



## Chichen Itzá: Mundo Virtual

### Descripción

A lo largo del presente reporte será dilucidado el concepto y el diseño planteado para construir un sistema que ofrezca recorridos o visitas virtuales a través de una reconstitución digital de la zona arqueológica Chichen Itzá. Este sistema pretende ser capaz de permitir que múltiples usuarios de cualquier parte del mundo se conecten y sean partícipes de esta experiencia. El diseño subyacente será realizado con base en una arquitectura a componentes y será estructurado en forma de Cliente-Servidor.

### Objetivos

El presente reporte mantiene el fin de detallar el diseño propuesto para la realización de este sistema, desde la estructuración inherente a los componentes mismos, la manera de distribución de los cambios de información local a ser transmitida globalmente a todos los usuarios de la red como a la relación sugerida para los objetos que la conformarán.

El sistema en sí mismo pretende proveer a sus usuarios la capacidad de conectarse al sitio, en principio, visualizar una versión virtual de la zona arqueológica, navegar con libertad en él mediante la utilización de un avatar en línea que efectuará todas las acciones que el usuario indique y con el cual ser capaz de interactuar con otros usuarios o con objetos inertes. Cada usuario será provisto con un rol que determinará sus permisos de acceso a ciertas áreas, niveles de interacción permitidos y acciones que puede llevar a cabo. Todos los usuarios serán capaces de visualizar, en tiempo real, el estado y las acciones actuales que desempeñan los otros usuarios del sistema.

### Especificaciones técnicas

La construcción de este proyecto está planeada a ser realizada sobre una arquitectura basada en componentes. Las ventajas y objetivos principales de un sistema basado en componentes son descritas por Krakoviak (2017) como “la meta (...) es mejorar la reusabilidad al cambiar el trabajo de desarrollo a integración, proveer soporte genérico para servicios comunes con el fin de permitir a los desarrolladores concentrarse en las necesidades de la aplicación y facilitar el mantenimiento, evolución y administración”.

La realización de este proyecto está planeada a ser consumada utilizando la tecnología Enterprise JavaBeans que, de acuerdo al *IBM Knowledge Center* “Enterprise JavaBeans es un modelo de componente estándar del lado del servidor para aplicaciones empresariales distribuidas”, por supuesto, la programación

del proyecto mismo deberá ser realizada con el lenguaje de programación de alto nivel orientado a objetos Java. La noción de cuántos clientes al mismo tiempo pueden ser administrados, sin embargo, requiere de cálculos inherentes a la infraestructura de software y de hardware con que este proyecto decida ser implementado de las cuales no se tiene información en este punto de la investigación y que exceden el alcance pretendido para este diseño.

### Análisis de viabilidad

Los requerimientos para este proyecto no representan un reto técnico para la tecnología con que se cuenta hoy día, la programación de una arquitectura basada en componentes es simple y directa debido a la naturaleza de esta, ya que al no tener que preocuparse por la implementación de elementos no funcionales como concurrencia, seguridad o transacciones, los desarrolladores pueden enfocarse en la programación de los aspectos relacionados puramente con la aplicación.

El aspecto monetario no es fácil de considerar en este punto, pues se desconoce a la fecha la organización que llevará a cabo la implementación de este bosquejo, no obstante, la inversión principal que debe realizarse para este efecto se reduce a la implementación de los servidores, la aplicación misma y el mantenimiento de la misma que, incluso si resultan en una inversión importante, las utilidades generadas por el proyecto deberían de ser suficientes para considerar redituable la inversión. El proyecto es viable, sin embargo, sin duda alguna requerirá de una inversión inicial importante que es relevante considerar.

### Metodología para el desarrollo del proyecto

En primer lugar, la estructura del proyecto está conformada en su totalidad por componentes dispuestos como la Figura 1 indica. El servidor del proyecto es representado por un componente compuesto que contiene componentes elementales conectados entre sí donde cada uno ofrece y requiere servicios específicos enunciados posteriormente. El cliente con el que el usuario interactuará también es representado como un componente elemental, las razones son detalladas a continuación.

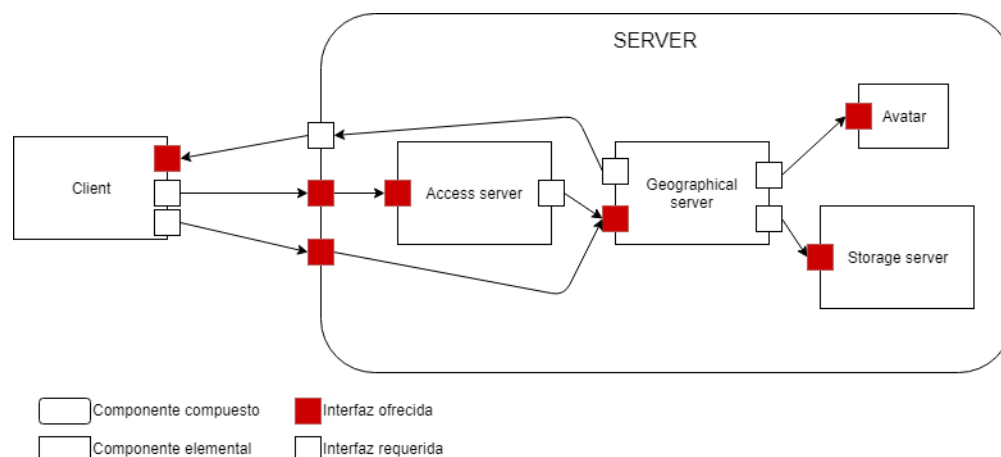


Figura 1. Diagrama de estructura de componentes

Cada uno de los componentes propuestos posee la siguiente función:

- Servidor de acceso: es el servidor al que cualquier cliente nuevo debe conectarse, el usuario enviará sus credenciales de autenticación y este servidor se encargará de verificar su validez. Si éstas son aceptadas el servidor de acceso solicitará al cliente información sobre su localización, con esta información el servidor redireccionará las instrucciones sucesivas hacia el servidor geográfico que le corresponda a ese cliente.
- Servidor geográfico: cada uno de estos servidores localizados por conveniencia en diferentes zonas alrededor del mundo ofrecerán los servicios que los clientes requieran para interactuar con el sitio virtual. Es responsabilidad de cada uno de estos servidores administrar a los clientes conectados junto con sus avatares respectivos y manejar las solicitudes de los primeros para que los segundos las ejecuten.
- Servidor de almacenamiento: en este servidor(es) se almacena la información necesaria para reconstituir el sitio virtual de Chichen Itzá, sus servicios son utilizados por los servidores geográficos quienes redirigen la información requerida por un cliente, por ejemplo, un área en específico para ser renderizada en la vista de su cliente.
- Avatar: representará las acciones realizadas por un usuario sobre el sitio virtual, sus representaciones serán visibles para todos los usuarios de la red permitiendo de esta manera la intercomunicación entre usuarios, objetos y áreas.
- Cliente: es el componente con el que el usuario interactuará directamente y que mantendrá comunicación con los componentes servidor. Es representado como un componente porque también define una interfaz que es utilizada por el servidor geográfico por nociones de *callback*, es decir, para que el servidor pueda enviarle notificaciones cuando alguna novedad ha sucedido en el servidor, potencialmente ocasionada por las acciones de otros usuarios.

A continuación, son definidas las interfaces provistas y requeridas por cada uno de los componentes. Las funciones nombradas aquí son perfectamente susceptibles de cambiar tanto sus parámetros como sus valores de retorno en tiempo de implementación y pueden ser agregadas funciones según sea requerido, pero la idea no debe ser modificada.

*Servidor de acceso:*

Interfaz provista

```
boolean connect(username, password)
boolean disconnect(username)
void geographicalInformation(coordinates)
```

Interfaz requerida

Este servidor depende de la interfaz provista por el Servidor Geográfico pues necesita enviarle información sobre que un nuevo cliente se ha conectado a ese servidor

*Servidor geográfico:*

Interfaz provista

```
void connect(username)
void disconnect(username)
void move(newPosition)
void talk(username, message)
void take(decoration)
void enter(area)
...
```

Interfaces requeridas

Este servidor necesita de la interfaz provista por cada cliente para enviarle notificaciones cuando algún nuevo suceso de su interés ha acontecido, principalmente aquellos realizados por otros usuarios. También necesita de la interfaz provista por Avatar para indicar las acciones que este debe desempeñar sobre cierto objeto u otro avatar o simplemente acciones de movimiento. Asimismo, requiere de la interfaz provista por el servidor de almacenamiento para solicitar la información necesaria para reconstituir cierta parte del sitio virtual

*Servidor de almacenamiento:*

Interfaz provista:

```
Area getArea(areaID)
Decoration getDecoration(decorationID)
```

*Avatar:*

Interfaz provista:

```
void move(newPosition)
void talk(username, message)
void take(decoration)
void wave()
...
```

*Cliente:*

Interfaz provista:

```
void update(news)
```

Interfaces requeridas:

Este componente requiere de las interfaces provistas por el Servidor de Acceso para verificar las credenciales introducidas por el usuario y de la interfaz provista por el Servidor Geográfico para efectuar todas las actividades que el usuario puede realizar en el sitio.

Para esclarecer la interacción que estos componentes comparten entre sí, se han realizado los siguientes diagramas de secuencia. La Figura 2 revela la línea de invocaciones sugerida para llevar a cabo cuando un cliente se conecta a la red; este proceso sigue de la mano todas las interacciones efectuadas desde el momento en que el usuario ingresa sus credenciales hasta que su avatar es creado de lado del servidor. La

figura 3 por otro lado, ilustra el cómo se actualizará la visión del estado de cada uno de los avatares cuando cualquiera de estos sea modificado.

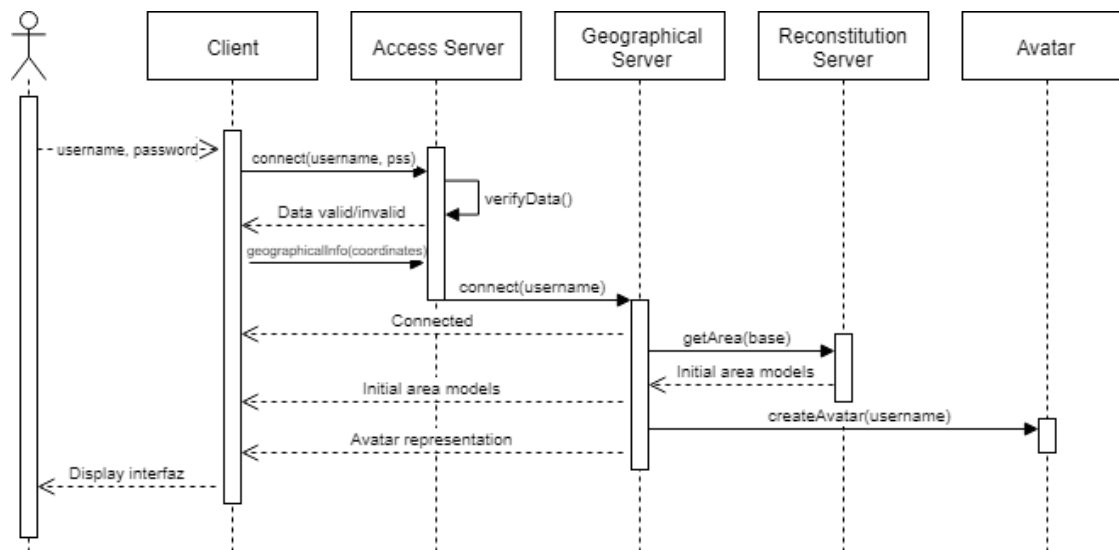


Figura 2. Proceso de conexión de un usuario a la red y creación de su avatar

El usuario introduce sus credenciales en su aplicación Cliente, la cual utiliza estos datos para tratar de conectarse en el Servidor de Acceso, donde los datos son verificados, si estos proceden, el Cliente envía información sobre la localización actual del usuario, mediante la cual el Servidor de Acceso conecta al Cliente con el Servidor Geográfico más adecuado. Una vez se le da aviso al Servidor Geográfico de que un nuevo cliente va a conectarse a él, éste solicita al Servidor de Almacenamiento que le provea la información necesaria para reconstituir la base, donde todos los clientes inician su recorrido, la cual es enviada al cliente junto con el Avatar que el Servidor geográfico crea para este usuario en específico. Una vez el Cliente obtiene la información sobre su área y avatar lo renderiza para que el usuario comience sus interacciones.

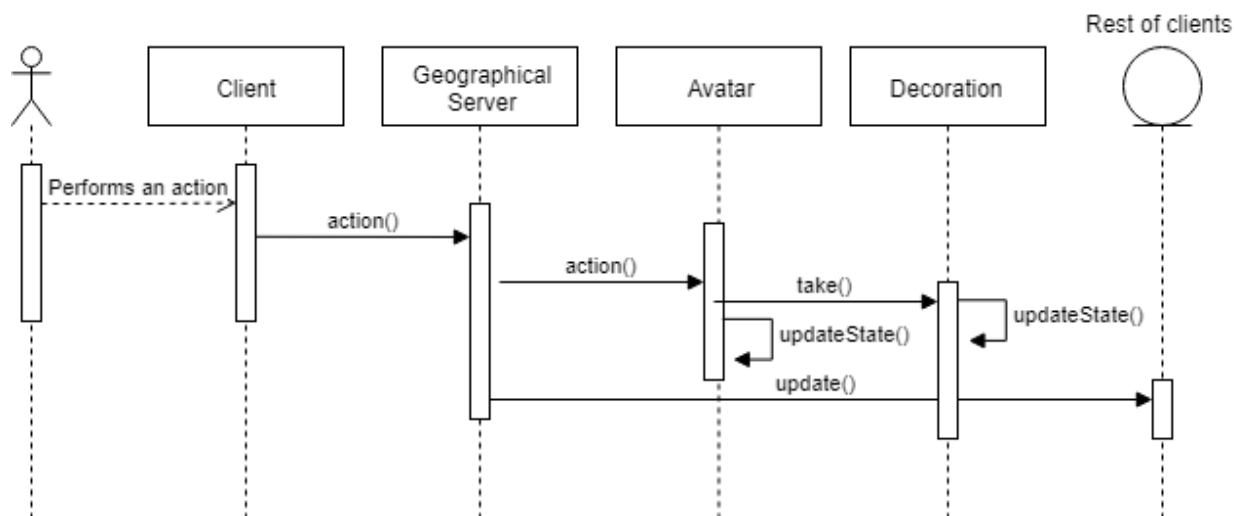


Figura 3. Proceso mediante el cual las vistas de los clientes son actualizadas

Cada vez que un usuario realiza una acción sobre su Cliente, éste se lo notifica al Servidor Geográfico al que está conectado, quien utiliza la interfaz del Avatar para efectuar esa acción, como ejemplo si el usuario trata de tomar una de las decoraciones disponibles, el Avatar del usuario que realizó la acción cambia su estado a “Interactuando con decoración” y la decoración en cuestión cambia su estado a “Tomada”; el Servidor geográfico entonces da aviso a todos los clientes conectados actualmente de que esta nueva ha acontecido para actualizar sus visiones —está incluido el cliente que realizó la acción—, de tal suerte que todos los usuarios conectados pueden visualizar que uno de los avatares está observando una decoración y, por lo tanto, esa decoración en particular se vuelve inaccesible hasta que esa acción termine.

La relación subyacente que existe entre este tipo de objetos con los que un Avatar puede iniciar una interacción es ilustrada en la figura 4.

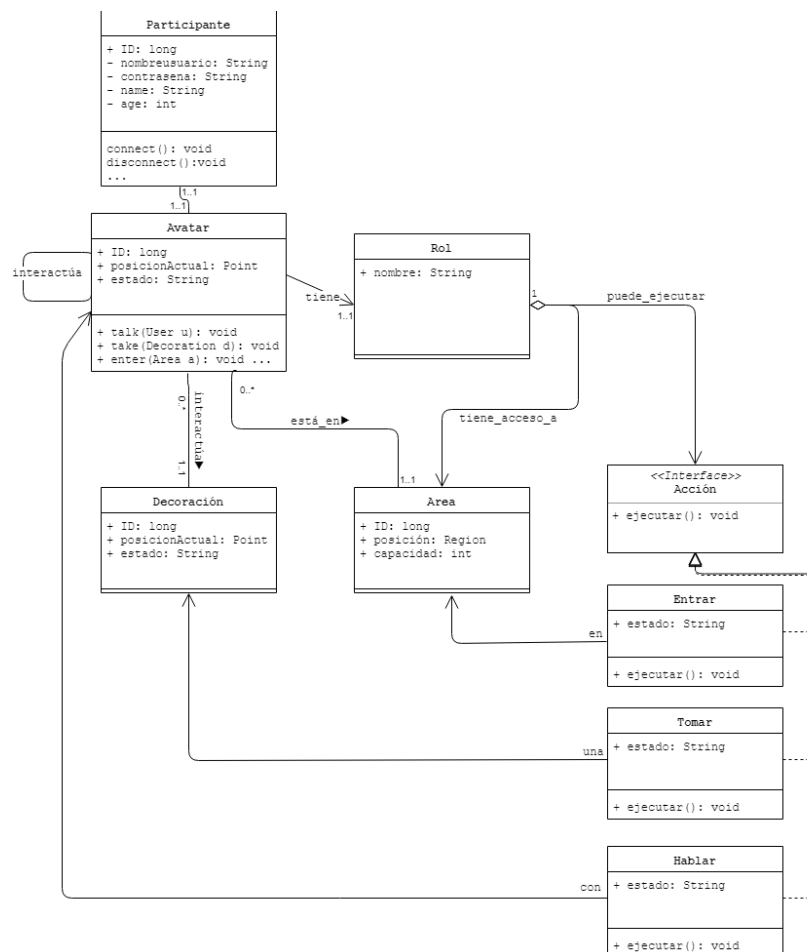


Figura 4. Interacción de un avatar con ciertos objetos

Cada Avatar tiene un rol definido, este rol es el que define las áreas a las que el Avatar tiene acceso y las acciones que puede ejecutar. Las acciones posibles están representadas mediante la aplicación del patrón de diseño de nombre *Command*, que permite encapsular instrucciones en

objetos de tal manera que puedan ser parametrizadas, distribuidas o transferidas, de esta manera, cuando el servidor se vea en la necesidad de realizar *callback* –como resultado de la aplicación de cualquiera de estas acciones–, estas acciones encapsuladas pueden ser redirigidas a todos los clientes para su correspondiente actualización.

## **Conclusiones**

Este proyecto representa una inversión redituable destinada para un sistema que no sólo fomentará el interés arqueológico global al permitir a cualquier persona en cualquier parte del mundo ser testigo de la belleza histórica de esta zona, sino que potencialmente podría generar el interés suficiente en los usuarios de este sistema para visitar esta nueva maravilla del mundo en la península de Yucatán, potenciando de esta manera el turismo nacional a niveles globales. Además, podría llegar a fungir como un prototipo de muchos proyectos similares a ser desarrollados alrededor del mundo sobre más sitios turísticos que muchas personas por cualquier motivo no son capaces de visitar y gracias a esto podrían al menos ver saldada su curiosidad e interés, lo cual puede llegar a influir como un gran iniciador de una sociedad global armónica más respetuosa por sitios de tan alto interés histórico o artístico.

## **Referencias y bibliografía**

- [1] Krakowiak, S. (2007). Middleware Architecture with Patterns and Frameworks [Ebook]. Creative Commons.
- [2] IBM Knowledge Center. (2019). Retrieved from [https://www.ibm.com/support/knowledgecenter/es/SS4JE2\\_7.5.5/org.eclipse.jst.ejb.doc.user/topics/ejbarc.html](https://www.ibm.com/support/knowledgecenter/es/SS4JE2_7.5.5/org.eclipse.jst.ejb.doc.user/topics/ejbarc.html)