

PONTIFICIA UNIVERSIDAD CATÓLICA DE VALPARAÍSO
FACULTAD DE INGENIERÍA
ESCUELA DE INGENIERÍA INFORMÁTICA

Action replay

GONZALO ANDRES ZEBALLOS ALVAREZ

INFORME FINAL DE PROYECTO
PARA OPTAR AL TÍTULO PROFESIONAL DE
INGENIERO DE EJECUCIÓN EN INFORMÁTICA

Noviembre 2017

PONTIFICIA UNIVERSIDAD CATÓLICA DE VALPARAÍSO
FACULTAD DE INGENIERÍA
ESCUELA DE INGENIERÍA INFORMÁTICA

Action Replay

GONZALO ANDRES ZEBALLOS ALVAREZ

Profesor Guía: Ismael Figueroa Palet

Profesor Co-referente: Ignacio Araya Zamorano

Carrera: Ingeniería de Ejecución en Informática

Noviembre 2017

Dedico esto a mi abuelo Doroteo, de lo orgulloso que estoy de ser como soy gracias a él, a mis padres Viviana Alvarez y Erick Zeballos por la ayuda, educación y valores que entregaron y lo afortunado que soy al tenerlos como padres, a mi hermana Makarena que me ayudo en todo momento para la realización de este proyecto, a mis amigos Felipe, Claudio, Michelle, Roberto y Jhohann por acompañarme en este camino que fue la universidad. Un agradecimiento especial a Claudio y Cristofer quienes sin ellos este proyecto no estaría completado, finalmente a mi abuela María Moreno quien siempre me ayudo con su amor incondicional.

“Si algo no lo entiendes, estúdialo, si algo no sabes hacerlo, apréndelo, pero nunca te quedes con la duda o las ganas de hacer algo”
Manuel Alvarez

G. Zeballos

Índice

Resumen.....	vi
Abstract	vi
Glosario.....	vii
Lista de Figuras.....	viii
Lista de Tablas	ix
1 Introducción	1
2 Marco general del proyecto.....	2
2.1 Descripción del problema	2
2.2 Descripción General del proyecto.....	2
2.3 Alcance y propósito	3
2.4 Objetivo general.....	3
2.5 Objetivos específicos	3
3 Situación de estudio	4
3.1 Situación actual en la escuela	4
3.2 Conocimientos Nacionales	4
3.2.1 Educación Personalizada.....	4
3.2.2 Evaluación de progreso del aprendizaje.....	5
3.2.3 Educación basada en competencia	5
3.2.4 Evaluación formativa y compartida	6
3.3 Retroalimentación usando PSeInt.....	8
4 Propuesta de solución.....	9
4.1 Descripción general de la solución	9
4.2 Metodología	9
4.3 Arquitectura lógica y física de la solución.....	10
4.4 Herramientas y tecnologías de desarrollo	11
4.4.1 Planificación del sistema.....	11
4.4.2 Herramientas de modelado de sistema	11
4.4.3 Plataforma de desarrollo.....	11
4.4.4 Lenguaje de programación	12
5 Análisis.....	13

5.1	Definición de requerimientos.....	13
5.1.1	Requerimientos funcionales	13
5.1.2	Requerimientos no funcionales	14
5.2	Casos de uso.....	14
5.2.1	Actores del sistema.....	14
5.2.2	Diagrama de caso de uso	14
5.3	Caso de uso narrativo.....	16
6	Diseño.....	17
6.1	Diagrama de secuencia	17
6.2	Diseño de la interfaz	17
6.3	Diseño de lectura de datos	19
6.3.1	XML Schema	20
6.4	Diseño de base de datos	21
7	Conclusión.....	23
8	Referencias	24
	Anexos	25
A	Caso de uso narrativo	26
A.1	Caso de uso narrativo nivel del actor profesor.....	26
B	Diagramas de secuencia	31
B.1	Seleccionar evento	31
B.2	Filtrar sucesos de eventos	32
B.3	Ejecutar código	32
C	Archivos XML	33
C.1	Ejemplo 1.....	33
C.2	Ejemplo 2.....	33
C.3	Ejemplo 3.....	33

Resumen

En este informe se detalla el diseño y desarrollo de una aplicación web orientada a la retroalimentación del docente hacia los alumnos de la Escuela de Ingeniería Informática de la Pontificia Universidad Católica de Valparaíso durante sesiones de programación. El profesor podrá evaluar no solo el resultado final de la tarea asignada, sino que también podrá valorar el proceso de desarrollo de las tareas encomendadas durante el semestre universitario mediante herramientas ofrecidas por la aplicación web. Este proyecto plantea desde distintas visiones la problemática que enfrentan muchas universidades en temas de evaluación que puedan ofrecer una ayuda más personalizada al alumno durante su proceso de aprendizaje.

Abstract

This report details the design and development of a web application oriented towards providing feedback to the students of the Escuela de Ingeniería Informática during programming sessions. The teacher can not only grade the result of the assigned work but also, he can assess the development process of the application via the presented tools on the web application. This project poses from different visions, the problem that many universities have face to regarding a more personalized evaluation that can serve as help to the students during their learning process.

Glosario

- **Alumno:** En este informe se hará referencia a los estudiantes de la escuela de Ingeniería Informática de la Pontificia Universidad Católica de Valparaíso, Chile.
- **Docente:** Persona que se halla legalmente autorizada para enseñar. En este informe se hará referencia puntualmente a los docentes o profesionales de la educación que se desenvuelven laboralmente en las dependencias de la PUCV.
- **Evaluación Académica:** Valoración de conocimientos, actitud y rendimiento de una persona.
- **Feedback:** es una palabra del inglés que significa retroalimentación
- **IDE:** *Entorno de desarrollo integrado* es una aplicación informática que proporcionan servicios integrados para facilitar el desarrollo de software.
- **XML:** metalenguaje extensible de etiquetas que fue desarrollado por w3c que permite la organización y el etiquetado de documentos.
- **Evento:** Suceso importante y programado, de índole académico. Para este documento se referirá a las acciones realizadas dentro de un entorno IDE.
- **XML Schema:** Lenguaje de esquema utilizado para describir estructuras y restricciones del contenido de un documento XML.

Lista de Figuras

Figura 3.1Ciclo de la evaluación formativa de la agencia de la calidad de la educación.....	7
Figura 4.1 Modelo de interacción-incremental.....	10
Figura 4.2 Logo Microsoft Word	11
Figura 4.3 Logo Microsoft Project	11
Figura 4.4 Logo Microsoft Visio	11
Figura 4.5 Logo NotePad++	12
Figura 4.6 Logo Visual Studio 2017	12
Figura 4.7 Logo C#.....	12
Figura 4.8 Logo HTML, CSS y JS	12
Figura 5.1Diagrama de caso de uso.....	15
Figura 6.1 Diagrama de secuencia de subir archivo	17
Figura 6.2 Mockup de error de compilación	18
Figura 6.3Mockup de cambios realizados al código	18
Figura 6.4 Mockup de ejecución de código por pantalla.....	19
Figura 6.5 Esquema de la estructura de un archivo XML.....	20
Figura 16 Diseño de la base de datos	21
Figura B.1Diagrama de secuencia seleccionar evento	31
Figura B.2 Diagrama de secuencia filtrar sucesos de eventos.....	32
Figura B.3 Diagrama de secuencia ejecutar código	32

Lista de Tablas

Tabla 5.1 Requerimientos funcionales	13
Tabla 5.2 Requerimientos no funcionales	14
Tabla 5.3 Caso de uso narrativo de subir archivos XML	16
Tabla A.1 Caso de uso narrativo seleccionar evento	26
Tabla A.2 Caso de uso narrativo Filtrar suceso de eventos.....	27
Tabla A.3 Caso de uso narrativo ver información empírica.....	28
Tabla A.4 Caso de uso narrativo ejecutar código.....	29
Tabla A.5 Caso de uso narrativo ver errores de compilación.....	30

1 Introducción

En el presente informe se detalla el proceso de creación de un sistema web de seguimiento para los alumnos de la carrera de Ingeniería en Informática de la Pontificia Universidad Católica de Valparaíso en el proceso de creación de un software durante sesiones de programación con la tutela de un profesor de la Escuela de Ingeniería Informática de la Pontificia Universidad Católica de Valparaíso, Chile. Este sistema tiene como fin ayudar a evaluar y mejorar las habilidades de desarrollo de software de los estudiantes y a su vez demostrar sus habilidades y conocimientos, permitiendo conocer el avance del proceso del programa realizado por el alumno.

En la actualidad existe una serie de preocupaciones en los establecimientos educativos por ofrecer una educación más personalizada [1] a los estudiantes, con el fin de orientar de mejor manera la enseñanza hacia los alumnos. Es por eso donde la evaluación juega un papel importante dentro del proceso de desarrollo para la autoevaluación, para así mejorar las tomas de decisiones realizadas por los pupilos. Por lo general la enseñanza en las instituciones universitarias sólo se ve reflejada por el nivel actual del estudiante [2], dejando de lado el posible progreso durante su desarrollo como profesional. Por este m muchos alumnos en el transcurso de una asignatura van reteniendo dudas, las cuales se van acumulando hasta el día de su evaluación donde recién ven reflejado que lo que estuvieron practicando o estudiando estaba equivocado. Es por esto por lo que se pretende crear un sistema que ayude a evaluar el desarrollo de las sesiones de programación de los alumnos de Informática, ofreciéndole al docente diferentes herramientas que ayuden a analizar la forma de programar del estudiante. Para así ofrecer un feedback más detallado de los errores producidos en el transcurso del desarrollo de las tareas de programación.

2 Marco general del proyecto

En la presente sección se presentará la descripción general de la problemática a la cual se desea abarcar en este proyecto, como también una descripción general de la solución, su alcance y los objetivos de este.

Para el propósito de este proyecto, las sesiones de programación se definirán a las tareas y/o trabajos realizados dentro de un entorno de programación efectuado dentro de salas de clases de la facultad de Ingeniería de la Escuela de Informática de la PUCV, bajo la guía de un profesor el cual estará apoyando durante la realización de dichas tareas y/o trabajos.

2.1 Descripción del problema

En Chile, hace mucho tiempo se habla del problema en la calidad de la educación superior, y uno de los factores es que las evaluaciones solo reflejan una parte del estudiante y no todo su conocimiento actual. Para resolver esto, se han planteado muchas soluciones que, en distintos puntos de vista, tratan de resolver el sistema de evaluación del país. Por lo mismo, en la enseñanza de la programación, generalmente se evalúa soluciones finales que pueden estar correcta o incorrectas, pero sin considerar el proceso usado por el estudiante para llegar a esa solución. Produciendo que los errores de programación reiterados puedan transformarse en malos hábitos para los estudiantes. Esto produce que no se logre ofrecer una retroalimentación a los estudiantes respecto al proceso que usaron para llevar a cabo su solución a un problema de programación en concreto.

Hoy existe una herramienta IDE [7] que graba las interacciones realizadas por los usuarios durante sesiones de programación, pero solo entrega un archivo XML como resultado de lo realizado por el alumno mediante secuencias de eventos, produciendo que sea tedioso y complicado de entender. Esta herramienta es una extensión para el programa PSeInt el cual al momento de desarrollar una aplicación en pseudo lenguaje de programación y luego guardarlo, este genera un archivo XML con los eventos realizados. En el anexo C se podrá apreciar un ejemplo de estos archivos XML.

2.2 Descripción General del proyecto

El proyecto en su integridad consta de una aplicación web orientada en la lectura de archivos XML de las diferentes sesiones de programación de los alumnos de la Escuela de Informática. El sistema podrá facilitar la lectura e interpretación del XML por parte del docente a evaluar, para así evaluar los resultados del aprendizaje de cada alumno de forma más detallada según sus conocimientos o criterios. A su vez el software entregará información al profesor de las diferentes interacciones realizadas durante el desarrollo del software. Todo lo mencionado anteriormente se podrá realizar mediante una línea de tiempo que facilitará el análisis del transcurso de la programación. El sistema ofrecerá diferentes tipos de herramientas que facilitaran el análisis de la sesión de programación, realizada por el estudiante. Algunas de estas herramientas será un sistema filtrado de las interacciones realizadas dentro del entorno de desarrollo para ofrecer una vista más dinámica de lo trabajado durante el proceso de la sesión de programación. Además, entregará todos los mensajes de errores que recibió el usuario durante el desarrollo del software, para que así el profesor o guía pueda ir analizando los errores

que realizó el estudiante durante el desarrollo. Mostrar de forma gráfica los cambios realizados durante el proceso desarrollo del software. Finalmente, la aplicación entregará una herramienta que permitirá ejecutar el código mostrado en la pantalla para así ir probando el código desarrollado hasta ese punto.

2.3 Alcance y propósito

El sistema desarrollado tiene como propósito ayudar a los docentes a realizar una evaluación más crítica de los estudiantes durante sesiones de programación para así mejorar la formación de los alumnos durante sus años de la carrera y orientarlos de mejor manera en sus errores habituales al momento de programar. Para llevar a cabo este proyecto se necesita la implementación de una herramienta que sea capaz de entregar el desarrollo del alumno durante realización de sus tareas de programación, lo cual permitirá realizar un feedback durante cada una de las etapas de aprendizaje.

El enfoque de esta solución apunta inicialmente a los estudiantes de primer año de la carrera de informática, ya que le permitirá al docente realizar un seguimiento exhaustivo del proceso de aprendizaje de cada alumno, pudiendo evaluar en tiempo real e incluso retrospectivamente el proceso completo que llevó a cabo el alumno durante las sesiones de programación, permitiendo realizar las correcciones durante el proceso y no únicamente del resultado, modo que se utiliza a la fecha. Esta diferencia en la evaluación favorecerá el aprendizaje de los estudiantes debido a que podrán conocer los errores que van cometiendo durante la elaboración del ejercicio y evitarlos en futuros trabajos.

2.4 Objetivo general

Desarrollar una aplicación web que ofrezca herramientas para analizar la interacción del usuario dentro de un entorno de programación, para así entregar un feedback del código que fue desarrollando durante la sesión de programación.

2.5 Objetivos específicos

- Generar una línea de tiempo de las interacciones realizadas por el usuario dentro un IDE.
- Filtrar las interacciones realizadas por el usuario
- Categorizar la información para organizarla en una línea de tiempo.
- Mostrar información empírica de los cambios realizados durante el desarrollo de la aplicación.

3 Situación de estudio

En esta sección se definirá, primero la descripción en grandes aspectos del problema, identificando la descripción del mismo, segundo el cómo se maneja actualmente la Escuela de Ingeniería Informática de la PUCV en términos de evaluación y los problemas que demanda a la hora de la enseñanza, la contextualización de los procesos de enseñanza, aprendizaje y solución, y finalmente un trabajo realizado por la Universidad de Valparaíso el cual presenta una solución al momento de evaluar los alumnos de programación.

3.1 Situación actual en la escuela

En la Escuela de Ingeniería Informática de la Pontificia Universidad Católica de Valparaíso en su mayoría se evalúa a los alumnos mediante evaluaciones de estado donde solo ve reflejado el objetivo final del ramo fue cumplido, donde, en caso de no cumplir con estos requerimientos el alumno quedará en estado reprobado y tendrá que repetir la asignatura. Esto provoca que las posibles dudas o inquietudes del alumno no se vean reforzadas hasta su próximo semestre, o quizás nunca pueda llegar a resolverlas. Es por eso que se busca nuevas formas de reforzar la enseñanza de los alumnos mediante ayudantías extras o usando nuevas metodologías de evaluación que ayuden a la enseñanza.

3.2 Conocimientos Nacionales

En esta sección se describirán y analizarán cuáles son las posturas, acciones e hitos más importantes a nivel nacional. Se consideran varias visiones relevantes; la primera es la del portal Educar Chile fundado por el ministerio de educación de Chile (Mineduc) y la Fundación de Chile (FCH), la segunda es de SEPA (Sistema de Evaluación de Progreso de Aprendizaje), la tercera del Consejo de Rectores de Universidades Chilenas (Cruch) y finalmente el Ministerio de Educación (Mineduc).

3.2.1 Educación Personalizada

Educar Chile desde su fundación ha tenido como misión el mejoramiento de la calidad de educación en todos sus niveles, ámbitos y modalidades, ampliando las oportunidades formación y aprendizaje. Es por eso por lo que ha investigado y construido diversas formas de educar dentro del país, generando educación personalizada para los estudiantes tanto en la enseñanza media como en la universitaria con la intención de buscar una forma precisa de enseñar.

La enseñanza personalizada está basada en una visión donde reúne tanto la persona como necesidades sociales y culturales en un todo [3]. Donde las técnicas y enseñanzas no pueden ser distanciadas del fundamento filosófico y psicológico. Aun así, mediante la creatividad se pueden adaptar los sistemas de educación personalizada para incorporarlas en otros tipos de evaluaciones. Un ejemplo es el método de la normalización usado en la educación personalizada, el que adapta la enseñanza a los estudiantes para ofrecer una mejor libertad y hacer un uso correcto de herramientas y materiales usados dentro de la sala de clase. Donde esto mismo se puede aplicar en las universidades para la enseñanza en el uso de herramientas que se requieran dentro del campo laboral de cada profesión.

Las ventajas de estas enseñanzas es que ningún momento un alumno se sentirá fuera del proceso de enseñanza, ayudando a mejorar sus habilidades personales, como también su forma de abordar los problemas frente alguna evaluación o tarea a realizar. Otra ventaja de la enseñanza personalizada es brindar seguridad al estudiante, para que sienta libertad de expresar sus opinión y sentimientos frentes a dudas o problemas.

Para esto se debe realizar en un grupo no mayor a 10 alumnos, donde el docente pueda reconocer las fortalezas de sus estudiantes como también sus debilidades para obtener un mayor desempeño [4]. Esto ayuda en la inspiración del alumno para cuestionar y cuestionarse, todo el tiempo ayudando a buscar las debilidades de cada uno y así mejorarlas con un mayor énfasis en ellas.

3.2.2 Evaluación de progreso del aprendizaje

SEPA consiste en un conjunto de pruebas especialmente diseñadas para evaluar, a lo largo del tiempo, los aprendizajes de los estudiantes en Lenguaje y Matemática, incorporando la metodología de Valor Agregado (VA). Esta metodología permite analizar, entre otros aspectos, el aporte que hace el establecimiento al desarrollo de los aprendizajes logrados por los estudiantes.

Hoy en día la enseñanza en Chile solo brinda una vista del estado actual del alumno mediante evaluaciones que solo se rigen para ver el nivel final del desempeño del alumno, dejando de lado el progreso o trayectoria que él realizó durante su proceso de enseñanza. Es por eso que SEPA busca crear un sistema de medición de progreso, en el cual otorgue una medición más fiel del aprendizaje del periodo educativo, comparar grupos de manera más equitativa y arrojar una alerta temprana en caso de que se necesite realizar un reforzamiento de la materia.

Para ello SEPA ofrece sus servicios a establecimientos educacionales, los cuales reciben los instructivos para aplicar el sistema y capacitan a los docentes para aplicar el nuevo sistema de evaluación, luego se entrega las pruebas y se aplican donde después SEPA retira el material y entrega sus resultados de estado al centro educativo para finalmente entregar una capacitación para el uso de los resultados obtenidos. En los años siguientes se va obteniendo resultados del progreso obtenido durante el avanzar del tiempo y el valor agregado. Con esto la escuela obtendrá una visión del progreso durante la evolución del alumno dentro de sus años de estudios.

3.2.3 Educación basada en competencia

Habiendo muchas y diversas definiciones en torno al significado de la palabra competencia, es necesario definir cuál es el significado que le da el Consejo de Rectores:

(Armanet, 2015) *“Las competencias son conjuntos dinámicos e integrados de conocimientos, habilidades y actitudes, que pueden ser inducidos durante el proceso formativo y cuyo grado de adquisición y/o desarrollo (logro de aprendizaje) es susceptible de evaluarse; las competencias, que están al servicio del perfil de egreso, requieren metodologías docentes renovadas cuyo centro es el estudiante” (p.18)*

La anterior definición, corresponde al concepto en común dentro de todas las universidades que componen este grupo. Donde según el Consejo de Rectores se clasifica la competencia en dos grandes categorías: Competencias Genéricas y Competencias Específicas.

- **Competencia Genéricas:** Corresponde a la competencia que apunta a la incorporación de recursos personales (actitudes, conocimientos y habilidades) con fines para el desempeño dentro del campo laboral. Esta competencia conforma la parte ética y laboral del trabajo profesional que son parte del perfil de egreso.
- **Competencia Específicas:** corresponde al comportamiento integrado de conocimientos científicos y técnicos, actitudes y valores que se aplican a la solución de un problema determinado o a una situación específica de carácter académico-profesional.

Si bien cuando se habla de una enseñanza tradicional o de competencia, no se puede dejar de lado las evaluaciones, ya que son éstas la base de cualquier nivel de medición dentro del campo educativo [6]. Así, después de realizadas las evaluaciones, se determinará las habilidades del estudiante que tiene hasta a la fecha o también las que es capaz de demostrar, una vez concluido el proceso.

En un modelo de formación mediante competencias, las competencias deben ser acordes con el perfil de egreso y las formas de aprendizaje para su buen desarrollo, así como también coherentes con los instrumentos de evaluación. Para así lograr un nivel de enseñanza más orientada en las necesidades del estudiante.

3.2.4 Evaluación formativa y compartida

Las evaluaciones formativas son un proceso donde profesores y estudiantes comparten metas de aprendizaje, donde el uso de evaluaciones constantes como parte del trabajo cotidiano dentro de la sala de clase, donde estas se utilizan para guiar el proceso de enseñanza y aprendizaje, para así la toma decisiones pertinentes que beneficien tanto a los estudiantes como profesores [5].

El proceso de las evaluaciones formativas ayuda a resolver las siguientes inquietudes:

- **¿Hacia dónde vamos?:** La intención de este punto es que tanto estudiantes como docentes tengan claro las metas y objetivos de la asignatura. No basta simplemente con entregar a los alumnos los objetivos, si no que explicar las metas de aprendizaje a las que se quieren alcanzar, como también las evaluaciones que se realizarán; de manera tal que los alumnos puedan autoevaluar su nivel de aprendizaje.
- **¿Dónde estamos?:** En este punto tanto los estudiantes como los profesores revisan en conjunto los conocimientos que se tiene sobre el objetivo de la asignatura. Para así tener claro los contenidos necesarios que se deben reforzar al momento comenzar el aprendizaje. De esta forma todos los estudiantes saben la situación actual del ramo y las metas por alcanzar.

- **¿Cómo podemos seguir avanzando?:** Esta última etapa de la evaluación formativa se enfoca en la retroalimentación efectiva por parte de los alumnos y docentes. La idea es que retroalimentación sea clara y esté centrada en lo que se realizó correctamente y lo que se debe hacer para mejorar. La idea es que sea una guía para los estudiantes donde ellos puedan corregir su desempeño personal con la ayuda de su profesor guía.

La idea de estos es llevar a un nuevo nivel la autoevaluación y coevaluación como modalidades que permitan a los alumnos ser parte del proceso evaluativo. Logrando así una modalidad evolutiva del proceso de aprendizaje que recibirá el alumno. Pudiendo así corregir, aumentar o mejorar el sistema de enseñanza que se esté partiendo en el momento.

La siguiente figura 3.1 representa las etapas de la evaluación formativa impartida en marcha blanca en el 2016 por el ministerio de educación (Mineduc) y la agencia de la calidad de la educación.



Figura 3.1 Ciclo de la evaluación formativa de la agencia de la calidad de la educación

Donde cada parte del ciclo de la evaluación formativa está detallado en pequeñas actividades las cuales ayudan conformar la visión de cada actividad que se debe realizar para realizar una evaluación formativa y compartida.

3.3 Retroalimentación usando PSeInt

La carrera de Ingeniería en Informática en sus primeros años imparte la asignatura de Fundamentos de Programación, donde se enseña en un principio el pseudo código con la intención de iniciar la enseñanza dentro de la programación. Dependiendo de las universidades se utilizan diferentes herramientas para la enseñanza de este algoritmo, en donde la PUCV se utiliza SLE y en el caso de este estudio realizado por la Universidad de Valparaíso usando PSEINT. El objetivo de esta última herramienta es para enseñar los fundamentos de la programación y el desarrollo de la lógica, con el fin introducir el concepto básico del uso de estructuras de control. Esta aplicación procura facilitar la tarea de escribir algoritmos de pseudo lenguaje utilizando un conjunto de ayudas y asistencias, las cuales brindan herramientas que ayudan en el aprendizaje de la lógica de programación.

El Ingeniero Informático Daniel Palma Cordovez de la Universidad de Valparaíso [7] implemento una extensión para PSeInt la que permite recopilar información relevante para el docente de la asignatura, con la intención de acceder a esta para un posterior análisis. La extensión registra un historial de los eventos más importantes realizadas dentro de la aplicación, donde estos eventos son clasificados y agrupados para luego ser almacenados en un archivo XML. El fin de esta extensión es ofrecerle al profesor información la cual pueda servirle de guía al momento de dictar el curso de “Fundamentos de programación”. Así podrá tener una noción inicial sobre las dificultades típicas de los estudiantes, o bien puede utilizar la información para dar hincapié en aquellos obstáculos más concurridos a medida que se imparte la asignatura.

Con el fin de continuar el trabajo realizado por Daniel Palma Cordovez donde en una primera parte se utilizará los archivos XML generados por la extensión desarrollada por él, con el fin de crear un sistema que sea capaz de procesar la información recolectada y representarla de tal forma que sirva para crear una retroalimentación útil para detectar los problemas más frecuentes dentro de los estudiantes cuando programa.

4 Propuesta de solución

En la siguiente sección se detalla la forma en la que se abordará el proyecto, como también los modelos de desarrollo y la descripción general de la propuesta de solución.

4.1 Descripción general de la solución

Como solución para esta problemática, se propuso que se realizara una plataforma web, ya que, considerando las facilidades de operación de una página web tanto en dispositivos móviles, como en cualquier computador. Donde las ventajas se encuentran en el apartado de una página web:

- Permite la utilización de la aplicación en cualquier lugar.
- Mejora la interacción de la aplicación con sistema externos que puedan incluirse dentro del software
- No requiere de instalaciones externas para operar.

La idea de esta aplicación es crear un medio donde los profesores puedan realizar una evaluación más completa de las actividades realizadas durante una sesión de programación para así ofrecer a los alumnos un Feedback más detallado de los errores o fortalezas que han logrado durante cada clase.

Para esto se creará una línea de tiempo donde muestre el progreso del alumno durante la sesión de programación, donde se irán detallando los pasos que fue realizando hasta llegar con la solución o hasta donde fue capaz de llegar. Donde mostrara en todo momento los errores que presenta el alumno durante actividad.

Además de lo mencionado también se implementará herramientas de compilación de lenguaje de programación, para así revisar en cualquier momento el código entregado por el alumno.

4.2 Metodología

Para el desarrollo de la aplicación se utilizará el desarrollo iterativo e incremental [8]. Debido a la poca experiencia del alumno en el desarrollo de aplicación en JavaScript, se decidió ir creando secciones reducidas del programa para ir realizando revisiones y correcciones al momento que se aumentan las habilidades de dicho lenguaje de programación. Además, como se acordaron diversas reuniones con el profesor guía, con la finalidad de ir estudiando el avance del proyecto para así corregir de manera oportuna los errores durante el proceso de desarrollo del software.

Otro modelo de desarrollo que se aproximaba a las necesidades del estudiante era el espiral, pero debido a que se desconocen todos los posibles riesgos al momento de desarrollar una aplicación en lenguaje JavaScript y en web, se optó por un modelo de desarrollo más amigable a los errores por descubrir durante el desarrollo del software. Los modelos de desarrollo como cascada o ágil no se utilizaron debido a que el primero requiere de una

programación estable y con conocimientos dentro del lenguaje de programación cosa que para este caso no se cumplía y para el desarrollo ágil; como es un desarrollo dinámico y no estandarizado no se podía cumplir con los hitos requeridos en el proyecto.

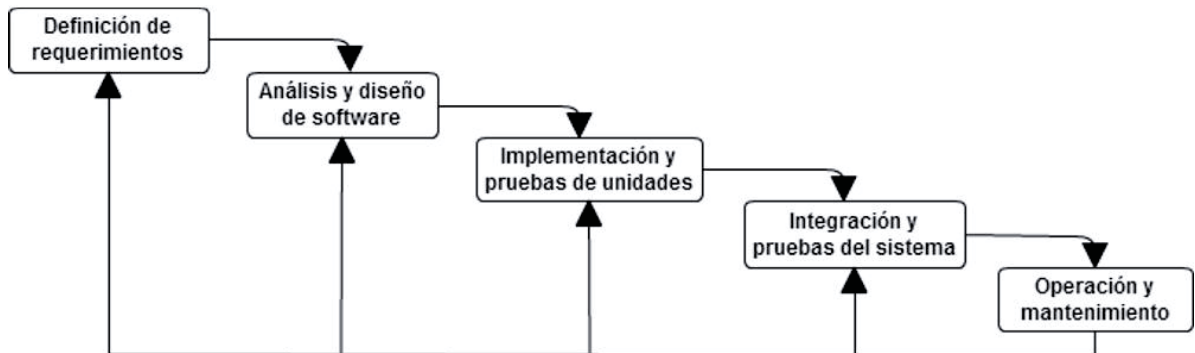


Figura 4.1 Modelo de interacción-incremental

4.3 Arquitectura lógica y física de la solución

Se definieron los siguientes niveles que se utilizaran en el proyecto para una correcta implementación y funcionalidad.

La arquitectura lógica consistirá en dos capas:

- La primera capa será la de presentación del cliente, la cual consistirá en la interfaz gráfica al consistir en un proyecto web. Se utilizará páginas basadas en HTML, CSS y Javascript.
- La segunda capa consistirá en la lógica de negocio del sistema, debido al uso de ASP.NET MVC Core <https://docs.microsoft.com/en-us/aspnet/core/>, se utilizará el patrón de diseño de Modelo-Vista-Controlador; donde toda la lógica y procesamiento se realizará en los controladores de lenguaje C#, los cuales se encargan de comunicar la interfaz con los archivos XML cargados en la página web.

La arquitectura física consistirá en dos niveles:

- Un servidor Azure donde se almacenará la página web y sesiones de programación realizadas. El servidor alojará el sistema web bajo IIS. Finalmente, esta página web estará operativa durante el desarrollo de esta tesis para que se puedan realizar pruebas de entorno y funcionalidades.
- Un servidor Linux para hacer las pruebas correspondientes de compatibilidad de la aplicación web en un entorno GNU, debido a que las instalaciones de la universidad cuentan con servidores Linux.

4.4 Herramientas y tecnologías de desarrollo

En esta sección se detalla las tecnologías a utilizar durante el desarrollo del sistema.

4.4.1 Planificación del sistema

- **Microsoft Office Word**
Programa informático desarrollado por Microsoft, orientado al procesamiento de textos [9].



Figura 4.2 Logo Microsoft Word

- **Microsoft Office Project**
Programa informático desarrollado por Microsoft orientada a la planificación, administración y desarrollo de proyectos y tareas [10].



Figura 4.3 Logo Microsoft Project

4.4.2 Herramientas de modelado de sistema

- **Microsoft Office Visio**
Programa informático desarrollado por Microsoft orientado a la creación de diagramas, organigramas, procesos empresariales, diagramas UML, entre algunos [11].



Figura 4.4 Logo Microsoft Visio

4.4.3 Plataforma de desarrollo

- **Notepad++**
Editor de código fuente libre que soporta una serie de lenguajes de programación con soporte nativo de Microsoft Windows [12].



Figura 4.5 Logo NotePad++

- **Microsoft Visual Studio**

Es un entorno de desarrollo integrado (IDE) para sistema operativos Windows. Que soporta múltiples lenguajes de programación y se utiliza para desarrollar en ASP.NET MVC Core [13]



Figura 4.6 Logo Visual Studio 2017

4.4.4 Lenguaje de programación

- **C#**

Es un lenguaje de programación orientado a objetos de desarrollo y estandarizado por Microsoft como parte de su plataforma de .NET [14]



Figura 4.7 Logo C#

- **HTML, Javascript y CSS**

Es el lenguaje que se emplea para el desarrollo de sitios web, está compuesto por una serie de etiquetas que el navegador interpreta y da forma en la pantalla, al igual que CSS son hojas de estilos que dan diseño al sistema web desarrollado [15].



Figura 4.8 Logo HTML, CSS y JS

5 Análisis

En esta sección del informe se mostrarán tanto los requerimientos funcionales como los no funcionales que tendrá el sistema, además de diagramas que puedan facilitar la comprensión del análisis desarrollado para el sistema.

5.1 Definición de requerimientos

En esta sección se detallarán los requerimientos funcionales y no funcionales para elaborar la herramienta que permitirá mostrar los datos empíricos obtenidos durante las sesiones de programación.

5.1.1 Requerimientos funcionales

A continuación, se presentarán los requerimientos funcionales.

Tabla 5.1 Requerimientos funcionales

ID	Sigla	Descripción	Prioridad
01	RF01	El profesor podrá cargar los datos de una sesión de programación almacenados en un archivo XML	Obligatorio
02	RF02	El sistema al momento de cargar el archivo XML mostrará una línea de tiempo con las acciones realizadas dentro del entorno de desarrollo.	Obligatorio
03	RF03	El profesor podrá filtrar los sucesos mostrados en la línea tiempo para facilitar su análisis.	Obligatorio
04	RF04	El profesor podrá compilar el código mostrado en pantalla para así revisar las diferentes etapas del software durante su desarrollo.	Obligatorio
05	RF05	El profesor podrá visualizar los errores de compilación del código.	Obligatorio
06	RF06	El profesor podrá ver información empírica de los cambios realizados durante las sesiones de programación.	Opcional
07	RF07	El profesor podrá ingresar notas informativas con el fin ayudar a la evaluación del alumno.	Obligatorio
08	RF08	El sistema ofrecerá sesiones para los estudiantes, en la cual podrán visualizar las notas y correcciones que le dejó el profesor evaluador.	Opcional

5.1.2 Requerimientos no funcionales

A continuación, se presentarán los requerimientos no funcionales.

Tabla 5.2 Requerimientos no funcionales

ID	Sigla	Descripción	Prioridad
09	RNF01	La aplicación web debe ser capaz de instalarse en un servidor Linux como Windows de la Escuela de informática de la PUCV	Obligatorio
10	RNF02	La Interfaz de usuario debe ser simplista para facilitar su uso.	Obligatorio

5.2 Casos de uso

En esta sección se explicará las funcionalidades de la aplicación junto a su actor correspondientes. Con el objetivo de representar de forma gráfica las acciones que podrá realizar el autor dentro del sistema, como también especificar la interacción entre dichas acciones.

5.2.1 Actores del sistema

En esta sección se detallará los diferentes tipos de usuarios del sistema, como también se señalarán las funcionalidades que ejercen cada uno.

Existen 2 tipos de usuario:

- **Profesor:** Es aquella que interactúa con el sistema. Con la intención de utilizar la información entregada por la aplicación durante las sesiones de programación, para luego ofrecer un Feedback a sus alumnos.
- **Servidor:** Decodificará los archivos XML para luego cargarlos en la interfaz de la aplicación web; junto a esto categorizar y procesará la información obtenida de este.

5.2.2 Diagrama de caso de uso

A continuación, en el siguiente diagrama se detallará el sistema en su totalidad, con la finalidad de explicar quienes interactúan con el sistema y las relaciones entre los casos de usos.

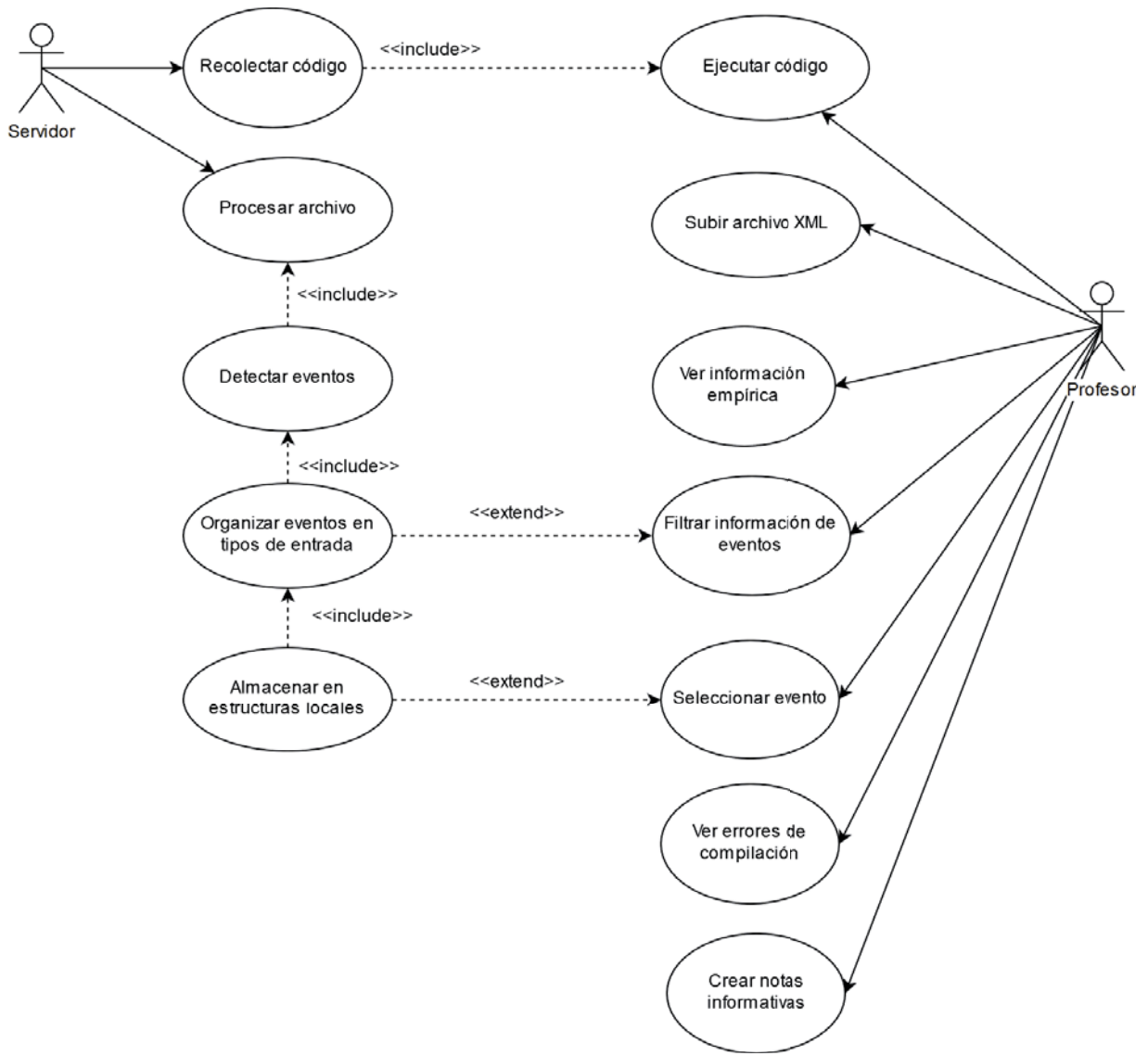


Figura 5.1 Diagrama de caso de uso

5.3 Caso de uso narrativo

En la tabla a continuación se presenta un caso de uso representativo del profesor cuando sube un archivo XML de una sesión de programación, en este se indica el transcurso de eventos que deben realizar el actor para hacer uso del caso de uso. En el Anexo B se podrán apreciar los restantes.

Tabla 5.3 Caso de uso narrativo de subir archivos XML

Caso de uso	Subir archivos XML
Actores	Profesor
Propósito	Cargar los datos del XML para así poder analizar los datos almacenados
Tipo	Primario
Descripción	El usuario abre la página web para luego cargar un archivo XML el cual contendrá la información necesaria para ser analizada en la página web.
Referencias Cruzadas	-
Curso normal de los eventos	
Acción de Actores	Respuesta del Sistema
1. El usuario accede a la página de la aplicación.	
	2. El sistema carga la información de la aplicación.
3. El usuario sube el archivo a la página web	
	4. La página carga los datos del XML y los muestra por pantalla.
5. El usuario realiza las actividades que el estime convenientes.	
Curso Alternativos	
Línea 2,4. El sistema se encuentra fuera de línea, se contacta al Proveedor de servicio que la página web presenta fallas de carga.	
Línea 3. Datos ingresados no válidos: se informa al usuario del error, indicando los campos incorrectos o el error.	

6 Diseño

En este capítulo se especificará los rasgos de diseño que implementará la aplicación web, abarcando diagramas de secuencias para esclarecer las interacciones entre los componentes del sistema y actores, y el diseño de la interfaz de usuario.

6.1 Diagrama de secuencia

La figura 6.1 se muestra el diagrama de secuencia para subir un archivo XML para su revisión, por parte del profesor. En el **Anexo B** se podrán apreciar el resto de los diagramas de secuencia que conformar el diseño del sistema.

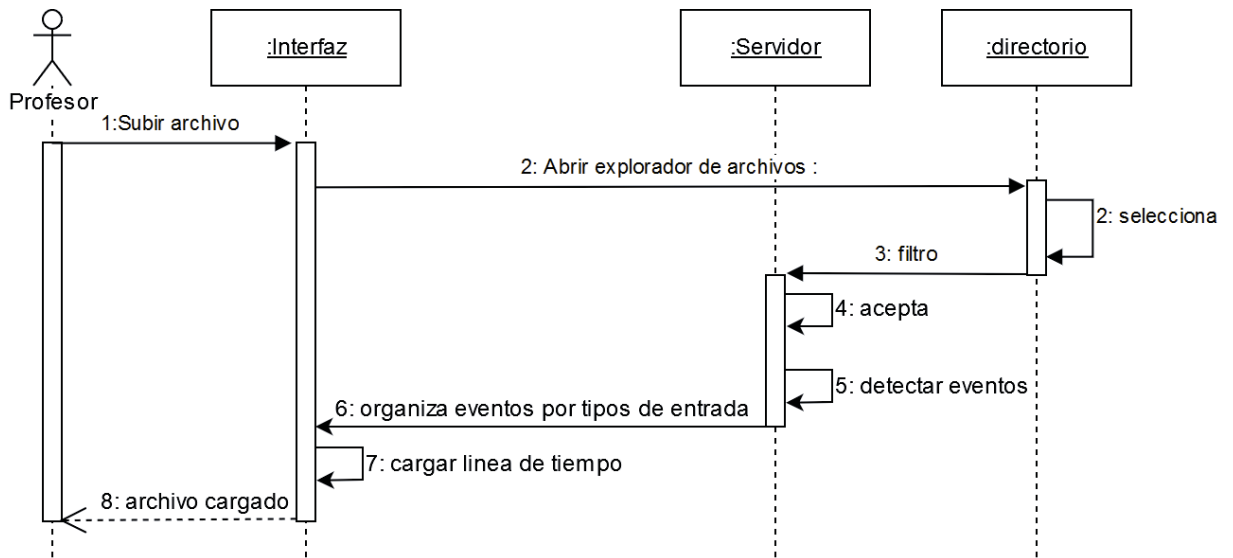


Figura 6.1 Diagrama de secuencia de subir archivo

6.2 Diseño de la interfaz

Las funcionalidades e implementación del sistema requieren del manejo de una interfaz usuaria, ya que uno de los objetivos de este sistema es facilitar el análisis de la información recolectada durante las sesiones de programación. La interfaz contará con un diseño minimalista para ofrecer solo las herramientas más necesarias dentro de un entorno evaluativo. A continuación, se mostrarán unos mockups que representarán un diseño no final de la solución propuesta.

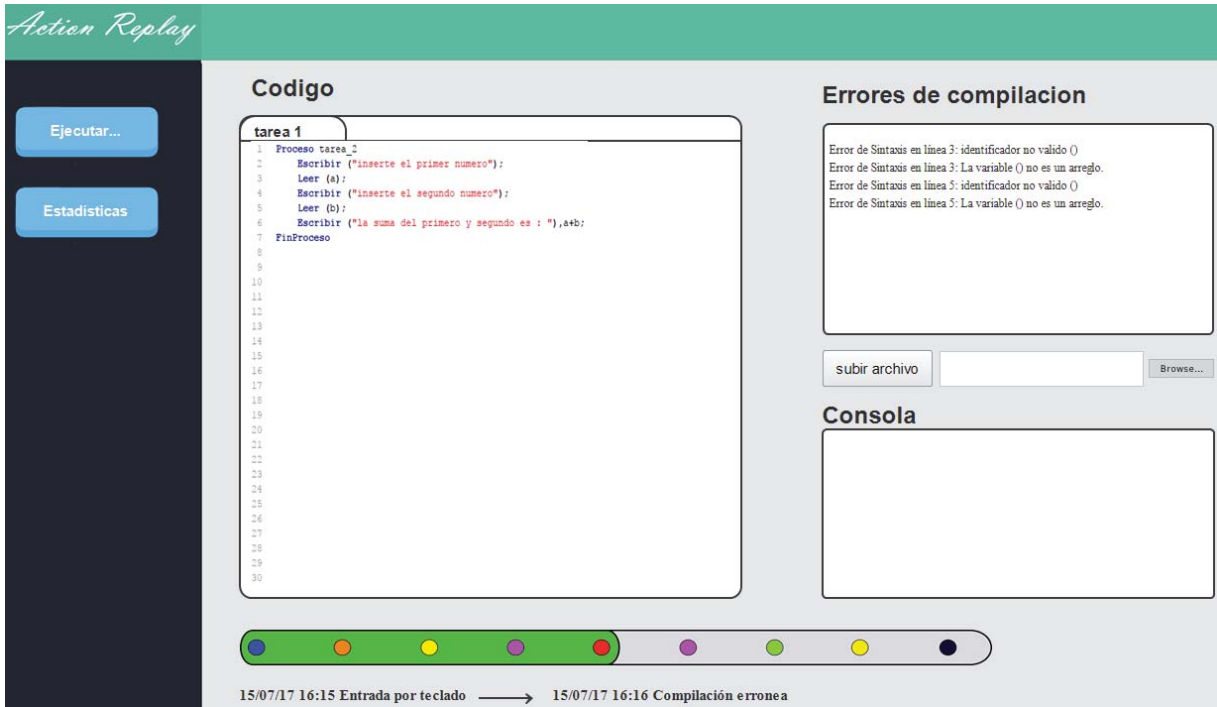


Figura 6.2 Mockup de error de compilación

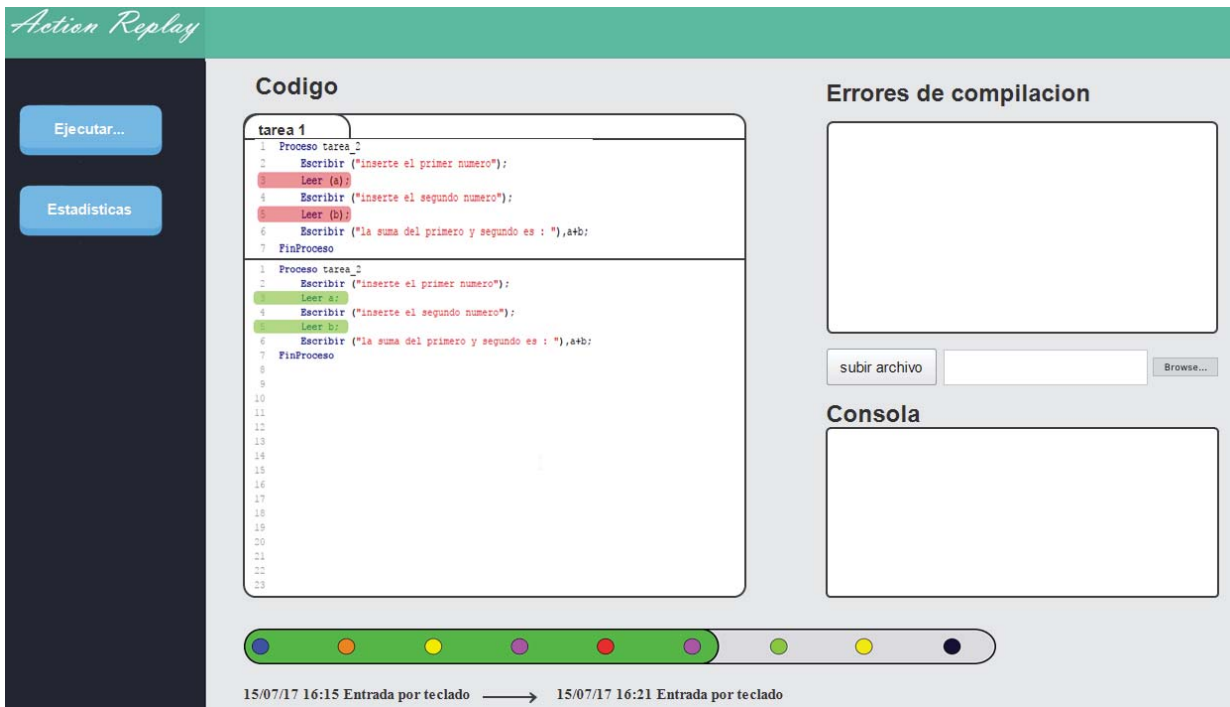


Figura 6.3 Mockup de cambios realizados al código

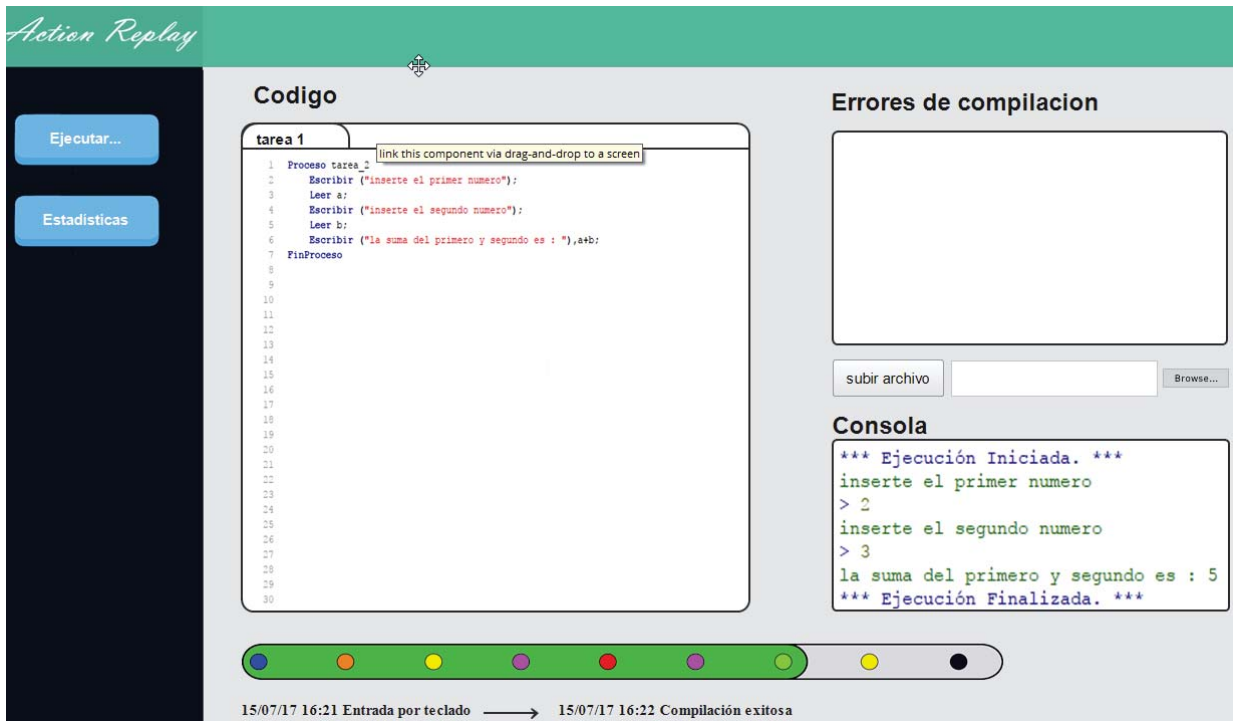


Figura 6.4 Mockup de ejecución de código por pantalla

6.3 Diseño de lectura de datos

La aplicación web se basa en la lectura de archivos XML el cual es un lenguaje de etiquetas el cual permite representar información estructurada en la web, de tal forma que la información pueda ser almacenada, procesada o visualizada por diversos tipos de aplicación y sistemas. El modelo de datos se base en XML Schema ya que permite instaurar una estructura de manera coherente y explicativa ayudando en su lectura y comprensión. También XML Schema permite ser manipulado mediante otros documentos XML logrando una mejora por sobre DTD, el cual solo es capaz de describir la estructura y sintaxis de los documentos XML.

Como el propósito de este proyecto es continuar el proyecto de Daniel Palma [7]. Se utilizará el XML Schema que almacena los archivos y registros desarrollado por él, como base para el desarrollo de una aplicación web capaz de leer, descifrar y comprender los archivos XML subidos a esta.

6.3.1 XML Schema

En el siguiente diagrama se muestra a grandes rasgos la estructura en que se almacenarán los registros en el XML.

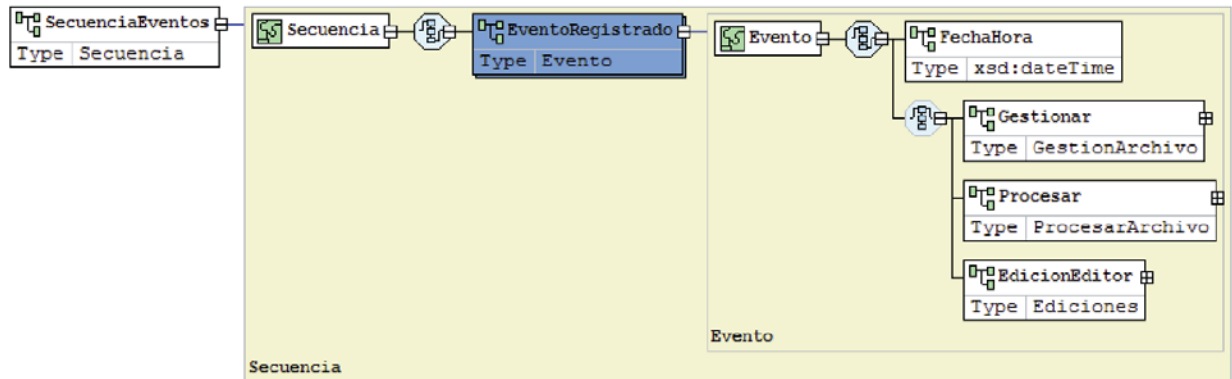


Figura 6.5 Esquema de la estructura de un archivo XML

Donde se pueden apreciar los siguientes elementos:

- **SecuenciaEventos**: Elemento del tipo de dato `Secuencia`, contiene una cadena indefinida de elementos del tipo eventos. Su propósito es anunciar la cantidad de eventos que posee el XML.
- **EventoRegistrado**: Elemento del tipo dato complejo `Evento`, el cual está conformado por:
 - **FechaHora**; contiene la fecha y hora exacta en la cual se realizó la acción
 - **Gestionar**; Elemento del tipo `GestionArchivo` el cual contendrá la información relacionada con la gestión de archivos almacenados, por ejemplo: abrir archivos y guardar archivos.
 - **Procesar**: Elemento del tipo `ProcesarArchivo`, posee información relacionada al código del archivo al momento de ejecutar una acción, en caso de errores se almacenará la cantidad de errores encontrados, el tipo de error y la descripción del error.
 - **EdicionEditor**: Elemento del tipo `Ediciones`, posee información relacionada con la edición o modificación del texto, almacenando cambios hechos al código, desde buscar, copiar y pegar hasta eliminación de líneas y comentarios.

En el **Anexo C** se podrán apreciar diferentes ejemplos de archivos XML generados a partir de la extensión realizada por Daniel Palma utilizando la aplicación PSeInt.

6.4 Diseño de base de datos

La aplicación web contará con un sistema de base de datos el cual almacenará los archivos XML cargados por el usuario, como también los comentarios y/o revisiones realizadas por el profesor encargado a evaluar los alumnos.

A continuación, se mostrará un diagrama en el cual representará las referencias entre las tablas de la base de datos.

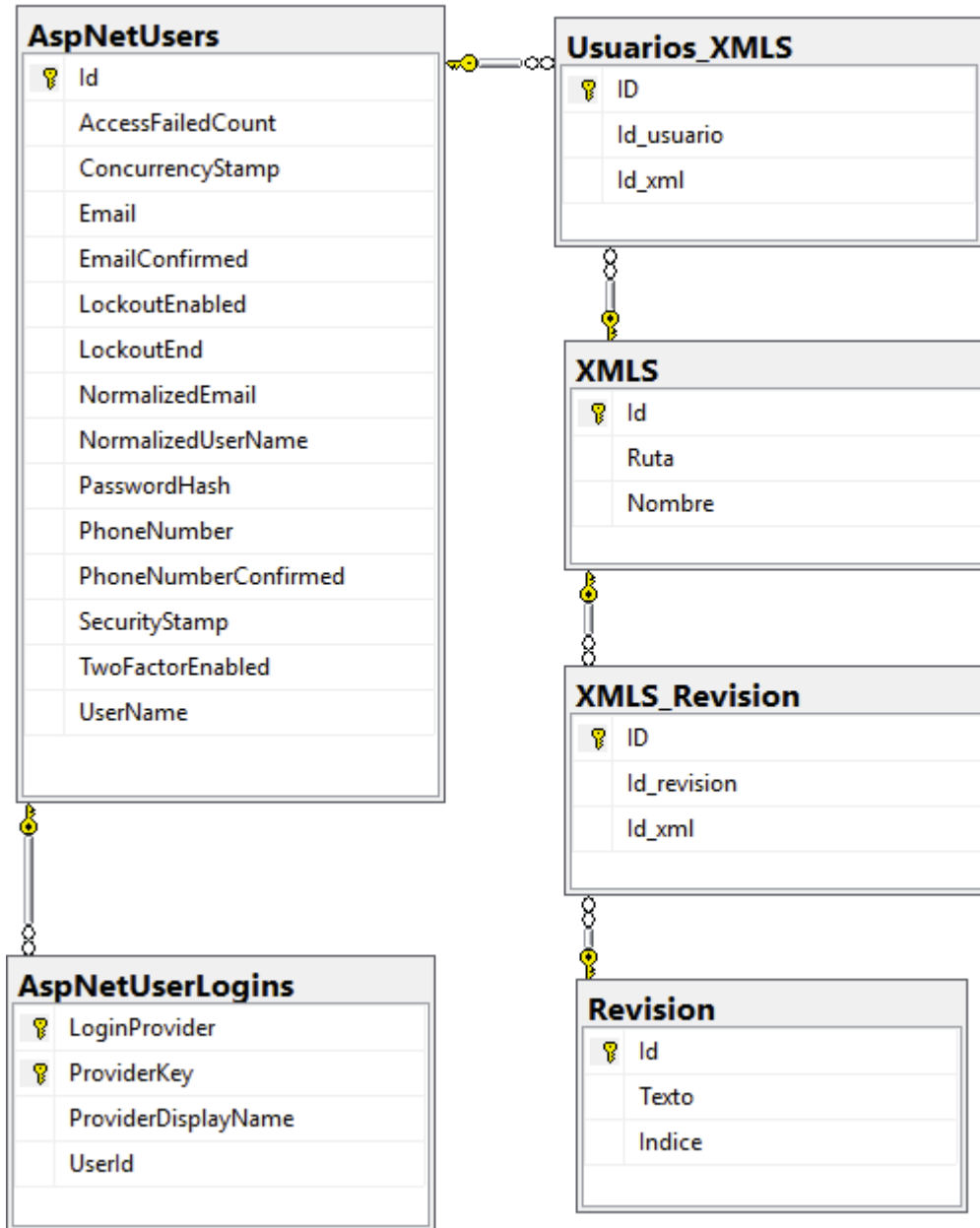


Figura 6 Diseño de la base de datos

En la tabla **AspNetUsers** se almacenará los datos del usuario, como correo, contraseña (la cual estará encriptada en todo momento), si la cuenta fue confirmada vía correo, su número de teléfono, nombre de usuario y otros datos los cuales se dejarán para una próxima entrega.

En la tabla **Usuarios_XMLS** contendrá un id del usuario perteneciente a dicho XML y el id del XML que le corresponde, esto debido a que un usuario puede ser dueño de múltiples XML como un XML puede ser dueño de múltiples usuarios.

En la tabla **XMLS** se almacenará los datos del XML como su nombre dentro de la plataforma y su ruta de almacenaje dentro de la base de datos.

En la Tabla **XMLS_Revision** contendrá un id de la revisión perteneciente a dicho XML y el id del XML que corresponde, esto debido a que un XML puede contener múltiples revisiones, como una revisión puede estar en múltiples XML.

En la tabla **Revision** se almacenará el texto de la revisión, como también un índice el cual se utilizará para saber en qué ubicación pertenece diseño comentario dentro de la línea de tiempo.

7 Conclusión

Como conclusiones, se puede señalar que la utilización de sistemas evaluativos personalizados puede ayudar en la autoevaluación por parte de los alumnos, pudiendo mejorar las habilidades personales de cada alumno. Con esto los profesores podrán crear planes de estudio personificados para cada curso o alumnos, con el fin de superar las metas impartidas al inicio del semestre escolar. De igual modo, se analizó que el uso de estas herramientas educativas aporta de gran manera en el correcto aprendizaje del alumnado.

El fin de este sistema fue ofrecer un feedback a los estudiantes durante sus sesiones de programación. Para esto se tuvo que utilizar un entorno de desarrollo con la capacidad de detectar las interacciones de los estudiantes y así generar un archivo XML el cual fue procesado posteriormente por la aplicación web mencionada en este informe. Este feedback ayudará a crear una conexión irrefutable entre el desarrollo de una aplicación y la toma de decisiones realizada por los estudiantes, con el fin de mejorar las capacidades cognitivas de este.

Durante el desarrollo de esta aplicación se pudo detectar la necesidad de crear un sistema de base de datos donde se pudiera almacenar los XML revisados dentro del entorno creado. Para así luego en una entrega posterior, se puedan entregar a los estudiantes como un informe los cuales podrán utilizar para estudiar y mejorar sus técnicas de programación.

La aplicación web diseñada en este trabajo permite entregar una herramienta para el análisis e iteración del usuario dentro de un entorno de programación para así entregar un feedback del código que fue desarrollando durante la sesión de programación, además de generar una línea de tiempo de las interacciones realizadas por el usuario dentro de un IDE, categorizar la información para entregarla en la línea de tiempo, filtrar las interacciones realizadas por el usuario, mostrar información implicara de los cambios realizados durante el desarrollo de la aplicación y tener un control de las revisiones almacenadas dentro de la aplicación facilitando su utilización.

De momento, la primera etapa de esta aplicación web, la cual cumple todos los requerimientos planteados a excepción del RF04 y el RF08, está finalizada. El RF04 no se pudo completar debido a que no se consideró adecuadamente el tiempo de desarrollo, producto de que se necesitaba desarrollar un compilador web de pseudocódigo, y esto llevo a que no se considere este requerimiento en lo que es la primera entrega de este proyecto, y el segundo no se desarrolló debido a que se dio prioridad a los demás objetivos del proyecto.

Finalmente, la aplicación web que se desarrolló será útil para los estudiantes de primer año de la Escuela de Ingeniería Informática de la Pontificia Universidad Católica de Valparaíso para refinar sus habilidades antes de entrar en el mundo de programación.

8 Referencias

- [1] Anónimo. (2015). Los exámenes, como instrumento de evaluación, podrían volverse obsoletos. El Comercio AFULL, www.elcomercio.com.
- [2] Paula Lacerna, Maria Angélica Mena. (2012). *Desafíos de evaluar el progreso en el Aprendizaje: La experiencia de SEPA UC*. 10/07/2017, de Colmee Sitio web: http://www.mideuc.cl/presentaciones_colmee/Paula_Lacerna_COLMEE_20121004.pdf
- [3] Educando Juntos. (2016). *Educación personalizada*. 15/07/2017, de Educarchile Sitio web: <http://www.educarchile.cl/ech/pro/app/detalle?id=138852>
- [4] Hurtado Echeverría, Carlos. (2001). *Educación personalizada y comunitaria*. Santiago, Chile: Trineo.
- [5] Educarchile. (2005). *Evaluación formativa*. 18/07/2017, de educarchile Sitio web: <http://www.educarchile.cl/ech/pro/app/detalle?ID=97665>
- [6] Elena Cano García. (2008). *La evaluación por competencias en la educación superior*. 20/07/2017, de Profesorado. Revista de Currículum y Formación de Profesorado Sitio web: <http://www.redalyc.org/pdf/567/56712875011.pdf>
- [7] Daniel Palma C. (2016) *Retroalimentación con un fundamento empírico para la enseñanza de fundamentos de programación usando PSeInt* (Tesis de pregrado). Universidad de Valparaíso, Valparaíso.
- [8] Ian Sommerville. (2005). *Ingeniería del software*. PEARSOB EDYCACUIB S.A MADRID: Addison Wesley.
- [9] Microsoft Office Word. <https://products.office.com/es/Word>
- [10] Microsoft Office Project. <https://products.office.com/es/Project>
- [11] Microsoft Office Visio. <https://products.office.com/es/Visio>
- [12] Editor de código fuente Notepad++. <https://notepad-plus-plus.org/>
- [13] Visual Studio. <https://www.visualstudio.com/es/>
- [14] Adam Freeman. (2012). *ASP.net MVC 4*. New York: Apress.
- [15] Juan Diego Gauchat. (2012). *El gran libro de HTML5, CSS3 y Javascript*. España: MARCOMBO.

Anexos

A Caso de uso narrativo

A continuación, en la siguiente sección del anexo se mostrarán los casos de uso narrativos mostrados anteriormente, en donde se identifica información relevante de ellos.

A.1 Caso de uso narrativo nivel del actor profesor

Tabla A.1 Caso de uso narrativo seleccionar evento

Caso de uso	Seleccionar eventos
Actores	Profesor
Propósito	Mostrar en una línea de tiempo todos los cambios realizados por el alumno durante el desarrollo de su aplicación.
Tipo	Primario
Descripción	El usuario abre la página web para luego cargar un archivo XML el cual contendrá la información necesaria para ser vista en un tiempo, en la cual podrá ir seleccionando los diferentes sucesos realizados durante el desarrollo de la aplicación (cortar, pegar, agregar, eliminar).
Referencias Cruzadas	Caso de uso: Si inicialmente no fue cargado algún XML el programa mantendrá la línea de tiempo bloqueada, en espera de que cargue algún XML.
Curso normal de los eventos	
Acción de Actores	Respuesta del Sistema
1. El usuario accede a la página de la aplicación.	
	2. El sistema carga la información de la aplicación.
3. El usuario sube el archivo a la página web	
	4. La página carga los datos del XML y los muestra por pantalla.
5. El usuario podrá seleccionar alguna acción o desplazar la línea de tiempo.	
	6. La aplicación mostrara todo lo desarrollado hasta el punto seleccionado en la línea de tiempo.
Curso Alternativos	
Línea 2,4. El sistema se encuentra fuera de línea, se contacta al Proveedor de servicio que la página web presenta fallas de carga.	
Línea 3. Datos ingresados no validos: se informa al usuario del error, indicando los campos incorrectos o el error.	

Tabla A.2 Caso de uso narrativo Filtrar suceso de eventos

Caso de uso	Filtrar sucesos de eventos
Actores	Profesor
Propósito	Filtrar los sucesos que muestra la línea de tiempo, para facilitar su análisis.
Tipo	Secundario
Descripción	El usuario seleccionara alguna opción de filtrado, produciendo que la línea de tiempo muestre información más específica y detallada a lo que se quiere buscar.
Referencias Cruzadas	Caso de uso: Si inicialmente no fue cargado algún XML el programa mantendrá la línea de tiempo bloqueada, en espera de que cargue algún XML.
Curso normal de los eventos	
Acción de Actores	Respuesta del Sistema
1. El usuario accede a la página de la aplicación.	
	2. El sistema carga la información de la aplicación.
3. El usuario sube el archivo a la página web	
	4. La página carga los datos del XML y los muestra por pantalla.
5. El usuario seleccionara algún nivel de filtrado en la línea de tiempo.	
	6. El programa mostrara solo la información solicitada por el usuario.
7. El usuario podrá seleccionar alguna acción o desplazar la línea de tiempo.	
Curso Alternativos	
Línea 2,4. El sistema se encuentra fuera de línea, se contacta al Proveedor de servicio que la página web presenta fallas de carga.	
Línea 3. Datos ingresados no validos: se informa al usuario del error, indicando los campos incorrectos o el error.	

Tabla A.3 Caso de uso narrativo ver información empírica

Caso de uso	Ver información empírica.	
Actores	Profesor	
Propósito	Mostrar en gráficos los cambios realizados durante el desarrollo de la aplicación (cortar, pegar, agregar, eliminar).	
Tipo	Secundario	
Descripción	El usuario podrá visualizar en gráficos de barra o torta la cantidad de líneas de código o fragmentos de códigos agregados, eliminados, cortados o pegados.	
Referencias Cruzadas	Caso de uso: Si inicialmente no fue cargado algún XML el programa mantendrá la línea de tiempo bloqueada, en espera de que cargue algún XML.	
Curso normal de los eventos		
Acción de Actores	Respuesta del Sistema	
1. El usuario accede a la página de la aplicación.		
	2. El sistema carga la información de la aplicación.	
3. El usuario sube el archivo a la página web		
	4. La página carga los datos del XML y los muestra por pantalla.	
5. El usuario seleccionara la opción de visualización de gráficos.	6. La aplicación mostrara los gráficos de los cambios realizados durante el desarrollo de la aplicación.	
Curso Alternativos		
Línea 2,4. El sistema se encuentra fuera de línea, se contacta al Proveedor de servicio que la página web presenta fallas de carga.		
Línea 3. Datos ingresados no validos: se informa al usuario del error, indicando los campos incorrectos o el error.		

Tabla A.4 Caso de uso narrativo ejecutar código

Caso de uso	Ejecutar código	
Actores	Profesor	
Propósito	Compilar el código mostrada en pantalla para su revisión.	
Tipo	Secundario	
Descripción	El usuario podrá compilar el código mostrado en pantalla para así revisar diferentes etapas del software durante el desarrollo de este.	
Referencias Cruzadas	Caso de uso: Si inicialmente no fue cargado algún XML el programa mantendrá la línea de tiempo bloqueada, en espera de que cargue algún XML.	
Curso normal de los eventos		
Acción de Actores	Respuesta del Sistema	
1. El usuario accede a la página de la aplicación.		
	2. El sistema carga la información de la aplicación.	
3. El usuario sube el archivo a la página web		
	4. La página carga los datos del XML y los muestra por pantalla.	
5. El usuario podrá seleccionar alguna acción o desplazar la línea de tiempo.		
6. El usuario seleccionara el lenguaje de programación y apretara el botón compilar.		
	7. La aplicación cargara los datos mostrados pantalla y realizara la compilación del código.	
Curso Alternativos		
Línea 2,4. El sistema se encuentra fuera de línea, se contacta al Proveedor de servicio que la página web presenta fallas de carga.		
Línea 3. Datos ingresados no validos: se informa al usuario del error, indicando los campos incorrectos o el error.		

Tabla A.5 Caso de uso narrativo ver errores de compilación

Caso de uso	Ver errores de compilación
Actores	Profesor
Propósito	Mostrar en la línea de tiempo los diferentes mensajes de error que experimento el usuario durante el desarrollo de la aplicación, para así analizar los posibles errores que cometió el alumno.
Tipo	Secundario
Descripción	El usuario abre la página web para luego cargar un archivo XML el cual contendrá la información necesaria para ser vista en una línea tiempo, en la cual podrá ir seleccionando los diferentes errores de compilación producidos durante el desarrollo de la aplicación para luego mostrarlos por pantalla.
Referencias Cruzadas	Caso de uso: Al recorrer la línea del tiempo la aplicación mostrara los diferentes mensajes de error o advertencia que se mostraron durante el desarrollo de la aplicación.
Curso normal de los eventos	
Acción de Actores	Respuesta del Sistema
1. El usuario accede a la página de la aplicación.	
	2. El sistema carga la información de la aplicación.
3. El usuario sube el archivo a la página web	
	4. La página carga los datos del XML y los muestra por pantalla.
5. El usuario podrá seleccionar algún error o desplazar la línea de tiempo donde se ubique algún error de compilación.	
	6. La aplicación mostrara todo lo desarrollado hasta el punto seleccionado en la línea de tiempo.
Curso Alternativos	
Línea 2,4. El sistema se encuentra fuera de línea, se contacta al Proveedor de servicio que la página web presenta fallas de carga.	
Línea 3. Datos ingresados no validos: se informa al usuario del error, indicando los campos incorrectos o el error.	

B Diagramas de secuencia

En este anexo se podrá observar los diagramas de secuencia de la aplicación.

B.1 Seleccionar evento

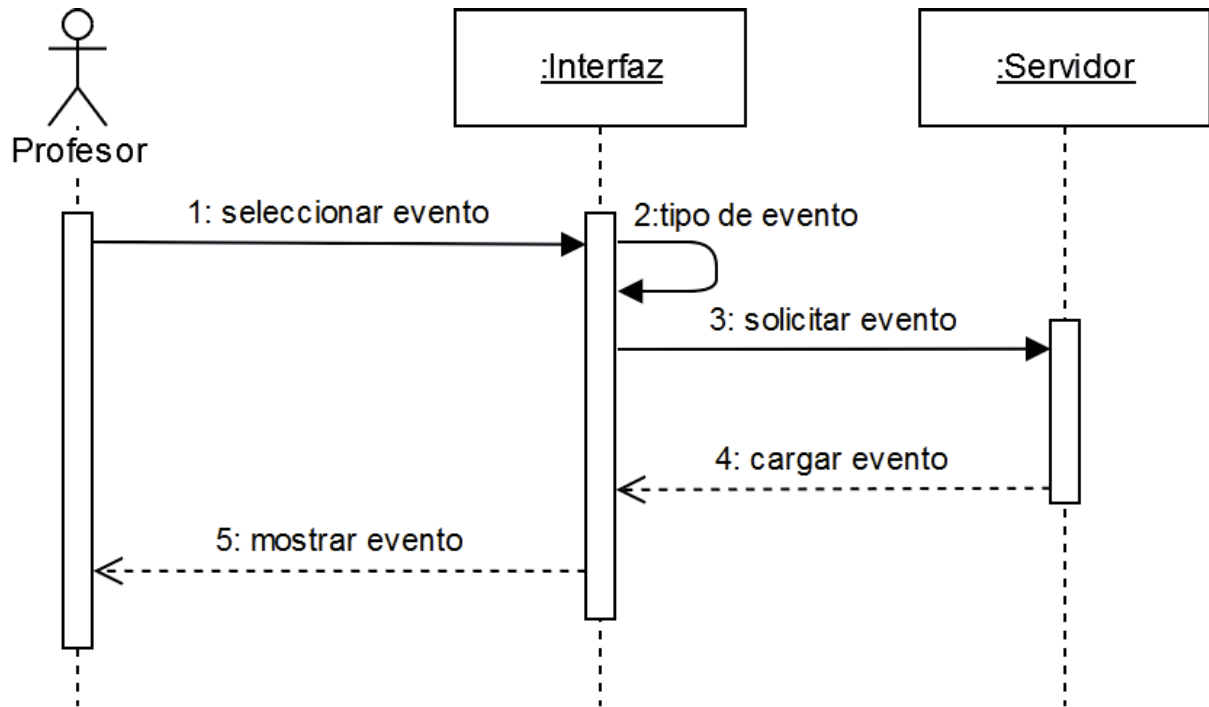


Figura B.1 Diagrama de secuencia seleccionar evento

B.2 Filtrar sucesos de eventos

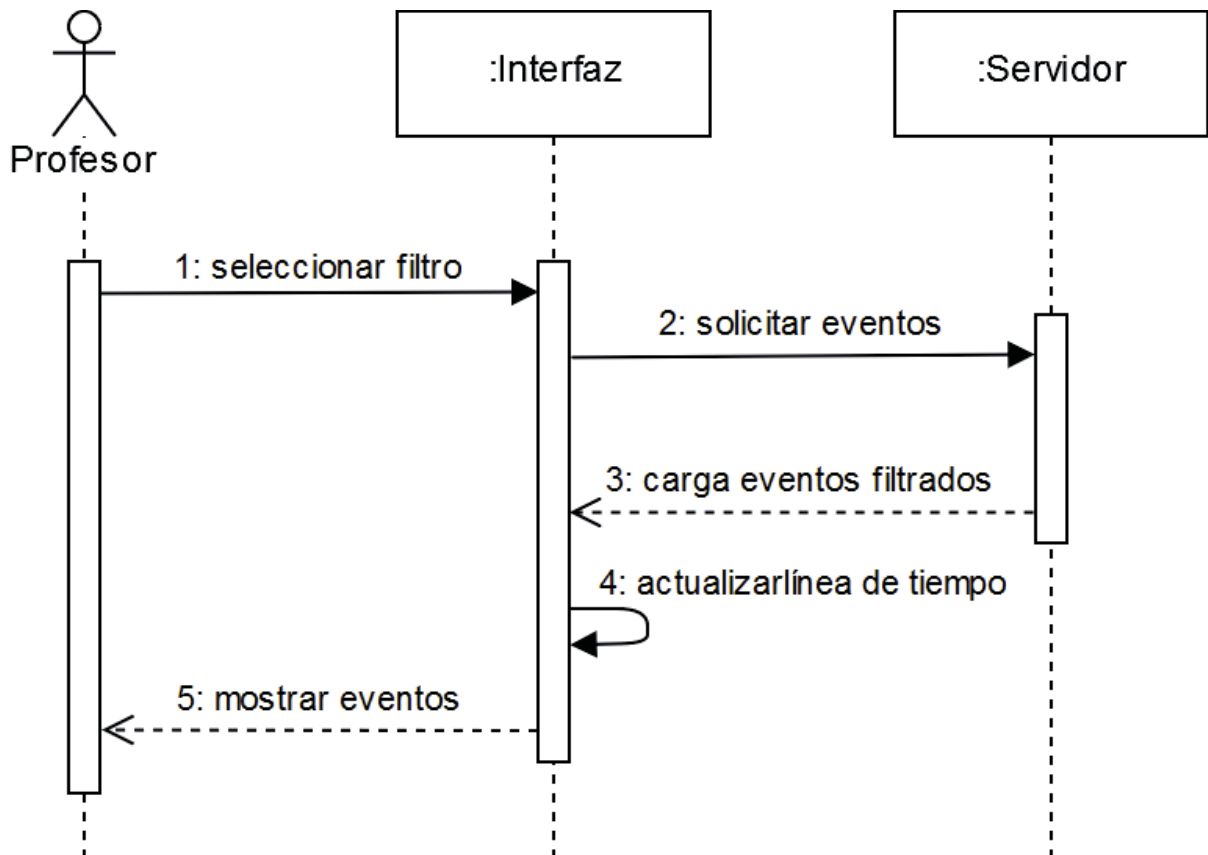


Figura B.2 Diagrama de secuencia filtrar sucesos de eventos

B.3 Ejecutar código

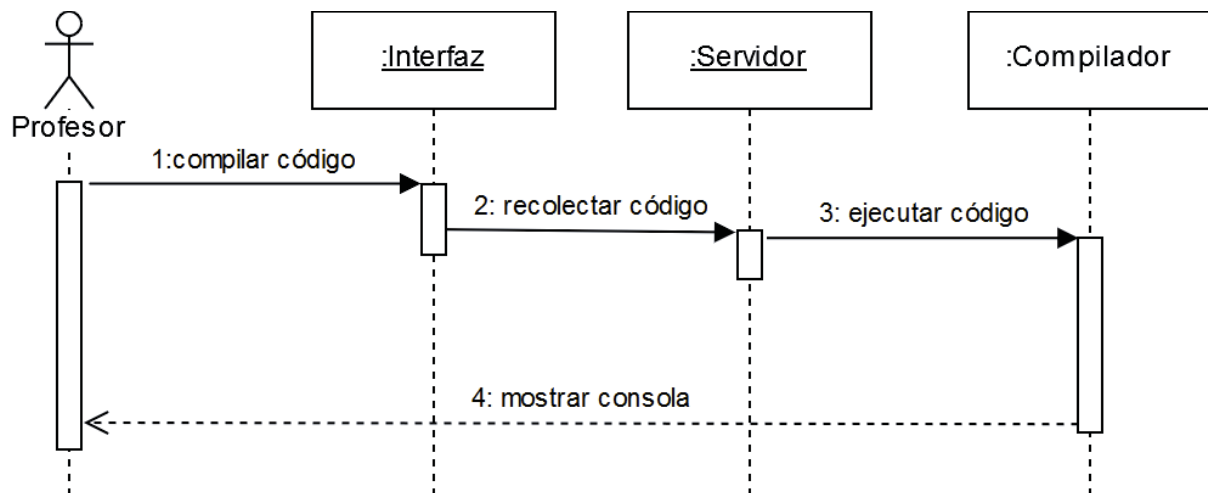


Figura B.3 Diagrama de secuencia ejecutar código

C Archivos XML

En el siguiente anexo se mostrarán diversos ejemplos de archivos XML generados a partir de la extensión de PSeInt.

C.1 Ejemplo 1

<https://1drv.ms/u/s!AsQVAT1XAYAJgcc5iml8Lda1s5zfGQ>

C.2 Ejemplo 2

<https://1drv.ms/u/s!AsQVAT1XAYAJgc1BUC9BDFYVnHqVjw>

C.3 Ejemplo 3

https://1drv.ms/u/s!AsQVAT1XAYAJggJUdEE0RA5Tuo_bxw