

ESCUELA TÉCNICA SUPERIOR DE INGENIERÍA DE
TELECOMUNICACIÓN
UNIVERSIDAD POLITÉCNICA DE CARTAGENA



Proyecto Fin de Carrera

**Desarrollo de una aplicación web para la evaluación de los estilos de aprendizaje
con la librería Javascript, Famo.us**



AUTOR: María Teresa López Palacios
DIRECTOR: Diego Alonso Cáceres,
Francisca Rosique Contreras
Septiembre / 2014

Índice de contenido

.....	4
Capítulo 1. Introducción.....	4
1.1 Motivación.....	4
1.2 Objetivos.....	4
1.3 Estructura del trabajo.....	5
Capítulo 2. Estudio de la técnica.....	6
2.1 Famo.us.....	6
2.1.1 Introducción a Famo.us.....	6
2.1.2 Cómo trabaja Famo.us.....	6
2.1.3 API de Famo.us.....	7
2.1.4 Render Tree.....	8
2.1.4.1 Creando el Render Tree.....	8
3.1.4.2 Extendiendo el Render Tree.....	8
2.1.4.3 Tipos de nodos.....	9
2.1.4.4 Encadenamiento de nodos.....	10
2.1.4.5 Vistas.....	11
2.1.4.6 Resumen y conclusiones.....	12
2.1.5 Comparativa con otras librerías 3D.....	12
2.1.6 Conclusiones.....	13
2.2 Felder & Soloman.....	14
2.2.1 Aproximación al uso de los estilos de aprendizaje como enfoque para elevar el rendimiento de alumnos universitarios de la modalidad de educación a distancia.	14
2.2.1.1 Introducción.....	14
2.2.1.2 Objetivos y tareas de investigación.....	15
2.2.1.3 Herramientas para la determinación de los estilos de aprendizaje.....	15
2.2.3.4 Discusión de los resultados.....	17
2.2.3.4.1 Resultados del uso del cuestionario basado en la PNL (Programación Neurolingüística).....	18
2.2.3.4.2 Resultados del cuestionario Felder & Soloman.....	19
2.2.3.4.3 Conclusiones y resultados.....	21
2.2.2 Análisis de las dimensiones de estilo de aprendizaje según Felder-Silverman.....	21
2.2.2.1 Introducción.....	21
2.2.2.2 Modelo de estilos de aprendizaje de Felder-Silverman.....	22
2.2.2.3 Colección y análisis de datos.....	22
2.2.2.4 Índice de estilos de aprendizaje (ILS).....	23
2.2.2.5 Resultados del estudio.....	24
2.2.2.6 Conclusiones e investigaciones futuras.....	27
Capítulo 3. Aplicación web. Test de Felder & Soloman.....	28
3.1 Análisis y diseño.....	28
3.1.1 Necesidad de la aplicación.....	28
3.1.2 Requisitos de usuario.....	28
3.1.3 Requisitos de software.....	29
3.1.4 Casos de uso.....	29
3.2 Implementación.....	35
3.2.1 Instalación de Famo.us.....	35
3.2.2 Lenguajes de programación y estructura del proyecto.....	36
3.2.3 Creación de la aplicación web.....	38
3.2.3.1 HeaderComponentLayout.....	39
3.2.3.2 Test.....	50

3.2.3.3 Gráfico 3D. El cubo.....	56
Capítulo 4. Conclusiones y trabajos futuros.....	63
ANEXOS.....	64
REFERENCIAS BIBLIOGRÁFICAS.....	69

Capítulo 1. Introducción

En este capítulo se dará entrada al resto del documento, proporcionando una visión general sobre el proyecto y explicando los objetivos y motivaciones que llevaron a la realización del mismo, así como una pequeña descripción del resto de contenidos.

1.1 Motivación

Las motivaciones que movieron la realización de este proyecto nacen de dos naturalezas muy diferenciadas. Una, parte del interés en la parte técnica de desarrollo y otra, parte de la curiosidad en los distintos tipos de aprendizaje de los alumnos de distintas ingenierías.

La parte que aborda el desarrollo técnico de la aplicación, nace de un interés en un desarrollo en tiempo real de la aplicación, donde se pueden ver los cambios simplemente al guardar el nuevo código. También entra en juego el desarrollo de gráficos en 3D de una manera simplificada.

Otra parte muy importante que se estudia en este proyecto es la diferenciación entre los estilos de aprendizaje de los estudiantes de distintas carreras de ingeniería. Este estudio, intenta clasificar a los alumnos según sus preferencias y así clarificar que la manera que tienen de estudiar y aplicar los conocimientos es la más eficiente para ellos. Está comprobado que cada persona tiene una manera en la que la información le llega de forma más clara y es importante tenerlo en cuenta, tanto por parte del alumno, como por parte del profesor. Por parte del alumno, para poder complementar la información de clase, pidiéndole al profesor material que se adecúe a sus preferencias, y modificando los hábitos de estudios para acomodarlos a éstas. El profesor, también obtendría información sobre los estilos de aprendizaje de sus alumnos y así poder adecuar las clases con distintos tipos de material, a las necesidades de éstos.

1.2 Objetivos

El presente Proyecto Fin de Carrera se centra en el estudio de la realización de una aplicación desarrollada con la librería JavaScript, Famo.us. Esta aplicación se centra en la realización de un test para determinar los estilos de aprendizaje de los alumnos según el estudio de Felder & Soloman. Para ello, se realiza un estudio exhaustivo sobre estos estilos de aprendizaje y a partir de ahí, se crea una aplicación donde se pueda realizar el test, mostrar los resultados, obtener información sobre las cuatro dimensiones diferenciadas de aprendizaje y una vez vistos los resultados, poder obtener consejos sobre como adecuar los hábitos de estudio según sean los resultados obtenidos.

Para la realización, debemos estudiar el framework, como funciona, su relación con distintas tecnologías y su desarrollo.

El objetivo principal es la realización de gráficos en 3D en los cuales se pueden mandar y recibir datos, probando en tiempo real.

1.3 Estructura del trabajo

En general, el trabajo tendrá la siguiente estructura:

1. Introducción: En este capítulo se realiza una introducción al proyecto, describiendo la motivación y los objetivos del mismo, sobre qué trata, qué problema se resuelve y un desglose del resto del documento.

2. Estudio de la técnica: En este apartado se hablará sobre las distintas librerías que pueden aportar un resultado similar al nuestro, las comparativas entre ellas y se obtendrán las conclusiones de por qué Famo.us es elegida finalmente.

También se hablará sobre distintos test para la obtención de estilos de aprendizaje, cómo han evolucionado y como se ha llegado a obtener el de Felder & Soloman.

3. Aplicación Test Felder & Soloman: En este apartado se explica el diseño y la implementación de la aplicación, ofreciendo una descripción de los elementos utilizados.

4. Conclusiones y futuras mejoras: En este capítulo se resumen las conclusiones obtenidas del estudio previo. Se confirmará que los objetivos han sido alcanzados y se plantearán mejoras para un desarrollo futuro.

5. Anexos

6. Referencias bibliográficas

Capítulo 2. Estudio de la técnica.

En este apartado se hablará sobre las distintas librerías que pueden aportar un resultado parecido, las comparativas entre ellas y obtendremos las conclusiones de por qué Famo.us es elegida finalmente. También se hablará sobre distintos test para la obtención de estilos de aprendizaje, cómo han evolucionado y como se ha llegado a obtener el de Felder & Soloman.

2.1 Famo.us

En este apartado se lleva a cabo la descripción y utilidad de la librería JavaScript, Famo.us. También serán vistas las alternativas que existen en el momento actual para llegar a obtener las conclusiones sobre cuál de ellas es la más apropiada para realizar la aplicación web del test Felder & Soloman.

2.1.1 Introducción a Famo.us

En primavera de 2014, la comunidad de desarrolladores JavaScript, presentó la versión beta del framework de código abierto Famo.us, el cual promete eliminar los cuellos de botella que impedían a JavaScript dominar la escena de desarrollo móvil, ya que ofrecía un interfaz de usuario lento y una pobre experiencia a los clientes.

Famo.us dirige la unidad de procesamiento de gráficos de hardware (GPU) en el dispositivo móvil para conseguir la mayor velocidad de fotogramas de representación posible, y se añade un motor más sofisticado para una experiencia de usuario gratificante. Los desarrolladores de JavaScript ya no se encuentran en desventaja en comparación con Objective-C, Swift, o los desarrolladores de Java cuando se crean interfaces de usuario para aplicaciones móviles.

Ha de destacarse también que Famo.us utiliza la tecnología WebGL. Ésta es una especificación estándar que está siendo desarrollada actualmente para mostrar gráficos en 3D en navegadores web. Permite mostrar gráficos en 3D acelerados por hardware (GPU) en páginas web, sin la necesidad de plug-ins en cualquier plataforma que soporte OpenGL 2.0 u OpenGL ES 2.0. Técnicamente es un API para JavaScript que permite usar la implementación nativa de OpenGL ES 2.0.

En los apartados posteriores, se presentan los conceptos fundamentales de Famo.us y será explorado su diseño.

2.1.2 Cómo trabaja Famo.us

La animación se crea por la visualización rápida de las páginas sucesivas (marcos) con elementos cambiados. La velocidad de fotogramas es el número de fotogramas que se muestran por segundo. Una velocidad de fotogramas alta crea la ilusión óptica de movimiento a causa de la latencia en la visión humana. Para crear la animación en las páginas web, los atributos de estilo de los elementos, la posición, el color y la opacidad, se modifican cada fotograma.

Con qué rapidez se pueden actualizar en última instancia estos atributos, determina la velocidad máxima de la trama de una interfaz de usuario. Para la interacción con las aplicaciones web, 60 fotogramas por segundo (fps) se considera que es la tasa óptima para una buena experiencia de usuario. Cuando no se puede lograr 60 fps consistentemente, se producen efectos indeseables en la

experiencia de usuario como pueden ser las sacudidas.

En su esencia, Famo.us es un cross-browser, ofrece el diseño de interfaz de usuario de alto rendimiento y una biblioteca optimizada con su propio motor de animación, escrita en JavaScript. Famo.us está optimizado para realizar su trabajo en el menor tiempo posible por cuadro. La interfaz de usuario de destino puede ser traducida por lo general un mínimo de 60 fps. El proyecto Famo.us se compromete a mantener o mejorar este rendimiento en todas las versiones posteriores.

En el siguiente diagrama, se muestra cómo funciona la librería de Famo.us.



Figura 1. Cómo funciona la librería de Famo.us

2.1.3 API de Famo.us

La API de Famo.us presenta su propio conjunto de objetos componibles de alto nivel como son las superficies, puntos de vista, y widget para el desarrollador. En su código Famo.us crea la escena gráfica de composición (actualmente llamada árbol Render), los cables del manejo de eventos, y dirige los horarios de las animaciones. Cada nodo en el escenario gráfico Famo.us es un nodo Render. Este nodo Render, evalúa la velocidad en fotogramas del escenario gráfico, en la actualidad 60 fps, y envía las actualizaciones necesarias a la DOM del navegador, a un elemento Canvas, o a WebGL. La aplicación beta de Famo.us dirige DOM del navegador solamente.

Famo.us está diseñado para dar salida a cualquiera de sus objetivos incluso en operaciones multimodales. Así que puede mostrar al mismo tiempo, varios de sus objetivos en la misma pantalla.

2.1.4 Render Tree

Una de las primeras cosas que se perciben en Famo.us, es la casi inexistente exposición al desarrollador de HTML y DOM. Ya que la interacción con DOM está plagada de problemas de rendimiento, Famo.us abstrae el funcionamiento de éste a través del árbol de directorios representado con JavaScript, llamado Render Tree.

Si se inspecciona una página web desarrollada con Famo.us, se puede ver que el DOM es muy plano en comparación con otras páginas web, donde éste, está altamente jerarquizado. En Famo.us, las relaciones son muy simples, los elementos se comportan unos de hermanos de otros. Con respecto al código HTML, Famo.us toma una posición totalmente diferente. Mantiene la estructura HTML en JavaScript y para nosotros, HTML es más una lista de cosas a dibujar en la pantalla y no la fuente principal del sitio web.

Para conseguir el posicionamiento relativo, la propagación de eventos y la estructura semántica, utilizamos los elementos anidados de HTML. Sin embargo, hay un coste para cada uno de ellos. El posicionamiento relativo hace que las páginas tengan flujos más lentos a la hora de mostrar contenido y animación. La propagación de eventos es cara si no se maneja con cuidado y la estructura semántica no está bien separada de la representación visual de HTML.

Para resolver estos problemas, aparece el Render Tree, ya que ofrece la solución para el posicionamiento relativo y para la estructura semántica. Abstrayendo el DOM, se consigue obtener un rendimiento de 60fps tal y como prometía Famo.us en un primer momento.

A continuación se verá como crear el árbol y de que componentes está compuesto.

2.1.4.1 Creando el Render Tree

Al punto de partida de un árbol se la llama raíz. Esta raíz en HTML, es la etiqueta <body>, mientras en Famo.us es un Contexto. Se puede instanciar a través del método `createContext()` lo cual, creará una etiqueta <div> con su atributo `class`. Quedaría de la siguiente manera:

```
var context = Engine.createContext();
```

3.1.4.2 Extendiendo el Render Tree

Un contexto no tiene representación visual. Para conseguir algo en la pantalla, se tendrá que ampliar el Render Tree añadiendo nodos a través del método `add`.

Una Superficie Famo.us (Surface), es un tipo de nodo, que corresponde vagamente con un <div> en HTML. Este <div> se desliza por el interior del anterior asignado al Contexto. De esta manera se construye HTML en Famo.us.

```
context
|
surface
var context = Engine.createContext();
context.add(surface);
```

2.1.4.3 Tipos de nodos

Un árbol se compone de nodos. En HTML, estos nodos son etiquetados como `<div>` o `<button>` mientras que en Famo.us, los nodos son de dos tipos: Renderables o Modificadores. A continuación se exponen estos tipos de nodos.

NODOS RENDERABLES

Son nodos que están inmersos en la pantalla. Se ha visto en el apartado anterior, como se asocia un Surface con un `<div>` pero hay más elementos que soporta Famo.us. Se puede ver en la tabla 4.

Tipo de superficie	Etiqueta asociada
Surface	<code><div></code>
ImageSurface	<code></code>
InputSurface	<code><input></code>
CanvasSurface	<code><canvas></code>
VideoSurface	<code><video></code>

Tabla 4. Elementos HTML soportados por Famo.us como nodos Renderables.

Existe un tipo más de Superficie llamado ContainerSurface y se asocia con un `<div>` que anida Surfaces. Se asocia principalmente en los recortes cuando la propiedad `{overflow: hidden}` se establece como propiedad CSS.

Famo.us es totalmente agnóstico a lo que se hace dentro de una superficie, ya sea hacer una plantilla, o utilizar un MVC (Modelo de Vista Controlador) para enlazar datos a su contenido. Sin embargo, si se quiere animar independientemente un trozo de código HTML, o vincular a los oyentes de DOM que interactúan con el resto de la aplicación, la mejor opción es encapsular dentro de una Surface. Las Surfaces son la unidad atómica renderizable en Famo.us, pero también apoya elementos compuestos renderizables más complejos. Estos se llaman Vistas y se verán en apartados posteriores.

MODIFICADORES

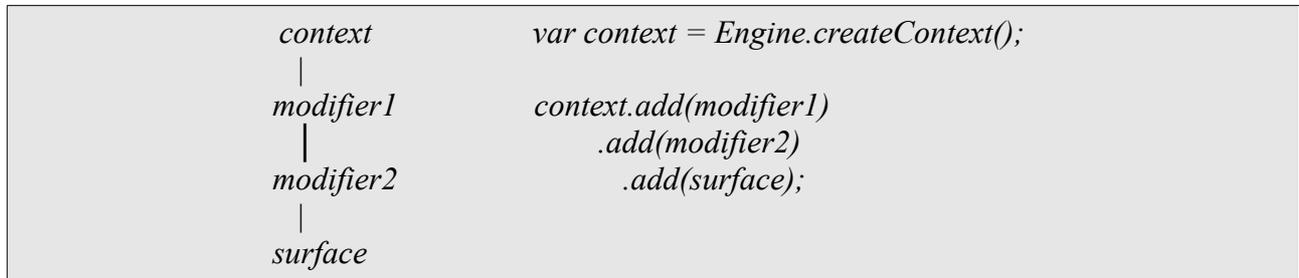
Un modificador es otro tipo de nodo en Famo.us. Es capaz de modificar los nodos por debajo de él en el Render Tree. Las superficies no saben dónde se encuentran en la página, o si son aún visibles; ese es el trabajo del modificador. Son responsables del diseño y la visibilidad del árbol Render por debajo de ellos. Se agrupan estos dos conceptos, diseño y visibilidad, porque las transformaciones CSS3 y la opacidad son, precisamente, las propiedades de hardware acelerado que pueden cambiar de forma permanente.

<i>context</i>	<code>var context = Engine.createContext();</code>
<i>modifier</i>	<code>var chain = context.add(modifier);</code>
<i>surface</i>	<code>chain = add(surface);</code>

Este ejemplo anterior, nos muestra una manera sencilla de asignar un modificador a una superficie.

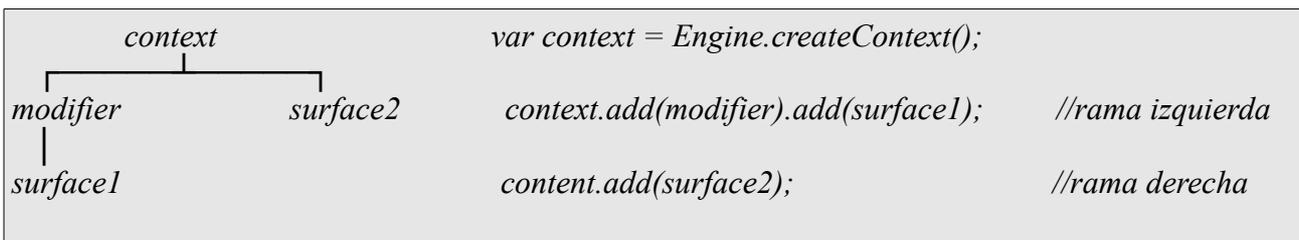
2.1.4.4 Encadenamiento de nodos

Los modificadores afectan el árbol Render por debajo de ellos. Eso significa que pueden afectar a otros modificadores. Por encadenamiento, sus transformadas están compuestas, y sus opacidades se multiplican. Esto hace que la separación de estado sea fácil de hacer. Un modificador puede manejar la opacidad, y otro, la rotación por ejemplo.

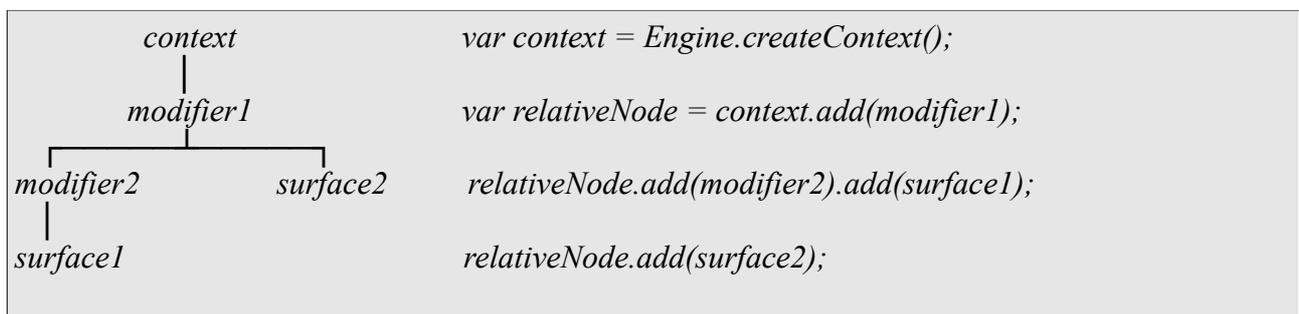


Hasta ahora los árboles han sido lineales: un nodo de forma secuencial después de otro. Lo más normal es que se necesite ramificar el árbol con más nodos y modificadores.

A continuación se muestra un ejemplo sencillo que muestra cómo crear una rama del árbol llamando al método `.add` sucesivamente en el mismo nodo.



La ramificación es clave para el posicionamiento relativo de objetos renderizables. Como en el siguiente ejemplo, donde `surface1` y `surface2` atienden al modificador1 y, `surface1` además tiene un modificador2, así las superficies no se solapan.



2.1.4.5 Vistas

Hasta ahora se ha visto que se pueden agregar modificadores y superficies a nuestro árbol Render. Estos pueden ser considerados como bloques de los widgets más complicados. Para ayudar a reducir la creación de un widget, Famo.us proporciona una clase base llamada View.

Una vista proporciona una interfaz para añadir al árbol Render (no es diferente de una superficie o modificador), recibir y mandar eventos, y disfrutar de los parámetros por defecto y las variables de estado. Famo.us también viene con una biblioteca de vistas comunes que siguen en construcción. Aquí, sólo se aborda la forma en que se pueden utilizar para ampliar el árbol Render, y cómo, encierran sus propios árboles internos Render.

En el siguiente ejemplo, se añade un ScrollView:

```
context      var context = Engine.createContext()
|
modifier     context.add(modifier).add(scrollview);
|
scrollview
```

Internamente, ScrollView tiene su propia lógica compleja, pero está oculta para el desarrollador, que simplemente puede incluirlo en su proyecto y agregarlo al árbol como cualquier otro nodo. Es el equivalente Famo.us de la DOM. Después de crear instancias de un ScrollView, podemos rellenarlo con cualquier otro renderizable a través de su interfaz SequenceFrom, construyendo así su árbol Render interno.

```
scrollview      scrollview.sequenceFrom([S1, S2, S3, ... , S10]);
├── S1
├── S2
├── S3
├── ...
└── S10
```

En el ejemplo anterior, S10 no tiene por qué ser una superficie. Puede ser una vista con sus propios modificadores, otros nodos, incluso otro scrollview. Usted podría tener un scrollview cuyo primer elemento tiene una opacidad transversal fundida entre dos superficies y que S10 sea una vista con la siguiente estructura:

```
S10
├── modifier1
│   └── surface1
└── modifier2
    └── surface2
S10.add(modifier1).add(surface1);
S10.add(modifier2).add(surface2);
```

Al encapsular lógica compleja en las vistas, la comprensión de una aplicación se hace más fácil. Y a diferencia de DOM, no hay degradación de rendimiento a partir de la estructura de anidación; todo está compactado hasta que llega a la DOM.

2.1.4.6 Resumen y conclusiones

En todos los ejemplos anteriores, se puede ver que existe un patrón: un árbol Render comienza con un contexto, se ramifica en un montón de modificadores, y termina con superficies. A diferencia del DOM, donde los nodos mezclan la representación visual con la agrupación sintáctica, el Render Tree hace una clara separación entre el diseño (modificadores), contenido (Superficies) y la estructura add(). De hecho, si usted quiere saber cuál es la posición, o la opacidad de una superficie en la parte inferior del Render Tree, sólo hay que multiplicar las opacidades y las transformaciones de los modificadores por encima de ella. Otro punto de divergencia es que el DOM ejecuta una reestructuración cada vez que se cambia el estilo de un nodo o el contenido de modo inmediato. Para resumir, en la tabla 5, se realiza una comparación entre el DOM tradicional, y el árbol Render de Famo.us:

	Famo.us Render Tree	DOM
Estructura de árbol	Sí	Sí
Nodos	Renderables y modificadores	Elementos HTML
Reflujos	No	Sí
Encapsulación	Vistas y Widgets	Shadow DOM
Significado	Estructura	Estructura, rendering
Ciclos Render	Modo retenedor	Modo inmediato
Lenguaje	JavaScript	HTML

Tabla 5. Comparación entre la estructura de Famo.us y DOM.

2.1.5 Comparativa con otras librerías 3D

La arquitectura de Famo.us es similar a WebGL 3D incluyendo Three.js y SceneJS. Se comparan y se comprueban las diferencias y similitudes para elegir la que mejor se adapte a nuestra aplicación web. Son las siguientes:

- Frame Rendering se desencadena por navegador rAF, requestAnimationFrame() en ambas arquitecturas. requestAnimationFrame() realiza una función de devolución de llamada como un argumento. El navegador llama a la función de devolución de llamada antes de la próxima actualización de la pantalla, por lo general dentro de la ventana de tiempo de 16,7 ms que corresponde a 60 fps. Las implementaciones de la biblioteca de animación deben llamar a rAF de nuevo durante el manejo de rAF para configurar la devolución de llamada para la siguiente trama. La compatibilidad entre navegador para rAF es desigual. Chrome y Firefox lo soportan por defecto mientras que Safari requiere un prefijo de proveedor especial en el código. Famo.us utiliza una solución de corrección polyfill internamente para acomodar las diferencias.
- En las bibliotecas 3D, se posicionan y colocan los triángulos y los objetos 2D como paredes o pisos y la geometrías 3D de alto nivel, tales como pirámides, esferas, cubos, o mallas de

polígonos, en el escenario gráfico. En Famo.us, la posición y el lugar de las superficies 2D o de nivel superior se colocan en objetos como son las vistas o widgets.

- En las bibliotecas 3D, se puede transformar un objeto mediante la adición y el encadenamiento de los nodos de transformación en el escenario gráfico, donde el objetivo es la hoja. En Famo.us, puede añadir nodos y modificadoras de la cadena en el escenario gráfico, donde una superficie u otro objeto renderizable es la hoja.
- En SceneJS, se puede especificar un escenario gráfico complejo usando JSON y modificarlo a través de los atributos ID en los nodos del grafo. Famo.us tiene un componente de escena que se puede utilizar para construir una escena gráfica compleja a través de JSON; lo modifica a través de los atributos ID al encadenar los nodos.
- En librerías 3D, los atributos de los objetos están animados en devoluciones de llamada rAF. En Famo.us, las propiedades de las superficies (en términos más generales, renderizables) se actualizan en cada tick del motor Famo.us, que es impulsado por devolución de llamada del navegador rAF.
- A pesar de que todos los objetos en una escena 3D se componen de triángulos, los desarrolladores rara vez necesitan trabajar con los triángulos individuales ya que las bibliotecas de renderizado 3D suelen incluir mallas de geometrías comunes como esferas y pirámides. A pesar de que la base renderizable en Famo.us es una superficie, rara vez se necesita trabajar con una individual ya que la biblioteca incluye muchos puntos de vista de interfaz de usuario de alto nivel prefabricados y widgets.
- El componente render de las librerías 3D, normalmente se dirige a Canvas, WebGL, y SVG, favoreciendo WebGL para su acceso casi directo al hardware subyacente GPU 3D-renderizable. El componente render de Famo.us se dirige a Canvas, DOM, y WebGL. Famo.us tiene un actualizador de DOM que utiliza optimizaciones de GPU ofrecidos por los principales navegadores.

2.1.6 Conclusiones

La creación de aplicaciones móviles de código nativo es difícil. Hay que subir empinadas curvas de aprendizaje, no sólo de múltiples sistemas operativos móviles, sino también de diferentes lenguajes de programación y cientos de API del sistema en cada plataforma. Mientras tanto, las tecnologías web para el desarrollo de aplicaciones móviles, a pesar de su promesa de plataforma cruzada, están muy lejos de la entrega de un código para la experiencia de usuario.

El marco Famo.us combina avances recientes en la optimización del navegador y conceptos maduros de las bibliotecas de renderizado 3D para ofrecer un alto rendimiento, la plataforma de interfaz de usuario-creación fácil de usar, altamente automatizable para aplicaciones web móviles. Ahora los desarrolladores de JavaScript pueden crear fácilmente aplicaciones móviles que ofrecen experiencias de usuario que las implementaciones de códigos nativos rivales aún no están en disposición de hacerlo.

2.2 Felder & Soloman

En este apartado, se ha hecho una búsqueda sobre cuál es la mejor herramienta para distinguir entre dimensiones de estilos de aprendizaje. En primer lugar, se aborda el estudio de distintas técnicas que ya están disponibles para diferenciar estos estilos y posteriormente, se muestra la información detallada sobre el test Felder & Soloman, qué estilos lo componen y cómo se calculan sus resultados.

2.2.1 Aproximación al uso de los estilos de aprendizaje como enfoque para elevar el rendimiento de alumnos universitarios de la modalidad de educación a distancia.

Este apartado trata sobre las herramientas que fueron estudiadas para llevar a cabo la diferenciación de los estilos de aprendizaje. Para ello, profundizamos en los estudios que ya están realizados sobre cada una de ellas, sus resultados y las conclusiones a las que se llegaron para elegir el test Felder & Soloman.

2.2.1.1 Introducción

La modalidad de estudios semipresenciales exige un balance entre el trabajo individual de los alumnos (la actividad no presencial) y el desempeño de éstos frente al profesor en la actividad presencial, que ha devenido en un tipo nuevo de clase: el encuentro. Éste, se convierte en un reto al profesor, ya que casi todos los docentes realizan su trabajo en modo presencial. Por ello es necesario elevar la eficacia del trabajo independiente del alumno para obtener el mayor beneficio posible. El trabajo muestra los resultados de la aplicación de dos test para determinar estilos de aprendizaje: una versión sobre la PNL (Programación Neurolingüística) y el otro, el de Felder and Soloman. Se compararon ambos resultados, los cuales se emplearon para estructurar las tareas para el trabajo independiente a desarrollar por los alumnos.

Teniendo en cuenta que el aprendizaje de los alumnos tiene características diferentes, se decidió desde el inicio de la carrera, período 2002-2007, realizar la investigación científico-pedagógica relacionada con el uso exploratorio de varias metodologías de trabajo en esta carrera. Esta investigación se divide en dos etapas diferentes:

- *Primera etapa:* Se introdujo el trabajo en equipo como base del estudio independiente de los alumnos, la formación educativa y el entrenamiento para exponer contenidos en público.
- *Segunda etapa:* Como complemento a lo anterior, se introdujeron herramientas para una mayor determinación de los estilos de aprendizaje de los alumnos en función de lograr mayor eficacia en su trabajo independiente.

Los resultados de la primera etapa no son presentados en este trabajo, pues se centran en los estilos de aprendizaje. Los resultados que los autores obtuvieron en la segunda etapa, es lo que se muestra a continuación.

2.2.1.2 Objetivos y tareas de investigación

El objetivo de esta investigación exploratoria realizada por el autor, consistió en identificar la relación entre el conocimiento del profesor y de los propios alumnos sobre los estilos de aprendizaje y el incremento de la eficacia del trabajo independiente de éstos, medida por sus resultados académicos en asignaturas dadas. En este estudio, las tareas de investigación fueron dos:

- Búsqueda de herramientas para la determinación de los estilos de aprendizaje de los alumnos.
- Aplicación de las herramientas encontradas y comparación de ellas para establecer la pertinencia de su empleo.

2.2.1.3 Herramientas para la determinación de los estilos de aprendizaje

Cathy Hsu presenta los resultados de la aplicación de una metodología que determina los estilos de aprendizaje introducidos por Kolb en 1984 en su Modelo de Aprendizaje a través de la Experiencia [1](Kolb, D.A. Experiential Learning).

Se presupone que los resultados del aprendizaje están vinculados a dos dimensiones fundamentales: la dimensión relativa a la “apropiación de la información” y la dimensión de la “transformación de la información”. Con este modelo surgieron cuatro estilos:

- **El divergente.** Combina los estilos de aprendizaje de la experiencia concreta y la observación reflexiva. Estas personas ven mejor las situaciones concretas desde diferentes puntos de vista, son imaginativos, entienden a las personas, perciben las relaciones entre diferentes situaciones, y son buenos en tormentas de ideas. Sin embargo, tienden a paralizarse ante las alternativas y les cuesta tomar decisiones.
- **El asimilador.** Combina los estadios de aprendizaje de la conceptualización abstracta y la observación reflexiva. Éstos son mejores para entender un amplio cúmulo de información dándole forma concisa y lógica. Son sistemáticos y buenos para la planificación, la creación de modelos y para el desarrollo de teorías. Pero también es cierto que encuentran dificultades para encontrar la aplicación práctica para sus ideas.
- **El convergente.** Éstos combinan los estadios de aprendizaje de la conceptualización abstracta y de la experiencia activa. Son las mejores personas para encontrar usos prácticos para las ideas y teorías. Tienden a ser eficaces como tomadores de decisiones aunque pueden estar solucionando el problema equivocado de una forma muy eficaz y pueden involucrarse en decisiones apresuradas.
- **El acomodador.** Éstos combinan los estilos de aprendizaje de la experiencia concreta a la experiencia activa. Tienen la habilidad de aprender fundamentalmente por la experiencia de “tocar con las manos”. Son buenos en lograr que las cosas se hagan, asumiendo riesgos para liderar. El problema es que pueden involucrarse en mejoras triviales y actividades con poco sentido.

La segunda herramienta presentada, es la basada en la Programación Neurolingüística (ANEXO I), Conoce tu estilo de aprendizaje y estudia mejor. Ésta considera que los estilos se clasifican en Visual, Auditivo y Kinestésico. Las características de estos estilos, son las siguientes:

APRENDIZAJE VISUAL.

- Se aprende preferentemente a través del contacto visual con el material educativo.
- Se piensa en imágenes. De esta es más fácil absorber grandes cantidades de información con rapidez.
- Hay una mayor capacidad de abstracción ya que al visualizar, se crean relaciones entre distintas ideas y conceptos.
- Las representaciones visuales del material como gráficos, cuadros y diagramas, mejoran este tipo de aprendizaje.
- Los vídeos, películas o programas de computación también mejoran este aprendizaje.
- Se recuerda mejor lo leído que lo escuchado.
- Se calcula que entre un 40% y 50% de la población general, pertenece a este estilo de aprendizaje.

APRENDIZAJE AUDITIVO.

- Se aprende fundamentalmente escuchando.
- Piensa y recuerda de manera secuencial y ordenada, por lo que prefiere los contenidos orales y los asimila mejor cuando pueden explicárselo a otras personas.
- Responden al estilo de enseñanza más habitual del sistema escolar, ya que se adaptan al formato de clase expositiva.
- Tienen una gran capacidad para aprender idiomas y/o música.
- Los discos, las discusiones en público y las lecturas en voz alta, mejoran su aprendizaje.
- Recuerdan mejor lo que escuchan a lo que leen.
- Entre un 10% y 20% de la población pertenece a este estilo de aprendizaje.

APRENDIZAJE KINESTÉSICO

- Se aprende al interactuar físicamente con el material educativo.
- Para aprender, necesitan asociar los contenidos con movimientos o sensaciones corporales.
- Sus aprendizajes son más lentos, y se desempeñan mejor en tareas de tiempo limitado y con descansos frecuentes.
- Las actividades físicas, el dibujo, la pintura, los experimentos de laboratorio, los juegos de rol, mejoran su aprendizaje.
- Pueden recordar mejor lo que hacen en lugar de lo que ven o escuchan.
- oseen la manera menos eficiente para almacenar información académica y la mejor para almacenar información que tenga que ver con lo deportivo o artístico.
- Se calcula que entre un 30% y un 50% pertenecen al estilo kinestésico. Este porcentaje se incrementa en la población masculina.

La tercera herramienta es la que estudió Felder y se basa en cinco dimensiones:

Pregunta	Dimensión del Aprendizaje y Estilos	Descripción de los estilos
¿Qué tipo de información perciben preferentemente los estudiantes?	Dimensión relativa al tipo de información: sensitivos – intuitivos	Básicamente, los estudiantes perciben dos tipos de información: información externa o sensitiva a la vista, al oído o a las sensaciones físicas e información interna o intuitiva a través de memorias, ideas, lecturas, etc.
¿A través de qué modalidad sensorial es más efectivamente percibida la información cognitiva?	Dimensión relativa al tipo de estímulos preferenciales: visuales – verbales	Con respecto a la información externa, los estudiantes básicamente la reciben en formatos visuales mediante cuadros, diagramas, gráficos, demostraciones, etc. o en formatos verbales mediante sonidos, expresión oral y escrita, fórmulas, símbolos, etc.
¿Con qué tipo de organización de la información está más cómodo el estudiante a la hora de trabajar?	Dimensión relativa a la forma de organizar la información: inductivos - deductivos	Los estudiantes se sienten a gusto y entienden mejor la información si está organizada inductivamente donde los hechos y las observaciones se dan y los principios se infieren o deductivamente donde los principios se revelan y las consecuencias y aplicaciones se deducen.
¿Cómo progresa el estudiante en su aprendizaje?	Dimensión relativa a la forma de procesar y comprensión de la información: secuenciales – globales	El progreso de los estudiantes sobre el aprendizaje implica un procedimiento secuencial que necesita progresión lógica de pasos incrementales pequeños o entendimiento global que requiere de una visión integral.
¿Con qué tipo de organización de la información está más cómodo el estudiante a la hora de trabajar?	Dimensión relativa a la forma de trabajar con la información: activos – reflexivos.	La información se puede procesar mediante tareas activas a través compromisos en actividades físicas o discusiones o a través de la reflexión o introspección.

Para la evaluación de los resultados de esta herramienta existe el Índice de Aprendizaje de Felder & Soloman, cuyo cuestionario se puede consultar en el ANEXO II. Cómo calcular este índice de aprendizaje, será abordado en apartados posteriores.

Este cuestionario de Felder y Soloman no contiene la forma de evaluar las respuestas, y de ahí determinar el estilo de aprendizaje del alumno. Por ello se aplica la técnica de la ingeniería inversa, es decir, sin conocer cuál es la metodología real, por un resultado se llega a conformar la metodología con un grado aceptable de incertidumbre. Este proceso consistió en responder el cuestionario por el autor de este trabajo y por la correspondencia de estilos del autor se llegó a la forma de agrupar las respuestas a las preguntas. Por este resultado se conformó el método a seguir, al no contar con él, para evaluar los resultados individuales de los alumnos y determinar sus estilos de aprendizaje en la presente investigación.

2.2.3.4 Discusión de los resultados

En este apartado se muestran los resultados de las dos últimas herramientas presentadas, el cuestionario basado en PNL y el de Felder and Soloman para poder elegir cuál es la más adecuada y discernir entre los estilos de aprendizaje.

2.2.3.4.1 Resultados del uso del cuestionario basado en la PNL (Programación Neurolingüística)

El cuestionario se aplicó a 35 alumnos de la carrera del curso 2006-2007 y los resultados obtenidos fueron:

a) Según la distribución de los puntos para cada estilo

Estilos	Cantidad de alumnos
Con estilos balanceados (los tres estilos se usan al mismo nivel)	27
Con predominio del Visual	6
Con predominio del Auditivo	1
Con predominio del Kinestésico	1

b) Por el estilo con la mayor puntuación obtenida para el alumno

Estilo con mayor cantidad de puntos	Cantidad de alumnos
Estilo Visual	20
Estilo Auditivo	6
Estilo Kinestésico	4
Auditivo-Kinestésico (puntuación igual)	4
Visual-Kinestésico (puntuación igual)	1

El hecho de que no exista una distribución marcada de los alumnos según sus estilos de aprendizaje, no permite una clara diferenciación de cuál es el estilo que caracteriza a cada alumno. Por este razonamiento, vemos que esta herramienta no es la adecuada para este estudio. El autor es partícipe de la opinión de que esta forma de determinar el estilo de aprendizaje no es lo suficientemente confiable y por ello, decidió buscar otras herramientas para realizar una nueva medición. Los resultados que sí aportó este estudio, son los siguientes:

- Los alumnos, por primera vez, conocen que existen los estilos de aprendizaje y cuál es el propio de cada uno.
- La vía más apropiada para que los alumnos elevaran la eficacia de su trabajo independiente es el uso de gráficos, esquemas, cuadros sinópticos, etc.

2.2.3.4.2 Resultados del cuestionario Felder & Soloman.

Soloman y Felder parten de que hay cuatro pares de estilos:

- Activo-reflexivo.
- Perceptivo-intuitivo.
- Visual-verbal.
- Secuencial-global.

Estos estilos serán analizados en apartados posteriores.

Los resultados se entregan por alumno en valores de cuatro escalas, una para cada par de estilos como la que se observa a continuación:

ACTIVO						REFLEXIVO					
								X			
11	9	7	5	3	1	1	3	5	7	9	11

La cruz encima del número cinco indica que ese es el valor resultante del par de estilos Activo-Reflexivo.

Felder plantea que si su valor en la escala está entre 1 y 3, es una muestra de que existe balance entre las dos dimensiones. Si el valor se encuentra entre 5 y 7, indica una preferencia moderada por una de las dos dimensiones de la escala y ese alumno aprenderá con mayor facilidad en un entorno de aprendizaje que favorezca esa dimensión. Y por último, si el valor está entre 9 y 11, muestra una fuerte preferencia por una de las dos dimensiones de la escala, por lo que puede confrontar serias dificultades en un ambiente de aprendizaje que se caracterice por métodos que favorezcan esa preferencia.

El autor aplicó esta herramienta a 18 de los alumnos que ya habían sido evaluados con la primera, y se obtuvieron los siguientes resultados:

Estilos	Cantidad de alumnos
Con estilos balanceados (todos los pares entre -3 y 3)	4
Con predominio de ACTIVO	4
Con predominio de REFLEXIVO	3
Con predominio de SENSORIAL	4
Con predominio de INTUITIVO	0
Con predominio de VISUAL	8
Con predominio de VERBAL	2
Con predominio de SECUENCIAL	5
Con predominio de GLOBALIZADOR	3

Otra forma interesante de ver los resultados es la siguiente:

Estilos	Cantidad de alumnos
Con un solo estilo dominante - VISUAL	2
Con un solo estilo dominante - VERBAL	1

El resto de los alumnos, tienen más de un estilo dominante en diferentes pares.

Como se puede observar, con esta herramienta se logra una mayor diferenciación de los alumnos y se llega a la conclusión de que cada uno de los alumnos evaluados contó con una valoración que le permitió organizar sus propias formas de trabajo independiente. Por parte del profesor, se siguió la estrategia de generar formas metodológicas de las actividades en los encuentros para el estilo visual, ya que en ambas herramientas éste era el estilo mayoritario de los alumnos.

Al analizar los resultados, observamos que se refleja de manera adecuada la realidad de los alumnos, pues la mayor parte de los recursos docentes que se emplean en las aulas están basados en el empleo de esquemas, gráficos y láminas de diferente tipo.

2.2.3.4.3 Conclusiones y resultados

Como resultado de la experiencia vemos que se ha cumplido el objetivo a través de esta investigación exploratoria ya que el conocimiento por parte de los alumnos sobre los estilos de aprendizaje, ha elevado la eficacia del trabajo independiente de éste y también, se ha conseguido adaptar por parte del profesor las tareas para el trabajo independiente y para los encuentros. Con respecto a estos resultados, las recomendaciones son las siguientes:

- Se debe continuar empleando la determinación de los estilos de aprendizaje de los alumnos de primer año de la carrera.
- Continuar con la investigación para lograr una validez estadística que sustente mejor la herramienta a emplear con el objetivo de aumentar la calidad de los recién ingresados en la carrera.
- Exponer los resultados en eventos científico-pedagógicos para su divulgación y como motivación a otros docentes para utilizar este enfoque.

2.2.2 Análisis de las dimensiones de estilo de aprendizaje según Felder-Silverman.

En este apartado se estudiará el modelo de Felder-Silverman como paso previo al modelo Felder and Soloman.

2.2.2.1 Introducción

Se está incrementando la incorporación de estilos de aprendizaje dentro de los estudios tecnológicos. Muchos estudios se están realizando en torno a este área, ya que estos estilos de aprendizaje pueden ser incorporados en muchos aspectos del sector educativo tecnológico y proveer cursos que se ajusten a los estilos individuales de cada alumno. Todos estos estudios están motivados por la realidad que presentan los estudiantes, ya que cada uno de ellos prefieren una manera diferente de investigar y aprender contenidos, de acuerdo a la facilidad con la que asimilan los nuevos conceptos. De esta manera, los estudiantes asimilan las materias de manera más rápida y fácil y alcanzan antes sus objetivos.

El objetivo de este apartado es analizar los datos sobre estilos de aprendizaje con respecto al modelo de estilos de aprendizaje de Felder-Silverman (FSLSM), para describir con mayor detalle las dimensiones de estos estilos. Esto lleva a identificar las características de cada una de las cuatro dimensiones del FSLSM para ser capaz de hacer una distinción gradual sobre ellas. También se analizará como representar cada característica dentro de su dimensión de estilo de aprendizaje. Si tenemos en cuenta las características y su relevancia en estos estilos, se logrará una mejor estimación de los resultados del planteamiento y por tanto, una aplicación de la información obtenida con más sentido.

2.2.2.2 Modelo de estilos de aprendizaje de Felder-Silverman

Hay diferentes estudios sobre estilos de aprendizaje como puede ser el de Kolb en 1984, Honey and Mumford en 1982 y el de Felder and Silverman en 1988 [2] (Learning Styles Questionnaire). El estudio se centra en el último caso, Felder and Silverman, ya que describe estos estilos con mayor detalle, distinguiendo entre las preferencias de cuatro dimensiones. También se centra en este modelo porque está basado en el estudio de tecnologías avanzadas, es el más recomendado para aplicaciones e-learning.

Hay cuatro dimensiones en FSLSM. Cada alumno tiene una preferencia específica en cada una de estas cuatro dimensiones. La primera dimensión distingue entre una manera activa o reflexiva de procesar la información. Los estudiantes activos, trabajan mejor aplicando y probando a partir del material de trabajo, suelen estar interesados en trabajar en grupo para poder discutir sobre los resultados obtenidos. En contraste, los estudiantes reflexivos, prefieren pensar y reflexionar sobre el materia y también trabajar solos o en grupos pequeños.

En la segunda dimensión, distinguimos entre alumnos perceptivos e intuitivos. A los alumnos perceptivos les gusta aprender hechos y concretar con el material de aprendizaje. Suelen resolver los problemas con un enfoque estándar y tienden a ser pacientes con los detalles. Son personas a las que les considera realistas, sensatas y prácticas, con lo cuál les gusta aplicar sus resultados al mundo real. Por otro lado, están los alumnos intuitivos que prefieren aprender con materiales abstractos y sus significados subyacentes. Son más capaces de descubrir posibilidades y relaciones entre varias materias y tienden a innovar y ser más creativos que los alumnos perceptivos.

En el tercer caso se encuentra la dimensión visual y escrita. La diferencia aquí radica en alumnos que prefieren recibir la información de manera visual, con diagramas, dibujos, diagramas de flujo, etc, o a través de una representación textual, sin tener en cuenta si es hablada o escrita.

El cuarto y último caso, los alumnos están caracterizados por la forma de entender una materia. Se dividen en secuenciales y globales. Un alumno secuencial, aprende en pequeños pasos y tiene una línea de progreso. Intentan seguir la manera lógica de llegar a los resultados. En contraste, tenemos los alumnos globales, a los cuales les gusta tener conocimiento sobre la materia completa y avanzan en grandes pasos. En un primer momento, no hay conexiones entre las diferentes partes del objeto de estudio pero después de haber entendido las distintas partes, de repente, encuentran la conexión y consiguen la visión de la materia completa. Suelen ser capaces de resolver problemas complejos, encontrar conexiones entre áreas diferentes pero encuentran dificultades para explicar como han llegado al resultado.

2.2.2.3 Colección y análisis de datos

Para investigar los estilos de aprendizaje de los estudiantes, se realizó un estudio con 207 participantes. Estos alumnos estudiaban Ingeniería Web e Informática de Gestión. Para detectar los estilos de cada uno de ellos, completaron un cuestionario desarrollado por Felder and Soloman en 1997. Este cuestionario se presenta en el Anexo II y en la siguiente sección se presentan los resultados del estudio de dicho cuestionario.

2.2.2.4 Índice de estilos de aprendizaje (ILS)

El Índice de Estilos de Aprendizaje (ILS), fue desarrollado por Felder y Soloman.

Es un cuestionario de 44 preguntas para identificar estos estilos según FLSM. Como se mencionó anteriormente, cada alumno tiene una preferencia personal por cada dimensión. Estas preferencias se expresan con valores entre +11 y -11 por dimensión con pasos de +/- 2. Estos rangos vienen dados en las 11 cuestiones de cada una de las cuatro dimensiones en las que se divide este modelo.

Cada pregunta tiene un valor de +1 si la respuesta coincide con la primera opción (respuesta a) y un valor de -1, si coincide con la segunda opción (respuesta b). La respuesta a corresponde a las preferencias de alumnos activos, perceptivos, visuales y secuenciales y las respuestas b, a los alumnos reflexivos, intuitivos, verbales o globales. Los resultados de este modelo se dividen en fuerte/moderado, si están entre 5 y 11 y equilibrado si están entre +/- 3.

Basándose en el modelo FLSM, se puede ver que cada estilo de aprendizaje se describe a partir de ciertas características, agrupándolas por similitud en la semántica. En la tabla 1 vemos los grupos semánticos identificados en cada estilo como las cuestiones que pertenecen a cada uno de ellos. Una cuestión puede aparecer varias veces en la tabla, si la respuesta apunta a diferentes grupos semánticos.

ESTILO	GRUPO SEMÁNTICO	CUESTIONES ILS(Respuesta a)	ESTILO	GRUPO SEMÁNTICO	CUESTIONES ILS(Respuesta b)
ACTIVO	Intentar, probar	1, 17, 25, 29	REFLEXIVO	Trabajo sobre el material	1, 5, 17, 25, 29
	Sociable	5, 9, 13, 21, 33, 37, 41		Introvertido	9, 13, 21, 33, 41, 37
PERCEPTIVO	Procesos estándar	2, 30, 34	INTUITIVO	Nuevos desarrollos	2, 14, 22, 26, 30, 34
	Material concreto	6, 10, 14, 18, 26, 38		Material abstracto	6, 10, 18, 38
	Interesado en los detalles	22, 42		No interesado en detalles	42
VISUAL	Imágenes	3, 7, 11, 15, 19, 23, 27, 31, 35, 39, 43	VERBAL	Palabras habladas	3, 7, 15, 19, 27, 35
				Palabras escritas	3, 7, 11, 23, 31, 39
				Dificultad con el estilo visual	43
SECUENCIAL	Orientado al detalle	4, 28, 40	GLOBAL	Visión completa	4, 8, 12, 16, 28, 40
	Progreso secuencial	20, 24, 32, 36, 44		Proceso no secuencial	24, 32
	De menos a más	8, 12, 16		Relaciones y conexiones	20, 36, 44

Tabla 1: Grupos semánticos asociados al cuestionario ILS

Para encontrar los grupos semánticos más representativos en cada dimensión, se llevaron a cabo un método para reducir la separación óptima lineal de dimensión, llamado Análisis de Discriminación Lineal Fisher. Este modelo se comparó con resultados empíricos de acuerdo a las frecuencias de análisis para validarlas.

2.2.2.5 Resultados del estudio

Los resultados del estudio indican cuales son las características que más impacto tienen dentro de cada dimensión de estilo de aprendizaje. Estas características son puntuadas con altos valores como se puede ver en la Tabla 2. Se resaltan los valores más significativos en negrita.

Estilos	Grupos semánticos	Act/Ref	Perc/Int	Vis/Ver	Sec/Global
Activos	Intentar, probar	0,639	0,113	0,536	0,211
	Sociable	0,452	0,146	0,190	0,180
Reflexivos	Trabajar sobre material	0,597	0,122	0,486	0,217
	Introverso	0,698	0,143	0,175	0,170
Perceptivos	Procesos estándar	0,237	0,568	0,301	0,174
	Materiales concretos	0,178	0,777	0,380	0,245
	Interesado en los detalles	0,147	0,409	0,329	0,456
Intuitivos	Procesos nuevos	0,193	0,678	0,309	0,237
	Material abstracto	0,225	0,715	0,453	0,173°
	No interesado en detalles	0,008	0,699	0,026	0,151
Visuales	Imágenes	0,238	0,227	0,944	0,167
Verbales	Palabras habladas	0,202	0,227	0,944	0,167
	Palabras escritas	0,171	0,199	1,086	0,258
	Dificultad con el estilo visual	0,297	0,388	0,789	0,078
Secuenciales	Orientado al detalle	0,224	0,218	0,290	0,800
	Progreso secuencial	0,100	0,237	0,432	0,686
	De menos a más	0,123	0,154	0,113	0,839
Globales	Visión completa	0,174	0,186	0,202	0,819
	Proceso no secuencial	0,140	0,175	0,520	0,715
	Relaciones y conexiones	0,074	0,278	0,375	0,869

Tabla 2: Relevancia de los grupos semánticos en las dimensiones de estilos de aprendizaje.

Interpretando la tabla anterior se pueden ver cuáles son las características que más importan dentro de cada dimensión. Se recorren para analizar los resultados.

Dentro del estilo **activo** se puede ver que la preferencia es probar e intentar con el material dado, no es solo reflexionar y pensar sobre él, sino hacer prácticos los casos teóricos. En el lado de los alumnos **reflexivos**, se ve que lo más importante para ellos es trabajar de forma individual por encima de reflexionar sobre el material dado.

La característica más importante para los alumnos **perceptivos** es tener material concreto mientras que en los alumnos **intuitivos** el material abstracto es su opción a destacar aunque también destaca el no ser muy cuidadosos con los detalles en un primer momento.

Para el estilo **visual** solo existe un grupo semántico, la facilidad para recordar diagramas de flujo, imágenes, etc. Este grupo es muy representativo para este estilo de aprendizaje. Con respecto al estilo **verbal**, el grupo que destaca es el lenguaje escrito aunque los otros dos grupos, lenguaje oral y dificultad con el estilo visual, son bastante representativos también.

En relación a la última dimensión de estilos de aprendizaje están a los alumnos **secuenciales**, los cuales le dan más importancia a tener una visión de la solución completa que al resto de grupos semánticos, aunque todos son representativos. Con respecto a los alumnos **globales**, se observa que priorizan en tener claras las conexiones y relaciones del área a estudiar.

Otra parte importante a analizar en los resultados, es qué cuestiones son las más representativas para cada dimensión. Tendremos en cuenta la frecuencia con la que los alumnos con una misma preferencia contestan la misma respuesta. Ésta se mide en porcentajes y la diferencia entre ellos es la que marca lo representativa que es una respuesta en cada dimensión. Estas respuestas han sido ordenadas en un ranking, y las cinco que mejor representan a cada estilo aparecen en la Tabla 3.

Según este estudio, se llega a la conclusión de que para la dimensión Activos/Reflexivos la primera, tercera y quinta pregunta del ranking están relacionadas a alumnos considerados sociables, que conocen al resto de estudiantes de clase y que les gusta trabajar en grupos. Otro aspecto que se refleja en esta dimensión, es la importancia que se le da a probar e intentar cosas con el material dado o por el contrario, reflexionar sobre él.

En la dimensión referente a los alumnos Perceptivos/Intuitivos, las primeras cuatro preguntas tratan sobre la necesidad de estudiar sobre hechos concretos y datos o sobre material abstracto como teorías o conceptos. La quinta pregunta está enfocada al concepto que tiene cada uno sobre si es realista o innovador que pertenece al grupo semántico que trata sobre hacer las cosas con procesos estándar o buscando nuevas maneras.

En relación a la tercera dimensión, Visual/Verbal, las preguntas hacen referencia a la preferencia de un lenguaje oral, escrito o si hay dificultades para entender un medio visual.

Por último, en la dimensión Secuencial/Global, la primera, segunda y cuarta cuestión tratan sobre si el alumno prefiere seguir una asignatura de manera secuencial, paso a paso o si por el contrario prefiere hacerse una idea sobre la estructura completa y después buscar relaciones entre las distintas partes que la contienen.

Estilo	Posición	Número de pregunta	Pregunta
Activo/Reflexivo	1	37	Soy considerado como (a) Sociable, (b) Reservado
	2	1	Entiendo algo mejor después de (a) Probarlo (b) Pensarlo
	3	13	En clases (a) Suelo conocer al resto de estudiantes (b) No suelo conocerlos
	4	25	Lo primero que suelo hacer es (a) Probar (b) Pensar en qué voy a hacer
	5	21	Prefiero estudiar (a) En grupo (b) Sólo
Perceptivo/Intuitivo	1	6	Si yo fuera el profesor, el curso sería impartido (a) Tratando hechos reales (b) Tratando ideas y teorías
	2	38	Prefiero los cursos que enfatizan (a) Material concreto (fechas, datos) (b) Material abstracto (Teorías, conceptos)
	3	18	Prefiero (a) Certeza (b) Teoría
	4	10	Encuentro fácil (a) Aprender hechos (b) Aprender conceptos
	5	2	Me considero (a) Realista (b) Innovador
Visual/Verbal	1	31	Cuando alguien me muestra datos, prefiero (a) Gráficos o diagramas (b) Textos que resuman los resultados
	2	11	En un libro con diagramas e imágenes, yo suelo (a) Mirarlos con detenimiento (b) Centrarme en el texto
	3	7	Prefiero obtener nueva información de (a) Imágenes, gráficos... (b) Información verbal
	4	19	Recuerdo mejor (a) Lo que veo (b) Lo que oigo
	5	3	Cuando pienso en qué hice ayer, suelo hacerlo (a) Con imágenes (b) Con palabras
Secuencial/Global	1	36	Cuando estoy aprendiendo algo nuevo, prefiero (a) fijar el objetivo y aprender todo lo que pueda (b) Intentar hacer conexiones entre distintas áreas relacionadas
	2	20	Lo más importante para mí en un profesor que (a) Exponga la materia en pasos secuenciales (b) Trate de hacer conexiones entre el material y otras áreas relacionadas
	3	8	Una vez que entiendo (a) Todas las partes, lo entiendo todo (b) La materia completa, paso a ver cada parte
	4	44	Cuando resuelvo problemas en grupo, suelo (a) Pensar en los pasos necesarios para resolver el problema (b) Pienso en las posibles consecuencias y aplicaciones de la solución en distintas áreas.
	5	4	Tiendo a (a) Entender los detalles de una materia pero suelo ser perezoso para ver la estructura completa (b) Entender la materia completa pero soy perezoso con los detalles

Tabla 3: Cinco respuestas más representativas para cada dimensión ILS de acuerdo al análisis de frecuencias..

El método LDA utilizado para calcular los coeficientes y hacer una estimación de los resultados parece el más adecuado, ya que a parte de ser capaz de incluir todas las características más representativas, da una indicación más aproximada de la importancia de cada una de estas características.

2.2.2.6 Conclusiones e investigaciones futuras

Este estudio se realiza para analizar de una manera profunda los datos del cuestionario ILS con el objetivo de conseguir una mayor descripción sobre los estilos de aprendizaje del modelo de Felder-Silverman. De esta manera, se divide cada dimensión de estilo de aprendizaje en grupos semánticos y se analiza el impacto de cada uno de estos grupos para cada estilo. Será utilizado un híbrido para detectar las características más interesantes desde distintas búsquedas y puntos de vista de aplicación.

En los últimos años, los estudios tecnológicos prestan atención a los distintos estilos de aprendizaje para mejorar la adaptabilidad en el sistema educativo. No solo incorporan dimensiones de estos estilos, sino que añaden las características de estas dimensiones para conseguir una representación más aproximada sobre los estilos de aprendizaje de cada alumno y también una mejor adaptación al medio en el que se desarrolla. También se busca poder mejorar los métodos pedagógicos, soportando un aprendizaje más efectivo y personalizado.

En próximas investigaciones se incluirán análisis estadísticos más exhaustivos para confirmar los resultados. También se recopilará más información sobre los distintos grupos semánticos para llegar a una adaptación al sistema educativo más realista. Por último, se intentará detectar de manera automática los estilos de aprendizaje teniendo en cuenta el comportamiento de los alumnos en los cursos on-line.

Capítulo 3. Aplicación web. Test de Felder & Soloman

En este capítulo se desarrollará el análisis, diseño e implementación de la aplicación. En él se desglosará toda la etapa de desarrollo, desde un primer análisis hasta el diseño de la aplicación, pasando por la implementación, para tener una visión más concreta del proyecto.

3.1 Análisis y diseño

Esta se podría catalogar como la fase de concepción del proyecto. En ella se van a documentar las condiciones y capacidades que deberá cumplir la aplicación, así como se estudiará la viabilidad de la misma y los posibles usuarios en potencia.

3.1.1 Necesidad de la aplicación

Ante la diversidad de alumnos en distintas carreras de ingeniería, y ante la diversidad de estilos de aprendizaje de cada uno de ellos, se plantea por parte de los profesores, la necesidad de conocer de qué manera pueden ofrecerles material y recursos, para ayudarles a optimizar su tiempo de estudio y mejorar las calificaciones de las asignaturas dadas.

El test de Felder and Soloman, se desarrolló para alumnos de ingeniería que toman clases de manera semipresencial, aunque en nuestro caso, se aplicará a los estudiantes de primer año de carrera de las distintas ingenierías, que cursan sus estudios de manera presencial.

Esta aplicación, se incluirá en el aula virtual de cada titulación, para que los alumnos puedan tenerla disponible cada vez que quieran consultar sus estilos de aprendizaje y recordar los consejos que se dan para mejorar sus resultados.

3.1.2 Requisitos de usuario

En este apartado se van a especificar los requisitos de usuario, que comprenderán las funcionalidades del sistema que han de ser resueltas. Dichos requisitos harán especial énfasis en la perspectiva del producto, sin entrar en nivel de detalle demasiado técnico. Se estudiará cuáles son los elementos que la aplicación debe ofrecer al usuario.

En la aplicación, los requisitos de usuario, se resumen en dos básicamente, ofrecer un diseño simple y una agradable experiencia de usuario (UX). Una vez que se accede a la aplicación, se deben encontrar los enlaces hacia todas las opciones de manera clara. De esta manera, se puede encontrar información sobre los estilos de aprendizaje antes de hacer el test, pasar a continuación a la realización de éste, mostrar los resultados y acceder de manera directa a los consejos ofrecidos.

En la realización del test y también para la muestra de información de estilos y consejos, se ha optado por superficies planas para conseguir que los usuarios que están utilizando la aplicación, puedan centrarse en la información ofrecida en vez de en los efectos que el framework Famo.us tiene disponibles a la hora del diseño de la interfaz.

Para mostrar los resultados, se ha optado por el estudio de gráficos 3D, intentando mantener la simplicidad en este aspecto, haciendo que un cubo 3D gire mostrando en cada una de sus caras laterales, el resultado de los estilos de aprendizaje. Una vez que se han visto y anotado, se ha optado

por animar el cubo que muestra los resultados, haciendo que rote en distintas direcciones y velocidades, consiguiendo así que sea más atractivo y divertido el diseño de la aplicación. De esta manera se consigue combinar la simplicidad y una experiencia agradable al usuario que está utilizando la aplicación web.

3.1.3 Requisitos de software

En este apartado se van a especificar los requisitos de software a cumplir por el sistema. Dichos requisitos proporcionarán información sobre la funcionalidad de nuestra aplicación. Estos requisitos nos muestran la funcionalidad de la aplicación web. Los requisitos de software de nuestra aplicación, son los siguientes:

- Aplicación de distintas vistas según las necesidades. Dependiendo del contenido, para poder adaptar la información de manera que quede con un diseño simple, se ha optado por utilizar Surfaces, GridLayout o ScrollViews.
- En la realización del test, se han identificado las respuestas disponibles de cada pregunta con un “name” específico. Así se puede asignar a cada pregunta, el estilo al que va asociada.
- Para mostrar los resultados, se decidió sacar del formulario el botón Resultados, ya que era necesario manejarlo con el evento Surface.on('click'). De esta manera, queda un Surface superpuesto a un ScrollView (el que muestra el test completo), que al ser pulsado, se dirige a una función específica que calcula los estilos de aprendizaje.
- A la hora de realizar el cubo 3D, se utiliza el elemento Quaternion que ofrece Famo.us en su librería Math. Este elemento facilita la movilidad del cubo, pudiendo jugar con los ejes en torno a los cuales gira y la velocidad de rotación.
- En Famo.us, se juega con la superposición de vistas y surfaces, así que para volver a la pantalla principal, se usa de la función reload(). Esta funcionalidad la es encontrada al pulsar la surface Atrás.

3.1.4 Casos de uso

Los casos de uso en el análisis de un proyecto software consisten en la descripción de la interacción entre actores y sistema en UML. En este caso, el usuario será dicho actor y su cometido será la realización del test Felder & Soloman a través de la aplicación web.

A continuación se representan los diferentes casos de uso que podrán darse en la aplicación de manera gráfica, por lo que facilitará la comprensión de las funcionalidades del sistema, y permitirá la captura y definición de los requisitos a cumplir posteriormente.

- **Caso inicial:** Pantalla principal.
El alumno entra en la aplicación y encuentra una pantalla principal en la cual tiene acceso a la realización del test, puede consultar los estilos de aprendizaje que existen y sus características y también puede consultar los consejos que se dan para cada estilo.
La opción de volver atrás existe, pero en este caso no es significativa, porque se vuelve a la pantalla que está presente en ese momento, la pantalla principal.

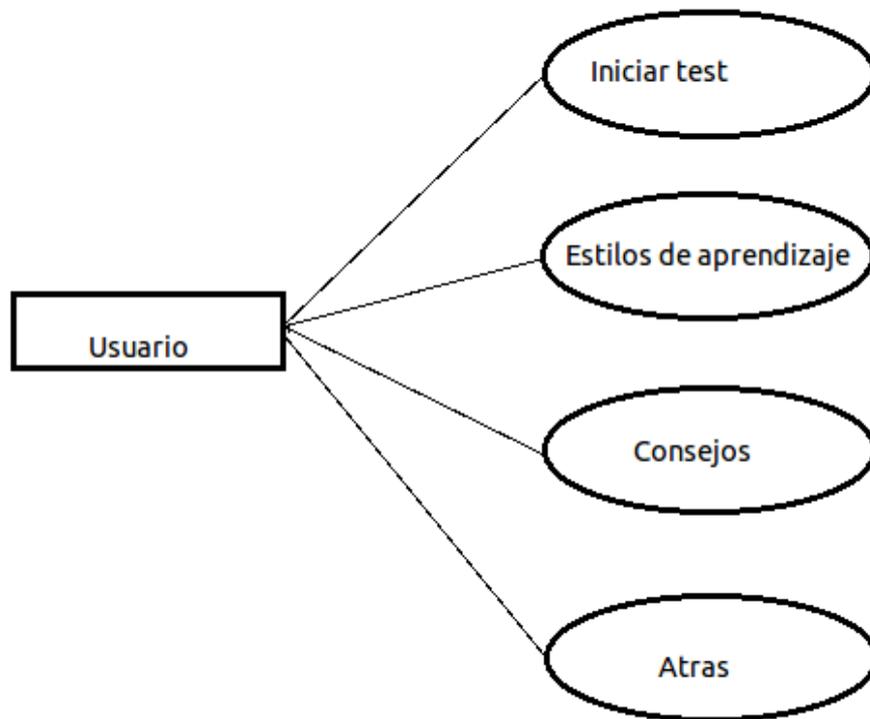


Figura 4.1 Caso inicial. Pantalla principal.

- **Segundo caso:** Realización del test.

Una vez que se accede al test, al realizarlo se pulsa el botón Resultados para que muestre el gráfico 3D. En la realización del test, se siguen teniendo disponibles en todo momento los enlaces a los Estilos de aprendizaje y Consejos, pudiendo dejar el test a medio hacer y volver a mirar la información. En este caso, se pierden los datos del test y al volver al apartado del test, se debe empezar de nuevo.

El botón Atras, también está disponible, pudiendo volver a la pantalla principal al pinchar sobre él.

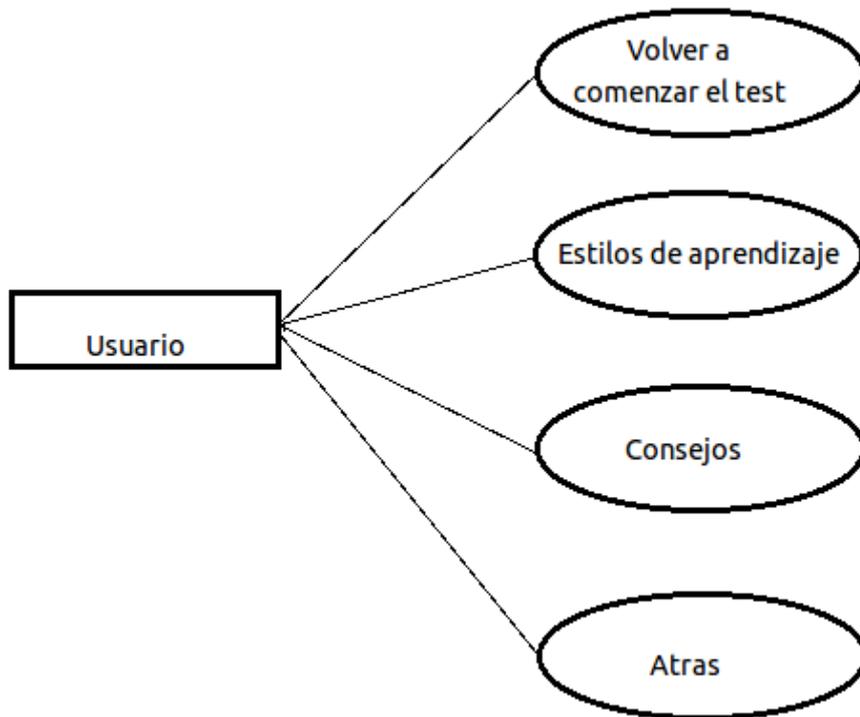


Figura 4.2 Segundo caso. Realización del test.

- **Tercer caso:** Estilos de aprendizaje.

Una vez que se está dentro del apartado Estilos de aprendizaje, el usuario puede volver al apartado Test, o al apartado Consejos o si quiere volver a la pantalla principal sólo tiene que pinchar el botón Atrás.

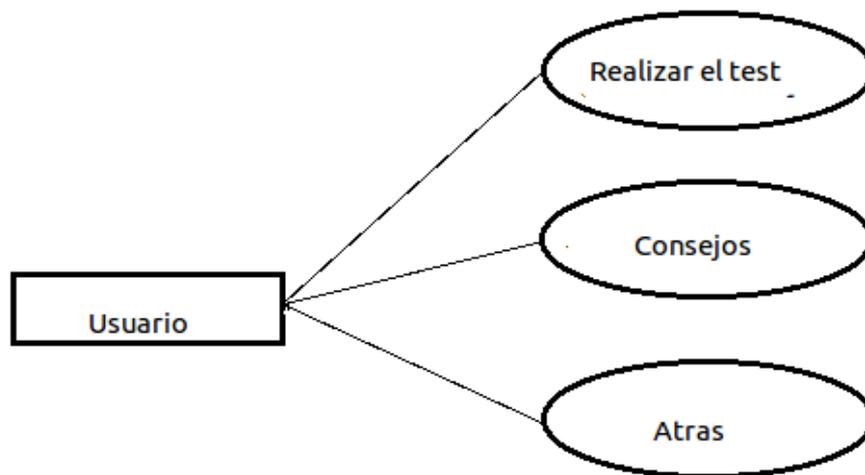


Figura 4.3 Tercer caso. Estilos de aprendizaje.

- **Cuarto caso:** Consejos.
Cuando se está dentro del apartado Consejos, el usuario puede volver al apartado Test, o puede volver a tener información sobre los Estilos de aprendizaje. Si quiere volver a la pantalla principal sólo tiene que pinchar el botón Atrás.

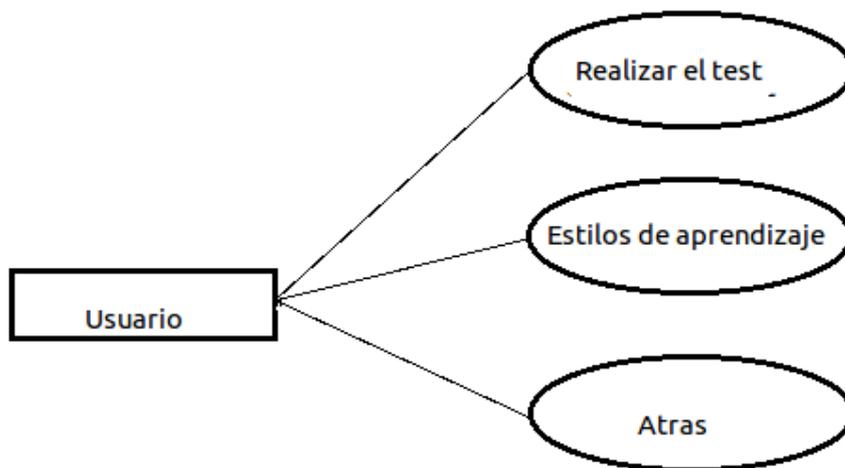


Figura 4.4 Cuarto caso. Consejos

- **Quinto caso:** Resultados

Una vez que se muestran los resultados, tenemos acceso a todos los apartados de la aplicación pudiendo realizar el test de nuevo, consultar la información sobre estilos de aprendizaje y consejos y volver a la pantalla principal pulsando el botón Atrás.

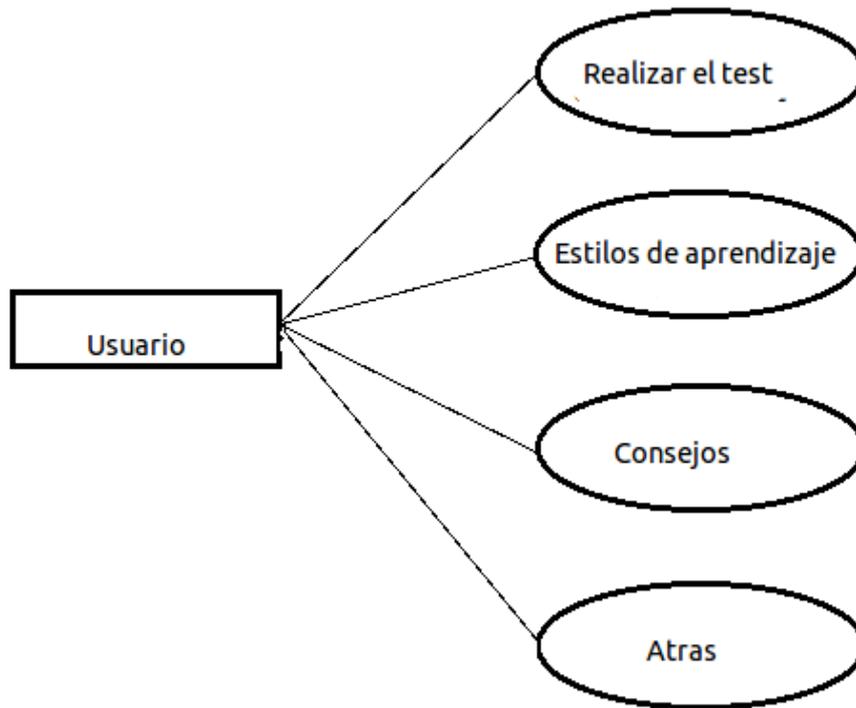


Figura 4.5 Quinto caso. Resultados.

3.2 Implementación

En el apartado que nos ocupa, se va a explicar la implementación de la aplicación una vez realizado el diseño, dando al lector una visión más detallada sobre el funcionamiento interno de la aplicación. Se explicarán también las decisiones tomadas a la hora de implementar ciertas funcionalidades.

3.2.1 Instalación de Famo.us

Para instalar Famo.us, sólo es necesario ir a su página web (www.famo.us) y descargar el famous-starter-kit. Una vez descargado es necesario un generador. Para tal caso, se utiliza yeoman generator. Este generador nos ofrece unas características, que hacen que nuestro sea generado y gestionado de manera más simple, como son un servidor de desarrollo que permite cargar los cambios en tiempo real, inyección del código que tenemos en configRequire.js y la utilización de un único comando para preparar la producción del proyecto. Para instalarlo, es necesario el repositorio de paquetes npm. Con este repositorio instalado en el sistema, se deben seguir los siguientes pasos:

- Instalar famous-generator:

```
$ npm install -g generator-famous
```

- Crear un directorio para nuestra aplicación y entrar en él. Una vez hecho ya podemos lanzar el generador de famous (yo famous):

```
$ mkdir famous-base  
$ cd famous-base  
$ yo famous
```

Para que funcione debe estar Grunt instalado en nuestro sistema.

Grunt es una herramienta que permite simplificar el proceso de construcción de proyectos en Javascript, ya que permite automatizar una serie de procesos a la hora de compilar el código Java. La razón de utilizar Grunt es que hoy en día, las aplicaciones Javascript se componen cada vez de más ficheros. Es inviable el desarrollo directamente en un entorno de producción, el mandar varios archivos Javascript al cliente es un proceso costoso y largo, ya que cada archivo supone una conexión HTTP, y con el crecimiento de las tecnologías móviles se busca minimizar las conexiones. Para ello se usan técnicas como la concatenación de varios archivos Javascript en un único archivo optimizado, de manera que solo se le mandará un archivo al cliente, ocultando así también la disposición de los archivos en el entorno de desarrollo. Para facilitar y automatizar estas tareas se utiliza Grunt. Usa el entorno de Node.js para funcionar, además del sistema de paquetes que proporciona Node (npm), por lo que es multiplataforma (siempre que la plataforma esté soportada por Node.js). El primer paso para instalar Grunt es instalar Node.js, si no se tiene ya instalado en el equipo, para ello basta con irse a la página oficial de Node y descargar e instalar el ejecutable de nuestra plataforma.

Una vez instalado Node se comprueba que esté bien escribiendo en la consola:

```
node --version
```

El siguiente paso será instalar Grunt en el sistema, para ello se usa el comando:

```
npm install -g grunt-cli
```

El "flag" -g indica a npm que se quiere instalar Grunt de manera global, lo que permitirá invocarlo desde cualquier parte dentro de la línea de comandos. Una vez instalado todo lo anterior y descargado el famous-starter-kit, ya se puede ir a la carpeta de la aplicación y lanzarla.

3.2.2 Lenguajes de programación y estructura del proyecto

A la hora de desarrollar en Famo.us no existe posibilidad de elección en cuanto a lenguajes a utilizar, pues todas las aplicaciones se crean sobre JavaScript, HTML5 y CSS3. En cuanto al IDE, se ha elegido Brackets.io ya que ofrece la posibilidad de tener una vista previa de los cambios en tiempo real.

A continuación se va a mostrar la estructura del proyecto dentro del entorno Brackets, que es aplicable a cualquier aplicación realizada con Famo.us:

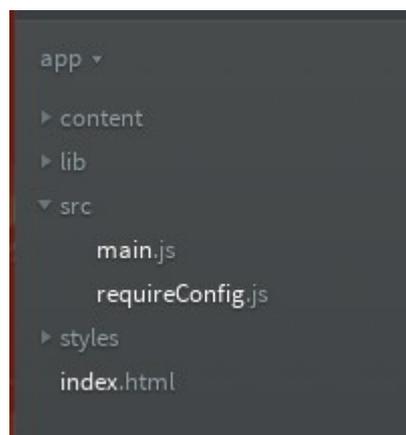


Figura 4.6 Estructura general del proyecto

Dentro de la aplicación, existen varios directorios. Dentro de /src, se encuentran los elementos raíz de la aplicación. En requireConfig.js está indicado el archivo que contiene el código con el que comienza la ejecución de nuestra aplicación, en este caso, main.js.

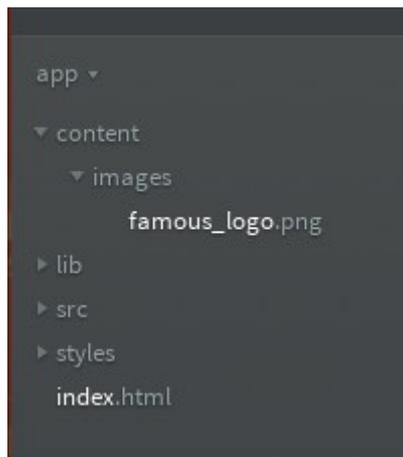


Figura 4.7 Directorio content

Dentro del directorio content, se encuentran las imágenes que se utilizan en la aplicación.

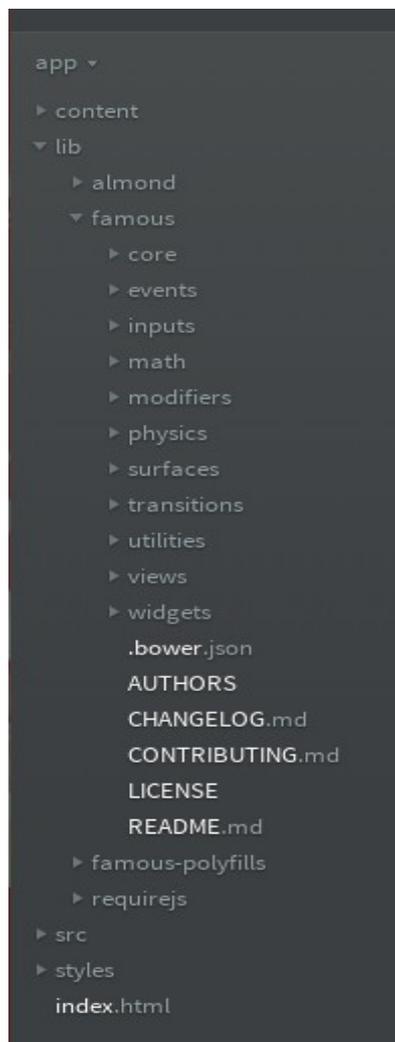


Figura 4.8 Directorio lib

Este directorio se corresponde con la librería, aquí se puede encontrar el código que ya está desarrollado, con el cual se puede ampliar la funcionalidad de la aplicación.

Las librerías que más se utilizarán, están dentro del directorio famous, ya que son las que contienen las funciones asociadas a Surfaces, Vistas, funciones matemáticas...

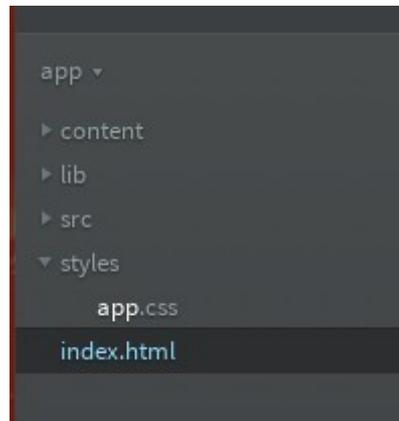


Figura 4.9 Directorio lib

En el directorio styles se encuentra el archivo app.css. Es el encargado del estilo general de la aplicación, aunque luego los elementos hijos irán cambiando sus propiedades, según lo vayan necesitando.

3.2.3 Creación de la aplicación web.

En este apartado se trata la parte de programación. Cómo ha sido codificada la aplicación, de qué partes está compuesta y también se hablará sobre los problemas que han ido surgiendo y cómo se han solucionado. Para tener una idea general, se ven a continuación las partes que componen la aplicación:

- HeaderFooterLayout: Es la estructura general de la página, donde se añaden las partes fundamentales de una página web ya que la aplicación se muestra en la pantalla principal como tal.
- Test: En esta parte se hablará sobre cómo ha sido realizado el test de Felder & Soloman.
- Cubo 3D: aquí se explicará cómo ha sido construido el cubo que muestra los resultados.

3.2.3.1 HeaderFooterLayout

Como es sabido, las páginas web se dividen en tres secciones: Header, Content y Footer. En esta aplicación web, hay una página principal y se decidió que tuviera este aspecto. Antes de nada, se muestra una imagen con el aspecto descrito.



Figura 4.11. Pantalla principal de la aplicación web

Como se puede observar, la cabecera o header tiene cuatro apartados: Test, Dimensiones en los estilos de aprendizaje, Consejos y Atrás. Cada uno de ellos ha sido tratado como un Surface. Un poco más adelante se verá como se comportan cuando son seleccionados.

El apartado central, el content, en un primer momento sólo mostrará un texto informativo que invita a la realización del test Felder & Soloman. Posteriormente, alojará el cubo 3D que muestra los resultados.

El pie de página o footer, se utiliza para darle un aspecto a la aplicación de una página web típica, ya que no ofrece ninguna funcionalidad.

Se pasa ahora a la descripción de cada uno de los elementos de la cabecera y cuáles son sus comportamientos cuando son seleccionados. Lo primero antes de todo, es ver como añadir la estructura de página web al Contexto:

```

mainContext = Engine.createContext();

var layout = new HeaderComponentLayout({
  headerSize: 100,
  footerSize: 50
});

.....
.....

mainContext.add(layout);

```

Con la función `createContext()`, se crea el elemento raíz, al cuál se irán añadiendo todos los elementos que pertenezcan a la aplicación.

A continuación se crea un elemento `HeaderFooterLayout`, el cuál se inicializa con los tamaños de la cabecera y del pie de página, el resto será el tamaño que tenga el content. Una vez creado y añadidos todos los elementos que formarán parte de él, se añade al elemento raíz, que en esta aplicación se llama `mainContext`.

TEST

La creación del `Surface Test`, se realiza indicando que pertenecerá a la cabecera. Esto se indica de la siguiente manera:

```

layout.header.add(testSurface = new Surface({
  size: [400, 100],
  content: 'Test',
  properties: {
    lineHeight: '100px',
    backgroundColor: '#610B5E',
    textAlign: 'center',
    color: '#fff'
  }
}));

```

Esta `Surface` se inicializa con las propiedades típicas de `CSS3`, como pueden ser el tamaño, el contenido, el color de la letra o del fondo, el alineado...

A continuación, se muestra como se ha codificado su comportamiento, es decir, que ocurre cuando se selecciona dicha `Surface`.

Cuando es seleccionada, debe mostrarnos el test. Esto se realiza con el manejo de eventos, para este caso en concreto, se utiliza la función `Surface.on('click')` que hace que el `Surface` quede a la escucha y cuando recibe un `'click'` realiza la función que se define en este evento.

```
testSurface.on('click', function() {  
    .....  
});
```

La funcionalidad del test se tratará en el siguiente apartado, de momento se muestra el aspecto de la aplicación cuando pinchamos sobre la surface Test.

The screenshot shows a web application interface with a navigation bar at the top containing four items: 'Test' (highlighted in dark purple), 'Dimensiones en los estilos de aprendizaje' (medium purple), 'Consejos' (light purple), and 'Atrás' (very light purple). Below the navigation bar, the 'Test' screen displays five questions, each with two radio button options:

- Pregunta 1.** Yo entiendo mejor algo después que yo:
 pruebo a hacerlo
 lo pienso
- Pregunta 2.** Yo prefiero que me consideren:
 realista
 innovador
- Pregunta 3.** Cuando pienso lo que hice ayer, lo que obtengo es:
 una imagen
 palabras
- Pregunta 4.** Yo tiendo a:
 entender los detalles del aspecto, pero algo indefinida su estructura general
 entender la estructura general, pero algo indefinidos los detalles
- Pregunta 5.** Cuando estoy aprendiendo algo nuevo, me ayuda:
 hablar sobre ello
 pensar sobre ello

Figura 4.12. Aplicación web al mostrar el Test.

El siguiente Surface se corresponde con Dimensiones en los estilos de aprendizaje. La programación de éste es muy similar a la anterior; la forma de crear el Surface es la misma y su funcionalidad se gestiona también con el evento Surface.on('click'). La diferencia de ésta radica en la posición, está a la derecha del surface Test, así que se necesita un nodo modificador para desplazarlo. Para poder gestionar este asunto, se encapsula la creación de ésta y su funcionalidad dentro de la función createEstilosModifiedSurface(), la cual es llamada posteriormente.

Antes de llamar a la función se debe crear su modificador de posición, para lo cual se utiliza la opción transform.translate() y por último sólo queda añadir la Surface con su modificador al elemento raíz mainContext. El código es el siguiente:

```

function createEstilosModifiedSurface() {
    estilosSurface = new Surface({
        size: [400, 100],
        content: 'Dimensiones en los estilos de aprendizaje',
        properties: {
            color: '#fff',
            lineHeight: '100px',
            textAlign: 'center',
            backgroundColor: '#CD69C9'
        }
    });

    // función que nos lleva a mostrar los cuatro estilos de aprendizaje
    estilosSurface.on('click', function() {
        .....
    });

    var estilosModifier = new StateModifier({
        transform: Transform.translate(400, 0, 0)
    });

    createEstilosModifiedSurface();

    mainContext.add(estilosModifier).add(estilosSurface);
}

```

La funcionalidad de este apartado se encapsula dentro de la función `Surface.on('click')`. En este caso, se muestra información sobre las cuatro dimensiones en los estilos de aprendizaje dentro de un `GridLayout`. Un `GridLayout`, es una vista que se compone de varias `Surfaces` con posición fija, no tienen scroll ni transiciones por ellas mismas.

El código que muestra la creación de este `GridLayout` es el siguiente:

```

estilosSurface.on('click', function() {

    var colorEstilos = ["#D02090", "#EE7AE9", "#DA70D6", "#E066FF"];
    var grid = new GridLayout({
        dimensions: [4, 1]
    });
    var surfaces = [];
    grid.sequenceFrom(surfaces);
    for(var i = 0; i < 4; i++) {
        surfaces.push(new Surface({
            content: estilosAprend[i],
            size: [undefined, undefined],
            properties: {
                marginTop: '100px',
                backgroundColor: colorEstilos[i],
                color: "black",
                lineHeight: '20px',
                textAlign: 'justify',
                paddingTop: '20px',
                paddingLeft: '20px',
                paddingRight: '15px',
                fontSize: '14px'
            }
        }));
    }
    mainContext.add(grid);
});

```

Lo primero que hay que hacer es crear un nuevo GridLayout de las dimensiones deseadas. Se comporta como una matriz y puede tener dimensiones nxm, en este caso se crea un grid de 4x1. A través del método sequenceFrom(), se añaden las superficies que se necesitan, para este caso son cuatro. Para inicializarlas se crea un bucle y por cada posición se asigna un contenido y un color diferente. El resto de las propiedades son las mismas para todas las Surfaces. El contenido se toma de un array que contiene el texto de cada uno de los estilos, de esta manera se recorre con el bucle for y en cada posición, toma el valor deseado. Este array en la aplicación se llama “estilosAprend”. Una vez recorrido el bucle completo e inicializadas las Surfaces, solo queda añadir el grid al elemento raíz mainContext.

Se muestra el resultado en la siguiente imagen:

Test	Dimensiones en los estilos de aprendizaje	Consejos	Atrás
<p>ACTIVO O REFLEXIVO</p> <p>La primera dimensión distingue entre una manera activa o reflexiva de procesar la información. Los estudiantes activos, trabajan mejor aplicando y probando a partir del material de trabajo, suelen estar interesados en trabajar en grupo para poder discutir sobre los resultados obtenidos. En contraste, los estudiantes reflexivos, prefieren pensar y reflexionar sobre el material y también trabajar solos o en grupos pequeños.</p>	<p>PERCEPTIVO O INTUITIVO</p> <p>En la segunda dimensión, distinguimos entre alumnos perceptivos e intuitivos. A los alumnos perceptivos les gusta aprender hechos y concretar con el material de aprendizaje. Suelen resolver los problemas con un enfoque estándar y tienden a ser pacientes con los detalles. Son personas a las que les considera realistas, sensatas y prácticas, con lo cual les gusta aplicar sus resultados al mundo real. Por otro lado, están los alumnos intuitivos que prefieren aprender con materiales abstractos y sus significados subyacentes. Son más capaces de descubrir posibilidades y relaciones entre varias materias y tienden a innovar y ser más creativos que los alumnos perceptivos.</p>	<p>VISUAL O VERBAL</p> <p>La diferencia en esta dimensión radica en que hay alumnos que prefieren recibir la información de manera visual, con diagramas, dibujos, diagramas de flujo, etc, o a través de una representación textual, sin tener en cuenta si es hablada o escrita.</p>	<p>SECUENCIAL O GLOBAL</p> <p>El cuarto y último caso, los alumnos están caracterizados por la forma de entender una materia. Un alumno secuencial, aprende en pequeños pasos y tiene una línea de progreso. Intentan seguir la manera lógica de llegar a los resultados. En contraste, tenemos los alumnos de tipo globales, a los cuales les gusta tener conocimiento sobre la materia completa y avanzan en grandes pasos. En un primer momento, no hay conexiones entre las diferentes partes del objeto de estudio pero después de haber entendido las distintas partes, de repente, encuentran la conexión y consiguen la visión de la materia completa. Suelen ser capaces de resolver problemas complejos, encontrar conexiones entre áreas diferentes pero encuentran dificultades para explicar como han llegado al resultado.</p>

Figura 4.13 Dimensiones en los estilos de aprendizaje.

El siguiente apartado a tratar es la Surface correspondiente a Consejos. La manera de crearla, definir su comportamiento y de encapsular toda la información es similar a la Surface Dimensiones en los estilos de Aprendizaje y el código correspondiente a su creación, definición de la función que maneja el evento .on y la creación de su nodo modifier es el siguiente:

```
function createConsejosModifiedSurface() {
    consejosSurface = new Surface({
        size: [400, 100],
        content: 'Consejos',
        properties: {
            color: '#fff',
            lineHeight: '100px',
            textAlign: 'center',
            backgroundColor: '#EE82EE'
        }
    });

    consejosSurface.on('click', function() {
        .....
    });
}
```

La diferencia entre los dos apartados radica en su funcionalidad. Aunque los dos se centran en mostrar información, lo hacen de forma diferente ya que en Dimensiones en los estilos de aprendizaje se utilizó un GridLayout que se compone de Surfaces estáticas y en Consejos se utiliza un ScrollView, que a diferencia del anterior, son Surfaces dinámicas y se mueven haciendo scroll en la pantalla. El código que lo define es el siguiente:

```
consejosSurface.on('click', function() {

    var scrollview = new Scrollview();
    var surfaces = [];
    var colorConsejos = ["#D02090", "#EE7AE9", "#DA70D6", "#E066FF"];

    scrollview.sequenceFrom(surfaces);
    for(var i=0; i<4; i++){
    draggable = new Draggable( {
        xRange: [0, 0],
        yRange: [0, 0],
    });

    var item = new Surface({
        content: consejos[i],
        size: [undefined, 500],
        properties: {
            backgroundColor: colorConsejos[i],
            lineHeight: "30px",
            textAlign: "justify",
            paddingLeft: '40px',
            paddingRight: '80px',
            paddingTop: '20px',
            paddingBottom: '20px',
            marginTop: '100px',
            fontSize: '15px'
        }
    });

    var node = new RenderNode(draggable);
    node.add(item);

    item.pipe(draggable);
    item.pipe(scrollview);

    surfaces.push(node);
    }
    mainContext.add(scrollview);
});
```

En un primer momento se define un elemento ScrollView y un array de Surfaces. A través del método sequenceFrom() se van añadiendo estas Surfaces al ScrollView, que será el resultado mostrado en la aplicación.

Para poder añadir las Surfaces al ScrollView, se crean elementos modificadores llamados Draggables. Estos elementos permiten que cada una de las surfaces tenga posibilidad de movimiento, definiendo con xRange e YRange la posición final en píxeles que pueden tomar. En nuestra aplicación, por cuestión de diseño, se dejan fijas inicializando estos dos parámetros con valor cero. Una vez creados estos elementos, se pasa a la creación de las surfaces a añadir inicializándolas con las propiedades indicadas en el código anterior. Esta vez el contenido se obtiene del array “consejos”, el cuál contiene el texto informativo para cada estilo de aprendizaje.

Los elementos Draggables necesitan de un contenedor, para eso se crea un nodo RenderNode, al cuál se añade cada surface creada anteriormente. Para que el método .on tenga efecto es necesario utilizar el evento pipe, el cual empuja los eventos hacia los elementos inferiores del Render Tree y así poder reaccionar ante el click en la superficie deseada. Ya sólo quedaría añadir el Scrollview al mainContext para que se pueda mostrar el resultado en pantalla.

A continuación se muestran varias imágenes con el resultado del ScrollView que corresponde al apartado Consejos:



Figura 4.14 Parte superior del apartado Consejos

Si eres un estudiante reflexivo, entenderás mejor la información cuando puedas ver la aplicación real de ésta. Por tanto, en clases donde abunde el material teórico y abstracto, puedes pedirle al profesor ejemplos específicos de conceptos y procedimientos, así como tratar de descubrir como esos conceptos se llevan a la práctica.

Si aún así, necesitaras más ejemplos se recomienda realizar tormentas de ideas con amigos y compañeros de clase.

SI ERES UN ALUMNO INTUITIVO

Si eres un estudiante intuitivo y perteneces a una clase donde todo el material debe ser memorizado, es probable que llegues a estar aburrido en poco tiempo y dejes de prestar atención. Una solución puede ser pedirle al profesor que proporciones teorías que vinculen los hechos o buscarlos por tí mismo. Recuerda ser paciente y tener en cuenta los detalles, así que toma tu tiempo para leer las preguntas ante un exámen y asegúrate de repasar las respuestas de éste.

SI ERES UN ALUMNO VISUAL

Si eres un estudiante de este tipo, busca diagramas, imágenes, fotografías o cualquier otra representación visual. Solicita al profesor material audiovisual sobre la asignatura, muchas de ellas contienen CD con toda la información.

También es útil preparar mapas conceptuales, así como codificar con colores los apuntes a la hora de estudiar, de manera que cada tópico se relacione con un color.

Figura 4.15 Parte intermedia del apartado Consejos

SI ERES UN ALUMNO VERBAL

Si eres un estudiante de tipo verbal, un buen recurso es redactar resúmenes con tus propias palabras, todo el material que tengas de clase.

También te beneficia trabajar en grupo, ya que puede incrementar tu comprensión escuchando las explicaciones de tus compañeros. Ésto se incrementa cuando son ellos los que dan las explicaciones y tú las recibes, aunque no te acomodes, porque darlas también tiene un buen resultado para tí.

SI ERES UN ALUMNO SECUENCIAL

Si eres un estudiante de este tipo y tu profesor no expone la asignatura con orden y paso a paso, solicítale que complete los pasos que falten o busca en material adicional para poder completar la información.

Toma tu tiempo para esquematizar el material siguiendo un orden lógico, a largo plazo, te hará ahorrar tiempo.

También puedes intentar mejorar tu capacidad global, intentando relacionar los nuevos tópicos con los conocimientos ya adquiridos.

SI ERES UN ALUMNO GLOBAL

Si eres un estudiante de tipo global, antes de empezar a leer las secciones de un tema, da una lectura completa a todo el contenido, evitarás volver muchas veces a partes del texto que no recuerdes al ir avanzando.

Es muy beneficioso que relaciones las partes que estás estudiando con lo que ya sabes, pidiéndole al profesor que te ayude a ver las conexiones o buscando en referencias alternativas. Sé paciente, al principio no verás la relación entre distintos conceptos pero según vayas avanzando se irá clarificando el camino.

Figura 4.16 Parte inferior del apartado Consejos

Para acabar con el apartado del header, sólo queda tratar la Surface Atrás. Ésta tiene un comportamiento muy sencillo, a través de la función `location.reload()` se hace que la aplicación vuelva a la pantalla principal este donde esté en el momento de seleccionarla. La creación de esta Surface, la definición de su comportamiento y su modificador de posición en la cabecera, se crea de forma similar a los apartados anteriores. Su código es el siguiente:

```
function createAtrasModifiedSurface() {
  backSurface = new Surface({
    size: [101, 100],
    content: 'Atrás',
    properties: {
      backgroundColor: '#EEAEEE',
      lineHeight: '100px',
      textAlign: 'center',
      cursor: 'pointer',
      color: 'white'
    }
  });

  backSurface.on('click',function(){
    location.reload();
  });
  mainContext.add(backSurface);
}
```

El resultado de pulsar el apartado Atrás, siempre será el mismo, la pantalla principal:



Figura 4.17 Pantalla principal tras pulsar Atrás.

Para añadir el content en la estructura principal de la aplicación, se utilizó la función `layout.content.add()`. Esta función crea una surface única para el contenido, se inicializan sus propiedades y finalmente, se añade al `mainContext`. El código es el siguiente:

```
layout.content.add(contentSurface = new Surface({
    size: [undefined, undefined],
    content: '<h3>¡OPTIMIZA TU TIEMPO DE ESTUDIO!</h3><br><br><p>En esta aplicación
encontrarás la información que necesitas sobre los distintos estilos de aprendizaje de los alumnos de
ingeniería.<br><br> Descubre cuáles son tus estilos y mira los consejos<br> que ofrecemos para
mejorar tus resultados.<br><br><br>Test de Felder & Soloman</p><br><br><br>',
    properties: {
        paddingTop: '75px',
        lineHeight:'20px',
        textAlign: 'center',
        backgrounColor: '#0cf'
    }
}));
```

Para añadir el footer o pie de página el procedimiento es similar al anterior, la diferencia es que para añadirlo, utilizamos la función `layout.footer.add()`. Se muestra en el siguiente código.

```
layout.footer.add(new Surface({
  size: [undefined, 50],
  content: 'Aula Virtual. UPCT',
  properties: {
    color: '#fff',
    paddingRight: '25px',
    lineHeight: '50px',
    textAlign: 'right',
    backgroundColor: '#610B5E',
    fontSize: '15px'
  }
}));
```

Y con el footer se termina este apartado sobre cómo añadir la cabecera a nuestra aplicación web. En el siguiente apartado se podrá comprobar cuál es la implementación del test de Felder & Soloman.

3.2.3.2 Test

En este apartado se trata la codificación del test Felder & Soloman.

En un primer momento, se optó porque el diseño fuera un `ScrollView`, cada pregunta se añadía a una `Surface` y luego todas se añadían a la vista mencionada. Esta solución no fue posible ya que, los elementos HTML que van relacionados entre sí, no pueden colocarse en superficies diferentes. Famo.us, no permite esta acción. La solución por la que se optó fue meter todo el contenido HTML en un único `Surface`, permitiendo así encontrar todas las preguntas y sus respuestas sin problema alguno.

Esta única `Surface`, se debe añadir a un `ScrollView` para poder ir bajando la pantalla y poder seguir rellenando las preguntas del test. Si no se hiciera de esta manera, la página sería estática y sólo se podrían ver las cinco primeras preguntas. La manera de crear esta surface viene dada en la siguiente porción de código:

```

var scrollview = new Scrollview();
// función que nos lleva a mostrar las preguntas del test
testSurface.on('click', function() {

    var surfaces = [];
    var scrollSize = 0;

    scrollview.sequenceFrom(surfaces);

    draggable = new Draggable( {
        xRange: [0, 0],
        yRange: [0, 0],
    });

    var item = new Surface({
        content: formulario,
        size: [undefined, 4900],
        properties: {
            paddingTop: '10px',
            paddingLeft: '20px',
            marginTop: '100px',
            backgroundColor: '#F6CEF5',
            textAlign: 'left'
        }
    });

    var node = new RenderNode(draggable);
    node.add(item);

    item.pipe(draggable);
    item.pipe(scrollview);
    surfaces.push(node);
    mainContext.add(scrollview);
}

```

La explicación de este código es similar al apartado Consejos de la cabecera. La diferencia aquí es en vez de crear varias superficies al Scrollview, se añade solamente una que es la que contiene todas las preguntas.

El contenido de la surface item, es un string llamado formulario que contiene en formato HTML todos los inputs correspondientes a las preguntas del test. Cada input se define con los siguientes atributos:

- type: 'radio'. Este tipo indica que la respuesta está acompañada de un radiobutton el cuál al ser pulsado activará el atributo checked a true.
- Name: 'respx'. Este atributo es un identificador con el cuál se pueden asociar las respuestas a las preguntas correspondientes. Las dos respuestas que pertenecen a una misma pregunta tendrán el mismo name. Para nuestro test, se tienen 44 identificadores name diferentes para poder calcular después los estilos de aprendizaje del alumno que está completando el test.
- Value: 'x'. Este valor es el asociado a la pregunta seleccionada. Cuando recogemos los valores checked, son éstos a los que se hacen referencia.

Para ver el resultado final del input, se muestra como ejemplo el código HTML de la primera pregunta, dando por sentado, que el resto serán similares pero cambiando el texto y el identificador name.

```
Pregunta 1. <br>
Yo entiendo mejor algo después que yo: <br>
<input type='radio' name='resp1' value = '1' >pruebo a hacerlo<br>
<input type='radio' name='resp1' value = '-1'/>lo pienso<br><br>
```

Una vez completado el test, se deben evaluar las respuestas para conseguir los resultados. Para esto se crea una surface que toma el papel de un botón Submit. Esta surface se coloca justo al final del Scrollview, para cuando el test sea finalizado, quede accesible a la hora de hacer click en él. Se añade justo después de la surface del test, con su modificador de posicionamiento de la misma manera que se añadió la surface correspondiente a las preguntas del test:

```
var resultSurface = new Surface({
  content: 'Resultados',
  size: [110, 25],
  properties: {
    zIndex: '1',
    backgroundColor: '#848484',
    textAlign: 'center',
    color: '#FFFFFF'
  }
});

var nodeResult = new RenderNode(draggable);
node.add(modifierTwo).add(resultSurface);
surfaces.push(node);
mainContext.add(scrollview);
```

La funcionalidad del botón Resultados es fácil de adivinar. Se centra en calcular los estilos de aprendizaje. Para ello, en la función surface.on('click'), se hace una llamada a la función calcularEstilos(), la cuál será explicada justo después de esta porción de código.

Aclarar que justo antes de llamar a la función debemos refrescar la pantalla principal para que desaparezcan las preguntas del test antes de que aparezcan los resultados. Con este motivo, se ha utilizado la función render.

```
resultSurface.on('click',function() {
  scrollview.render = function(){return null;}
  contentSurface.render = function(){return null;}
  estilos = calcularEstilos();

  box = createBoxResults(estilos);

  mainContext.add(box);
});
```

Justo después se hace una llamada a la función `createBoxResults()`, la cuál es la encargada de mostrar el cubo 3D con los resultados. Esta función será estudiada en el siguiente apartado.

Para concluir este apartado, nos queda el estudio de la función `calcularEstilos()`. La primera parte de esta función se centra en rescatar la respuesta seleccionada para cada pregunta. Esto se hace utilizando la función `getElementsByName('name')` siendo `name` el identificador dado en cada `input` de cada pregunta. Mostramos como se rescatan los valores de la pregunta uno como ejemplo:

```
resp1 = document.getElementsByName('resp1');
for (var i = 0; i < resp1.length; i++){
    if(resp1[i].checked==true)
        respChecked1 = resp1[i].value;
}
respChecked = respChecked.concat(respChecked1);
```

Con la variable `resp1` recogemos los dos valores asociados al `name`: `'resp1'`. Una vez recogidos, se recorren con un bucle `for` para comprobar cuál de estas dos opciones tiene el atributo `checked` a `true`, cuando da con él, guarda el valor en una variable intermedia y luego lo concatena a una variable que contendrá todos los valores de las respuestas chequeadas. Esta variable en la función se llama `respChecked`.

Una vez hecho esto para todas las preguntas, el siguiente paso es comprobar que todas las respuestas han sido contestadas. Se recorre con otro bucle `for` el array `respChecked` y si encuentra alguna posición a cero, entonces se mostrará en pantalla un mensaje de alerta indicando el número de la pregunta que no se ha contestado. El resultado en la aplicación se muestra en la siguiente imagen:

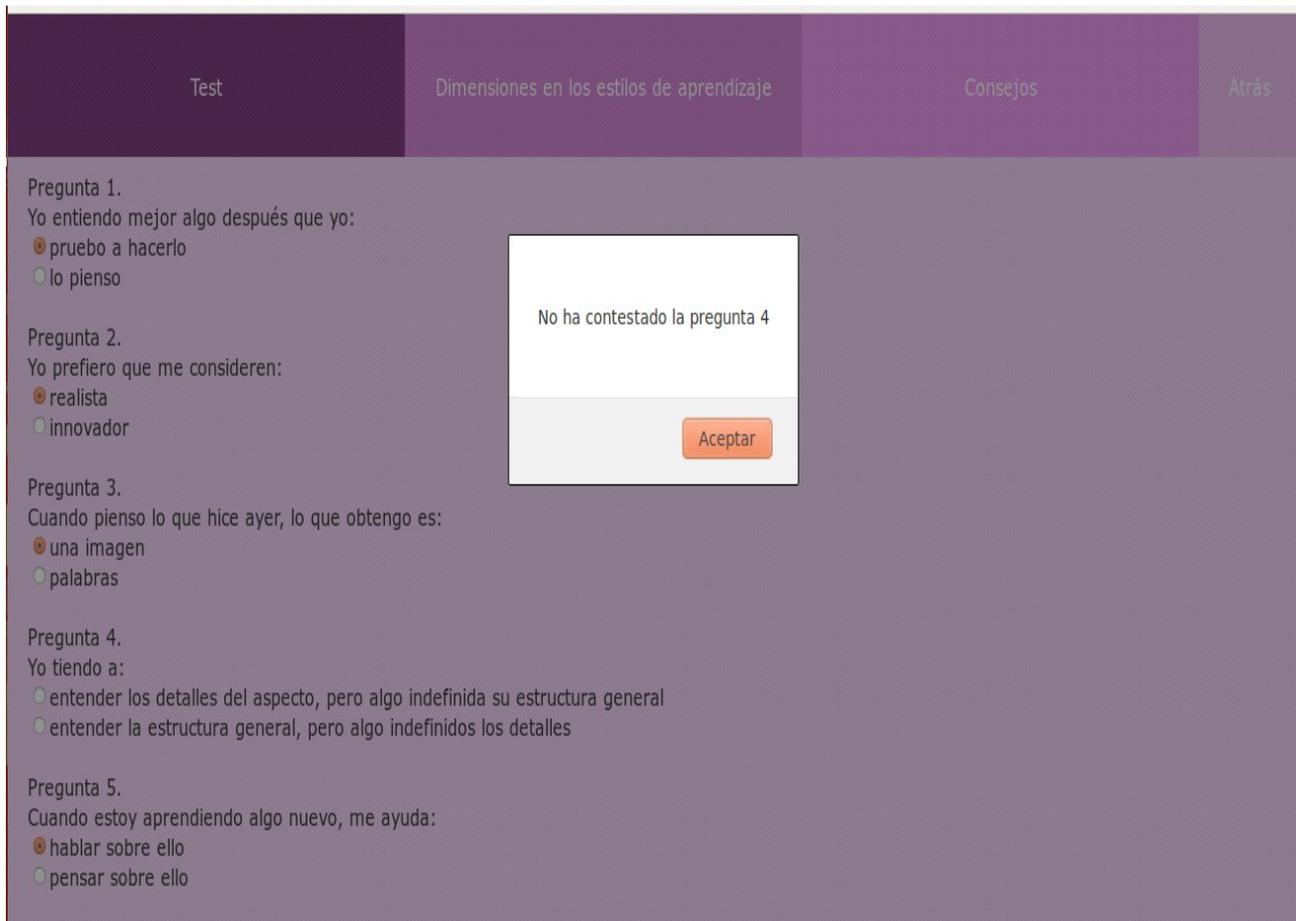


Figura 4.18 Mensaje de alerta si no se ha contestado alguna pregunta.

Lo que sigue en la función es el cálculo de los cuatro estilos. Atendiendo a la Tabla 1: Grupos semánticos asociados al cuestionario ILS, podemos ver que pregunta está asociada a cada estilo de aprendizaje.

Una vez sumandos todos los valores de las respuestas (1 y se elige la opción a y -1 si se elige la opción b), se sabe cuál es la tendencia dentro de cada estilo. Esta tendencia puede ser equilibrada, moderada o fuerte. Al encontrar las cuatro variables con sus tendencias, se crea un array estilos que contendrá esas cuatro variables, las cuáles serán enviadas al cubo 3D para mostrar así los resultados. Esta parte, será estudiada en el apartado siguiente. El código del calculo del array estilos es el siguiente:

```
// Asignación del valor para la dimensión Activo/Reflexivo
if(-11<=actRef<=-9)
  actRefSolucion = "Reflexivo. Tendencia fuerte";
else if (-7<=actRef<=-5)
  actRefSolucion = "Reflexivo. Tendencia moderada";
else if (-3<=actRef<=-1)
  actRefSolucion = "Reflexivo. Tendencia equilibrada";
else if (1<=actRef<=3)
  actRefSolucion = "Activo. Tendencia equilibrada";
else if (5<=actRef<=7)
```

```

        actRefSolucion = "Activo. Tendencia moderada";
        else if (9<=actRef<=11)
            actRefSolucion = "Activo. Tendencia fuerte";

// Asignación del valor para la dimensión Perceptivo/Intuitivo
if(-11<=percInt<=-9)
    percIntSolucion = "Intuitivo. Tendencia fuerte";
else if (-7<=percInt<=-5)
    percIntSolucion = "Intuitivo. Tendencia moderada";
else if (-3<=percInt<=-1)
    percIntSolucion = "Intuitivo. Tendencia equilibrada";
else if (1<=percInt<=3)
    percIntSolucion = "Perceptivo. Tendencia equilibrada";
else if (5<=percInt<=7)
    percIntSolucion = "Perceptivo. Tendencia moderada";
else if (9<=percInt<=11)
    percIntSolucion = "Perceptivo. Tendencia fuerte";

// Asignación del valor para la dimensión Visual/Verbal
if(-11<=visVerb<=-9)
    visVerbSolucion = "Verbal. Tendencia fuerte";
else if (-7<=visVerb<=-5)
    visVerbSolucion = "Verbal. Tendencia moderada";
else if (-3<=visVerb<=-1)
    visVerbSolucion = "Verbal. Tendencia equilibrada";
else if (1<=visVerb<=3)
    visVerbSolucion = "Visual. Tendencia equilibrada";
else if (5<=visVerb<=7)
    visVerbSolucion = "Visual. Tendencia moderada";
else if (9<=visVerb<=11)
    visVerbSolucion = "Visual. Tendencia fuerte";

// Asignación del valor para la dimensión Secuencial/Global
if(-11<=secGlob<=-9)
    secGlobSolucion = "Global. Tendencia fuerte";
else if (-7<=secGlob<=-5)
    secGlobSolucion = "Global. Tendencia moderada";
else if (-3<=secGlob<=-1)
    secGlobSolucion = "Global. Tendencia equilibrada";
else if (1<=secGlob<=3)
    secGlobSolucion = "Secuencial. Tendencia equilibrada";
else if (5<=secGlob<=7)
    secGlobSolucion = "Secuencial. Tendencia moderada";
else if (9<=secGlob<=11)
    secGlobSolucion = "Secuencial. Tendencia fuerte";

estilos = [actRefSolucion,percIntSolucion,visVerbSolucion,secGlobSolucion];
return estilos;

```

Una vez explicada la función `calcularEstilos()`, sólo queda ver cómo se crea el cubo 3D para mostrar los resultados. De esto precisamente trata el siguiente apartado.

3.2.3.3 Gráfico 3D. El cubo.

En este apartado se aborda la explicación de cómo crear un gráfico 3D, en concreto, un cubo. A parte de crear un cubo, se le dotará de movimiento con un click de ratón y además mostrará los resultados obtenidos en el apartado anterior.

Se comienza por ver cómo crear un cubo. Se define la función `createBox(width, height, depth)` para ello y los argumentos de entrada, son los que definirán el tamaño de éste.

El concepto fundamental que hay que conocer a la hora de crear el cubo es el siguiente: el cubo está compuesto por seis caras, cuatro laterales, una superior y una inferior. Estas cuatro caras, serán interpretadas como superficies. Para la creación de éstas, definimos la función `createSide(params)`, la cuál será llamada cuando se quiera dar forma a las caras del cubo.

El código que define la función `createSide(params)` es el siguiente:

```
function createSide(params){
    var surface = new Surface({
        size: params.size,
        content: params.content,
        classes: params.classes,
        properties: params.properties,
    });

    var modifier = new Modifier({
        transform: params.transform
    });

    .....
}
```

Se puede observar como la creación de superficies con esta función queda expuesta a los parámetros pasados como argumentos. De esta manera, cada superficie puede tener un estilo propio. Al igual que la función, existe un modificador general, que dependerá en cada caso de los argumentos ofrecidos en la llamada a la función `createBox()`.

Se puede decir que esta es la parte más importante a la hora de crear el cubo, aunque quedan un par de detalles por mencionar para completar la funcionalidad de la función `createBox()`. En primer lugar, debemos mencionar que cada superficie debe ir asociado a un `RenderNode`. Igual que en el caso del `ScrollView`, aquí se debe asociar las superficies a un nodo `RenderNode` para poder tener flexibilidad a la hora de dotar de movimiento y ajustes a las superficies. Por tanto, el elemento `box`, será de este tipo.

La parte siguiente es crear las surfaces, una por cada cara. El código queda de la siguiente manera en la aplicación:

```
// Cara frontal. ¿ACTIVO O REFLEXIVO?
createSide({
  size: [width, height],
  content: '<h5>¿ACTIVO O REFLEXIVO?</h5><p>Tu estilo es: ' + estilos[0] + '</p>',
  classes: ['red-bg'],
  properties: {
    lineHeight: '20px',
    textSize: '20px',
    textAlign: 'center',
    backgroundColor : '#993663'
  },
  transform: Transform.translate(0, 0, depth / 2)
});

// Cara trasera. ¿VISUAL O VERBAL?
createSide({
  size: [width, height],
  content: '<h5>¿VISUAL O VERBAL?</h5><p>Tu estilo es: ' + estilos[2] + '</p>',
  properties: {
    lineHeight: '20px',
    textAlign: 'center',
    backgroundColor: '#BF5D99',
    fontSize: '18px',
    color: 'black'
  },
  transform: Transform.multiply(Transform.translate(0, 0, - depth / 2),
Transform.multiply(Transform.rotateZ(Math.PI), Transform.rotateX(Math.PI))),
});

// Cara de arriba. Título del test
createSide({
  size: [width, depth],
  content: '<h5>TEST FELDER & SOLOMAN</h5>',
  properties: {
    lineHeight: '20px',
    textAlign: 'center',
    backgroundColor: '#3C1527',
    color: 'white'
  },
  transform: Transform.multiply(Transform.translate(0, -height / 2, 0),
Transform.rotateX(Math.PI/2)),
});

// Cara de abajo. Amplía tus resultados
createSide({
  size: [width, depth],
  content: '<h5>Si quieres más información pincha en la pestaña consejos</h5>',
```

```

    properties: {
      lineHeight: '20px',
      textAlign: 'center',
      backgroundColor: '#EAC5D7',
      color: 'black'
    },
    transform: Transform.multiply(Transform.translate(0, height / 2, 0),
Transform.multiply(Transform.rotateX(-Math.PI/2), Transform.rotateZ(Math.PI))),
  });

// Cara izquierda. ¿SECUENCIAL O GLOBAL?
createSide({
  size: [depth, height],
  content: '<h5>¿SECUENCIAL O GLOBAL?</h5><p>Tu estilo es: ' + estilos[3]
+'</p>',
  properties: {
    lineHeight: '20px',
    textAlign: 'center',
    backgroundColor: '#8F1741',
    color: 'black'
  },
  transform: Transform.multiply(Transform.translate(-width / 2, 0, 0),
Transform.rotateY(-Math.PI/2))
});

// Cara derecha. ¿PERCEPTIVO O INTUITIVO?
createSide({
  size: [depth, height],
  content: '<h5>¿PERCEPTIVO O INTUITIVO?</h5><p>Tu estilo es: ' + estilos[1]
+'</p>',
  properties: {
    lineHeight: '20px',
    textAlign: 'center',
    backgroundColor: '#CD729C',
    color: 'black',
  },
  transform: Transform.multiply(Transform.translate(width / 2, 0, 0),
Transform.rotateY(Math.PI/2))
});

```

La parte a destacar del código, es el uso de `Transform.translate` y `Transform.multiply`. Con `Transform.translate` se consigue la posición deseada de cada cara, haciendo que encajen todas las superficies, dando como resultado la forma de cubo.

`Transform.multiply` es utilizado para jugar con la opacidad de las superficies. En `Famo.us`, se juega con esta propiedad utilizando este modificador, ya que las opacidades se consiguen multiplicando todas las que entren en juego a la hora de representar toda la información.

Una vez creadas todas las caras y añadidas al RenderNode box, sólo queda hacer un return del cubo creado.

Queda una parte importante por ver y es la que pone en movimiento el gráfico. Dentro de la función createBoxResults() se encuentra en primera instancia el siguiente código:

```
var quaternion = new Quaternion(1, 0, 0, 0);
    // El cubo comienza rotando sobre el eje y.
    var smallQuaternion = new Quaternion(185, 0, 1, 0);

    var rotationModifier = new Modifier({
        origin: [0.5, 0.5]
    });

    // Asociamos la rotación de la caja al quaternion
    rotationModifier.transformFrom(function() {
        return quaternion.getTransform();
    });

    mainContext.add(rotationModifier)
        .add(createBox(230, 230, 230));

    // Aquí se crea la rotación
    Engine.on('prerender', function() {
        // You combine rotations through quaternion multiplication
        quaternion = quaternion.multiply(smallQuaternion);
    });

    // Modificación de la velocidad y la dirección de rotación cuando hacemos click en la
pantalla
    Engine.on('click', function() {
        var x = (Math.random());
        var y = (Math.random());
        var z = (Math.random());
        // smallQuaternion = new Quaternion(220, x, y, z);
        smallQuaternion = new Quaternion(220, x, y, 0);

    });
```

Quaternion es un elemento que ofrece la librería Math perteneciente a Famo.us. Este elemento permite jugar con las rotaciones y velocidades de los elementos como se verá ahora.

En un primer momento se desea que el cubo rote pero sólo sobre el eje Y y para poder ver los resultados de una manera más fácil y entendible. Para eso, se crea un Quaternion indicándole a sus parámetros que el cubo debe rotar sólo en el eje Y (tercer argumento), y también se le indica la velocidad a la que se quiere que rote en un primer momento con el primer argumento.

```
var smallQuaternion = new Quaternion(185, 0, 1, 0);
```

Una vez indicado lo anterior, hay que indicar en que posición de la pantalla se desea que esté centrado nuestro gráfico. Para eso, utilizamos el siguiente modificador con el atributo origin inicializado para tal caso.

```
var rotationModifier = new Modifier({  
    origin: [0.5, 0.5]  
});
```

Sólo queda asociar la rotación de la caja al elemento Quaternion. Esto será realizado con la función transformFrom().

```
// Asociamos la rotación de la caja al quaternion  
rotationModifier.transformFrom(function() {  
    return quaternion.getTransform();  
});
```

Y ahora sí, ya se puede añadir al mainContext el gráfico 3D.

```
mainContext.add(rotationModifier)  
    .add(createBox(230, 230, 230));
```

Sólo por probar efectos sobre el gráfico se decidió hacer rotar al cubo sobre sus tres ejes al pulsar click en cualquier parte de la pantalla. El efecto de rotación y velocidad cambia en cada click de manera aleatoria. Este efecto se consigue con el siguiente código

```
// Aquí se crea la rotación
Engine.on('prerender', function() {
  // You combine rotations through quaternion multiplication
  quaternion = quaternion.multiply(smallQuaternion);
});

// Modificación de la velocidad y la dirección de rotación cuando hacemos click en la
pantalla
Engine.on('click', function() {
  var x = (Math.random());
  var y = (Math.random());
  var z = (Math.random());
  // smallQuaternion = new Quaternion(220, x, y, z);
  smallQuaternion = new Quaternion(220, x, y, 0);

});
```

Con la función `Engine.on("prerender")` se consigue recalculer el giro de rotación mientras que con la función `Engine.on('click')` se obtiene en cada click del ratón una velocidad distinta.

De esta manera, se anima el cubo explorando así algunas de las funciones existentes dentro de la librería Famo.us. La siguiente imagen muestra el cubo con los resultados del test:

¿PERCEPTIVO O INTUITIVO?

Tu estilo es: Perceptivo.
Tendencia equilibrada

Figura 4.19 Mostrando los resultados con el gráfico 3D.

Capítulo 4. Conclusiones y trabajos futuros

Para la realización de este proyecto se ha realizado un estudio sobre la librería JavaScript, Famo.us en el que se ha aprendido bastante sobre ella además de saber como utilizarla para la realización de aplicaciones.

Estudios anteriores dieron notoriedad a la necesidad de distinguir entre distintos estilos de aprendizaje, ya que de esta manera los alumnos pueden mejorar sus resultados optimizando su tiempo de estudio. Así, se observa que esta aplicación web es una herramienta útil, que el alumno tiene entre otras muchas, para poder evaluar sus estilos propios de aprendizaje. Los puntos a destacar de esta aplicación son los siguientes:

- Sencilla e intuitiva. No es necesario un estudio previo de la aplicación para su utilización.
- Recurso inmediato. Es un recurso que el alumno podrá encontrar como enlace en la página de su universidad para que sea utilizado en cualquier momento.
- Experiencia de usuario agradable. El que una herramienta sea intuitiva, fácil y además contenga algún elemento novedoso, siempre agrada al usuario que la utiliza.

Como conclusión personal he de destacar que en la realización de este proyecto he abordado un nuevo lenguaje de programación para mí, JavaScript. Nunca antes había desarrollado ninguna aplicación con él y es gratificante ver cómo he podido desenvolverme a pesar de esto. Por otra parte quedo satisfecha al realizar una aplicación que puede ayudar a ver que todos los alumnos no son iguales, distinguiendo entre tipos de personas diferentes, que por tanto, aprenden de manera diferente. Cualquier herramienta que pueda ayudar en este sentido, no sobrá en el panorama educativo.

Y aunque la aplicación es plenamente funcional, siempre cabe alguna mejora. Estas podrían ser:

- Guardar los resultados en una base de datos por alumnos y asignaturas. Así los profesores pueden tener disponibles gráficos para saber ante qué tipo de estudiantes están ese año académico.
- Añadir un tablón de anuncios, donde los alumnos puedan intercambiar materiales y consejos dependiendo de su estilo de aprendizaje.
- Un gráfico menos dinámico, donde se muestre menos las opciones de la librería de Famo.us y sean más legibles los resultados de los estudiantes.

ANEXOS

- ANEXO I.

TEST PARA LA DETERMINACIÓN DEL ESTILO DE APRENDIZAJE POR LA PNL.

Inventario sobre estilos de APRENDIZAJE						
<p>INVENTARIO SOBRE ESTILOS DE APRENDIZAJE (De acuerdo al modelo PNL)</p> <p>Nombre:..... Fecha:.....</p> <p>Este inventario es para ayudarle a descubrir su manera preferida de aprender. Cada persona tiene su manera preferida de aprender. Reconocer sus preferencias le ayudará a comprender sus fuerzas en cualquier situación de aprendizaje. Por favor, responda Ud. verdaderamente a cada pregunta. Responda Ud. según lo que hace actualmente, no según lo que piense que sea la respuesta correcta. Use Ud. la escala siguiente para responder a cada pregunta: Ponga un círculo sobre su respuesta.</p> <p>1 = Nunca 2 = Raramente 3 = Ocasionalmente 4 = Usualmente 5 = Siempre</p>						
1	Me ayuda trazar o escribir a mano las palabras cuando tengo que aprenderlas de memoria	1	2	3	4	5
2	Recuerdo mejor un tema al escuchar una conferencia en vez de leer un libro de texto	1	2	3	4	5
3	Prefiero las clases que requieren una prueba sobre lo que se lee en el libro de texto	1	2	3	4	5
4	Me gusta comer bocados y mascar chicle, cuando estudio	1	2	3	4	5
5	Al prestar atención a una conferencia, puedo recordar las ideas principales sin anotarlas	1	2	3	4	5
6	Prefiero las instrucciones escritas sobre las orales	1	2	3	4	5
7	Yo resuelvo bien los rompecabezas y los laberintos	1	2	3	4	5
8	Prefiero las clases que requieran una prueba sobre lo que se presenta durante una conferencia	1	2	3	4	5
9	Me ayuda ver diapositivas y vídeos para comprender un tema	1	2	3	4	5
10	Recuerdo más cuando leo un libro que cuando escucho una conferencia	1	2	3	4	5

11	Por lo general, tengo que escribir los números del teléfono para recordarlos bien	1	2	3	4	5
12	Prefiero recibir las noticias escuchando la radio en vez de leerlas en un periódico	1	2	3	4	5
13	Me gusta tener algo como un bolígrafo o un lápiz en la mano cuando estudio	1	2	3	4	5
14	Necesito copiar los ejemplos de la pizarra del maestro para examinarlos más tarde	1	2	3	4	5
15	Prefiero las instrucciones orales del maestro a aquellas escritas en un examen o en la pizarra	1	2	3	4	5
16	Prefiero que un libro de texto tenga diagramas gráficos y cuadros porque me ayudan mejor a entender el material	1	2	3	4	5
17	Me gusta escuchar música al estudiar una obra, novela, etc.	1	2	3	4	5
18	Tengo que apuntar listas de cosas que quiero hacer para recordarlas	1	2	3	4	5
19	Puedo corregir mi tarea examinándola y encontrando la mayoría de los errores	1	2	3	4	5
20	Prefiero leer el periódico en vez de escuchar las noticias	1	2	3	4	5
21	Puedo recordar los números de teléfono cuando los oigo	1	2	3	4	5
22	Gozo el trabajo que me exige usar la mano o herramientas	1	2	3	4	5
23	Cuando escribo algo, necesito leerlo en voz alta para oír como suena	1	2	3	4	5
24	Puedo recordar mejor las cosas cuando puedo moverme mientras estoy aprendiéndolas, por ejemplo, caminar al estudiar, o participar en una actividad que me permita moverme, etc.	1	2	3	4	5

Fuente: Metts Ralph (1999) "Teorías y ejercicios", Santiago de Chile, pp. 32.
Derechos de propiedad literaria 1987 Ralph Metts S.J.

• ANEXO 2.

TEST DE SOLOMAN-FELDER PARA DETERMINAR EL ESTILO DE APRENDIZAJE.

CUESTIONARIO SOBRE ESTILOS DE APRENDIZAJE

NOMBRE: _____ **GRUPO:** _____

Seleccione de las dos opciones en cada pregunta la que más lo caracterice marcando con una “x” en la línea:

1. Yo entiendo mejor algo después que yo
 (a) pruebo a hacerlo
 (b) lo pienso
2. Yo prefiero que me consideren
 (a) realista
 (b) innovador
3. Cuando pienso lo que hice ayer, lo que obtengo es
 (a) una imagen
 (b) palabras
4. Yo tiendo a
 (a) entender los detalles del aspecto, pero algo indefinida su estructura general
 (b) entender la estructura general, pero algo indefinidos los detalles
5. Cuando estoy aprendiendo algo nuevo, me ayuda
 (a) hablar sobre eso
 (b) pensar sobre eso
6. Si yo fuera profesor, yo preferiría impartir un curso
 (a) que tuviera que ver con hechos y situaciones de la vida real
 (b) que tuviera que ver con ideas y teorías
7. Yo prefiero que me den la información nueva en
 (a) láminas, diagramas, gráficos o mapas
 (b) directrices escritas o información hablada
8. Una vez que entiendo
 (a) todas las partes, entiendo la cuestión completa
 (b) la cuestión completa, me doy cuenta como encajan las partes
9. En un grupo de estudio que trabaja un material difícil, yo por lo regular
 (a) hablo y contribuyo con mis ideas
 (b) me siento y escucho
10. A mí me es más fácil
 (a) aprender los hechos
 (b) aprender los conceptos
11. En un libro con muchas láminas y esquemas, yo por lo regular
 (a) observo detenidamente las láminas y los esquemas
 (b) me concentro en la parte del texto
12. Cuando resuelvo problemas matemáticos
 (a) por lo regular busco la solución paso a paso
 (b) veo regularmente la solución, pero me cuesta trabajo determinar los pasos que me conducen a ella
13. En las clases a las que he asistido
 (a) por lo regular he conocido a muchos de los estudiantes
 (b) casi nunca he conocido muchos de los estudiantes
14. Cuando leo algo, lo que prefiero es
 (a) lo que me muestre nuevos hechos o me indique como hacer algo
 (b) que me de nuevas ideas en las que pensar
15. Me gustan los profesores que
 (a) que ponen muchos diagramas en la pizarra
 (b) que dedican mucho tiempo a explicar
16. Cuando analizo un cuento o una novela

- (a) pienso en los incidentes y trato de ponerlos juntos para lograr entender los diferentes aspectos
 (b) solo me doy cuenta de los aspectos cuando termino de leer y entonces regreso para buscar los incidentes que los demuestran
17. Cuando empiezo a trabajar en un problema de la tarea, por lo regular tiendo a
- (a) comenzar a trabajar de inmediato en la solución
 (b) tratar de entender primero completamente el problema
18. Yo prefiero la idea de
- (a) lo cierto
 (b) la teoría
19. Yo recuerdo mejor
- (a) lo que veo
 (b) lo que oigo
20. Es mucho más importante para mí que el profesor
- (a) exponga el material en una secuencia de pasos bien claros
 (b) me de una idea general de lo que va a tratar y lo vaya relacionando con otras cuestiones
21. Yo prefiero estudiar
- (a) en un grupo de estudio
 (b) solo
22. Me gustaría más que me consideraran
- (a) cuidadoso con los detalles de mi trabajo
 (b) creativo por la forma en que hago mi trabajo
23. Cuando me dan la forma de llegar a un lugar prefiero que
- (a) me lo den en un mapa
 (b) en instrucciones escritas
24. Yo aprendo
- (a) a un paso aceptablemente adecuado. Si estudio duro, lo entenderé
 (b) por “paradas y arranques”. Me siento completamente confundido y de pronto todo “funciona”
25. Yo preferiría primero
- (a) probar a hacer las cosas
 (b) pensar en como voy a hacerlas
26. Cuando estoy leyendo como entretenimiento, me gusta que los autores
- (a) expresen claramente lo que quieren hacer entender
 (b) digan las cosas en forma creativa e interesante
27. Cuando veo en la clase un diagrama o un croquis, por lo regular recuerdo
- (a) una imagen
 (b) lo que dijo el profesor
28. Cuando analizo un monto de información, por lo regular
- (a) me concentro en los detalles y no veo la idea completa
 (b) trato de entender la idea grande antes de entrar en los detalles
29. Yo recuerdo más fácil
- (a) algo que he hecho
 (b) algo en lo que he pensado mucho
30. Cuando tengo que ejecutar algo, yo prefiero
- (a) dominar una forma de hacerlo
 (b) encontrar nuevas formas de hacerlo
31. Cuando alguien me está enseñando un conjunto de datos, yo prefiero
- (a) los gráficos o esquemas
 (b) un escrito que resuma los resultados
32. Cuando escribo un trabajo, por lo regular
- (a) hago (pienso o escribo) el comienzo del mismo y después sigo
 (b) hago (pienso o escribo) las diferentes partes del trabajo y después lo ordeno
33. Cuando tengo que trabajar en grupo en un proyecto, lo primero que deseo es
- (a) que se haga una “tormenta de ideas” en la que todos contribuyen con sus ideas
 (b) que cada uno individualmente haga su “tormenta de ideas” y entonces que se reúna el grupo para comparar las ideas
34. Considero que para una persona es mejor que lo consideren
- (a) sensitivo
 (b) imaginativo

35. Cuando conozco personas en una fiesta, por lo regular lo que recuerdo es
__ (a) como eran ellos
__ (b) que decían sobre ellos mismos
36. Cuando voy a aprender una cuestión nueva, yo prefiero
__ (a) concentrarme en esa cuestión, aprendiéndola lo más posible
__ (b) tratar de relacionarla a ella con otras cuestiones
37. Me parece que me consideran
__ (a) expresivo
__ (b) reservado
38. Yo prefiero los cursos que hacen énfasis en
__ (a) contenidos concretos (hechos, datos)
__ (b) contenidos abstractos (conceptos, teorías)
39. Como entretenimiento prefiero
__ (a) mirar televisión
__ (b) leer un libro
40. Algunos profesores empiezan sus clases con un panorama de lo que van a tratar. Ese panorama para mí es
__ (a) algo que me ayuda un poco
__ (b) de mucha ayuda
41. La idea de hacer la tarea en colectivo, en grupo, para obtener una nota para el grupo completo
__ (a) me gusta
__ (b) no me gusta
42. Cuando hago cálculos muy largos
__ (a) tiendo a repetir los pasos que he hecho y comprobarlos con mucho cuidado
__ (b) lo encuentro cansón y me tengo que obligar a mí mismo a hacerlos
43. Tiendo a describir los lugares donde he estado
__ (a) fácilmente y con bastante exactitud
__ (b) con gran dificultad y sin muchos detalles
44. Cuando soluciono problemas en grupo, me gustaría más
__ (a) pensar en los pasos necesarios para su solución
__ (b) pensar en las posibles consecuencias o sus diferentes aplicaciones en un gran cúmulo de áreas

REFERENCIAS BIBLIOGRÁFICAS

- [1] Kolb, D.A. *Experiential Learning: Experience as a source of learning and development*. Englewood Cliffs, NJ: Prentice Hall.
- [2] Soloman Barbara A., Ferder, Richard M. of Learning Styles Questionnaire, North Carolina State University.
- [3] <http://www.famo.us/>
- [4] <http://www.ibm.com/developerworks/web/library/wa-famous/index.html>
- [5] <https://github.com/Famous>
- [6] <https://github.com/Famous/generator-famous>
- [7] <http://gruntjs.com/getting-started>
- [8] http://librosweb.es/javascript/capitulo_7/utilidades_basicas_para_formularios.html
- [9] http://www.w3schools.com/jsref/dom_obj_radio.asp
- [10] <https://github.com/Famous/guides/blob/master/dev/2014-04-09-render-tree.md>
- [11] <http://famousco.de/>
- [12] http://www.ogre3d.org/tikiwiki/Quaternion+and+Rotation+Primer#Quaternion_Rotations
- [13] <http://codepen.io/befamous/tag/physics/>