



Departamento de Informática
Universidad Técnica Federico Santa María



Requisitos de Software

Proyecto: BuscaTek



BUSCATEK

Integrantes:

Nombres y Apellidos	Email	ROL USM
Tomás Gómez Molina	tomas.gomez.13@sansano.usm.cl	201373078-7
Diego Jara Caballero	diego.jara.13@sansano.usm.cl	201304530-8

Contexto del proyecto

Objetivo de proyecto

El objetivo del proyecto es crear un software que permita buscar material de estudio adecuado para los cursos de física de la UTFSM, ya sean textos, imágenes, audios, etc., y que además se adapte a los estilos de aprendizaje de Kolb.

Resumen del proyecto

La búsqueda de un material de apoyo académico adecuado es un problema constante durante la vida académica de la mayoría de los estudiantes de universidades, institutos o CFT. Y sobre todo en la UTFSM, en donde la lectura y ejercitación son prácticas indispensables para la aprobación de los ramos más matemáticos. Lo que nos lleva a Física General, probablemente una de las ramas más difíciles (y temidas) de la universidad. Es esto último lo que nos motiva a desarrollar un software de búsqueda de material de apoyo para los ramos de física, el cual, además de ser preciso y certero en el momento de entregar resultados, debe ser rápido, escalable y se debe adaptar a los Estilos de aprendizaje de Kolb, con el objetivo de entregar resultados que previamente hayan sido evaluados por un experto, y que además se organicen de manera personalizada de acuerdo al perfil de cada usuario.

Desarrollo del Prototipo

Para el desarrollo del prototipo, se buscó la implementación de una base donde trabajar en el buscador que es lo que tiene más prioridad dentro del proyecto, por lo tanto, se implementaron las funcionalidades que permitían este avance.

Las funcionalidades son las siguientes:

1.- Registro de usuarios:

Se implementó un registro de usuarios, que por el momento no distingue entre profesor y estudiantes. El registro logra obtener los datos requeridos y guardarlos correctamente en la base de datos MongoDB utilizada.

2.- Inicio de sesión:

Se implementó un inicio de sesión para los usuarios, estos ingresan en un perfil único, que muestra sus datos personales. El perfil, al igual que el registro, fueron implementados con las herramientas Express y Node.js, estas poseen un sistema de Passport que permite de manera rápida implementar la autenticación.

3.- Realizar búsqueda:

Se implementó un buscador, utilizando la herramienta de Google Custom Search, esta permite realizar búsqueda sobre sitios web predeterminados. Se está desarrollando que el profesor controle estos sitios web sobre los cuales se puede buscar información.

4.- Mostrar resultados de búsqueda:

Con el mismo motor de búsqueda implementado, se logró mostrar los resultados de búsqueda sobre las páginas predeterminadas (por ahora predeterminadas por el grupo de trabajo).

(Se adjuntan imágenes del prototipo en el Anexo)

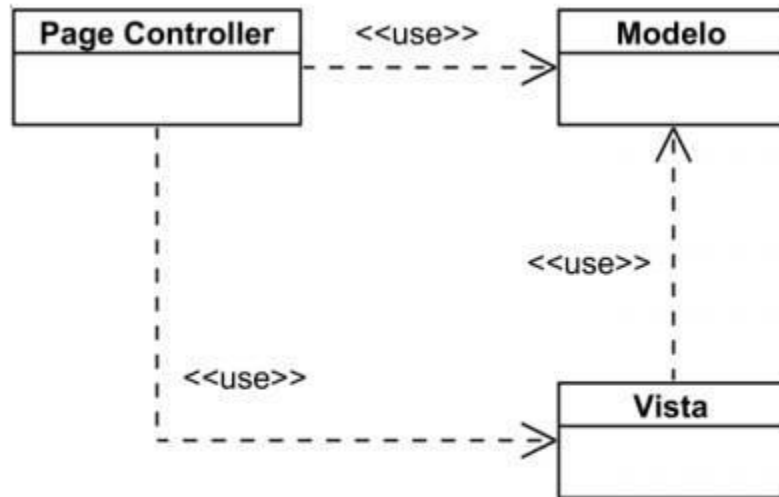
Selección de Patrones de Diseño

Intención	Patrón de Diseño	Razonamiento
Se desea mostrar en el Diagrama de Clases la gestión en el Sistema cuando un estudiante desea buscar material de estudio en la web y el resultado de la búsqueda debe ser organizado según el perfil del estudiante.	Page Controller	El razonamiento tiene que ver con la elección de MEAN para el desarrollo de nuestro proyecto. El stack MEAN sugiere un enfoque en MVC ya que su estructuración interna, que es en “paquetes”, invita a la modularidad del proyecto. Bajo esta tabla se puede ver como la estructura de archivos del proyecto indica similitud con MVC.
En el Diagrama de Clases debe existir la capacidad de mostrar el resultado de la búsqueda de manera inmediata al usuario, sin tener que recurrir a un objeto extra para la reproducción de este.	Template View	
Se requiere que en el Diagrama de Clases se muestre la gestión eficaz de un conjunto de transacciones que serán reiterativas, ya que se espera que los estudiantes busquen datos relacionados con la asignatura que estén cursando y por lo tanto se puedan agrupar ciertos tipos de búsqueda obteniendo resultados rápidos.	Transaction Script	

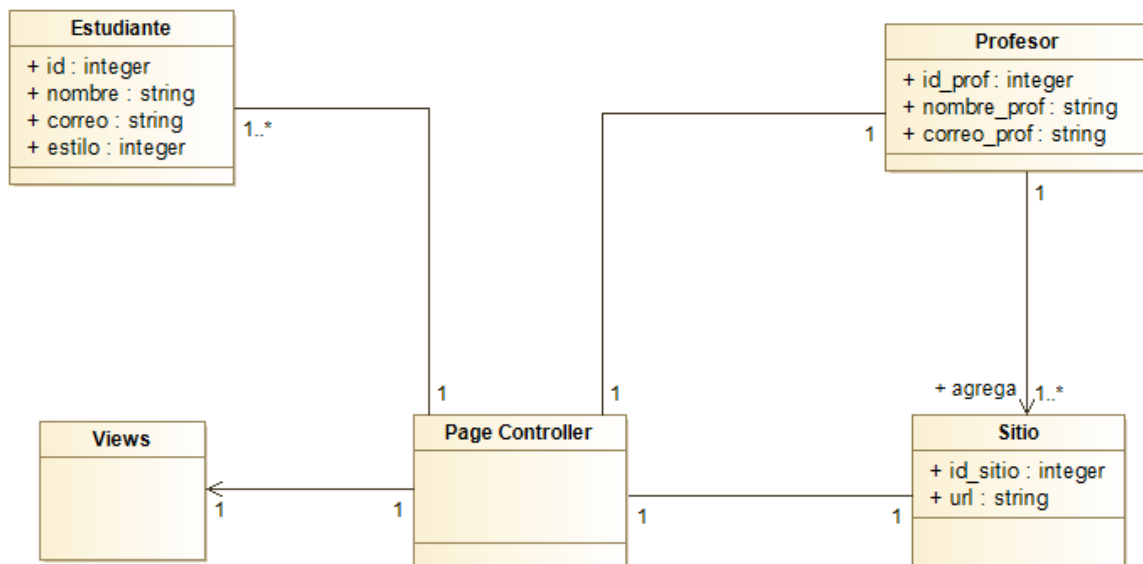
Estructura de archivos sugerida por MEAN.IO y MEAN.JS (nosotros usamos MEAN.IO).

```
-server
--- config           # Archivos de configuración
--- controllers      # Lógica del servidor (nuestro Page Controller)
--- models           # modelos
--- routes           # api REST
--- views            # vistas html (nuestro Template View)
```

Bosquejo MVC



Creación de Diagrama de Clases



¿Qué clases son necesarias?

Estudiante: esta clase corresponde a un estudiante que desea buscar contenido en el sistema. Como atributos tiene:

1. id: identificador de tipo entero
2. nombre: nombre del estudiante (string)

3. correo: correo del estudiante (string)
4. estilo: estilo de aprendizaje asociado al estudiante y que no servirá para diferenciar el contenido mostrado (entero)

Profesor: esta clase corresponde a un profesor que administra los sitios validados. El profesor puede hacer CRUD de los sitios. Como atributos tiene:

1. id_profesor: identificador de tipo entero
2. nombre_prof: nombre del profesor
3. correo_prof: correo del profesor

Sitio: esta clase representa a los sitios validados por el profesor y que influirán en el motor de búsqueda. Sus atributos son:

1. id_sitio: un identificador de tipo entero
2. nombre_sitio: nombre del sitio
3. url_sitio: la dirección de internet del sitio

Creemos que estas clases son primordiales para nuestro proyecto y dudamos que sean cambiadas, pero no descartamos que se agreguen otras clases, cambiando entonces el diagrama final.

¿Cómo estarán conectadas estas clases?

La clase Profesor se conecta con la clase Sitio ya que un profesor puede agregar múltiples sitios, pero un sitio es agregado solo por un profesor (en el diagrama de clases esto se puede apreciar mejor).

¿Qué queremos saber sobre los objetos?

Esta detallado en el punto “¿Qué clases son necesarias?”

Diagramas de Secuencia

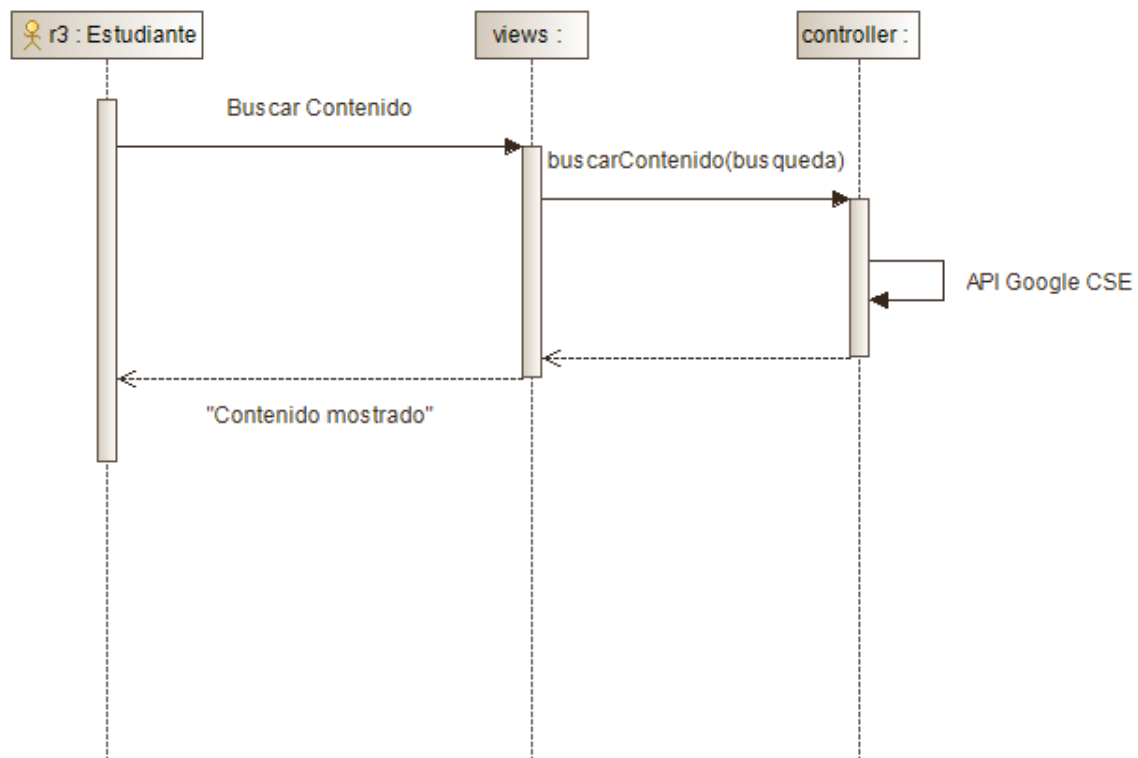


Diagrama de secuencia 1: Buscar contenido

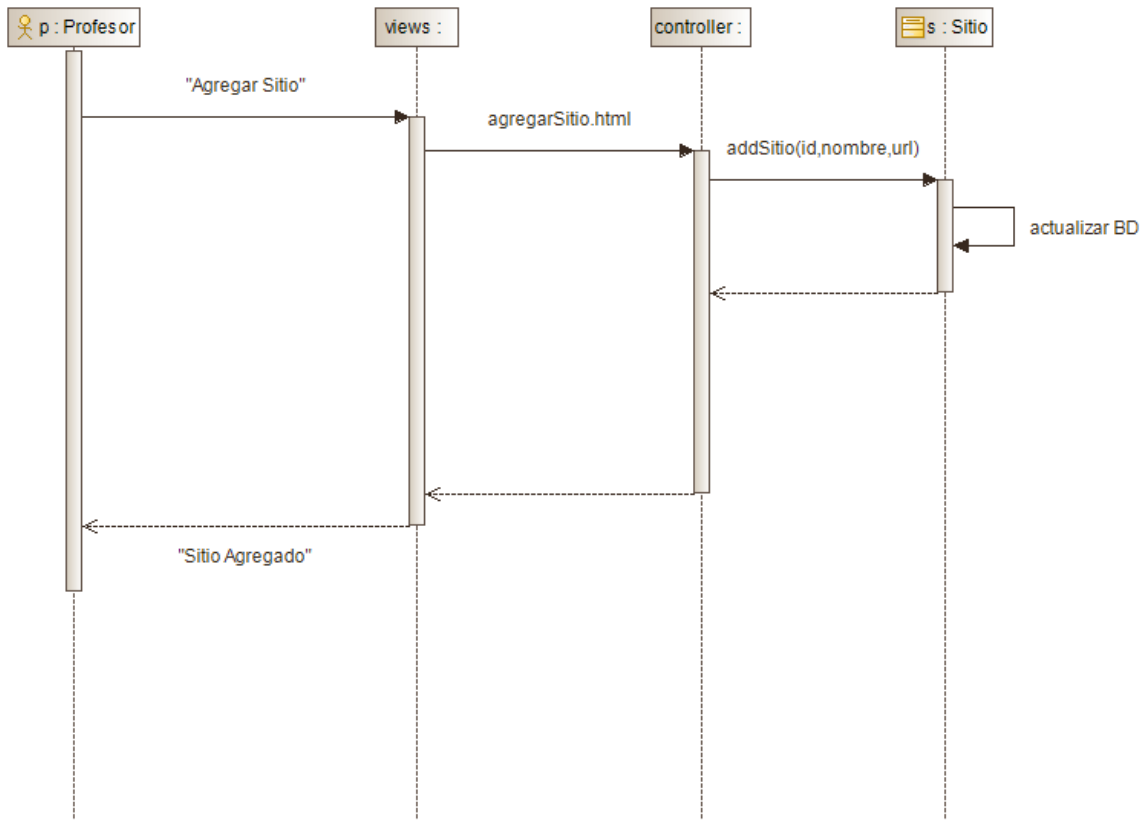


Diagrama de secuencia 2: Agregar Sitio

Análisis de Trade-off

Puesto que no se realizaron las presentaciones, no se logró obtener un feedback del cliente. El grupo se acercó a intentar obtener un feedback, pero no se logró concretar la reunión debido a inconvenientes de tiempo con el cliente.

Debido a la razón antes presentada, como grupo, se creó una funcionalidad que el cliente podría requerir (según lo observado en las reuniones anteriores). Esto para tener una sección de Trade-off y cumplir con el segundo entregable.

La nueva funcionalidad será agregar un *componente que recolecte los enlaces más recientes y repetidos y los muestre* a los estudiantes que van a realizar una búsqueda.

Question.

Al realizar la pregunta, ¿Qué cambios se deben realizar en el Sistema para agregar un nuevo componente que recolecte los enlaces más recientes y repetidos y los muestre a los estudiantes?

Se obtienen las siguientes respuestas u *opciones*.

Options.

O1: Agregar controlador que cada cierto tiempo registre las últimas cinco búsquedas en la BD, y las recomiende a los usuarios que desean buscar material de clases dentro de ese tiempo.

O2: Modificar la clase de Sitios para agregar un conteo de cada enlace buscado y luego seleccionado por el usuario, para luego mostrar los que tengan mayor puntaje de búsqueda.

O3: Agregar un calificador de enlace, donde el estudiante califica si el enlace fue útil o no lo fue, luego se muestran los enlaces más votados por medio de un controlador.

Luego para comparar las opciones se definen los criterios de comparación.

Criteria.

C1: Dificultad

C2: Confiabilidad

C3: Eficiencia

Finalmente se realiza el análisis o “Trade Off”, para asignar una correlación a cada criterio a partir de las opciones. El puntaje que se utilizará será el siguiente:

++ (Muy fuerte)

+ (Fuerte)

0 (No disponible)

- (Baja)

-- (Muy baja).

La tabla que contiene los puntajes asignados para cada criterio sobre cada opción es la siguiente:

La interpretación del análisis sobre la tabla es el siguiente:

Criterio\Opciones	O1	O2	O3
C1	+	+	++
C2	-	+	++
C3	-	0	0

En conclusión, la opción seleccionada por el grupo es la opción O2, puesto que su dificultad es igual o menor que las demás, es una forma confiable de obtener los datos

más útiles para mostrar como recomendación de búsqueda y por último su eficiencia no es mala, ya que depende de la implementación, pero está sobre o igual a las demás opciones. En esta elección se compromete quizás un poco con la dificultad, pero es una dificultad que no se escapa del marco de lo que es posible por realizar y se gana confiabilidad en la entrega de un resultado que es óptimo y útil para el estudiante, acerca de su eficiencia, depende de la implementación.

Anexo


Imagen 1 de prototipo – Registro

Detalles de Registro


[Volver](#)

[Contactar Soporte BuscaTek USM](#)
BuscaTek Copyright © 2016

Imagen 2 de prototipo – Inicio de Sesión



Inicio su Sesión



[Registrarse](#)

[Contactar Soporte BuscaTek USM](#)
BuscaTek Copyright © 2016

Imagen 3 de prototipo – Perfil de Usuario



Imagen 4 de prototipo – Mostrar Búsqueda

