrado que cumplen o superan los estándares educativos que deben ser aceptables para permitir el registro profesional en Canadá (CEAB, 2011). Algunos criterios de acreditación que considera este organismo y que debe cumplir un programa son (CEAB, 2011):

- **Una base de conocimientos de la ingeniería:** Competencia en matemáticas a nivel universitario, ciencias, fundamentos de ingeniería y especializados de ingeniería apropiado para el programa de conocimientos.
- **Análisis de problemas:** La capacidad de utilizar los medios conocimientos y habilidades para identificar, formular, analizar y resolver problemas complejos de ingeniería con el fin de llegar a conclusiones fundamentadas.

- **Investigación:** La capacidad para realizar investigaciones de problemas complejos por métodos que incluyen experimentos adecuados, análisis e interpretación de datos, y síntesis de la información con el fin de llegar a conclusiones válidas.
- **Diseño:** La capacidad de diseñar soluciones para los complejos, problemas abiertos de ingeniería y el diseño de sistemas, componentes o procesos que cumplan necesidades específicas con la debida atención a riesgos de salud y seguridad, normas aplicables, y económicos, ambientales, culturales y sociales.

Desde el año 2004, la Facultad de Ciencias e Ingeniería de la PUCP con el apoyo del Instituto de Calidad (IC-PUCP) participa en el proyecto de mejora de procesos para la acreditación. (DAPE, 2007).

1.2.9. Plan estratégico Institucional

Con el deseo de mejorar la calidad de la educación en la comunidad PUCP, se presentó una propuesta para elaborar el plan estratégico institucional que formalice todos los procesos y actividades que buscan la calidad. Por ello en el 2007, se formalizó en un documento que trata de una gestión organizada en torno a proyectos (DAPE, 2007).

El plan contempla la consecución de logros a nivel de los procesos esenciales de docencia e investigación, los cuales serán alcanzados a través del trabajo coordinado de la comunidad académica de la PUCP. Dichos resultados surgen como expresión deliberada de lo que nuestros académicos visualizan como necesario y urgente para la institución. Asimismo, son propuestas de logro que han tomado en cuenta un análisis serio de las actuales mega tendencias para la educación superior, a nivel nacional e internacional (DAPE, 2007).

1.3. Estado del Arte y de la Práctica

En esta sección se presenta el estado del arte, una tabla de comparación de los productos revisados en el estado del arte y el estado de la práctica.

1.3.1. Estado del Arte

Se presenta una revisión de las principales soluciones informáticas existentes en el mercado y desarrollados por los propios docentes para el soporte a procesos de evaluación y afines basado en rúbricas.

- **Uso de hojas de cálculo**

El uso de hojas de cálculo es la herramienta más utilizada por los evaluadores y estudiantes para la evaluación basada en rúbricas.

	B	C	D	E	
1	Aspecto	Criterio/Nivel	1	2	
2	A (Aplica los conocimientos relacionados a las matemáticas, ciencias e ingeniería)				
3	Matemáticas	Lógica	Aplicar operaciones lógicas (causa-efecto) en situaciones simples de manera deficiente	Aplicar operaciones lógicas (causa-efecto) en situaciones simples	Aplicar op efecto) er
4	Ing. Informática	Algoritmos	Ser capaz de leer código fuente en lenguaje de alto nivel y entender parcialmente el algoritmo	Ser capaz de leer código fuente en lenguaje de alto nivel y entender el algoritmo	Tener la c algoritmo
5		Lenguajes de programación	Conocer paradigmas de programación	Aplicar paradigmas de programación	Implemen
6	B (Diseña y conduce experimentos, así como analiza e interpreta los datos)				
7	Diseña	Evolución del experimento	Tener problemas al identificar y definir las variables involucradas	Identificar y definir las variables involucradas	Establece
8	Conduce e interpreta resultados	Evolución del experimento	Ejecutar con dificultad un experimento	Ejecutar un experimento	Presentar organizad
9	C (Diseña sistemas, componentes o procesos que satisfagan las necesidades presentadas)				

Figura 1-1 Ejemplo de uso de hojas de cálculo en la elaboración de rúbricas. (Extracto de la Rúbrica de la Especialidad de Ingeniería Informática PUCP)

- **Sistema avanzado de calificaciones de MOODLE**

Moodle, es un Sistema de Código Abierto para Gestión de Cursos (Open Source Course Management System, CMS). Es una aplicación web gratuita donde los educadores puedan desarrollar el aprendizaje virtual basándose en diversas herramientas que le ofrece este sistema (Moodle, 2012).

Desde la versión 2.2 de Moodle, se aprecia una nueva característica que consiste en un sistema de calificación basado en rúbricas que permite al evaluador la definición de niveles por criterio al momento de evaluar a un estudiante (Moodle, 2012).

- **Principales Características**

Respecto al sistema avanzando de calificación de rúbricas que se encuentra integrado en el módulo “Actividades” de Moodle:

- Fue desarrollado por Moodle HQ, pero fue inspirado por Moodle Rooms, organización que provee soluciones e-learning.
- Proporciona un editor de rúbricas.
- Provee métodos de evaluación de las calificaciones basado en rúbricas.
- Las rúbricas son el primer plug-in de un nuevo tipo de plug-in de “Calificación Avanzada”
- Fue creado por Martin Dougiamas en Australia
- Se integró al sistema MOODLE en el año 2011.

➤ **Editor de rúbricas**

Para comenzar a realizar la elaboración de rúbricas se siguen los siguientes pasos:

1. Se selecciona “Clasificación Avanzada” en el bloque de “Ajustes de Actividad” (Ver en Figura 1-2).
2. Se procede a colocar el nombre, la descripción de la rúbrica.
3. Luego se procede a definir los criterios seleccionando “Agregar Criterio”.
4. En cada criterio se procederá a indicar los niveles que se considere necesario. Para ello se selecciona “Agregar nivel”.
5. En cada nivel, se define el nivel y el número de puntos asociados con el nivel.
6. Finalmente se procede a guardar la rúbrica. Se selecciona para ello “Guardar rúbrica y que quede listo” o “Guardar como borrador”.

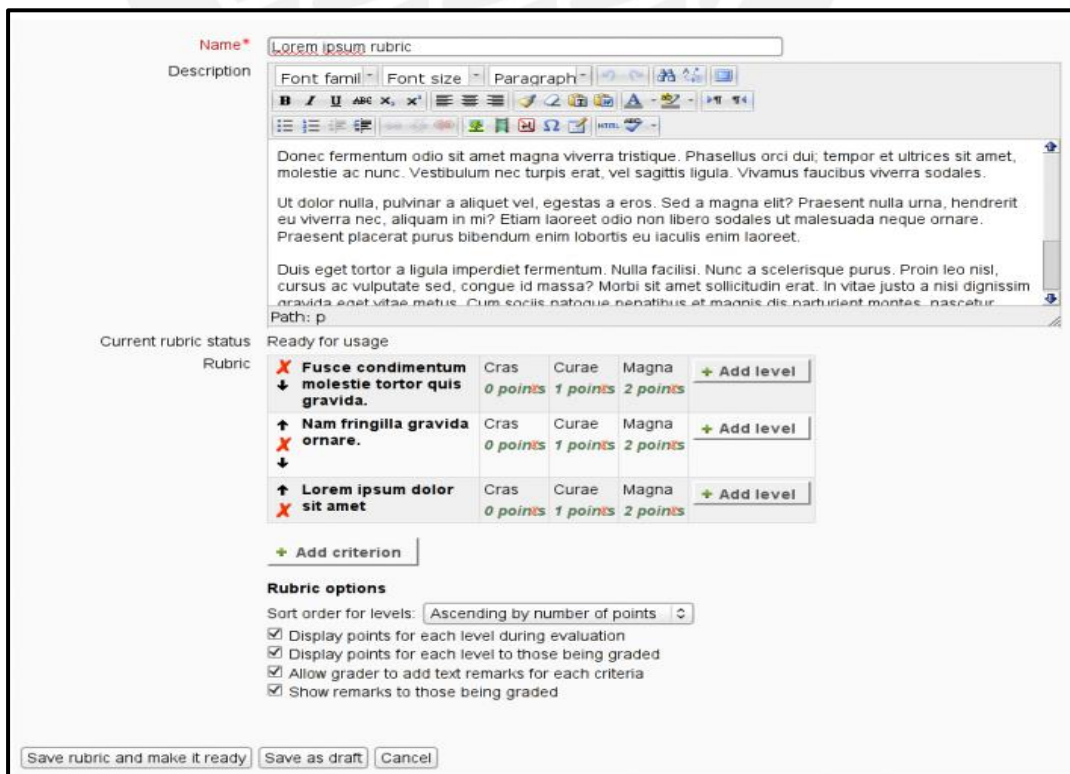


Figura 1-2 Editor de rúbrica de Moodle versión 2.2

➤ **Evaluación basada en rúbrica**

Luego de la elaboración de la rúbrica se procede a explicar los pasos que se sigue en este sistema para realizar la evaluación basada en rúbricas.

1. El evaluador al visualizar la rúbrica selecciona el nivel que mejor describa el rendimiento del evaluado (Ver en Figura 1-3).
2. Los niveles que han sido seleccionados se resaltan en verde claro. Si en caso la rúbrica es editada luego de realizar la selección el nivel estará resaltado en rojo. Lo válido es seleccionar un solo nivel por criterio.
3. Si se desea realizar comentarios al criterio se puede especificar opcionalmente, los cuales pueden servir para fines de retroalimentación.

Grades

Grade:

Fusce condimentum molestie tortor quis gravida.	Cras 0 points	Curae 1 points	Magna 2 points	
Nam fringilla gravida ornare.	Cras 0 points	Curae 1 points	Magna 2 points	
Lorem ipsum dolor sit amet	Cras 0 points	Curae 1 points	Magna 2 points	

Current grade in gradebook: 7.00

Figura 1-3 Evaluación en el Sistema Avanzado de Calificación basado en rúbricas.

➤ **Cálculo de la Calificación**

La rúbrica contiene una puntuación con grados de porcentaje y se calcula de la siguiente manera:

$$G_s = \frac{\sum_{i=1}^N (g_i - \min_i)}{\sum_{i=1}^N (\max_i - \min_i)}$$

Figura 1-4 Fórmula de Cálculo de Calificación Moodle

Dónde:

- **gi**: es el número de puntos dado el i-ésimo criterio.
- **min i**: es el mínimo número posible de puntos para el i-ésimo criterio.
- **max i**: es el máximo número posible de puntos para el i-ésimo criterio.
- **N**: es el número de criterios de la rúbrica.

Ejemplo:

Se definen 2 criterios con 4 niveles cada uno (1, 2, 3, 4). El evaluador elige el nivel 2 para el primer criterio y 3 para el segundo criterio. Para ello se utiliza la fórmula de la siguiente manera:

$$G_s = \frac{(2 - 1) + (3 - 1)}{(4 - 1) + (4 - 1)} = \frac{3}{6} = 50\%$$

Figura 1-5 Ejemplo de Uso de Fórmula Calificación Moodle 2.2

- **RubiStar**

RubiStar es una herramienta Web libre para ayudar al docente a realizar rúbricas (RubiStar, 2012).

- **Principales Características:**

- Fue creada por el gobierno de los Estados Unidos.
- Todas las rúbricas están traducidas al castellano.
- Permite modificar las rúbricas en función de las diferentes actividades y trabajos planteados.
- Permite descargar y usarlo en una hoja de cálculo
- Está enfocada a facilitar la elaboración de la rúbrica, no automatiza la evaluación basado en las mismas.

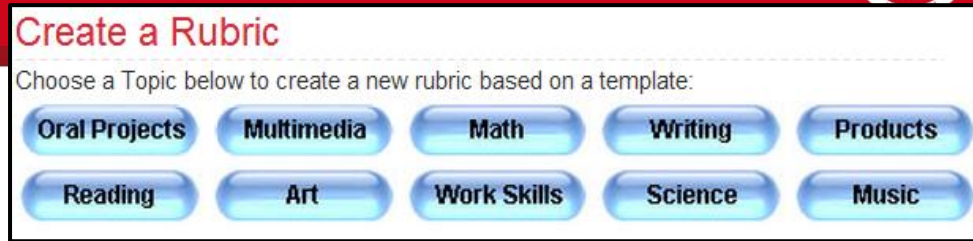


Figura 1-6 Rúbricas de diversas actividades (RubiStar, 2012)

Figura 1-7 Registrar información básica de la rúbrica (RubiStar, 2012)

Figura 1-8 Edición de la rúbrica, definiendo niveles y criterios (RubiStar, 2012)

➤ **SkillStation**

SkillStation es un software el cual ayuda a gestionar las competencias del personal dentro de una organización (SKILLSTATION, 2012).

➤ **Principales Características:**

- Fue desarrollado por la compañía e4 Learning Solution en Inglaterra.
- Permite revisar e informar sobre la competencia y la no competencia dentro de su organización.
- Con la evaluación que realiza permite identificar y desarrollar a las personas adecuadas para determinadas funciones dentro de la organización.

• **ECO**

Su nombre proviene de Evaluación por Competencias, es una aplicación informática para Windows que facilita la evaluación de competencias desde la perspectiva de trabajo en cada curso. La idea surgió en respuesta a la legislación que establece competencias básicas en la educación de forma obligatoria.

➤ **Principales Características:**

- Fue diseñado y desarrollado en el año 2009 en Valencia, España, por Miguel Ángel Jiménez, vicedecano de psicopedagogía de la Universidad Católica de Valencia “San Vicente Martin” (UCV) y el ingeniero informático Pablo Bayarri.
- Se basa en rúbricas para la evaluación.
- Los criterios de evaluación, el programa los define como rasgo.
- El sistema genera una tabla detallada de las calificaciones de cada área por separado y de las competencias de tal forma que pueda ofrecer a los estudiantes y sus padres la información académica (UCV, 2009).
- Permite realizar la retroalimentación de una evaluación dada.

• **Talent Test v3.0**

Es un software para la evaluación de competencias laborales genéricas para determinar si cualquier candidato, aspirante a un puesto de una empresa, alcanza los logros requeridos (TalentTest, 2012).

➤ Principales Características:

- Fue desarrollado por una empresa en México.
- Permite entregar el nivel de desempeño de más de 150 destrezas laborales a través del tiempo, donde se logra visualizar el crecimiento o decrecimiento de la persona.
- No es libre, tiene un costo.
- Tiene tres versiones: Premium, Habilidades y Facetas.
- Evaluación consistente basada en estándares.
- Provee de información de la evaluación al evaluado.
- Permite tomar decisiones a partir de la evaluación.
- Mide el desempeño a través del tiempo.

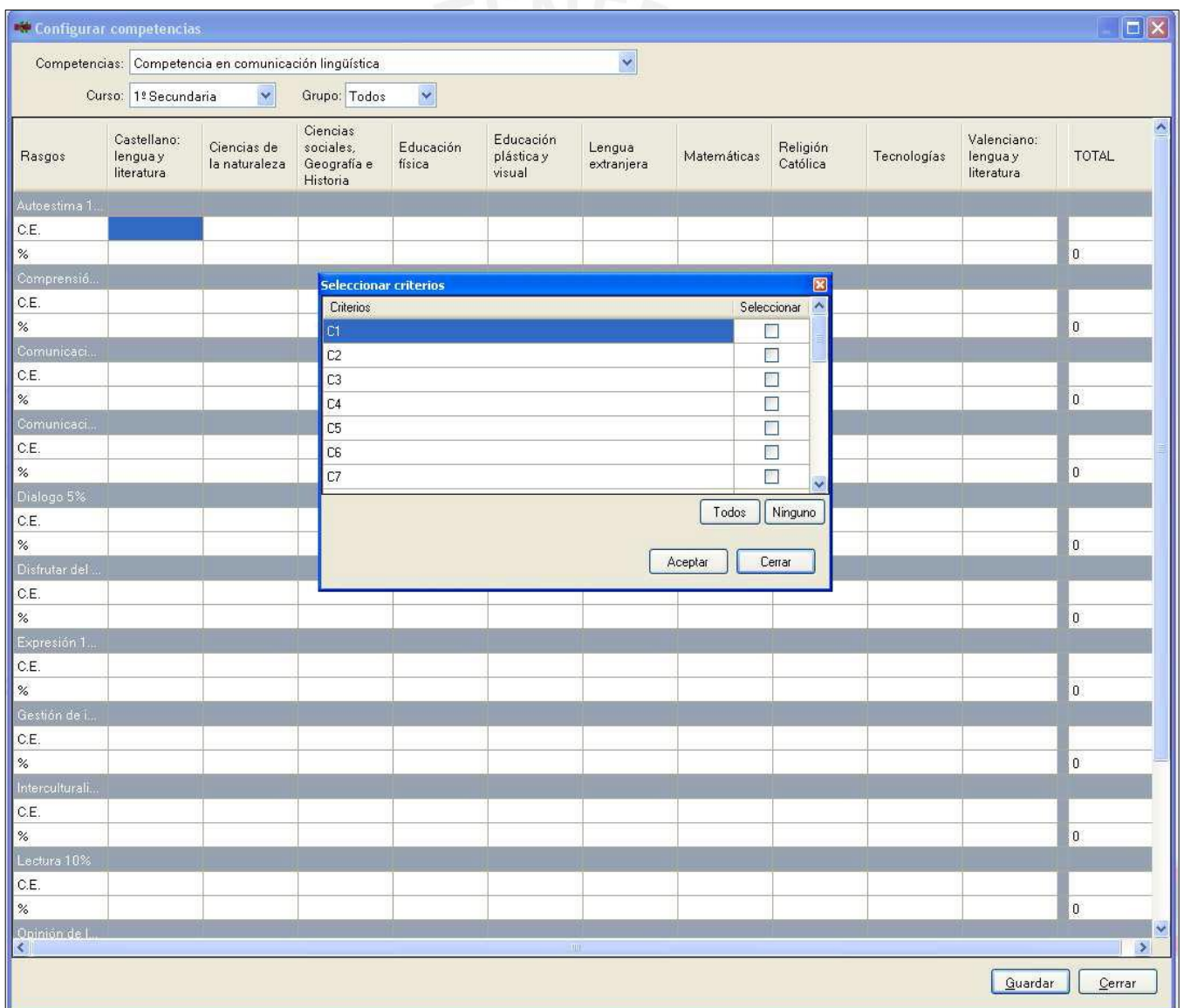


Figura 1-9 Definición de criterios en ECO (ECO, 2012)



Figura 1-10 El sistema Talent Test v3.0 permite evaluar al trabajador para un determinado puesto (Talenttest, 2012).

- **Sistema de Evaluación de Desempeño**

Este sistema fue presentado como proyecto de fin de carrera con mención en Ingeniería Informática de la PUCP (Martínez, 2005).

- **Principales Características:**

- Se centra en evaluar el desempeño de los proyectos en los cursos de fin de carrera de la facultad de Ingeniería Informática.
 - Utiliza el concepto de criterios y estándar para la evaluación.
 - Ofrece visualización de los resultados de la evaluación de desempeño.

- **Diferencias con el presente proyecto:**

- No se basa en calificaciones basadas en rúbricas.
 - El contexto sólo abarca a la especialidad de Ingeniería Informática.
 - No se centra en obtener resultados para fines de acreditación

- **Herramienta Web de Evaluación del Desempeño por Competencias- Evaluación de 360 grados.**

Esta herramienta Web fue presentada como proyecto de fin de carrera con mención en Ingeniería Informática de la PUCP (Vivanco, 2010).

- **Principales Características:**

- Se enfoca en evaluar competencias en el área laboral.
 - Permite evaluar al personal para comprobar si cumple el perfil del puesto en el cual se encuentra.
 - Facilita la retroalimentación a los evaluados

1.3.2. Cuadro Comparativo

En esta sección se presenta la comparación de los productos revisados en el estado del arte en la Tabla 1-3.

SOLUCIÓN	USO DE HOJA DE CÁLCULO	SISTEMA AVANZADO CALIFICACIÓN MOODLE	RUBISTAR	SKILL STATION	ECO	TALENT TEST V3.0	SISTEMA DE EVALUACIÓN DE DESEMPEÑO	HERRAMIENTA WEB DE EVALUACIÓN DEL DESEMPEÑO
País de Procedencia	Depende	Australia	EEUU	Inglaterra	España	México	Perú	Perú
Licenciado/ Libre	Depende	Libre	Libre	Licenciado	Licenciado	Licenciado	Libre	Libre
Aplicación Web	Depende	Sí	Sí	No	No	No	Sí	Sí
Aplicación de Escritorio	Depende	No	No	Sí	Sí	Sí	No	No
Área/Objetivo	Educación	Educación	Cualquier área especificada en la web	Laboral	Educación	Laboral	Educación	Laboral
Elaboración de Rúbricas	Sí	Sí	Sí	No	Sí	Sí	No	No
Evaluación basado en rúbricas	Permite	Sí	No	No	Sí	Sí	No	No
Otras funcionalidades	No	Retroalimentación	Permite descarga de hojas de cálculo.	Retroalimentación.	Retroalimentación	Información resultados de evaluación	Información resultados de evaluación	Retroalimentación

Tabla 1-3 Cuadro Comparativo del Estado del Arte

En todos los software y herramientas revisados se pueden observar que hacen posible la evaluación cualitativa de diversos puntos de vista. Las principales herramientas, como hojas de cálculo, son mayormente utilizadas para la solución de la problemática. Asimismo, existen otros programas software que buscan automatizar el uso de hojas de cálculo pero que han sido orientados a evaluación de competencias en cursos de diferentes niveles educativos o evaluación de competencias en el ámbito laboral, o simplemente proporcionan un entorno donde realizar la elaboración de rúbricas, más no aportando a la evaluación basada en las mismas.

El presente proyecto propone desarrollar un sistema con un enfoque diferente, el cual ayudará básicamente a automatizar la evaluación de los resultados del programa orientado principalmente a proyectos universitarios con la finalidad de agilizar el proceso de dicha evaluación permitiendo así que los resultados se hagan de una forma precisa, rápida y de fácil acceso tanto para estudiantes y evaluadores.

1.3.3. Estado de la Práctica

En esta sección se procederá a explicar el estado de la práctica de la gestión de evaluaciones basadas en rúbricas en los cursos INF226 e INF227 de la especialidad de Ingeniería Informática de la PUCP.

Al iniciar el curso, el docente se encarga de realizar la distribución de equipos de estudiantes y asignación de asistentes de docencia. Una vez conformados los equipos, el docente utiliza una rúbrica ya elaborada, donde contienen criterios y niveles ya establecidos. Luego de tener todo definido, el docente empieza a gestionar la evaluación de pares, el cual es una evaluación realizada entre equipos de trabajos, de acuerdo a ciertas medidas que el docente indicará.

Posteriormente, el docente proporciona a los miembros de cada equipo los criterios y niveles de la rúbrica con el cual se va a realizar la evaluación de desempeño para que se evalúen de forma interna en el equipo.

La proporción de información que se le da al estudiante de la rúbrica se gestiona mediante correo (en este caso el correo de la universidad). Para más detalle, el docente envía un correo con el contenido de la rúbrica a los equipos y luego los miembros de estos equipos deben enviarle la evaluación ya realizada por el mismo medio de comunicación. Luego en el caso particular de este curso los miembros de los equipos cambian, así como la asignación de asistentes de docencia que también van variando entre equipos.

Este proceso, resulta complejo cuando se gestionan varios equipos y varios cursos para un mismo docente y asistentes de docencia. Esto se debe a que, luego de realizada la evaluación, se debe realizar la retroalimentación respectiva, volviendo a utilizarse el correo electrónico de la universidad. Además, no sólo se trata de una evaluación de desempeño por cursos, sino que el docente gestiona varias evaluaciones durante el ciclo académico.

1.4. Propuesta del Proyecto

Se presenta en esta sección la propuesta del presente proyecto.

1.4.1. *Objetivo general*

Desarrollar un sistema de información para la gestión de la evaluación de estudiantes mediante el uso de rúbricas en cursos orientados a proyectos en el contexto de un programa acreditado.

1.4.2. *Objetivos específicos*

Los objetivos específicos del presente proyecto de fin de carrera son los siguientes:

1. Definir las reglas que se siguen para la realización de las evaluaciones del logro de los resultados del programa basado en rúbricas.
2. Identificar las estructuras de datos necesarias para la recolección y consolidación de datos del logro de los resultados del programa.
3. Definir una arquitectura que permita la adaptación del software a configuraciones de evaluación por pares y por docentes.
4. Construir y probar el software con datos obtenidos de un curso de proyectos del programa de la carrera de Ingeniería Informática de la PUCP.

1.4.3. *Resultados esperados*

Los resultados esperados que se presentan a continuación corresponden respectivamente a cada uno de los objetivos específicos (OE) antes indicados según su número:

1. Informe técnico donde se describen las reglas que se siguen para la realización de las evaluaciones del logro de los resultados del programa (OE1).
2. Informe técnico con las estructuras de datos que permitan la consolidación y recolección de datos del logro de los resultados del programa (OE2).
3. Informe técnico de la arquitectura de software (OE3).
4. Software desarrollado y probado en un curso de proyectos del programa de Ingeniería Informática (OE4).

1.4.4. Métodos, metodologías y procedimientos

En esta sección se presenta el mapeo de los resultados esperados con los métodos, metodologías y procedimientos utilizados en el presente proyecto para el logro de los resultados esperados.

- **Mapeo:** En esta sección se realiza un mapeo de las herramientas, métodos y metodologías para cada uno de los resultados esperados.

Resultados Esperados	Herramientas/Métodos/Metodologías
<p>RE1: Informe técnico donde se describen las reglas que se siguen para la realización de las evaluaciones del logro de los resultados del programa.</p>	<ul style="list-style-type: none"> • Guía de Reglas de Negocio: clasificación que distingue a las reglas de negocio. • Historias de Usuario: representación para la toma de requisitos funcionales del sistema (XP, 2013). • Maqueta o modelo a escala (Mockup): la elaboración de prototipos es una técnica que ayuda a determinar los requisitos funcionales del cliente.
<p>RE2: Informe técnico con las estructuras de datos que permitan la consolidación y recolección de datos del logro de los resultados del programa.</p>	<ul style="list-style-type: none"> • MySql WorkBench: herramienta para modelar diagramas de entidad relación para base de datos MySql. • UML 2.0: lenguaje unificado de modelado para el diseño de diagramas de análisis. • Bitácora: hoja de cálculo que se utilizan actualmente para la gestión del proceso.

Resultados Esperados	Herramientas/Métodos/Metodologías
<p>RE3: Informe técnico de la arquitectura de software.</p>	<ul style="list-style-type: none"> • Gliffy: es un editor de diagramas para el diseño de la arquitectura de software. • CodeIgniter: framework que implementa el patrón MVC
<p>RE4: Software desarrollado y probado en un curso de proyectos del programa de Ingeniería Informática.</p>	<ul style="list-style-type: none"> • Extreme Programming (XP): planificación, versiones pequeñas, diseño simple, pruebas, integración del código. • CodeIgniter: Es un framework que implementa el patrón MVC (Modelo, Vista y Controlador). • Netbeans 7.2.1: entorno de desarrollo para la programación del producto • XAMPP: servidor independiente de plataforma, software libre que consiste en base de datos MYSQL, servidor web Apache y PHP.

Tabla 1-4 Mapeo Herramientas, Métodos y Metodologías

- **Metodologías**

Siendo un proyecto un esfuerzo temporal, con una fecha de inicio y de fin, es necesario elegir principios para una adecuada gestión, que permita dirigir el proyecto en todo su ciclo de vida. Asimismo, es importante elegir principios de una metodología para el desarrollo de los resultados esperados.

Para el presente proyecto, se ha decidido hacer uso de principios del marco de trabajo de SCRUM los cuales son aplicados para la gestión del presente proyecto de tesis. Por otro lado, para hacer frente al desarrollo del producto, los principios utilizados son de la metodología Extreme Programming (XP).

A continuación se presentan los principios utilizados de ambas metodologías para el presente proyecto y las razones por las cuales se están utilizando.

- **Principios utilizados para la Gestión del Proyecto**

Para el logro de los objetivos del proyecto es necesario el uso de principios que guíen el cumplimiento de su gestión. Scrum, un marco de trabajo ágil, es un proceso iterativo e incremental basado en el trabajo en equipo, apoyándose en iteraciones conocidas como *Sprints* (Scrum, 2013).

El marco de trabajo de Scrum destaca el valor del producto sobre la documentación ya que finalmente es el que genera valor para los usuarios (SCRUM, 2013). Por ello, la elección de este marco de trabajo trasciende en obtener resultados que reflejen el producto en sí, considerando el tiempo relativamente corto y cumpliendo con las necesidades del cliente. Uno de los requisitos trascendentales de las prácticas de este marco de trabajo es contar con constantes reuniones con el cliente para la planificación de los Sprint. Por ello, las reuniones fueron registradas de tal forma que se cumpla con lo indicado. Asimismo, uno de las principales componentes de Scrum es el *Product Backlog*, que consiste en la lista de funcionalidades (del producto a desarrollar) ordenadas por prioridad.

A continuación se empieza a detallar qué prácticas se toman de Scrum para la gestión del proyecto de final de carrera. Se consideran dos aspectos importantes: Roles y Actividades para el detalle.

- **Roles:** Scrum define tres roles importantes: Product Owner, Scrum Master y Team (Sutherland, 2007). En la gestión del presente proyecto se detalla de la siguiente forma:
 - ❖ **Product Owner:**

En este proyecto la persona encargada fue un docente del curso de Desarrollo de Programas 1 y Desarrollo de Programas 2 de la carrera de Ingeniería Informática. La elección del docente como Product Owner se justifica porque es el que conoce los procesos de la gestión de evaluación basados en rúbricas en el contexto de una carrera acreditada.
 - ❖ **Scrum Master:**

Apoya al equipo para la aplicación correcta del marco de trabajo. Sin embargo, como no se trata de un equipo en sí, ya que el proyecto es desarrollado por una sola persona, el rol lo asume la persona que desarrolla este proyecto.
 - ❖ **Team:**

Por las razones explicadas al definir el Scrum Master, el proyecto es desarrollado por una sola persona, el cual conforma el equipo cumpliendo con lo requerido por el Product Owner.
- **Fases:** El marco de trabajo Scrum describe cinco fases importantes: Realización de la Pila del Producto, Planificación del Sprint, Realización

de los elementos del Sprint, Revisión del Sprint y Retrospectiva del Sprint (SCRUM, 2011). A continuación se presentan las consideraciones realizadas para aplicar las fases en este proyecto.

❖ **Realización de la Pila del Producto**

Consiste en determinar los requisitos del producto para el logro de los objetivos del proyecto. Scrum no define qué herramientas utilizar para el levantamiento de esta información, por lo que deja libertad en cómo realizarlo. En este proyecto, por lo tanto, se definieron requisitos con: entrevistas con los usuarios, desarrollo de prototipos y realización de estos requisitos finalmente en historias de usuario.

❖ **Planificación del Sprint**

Es importante que cada Sprint se planifique adecuadamente. Para el desarrollo de este proyecto se estableció la duración de cada Sprint de acuerdo a los esfuerzos y prioridades de cada entregable. Para cada Sprint se consideró la asistencia del Product Owner, la obtención de historias de usuario y sus prioridades.

❖ **Realización de los elementos del Sprint**

Se realiza las actividades definidas en la planificación del Sprint que incluyen las funcionalidades del sistema de información con la finalidad de presentar organización según los tiempos establecidos para cada uno de ellos. En el proyecto, el cumplimiento de los elementos de la Pila de Sprint se controló, de tal manera que al final se logró un sistema de evaluación basado en rúbricas.

❖ **Revisión del Sprint**

Se estimó un día para la revisión del Sprint, donde se debió concluir el incremento construido del sistema al cliente con las funcionalidades que se especificó en la Pila de Sprint.

❖ **Retrospectiva del Sprint**

Al finalizar la revisión se debe obtener una retroalimentación para generar mayor valor al producto final. Se estimó un día para la realización de la misma a través de reuniones donde deben estar presentes el Product Owner y la persona que está llevando a cabo el proyecto.

Las prácticas del marco de trabajo Scrum que no fueron tomadas para realizar la gestión del presente proyecto fueron básicamente las que se dirigían a gestión del equipo de trabajo. Este proyecto fue desarrollado por una sola persona, por ello no se usaron aquellos principios.

➤ **Principios utilizados para el Desarrollo del Producto**

La metodología que se utiliza para el desarrollo del producto del presente proyecto es Extreme Programming. La elección de adoptar algunas prácticas de esta metodología se debe a que se enfoca en el desarrollo del producto y se complementa bien con el marco de trabajo ágil Scrum. Se presentan, a continuación, algunas de prácticas básicas del XP para el desarrollo del sistema de evaluación basadas en rúbricas.

❖ **Versiones Pequeñas**

Las versiones de cada entregable fueron pequeñas realizados en dos semanas. Estas versiones ofrecieron resultados funcionales del producto según las prioridades establecidas en Planeación.

❖ **Diseño Simple**

Se elaboró un diseño simple de las funcionalidades implementadas que pueden ser modificados según las necesidades de la implementación. Se estableció un modelo de entidad relación para la definición de las estructuras de datos que forman parte de este diseño.

❖ **Integración Continua**

El proyecto fue realizado por una sola persona, lo cual permitió que el código se mantenga integrado.

❖ **Test**

El desarrollador del producto estableció pruebas para validar las versiones pequeñas.

Los principios que no fueron tomados de la metodología XP para realizar la implementación del producto del presente proyecto fueron básicamente las que se dirigían a las tareas en el equipo de trabajo. Este proyecto fue desarrollado por una sola persona, por ello no se usan aquellos principios. También, se evitaron algunos principios por tratarse de un proyecto a corto plazo.

- **Herramientas**

Se presenta la descripción de las herramientas utilizadas para el cumplimiento de los resultados esperados. Asimismo se justifica la elección de cada una de ellas.

- **Guía de Reglas de Negocio:**

La descripción de las reglas de negocio se basó en la guía descrita por James Odell quien proporciona una clasificación entre las reglas de negocio como las reglas de restricción, reglas de operación, reglas de derivación entre otras (Odell, 1998). La elección de esta clasificación se basó en que recoge de manera práctica la identificación de las reglas de negocio y permite ser expresados en lenguaje natural, lo cual facilita el entendimiento de las reglas a los usuarios u otra persona ajena al proceso.

- **Historias de Usuario:**

Una historia de usuario es la representación para la toma de requisitos funcionales del sistema (XP, 2013). La elección de historias de usuario se debe a que no requiere de un lenguaje técnico porque lo primordial se centra en captar las necesidades de cliente.

- **Maqueta o Modelo (Mockup):**

Una maqueta o mockup es un modelo a escala del diseño de las pantallas del sistema (Interaction, 2013). La elección de estos prototipos fue por la necesidad de toma de requisitos del sistema y la recolección de las reglas de negocio. Asimismo permitió una visión más clara del producto para la construcción del producto ya que permitió que los usuarios indiquen sus comentarios de acuerdo al diseño presentado en los mockups.

- **UML2.0**

UML es el lenguaje unificado de modelado de software respaldado por el grupo OMG. Proporciona un lenguaje gráfico para el diseño de procesos y métodos de un sistema (UML, 2013).

La aplicación de este lenguaje gráfico se realizó para la diagramación de las estructuras de datos del sistema tales como el diagrama de clases de análisis.

➤ **MySql WorkBench**

Es una herramienta creada por la empresa SUN Microsystems que permite la elaboración de diagramas para base de datos relacionales (MYSQL, 2013). La elección de esta herramienta fue porque permite la representación de tablas, vistas y relaciones entre las entidades de las estructuras de datos. Asimismo la herramienta facilitó la sincronización con el manejador de base de datos MySql, la cual fue utilizada para el desarrollo de la solución del proyecto.

➤ **Bitácora**

Es una serie de hojas de cálculo utilizadas en un curso de la especialidad de Ingeniería Informática de la PUCP que permitió el análisis de la cantidad de información utilizado en la recolección de resultados del programa.

➤ **Gliffy**

Es una herramienta web para editar diagramas de aplicación para negocios (Gliffy, 2013). La elección de esta herramienta fue por la plataforma bien intuitiva de creación de diagramas y la sincronización con la cuenta de un repositorio, lo cual permite controlar versiones y facilita el seguimiento de las actualizaciones del mismo.

➤ **CodeIgniter**

Es un framework que implementa el patrón MVC para aplicaciones web con el lenguaje PHP (CodeIgniter, 2013). En el diseño de la arquitectura del sistema se utilizó este marco de trabajo. La elección de esta herramienta fue por la implementación del patrón arquitectónico definido en el proyecto.

Las herramientas NetBeans 7.3.1 y XAMPP son descritas a mayor detalle en el capítulo 3: **Construcción y Pruebas**.

1.4.5. Alcances y limitaciones

Se presentarán, a continuación, los alcances y limitaciones del presente proyecto de fin de carrera.

- **Alcance**

El proyecto sitúa su ámbito de aplicación en carreras universitarias reconocidas substancialmente o acreditadas por un ente acreditador, específicamente en carreras de ingeniería donde generalmente existen cursos de proyectos de fin de carrera que son necesarios evaluar de forma cualitativa. Se ha elegido este ámbito por la importancia de la cantidad de información que maneja la evaluación de estos cursos, los cuales al pertenecer a una carrera acreditada, deben mantener la calidad de la enseñanza y por ello estas evaluaciones deben permitir proporcionar a los evaluadores resultados reales y precisos.

El proyecto va dirigido a aquellas personas relacionadas con el proceso de evaluación como docentes, asistentes de docencia y estudiantes.

Con este proyecto se buscó implementar las siguientes funcionalidades: registro de rúbricas, carga de cursos al sistema, gestión de cursos, definición de equipos y multiequipos de trabajo en un determinado curso, registro de planes de evaluación, evaluaciones basadas en rúbricas de estudiantes de forma individual o grupal, registro de las evaluaciones, indicadores y reportes finales de resultados de las evaluaciones del programa. Asimismo, se incluyeron el manejo de diversos roles que contarán con determinados permisos sobre el sistema.

El sistema de información será desarrollado en una plataforma web siguiendo estándares de codificación open source.

En cuanto a la metodología, para la administración del proyecto y el desarrollo del sistema de información propuesto se trabajará con algunos principios de la metodología SCRUM y Extreme Programming.

Adicionalmente, el proyecto considerará las características y reglas que determinan la evaluación basadas en rúbricas y resultados orientados a las mismas. En pocas palabras, lo que abordará este proyecto será: Desarrollar un software para el registro, evaluación y salida de resultados de las evaluaciones de cursos que están basadas en rúbricas.

- **Limitaciones**

Existen ciertas limitaciones para el desarrollo del proyecto las cuales se describen a continuación:

- **Disponibilidad** de los usuarios del sistema: se requiere que la comunicación con los usuarios sea constante durante el desarrollo del mismo, para que se pueda elaborar de manera correcta según las funcionalidades que se desee que tenga el sistema. Los usuarios, por falta de tiempo, pueden no encontrarse disponibles, lo que afectaría en el desarrollo del proyecto y posiblemente no se logre cumplir todas las expectativas que el usuario del sistema hubiera deseado.
- **Plazo del desarrollo de la solución del proyecto:** El desarrollo de la solución del proyecto comprende un plazo de 5 meses

1.4.6. Riesgos

Se presenta en la Tabla 1-5 los riesgos del presente proyecto de fin de carrera, los impactos que causan y las medidas correctivas para mitigar los mismos.

Riesgo identificado	Impacto en el proyecto	Medidas correctivas para mitigar
Pérdida de los documentos del proyecto de tesis.	Retraso en el plan de proyecto.	Uso de repositorio para el almacenamiento y respaldo de estos documentos, con manejo de control de versiones.
Falta de disposición por parte del cliente para el levantamiento de requisitos	Falta de información para el desarrollo de la solución del proyecto.	Establecer horarios flexibles para las reuniones con el cliente. Uso de medios de comunicación como aplicaciones con video llamadas.
Curva de aprendizaje lento para el uso de las herramientas del proyecto	Retraso en el plan de proyecto. Dedicar días del proyecto para aprender el uso de las herramientas.	Uso de manuales adecuados para el manejo de las herramientas.

Riesgo identificado	Impacto en el proyecto	Medidas correctivas para mitigar
Pérdida del código fuente de la solución del proyecto.	Retraso en el plan de proyecto. Dedicar días que no estaban planeados para rehacer lo avanzando.	Uso de repositorio para el almacenamiento y respaldo del código fuente, con manejo de control de versiones.

Tabla 1-5 Riesgos del Presente Proyecto

1.4.7. Justificación y viabilidad del proyecto

Se presentan, a continuación, la justificación y viabilidad del presente proyecto:

- **Justificación**

Las razones que justifican el proyecto son:

1. La necesidad de contar con una herramienta que automatice el registro de rúbricas y calificaciones de un determinado curso de proyectos y que permita facilitar esta labor a los docentes, los asistentes de docencia y los estudiantes durante cada período lectivo.
2. Los beneficios tales como la gestión organizada de las evaluaciones que permita la asignación de cursos, estudiantes y asistentes de docencia, con la finalidad de realizar las evaluaciones basadas en rúbricas.
3. El establecimiento de un canal de comunicación entre los docentes y estudiantes en lo que respecta al desempeño del equipo de trabajo en sus proyectos.
4. La gestión de las calificaciones apoyadas en un esquema de seguridad que permita la privacidad de las mismas.

- **Viabilidad**

Para poder determinar la viabilidad del proyecto se ha realizado el análisis correspondiente que comprende los siguientes aspectos:

- **Viabilidad Técnica**

Se plantea que el presente proyecto es un sistema web que requiere de una base de datos, donde el contenido del mismo permita registrar las evaluaciones de cursos de proyectos basados en rúbricas. Lo

mencionado es viable técnicamente debido a lo que se detalla a continuación:

- Para el desarrollo del software, desde el punto de vista técnico, se utilizaron herramientas a las cuales se tiene acceso y conocimiento. La solución requirió de un sistema de gestión de base de datos en la cual se almacene todos los datos para este proyecto. Por ello se utilizó el manejador de base de datos MySQL, el cual es accesible y trabaja muy bien cuando se desarrollan soluciones de software libre. El uso del lenguaje PHP es utilizado en la programación web.
- Como el desarrollo del sistema es web, también se necesitaron lenguajes y herramientas que facilitaron apoyo para su construcción, por lo que se utilizó Cascading Style Sheets (CSS) para el diseño de la página web y Chrome Developer Tool para monitorear, editar y depurar el código fuente CSS, HTML y Javascript. Todas las herramientas mencionadas interactúan con PHP y MySQL, por lo que el desarrollo de la solución es viable técnicamente.

➤ **Viabilidad Temporal**

El sistema será desarrollado en tiempo limitado, aproximadamente en 5 meses. Para el control del proyecto se tiene previsto usar el software Microsoft Project 2010 y la programación de reuniones con el asesor semanalmente.

➤ **Viabilidad Económica**

La estructura de costos, para el presente proyecto, se encuentra en los siguientes rubros: recursos informáticos, recursos materiales y recursos humanos.

Los recursos informáticos corresponden a las diferentes herramientas de software libre para programación y diseño del producto, por lo que no se requiere de la compra de ninguna licencia ni permiso para su uso.

La siguiente Tabla 1.6 muestra los recursos informáticos necesarios para el proyecto:

Descripción	Licencia
Motor de Base de Datos MySQL	No requiere
Lenguaje de Programación PHP	No requiere

IDE NetBeans	No requiere
Cascading Style Sheets (CSS)	No requiere
Chrome Developer Tool	No requiere
CodeIgniter	No requiere

Tabla 1-6 Recursos informáticos

Por otro lado, los recursos materiales que se emplean en el proyecto se centra en: impresión de informes, servicio eléctrico, servicio de internet y computadora personal portátil. El costo de estos recursos las asume el tesista debido a que son accesibles y en algunos casos se dispone de los mismos.

Asimismo, existen recursos humanos que se utilizan principalmente en las tareas de análisis, diseño, desarrollo del sistema y control de la gestión del proyecto, así como en el apoyo a las pruebas necesarias que requiere el producto. Para el presente proyecto, no se incurre en costos en este rubro debido a que el recurso humano fue la tesista.

Desde el punto de vista económico, el proyecto se considera viable, por las razones explicadas en cada uno de los componentes que conforman la estructura del costo presentada para el presente proyecto.

1.5. Planificación del proyecto

En esta sección se presenta cómo se realizó la planificación del presente proyecto. Las actividades previstas se presentan a través de una EDT (Estructura de descomposición de trabajo) en la Figura 1.13 y a través de un diagrama de Gantt en la Figura 1.14 y 1.15.

1.5.1. Estructura de Desglose de Trabajo (EDT)

La estructura de desglose de trabajo del proyecto comprende las actividades de: elaboración de tesis, la planificación del proyecto, el análisis y diseño del software, la implementación, el control del proyecto y el cierre del mismo. Para cada una de estas actividades se detallaron las sub- actividades que se presentan en la Figura 1.13

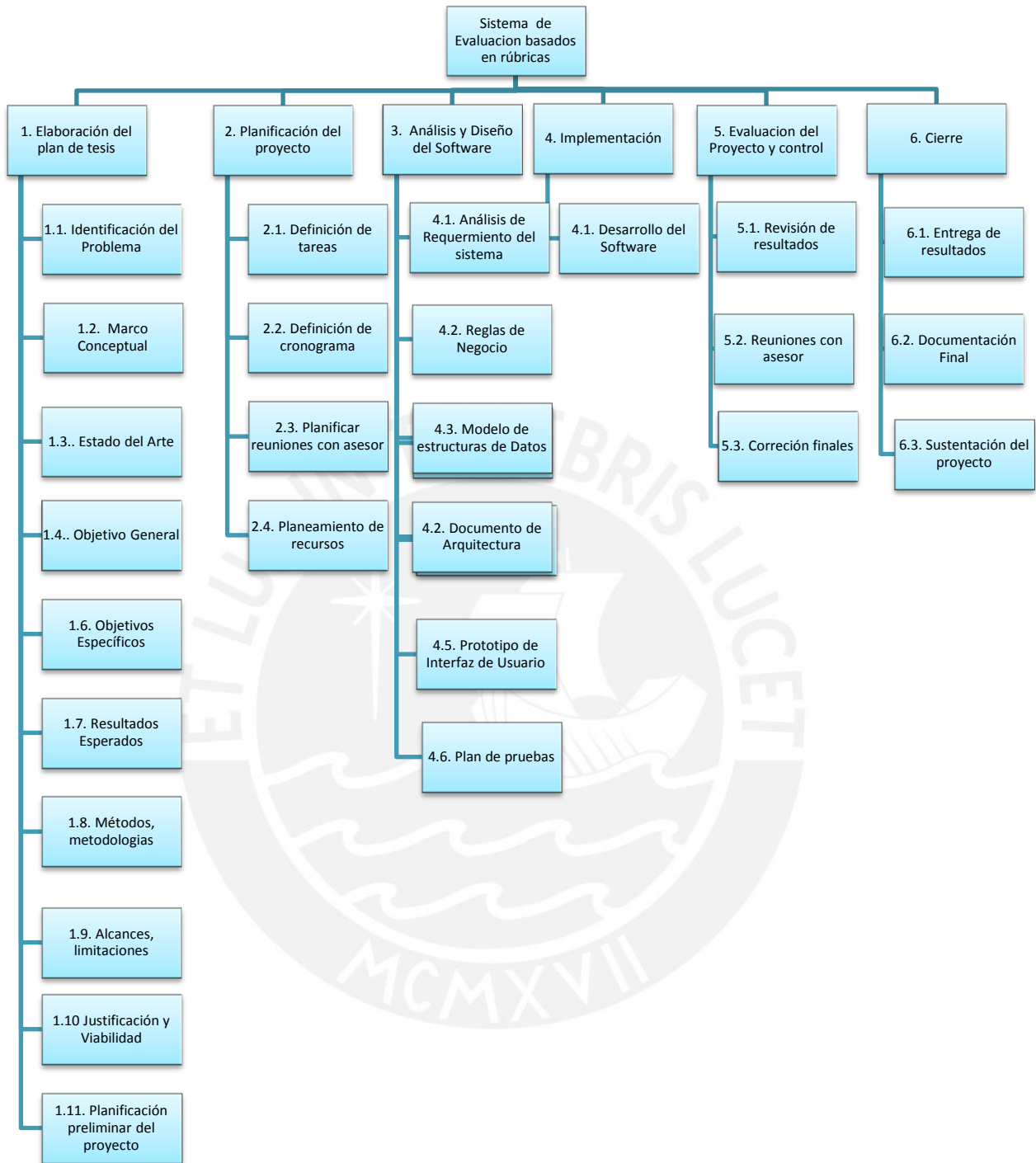


Figura 1-11 Estructura de descomposición del Trabajo (EDT)

1.5.2. Cronograma del Proyecto (Diagrama de Gantt)

Se presenta el cronograma en la Figura 1-14 y 1-15 que se siguió en el desarrollo del proyecto de Tesis.



Figura 1-12 Cronograma del Proyecto- Inicio del Proyecto

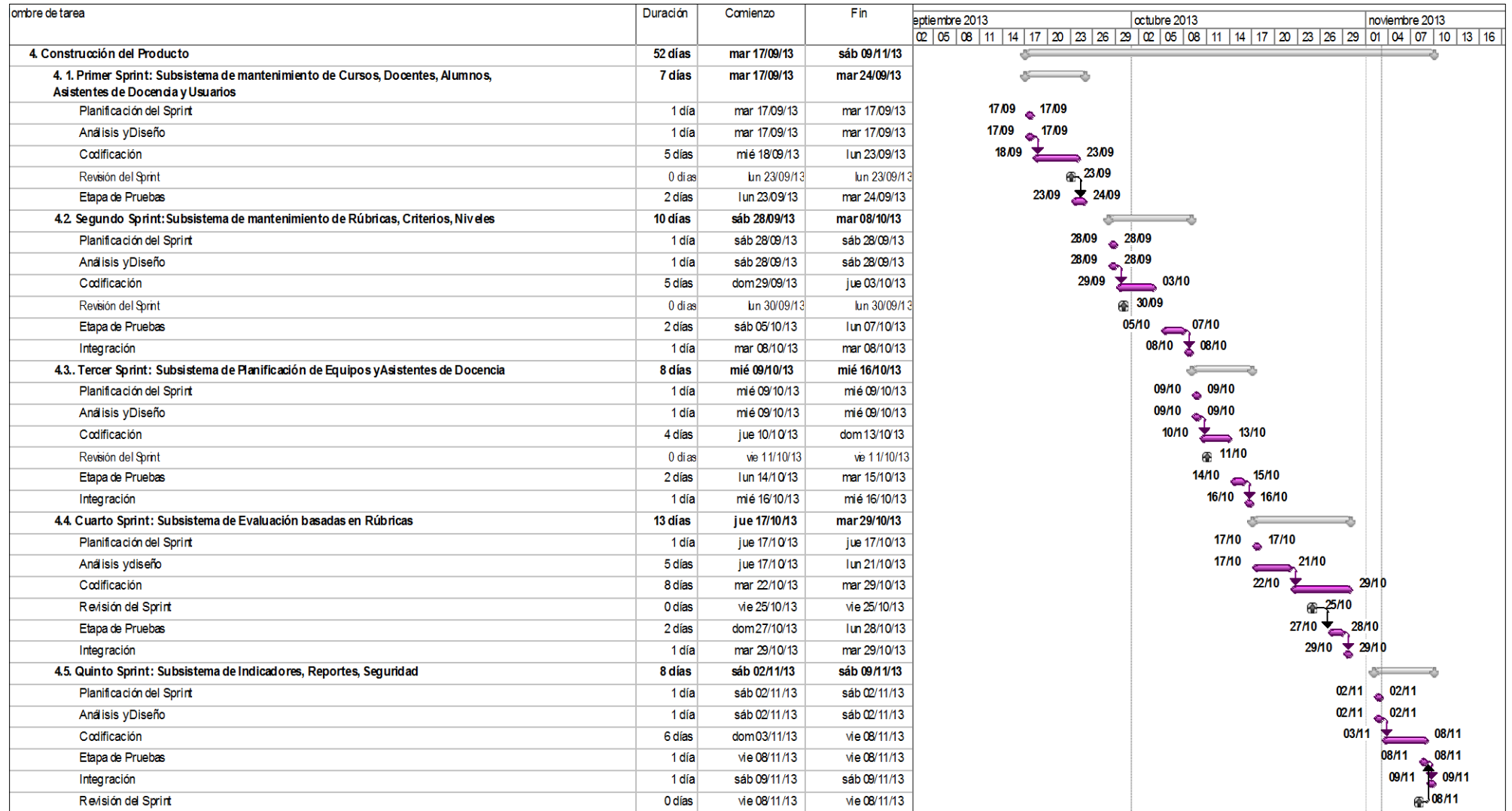


Figura 1-13 Cronograma del Proyecto –Ejecución y Control

Finalmente se presenta la etapa final del proyecto en la Figura 1-16.

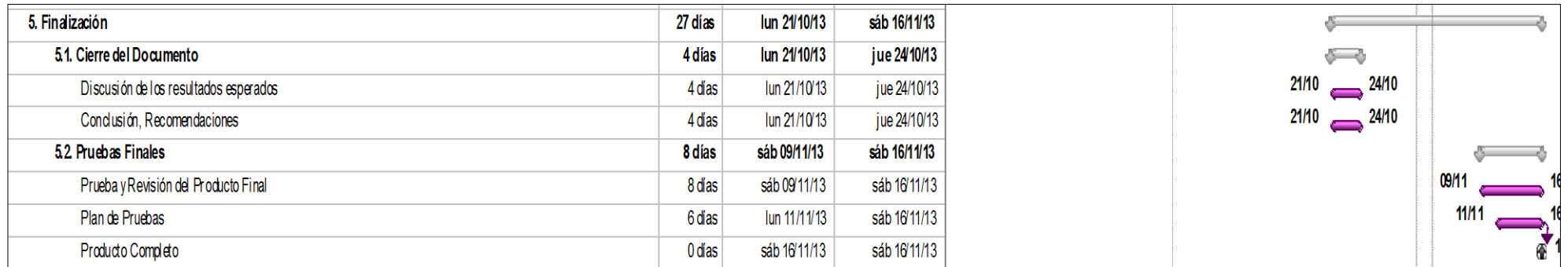


Figura 1-14 Cierre del Proyecto

CAPÍTULO 2: DEFINICIÓN DEL SISTEMA

En este capítulo se presenta la definición de la solución propuesta del presente proyecto siguiendo principios de la metodología Scrum y la utilización de algunas herramientas y métodos para la obtención de los resultados esperados del proyecto. Esta sección está dividida en tres partes: la definición de requisitos (que se observan finalmente en historias de usuario), el análisis de la solución y el diseño del sistema.

2.1. Definición de Requisitos

Para el comienzo de la solución del proyecto se realizó la búsqueda de los requisitos funcionales del sistema. Por ello, se realizaron entrevistas y reuniones con el cliente para el levantamiento de estas necesidades. En esta sección se muestra quiénes son los usuarios finales del sistema, la identificación de los requisitos, la lista de historias de usuarios y finalmente la representación de los requisitos en historias de usuario.

2.1.1. Usuarios

Los principales usuarios del sistema identificados son:

- **Administrador**
El administrador es el usuario que realiza la carga inicial de datos en el sistema del proceso de evaluación basado en rúbricas.
- **Docente**
El docente es el usuario que realiza la conformación de equipos de estudiantes, la gestión de la planificación de las evaluaciones, la asignación de los asistentes de docencia a los equipos para obtener los resultados finales de las evaluaciones.
- **Asistente de Docencia**
El asistente de docencia es el usuario que también realiza las evaluaciones basados en rúbricas de los equipos o estudiantes que le fueron asignados.

- **Estudiante**

El estudiante es el usuario evaluado por el asistente de docencia o el docente. También puede evaluarse a sí mismo o a su equipo según la planificación realizada por el docente.

2.1.2. Identificación de Requisitos

Para conocer las necesidades del sistema de gestión de evaluaciones basados en rúbricas se utilizaron técnicas de levantamiento de información que permita la identificación de las mismas de una forma rápida por el corto plazo del presente proyecto. En consecuencia, la técnica utilizada fue el diseño de prototipos del sistema. La presentación de los prototipos fue realizado en reuniones con el cliente para conocer sus comentarios al respecto y así evaluar rápidamente sus principales necesidades. Asimismo, la persona que desarrolla este proyecto realizó la observación de algunas actividades del proceso principal en el curso de Desarrollo de Programas 1 de la Especialidad de Ingeniería Informática, lo cual también facilitó la detección de requisitos del sistema.

2.1.3. Prototipos

Según los requisitos y observaciones en el proceso de evaluación basado en rúbricas del curso, se identificó lo que realiza cada usuario y las necesidades para el producto final representadas en prototipos del sistema.

A continuación se presentan algunos casos puntuales sobre el proceso de recolección basado en el diseño de prototipos

- **Creación de una rúbrica general**

La creación de una rúbrica de la especialidad en el sistema, según las necesidades del usuario y el análisis del tesista, lo realiza un usuario administrador, el cual registra todos los componentes de la rúbrica como los resultados, los aspectos, los criterios y la definición de los niveles por criterio. Los niveles registrados pueden ser variables de acuerdo a cómo lo definan las entidades acreditadoras.

En la Figura 2.1 se presenta el prototipo utilizado para levantar las necesidades del usuario para la creación de una rúbrica.

EvaRu

← → ↻

Sistema de Evaluación Basado en Rúbricas

Inicio	<h3 style="color: blue;">Registrar Rúbrica</h3> <h4>1. Información Rúbrica</h4> <p>Nombre Rúbrica: <input type="text" value="Rúbrica del programa de Ing. Informática"/></p> <p>Descripción: <input type="text" value="Rúbrica para la evaluación de resultados del programa de Ingeniería Informática de la PUCP. Cuenta con 8 resultados divididos en aspectos, criterios y niveles."/></p> <p>Tipo Nivel: <input type="text" value="Escala ABET"/></p> <h4>2. Detalle Rúbrica</h4> <div style="display: flex; align-items: flex-start;"> <div style="border: 1px solid gray; padding: 5px; width: 200px;"> <ul style="list-style-type: none"> ▼ A. R1 <ul style="list-style-type: none"> ▼ Matemáticas <ul style="list-style-type: none"> <li style="background-color: #e0e0e0;">Insuficiente Aceptable Esperado Notable ▼ B. R2 <ul style="list-style-type: none"> ▼ Diseña <ul style="list-style-type: none"> ▼ Evolución Exp <ul style="list-style-type: none"> Insuficiente Aceptable Esperado </div> <div style="margin-left: 10px;"> <p>+Resultado +Aspecto +Criterio Eliminar</p> <div style="border: 1px solid gray; padding: 10px; width: 250px;"> <p>Nivel:</p> <p>Nombre Corto: <input type="text" value="Insuficiente"/></p> <p>Descripción: <input type="text" value="Presenta problemas en lógica."/></p> <p>Valor(Opcional) <input type="text" value="14"/></p> </div> </div> </div>
Carga de Datos	
Rúbricas	
Accesos	

Figura 2-1 Prototipo para la creación de una rúbrica general

- **Creación de una rúbrica por curso**

La creación de una rúbrica por curso está basada en una rúbrica general de una especialidad. Este registro lo realiza un usuario administrador, quien creará las rúbricas por curso con los criterios que le corresponda. Se presenta el prototipo funcional para el caso explicado en la Figura 2-2.

EvaRu

← → ↻

Sistema de Evaluación Basado en Rúbricas

Inicio	<h3>Registrar Rúbrica por Curso</h3> <p>1. Elegir Rúbrica y Curso</p> <p>Rúbrica: <input type="text" value="Rúbrica de la Especialidad de Inf."/> ▼</p> <p>Curso: <input type="text" value="Desarrollo de Programas 1"/> ▼</p> <p>2. Detalle Rúbrica</p> <div style="border: 1px solid gray; padding: 5px;"> <p>▼ A. R1</p> <p><input checked="" type="checkbox"/> ▼ Matemáticas</p> <p><input checked="" type="checkbox"/> ▼ Lógica</p> <p>▼ B. R2</p> <p><input type="checkbox"/> ▼ Diseña</p> <p><input type="checkbox"/> ▼ Evolución Exp</p> </div> <div style="border: 1px solid gray; padding: 5px; margin-top: 10px;"> <p>Criterio:</p> <p>Nombre Corto: <input type="text" value="Logica"/></p> <p>Descripción: <input type="text" value="Criterio relacionado a las competencias logicas del estudiante."/></p> </div> <p style="text-align: right;"> <input type="button" value="Asignar"/> <input type="button" value="Atrás"/> </p>
Carga de Datos	
Rúbricas	
Accesos	

Figura 2-2 Prototipo para la creación de una rúbrica por curso

- **Planificación de Evaluaciones**

La planificación de evaluaciones durante el período lectivo lo realiza el usuario Docente, quien crea las evaluaciones conformados por un conjunto de criterios de la rúbrica del curso. También dentro de la planificación el usuario debe indicar a qué semana corresponde dicha evaluación y realizar la configuración de evaluados dentro de una evaluación. Se presenta el prototipo funcional para la planificación en la Figura 2-3.

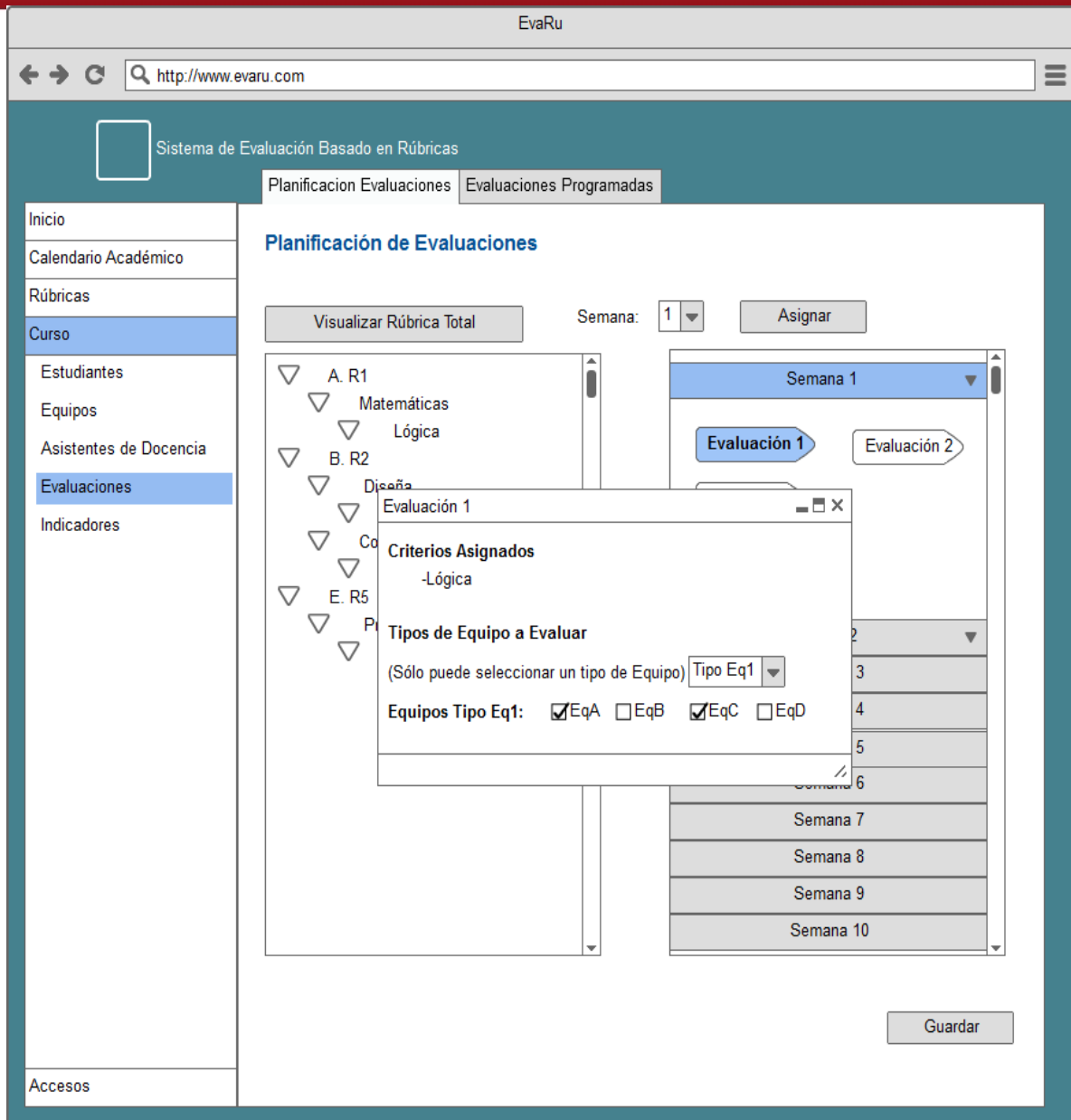


Figura 2-3 Planificación de Evaluaciones

Todos los prototipos del sistema se pueden ver en el *Anexo B Prototipos del Sistema*.

Luego de las entrevistas, observaciones y la presentación de los prototipos fue necesario establecer prioridades en los diversos requisitos del sistema identificados.

La lista de prioridades o también denominando Product Backlog, según la metodología Scrum, se presenta en la siguiente sección.

2.1.4. Lista de Historias de Usuario

Una vez finalizado la recolección de requisitos del sistema, se procede a ordenarlos según prioridades y dificultad en el desarrollo del producto.

En la Tabla 2.1 se muestra la escala de prioridad (1: prioridad baja, 2: prioridad media, 3: prioridad alta) a seguir para la clasificación de cada uno de los requisitos que son representados en historias de usuario. Asimismo, en la Tabla 2.2 se presenta la escala de dificultad (1: dificultad baja, 2: dificultad media, 3: dificultad alta)

Basado en ello, el desarrollador realizó el orden ascendente de historias de usuario de acuerdo a la dificultad de desarrollo de las mismas y a las prioridades de las funcionalidades considerando las necesidades del usuario

Prioridad	1	2	3
	Baja	Media	Alta

Tabla 2-1 Prioridad en historias de usuario

Dificultad	1	2	3
	Baja	Media	Alta

Tabla 2-2 Dificultad en desarrollo de las historias de usuario

- **Pila del Producto**

Para determinar la pila de producto o las prioridades de las historias de usuario se entregó al cliente la lista de historia de usuarios recolectados sin orden alguno para que pueda establecer las prioridades, según la escala presentada en la tabla 2.1.

Luego el desarrollador del producto realizó el orden según la Tabla 2.2 teniendo en cuenta el primer orden realizado por el cliente. Finalmente de ambos resultados se obtuvo la pila de producto o *Product Backlog* que se presenta a continuación en la Tabla 2.3.

Orden	Nombre de Historia de Usuario	N° Sprint
1	Iniciar sesión	1
2	Administrar calendario académico	1
3	Administrar cursos, horarios y docentes	1
4	Administrar asistentes de docencia	1
5	Administrar estudiantes	1
6	Visualizar calendario académico	1
7	Visualizar cursos	1
8	Visualizar estudiantes por curso	1
9	Administrar Rúbricas General	2
10	Administrar Rúbricas por Curso	2
11	Administrar Tipo de Niveles	2

Orden	Nombre de Historia de Usuario	N° Sprint
12	Visualizar Rúbrica General	2
13	Visualizar Rúbrica por Curso	2
14	Generar niveles por criterio	2
15	Generar estructura de Rúbrica General	2
16	Generar estructura de Rúbrica por curso	2
17	Configurar Tipo de Equipos por Curso	3
18	Asignar estudiantes a equipos y tipos de equipos	3
19	Asignar asistente de docencia a equipos	3
20	Planificar Evaluaciones	3
21	Asignar evaluaciones a tipos de equipos	3
22	Visualizar Equipos	3
23	Asignar criterios a evaluaciones	3
24	Generar aviso de evaluaciones programadas	4
25	Realizar evaluación	4
26	Visualizar evaluaciones programadas por semana	4
27	Visualizar resultados de evaluación	4
28	Buscar histórico de evaluaciones	4
29	Generar indicadores de criterios cumplidos por estudiante	5
30	Generar indicadores de curva de notas por estudiante	5
31	Generar reporte de evaluaciones realizadas	5
32	Generar reporte final de criterios por estudiante	5
33	Generar permisos de acceso por perfil de usuario	5
34	Enviar accesos a usuarios	5
35	Generar seguridad al producto	5

Tabla 2-3 Lista de historias de usuario

2.1.5. Historias de Usuario

Las historias de usuario cuentan con la siguiente estructura:

- **Cabecera:** Título de las historia de usuario. Todas tienen de título: *Historia de Usuario*
- **Número:** Número de la historia de usuario
- **Nombre:** Historia: Nombre de la historia de Usuario
- **Usuario:** persona que realiza la funcionalidad descrita en la historia de usuario.
- **Prioridad en el negocio:** Baja, Media, Alta.
- **Puntos Estimados:** Indica la dificultad expresado en números: 1: Baja, 2: Media, 3: Alta

- **Iteración asignada:** Número de Sprint en que se desarrolló la historia de usuario
- **Descripción:** Descripción de la historia de usuario.
- **Observaciones:** Observaciones del desarrollador para la historia de usuario.

En la Tabla 2.4 y 2.5 se presentan dos historias de usuario para el desarrollo del sistema: “Administrar Rúbricas y “Asignar asistente de docencia a equipos”.

HISTORIA DE USUARIO	
Número: 9	Usuario: Administrador
Nombre historia: Administrar rúbrica general	
Prioridad: Alta	
Dificultad: Media	
Descripción: Se debe registrar, modificar o eliminar la rúbrica que está compuesta por todos los resultados de la especialidad. Los campos de registro son nombre de la rúbrica general, descripción de la rúbrica, y estructura de la rúbrica.	
Observaciones: Un resultado está compuesto por un conjunto de aspectos. Un aspecto a su vez está compuesto por criterios, los cuales están bajo una escala de valoración o niveles asignados.	

Tabla 2-4 Administrar Rúbrica General

HISTORIA DE USUARIO	
Número: 17	Usuario: Docente
Nombre historia: Configurar Tipo de Equipos por Curso	
Prioridad: Alta	
Dificultad: Media	
Descripción: Se debe configurar previamente la cantidad de tipos de equipos y sus evaluadores (docente, asistente de docencia, autoevaluación) en el curso.	
Observaciones: Se debe considerar que para cada tipo de equipo, los evaluadores pueden variar.	

Tabla 2-5 Configurar Tipo de Equipos por Curso

Todas las historias de usuario se pueden ver en el *Anexo A Historias de usuario*.

2.2. Análisis de la Solución

Luego de la obtención de requisitos del sistema, se procedió al análisis de la solución. Para ello, fue necesaria la definición de reglas de negocio que se siguen en el proceso de evaluación basados en rúbricas en cursos de proyectos universitarios. En este caso, se centró el análisis en la especialidad de Ingeniería Informática de la PUCP.

2.2.1. Reglas de Negocio

Una regla de negocio define o limita aspectos del algún negocio en particular y establece una estructura que condiciona el comportamiento de los actores en este negocio (Business Rules Group, 1993). La identificación de estas reglas, para este proyecto, se realizó a través de entrevistas con un usuario docente y con la observación de algunas actividades en este proceso.

- **Especificación de Reglas de Negocio**

Para realizar la especificación de reglas se utiliza la clasificación por restricción y derivación (Odell, 1998). Esta clasificación es sencilla pero completa y es expresada en lenguaje natural. Para el desarrollo del sistema se consideró un conjunto de estas reglas de negocio para la implementación.

Para ordenar el conjunto de reglas de negocio definidas para el proyecto, se realizó una clasificación de la siguiente manera:

- **Reglas de Negocio asociadas a los docentes:** todas las reglas de lo que realiza y no realiza el docente en este proceso.

Por ejemplo:

“El docente es el único que realiza la asignación de los asistentes de docencia a los equipos.”

- **Reglas de Negocio asociadas a los estudiantes:** todas las reglas de lo que realiza y no realiza el estudiante en este proceso.

Por ejemplo:

“El estudiante no puede visualizar los resultados de lo evaluado por sus compañeros.”

- **Reglas de Negocio asociadas a los asistentes de docencia:** todas las reglas de lo que realiza y no realiza el asistente de docencia en este proceso.

Por Ejemplo:

“El asistente de docencia no puede ser evaluado.”

- **Reglas de Negocio asociadas a las evaluaciones:** todas las reglas que corresponden a las evaluaciones en este proceso.

Por Ejemplo:

“Una evaluación está asociada a una semana del calendario académico del periodo lectivo.”

- **Reglas de Negocio asociadas a las rúbricas:** todas las reglas que corresponden a las rúbricas en este proceso.

Por Ejemplo:

“Cada curso maneja una rúbrica en particular.”

Se muestra en las Tablas 2.6 y 2.7 las reglas para las rúbricas y los estudiantes a continuación.

Nombre	Clasificación	Descripción	Estática o Dinámica
RN01	Operación	Las rúbricas están asociadas a escalas de calificación	Estática
RN02	Operación	Las rúbricas tienen escalas de 4 niveles (acreditación actual)	Dinámica
RN03	Operación	Las rúbricas tienen escalas de 7 niveles lo que se conoce como evaluación de desempeño	Dinámica
RN04	Restricción	Cada curso maneja una rúbrica en particular.	Estática
RN05	Operación	La rúbrica en la especialidad de Ingeniería Informática se divide en resultados del programa donde cada uno contiene aspectos, criterios y niveles de valoración.	Estática
RN06	Operación	Un mismo conjunto de criterios de la rúbrica pueden ser utilizados con distintas escalas de valoración o niveles.	Dinámica

Tabla 2-6 Reglas para las rúbricas

Nombre	Clasificación	Descripción	Estática o Dinámica
RN01	Restricción	El estudiante es el único que puede ser evaluado.	Estática
RN02	Operación	El estudiante puede evaluarse a sí mismo y a sus compañeros.	Estática
RN03	Restricción	El estudiante no puede visualizar los resultados de lo evaluado por sus compañeros.	Estática
RN04	Operación	El estudiante es evaluado según cronograma establecido en el curso-horario durante el periodo lectivo	Dinámica
RN05	Operación	El estudiante recibe los resultados de la evaluación.	Estática
RN06	Restricción	El estudiante no puede ser evaluado por más de un asistente de docencia por un mismo criterio en un período definido.	Estática

Tabla 2-7 Reglas de Negocio para los estudiantes

Todos los detalles de las reglas de negocio de este proceso del sistema se pueden ver en el Anexo C “*Informe Técnico de Reglas de Negocio*”.

2.3. Diseño del Sistema

Luego de las reglas de negocio definidas en este proceso de evaluaciones basadas en rúbricas, en esta sección se presentan la arquitectura de información y el diseño de la arquitectura planteada para este proyecto.

2.3.1. Estructuras de Datos

Para la consolidación de los datos que procesa el sistema fue necesario definir una estructura adecuada. El concepto de estructura de datos se explica como la relación física o lógica entre elementos de datos diseñados para soportar las funciones de manipulación de datos (ISO/IEC 24765, 2011).

Para este proyecto se consideraron las siguientes dos estructuras:

- **Diseño Lógico**

Comprende las entidades que proporcionan la lógica para el manejo de la solución. Entre ellas se encuentran:

Clase	Detalle
Usuario	Clase que agrupa los datos principales de un usuario en el sistema.
Perfil	Clase que agrupa los datos del perfil de un usuario. El perfil puede ser docente, asistente de docencia o estudiante.
Nivel	Clase que agrupa los datos de los niveles de una rúbrica.
Criterio	Clase que agrupa los datos de los criterios de una rúbrica.
Aspecto	Clase que agrupa los datos de los aspectos de una rúbrica
Rúbrica	Clase que agrupa los datos de los criterios, niveles, aspectos y resultados del programa.
Resultado	Clase que agrupa los datos de los resultados del programa de una rúbrica
Evaluación	Clase que agrupa los datos de una evaluación basado en criterios de una rúbrica.
Salon	Clase que agrupa los datos de un horario, ciclo y curso.
Equipo	Clase que agrupa los datos de un horario, curso y ciclo con un grupo de estudiantes.
Registro Notas	Clase que agrupa los resultados de una evaluación de un determinado curso, ciclo y horario.
Usuario xSalon	Clase que agrupa la relación entre un usuario y un determinado curso, ciclo y horario.

Tabla 2-8 Principales entidades lógicas

- **Modelo de Base de Datos**

Para el modelo de base de datos del proyecto se consideran todas las entidades necesarias para el almacenamiento de los datos del sistema; para su representación se utilizó un diagrama entidad-relación que permite observar la relación entre todas ellas.

Por ejemplo, se presenta en la Figura 2.2 la relación entre las entidades Rúbrica, Resultado, Aspecto, Criterio, CriterioxNivel, Nivel y TipoNivel.

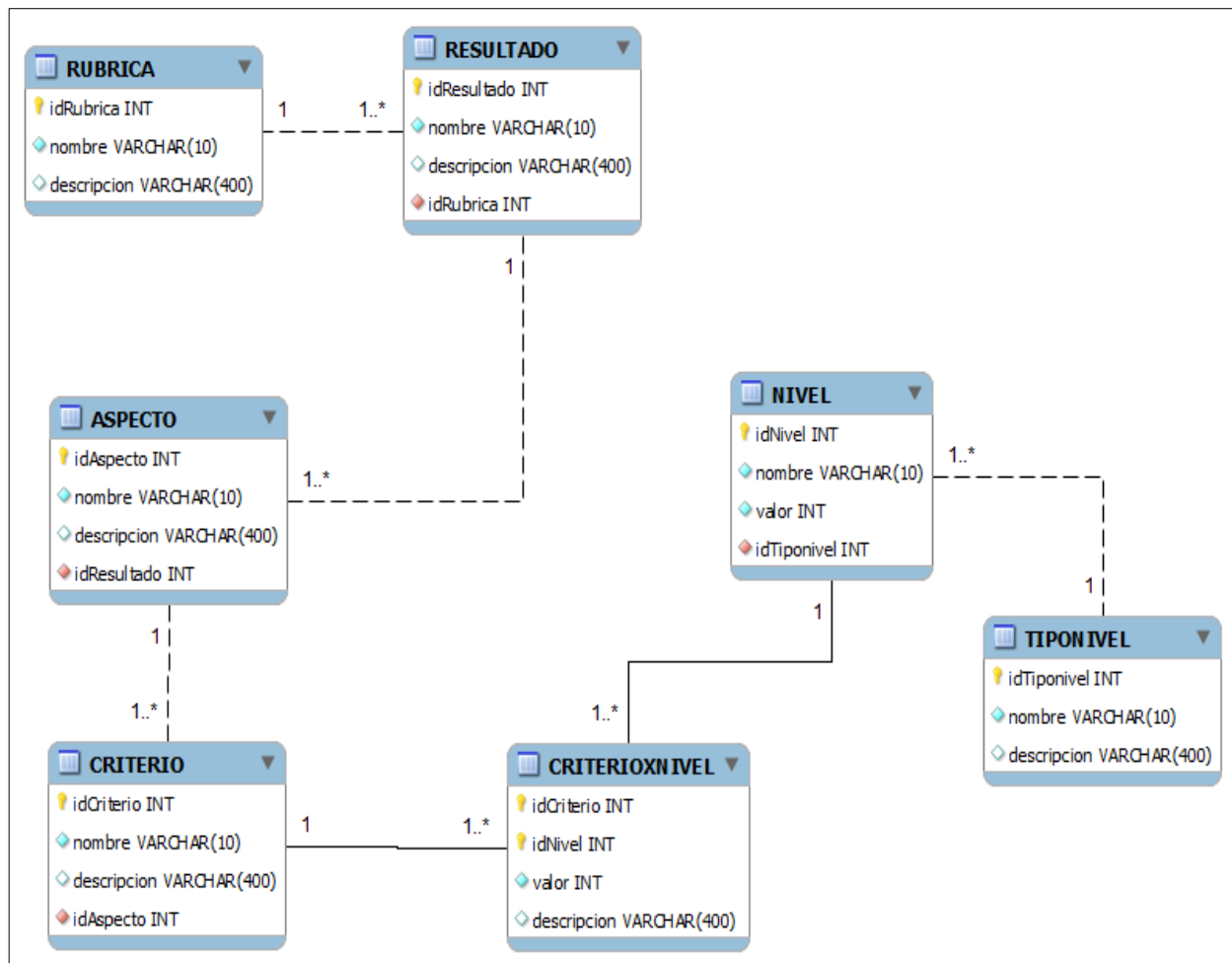


Figura 2-4 Diagrama Entidad-Relación para la estructura de datos de las rúbricas.

Para el diseño de la estructura de datos del sistema se consideró un conjunto de las reglas de negocio, los cuales permitieron establecer la relación entre todas ellas.

Todos los detalles de las entidades de la estructuras de datos del sistema se pueden ver en el Anexo D “Informe Técnico de Estructuras de Datos”.

2.3.2. Arquitectura de la solución

En esta sección, se explica la arquitectura de la solución donde se muestra una vista de la arquitectura básica del sistema y se detallan las principales interacciones de los componentes y tecnologías que se utilizaron en el diseño del presente proyecto.

- **Representación de la arquitectura**

El patrón de diseño de arquitectura de software utilizado para este sistema fue Modelo-Vista-Controlador (MVC), el cual permite la separación de la vista (interfaz del usuario), la lógica del negocio y los datos (Buschmann, 1996). Estas tres capas ofrecen una estructura adecuada para un sistema web (producto final que se desarrolló como solución para este proyecto).

Asimismo, fue necesario hacer uso de algunas tecnologías y marcos de trabajo para el diseño de esta arquitectura de software. Entre algunas de las tecnologías y marcos de trabajo importantes que se utilizaron se encuentran:

- **CodeIgniter (CI)**

En el diseño de la arquitectura del sistema se utilizó CodeIgniter como marco de trabajo.

- **Front End y Back End**

Se denomina Front End a la parte del sistema de software que interactúa con el usuario. Por otro lado, se denomina Back End a la parte del sistema de software que comprende los componentes que procesan la salida del Front End como el caso de acceso a la base de datos.

Las capas del patrón MVC para la solución del proyecto cuentan con componentes en cada una de ellas como se describe a continuación:

- **Vista**

En esta capa se realiza el envío de las peticiones del usuario (Cliente web) al Controlador.

Por Ejemplo:

Se ubican los siguientes elementos:

- Plantillas de las interfaces del sistema: *carga_datos.php*, *crear_rubrica.php*, entre otros.
- Validación de datos: “*Debe completar los campos*”

- Visualización de mensajes: “Se registraron los cursos correctamente”

➤ **Controlador**

En esta capa se recibe las solicitudes de servicios del usuario. Cuando el controlador recibe estos requerimientos por parte del cliente procede a invocar a las interfaces de la capa Modelo. Cuando la capa Modelo envía lo solicitado a la capa Controlador. Este procesa y lo envía a la capa Vista para que pueda entregar al cliente web lo requerido.

Por Ejemplo:

Se ubican los siguientes elementos:

- Pase de variables por Post: clase PostController
- Acción del controlador: show

➤ **Modelo**

Esta capa proporciona la lógica para la gestión de datos y transacciones de los mismos. Recibe la solicitud del usuario por parte del Controlador y lo solicita a la estructura de información de base de datos. Posteriormente la base de datos envía lo solicitado a la capa Modelo y luego lo envía a la capa Controlador.

Por Ejemplo:

Se ubican los siguientes elementos:

- Entidades necesarias para el sistema: usuario, rubricas, evaluación, entre otras
- Conexión a la Base de Datos: Clases: mapper nivel, mapper rúbricas, entre otras

Una vez presentado los componentes de las capas y los marcos de trabajo que se utilizaron en la Figura 2.3 se presentan el uso del patrón MVC en la solución del proyecto para la “Creación de Rúbricas”.

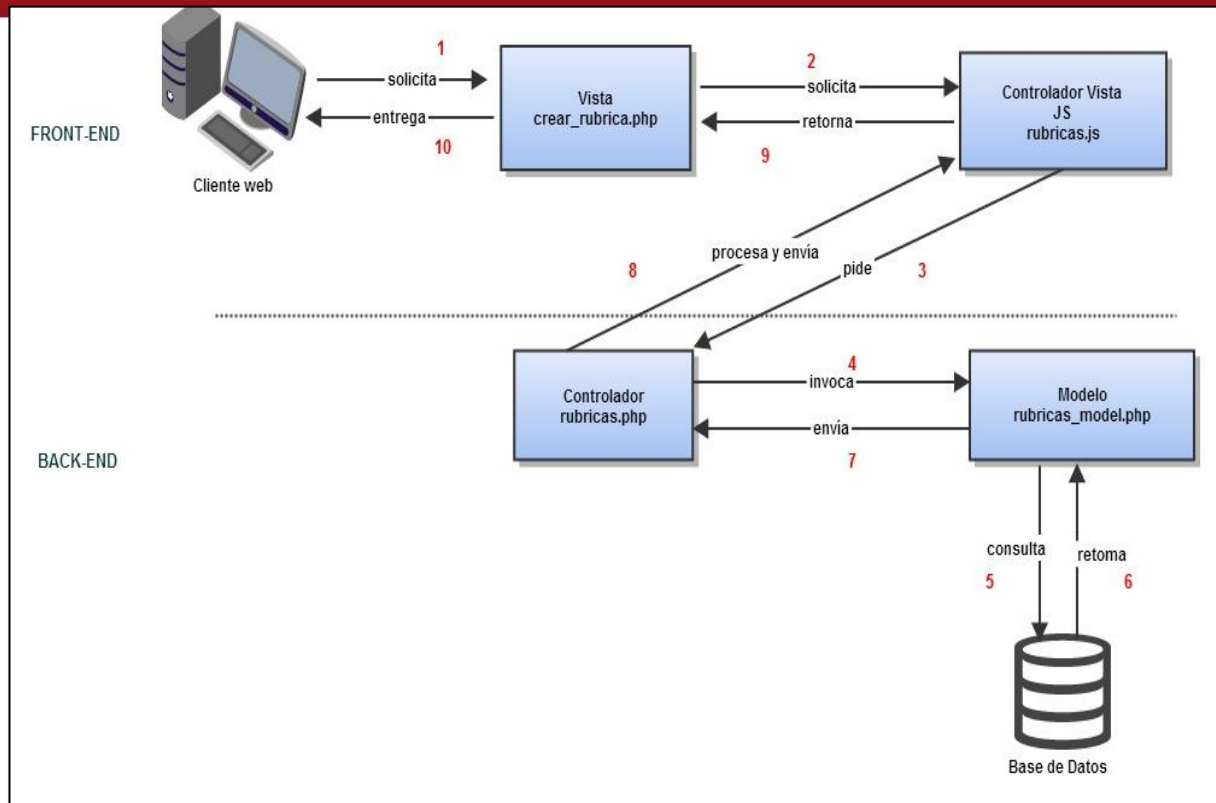


Figura 2-5 Uso del patrón MVC aplicado a la creación de rúbricas

El proceso del patrón MVC comienza en el **Front End** cuando el cliente web solicita alguna acción por ejemplo: **Crear Rúbrica** (botón ubicado en la vista **crear_rubrica.php**). Luego esta solicitud se envía hacia el controlador de la vista **rubricas.js** que reconocerá el evento del botón y solicitará al controlador del **Back End** el servicio solicitado de creación y esperará respuesta.

En el **Back End** el controlador **rubricas.php** invocará al modelo **rúbricas_model.php** para que realice el acceso a la base de datos (en este caso inserción de una nueva rúbrica) con la finalidad de que retorne con la petición solicitada al controlador (información registrada exitosamente) y así mismo el controlador retorne la respuesta pendiente al controlador de la vista **rubricas.js** donde finalmente envía el servicio solicitado a la vista inicial **crear_rubrica.php**.

Todos los detalles de las arquitecturas de software del sistema se pueden ver en el Anexo E "*Informe Técnico de Arquitectura de Software*".

CAPÍTULO 3: CONSTRUCCIÓN Y PRUEBAS

El presente capítulo se divide en dos secciones siguiendo los principios para la implementación del software de la metodología Extreme Programming (XP). Por un lado, se empieza con el detalle de las herramientas utilizadas para la implementación de la solución y las buenas prácticas consideradas en el desarrollo. Por otro lado, se presentan las pruebas realizadas a la solución para garantizar el funcionamiento correcto de cada una de las funcionalidades definidas en la pila del sprint.

3.1. Construcción

En esta sección se presentan las herramientas técnicas que se consideraron en el proceso de implementación del sistema. Asimismo, se explican los detalles de los estándares de programación utilizados.

3.1.1. Herramientas de desarrollo

En el proceso de implementación del sistema se utilizaron herramientas que pertenecen al grupo de software libre. Se describe cada una de estas tecnologías a continuación:

- **NetBeans IDE 7.3.1.** es una plataforma de desarrollo que proporciona edición y depuración de código. En el desarrollo del sistema se utilizó la última versión 7.3.1.
- **Servidor Web: XAMPP:** es un servidor libre que agrupa a la base de datos MySQL, el servidor web Apache y los lenguajes PHP (XAMPP, 2013). La elección de este servidor para el desarrollo del proyecto fue porque se realizó uso del lenguaje de programación PHP y la base de datos MySQL, tal como se indica en la sección de Arquitectura de Software.
- **Chrome Developer Tools:** Herramienta de depuración de sistemas web proporcionada por el navegador Google Chrome para desarrolladores (Google Developers, 2013). En el caso del sistema desarrollado facilitó la depuración de devolución de mensajes desde el controlador a la vista.

Por Ejemplo:

Visualización de mensajes como: *“Se registró con éxito”* o *“Criterio falta ser evaluado”*.

- **Bootstrap:** Marco de Trabajo de diseño web y móvil que proporciona estándares para el diseño del sistema (Bootstrap, 2013). Fue utilizado en el desarrollo porque facilita la creación de las pantallas del sistema y su diseño se adapta a cualquier pantalla de cualquier dispositivo como laptops o móviles.

3.1.2. Implementación de Componentes

A continuación se presentan los detalles de la implementación del sistema.

- **Estructura Interna de la Solución**

La implementación de la solución está basada principalmente en los lenguajes de programación PHP y HTML. Por ello, para el desarrollo del producto se realizó la búsqueda de un framework que permita separar todos los componentes que utilizó el proyecto.

CodeIgniter, framework elegido para el desarrollo del sistema, presenta una estructuración interna basado en el modelo MVC (patrón arquitectónico del proyecto) como se visualiza en la Figura 3-1.

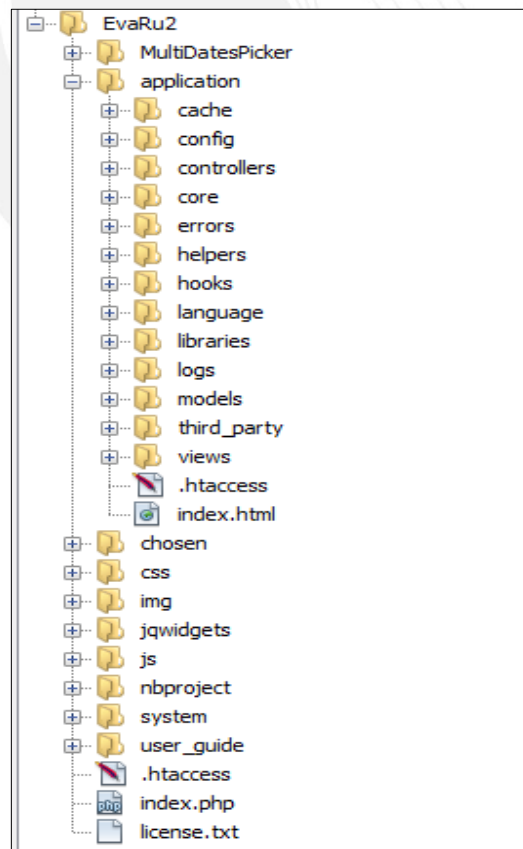


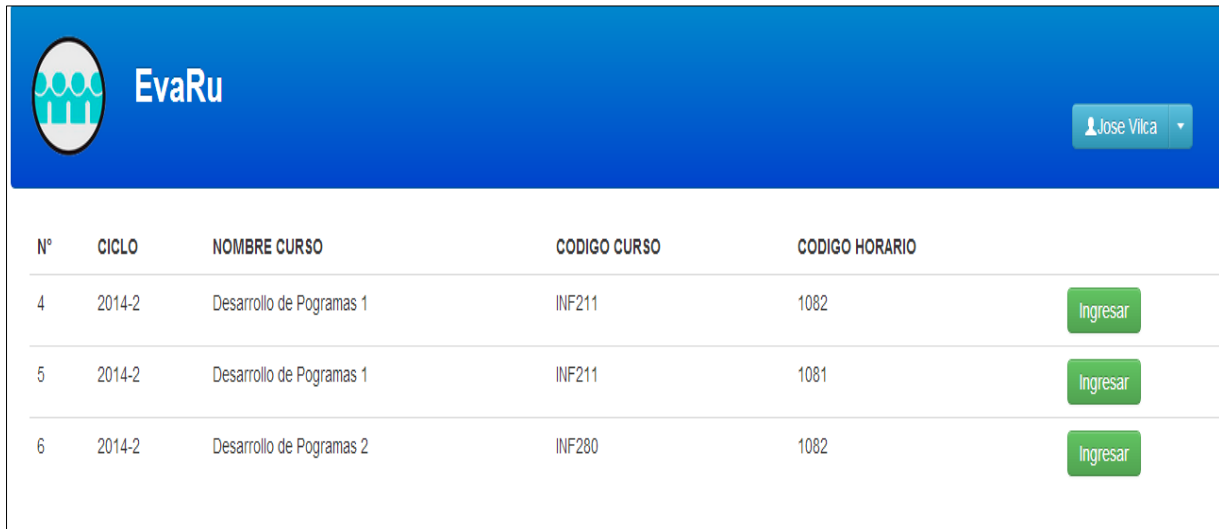
Figura 3-1 Estructura Interna de la Solución

- **Interfaz Gráfica**

Se presenta las principales interfaces del producto final del sistema y los detalles particulares de su implementación.

- **Interfaz Pantalla Inicial de Acceso al Sistema**

La interfaz que se presenta en la Figura 3-2 es la pantalla principal del sistema para los usuarios docentes, asistentes de docencia y estudiantes. Para ingresar a las demás funcionalidades es necesaria la elección de un curso presentado en esta interfaz.



N°	CICLO	NOMBRE CURSO	CODIGO CURSO	CODIGO HORARIO	
4	2014-2	Desarrollo de Pogramas 1	INF211	1082	Ingresar
5	2014-2	Desarrollo de Pogramas 1	INF211	1081	Ingresar
6	2014-2	Desarrollo de Pogramas 2	INF280	1082	Ingresar

Figura 3-2 Pantalla inicial de acceso al sistema

Luego de la selección de un curso en esta pantalla (a través del botón Ingresar) se aprecia los datos y funcionalidades propias para el usuario que se autenticó en el sistema.

Para la diferencia de los datos y accesos particulares por usuario se utilizó variables del tipo **Session**, propias del lenguaje de programación PHP. Una de las características de este tipo de variables es que el valor almacenado no se pierde hasta cuando el usuario cierre la sesión del sistema o lo deje de utilizar por más de 20 minutos (dato parametrizable). Para el caso del proyecto, este tipo de variable fue importante utilizarla para el almacenamiento de los datos del horario, curso y ciclo ya que con ello fue posible la diferencia de las diferentes funcionalidades por curso y usuario.

En la Figura 3-3 se observa el manejo de la variable Session para almacenar los valores mencionados.

```

function ingresar_curso() {
    $this->_esta_logueado();
    $data = array(
        'ciclo' => $this->input->post('ciclo'),
        'curso' => $this->input->post('curso'),
        'horario' => $this->input->post('horario'),
        'nbrcurso' => $this->input->post('nbrcurso')
    );

    $this->session->set_userdata($data);

    echo "ok";
}

```

Figura 3-3 Uso de la variable Session de PHP en el proyecto

Asimismo en la Figura 4-4 se observa el manejo de las variables **Session** almacenado para la implementación de una de las funcionalidades propias del curso como la búsqueda de estudiantes que le corresponde.

```

function buscar_estudiantes() {

    $query = $this->db->query("SELECT
        CONCAT_WS(' ',U.nombres,U.apeMaterno,U.apePaterno) AS nombres,
        U.codigo AS codigo,
        U.email AS email,
        U.idUsuario
    FROM usuario U, usuariosalon US
    WHERE US.idUsuario = U.idUsuario
    AND US.idCurso = " . $this->session->userdata('curso') . "
    AND US.idHorario = " . $this->session->userdata('horario') . "
    AND US.idCiclo = " . $this->session->userdata('ciclo') . "
    AND U.idPerfil=4");

    if ($query->num_rows() > 0) {
        return $query->result_array();
    } else {
        return false;
    }
}

```

Figura 3-4 Uso de las variables Session para buscar estudiantes

➤ **Interfaz de Registro de Rúbrica Inicial**

El usuario administrador en el sistema es el encargado del registro de rúbricas. En un inicio una rúbrica general es registrada en el sistema, la cual contiene todos los resultados del programa. Posteriormente se tomará esta rúbrica general para la configuración de las rúbricas por curso.

Se aprecia en la Figura 3-5 la interfaz del registro de la rúbrica general.

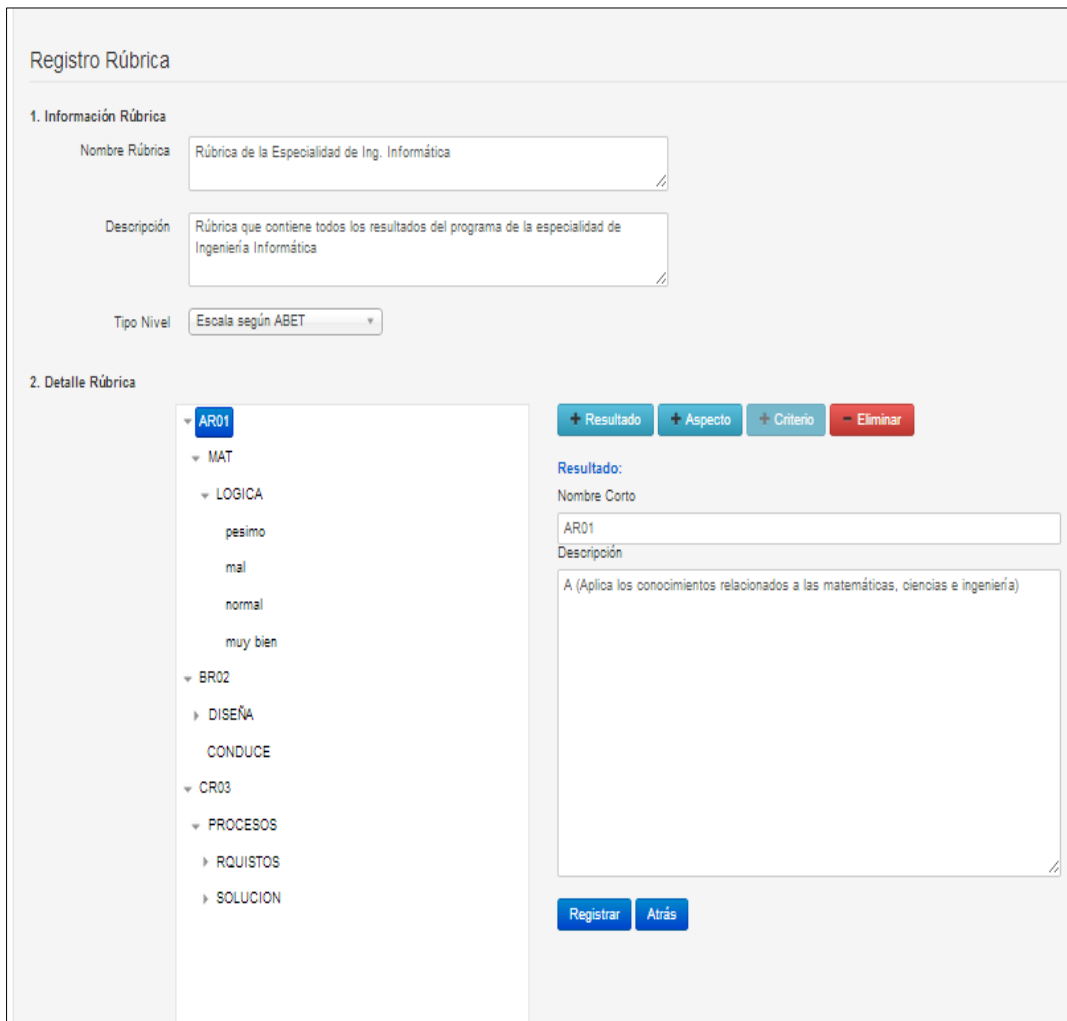


Figura 3-5 Registro Rúbrica General

Se puede observar en la pantalla que el registro de la rúbrica se divide en dos secciones:

1. **Información de la Rúbrica:** nombre y descripción de la rúbrica, y el tipo de nivel o escala de valoración a utilizar.

2. **Detalle de la Rúbrica:** registro de cada uno de los componentes de la rúbrica: resultados, aspectos, criterios y niveles.

Para el caso del registro de niveles éste puede ser variable ya que se puede contar con diferentes escalas de evaluación para los mismos criterios. Por tal motivo, se implementó dinámicamente la actualización y cambio de los niveles.

En la siguiente Figura 3-6 se puede observar que cada vez que se agrega un criterio realiza la búsqueda dinámica de niveles de acuerdo a lo especificado por el usuario.

```

$('#AddCriterio').click(function () {
    if ($('#btnCriterio').val() === "1") {
        var selectedItem = $('#jqxTree').jqxTree('selectedItem');
        var pass = password(8);
        if (selectedItem !== null) {
            $('#jqxTree').jqxTree('addTo', { id:pass+"_id", label: 'Criterio' }, selectedItem.element, false,
                "id="+pass+" onclick=activarFormulario("+pass+")");
            $('#jqxTree').jqxTree('expandItem', selectedItem.element);
            $('#jqxTree').jqxTree('selectItem', $("#"+pass+"_id")[0]);
            $('#jqxTree').jqxTree('ensureVisible', selectedItem.element);
            $('#jqxTree').jqxTree('render');
            var nivel = $("#"+pass+"_id").parents('ul').length;

            elementosArray.push(pass);

            if (nivel === 3 ) {
                var idTiponivel = $("#tiponiveles").val();

                var datos = {
                    idTiponivel: idTiponivel
                };

                $.ajax({
                    url: "../rubricas/buscar_niveles",
                    type: "POST",
                    data: datos,
                    dataType: 'json',
                    success: function(niveles) {
                        $.each(niveles, function(i, item) {
                            agregarNivel(item.idNivel, item.nombre, item.valor);
                        });
                        $("#"+pass+"_id").click();
                    }
                });
            }
        }
    }
});

```

Figura 3-6 Búsqueda de Niveles

➤ **Interfaz de Registro de Rúbrica por Curso**

Una rúbrica por curso está conformada por un conjunto de criterios del programa en particular.

Se presenta en la Figura 3-7 el registro de la rúbrica por curso.

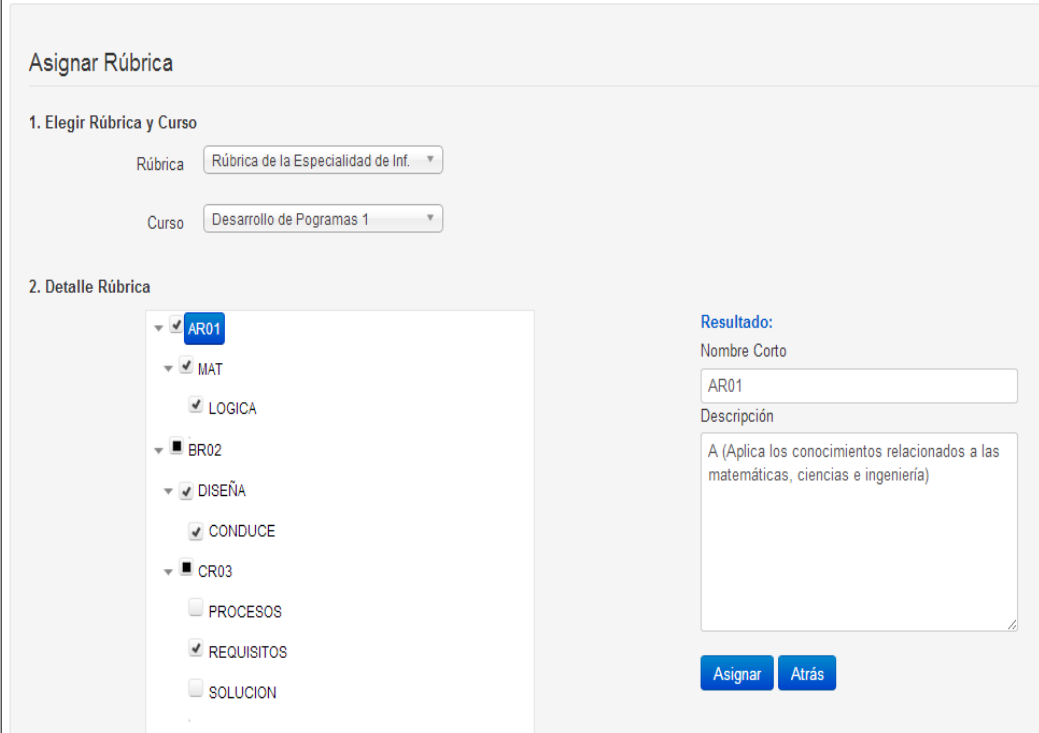


Figura 3-7 Registro de Rúbrica por Curso

Esta interfaz se divide en dos secciones:

1. Elegir Rúbrica y Curso: se elige una rúbrica general registrada con todos los resultados del programa debido a que varios cursos pueden tener resultados en común y con esta funcionalidad evitar el doble esfuerzo de registro de la rúbrica.

1. Detalle de la Rúbrica: se aprecia la rúbrica general anteriormente registrada y con unas opciones para la selección de criterios que serán evaluados para el curso elegido.

➤ Interfaz de Planificación de Evaluaciones

La planificación de las evaluaciones se realiza por curso. Una evaluación está compuesta por un conjunto de criterios de la rúbrica del curso, lo cual se asigna a una determinada semana según lo planificado por el docente.

Se presenta en la Figura 3-8 la interfaz de la planificación de evaluaciones, donde se observa al lado izquierdo, la rúbrica del curso y a la derecha las semanas del período lectivo. Para realizar la selección de los criterios, se selecciona de la estructura árbol de la rúbrica los criterios y luego la semana que se desea asignar esa evaluación. Finalmente se observa la creación de la nueva evaluación.

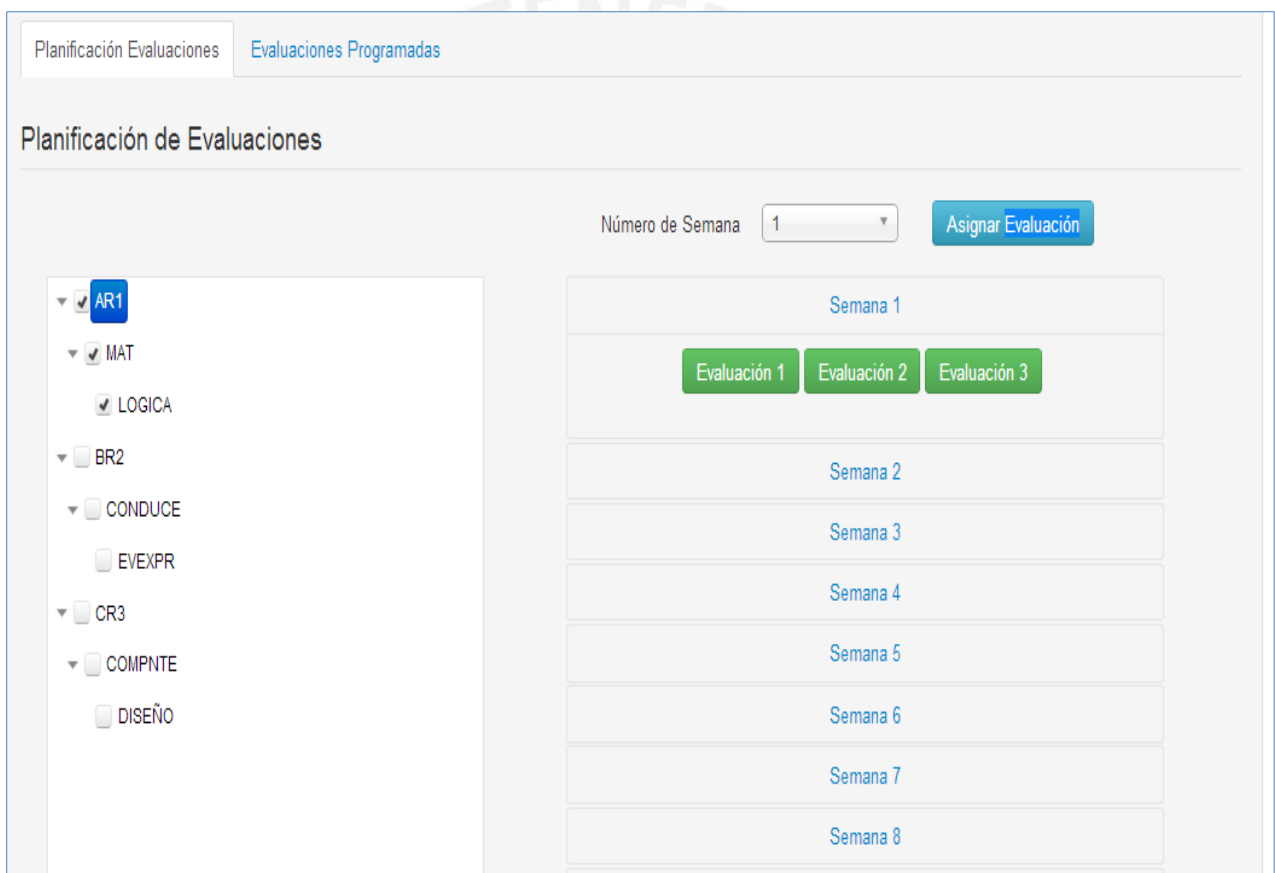
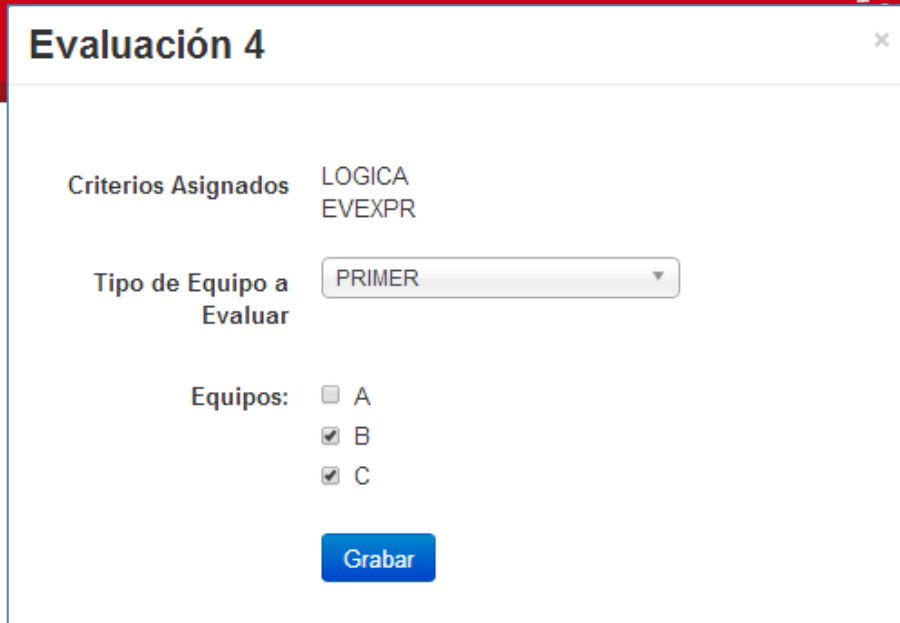


Figura 3-8 Planificación de Evaluaciones

En la Figura 3-9 se observa que, luego de asignar los criterios, se debe acceder a cada evaluación para asignar los evaluados y evaluadores de esa evaluación.



Evaluación 4

Criterios Asignados LOGICA
EVEXPR

Tipo de Equipo a Evaluar PRIMER

Equipos: A
 B
 C

Grabar

Figura 3-9 Asignación de equipos evaluados

3.1.3. Buenas prácticas para el desarrollo

En el desarrollo del sistema web fue necesario establecer estándares de programación que permita el mantenimiento y depuración del código fuente en futuras actualizaciones o mejoras. Asimismo, la definición de estándares de programación permite la reducción de código complejo y difícil de entender.

Todos los detalles de los estándares de programación se pueden ver en el Anexo G “Estándares de Programación”.

3.2. Pruebas

En esta sección se explica las pruebas realizadas para las funcionalidades del sistema. De acuerdo a la metodología XP, las pruebas de aceptación o funcionales se dedican a evaluar si al final del Sprint se consiguió la funcionalidad requerida por el cliente final del sistema (XP, 2013).

A continuación se presentan las pruebas realizadas en el sistema del presente proyecto.

- **Pruebas de aceptación o funcionales por Sprint**

Se listan las historias de usuario por Sprint consideradas para las pruebas de aceptación del sistema:

➤ **Sprint 1**

- Iniciar sesión
- Administrar calendario académico
- Administrar cursos, horarios y docentes
- Administrar asistentes de docencia
- Administrar estudiantes
- Visualizar calendario académico
- Visualizar cursos
- Visualizar estudiantes por curso

➤ **Sprint 2**

- Administrar Rúbricas General
- Administrar Rúbricas por Curso
- Administrar Tipo de Niveles
- Visualizar Rúbrica General
- Visualizar Rúbrica por Curso
- Generar niveles por criterio
- Generar estructura de Rúbrica General
- Generar estructura de Rúbrica por curso

➤ **Sprint 3**

- Configurar Tipo de Equipos por Curso
- Asignar estudiantes a equipos y tipos de equipos
- Asignar evaluadores a equipos
- Planificar Evaluaciones
- Asignar evaluaciones a tipos de equipos
- Visualizar Equipos
- Asignar criterios a evaluaciones

➤ **Sprint 4**

- Generar aviso de evaluaciones programadas
- Realizar evaluación
- Visualizar evaluaciones programadas por semana
- Visualizar resultados de evaluación
- Buscar histórico de evaluaciones

➤ **Sprint 5**

- Generar indicadores de criterios cumplidos por estudiante
- Generar indicadores de curva de notas por estudiante
- Reporte de evaluaciones realizadas
- Generar reporte final de criterios por estudiante
- Permisos de acceso por perfil de usuario
- Generar seguridad al producto

Todos los detalles de las pruebas de aceptación del sistema se pueden ver en *Anexos F "Pruebas del Sistema"*.



CAPÍTULO 4: OBSERVACIONES, CONCLUSIONES Y RECOMENDACIONES

Este capítulo tiene como objetivo presentar las observaciones encontradas en este proyecto y resumir las conclusiones que se infieren a partir del desarrollo de la solución. Asimismo se sugieren algunas recomendaciones para trabajos futuros relacionados al tema de proceso de evaluaciones basadas en rúbricas en el contexto de una carrera acreditada.

4.1. Observaciones

Se presenta las observaciones encontradas durante el desarrollo del proyecto de fin de carrera:

- ✓ Las tecnologías utilizadas en el proyecto fueron con las versiones más recientes y elegidas por la experiencia del tesista en el manejo de las mismas.
- ✓ Se ha logrado identificar y proponer indicadores que podrían contribuir a llevar un mejor control sobre la planificación y evaluación de los estudiantes. Indicadores como: Criterios evaluados por estudiante y por curso, Curva de resultados de las evaluaciones por estudiante.

4.2. Conclusiones

Una vez finalizado el trabajo del proyecto de fin de carrera se puede concluir lo siguiente:

- ✓ Se ha logrado construir un sistema para la planificación y consolidación de las evaluaciones basadas en rúbricas para la especialidad de Ingeniería Informática.
- ✓ Se ha logrado definir las reglas de negocio del proceso de evaluación basados en rúbricas mediante entrevistas a los usuarios y observación de actividades en un curso de proyectos de la carrera de ingeniería informática de la PUCP con la finalidad de conocer las principales interacciones entre los usuarios con el proceso.

- ✓ Se ha logrado identificar las estructuras de datos necesarias para el manejo y almacenamiento de los datos que ingresan al sistema.
- ✓ Se ha logrado definir una arquitectura de software basado en el patrón (Modelo- Vista- Controlador) que permitió la construcción de un sistema que soporte la configuración de las evaluaciones por pares y por docentes.
- ✓ Se ha logrado realizar la toma de requisitos y diseño final de las pantallas mediante la elaboración de prototipos del sistema, las cuales se realizaron varias veces y ayudaron a la definición de requisitos.
- ✓ Se ha logrado desarrollar un sistema que permite la administración, planificación, evaluación y la generación de información para obtener resultados del programa.
- ✓ Se ha logrado realizar las pruebas del sistema con datos del curso INF226 de la especialidad de ingeniería informática de la PUCP.

4.3. Recomendaciones

Se recomienda para trabajos futuros relacionados al tema lo siguiente:

- ✓ Se puede agregar mayores funcionalidades al sistema como las reglas de negocio que no se consideraron para el alcance de este proyecto. Por ejemplo: reglas de evaluación de desempeño.
- ✓ Se puede diseñar una aplicación móvil para lograr la interacción de los usuarios con el sistema en diversos dispositivos.
- ✓ Se puede agregar sugerencias automáticas para la creación de una rúbrica para convertirse en un sistema experto.
- ✓ Se puede realizar las pruebas del sistema en diversas especialidades de la universidad para fines de acreditación u otros fines académicos.

REFERENCIAS BIBLIOGRÁFICAS

1. [ABET, 2013] ACCREDITATION BOARD FOR ENGINEERING AND TECHNOLOGY
2012 Consulta: 13 de septiembre de 2012
< www.abet.org>

2. [BUJAR, 2011] BUJAN VIDALES, Karmele; REKALDE RODRÍGUEZ, Itziar; ARAMENDI JÁUREGUI, Pello
2011 *La evaluación de competencias en la educación superior las rúbricas como instrumento de evaluación.* Primera edición. Bogotá: Editorial MAD.

3. [BUSINESS, 2013] BUSINESS RULES GROUP
2013 The Business Rules Approach.
Consulta: 31 de agosto de 2013
www.businessrulesgroup.org/brmanifesto/BRManifiesto.pdf

4. [DAVILA, 2003] DÁVILA RAMÓN, Abraham. E.
2003 Aprendizaje Orientado por Proyectos: Una Aplicación en los Cursos de Ingeniería de Software.

5. [DAPE, 2007] DIRECCION ACADÉMICA DE PLANEAMIENTO Y EVALUACIÓN PUCP
2007 Plan Estratégico Institucional. Lima.
Consulta:
20 de septiembre de 2012
http://www.pucp.edu.pe/EN/documento/pucp/plan_estrategico_pucp.pdf

6. [ECO,2012] ECO
2012 Aplicación informática ECO
Consulta: 12 de septiembre de 2012
<http://miguelangeljimenezrodriguez.blogspot.com/p/eco-evaluacion-por-competencias.html>

7. [XP,2013] Extreme Programming (XP)
2013 Fecha de consulta: 05 de Octubre de 2012
< <http://www.extremeprogramming.org/>>
8. [GARCIA, 2009] GARCÍA GARCÍA, María José, TERRÓN LÓPEZ, María José y BLANCO ARCHILLA, Yolanda
2009 “Desarrollo de Recursos Docentes para la Evaluación de Competencias Genéricas”, Revista de AENUI volumen 3, número 2. Consulta: 12 de septiembre de 2012. < <http://www.aenui.net/ojs/index.php?journal=revision>>
9. [ISO/IEC 24765, 2011] Systems And Software Engineering - Vocabulary
2011 EEUU
10. [SOTO, 2006] MARTÍNEZ SOTO, Juan Carlos
2005 *Sistema de evaluación de desempeño de estudiantes en proyectos de cursos universitarios*. Tesis de Licenciatura en Ciencias e Ingeniería con mención en Ingeniería Informática. Lima: Pontificia Universidad Católica del Perú, Facultad de Ciencias e Ingeniería.
11. [MCDAVID, 2006] MCDAVID, James y INGLESÓN, Laura R.
2006 *Program Evaluation and Performance Measurement: An Introduction to Practice*. Sage Publications. Segunda edición. Canada: Consulta: 12 de septiembre de 2012. <http://www.sagepub.com/upm-data/6195_Chapter_5___McDavid_I_Proof_3.pdf>
12. [MINTE, 2008] MINTE MÜNZENMAYER, Andrea
2008 *Una mirada a la evaluación por competencias en la educación superior*. Universidad: revista de la Asamblea Nacional de Rectores -- No. 12.

13. [MOODLE, 2012] MOODLE Moodle.org
Consulta: 12 de septiembre de 2012
<http://docs.moodle.org/22/en/Rubrics>
14. [ODEL, 1998] JAMES MARTIN, JAMES J. ODELL
1998 Object-Oriented Methods: A Foundation.
Prentice Hall.
Consulta: 28 de agosto de 2013
http://subs.emis.de/LNI/EMISA-Forum/Volume18_2/Fachbeitrag.pdf
15. [RAE, 2012] RAE
2012 Real Academia Española
Consulta: 15 de septiembre de 2012
<<http://www.rae.es/rae.html>>
16. [REDDY, 2010] REDDY, Y. Malini y ANDRADE, Heidi
2010 *A review of rubric use in higher education.*
Assessment & Evaluation in Higher Education. Volumen 35, número 4.
Consulta 20 de septiembre 2012.
<http://class.web.nthu.edu.tw/ezfiles/669/1669/img/1381/6.Areviewofrubricuseinhighereducation.pdf>
17. [RUBISTAR, 2012] RUBISTAR
2012 Herramienta web de creación de rúbricas.
Consulta: 20 de septiembre 2012.
< <http://rubistar.4teachers.org/>>
18. [SIMOS, 2001] SIMON, Marielle y Renée Forgette-Giroux
2001 “A rubric for scoring postsecondary academic skills”. Practical Assessment, Research & Evaluation, volumen 7, número 18. Consulta: 14 de septiembre de 2012. <
<http://pareonline.net/getvn.asp?v=7&n=18>
19. [SKILL, 2012] SKILLSTATION
2012 Sistema de Gestión de Competencias.
Consulta: 15 de septiembre de 2012.
< <http://www.e4learningsolutions.co.uk/software/default.aspx>>

20. [STEVENS,2005] STEVENS, Danelle D.
2005 *Introduction to rubrics: An assessment tool to save grading time, conveys effective feedback, and promote student learning.* Primera edición. Sterling, Va: Stylus Pub.2.
21. [SCRUM, 2013] SCRUM
2011 SCRUM Manager
Fecha de consulta: 01 de Octubre de 2012
<<http://home.hib.no/ai/data/master/mod251/2009/articles/scrum.pdf>>
22. [TALENT,2012] TALENT TEST V3.0
2012 Sistema profesional de evaluación de personal Consulta: 15 de septiembre de 2012 < <http://www.talent-test.com.mx/>>
23. [TOBON, 2006] TOBON, Sergio
2006 *Competencias, calidad y educación superior.* Bogotá: Cooperativa Editorial Magisterio
24. [TORREALVA, 2010] TORREALVA DÁVILA, Daniel
2010 "Ponencia presentada en el Conversatorio de Los Retos de la Educación Superior y la Acreditación y Autoevaluación Universitaria. Cajamarca. Consulta: 12 de septiembre de 2011.
<<http://www.pucp.edu.pe/educacion360/#/videos>>
25. [ISO29110,2011] TECHNICAL REPORT ISO/IEC TR 29110-5-1-2
2011 Software engineering — Lifecycle profiles for Very Small Entities (VSEs). Primera edición.

26. [VIVANCO.2010] VIVANCO ORTIZ, Yoshi Abel
2010 *Análisis, diseño e implementación de una herramienta Web de evaluación del desempeño por competencias: evaluación de 360° grados.* Tesis de Licenciatura en Ciencias e Ingeniería con mención en Ingeniería Informática. Lima: Pontificia Universidad Católica del Perú, Facultad de Ciencias e Ingeniería.

