



Departamento de Informática
Universidad Técnica Federico Santa María



Entregable IV

Proyecto: The Adaptive Learning Wiki

Integrantes:

Nombres y Apellidos	Email	ROL USM
Yerko Cuzmar Ortiz	yerko.cuzmar.14@sansano.usm.cl	201473594-4
Felipe Montero Díaz	felipe.montero.13@sansano.usm.cl	201356565-4
Cristian Vallejos Pérez	cristian.vallejos.14@sansano.usm.cl	201473551-0

Post-Mortem Metodológico

Los resultados de la evaluación están adjuntos en un PDF dentro del mismo repositorio. Con respecto al análisis, el equipo de trabajo acordó lo siguiente:

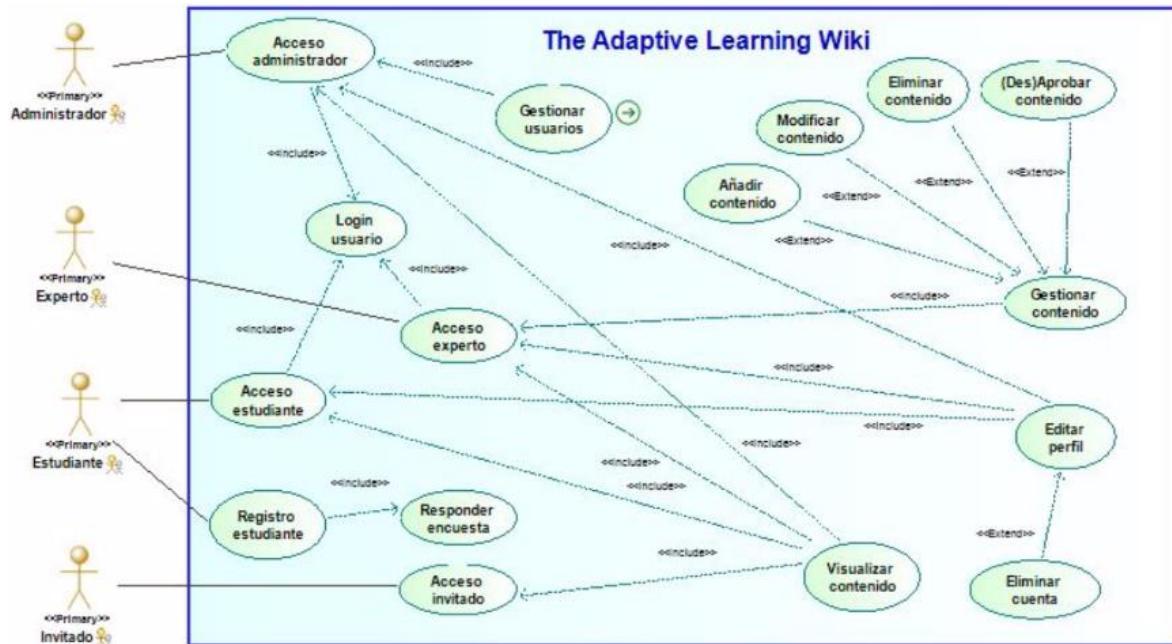
Del resultado obtenido en el documento, es posible ver que ninguna de las 10 áreas posee un nivel superior al medio. Esto se debe mayoritariamente a que, como equipo de trabajo, no implementamos conscientemente prácticas de desarrollo ágil durante la duración del proyecto. Creemos que se debe mayoritariamente a que el desarrollo se realizó de manera informal más que como un equipo de desarrollo de software profesional.

Entre las malas prácticas identificadas, se encuentran:

- No hay desarrollo continuo
- Desarrollo Incremental solo cerca de plazos de entrega
- Soluciones "Parche"
- Mala gestión de tareas
- Centralización respecto a ciertos features en específico
- Ninguna clase de reunión para establecer deficiencias
- Inequidad en la división de trabajo
- Entre muchas otras.

Como equipo, el primer paso para solucionar estas falencias de desarrollo es, de partida, organizarnos de forma en que todos los miembros compartamos una carga igualitaria. Una vez establecido un equipo de desarrollo organizado, el siguiente paso sería, incrementalmente, empezar a utilizar prácticas ágiles, una a la vez, analizando su impacto en el proceso de desarrollo. Si se considera que son un aporte medible para el equipo, mantenerla y continuar con otra, y si no, ignorarla y seguir de todas formas. De esta manera, sería posible agilizar enormemente nuestro proceso de desarrollo como equipo para futuros proyectos.

Diagrama de Casos de Uso (final)



El diagrama de Casos de Uso no se vio modificado en relación al original, puesto que el equipo de trabajo consideró que cubre las funcionalidades más importantes dentro de las descritas por el cliente durante el diseño y desarrollo del proyecto.

Patrones de diseño y Frameworks (final)

Intención	Patrón de Diseño	Razonamiento
Se desea mostrar en el diagrama de clases la interacción entre los componentes del sistema, mediante el uso de controladores que comuniquen la vista final y el modelo.	MVC	Las clases del dominio deben poder comunicarse entre sí de manera adecuada, al mismo tiempo que el usuario activo debe poder buscar/visualizar Contenido y acceder a los distintos recursos de la plataforma. MVC es práctico en este sentido, pues permite acceder fácilmente a los distintos objetos de la aplicación.

Se busca esconder las interacciones complejas del sistema detrás de métodos que simplifiquen dicha interacción	Facade	La búsqueda de información (<i>por parte del sistema</i>) según los parámetros brindados por el experto, debe verse simple para este último, de modo que la verdadera complejidad del sistema de búsqueda -y el sub-sistema de filtrado- debe ser imperceptible para quien gestiona la validación de contenidos en la aplicación web.
Se busca mostrar en el diagrama de clases el manejo de los requests de autenticación de usuarios en el sistema	Front Controller	La necesidad de resolver las peticiones de autenticación de usuarios es clave en la plataforma. Además, es fundamental poder delegar ciertas solicitudes que deben ser resueltas por otros controladores, lo cual se ve facilitado por la inclusión de este patrón de diseño.

Los patrones de diseño seleccionados para esta entrega son los mismos que los del Entregable II. Esto ocurre debido a que el equipo de trabajo no incorporó nuevos frameworks al entorno de trabajo durante el transcurso del desarrollo del proyecto.

Modelo de Dominio y Diagrama de Clases (final)

Modelo de Dominio

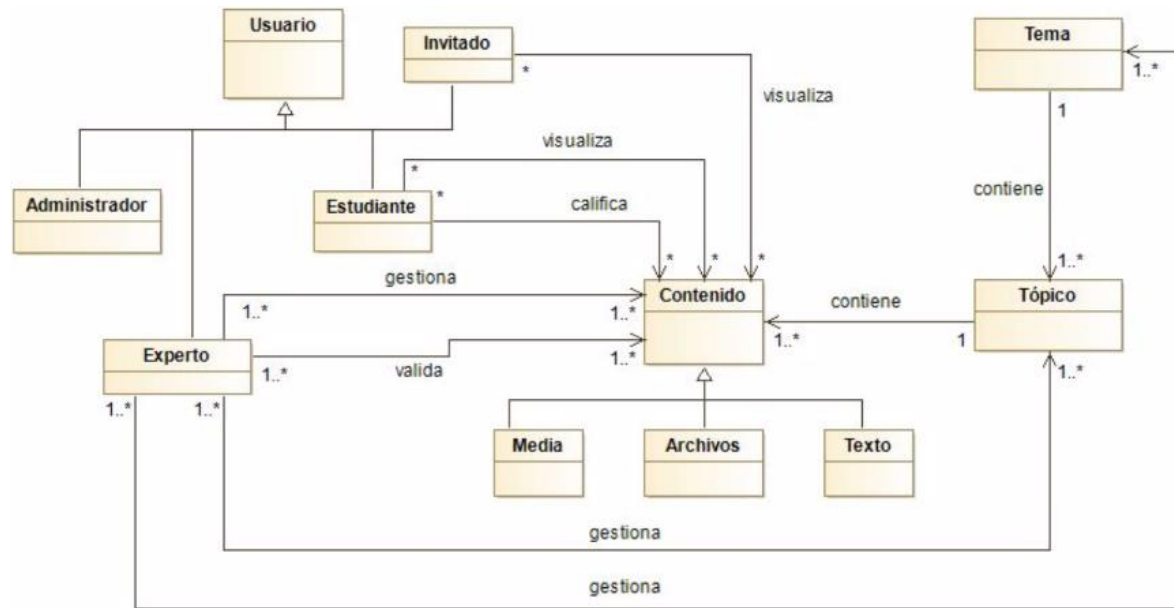
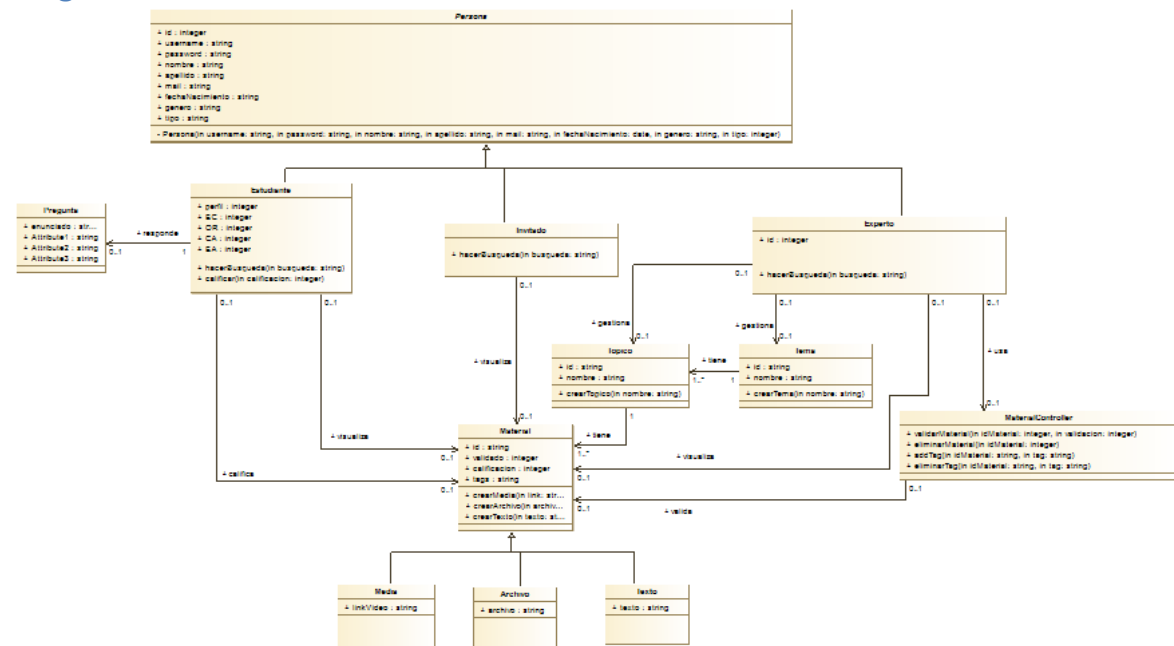


Diagrama de Clases



Tanto el modelo de Dominio como el de Clases no se vieron alterados respecto a los entregables anteriores, debido a que el equipo de trabajo consideró que abarcan las entidades y relaciones más importantes relacionadas con el dominio del problema, además de permitir una buena distribución de la información dentro de la base de datos que se utilizó.

Pruebas de Software (actualización)

Respecto a esta sección del informe, nuestro grupo consultor QA no se manifestó durante el periodo correspondiente, por lo que como equipo desarrollador nos es imposible responder de forma correspondiente, ya que necesitamos al grupo consultor para verificar si los cambios hechos al proyecto solucionaron las falencias encontradas por ellos en el entregable anterior.