

# PONTIFICIA UNIVERSIDAD CATÓLICA DEL PERÚ

## FACULTAD DE CIENCIAS E INGENIERÍA



PONTIFICIA  
**UNIVERSIDAD**  
**CATÓLICA**  
DEL PERÚ

### IMPLEMENTACIÓN DE UN SISTEMA DE INFORMACIÓN DE APOYO AL PROCESO DE EVALUACIÓN DE CURSOS PARA UNA INSTITUCIÓN DE EDUCACIÓN INFORMÁTICA

Tesis para optar por el Título de **Ingeniero Informático**, que presenta el bachiller:

**MANUEL HAZAEL COLOMA BARDALES**

**ASESOR: ING. BRAULIO OSCAR MURILLO VELIZ**

Lima, Noviembre de 2015

## RESUMEN

El Instituto de Informática de la Pontificia Universidad Católica del Perú (InfoPUC) es una organización dedicada a la educación informática, tanto para alumnos de la comunidad universitaria, como para los organismos privados y públicos relacionados a ella. Como parte de los servicios que brinda, está el Programa de Desarrollo Educativo con el uso de las TIC. Dentro de este programa se ofrecen cursos a través de una plataforma virtual, los cuales se evalúan de manera presencial en cada colegio. Para estas evaluaciones, InfoPUC envía evaluadores contratados para dicho trabajo, en su mayoría alumnos universitarios con horarios diversos. En estos momentos la organización cuenta con más colegios en convenio educativo (80) que al inicio del programa (5 a 10), y es muy probable que esta cifra siga en aumento. Según manifiestan, desde el 2013 les ha sido engorroso trabajar y organizar la cantidad de información de las evaluaciones mediante hojas de cálculo. La disponibilidad de tiempo de los evaluadores es un tema complicado de gestionar, ya que como se mencionó, son en mayoría alumnos de la universidad con horarios muy variados y cuyo tiempo disponible para trabajar para InfoPUC es distinto día a día. Asignar evaluadores a cada evaluación manualmente con la cantidad actual de evaluaciones por colegio, es una actividad que consume muchas horas hombre y que conforme crece el número de colegios en convenio se convierte en un cuello de botella. Es por ello que el presente proyecto de fin de carrera brinda una solución mediante la implementación de un sistema de información que permita realizar el proceso de asignación de evaluadores a evaluaciones de forma automática, con ello se reducirá el tiempo empleado en esta actividad y permitirá mantener tiempos óptimos de asignación en los siguientes semestres y años en que crecerán el número de colegios en convenio.



## Contenido

<b>CAPÍTULO 1</b> .....	<b>6</b>
<b>1. Problemática</b> .....	<b>6</b>
<b>1.1. Objetivo general</b> .....	<b>8</b>
<b>1.2. Objetivos específicos</b> .....	<b>8</b>
<b>1.3. Resultados esperados</b> .....	<b>9</b>
<b>2. Herramientas, métodos, metodologías y procedimientos</b> .....	<b>9</b>
<b>2.1. Herramientas</b> .....	<b>9</b>
<b>2.2. Metodologías</b> .....	<b>10</b>
<b>2.2.1. Metodología para la gestión del proyecto</b> .....	<b>11</b>
<b>2.2.2. Metodología para la gestión del producto</b> .....	<b>12</b>
<b>3. Alcance</b> .....	<b>12</b>
<b>3.1. Limitaciones</b> .....	<b>12</b>
<b>3.1.1. Limitaciones de la solución</b> .....	<b>12</b>
<b>3.2. Riesgos</b> .....	<b>13</b>
<b>4. Justificativa y viabilidad del proyecto</b> .....	<b>14</b>
<b>4.1. Justificativa</b> .....	<b>14</b>
<b>4.2. Viabilidad</b> .....	<b>14</b>
<b>4.2.1. Viabilidad técnica</b> .....	<b>14</b>
<b>4.2.2. Viabilidad temporal</b> .....	<b>15</b>
<b>4.2.3. Viabilidad económica</b> .....	<b>15</b>
<b>CAPÍTULO 2</b> .....	<b>16</b>
<b>1. Marco teórico</b> .....	<b>16</b>
<b>1.1. Marco conceptual</b> .....	<b>16</b>
<b>1.1.1. Conceptos relacionados con el problema</b> .....	<b>16</b>
<b>1.1.1.1. Colegio o institución educativa</b> .....	<b>16</b>
<b>1.1.1.2. Colegio en convenio con InfoPUC</b> .....	<b>16</b>
<b>1.1.1.3. Cursos contratados</b> .....	<b>16</b>
<b>1.1.1.4. Asesora educativa</b> .....	<b>16</b>
<b>1.1.1.5. Coordinadora</b> .....	<b>17</b>
<b>1.1.1.6. Evaluador</b> .....	<b>17</b>
<b>1.1.1.7. Modalidades de evaluación</b> .....	<b>17</b>

1.1.1.8.	Proceso de evaluación de los cursos contratados.....	17
1.1.1.9.	Periodo de Pagos .....	17
1.1.2.	Conceptos relacionados a la propuesta de solución .....	18
1.1.2.1.	La fórmula de Haversine .....	18
1.1.2.2.	El API de Direcciones de Google .....	18
1.2.	Marco Legal .....	18
	Ley de Protección de Datos Personales .....	19
2.	Estado del arte.....	19
2.1.	OPTIHPER (Asignación Optimizada de Horarios al Personal) (OPITHPER, 2014) 20	
2.2.	DRoster Employee Scheduling Software (KAPPIX, 2014).....	21
2.3.	aSg Horarios.....	22
2.4.	GHC 17.....	22
2.5.	Conclusiones sobre el estado del arte .....	22
	CAPÍTULO 3 .....	24
	Modelado del proceso de evaluación.....	24
	CAPÍTULO 4 .....	26
1.	Lista de requerimientos.....	26
2.	Arquitectura de la solución .....	28
2.1.	Términos a utilizar.....	28
2.2.	Representación de la arquitectura .....	28
2.3.	Vista Lógica .....	30
2.4.	Vista de Despliegue .....	31
2.5.	Vista de Implementación o Desarrollo .....	32
2.6.	Metas y Restricciones de la arquitectura.....	33
2.6.1.	Metas.....	33
2.6.2.	Restricciones .....	33
3.	Estándares de la interfaz gráfica .....	33
3.1.	Estructura de la plantilla.....	33
3.2.	Mensajes de confirmación de acciones .....	34
3.3.	Mensajes al usuario.....	35
3.4.	Botones .....	35
3.5.	Colores.....	35

<b>CAPÍTULO 5 .....</b>	<b>36</b>
1. Módulos del sistema .....	36
2. Prototipo funcional del sistema.....	38
3. Gestión de pagos .....	45
3.1. Periodo de evaluaciones.....	45
3.2. Administración de pagos .....	45
3.3. Conclusión .....	47
<b>CAPITULO 6 .....</b>	<b>48</b>
1. Asignación automática de evaluadores.....	48
1.1. Criterio de selección .....	48
1.2. Selección de evaluadores más óptimos.....	48
1.3. Funcionalidad de asignación automática.....	49
1.4. Conclusión .....	51



## ÍNDICE DE TABLAS

TABLA 1: TABLA DE RESULTADOS ESPERADOS CON LAS HERRAMIENTAS A USAR EN CADA UNO DE ESTOS. [ELABORACIÓN PROPIA] .....	9
TABLA 2: RIESGOS IDENTIFICADOS EN EL PROYECTO DE FIN DE CARRERA. [ELABORACIÓN PROPIA] .....	13
TABLA 3: COSTOS DEL PROYECTO. [ELABORACIÓN PROPIA] .....	15
TABLA 4: TABLA COMPARATIVA DEL ESTADO DEL ARTE ENCONTRADO. [ELABORACIÓN PROPIA]...	22
TABLA 5: LISTA DE REQUERIMIENTOS DEL SISTEMA. [ELABORACIÓN PROPIA] .....	26
TABLA 6: CLASES USADAS PARA LOS MENSAJES. [ELABORACIÓN PROPIA] .....	35
TABLA 7: CLASES USADAS PARA LOS BOTONES. [ELABORACIÓN PROPIA] .....	35
TABLA 8: COLORES USADOS EN LA INTERFAZ. [ELABORACIÓN PROPIA] .....	35

## ÍNDICE DE IMÁGENES

IMAGEN 1: EJEMPLO DE HORARIO DE UN EVALUADOR. [ELABORACIÓN PROPIA] .....	7
IMAGEN 2: EJEMPLO DE UN ESCENARIO TÍPICO DEL SISTEMA OPTIHPER. [OPTIHPER].	20
IMAGEN 3: VISTA RESUMEN DE ASIGNACIÓN DE TAREAS POR EMPLEADO. [KAPPIX] ...	21
IMAGEN 4: PROCESO DE EVALUACIÓN DE CURSOS CONTRATADOS. [ELABORACIÓN PROPIA].....	24
IMAGEN 5: SUBPROCESO DE ASIGNACIÓN DE EVALUADORES. [ELABORACIÓN PROPIA] .....	25
IMAGEN 6: MODELO DEL PROCESO ADECUADO AL SISTEMA A IMPLEMENTAR. [ELABORACIÓN PROPIA] .....	25
IMAGEN 7: MODELO VISTA CONTROLADOR EN UNA APLICACIÓN WEB. [ELABORACIÓN PROPIA].....	30
IMAGEN 8: VISTA LÓGICA. [ELABORACIÓN PROPIA].....	31
IMAGEN 9: VISTA DE DESPLIEGUE. [ELABORACIÓN PROPIA].....	32
IMAGEN 10: VISTA DE IMPLEMENTACIÓN. [ELABORACIÓN PROPIA].....	32
IMAGEN 11: ESTRUCTURA DEL CONTENIDO DEL SISTEMA. [ELABORACIÓN PROPIA] .....	34
IMAGEN 12: MENSAJE DE CONFIRMACIÓN. [ELABORACIÓN PROPIA] .....	34
IMAGEN 13: CASOS DE USO DEL MÓDULO DE USUARIOS. [ELABORACIÓN PROPIA] .....	36
IMAGEN 14: CASOS DE USO DEL MÓDULO DE CURSOS. [ELABORACIÓN PROPIA] .....	36
IMAGEN 15: CASOS DE USO DEL MÓDULO DE COLEGIOS. [ELABORACIÓN PROPIA] .....	37
IMAGEN 16: CASOS DE USO DEL MÓDULO DE EVALUADORES. [ELABORACIÓN PROPIA] .....	37
IMAGEN 17: CASOS DE USO DEL MÓDULO DE EVALUACIONES. [ELABORACIÓN PROPIA] .....	38
IMAGEN 18: CASOS DE USO DEL MÓDULO DE PAGOS. [ELABORACIÓN PROPIA] .....	38
IMAGEN 19: INGRESO AL SISTEMA. [ELABORACIÓN PROPIA].....	39
IMAGEN 20: PESTAÑA PARA ADMINISTRAR CURSOS. [ELABORACIÓN PROPIA].....	39
IMAGEN 21: PESTAÑA PARA ADMINISTRAR COLEGIOS. [ELABORACIÓN PROPIA] .....	40
IMAGEN 22: PESTAÑA PARA ADMINISTRAR EVALUADORES. [ELABORACIÓN PROPIA].....	41
IMAGEN 23: VENTANA PARA ASIGAR DISPONIBILIDAD DE LOS EVALUADORES. [ELABORACIÓN PROPIA] .....	42
IMAGEN 24: BÚSQUEDA DE UNA EVALUACIÓN. [ELABORACIÓN PROPIA].....	43
IMAGEN 25: PESTAÑA PARA ADMINISTRAR USUARIOS. [ELABORACIÓN PROPIA].....	44
IMAGEN 26: VENTANA PARA ADMINISTRAR LOS PAGOS. [ELABORACIÓN PROPIA] .....	44
IMAGEN 27: VENTANA PARA CREAR UN PERIODO DE PAGOS. [ELABORACIÓN PROPIA].....	45
IMAGEN 28: RESULTADO DE LA BÚSQUEDA DE UN PERIODO. [ELABORACIÓN PROPIA] .....	46
IMAGEN 29: PÁGINA DE ADMINISTRACIÓN DE ASISTENCIA DEL ADMINISTRADOR DEL SISTEMA. [ELABORACIÓN PROPIA] .....	47
IMAGEN 30: PÁGINA DE ADMINISTRACIÓN DE ASISTENCIA DE LA ASESORA EDUCATIVA. [ELABORACIÓN PROPIA].....	47
IMAGEN 31: FORMULARIO DE ASIGNACIÓN AUTOMÁTICA [ELABORACIÓN PROPIA].....	49
IMAGEN 32: RESULTADO DE LA ASIGNACIÓN AUTOMÁTICA.[ELABORACIÓN PROPIA].....	50
IMAGEN 33: ASIGNACIÓN POR EVALUACIÓN. [ELABORACIÓN PROPIA].....	50

## CAPÍTULO 1

### 1. Problemática

El Instituto de Informática de la Pontificia Universidad Católica del Perú (InfoPUC) es una organización dedicada a la educación informática, tanto para alumnos de la comunidad universitaria, como para los organismos privados y públicos relacionados a ella. Como parte de los servicios que brinda, está el Programa de Desarrollo Educativo con el Uso de las TIC (Tecnologías de la información y la comunicación). Dicho programa ofrece a los colegios en convenio educativo, la plataforma virtual educativa PAIDEIA, la cual les sirve como aula virtual donde pueden tener sus cursos de manera digital. También brindan cursos virtuales vía PAIDEIA, cuyas evaluaciones se realizan de manera presencial en cada colegio. Los alumnos reciben una certificación a nombre de InfoPUC al finalizar estos cursos. (INFOPUC, 2012)

Este programa tiene dos procesos principales: i) el proceso de preparación de la plataforma virtual educativa PAIDEIA para los colegios en convenio y ii) el proceso de evaluación para los cursos contratados. Esta solución se enfoca en el segundo proceso, que es donde se presenta la problemática y es éste el que se busca solucionar mediante este proyecto de fin de carrera.

Cada trimestre InfoPUC brinda a los colegios un periodo de tres a cuatro semanas para que puedan acomodar la fecha y hora de las evaluaciones de los cursos contratados. Las evaluaciones se llevan a cabo en el local de cada colegio, donde InfoPUC envía evaluadores contratados para dicho trabajo, los cuales tienen como función controlar la correcta ejecución de la evaluación, apoyar al alumno ante alguna duda y, posteriormente corregir la evaluación y entregarla a InfoPUC. En su mayoría, los evaluadores son alumnos de la universidad.

El primer paso de este proceso es pedir a los colegios las fechas en las que se realizarán las evaluaciones a los cursos que hayan contratado en el trimestre vigente. Esta información es pedida por cada una de las asesoras educativas, quienes tienen a su cargo un número determinado de colegios. Después de recibir dicha información se la envían a la coordinadora, la cual es la encargada de armar el cronograma de evaluaciones del trimestre.

Una vez que se tiene este cronograma de evaluaciones, se asigna para cada evaluación a uno o dos evaluadores según el número de alumnos a evaluar.

Una vez que se tienen las evaluaciones a las que asistirá cada evaluador, se le hace llegar dicha información vía correo electrónico. Luego de asistir a la evaluación, el evaluador debe corregirla y poner la nota que le corresponda por medio de la plataforma PAIDEIA.

Este proceso termina con el pago a los evaluadores. Las asesoras educativas se encargan de verificar que estos hayan cumplido su trabajo, y hacen un informe a la coordinadora, la cual realiza el consolidado de pagos correspondiente a cada evaluador. Esta información es enviada al área encargada de pagos.

En un inicio, InfoPUC contaba con un número de colegios relativamente bajo, entre cinco y diez, por lo cual para gestionar el proceso de evaluación de los cursos contratados les era suficiente y cómodo usar Microsoft Word. Después, los colegios aumentaron a una cantidad entre 20 y 50, por lo cual Microsoft Word ya no era una herramienta cómoda; fue entonces que decidieron usar Microsoft Excel. En estos momentos la organización cuenta con más colegios en convenio educativo (80), y es muy probable que esta cifra siga en aumento. Según manifiestan las asesoras educativas, desde el 2013 les ha sido engorroso trabajar y organizar la cantidad de información de las evaluaciones mediante hojas de cálculo.

El principal problema es saber a qué evaluador se le asignará una evaluación. Hoy día, para esta elección se toma en cuenta principalmente los siguientes tres factores:

- Disponibilidad de tiempo: El evaluador debe tener disponible en su calendario las fechas y horas de las evaluaciones acordadas con el colegio.
- Distancia al colegio: Para asegurar que el evaluador pueda ir a la evaluación, normalmente se elige de entre los evaluadores que cumplen con la disponibilidad de tiempo, aquellos que vivan o se encuentren más cerca del colegio.
- Conocimiento del evaluador: Dependiendo del contenido del curso, se prefiere asignar a un evaluador que domine o conozca del tema, pues debe despejar las dudas del alumno y corregir las evaluaciones.

Con respecto a la disponibilidad de tiempo de los evaluadores, son en su mayoría alumnos de la universidad y tienen horarios disponibles para evaluaciones de manera variada, como se puede apreciar en la Imagen 1.

	Lunes	Martes	Miércoles	Jueves	Viernes
8:00 – 9:00		Libre	Libre		Libre
9:00 – 10:00		Libre	Libre		Libre
10:00 – 11:00	Libre		Libre		Libre
11:00 – 12:00	Libre			Libre	Libre
12:00 – 13:00	Libre			Libre	
13:00 – 14:00				Libre	
14:00 – 15:00		Libre	Libre		
15:00 – 16:00		Libre	Libre		
16:00 – 17:00		Libre			
17:00 – 18:00					
18:00 – 19:00					
19:00 – 20:00					
20:00 – 21:00					
21:00 – 22:00					

**Imagen 1: Ejemplo de horario de un evaluador. [Elaboración propia]**

Asignar evaluadores a evaluaciones manualmente con la cantidad actual de evaluaciones por colegio (en promedio ocho evaluaciones por colegio), para ochenta colegios, es una actividad que consume mucho tiempo para las asesoras.

Realizar esta actividad manualmente es muy propenso a que se cometan errores. Por ejemplo, en una de las reuniones con las asesoras educativas manifiestan que ha sucedido que asignan a un evaluador que ya estaba asignado ese mismo día a la misma hora en otro colegio. Si esto no es detectado en el momento adecuado, el segundo colegio se quedará esperando a un evaluador que nunca se presentará. Esto puede causar

que el colegio quede en descontento con la Institución, llegando incluso a no querer renovar contrato con InfoPUC al año siguiente.

Por otro lado está el tema de los pagos. Antes se pagaba a los evaluadores por día un monto fijo, así estos hayan asistido a una o más evaluaciones en un mismo día. Las asesoras tienen que comprobar que el evaluador acudió a la evaluación y que la corrigió. Actualmente la organización quiere cambiar esta política, pues a algunos evaluadores se les pagaba igual que a otros que trabajaban menos, por lo que ahora aplicarán penalidades por evaluaciones no corregidas.

La coordinadora del área tiene que realizar el consolidado de pagos de cada evaluador, actividad manual que demanda mucho tiempo.

En resumen, se tiene un proceso que, dadas las condiciones actuales se ha vuelto complicado de gestionar con las tecnologías de información con las que se venía trabajando y que consume muchas horas hombre, tiempo que puede ser utilizado en otras labores.

Por ello, se propone como solución a la problemática planteada, diseñar un sistema de información que permita:

- Reducir el tiempo empleado por la coordinadora en asignar evaluadores a las evaluaciones.
- Mantener tiempos de asignación óptimos en los siguientes semestres y años en que crecerán el número de colegios y evaluaciones.
- Asignar a los evaluadores de forma que la calidad del servicio de evaluación se mejore en el tiempo, asignando a los mejores evaluadores, es decir mejorando indicadores de asistencia, no tardanzas, evaluaciones calificadas y subidas al PAIDEIA.
- Gestionar el pago a los evaluadores de una forma más ordenada y rápida.

### **1.1. Objetivo general**

Analizar, diseñar e implementar un sistema de información para una organización dedicada a la educación informática que permita gestionar el proceso de evaluación de los cursos que ésta ofrece.

### **1.2. Objetivos específicos**

- 1) Diseñar el modelo del proceso de evaluaciones de los cursos contratados por los colegios.
- 2) Elaborar la arquitectura de software que soporte la solución planteada.
- 3) Elaborar un prototipo funcional del sistema que cumpla con los requisitos levantados.
- 4) Implementar una función que permita filtrar (según disponibilidad de tiempo, distancia del evaluador a la evaluación y conocimientos sobre la evaluación) a los evaluadores mostrando de forma ordenada a los más idóneos para la evaluación.

- 5) Implementar la asignación automática de evaluadores más aptos a cada evaluación.

### 1.3. Resultados esperados

- Resultado 1 para el objetivo 1: Modelo del proceso al cual se busca dar soporte.
- Resultado 2 para el objetivo 2: Diseño de la arquitectura de software para la solución ofrecida.
- Resultado 3 para el objetivo 3: Prototipo funcional del sistema.
- Resultado 4 para el objetivo 4: Filtro de evaluadores según disponibilidad de tiempo, distancia del evaluador a la evaluación y conocimientos sobre la evaluación.
- Resultado 5 para el objetivo 5: Función de asignación automática.

## 2. Herramientas, métodos, metodologías y procedimientos

Este capítulo tiene como finalidad mostrar las herramientas, métodos, metodologías y procedimientos que serán usados para la gestión y el desarrollo del presente proyecto de fin de carrera. En la siguiente tabla se muestra un mapeo entre estos y los resultados esperados.

**Tabla 1: Tabla de resultados esperados con las herramientas a usar en cada uno de estos.**  
[Elaboración propia]

Resultados esperados	Herramientas a usarse
RE1: Modelo del proceso al cual se busca dar soporte	BPMN, Lucidchart
RE2: Diseño de la arquitectura de software para la solución ofrecida	StarUML
RE3: Prototipo funcional del sistema	Yii Framework, PHP, Bootstrap, jQuery, Base de datos Mysql, Servidor Apache
RE4: Filtro de evaluadores según disponibilidad de tiempo, distancia del evaluador a la evaluación y conocimientos sobre la evaluación	Yii Framework, PHP, Bootstrap, Base de datos Mysql, Servidor Apache, API de matriz de distancia de Google, Angular JS
RE5: Función de asignación automática	Base de datos Mysql, Servidor Apache, API de matriz de distancia de Google, Angular JS

### 2.1. Herramientas

- **BPMN**

BPMN es el acrónimo de Business Process Modeling Notation, notación gráfica usada para el modelamiento de procesos. Tiene como propósito proporcionar a las empresas la capacidad de comprensión de sus procedimientos internos de negocio en una notación gráfica, ésta da a las organizaciones la capacidad de comprender estos procedimientos de manera estándar (Object Management Group, 2014).

- **StarUML**

StarUML es un programa que permite hacer gráficos en la notación UML 2.0.

- **Yii Framework**

Yii es un marco de programación PHP basado en componentes de alta performance para desarrollar aplicaciones Web de gran escala. El mismo permite la máxima reutilización en la programación web y puede acelerar el proceso de desarrollo (Yii Software LLC, 2014). Se eligió este marco de programación de desarrollo por motivos de experiencia en su uso, además de poseer una estructura bien definida sobre el patrón MVC y gran cantidad de documentación y complementos que permiten tener un mejor producto.

- **PHP**

PHP (acrónimo recursivo de *PHP: Hypertext Preprocessor*) es un lenguaje de código abierto muy popular especialmente adecuado para el desarrollo web y que puede ser incrustado en HTML (The PHP Group, 2001).

Para este proyecto se requería tener un sistema que pueda ser accedido no solo desde la oficina, sino desde otros lugares como en el hogar de las asesoras educativas o cuando estén visitando un colegio en provincia, esto motivó a realizar una aplicación web y se optó por PHP ya que es un lenguaje de programación web con muchos años en el mercado, cuenta con abundante documentación. Otro motivo por el que se optó por este lenguaje es la experiencia en su uso.

- **Bootstrap**

Es un marco de desarrollo para realizar el maquetado y diseño de páginas web, que brinda facilidades para realizar productos con propiedades de diseño responsivo (diseño que responde al tamaño del dispositivo desde el que se está visualizando la web), permitiendo adaptar su diseño a diferentes resoluciones y dispositivos. Se utilizará este marco de diseño pues es el que se usa en los proyectos desarrollados en la Dirección Informática Académica, por lo que se busca mantener uniformidad en el diseño.

- **jQuery**

jQuery es una librería JavaScript ligera y rápida que cuenta con muchas buenas características. Se usará esta librería ya que permite realizar distintas acciones como: manipulación del DOM, manejar eventos, animaciones y uso de Ajax de una manera sencilla y gracias a esta API, el sistema podrá funcionar correctamente en múltiples navegadores web (The jQuery Foundation, 2014).

- **Base de Datos MySQL**

Base de datos relacional, multihilo y multiusuario. Como el sistema no requiere de una gran cantidad de concurrencia, esta base de datos se adapta muy bien.

## 2.2. Metodologías

A continuación se detallan las metodologías usadas, tanto para la gestión del Proyecto como para el desarrollo del mismo.

## 221. Metodología para la gestión del proyecto

Para un proyecto de fin de carrera como éste, se requieren emplear ciertas buenas prácticas sobre dirección de proyectos que permitan gestionar de forma adecuada y por ende garantizar la entrega a tiempo de los entregables. PMBOK presenta nueve áreas de conocimiento de la dirección de proyectos que ayudan en esto. A continuación se detallan cuáles de estas áreas se utilizaron.

### Gestión del Alcance del Proyecto

La Gestión del Alcance del Proyecto contiene los procesos necesarios para garantizar que el proyecto incluya todo (y únicamente todo) el trabajo requerido para completarlo con éxito. Esta gestión tiene como objetivo definir y controlar qué se incluye y qué no se incluye en el proyecto (PMBOK, 2013: 105).

De esta fase se realizó lo siguiente:

- Recopilar requisitos
- Definir alcance del proyecto y producto
- Crear la EDT

### Gestión del Tiempo del Proyecto

La Gestión del Tiempo del Proyecto incluye los procesos requeridos para administrar la finalización del proyecto a tiempo (PMBOK, 2013: 141).

Este proyecto de fin de carrera tiene una duración de un año, dentro de este tiempo se debe cumplir con todo lo que se especifique en la EDT y esto se podrá hacer siempre y cuando se cumpla en el plazo establecido cada una de las actividades definidas.

Siguiendo lo que PMBOK recomienda realizar, se hizo lo siguiente:

- Definir las actividades
- Secuenciar las actividades
- Estimar la duración de las actividades
- Desarrollar el cronograma

### Gestión de los Riesgos del Proyecto

La Gestión de los Riesgos del Proyecto incluye los procesos relacionados con llevar a cabo la planificación de la gestión, la identificación, el análisis, la planificación de respuesta a los riesgos, así como su monitoreo y control en un proyecto. Tiene como objetivo aumentar la probabilidad y el impacto de eventos positivos, y disminuir la probabilidad y el impacto de eventos negativos para el proyecto (PMBOOK, 2013: 309).

Dada la importancia de conocer los riesgos presentes en el Proyecto, se ha desarrollado la matriz de riesgos que se aprecia en la Tabla 2.

## **222 Metodología para la gestión del producto**

Ya que se trabajará sobre un entorno donde la relación con el usuario del sistema es bastante estrecha y se tienen procesos definidos, se considerarán algunos puntos de la metodología RUP (Rational Unified Process) para este desarrollo.

De la recopilación de requisitos que se realizó en la gestión del alcance del proyecto se desarrolló el documento de especificación de requisitos de software, donde se detalla lo que el sistema realiza para cubrir las necesidades del usuario, también sirve para definir cómo es que los usuarios harán uso del mismo. Teniendo esto se tendrá una mejor visión de lo que el sistema hace y cómo es que lo hace.

Así mismo se desarrolló el documento de arquitectura y los estándares de interfaz gráfica que se siguen.

### **3. Alcance**

La solución expuesta es un proyecto de implementación y se encuentra relacionada con las instituciones educativas, específicamente dedicadas a la educación mediante el uso de las tecnologías de información y comunicaciones. En la institución donde se desarrolló el Proyecto en específico, hay deficiencias en el proceso de evaluación de sus cursos, esto principalmente en la actividad de asignar evaluadores a las evaluaciones y en la gestión de pagos.

Es por ello que este proyecto de fin de carrera se centra en la implementación de un sistema de información que permita ayudar en estas dos actividades: optimizar el tiempo empleado para la asignación de evaluadores y en la gestión de pagos.

Para optimizar la asignación de evaluadores a las evaluaciones se proporcionará un módulo dentro del sistema que cuente con una función que permita realizar una asignación automática de evaluadores, tomando en cuenta su disponibilidad de tiempo, cercanía al lugar de la evaluación y conocimiento sobre el tema a evaluar.

Para optimizar la gestión de pagos, la cual se viene realizando mediante diferentes hojas de cálculo en MS-Excel, se proporcionará un módulo en el sistema que permita llevar un registro más ordenado de los pagos por día que se da a los evaluadores, así como registrar penalidades (reducción del monto del pago) que manejan en la organización. La gestión de penalidades la podrán hacer diferentes asesoras educativas, esta descentralización de trabajo permitirá reducir el tiempo empleado por la coordinadora del área en esta actividad.

#### **3.1. Limitaciones**

Se tomaron en cuenta las siguientes limitaciones:

##### **3.1.1. Limitaciones de la solución**

Parte de la solución planteada por este proyecto de fin de carrera es implementar un módulo en el sistema que permita realizar la gestión de pagos a los evaluadores. Este

módulo tiene como finalidad mostrar un reporte sobre el pago a realizar a cada evaluador y otro donde se detalle el pago de un evaluador en específico. Este reporte es usado para ser enviado a otra unidad de la Universidad, la cual es la encargada de realizar los pagos, es por este motivo que el sistema no lleva registros sobre recibos por honorarios o ninguna otra clase de documento contable.

### 3.2. Riesgos

En la Tabla 2 se muestran los riesgos que se han podido identificar, la calificación de estos riesgos y los planes de mitigación y contingencia respectivos para hacer frente a estos.

**Tabla 2: Riesgos identificados en el proyecto de fin de carrera. [Elaboración propia]**

Id	Riesgo	Impacto	Probabilidad	Severidad	Plan de Mitigación	Plan de Contingencia
R1	Perdida del código fuente o entregables del proyecto	5	6	30	Llevar un adecuado control de versiones del código fuente, así como también de los documentos. Hacer copias de seguridad del código fuente y la base de datos	Utilizar las copias de seguridad que se debieron hacer como plan de mitigación
R2	Falta de usuarios con los que probar el sistema	6	7	42	Llevar una buena comunicación con los usuarios involucrados y avisar con anticipación para evitar contratiempos	Obtener opiniones de otros usuarios, capacitar rápidamente a nuevos usuarios en el uso del sistema y hacer las pruebas.
R5	Mala revisión del sistema	8	5	40	Hacer revisiones sobre la revisión, para disminuir posibles errores no encontrados	Corregir los errores que se presenten lo más rápido posible
R4	Retraso en la presentación de entregables	7	8	56	Ordenar los horarios de forma adecuada y llevar un buen control de cuando es cada entrega	Acomodar los tiempos reduciendo el tiempo que se había separado para otras actividades con el fin de entregar a tiempo cada uno de los entregables
R5	Mala identificación de procesos de negocio	9	5	45	Pedir aprobación de los usuarios expertos sobre el diseño de procesos a realizar	Acortar funcionalidades de menos prioridad para hacer el ajuste de las que tienen mayor prioridad
R6	Cambios en los requisitos a poco tiempo de la entrega final	9	5	45	Haber hecho una exhaustiva etapa de análisis, comprobando que los requisitos tomados son los últimos.	Acondicionar los tiempos para poder incluir el nuevo requisito si es que es un requisito crítico para el sistema
R7	Cambios en las reglas de negocio	8	4	32	Corroborar con los usuarios finales que los requisitos no hayan cambiado	Acortar algunas funcionalidades de menor prioridad
R8	Mala identificación de la arquitectura de la solución	9	3	27	Realizar pruebas sobre la arquitectura planteada antes de entrar a la fase de producción	Realizar otro diseño de arquitectura

R9	Pérdida de asesor	10	3	30	Conocer el tiempo y disponibilidad del asesor antes y durante el tiempo que dure el proyecto de Tesis	Cambio de asesor
Impacto: 10 (Muy alto), 1 (Muy bajo)						
Probabilidad: 10 (Muy alto), 1 (Muy bajo)						
Severidad = Impacto * Probabilidad						

#### 4. Justificativa y viabilidad del proyecto

A continuación se expondrá la justificación y viabilidad del presente proyecto de fin de carrera.

##### 4.1. Justificativa

El desarrollo de este proyecto de fin de carrera servirá para reducir el tiempo empleado en el proceso de evaluación de los cursos brindados por InfoPUC, así como también evitar errores en la programación de estas evaluaciones (que conllevan a incrementar el tiempo empleado para el proceso).

Esta solución será de ayuda a la Institución ya que al reducir el tiempo que emplean las asesoras educativas en la planificación de la programación de evaluaciones, dispondrán de tiempo para otras actividades importantes del negocio. El tiempo que les toma las actividades de recolectar la información de colegios y evaluadores, y asignar evaluadores a evaluaciones es tres semanas, ahora se estima que les tomará una semana. Esto equivale a un ahorro de dos semanas en horas persona. Esto indica que ya no se tendrá que contratar a otra persona para apoyar en esta tarea si en un futuro el número de colegios en convenio y de evaluadores aumenta.

No solo se beneficiará la institución, también los colegios en convenio con InfoPUC y los evaluadores contratados. Los colegios se beneficiarían al disminuir las posibilidades de que un evaluador falte, a causa de ser asignado a una evaluación de un colegio muy alejado a él o que por culpa de un error en la asignación manual se programe a un evaluador que ya estaba ocupado en otro colegio.

Y los evaluadores también se podrían beneficiar, pues al ser programados en lugares más cercanos para ellos, reducen su gasto en pasajes y ahorran el tiempo que demoran en ir a colegios lejanos a ellos.

##### 4.2. Viabilidad

En este punto se detalla la viabilidad del proyecto desde distintos aspectos.

##### 4.2.1. Viabilidad técnica

En la fase de análisis del proyecto se necesita interactuar constantemente con el usuario final para poder conocer a fondo sus necesidades. Esta extracción de información será por medio de entrevistas. Dado que el Proyecto se realiza en la Institución en la cual realizo mi práctica pre-profesional, es muy factible conseguir reuniones con los

principales interesados y si surge cualquier duda durante cualquier etapa de la implementación del proyecto, se puede preguntar directamente a los interesados, personalmente o por correo electrónico.

En cuanto a los conocimientos que se necesitan que tener para el desarrollo, si bien ya hay un conocimiento relacionado a programación web, se aprenderá más a fondo el manejo de JavaScript. También se tiene que investigar en mayor detalle lo visto en el estado del arte para tener un marco teórico más profundo, principalmente sobre el algoritmo a usar para encontrar la mejor solución. Todo este estudio se planea realizar en el tiempo que queda hasta antes de la implementación del proyecto, el cual es un mes aproximadamente.

#### 4.2.2 Viabilidad temporal

De acuerdo a las necesidades planteadas, al alcance presentado y a los resultados esperados se ha determinado que el desarrollo del proyecto tenga una duración de tres meses. El tiempo empleado para el desarrollo del proyecto toma parte de las horas semanales del trabajo que realizo en InfoPUC.

#### 4.2.3 Viabilidad económica

Las herramientas usadas para la implementación del sistema de información son de distribución gratuita, por lo que no se necesita una inversión económica para el software a utilizar. En cuanto al hardware que se usará: para el desarrollo se usará una computadora personal, la cual ya está comprada (propiedad personal) y para poner en producción el producto final, la Institución cuenta con un espacio en Amazon, servidor en el cual se alojará el servicio.

En términos generales tanto lo que ya se invirtió en hardware como lo que se pagará por el espacio en Amazon, representan para la Institución un costo hundido, pues ese gasto ya no se recuperará.

Por estos motivos es que el proyecto es factible en términos económicos ya que no se necesitará inversiones económicas adicionales a los que ya se realizan actualmente de manera periódica.

La Tabla 3 muestra los costos promedios de la implementación de este proyecto.

**Tabla 3: Costos del Proyecto. [Elaboración propia]**

<b>Elemento de costo</b>	<b>Costo (S/.)</b>
Subvención económica del practicante por seis meses	4500.00
Útiles de escritorio	22.00
Computadora para desarrollo (depreciación a seis meses)	330.50
Herramientas de código abierto	0.00
<b>Total</b>	<b>4852.50</b>

## CAPÍTULO 2

### 1. Marco teórico

#### 1.1. Marco conceptual

A continuación se describen los conceptos relacionados directamente con el problema y la propuesta de solución presentada.

##### 1.1.1. Conceptos relacionados con el problema

###### 1.1.1.1. Colegio o institución educativa

Colegio es un “establecimiento de enseñanza para niños y jóvenes de uno u otro sexo” (RAE, 2001).

###### 1.1.1.2. Colegio en convenio con InfoPUC

Las instituciones educativas en convenio con InfoPUC son colegios particulares y con una cantidad de alumnos tal que ameritan que se envíen controladores para asistir en las evaluaciones.

Para que un colegio pase a estar en convenio con InfoPUC tiene que pasar por una evaluación y firmar un contrato.

###### 1.1.1.3. Cursos contratados

Son cursos de informática que InfoPUC brinda a los colegios, los que están separados por niveles. El material completo de estos cursos está en formato digital (fichas de aprendizaje, documentos, videos, tareas, etc.) y se sube a la plataforma virtual educativa PAIDEIA contratada por el colegio a principios del año escolar.

Dado que InfoPUC otorga un certificado a los alumnos que aprueban estos cursos, deben ser evaluados para comprobar su aprendizaje.

###### 1.1.1.4. Asesora educativa

Trabajadora de InfoPUC que tiene asignada una cantidad “x” de colegios, todas las trabajadoras del área de colegios son mujeres por regla del negocio. Es el enlace entre el colegio e InfoPUC.

Sus tareas son responder cualquier duda de los colegios que tiene a su cargo y pedir el cronograma para los periodos de evaluaciones.

#### **1.1.1.5. Coordinadora**

Es la persona con el puesto más alto en el área de colegios, está a cargo de las asesoras educativas. Su función en el proceso de evaluación de cursos es asignar a las personas que controlarán cada evaluación.

#### **1.1.1.6. Evaluador**

Persona contratada por InfoPUC para la labor de ayudar a los alumnos que tengan alguna duda durante una evaluación, ver que los alumnos no cometan alguna falta (plagio) que pueda conllevar a desaprobado la evaluación. También son encargados de corregir evaluaciones y subir las notas correspondientes a la plataforma PAIDEIA.

#### **1.1.1.7. Modalidades de evaluación**

Existen dos modalidades de evaluación, virtual y semi-virtual. A continuación se explicará cada una.

- Virtual

Evaluación que se lleva a cabo por medio de la plataforma PAIDEIA, en esta modalidad el evaluador corregirá las evaluaciones directamente en el PAIDEIA del curso.

- Semi-virtual

Evaluación que tiene una parte en la plataforma PAIDEIA y otra es escrita. En esta modalidad el asesor deberá recoger de InfoPUC las pruebas escritas, desplazarse al colegio y ejecutar la evaluación, corregir tanto la parte escrita como la de la parte virtual de la prueba y subir las notas de la evaluación al PAIDEIA del curso.

#### **1.1.1.8. Proceso de evaluación de los cursos contratados**

La evaluación de los cursos contratados por los colegios a InfoPUC se realiza cuatro veces en el año que dura el curso. InfoPUC da un lapso de tiempo de cuatro semanas para que los colegios puedan acomodar en ese tiempo los días en los que pueden ser evaluados. Este proceso abarca desde gestionar con los colegios el día y hora en que se hará la evaluación de cada curso, qué evaluador irá a cada evaluación y termina con la gestión del pago al evaluador.

#### **1.1.1.9. Periodo de Pagos**

Rango de fechas en las que se rinden las evaluaciones en los colegios, generalmente dura cuatro semanas. Este periodo de evaluaciones es tomado como un periodo de pagos.

## 1.1.2 Conceptos relacionados a la propuesta de solución

En este apartado se muestran los conceptos utilizados en el desarrollo del presente Proyecto de fin de carrera.

### 1.1.2.1. La fórmula de Haversine

La fórmula de Haversine es una ecuación importante usada en navegación para determinar la menor distancia entre dos puntos en un cuerpo esférico usando sus latitudes y longitudes. (Prof. Nitin R. Chopde – Mr. Mangesh K. Nichat, 2013)

Usando la fórmula de Haversine se asume que la tierra es una esfera exacta, aunque esta no lo sea. A pesar de esto, la fórmula de Haversine tiene mucha exactitud a excepción de puntos directamente opuestos en la Tierra (Jovin J. Mwemezi - Youfang Huang, 2011), el cual no podría darse en el contexto del presente proyecto de fin de carrera.

La fórmula de Haversine es la siguiente:

$$d = 2r \sin^{-1} \left( \sqrt{\cos^2\left(\frac{\phi_1 - \phi_2}{2}\right) + \cos(\phi_1) \cos(\phi_2) \sin^2\left(\frac{\lambda_1 - \lambda_2}{2}\right)} \right)$$

Donde  $d$  es la distancia entre dos puntos con longitud y latitud  $(\phi_1, \lambda_1)$  y  $(\phi_2, \lambda_2)$  y  $r$  es el radio de la Tierra.

Parte del problema de asignación, es elegir a quién está más cerca al colegio. Para el cálculo de la distancia entre un evaluador y el lugar de la evaluación se usó esta fórmula.

### 1.1.2.2. El API de Direcciones de Google

El API de direcciones de Google es un servicio que calcula direcciones entre locaciones usando una petición http. Este servicio está diseñado para el cálculo de direcciones conocidas con antelación, no está diseñado para responder en tiempo real. (GOOGLE)

Cuando se busca a los evaluadores disponibles para una evaluación en particular, puede darse el caso de que el evaluador se encuentre en otro colegio, por lo que se debe tomar en cuenta el tiempo que demora en ir desde donde se encuentra hasta el lugar de la evaluación. Para este cálculo se hizo uso de esta API.

## 1.2. Marco Legal

El presente apartado tiene como objetivo mostrar las implicancias legales de la solución y plantear recomendaciones a seguir para alinearse al marco legal.

A continuación se describirán los conceptos legales que se encuentran directamente relacionados con la solución planteada.

## Ley de Protección de Datos Personales

Para obtener la distancia entre el evaluador y el colegio se necesita que la persona proporcione su dirección, esto se considera como un dato personal pues según el punto cuatro del artículo 2 de la ley se describe a un dato personal como toda información sobre una persona natural que la identifica o la hace identificable a través de medios que pueden ser razonablemente utilizados. (Ley N° 29733, 2011)

Los datos sensibles son los datos personales constituidos por los datos biométricos que por sí mismos pueden identificar al titular; datos referidos al origen racial y étnico; ingresos económicos, opiniones o convicciones políticas, religiosas, filosóficas o morales; afiliación sindical; e información relacionada a la salud o a la vida sexual. (Ley N° 29733, 2011)

Ya que la dirección de una persona no forma parte de lo considerado como dato sensible por la Ley de Protección de Datos Personales, no requiere un tratamiento especial.

De la ley se puede rescatar los siguientes artículos:

### Artículo 5. Principio de consentimiento

Para el tratamiento de los datos personales debe mediar el consentimiento de su titular. (Ley N° 29733, 2011)

### Artículo 7. Principio de proporcionalidad

Todo tratamiento de datos personales debe ser adecuado, relevante y no excesivo a la finalidad para la que estos hubiesen sido recopilados. (Ley N° 29733, 2011)

### Artículo 9. Principio de seguridad

El titular del banco de datos personales y el encargado de su tratamiento deben adoptar las medidas técnicas, organizativas y legales necesarias para garantizar la seguridad de los datos personales. Las medidas de seguridad deben ser apropiadas y acordes con el tratamiento que se vaya a efectuar y con la categoría de datos personales de que se trate. (Ley N° 29733, 2011)

De todo lo anterior se concluye en que se debe pedir consentimiento del evaluador para utilizar su dirección y se debe proteger el acceso a estos datos.

Se recomienda al Instituto de Informática que inscriba la base de datos personal en el Registro Nacional de Protección de Datos personales pues de acuerdo al artículo 38, de no hacer esto se estaría cometiendo una infracción grave y las infracciones graves son sancionadas con multa desde más de cinco unidades impositivas tributarias (UIT) hasta cincuenta unidades impositivas tributarias (UIT) (Ley N° 29733, 2011)

## 2. Estado del arte

El siguiente apartado tiene como objetivo dar a conocer como se ha tratado de solucionar problemas similares a este. También se busca demostrar que no hay un producto que de la solución específica a esta problemática.

A continuación se muestran los sistemas de información encontrados en el mercado que permiten resolver la problemática planteada.

### 2.1. OPTIHPER (Asignación Optimizada de Horarios al Personal) (OPITHPER, 2014)

Software desarrollado por la Universidad Politécnica de Valencia para la asignación automática de horarios y tareas al personal de una empresa, teniendo en cuenta el conjunto de tareas a realizar, el personal disponible y su cualificación, restricciones y preferencias existentes.

La Universidad Politécnica de Valencia indica a través de su página web que es un sistema altamente flexible, que se adapta a diversas tipologías de tareas, restricciones, criterios, etc., así como a las diferentes capacidades, turnos, disponibilidad, y preferencias del personal y la organización. También menciona que OPTIHPER cuenta con las siguientes características:

- Ser muy eficiente, capaz de realizar un gran volumen de tareas/personas en poco tiempo.
- Tiene la funcionalidad de estimar los requerimientos del personal tanto habitual como esporádico.
- Re-planificación de la asignación de tareas en caso de incidencias y cambios.
- Obtener estadísticas e informes útiles para la toma de decisiones.

OPTIHPER está implementado en ANSI C, es multiplataforma, no hace uso de rutinas o código protegido por terceros y está en idioma español (OPITHPER, 2014).

Para una idea más clara de cómo sería un escenario típico en solución mediante el software OPTIHPER ver la Imagen 2.

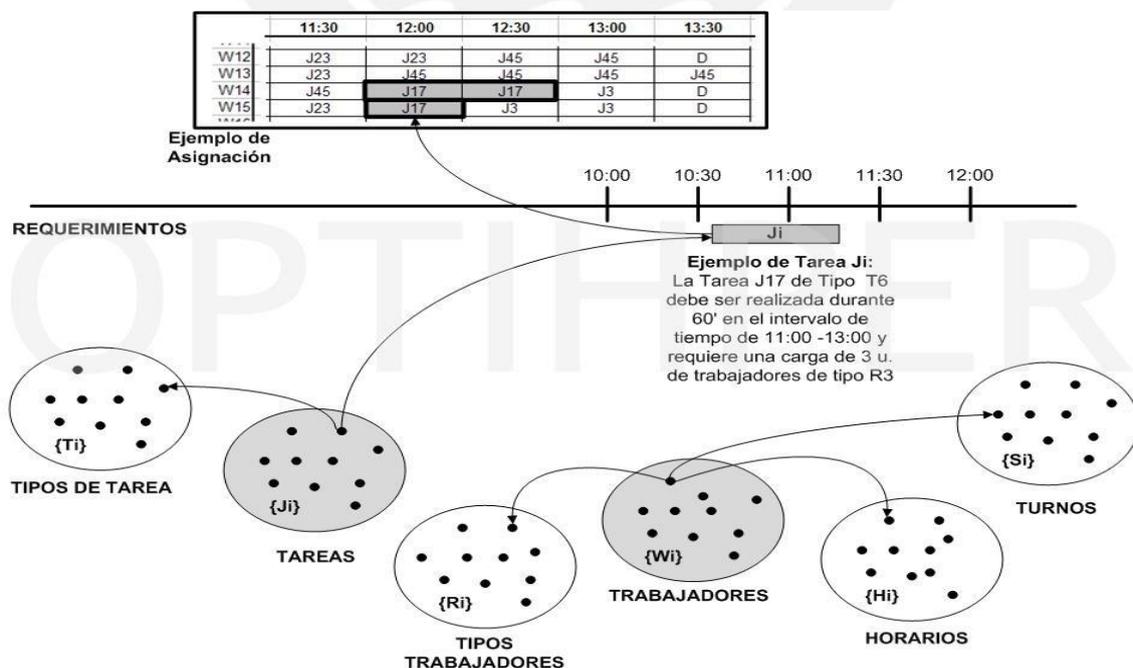


Imagen 2: Ejemplo de un escenario típico del sistema OPTIHPER. [OPTIHPER]

## 2.2. DRoster Employee Scheduling Software (KAPPIX, 2014)

Sistema para la planificación automática de horarios y tareas desarrollado por la empresa Kappix. Es muy flexible ya que se adapta a diferentes escenarios como pueden ser: empresas de producción, instituciones educativas, turnos en empresas de transporte u hospitales, etc. Tiene muchas características, entre ellas se pueden destacar:

- Permite guardar los planes de horarios creados para reutilizarlos en futuras ocasiones como si fueran plantillas.
- Ver la disponibilidad completa de los empleados, de acuerdo a como estos han sido asignados a distintas tareas. Permite conocer qué empleados están libres a tiempo parcial, cuándo y por qué están disponibles.
- Permite manejar tiempos extras y evita la sobrecarga de trabajo.
- No permite que existan conflictos entre los horarios que se arman.
- No deja empleados sin asignar.
- Tiene once tipos de reportes, cada uno de estos con la posibilidad de ver los resultados como gráficos, calendario, barras, histogramas, etc.

Cuenta con un motor basado en reglas que le permite manejar número ilimitado de turnos, así mismo el número de empleados, tareas, lugares que permite administrar es también ilimitado (KAPPIX, 2014). Tiene una vista de las tareas de cada empleado por nombre tal como se puede ver en la Imagen 3.

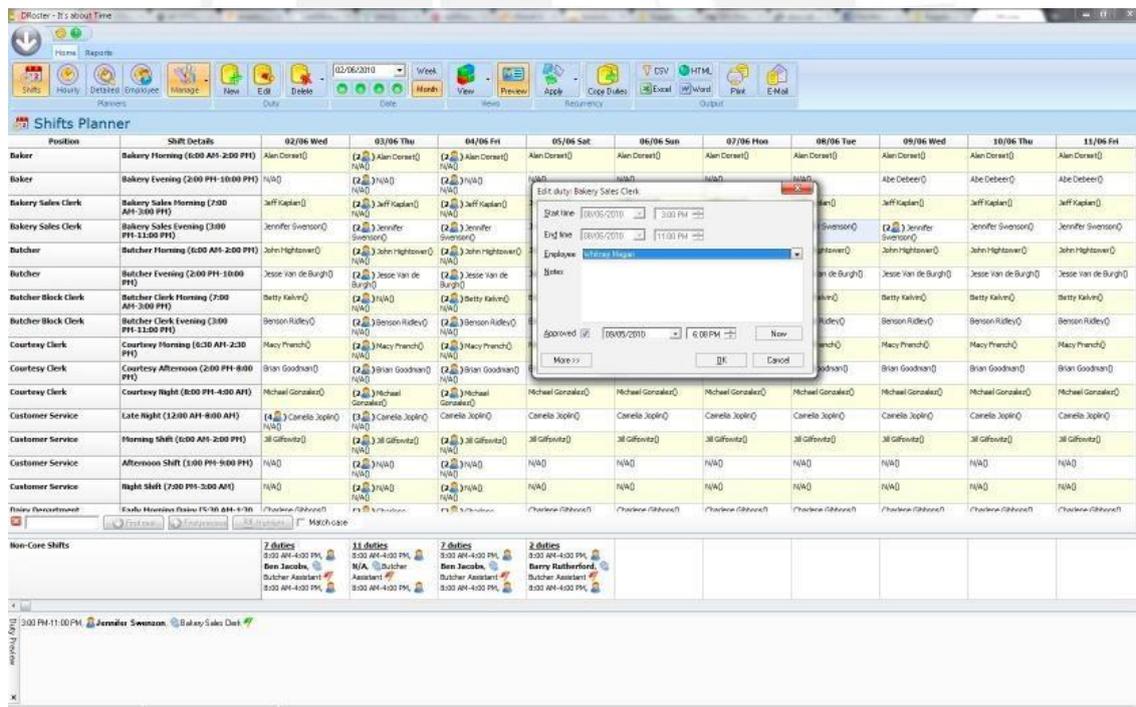


Imagen 3: Vista resumen de asignación de tareas por empleado. [Kappix]

### 2.3. aSg Horarios

Software producido por la empresa Applied Software Consultants. Empresa que cuenta con veinte años en el mercado, está presente en ciento setenta y tres países y mil quinientos centros de estudios (ASC, 2014).

Este sistema cuenta con la funcionalidad de generación automática de horarios a partir de los requisitos que se introduzca. Después de generado el horario se pueden realizar ajustes manuales. Utiliza un algoritmo que comprueba rápidamente el horario en busca de algún conflicto evitando empalmes. Según los autores, el software ha sido diseñado para que se adapte a los requisitos de cada zona del planeta.

### 2.4. GHC 17

GHC 17 es un software de escritorio para la generación de horarios para centros de enseñanza, su objetivo fundamental es encajar los horarios escolares semanales observando todas las condiciones necesarias en cada centro de enseñanza (PEÑALARA, 2015).

GHC 17 ofrece un motor de generación de horarios capaz de ofrecer diferentes soluciones que cumplan con los requisitos académicos de todas las asignaturas, el número de aulas y profesores. Ofrece encontrar nuevas soluciones a partir de cambios que ocurran.

Los componentes que conforman la funcionalidad básica de GHC 17 son: el planificador, el motor y el editor de horarios.

### 2.5. Conclusiones sobre el estado del arte

A continuación se muestra una tabla comparativa de los productos mencionados anteriormente.

Tabla 4: Tabla comparativa del estado del arte encontrado. [Elaboración propia]

Característica	OPTIHPER	KAPPIX	aSg Horarios	GHC 17
Asignación automática	X	X	X	X
Código abierto				
Plataforma web				
Prioridad por distancia				
Prioridad por conocimiento	X		X	

Después de la investigación acerca del estado del arte que se pudo realizar, se puede apreciar que en el mercado actual existen diversos productos para resolver el problema de armar un horario automático para empresas productoras o instituciones educativas

como la que es tema de este proyecto de fin de carrera. Sin embargo, estas soluciones carecen de características necesarias para este problema en particular. Estas características son:

- Prioridad de selección de acuerdo a la distancia de la persona.
- Código abierto en caso se quiera ajustar a la necesidad actual o adecuarlo a algún cambio futuro.
- Plataforma Web para repartir la función de llenado de disponibilidad a los evaluadores.

Como se aprecia en la Tabla 4, las soluciones encontradas no presentan las características requeridas.

Por lo tanto, se puede concluir que la solución que se adapta mejor a la problemática descrita antes es la que se desarrollará en este proyecto de fin de carrera ya que al no existir solución exacta, se procede a desarrollar una ad-hoc.



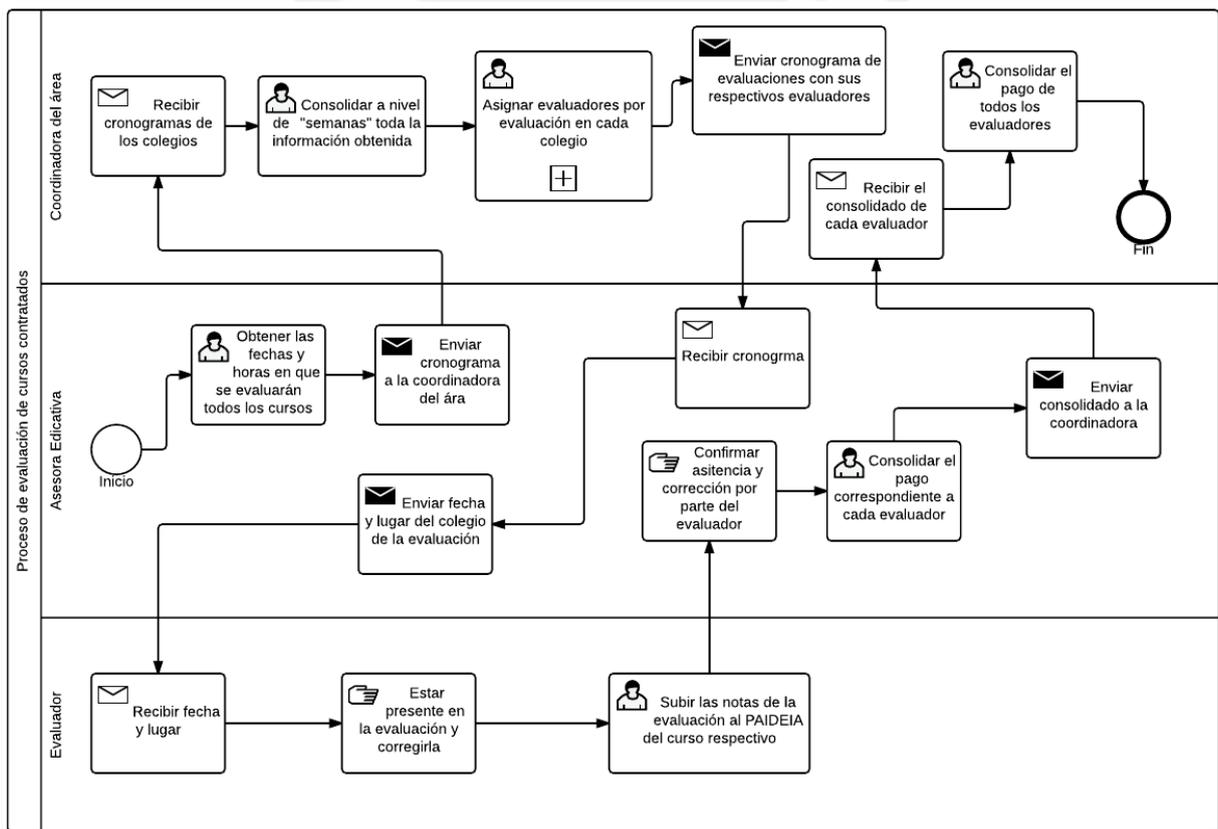
### CAPÍTULO 3

#### Modelado del proceso de evaluación

El modelado de procesos es una herramienta muy útil que permite observar gráficamente el flujo de un proceso, esto permite identificar las actividades que se realizan a lo largo del proceso, así como también a las personas que forman parte de éste. También es una herramienta importante para la transferencia de conocimiento, pues ayuda a un trabajador nuevo a entender cómo funciona determinado proceso.

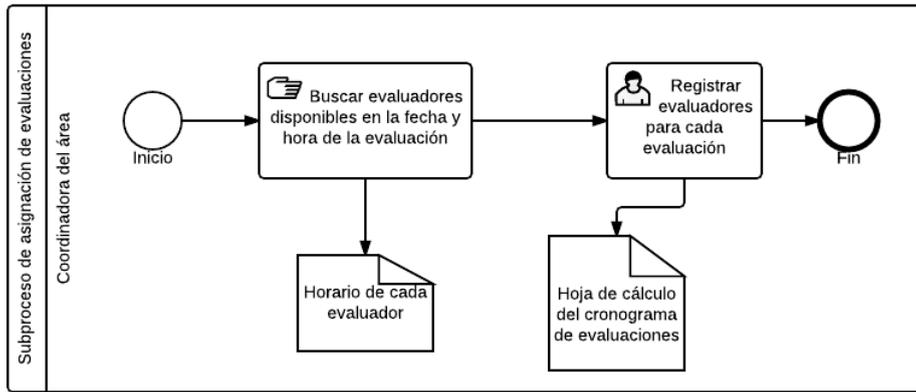
Esta solución busca dar soporte al proceso de evaluación de los cursos contratados por los colegios en convenio con InfoPUC, dicho proceso no se encuentra modelado. Dada la importancia de tener el modelado del proceso para un mejor entendimiento de éste y su utilidad para la Institución es que se vio conveniente su diseño.

Tomando como referencia lo expuesto por las personas a cargo esta actividad y lo explicado en la problemática, se realizó el modelo que se aprecia en la Imagen 4.



**Imagen 4:** Proceso de evaluación de cursos contratados. [Elaboración propia]

En la Imagen 5 se muestra el subproceso “Asignar evaluadores por evaluación en cada colegio”.



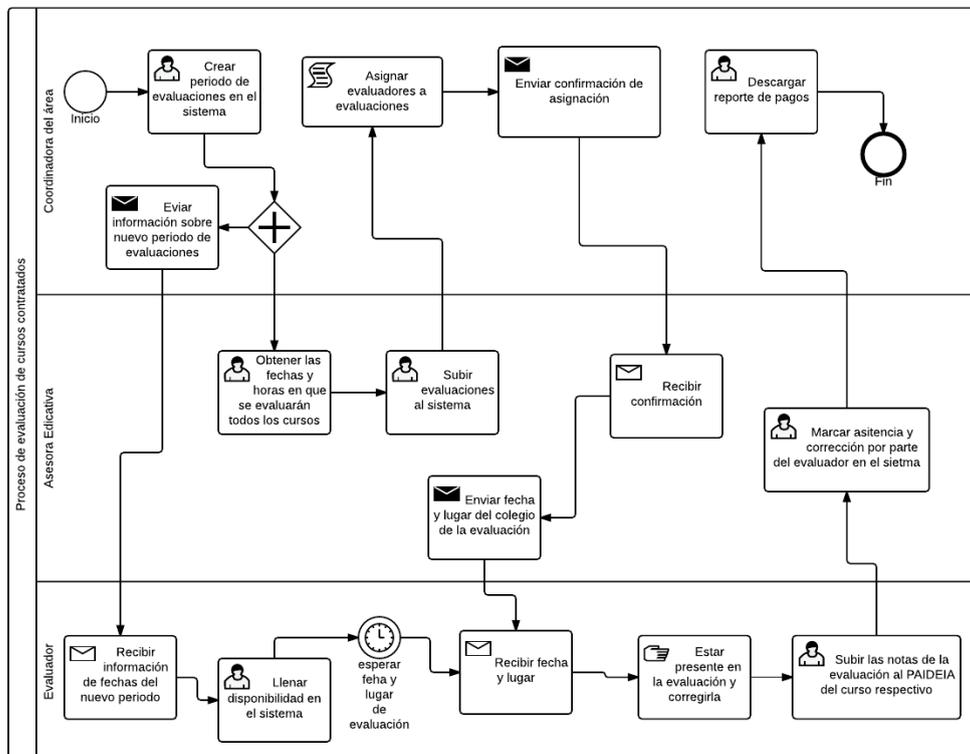
**Imagen 5:** Subproceso de asignación de evaluadores. [Elaboración propia]

Como se puede observar en las Imágenes 4 y 5, la coordinadora del área realiza las siguientes tareas manuales:

- Consolidar a nivel de semanas los cronogramas por cada colegio.
- Asignar evaluadores para cada evaluación en cada colegio.
- Consolidar el pago de todos los evaluadores.

La solución propuesta busca reducir el tiempo utilizado para este proceso. Para lograr esto, el sistema hace que la coordinadora ya no realice las tres tareas manuales anteriores.

La Imagen 6 muestra el modelado del proceso de evaluaciones actualizado con la solución actual.



**Imagen 6:** Modelo del proceso adecuado al sistema a implementar. [Elaboración propia]

## CAPÍTULO 4

## 1. Lista de requerimientos

A continuación en la Tabla 5 se presenta la lista de requisitos a implementar en el desarrollo del sistema.

**Tabla 5: Lista de requerimientos del Sistema. [Elaboración propia]**

N°	Descripción	Tipo	Exigencia	Prioridad
<b>Cursos</b>				
1	El sistema deberá permitir crear, editar y eliminar cursos.	Funcional	Exigible	1
2	El sistema podrá buscar cursos por el nombre o por la descripción.	Funcional	Exigible	1
<b>Colegios</b>				
3	El sistema deberá permitir crear, editar y eliminar colegios.	Funcional	Exigible	1
4	El sistema podrá buscar colegios por el nombre o por el distrito al que pertenece.	Funcional	Exigible	1
5	Las asesoras educativas deberán tener acceso solo a los colegios a los que hayan sido asignados.	Funcional	Exigible	1
6	El sistema permitirá visualizar los evaluadores asignados a las evaluaciones de determinado colegio por semanas.	Funcional	Exigible	1
<b>Evaluadores</b>				
7	El sistema permitirá crear, editar y eliminar evaluadores del sistema.	Funcional	Exigible	1
8	Los evaluadores deberán tener un usuario asociado para llenar su disponibilidad.	Funcional	Exigible	1
9	Las cuentas de usuario de los evaluadores podrán ser desactivadas y activadas para evitar cambios en su disponibilidad pasada la fecha límite.	Funcional	Exigible	1
10	El administrador del sistema podrá llenar la disponibilidad de cualquier evaluador.	Funcional	Exigible	1
11	El sistema debe permitir visualizar a los evaluadores que no han llenado disponibilidad.	Funcional	Exigible	1
12	Al crear un evaluador el sistema debe enviar un correo electrónico al evaluador indicando sus datos de acceso.	Funcional	Exigible	1
13	El sistema debe enviar un correo electrónico a todos los evaluadores activos indicando el nuevo periodo de evaluaciones y las fechas límite que disponen para llenar su disponibilidad.	Funcional	Exigible	1
<b>Evaluaciones</b>				

14	El sistema permitirá crear, editar y eliminar evaluaciones del sistema.	Funcional	Exigible	1
15	El sistema permitirá subir evaluaciones masivamente mediante un archivo Excel.	Funcional	Exigible	1
16	El sistema mostrará una pantalla de confirmación antes de subir las evaluaciones importadas desde el archivo Excel, si hubiese errores también los debe mostrar.	Funcional	Exigible	1
17	El sistema permitirá buscar evaluaciones por colegio, fechas de evaluaciones y estado (evaluaciones asignadas y no asignadas).	Funcional	Exigible	1
18	El sistema permitirá asignar evaluadores a evaluaciones.	Funcional	Exigible	1
19	El sistema permitirá desasignar evaluadores de evaluaciones.	Funcional	Exigible	1
20	El sistema deberá mostrar a los evaluadores más aptos para cada evaluación en la pantalla de asignación. Estos deben estar ordenados por distancia al colegio y se deben poder filtrar aquellos que tienen conocimientos relacionados con la evaluación.	Funcional	Exigible	2
<b>Usuarios</b>				
21	El sistema debe permitir crear, editar y eliminar usuarios del sistema.	Funcional	Exigible	1
22	El sistema manejará tres tipos de usuarios (administrados, asesora y evaluador).	Funcional	Exigible	1
23	El usuario administrador será el encargado de administrar usuarios, colegios, cursos, evaluadores, evaluaciones, asignar evaluadores a evaluaciones y administrar los periodos de pago.	Funcional	Exigible	1
24	El usuario asesora podrá ver las evaluaciones del colegio al que pertenece, subir evaluaciones y editarlas, pero no puede asignar evaluadores.	Funcional	Exigible	1
25	El usuario asesora podrá marcar la asistencia de un evaluador a una evaluación en la pantalla “marcar asistencia” ubicada en la administración de colegios.	Funcional	Exigible	1
26	El usuario evaluador sólo tiene acceso al sistema durante un periodo de tiempo. Podrá marcar su disponibilidad de tiempo y modificar su información de perfil (cambiar contraseña, nombre de usuario y dirección).	Funcional	Exigible	1
<b>Pagos</b>				
27	El sistema debe permitir administrar el monto a pagar a los evaluadores dentro de un periodo de evaluaciones.	Funcional	Exigible	1
28	El sistema debe permitir crear un periodo de	Funcional	Exigible	1

	evaluaciones.			
28	El sistema permitirá al administrador del sistema marcar la asistencia de cada evaluador en un periodo de evaluaciones.	Funcional	Exigible	1
29	El sistema permitirá generar un reporte sobre los pagos a realizar a todos los evaluadores dentro de un periodo de evaluaciones.	Funcional	Exigible	1
30	El sistema permitirá generar un reporte detallado del pago de cada evaluador.	Funcional	Exigible	1
31	El sistema debe permitir enviar un correo a cada evaluador con su reporte de pago.	Funcional	Exigible	2

## 2. Arquitectura de la solución

En este apartado se presenta la arquitectura a emplear en el sistema y los componentes por capas. Para ello se usarán diagramas UML realizados en la herramienta StarUML, estos diagramas servirán para apreciar los componentes y cómo es que están organizados.

### 2.1. Términos a utilizar

- **Apache:** Servidor web de código abierto que funciona en diferentes plataformas Unix y Windows. Se caracteriza por no necesitar una configuración avanzada para comenzar a utilizarlo.
- **Yii Framework:** Framework PHP basado en componentes de alta performance para desarrollar aplicaciones Web de gran escala usando el patrón Modelo Vista Controlador.
- **PHP:** Lenguaje de programación utilizado para la programación del lado del servidor.
- **JavaScript:** Lenguaje de programación usado para la programación del lado del cliente, se compila en el navegador web.
- **AngularJS:** Framework de JavaScript de código abierto desarrollado por Google que permite ejecutar código en el navegador e interactuar asincrónicamente con la vista y el modelo.
- **Modelo Vista Controlador (MVC):** Patrón de diseño de arquitectura de software que separa la lógica del negocio y los datos presentes en una aplicación.

### 2.2. Representación de la arquitectura

Parte de la reducción de tiempo en este proceso es gracias a la descentralización del flujo de trabajo en este proceso. Para poder lograr esto, los mismos evaluadores llenan su disponibilidad de tiempo y cada asesora educativa sube las evaluaciones de los colegios con los que trabaja. Por estos motivos es que opta por un sistema web, pues permitirá realizar lo mencionado anteriormente sin la necesidad de que los evaluadores tengan el sistema instalado en sus computadoras.

La solución brindada está basada en la arquitectura de n-capas, en esta solución se empleó las siguientes tres capas:

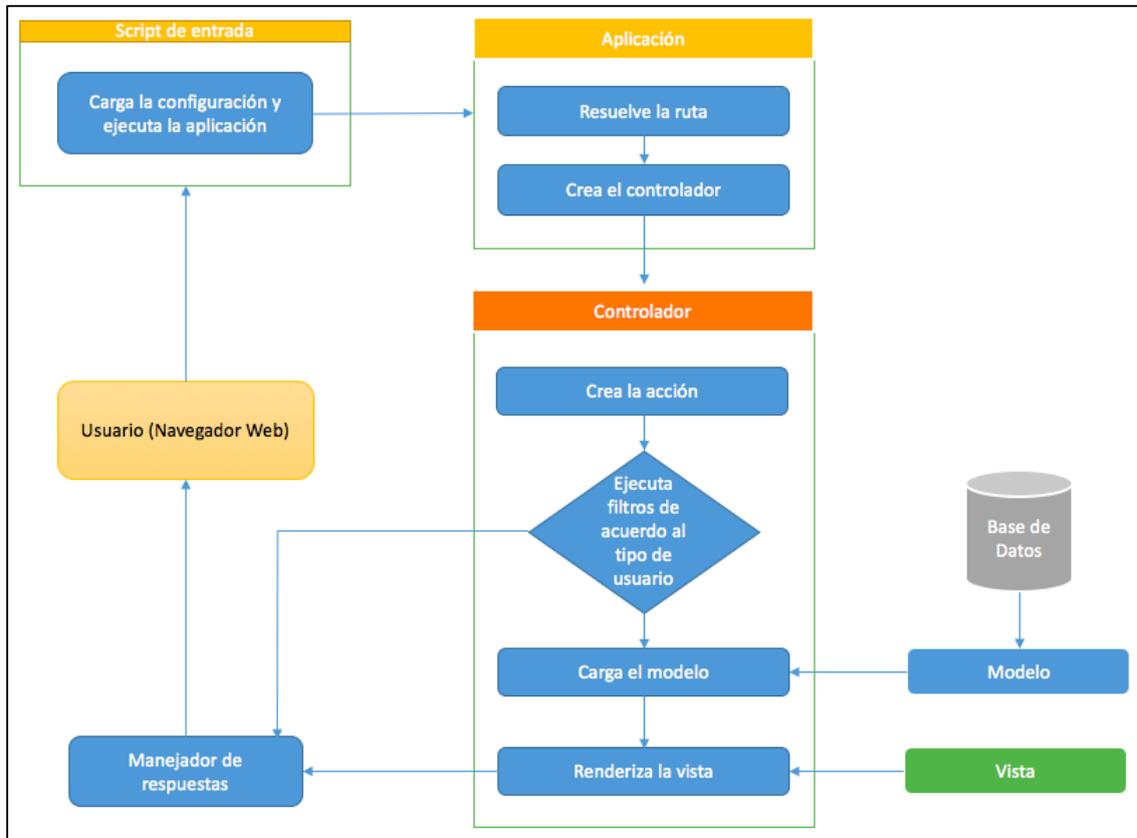
- **Capa de Presentación:** Esta capa está compuesta por el navegador web con el que el usuario interactúa con el sistema. La función de los navegadores es hacer las peticiones hacia el servidor para que éste provea las páginas y sean mostradas al usuario.
- **Capa de Negocio:** La parte principal de la arquitectura del sistema, ya que es en esta capa donde se implementan los procesos que harán funcionar la aplicación a desarrollar. Para la implementación de las clases que representan los maestros de información del sistema se hará uso del lenguaje de programación PHP, mientras que para hallar a los evaluadores más cercanos al colegio donde se da una evaluación se usará el lenguaje JavaScript. Así mismo para la obtención de los datos de la Capa de Datos se usarán clases escritas en PHP.
- **Capa de Datos:** Es la encargada de la persistencia de los datos, para este proyecto se usará un servidor de base de datos MySQL, base de datos relacional que se complementa perfectamente con PHP y es de libre uso. En esta capa se guardarán los datos correspondientes a los maestros de información y el log de actividades del sistema.

Como patrón de diseño de arquitectura se usará Modelo Vista Controlador, uno de los más usados en desarrollo web ya que permite separar la lógica de la aplicación de la interfaz gráfica y de los datos.

Los componentes de este patrón son los siguientes:

- **Modelo:** Componente responsable de acceder a los datos con los que el sistema trabaja.
- **Vista:** Es el componente encargado de representar la interfaz mediante la cual el usuario podrá interactuar con los datos.
- **Controlador:** Este componente responde a eventos generados en la vista (interacción del usuario con la vista), se encarga de obtener datos a través del modelo y devolverlos a la vista.

La Imagen 7 describe el funcionamiento del sistema web realizado siguiendo el patrón Modelo Vista Controlador.



**Imagen 7:** Modelo Vista Controlador en una aplicación web. [Elaboración propia]

### 2.3. Vista Lógica

La vista lógica de la arquitectura refleja los requisitos funcionales, la interacción entre el sistema y el usuario. Para el desarrollo del sistema se usará Yii Framework 1.4, tanto en la administración de rutas como para la interacción con la base de datos por medio del ORM que provee Yii.

Para definir la vista lógica del sistema se utilizan paquetes, estos representan los componentes del patrón de arquitectura MVC, estos paquetes son los siguientes:

- **Paquete de interfaz web:** Representa a la vista, este paquete contiene todas las páginas HTML que son creadas para la interface con el usuario. También se encuentran las páginas creadas con JavaScript a través del Framework AngularJS.
- **Paquete de lógica del negocio:** Representa al modelo, este paquete contiene las clases que interactúan con la base de datos y son instanciados desde las clases del controlador.
- **Paquete de servicios del negocio:** Representa al controlador, en este paquete se encuentran todos los servicios que usará la interfaz web.

La Imagen 8 muestra el diagrama UML de la vista de despliegue

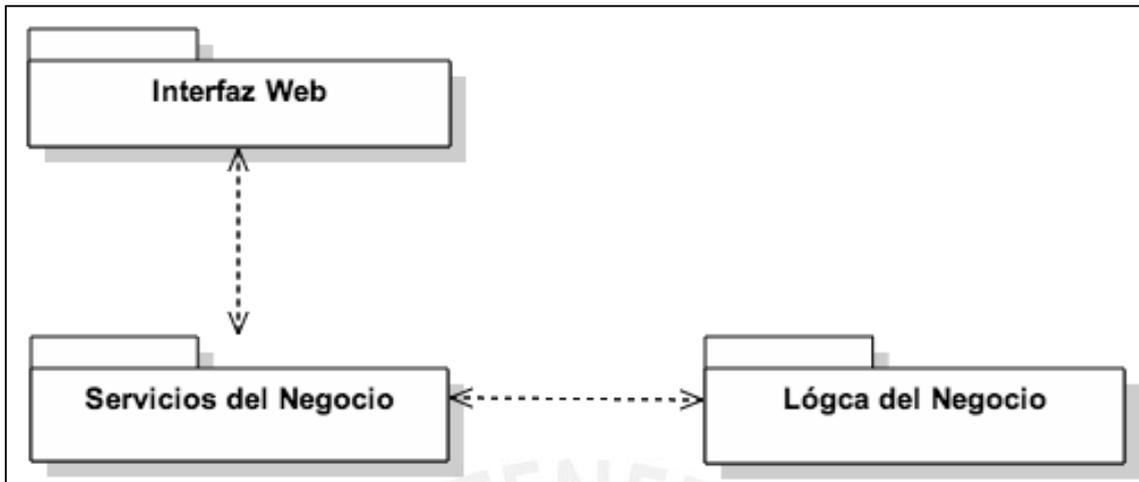


Imagen 8: Vista Lógica. [Elaboración propia]

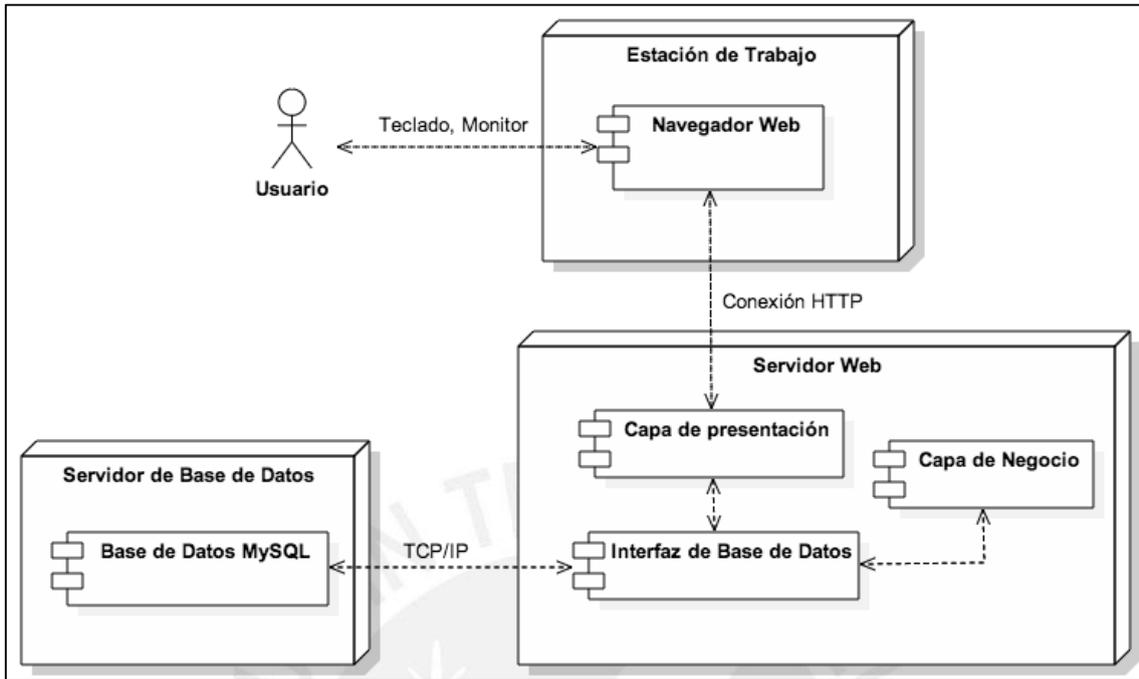
## 2.4. Vista de Despliegue

El navegador web será el encargado de compilar el código HTML o el código PHP para transformarlo en HTML que se enviará desde el servidor web cuando un cliente haga una petición a través de la url.

El servidor web será el encargado de almacenar las páginas del sistema y la lógica del negocio que será traducida en código HTML para ser apreciado gráficamente por el usuario por medio del navegador web.

El servidor de base de datos será el encargado de almacenar toda la información referente a las entidades del sistema.

La Imagen 9 muestra el diagrama UML de la vista de despliegue del sistema.

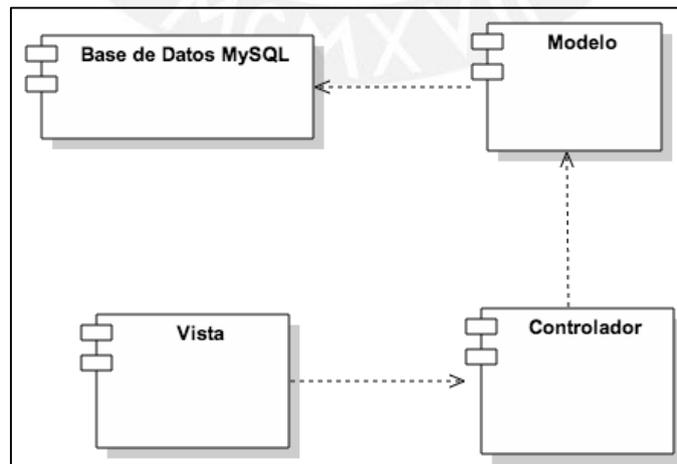


**Imagen 9:** Vista de Despliegue. [Elaboración propia]

## 2.5. Vista de Implementación o Desarrollo

En esta vista se muestra la organización del código a través de componentes. Los niveles a utilizar son las capas propuestas por el patrón de diseño Modelo Vista Controlador y un componente para la persistencia de datos.

Estos componentes se muestran en la Imagen 10.



**Imagen 10:** Vista de Implementación. [Elaboración propia]

## 2.6. Metas y Restricciones de la arquitectura

Es importante mencionar que la arquitectura a desarrollar cuenta con las siguientes metas y restricciones.

### 26.1. Metas

- El sistema debe ser implementado siguiendo el modelo cliente-servidor.
- El sistema permitirá el acceso a todos los usuarios que cuenten con un navegador web en su equipo de trabajo.
- El sistema debe ser accedido las 24 horas del día.
- El sistema debe ser utilizado desde dispositivos móviles.
- El sistema debe ser usado en los navegadores Google Chrome, Mozilla Firefox, Internet Explorer 9 hacia adelante y Safari.

### 26.2. Restricciones

- El motor de base de datos a usar debe ser MySQL.
- El sistema debe ser desarrollado con el Marco de Programación Yii en su versión 1.4.
- Los usuarios deberán tener acceso a internet para poder hacer uso del sistema.
- Los usuarios deben usar navegadores web que soporten el uso de JavaScript, en la actualidad todos los navegadores en sus versiones más recientes.

## 3. Estándares de la interfaz gráfica

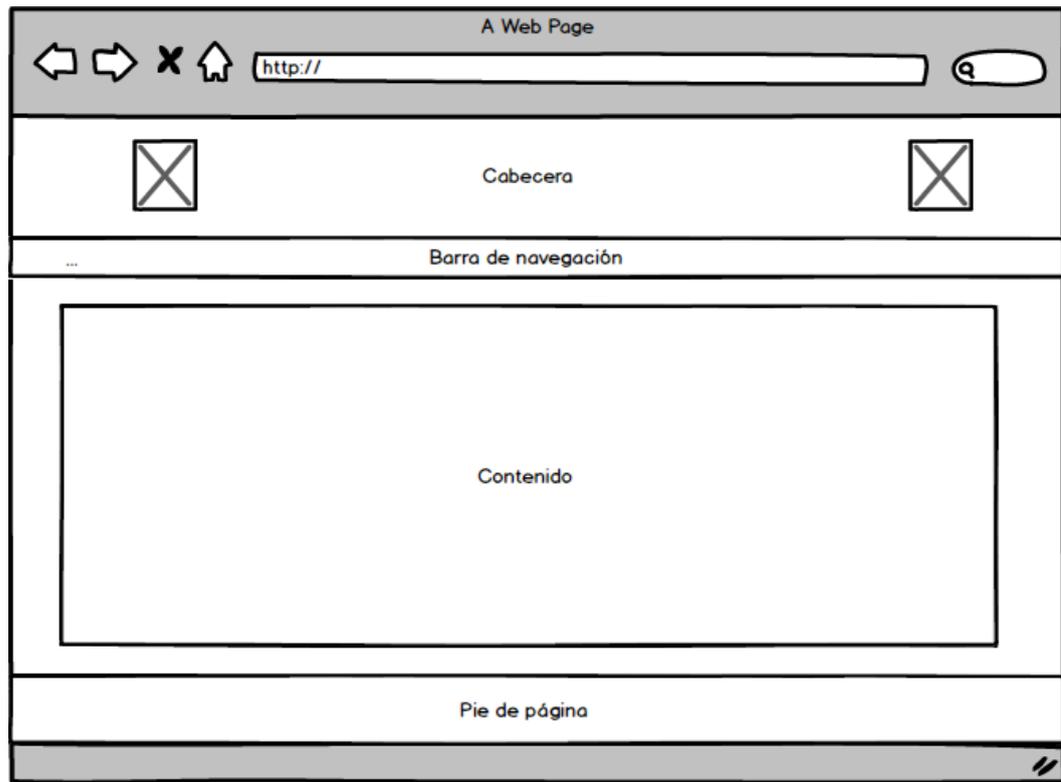
Se han definido estándares generales para el desarrollo del prototipo del sistema, a continuación se detallará cada uno de los estándares seguidos.

### 3.1. Estructura de la plantilla

Todos los módulos siguen un diseño de plantilla, el cual está conformado por las siguientes partes:

- Cabecera: Muestra el nombre de la institución al lado izquierdo y al lado derecho el logo de la institución.
- Barra de navegación: Muestra los enlaces a cada uno de los módulos del sistema y el botón para salir del sistema.
- Contenido: Espacio designado para mostrar el contenido de cada uno de los módulos del sistema y las acciones de cada uno de estos.
- Pie de página: Muestra información general de la institución (nombre, dirección y año)

La Imagen 11 muestra lo descrito anteriormente.



**Imagen 11:** Estructura del contenido del sistema. [Elaboración propia]

### 3.2. Mensajes de confirmación de acciones

Dado que es un sistema web, el cuadro del mensaje de confirmación varía dependiendo del navegador, el texto sí es propio a la acción a realizar. La Imagen 12 muestra el mensaje de confirmación en el navegador Chrome.



**Imagen 12:** Mensaje de confirmación. [Elaboración propia]

### 3.3. Mensajes al usuario

Para los mensajes al usuario se usará el estilo definido por el marco de diseño Bootstrap. La Tabla 6 muestra las clases a usar para cada tipo de mensaje.

Tabla 6: Clases usadas para los mensajes. [Elaboración propia]

Tipo de mensaje	Clases
Confirmación	alert alert-success
Información	alert alert-info
Error	alert alert-danger

### 3.4. Botones

Para los botones también se usará el estilo de Bootstrap. La Tabla 7 muestra las clases a usar para cada los distintos botones a usar.

Tabla 7: Clases usadas para los botones. [Elaboración propia]

Botón	Clases
Ver	glyphicon glyphicon-eye-open
Editar	glyphicon glyphicon-pencil
Eliminar	glyphicon glyphicon-trash
Acciones de módulos	btn btn-default
Envío de formulario	btn btn-primary

### 3.5. Colores

Los colores a seguir para el desarrollo de la interfaz gráfica se presentan en la Tabla 8.

Tabla 8: Colores usados en la interfaz. [Elaboración propia]

Botón	Color (RBG)
Fondo	255,255,255
Fuente	0,0,0
Fondo de la barra de navegación	0,117,200
Letras de la barra de navegación	0,0,0
Fondo del pie de página	51,51,51
Letras del pie de página	153,153,153

## CAPÍTULO 5

### 1. Módulos del sistema

Después de haber realizado la etapa de análisis, se ha definido que el sistema debe contener los siguientes módulos:

- Módulo de usuarios: Este módulo se usa para hacer el mantenimiento de los usuarios que harán uso del sistema. La Imagen 13 muestra los casos de uso que se realizan en el presente módulo.

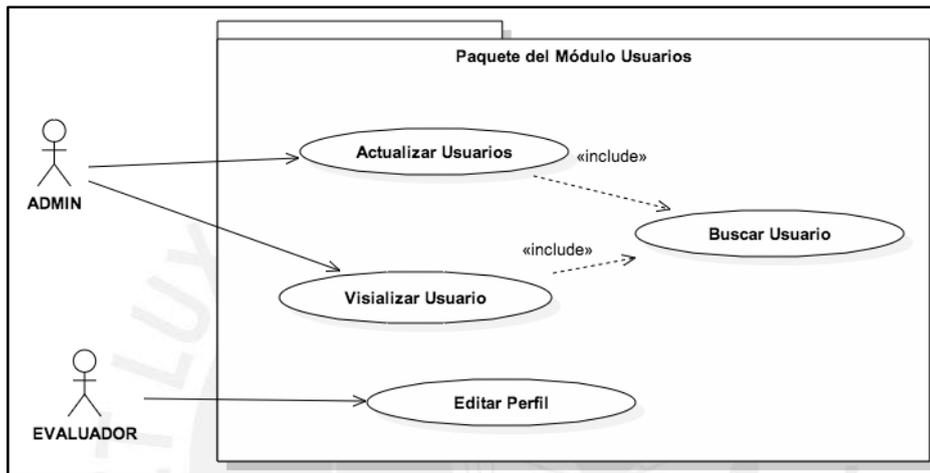


Imagen 13: Casos de uso del módulo de Usuarios. [Elaboración propia]

- Módulo de cursos: Este módulo se usa para hacer el mantenimiento de los cursos que se registran en el sistema. La Imagen 14 muestra los casos de uso que se realizan en el presente módulo.

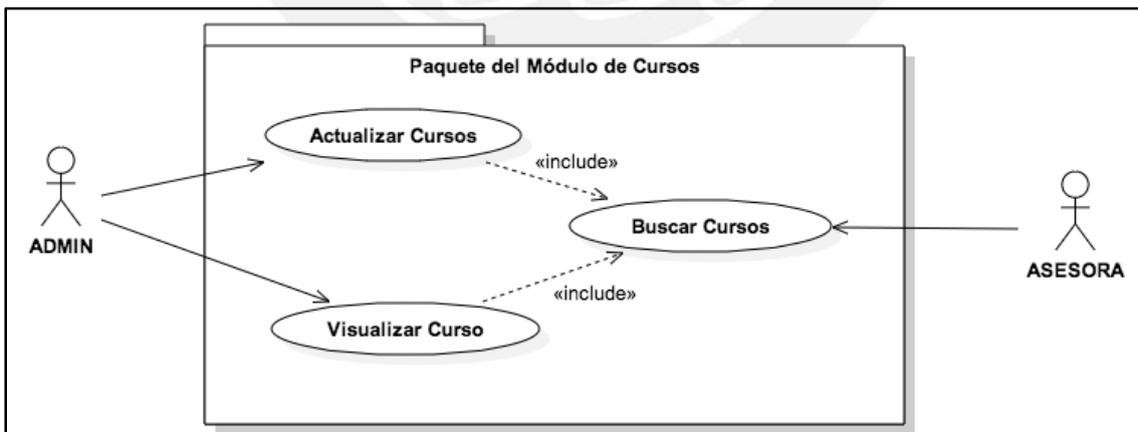


Imagen 14: Casos de uso del módulo de cursos. [Elaboración propia]

- Módulo de colegios: Este módulo se usa para hacer el mantenimiento de los colegios que se registran en el sistema. La Imagen 15 muestra los casos de uso que se realizan en el presente módulo.

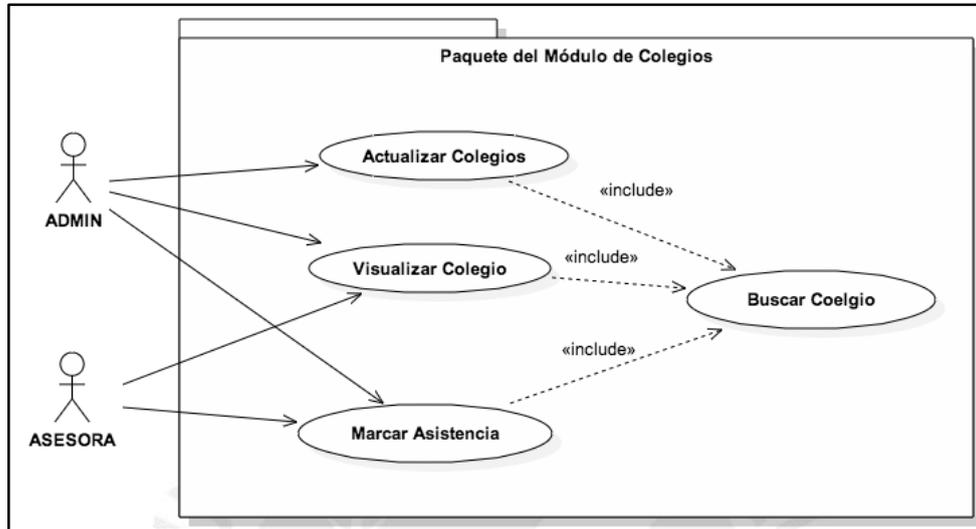


Imagen 15: Casos de uso del módulo de colegios. [Elaboración propia]

- Módulo de evaluadores: Este módulo se usa para hacer el mantenimiento de los evaluadores que se registran en el sistema. La Imagen 16 muestra los casos de uso que se realizan en el presente módulo.

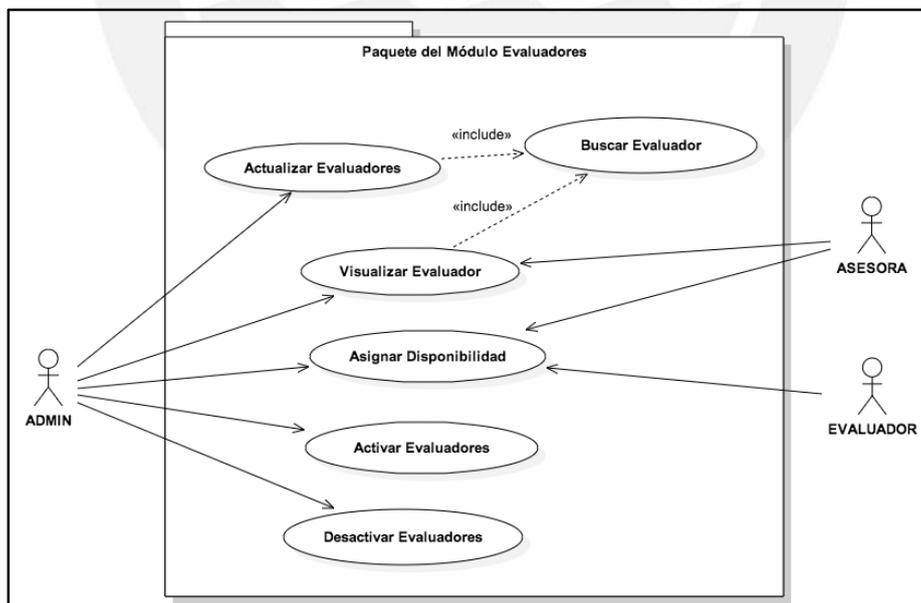


Imagen 16: Casos de uso del módulo de Evaluadores. [Elaboración propia]

- M to de las evaluaciones que se registren en el sistema. La Imagen 17 muestra los casos de uso que se realizan en el presente módulo.

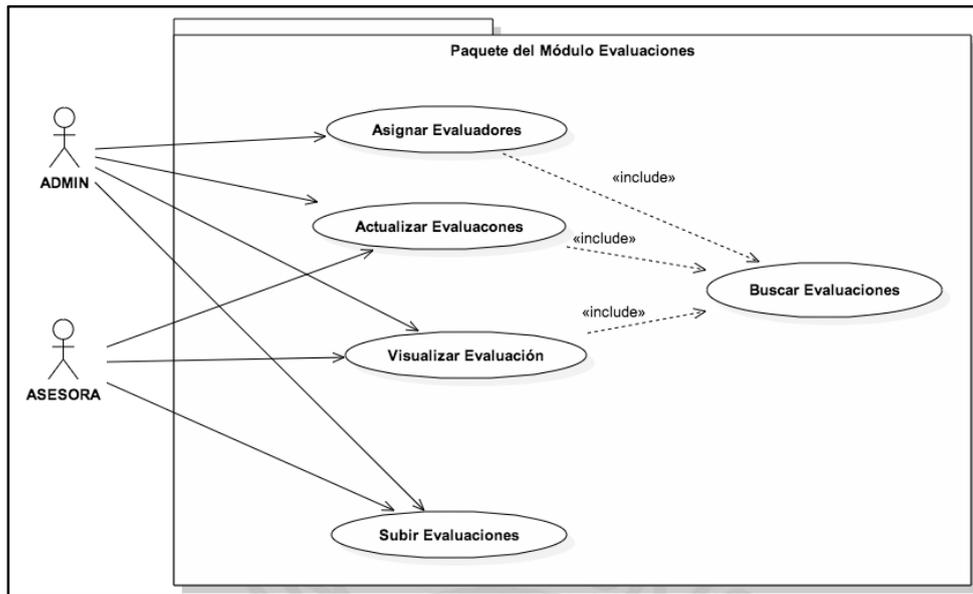


Imagen 17: Casos de uso del módulo de Evaluaciones. [Elaboración propia]

- Módulo de pagos: Este módulo se usa para hacer gestionar los pagos de los evaluadores. La Imagen 18 muestra los casos de uso que se realizan en el presente módulo.

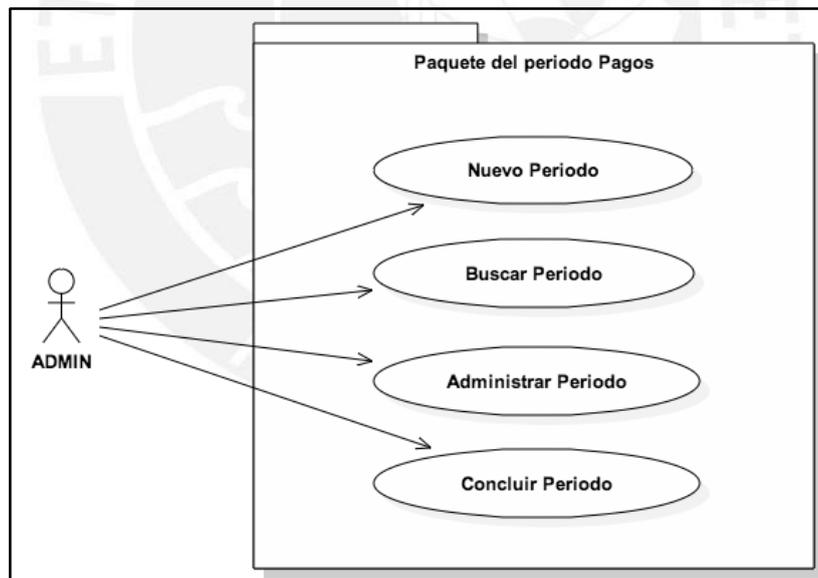


Imagen 18: Casos de uso del módulo de pagos. [Elaboración propia]

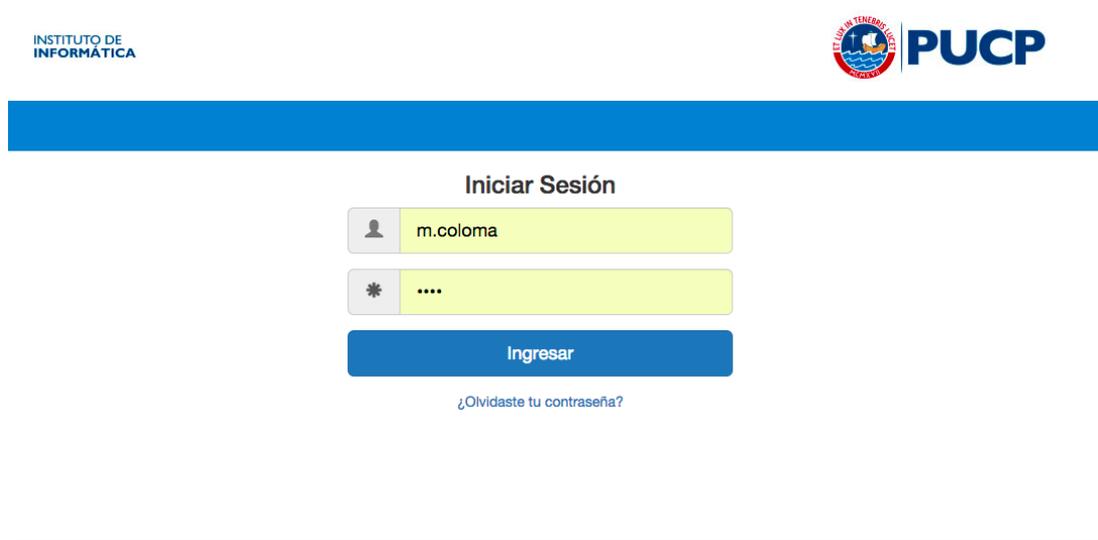
El prototipo del sistema debe contener cada uno de estos módulos, debe mostrar la interfaz donde se implementarán las funcionalidades de cada uno de estos módulos.

## 2. Prototipo funcional del sistema

En base a los estándares de interfaz gráfica definidos y la especificación de casos de uso del Anexo ERS se han construido el prototipo funcional del sistema.

A continuación se muestran las principales imágenes del sistema, el resto se presentará en el Anexo Prototipos del Sistema.

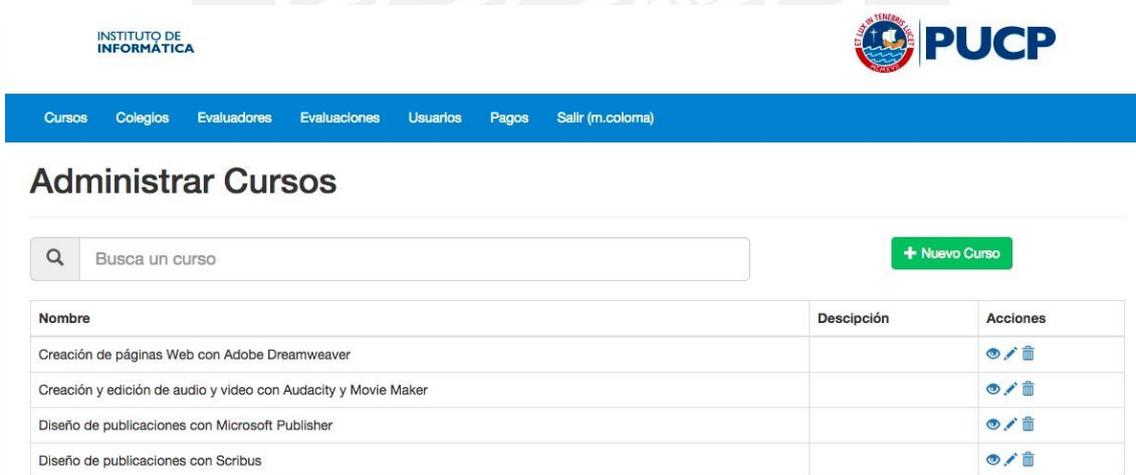
- **Ingreso al sistema**



**Imagen 19:** Ingreso al sistema. [Elaboración propia]

- **Cursos**

Pestaña diseñada para añadir las funcionalidades del módulo de cursos. La Imagen 20 muestra la pestaña de administración.

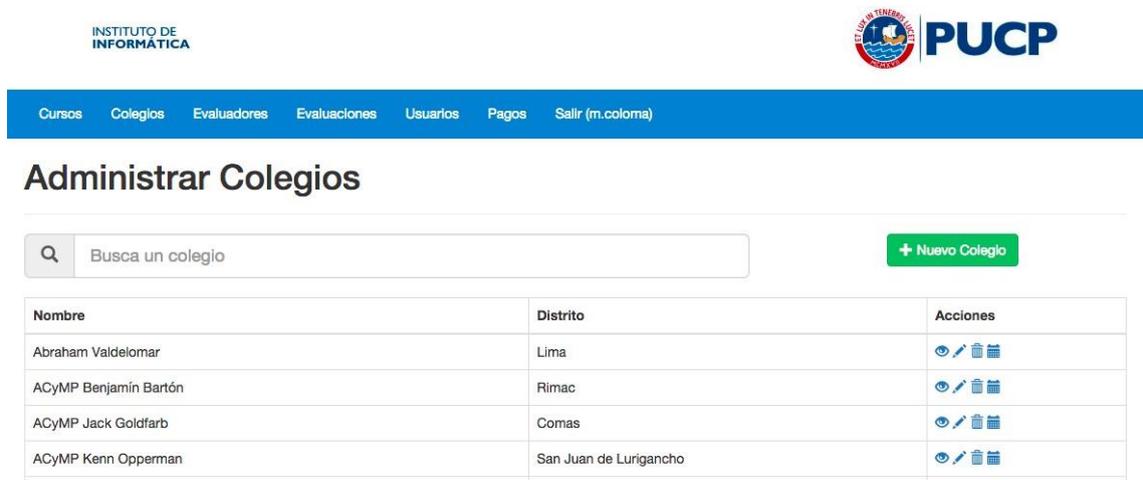


**Imagen 20:** Pestaña para administrar cursos. [Elaboración propia]

Como se aprecia en la Imagen 20, se cuenta con una tabla con paginación que muestra los cursos de 10 en 10. Debajo del título “Administrar Cursos” se tiene una caja de texto que permite buscar un curso ya sea por nombre o por descripción. Al lado derecho de cada curso estarán las opciones de visualización, edición y eliminación. Las imágenes que muestran los formularios para agregar y editar se pueden apreciar en el Anexo Prototipos del Sistema.

- Colegios

Pestaña diseñada para añadir las funcionalidades del módulo de colegios. La Imagen 21 muestra la pestaña de administración.

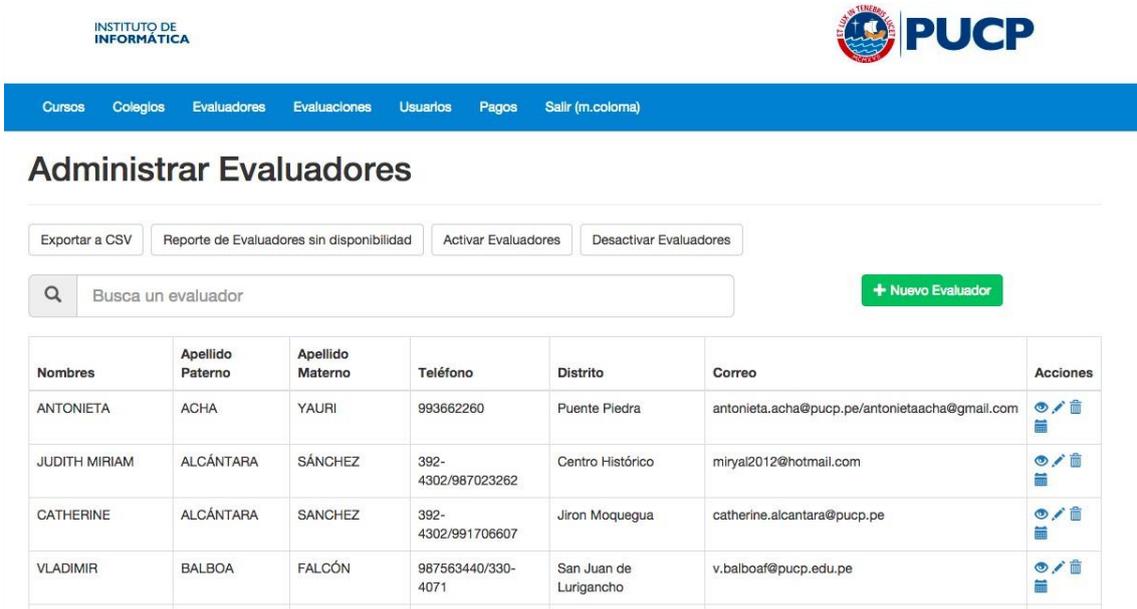


**Imagen 21:** Pestaña para administrar colegios. [Elaboración propia]

Al igual que el módulo de cursos, se cuenta con una tabla con paginación que muestra los colegios registrados de 10 en 10. Debajo del título “Administrar Colegios” se tiene una caja de texto que permite buscar un curso ya sea por nombre o por distrito. Al lado derecho de cada colegio estarán las opciones de visualización, edición, eliminación y llenado de asistencia. Las imágenes que muestran los formularios para agregar y editar se pueden apreciar en el Anexo Prototipos del Sistema.

## ▪ Evaluadores

Pestaña diseñada para añadir las funcionalidades del módulo de evaluadores. La Imagen 22 muestra la pestaña de administración.



Nombres	Apellido Paterno	Apellido Materno	Teléfono	Distrito	Correo	Acciones
ANTONIETA	ACHA	YAURI	993662260	Puente Piedra	antonieta.acha@pucp.pe/antonietaacha@gmail.com	  
JUDITH MIRIAM	ALCÁNTARA	SÁNCHEZ	392-4302/987023262	Centro Histórico	miryal2012@hotmail.com	  
CATHERINE	ALCÁNTARA	SANCHEZ	392-4302/991706607	Jiron Moquegua	catherine.alcantara@pucp.pe	  
VLADIMIR	BALBOA	FALCÓN	987563440/330-4071	San Juan de Lurigancho	v.balboaf@pucp.edu.pe	  

**Imagen 22:** Pestaña para administrar evaluadores. [Elaboración propia]

Al igual que los módulos de cursos y colegios, se cuenta con una con paginación que muestra los evaluadores registrados de 10 en 10. Debajo del título “Administrar Evaluadores” se tiene una caja de texto que permite buscar un evaluador por cualquiera de los campos de la tabla. Al lado derecho de cada evaluador estarán las opciones de visualización, edición, eliminación y asignación de horas disponibles. Las imágenes que muestran los formularios para agregar y editar se pueden apreciar en el Anexo Prototipos del Sistema.

Cada evaluador podrá llenar su disponibilidad, para ello cuenta con un calendario de marcado de disponibilidad. La Imagen 23 muestra la ventana que se cuenta para realizar esta acción.

Cursos
Colegios
Evaluadores
Evaluaciones
Usuarios
Pagos
Salir (m.coloma)

### Asignar Disponibilidad al Evaluador : ANTONIETA ACHA

<< Semana anterior
Semana Siguiente >>

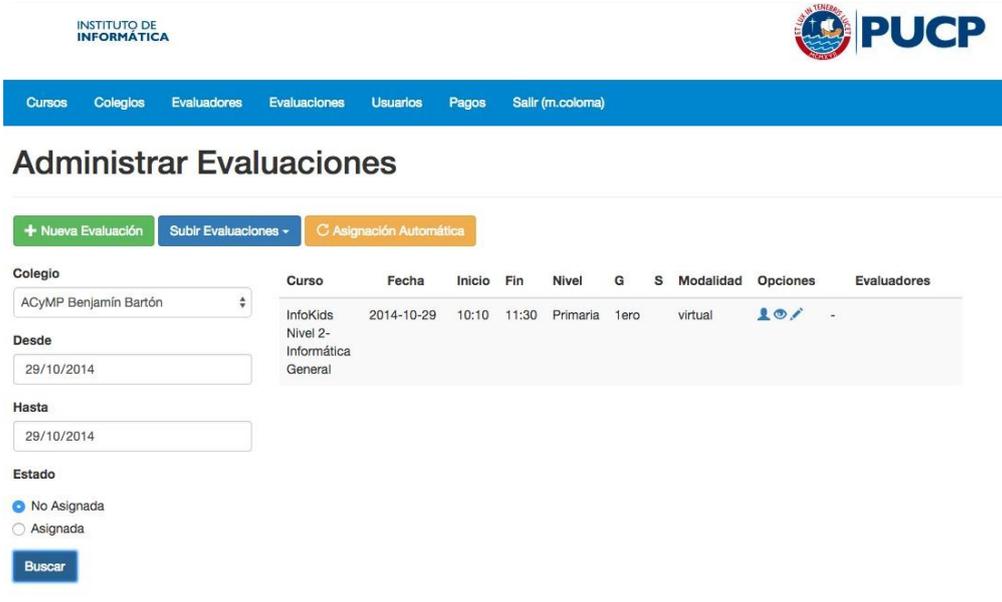
Hora	Lun 13/04/2015	Mar 14/04/2015	Mié 15/04/2015	Jue 16/04/2015	Vie 17/04/2015
7:00 - 8:00	<input type="checkbox"/> No asignado				
8:00 - 9:00	<input type="checkbox"/> No asignado				
9:00 - 10:00	<input type="checkbox"/> No asignado				
10:00 - 11:00	<input type="checkbox"/> No asignado				
11:00 - 12:00	<input type="checkbox"/> No asignado				
12:00 - 13:00	<input type="checkbox"/> No asignado				
13:00 - 14:00	<input type="checkbox"/> No asignado				
14:00 - 15:00	<input type="checkbox"/> No asignado				
15:00 - 16:00	<input type="checkbox"/> No asignado				
16:00 - 17:00	<input type="checkbox"/> No asignado				
17:00 - 18:00	<input type="checkbox"/> No asignado				
18:00 - 19:00	<input type="checkbox"/> No asignado				

Asignar Disponibilidad

**Imagen 23:** Ventana para asignar disponibilidad de los evaluadores. [Elaboración propia]

## ▪ Evaluaciones

Pestaña diseñada para añadir las funcionalidades del módulo de evaluaciones. La Imagen 24 muestra la pestaña de administración.



INSTITUTO DE INFORMÁTICA

ET LUX IN TENEBRIS LUCET

PUCP

Cursos Colegios Evaluadores Evaluaciones Usuarios Pagos Salir (m.coloma)

### Administrar Evaluaciones

+ Nueva Evaluación Subir Evaluaciones - Asignación Automática

Colegio: ACyMP Benjamín Bartón

Desde: 29/10/2014

Hasta: 29/10/2014

Estado:
   
 No Asignada
   
 Asignada

Buscar

Curso	Fecha	Inicio	Fin	Nivel	G	S	Modalidad	Opciones	Evaluadores
InfoKids Nivel 2- Informática General	2014-10-29	10:10	11:30	Primaria	1ero		virtual	  	-

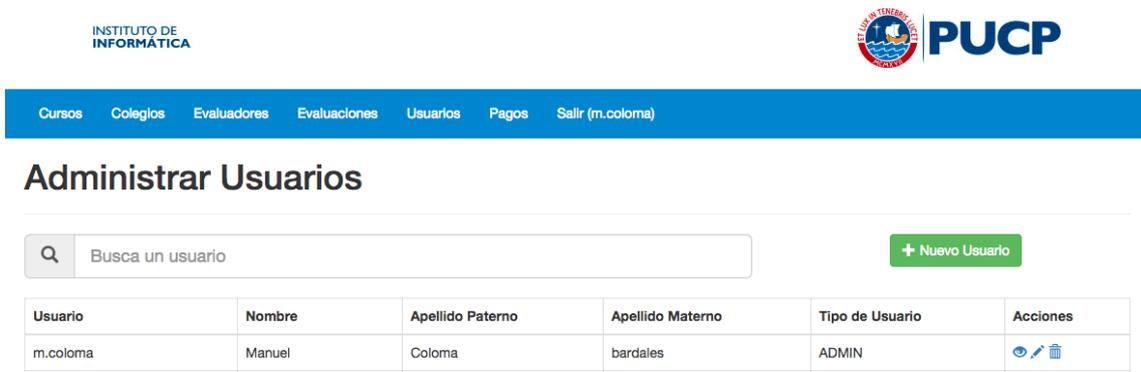
**Imagen 24:** Búsqueda de una evaluación. [Elaboración propia]

Debajo del título “Administrar Evaluaciones” se tiene el botón para agregar una evaluación y el botón para subir evaluaciones de forma masiva. Se cuenta con un formulario de búsqueda de evaluaciones.

Al realizar una búsqueda se muestra una tabla con los resultados y al lado derecho de cada evaluación se tiene los botones para asignar evaluadores, ver el detalle de la evaluación, editar y eliminar. Las siguientes imágenes que muestran la ventana para añadir evaluaciones en masa y la ventana de asignación de evaluadores se pueden apreciar en el Anexo Prototipos del sistema.

## ▪ Usuarios

Pestaña diseñada para añadir las funcionalidades del módulo de usuarios. La Imagen 25 muestra la pestaña de administración.

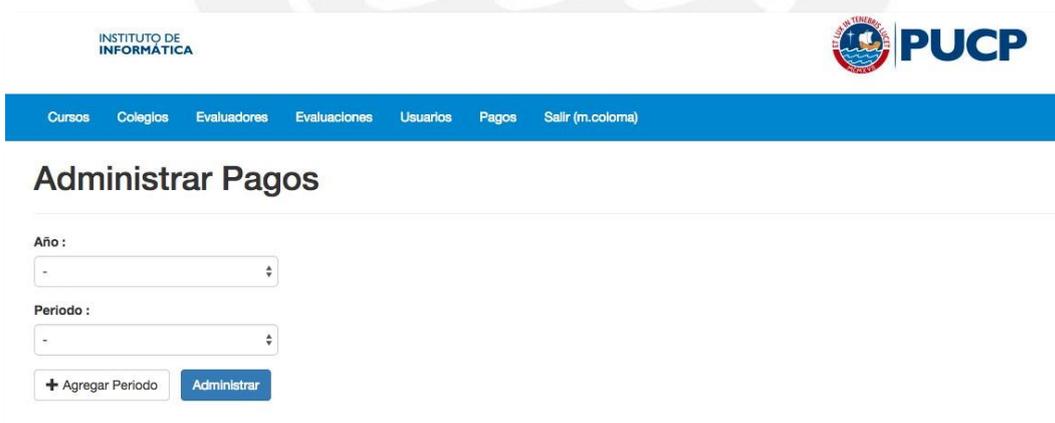


**Imagen 25:** Pestaña para administrar usuarios. [Elaboración propia]

Debajo del título “Administrar Usuarios” se tiene una caja de texto que permite buscar usuarios por cualquiera de los campos de la tabla y al costado se tiene el botón para agregar un nuevo usuario. Al lado derecho de cada usuario se pueden acceder a las opciones para ver el detalle, editar y eliminar. Las imágenes que muestran los formularios para agregar y editar se pueden apreciar en el Anexo Prototipos del Sistema.

## ▪ Pagos

Pestaña diseñada para añadir las funcionalidades del módulo de pagos. La Imagen 26 muestra la pestaña de administración.



**Imagen 26:** Ventana para administrar los pagos. [Elaboración propia]

Debajo del formulario de búsqueda de periodo de pagos se encuentra el botón para agregar un periodo de pagos. La Imagen 27 muestra el formulario para dicha funcionalidad.

INSTITUTO DE INFORMÁTICA

PUCP

Cursos Colegios Evaluadores Evaluaciones Usuarios Pagos Salir (m.coloma)

## Agregar Periodo de Pagos

Datos del Periodo de evaluaciones:

Año  
Año del periodo

Bimestre  
Bimestre

Desde  
dd/mm/aaaa

Hasta  
dd/mm/aaaa

Penalizaciones:

Por Falta  
S/. 75.00

Por Tardanza  
S/. 20.00

Por No Corrección  
S/. 40.00

Crear Periodo

**Imagen 27:** Ventana para crear un periodo de pagos. [Elaboración propia]

### 3. Gestión de pagos

A continuación se presenta el desarrollo del módulo que permite gestionar el pago a cada evaluador.

#### 3.1. Periodo de evaluaciones

La evaluación de un curso se realiza entre cuatro y cinco veces al año. Para gestionar las evaluaciones de todos los cursos contratados por los colegios en convenio se crea un periodo de evaluaciones. Este periodo se define con un nombre de periodo y un rango de fechas.

Por ejemplo para el periodo 2015-1 que va desde el 06/04/2015 hasta el 24/04/2015 se tiene que hacer el pago a cada evaluador que haya asistido a las evaluaciones pertenecientes a este rango de fechas. El pago se realiza por día de trabajo y está sujeto a penalidades por tardanza, falta y no corrección.

Para hacer esta gestión se ha creado una página que permite crear periodos de evaluaciones y se utiliza la pestaña de administración de pagos del módulo Pagos.

La Imagen 27 se muestra el formulario para la creación de un periodo.

#### 3.2. Administración de pagos

Para descentralizar la gestión de pagos, tanto la persona que administra el sistema como las asesoras educativas pueden realizar esta administración de pagos.

El administrador podrá gestionar el pago de todos los evaluadores que hayan asistido a alguna evaluación dentro del periodo y las asesoras educativas sólo podrán gestionar la asistencia de los evaluadores en las evaluaciones de sus colegios a cargo.

Dado que el pago a cada evaluador es por día evaluado, la gestión de los pagos se realiza marcando penalidades a los días asistidos, esto significa marcar si faltó o si llegó tarde o si no corrigió evaluaciones. En base a este marcado de penalidades es que se va descontando al pago por día.

Cuando se crea un periodo, se asigna el pago a cada evaluador en base a los días que trabajó por el pago por día y en base al marcado de penalidades es que se obtiene el pago total para un periodo de evaluaciones.

Se usará la siguiente fórmula para hallar el pago de cada evaluador.

$$PT = (P_{DIA} - (P_{TARDE} + P_{FALTAS} + P_{NO CORREGIR})) \times (PTA) \times (P_{DIA})$$

Para administrar un periodo de pagos hay que seleccionar el año y periodo, en la Imagen 31 se muestra el formulario para buscar un periodo.

La Imagen 28 muestra la tabla de pagos que resulta de hacer una búsqueda de periodos.

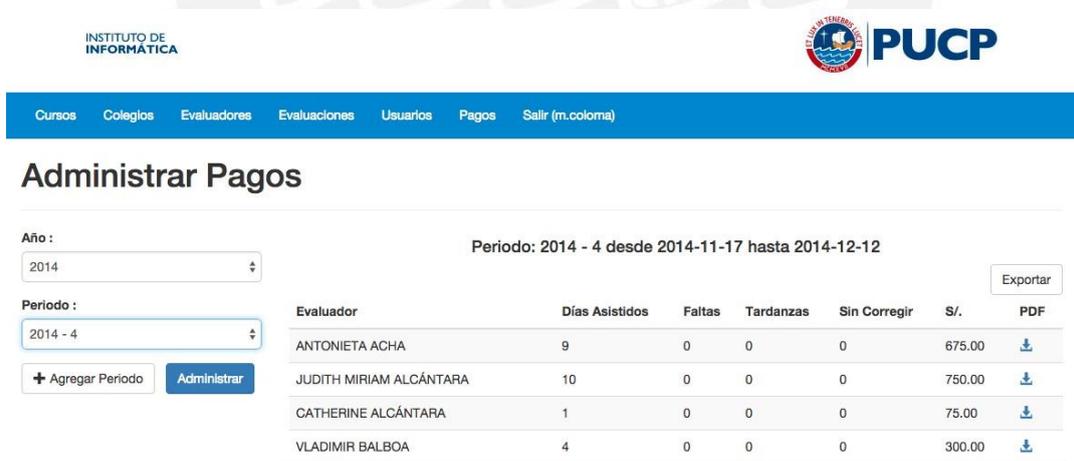


Imagen 28: Resultado de la búsqueda de un periodo. [Elaboración propia]

Las Imágenes 29 y 30 muestran la página de administración de pagos del administrador del sistema y la que utiliza una asesora educativa respectivamente para marcar la asistencia de un colegio en un periodo de pagos.

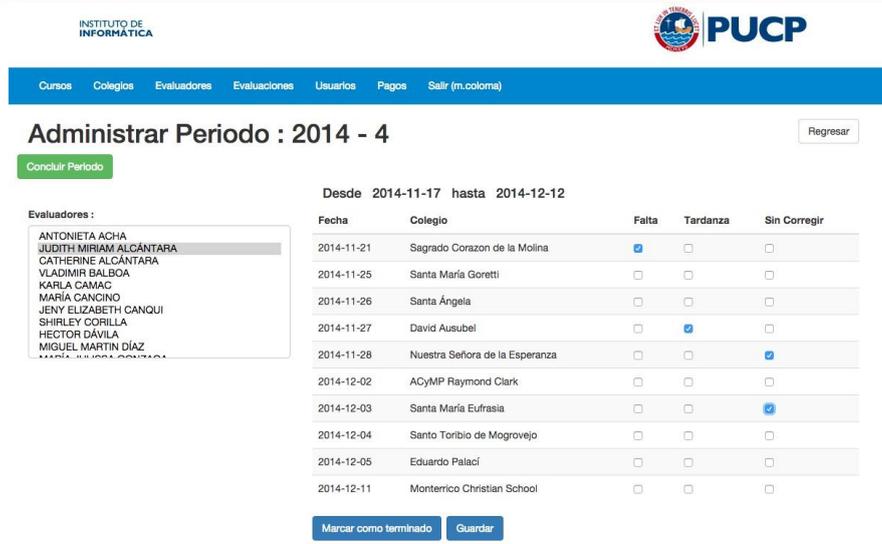


Imagen 29: Página de administración de asistencia del administrador del sistema. [Elaboración propia]

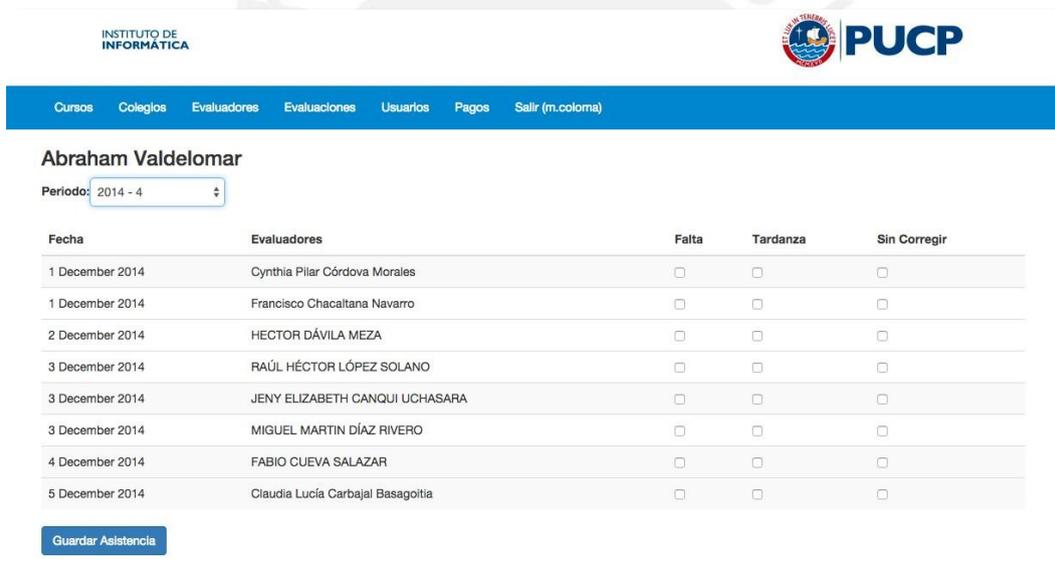


Imagen 30: Página de administración de asistencia de la asesora educativa. [Elaboración propia]

### 3.3. Conclusión

El sistema implementado facilita que cada asesora educativa controle la asistencia de las evaluaciones en los colegios que tiene a cargo de manera más práctica, facilitando y ahorrando el tiempo que empleaba la coordinadora para gestionar los pagos.

Este ahorro de tiempo puede ser constatado por la coordinadora del área al término de la gestión de pagos de un periodo.

También se elimina el riesgo de cometer errores habituales que se dan al realizar un trabajo manual como usar una hoja de cálculo para la gestión de pagos.

## CAPITULO 6

### 1. Asignación automática de evaluadores

Este capítulo muestra el desarrollo realizado para la asignación automática de evaluadores a cada evaluación de un periodo de evaluaciones.

#### 1.1. Criterio de selección

Para elegir al evaluador más apto de cada evaluación se asignará un peso a cada evaluador disponible de tiempo. Este peso se dará en función a su cercanía al lugar donde se lleva a cabo la evaluación y si es que posee conocimientos afines al curso a evaluar. Luego se ordenará de forma descendente en relación de este peso y se elegirá a los que tengan mayor peso (la cantidad de evaluadores a asignar por evaluación es indicada por la persona que use el sistema).

La fórmula usada para calcular el peso es la siguiente:

$$P = \left( \frac{1}{PD} \right)^M * K + T * M$$

Dónde:

N = Número de evaluadores disponibles

PD = Posición del evaluador en el arreglo ordenado de acuerdo a la distancia a la evaluación

TC = 1 o 0, representa si el evaluador tiene conocimientos o no respectivamente

K = Peso por distancia

M = Peso por conocimiento

La prioridad de selección es elección de la persona que realiza la asignación, puede elegir dar más prioridad (peso) a la distancia o mayor prioridad al conocimiento. Esta elección se aprecia en la Imagen 31.

Para determinar la distancia entre la dirección del evaluador y la del colegio se usará la fórmula de Haversine y para conocer si un evaluador cuenta con conocimientos afines al curso a evaluar, se buscará si dentro del matriz de conocimientos del evaluador hay al menos uno que esté incluido en la matriz de conocimientos del curso.

#### 1.2. Selección de evaluadores más óptimos

A continuación se explicará cómo se obtienen a los evaluadores más idóneos para cada evaluación.

Primero se deben obtener a los evaluadores que disponen de tiempo para asistir a una evaluación. Si un evaluador se encuentra en un colegio y dispone de tiempo para asistir a la evaluación, se hará uso del API de direcciones de Google para calcular el tiempo promedio que demora en ir desde el colegio donde se encuentra hasta el colegio de la evaluación. Si adicionando el tiempo que devuelve el API, más un adicional de 15 minutos por tráfico, el evaluador puede llegar a la hora de la evaluación es tomado en cuenta, caso contrario se descarta.

Luego de obtener a los evaluadores que disponen de tiempo para asistir a una evaluación, se verifica si poseen coordenadas de origen, de no poseer coordenadas se le asigna las coordenadas de la Universidad. También se verifica si los evaluadores poseen conocimientos afines.

A continuación se ordenan respecto a la cercanía al colegio donde se rinde la evaluación (primero a los que están más cercanos) y se les asigna el peso correspondiente por cercanía y conocimientos. Para hacer este ordenamiento se hace uso de la fórmula de Haversine mostrada en Marco Conceptual.

Por último se ordenan en forma descendente de acuerdo al peso obtenido.

### 1.3. Funcionalidad de asignación automática

Tal como se aprecia en la Imagen 24, en la pestaña de evaluaciones se cuenta con un botón nombrado “Asignación Automática”, al seleccionarlo aparecerá el formulario para la dicha asignación.

La Imagen 31 muestra el formulario para realizar la asignación automática.



**Imagen 31: Formulario de asignación automática [Elaboración propia]**

Luego de seleccionar el periodo de evaluaciones se cargará debajo del formulario la lista de evaluaciones por asignar, agrupadas por colegio. Luego de llenar los demás datos de podrá realizar la asignación.

Al finalizar la asignación se mostrará las evaluaciones asignadas con sus respectivos evaluadores y de no contar con evaluadores disponibles para algunas evaluaciones, éstas se mostrarán luego de este listado. La Imagen 32 muestra el resultado de la asignación.

Las siguientes evaluaciones fueron asignadas correctamente.

Hay un total de 62 evaluaciones

ACyMP Kenn Opperman							
Curso	Fecha	Inicio	Fin	Nivel	G	S	Evaluadores
InfoKids Nivel 4-Infornática General	2015-04-29	08:00	09:30	Primaria	4	A	Francisco Chacaltana Lourdes Cartolin
InfoKids Nivel 5-Infornática General	2015-04-29	09:30	11:00	Primaria	5	A	Francisco Chacaltana Lourdes Cartolin
InfoKids Nivel 6-Infornática General	2015-04-29	11:00	13:00	Primaria	6	A	Francisco Chacaltana Lourdes Cartolin

ACyMP Raymond Clark							
De Jesús							
María Goretti							

**Imagen 32: Resultado de la asignación automática. [Elaboración propia]**

Si un evaluador cancela su asistencia luego de ser comunicado que está asignado a una evaluación, se puede buscar la evaluación en la pestaña de evaluaciones e ingresar al ícono de asignación en la columna de opciones como se muestra en la Imagen 24. Al ingresar a la ventana de asignación por evaluación se visualiza las listas de evaluadores asignados y disponibles, al lado de cada evaluador asignado hay una opción para desasignarlo de la evaluación.

La Imagen 33 muestra lo descrito en el párrafo anterior.

**InfoKids Nivel 4-Infornática General**

Fecha: 2015-05-04  
 Inicio: 08:10:00  
 Fin: 09:40:00  
 Colegio: ACyMP Raymond Clark  
 Distrito: Pueblo Libre

**Evaluadores Asignados:**

- ALFRED QUICHIZ MENDIVIL Desasignar
- JÉSSICA INÉS ABANTO LEYVA Desasignar

**Evaluadores disponibles:**

- LUIS RODRIGO CANAL ROJAS (a 2.9 Km)
- GUSTAVO SEGURA CHÁVEZ (a 2.9 Km)

**Imagen 33: Asignación por evaluación. [Elaboración propia]**

#### 1.4. Conclusión

Esta función de asignación automática reduce significativamente el tiempo empleado para asignar evaluadores, esta tarea que demoraba horas e incluso días, ahora es realizada en pocos minutos.

Esto ayudará a que en un futuro se puedan aumentar colegios y evaluadores sin que esta actividad se convierta en un cuello de botella para la Institución.



## Referencias bibliográficas

### Libros

RABASA DOLADO Alejandro, SANTAMARÍA ARANA Laureano  
2004 Metodología de programación, Principios y aplicaciones.  
Alicante España: Editorial Club Universitario

PROJECT MANAGEMENT INSTITUTE  
Cuarta Edición Guía de los Fundamentos para la Dirección de Proyectos (Guía del  
PMBOK).  
Pennsylvania EE.UU.

### Diccionarios

REAL ACADEMIA ESPAÑOLA (RAE)  
2001 Diccionario de la lengua española (DRAE). Madrid, España.

### Leyes

CONGRESO DE LA REPÚBLICA DEL PERÚ  
2011 Ley 29733. Protección de datos personales. 03 de Julio.

### Artículos

SCHAERF Andra, DI GASPERO Luca.  
2008 “Local Search Techniques for Educational Timetabling Problems”. Universidad de  
Udine, Italia.

Prof. Nitin R.Chopde , Mr. Mangesh K. Nichat  
2013 “Landmark Based Shortest Path Detection by Using A\* and Haversine Formula”.  
International Journal of Innovative Research in Computer and Communication  
Engineering

Jovin J. Mwemezi, Youfang Huang  
2011 Optimal Facility Location on Spherical Surfaces: Algorithm and Application. New  
York Science Journal.

### Tesis

BEJARANO NICHÓ, Gissella  
2010 “Planificación de horarios del personal de cirugía de un hospital del estado  
aplicando algoritmos genéticos (timetabling problema)”. Tesis para obtener título de  
ingeniero informático. Lima: Pontificia Universidad Católica del Perú.

BAÑON FELIX, Luis

2013 “Análisis, diseño e implementación de un sistema de información para planificar la distribución de productos electrodomésticos optimizando los costos”. Tesis para obtener título de ingeniero informático. Lima: Pontificia Universidad Católica del Perú.

LÉVANO CHUQUILLANQUI, Cecilia

2012 “Software para la gestión de horarios en colegios fe y alegría”. Tesis para la obtención del título profesional de ingeniero de software. Lima: Universidad Peruana de Ciencias Aplicadas.

### Referencias Web

INFOPUC

2012 Instituto de informática académica de la PUCP. Programa de desarrollo con el uso de la TIC.  
Consulta: 11 de abril de 2014.  
<<http://infopuc.pucp.edu.pe/programa-de-desarrollo-educativo-con-el-uso-de-las-tics/>>

GOOGLE

2013 Google Developers. El Api de matriz de distancia.  
Consulta: 17 de abril de 2014.  
<<https://developers.google.com/maps/documentation/distancematrix/?hl=es>>

OPTIHPER

2009 Asignación Optimizada de Horarios al Personal. Universidad Politécnica de Valencia, España.  
Consulta: 18 de abril de 2014.  
<<http://users.dsic.upv.es/grupos/gps/optihper/>>

KAPPIX

2007 DRoster Employee Scheduling Software  
Consulta: 18 de Abril de 2014.  
<<http://www.kappix.com>>

ASC

2014 asc Horarios. aSc Applied Software Consultants.  
Consulta: 18 de abril de 2014.  
<[http://www.asctimetables.com/timetables\\_es.html#!/home](http://www.asctimetables.com/timetables_es.html#!/home)>

YII

2014 Yii Software LLC.  
Consulta: 30 de mayo de 2014.  
<<http://www.yiiframework.com>>

PHP

2001 The PHP Group.  
Consulta: 30 de mayo de 2014.  
<<http://www.php.net>>

## JQUERY

2014

The jQuery Foundation.  
Consulta: 30 de mayo de 2014.  
<<http://jquery.com>>

## OMG

2014

Object Management Group. Business Process Model and Notation  
Consulta: 24 de abril de 2015.  
< <http://www.bpmn.org/>>

## PEÑALARA

2015

GHC 17 Gestor de horarios para centros educativos. España.  
Consulta: 06 de Junio de 2015.  
<<http://www.penalara.com/>>

