**Arquitectura de Software**

**Experimento 1**



**Realizado por:**

|  |  |  |
| --- | --- | --- |
| **Persona** | **Rol** | **CódigoUniandes** |
| Carlos Ernesto González Vargas | Ingeniero de Requerimientos | 200819123 |
| Sandra Milena Gómez Ríos | Ingeniero de Producción | 201110951 |
| Andrés Mauricio Erazo Benavides | Ingeniero de Soporte | 201110949 |
| David Pérez Chibuque | Ingeniero de Calidad | 201117818 |
| Willian Alejandro Idrobo Luna | Arquitecto Jefe | 201110544 |
| Erik Fernando Arcos Franco | Ingeniero de Administración | 201110856 |

**Contenido**

**Pag.**

[**1.** **Definición del Sistema** 3](#_Toc296874546)

[**1.1.** **Identificación de patrones** 3](#_Toc296874547)

[**1.2.** **Punto de vista funcional** 4](#_Toc296874548)

[**1.2.1.** **Diagrama de descomposición** 4](#_Toc296874549)

[**1.2.2.** **Diagrama de despliegue** 5](#_Toc296874550)

[**2.** **Experimento A – Latencia** 6](#_Toc296874551)

[**2.1.** **A1 – Registro de Alarma** 6](#_Toc296874552)

[**2.2.** **A2 – Consulta de Información Consolidada** 8](#_Toc296874553)

[**3.** **Experimento B – Escalabilidad** 11](#_Toc296874554)

[**3.1.** **B1 – Escalabilidad del Sistema** 11](#_Toc296874555)

**Arquitectura de Software**

**Experimento 1**

1. **Definición del Sistema**

La empresa **Alpes Smart Home (ASH)** desea entrar a participar en el mercado de las casas inteligentes dado el considerable avance de la tecnología en dispositivos móviles, sensores inalámbricos y sistemas embebidos que ha habido durante los últimos años. Por esta razón la empresa **ASH** plantea el proyecto ***Casa Inteligente***, el cual surge para dar respuesta al diseño de arquitectura de solución de un sistema que permita la administración de viviendas inteligentes.

El sistema se debe cumplir con las necesidades iniciales del sistema, descritas a continuación:

* Analizar la información enviada por los sensores de humo, ventanas y puertas de las oficinas y viviendas monitoreadas.

Una vez se detecta humo en una posición específica de la vivienda u oficina, se procede a notificar a los servicios de control de emergencias. Si se detecta que la vivienda está habitada, se informa a los bomberos y servicios de ambulancias, el número estimado de personas que están en ese momento en el lugar, así como las posibles ubicaciones.

Si se trata de un sensor de ventana o puerta abierta sin autorización, se debe notificar a la policía sobre el evento, enviando como información adicional, la localización de la ventana dentro del inmueble y el número estimado de personas que están en la vivienda o los sectores de la vivienda en los que se ha detectado movimiento.

* Generar un reporte consolidad mostrando todos los eventos registrados (ordenadas por tipo de evento) de forma cronológica (hora, día, mes, año).
* Por cada inmueble u oficina se tienen en promedio 50 tags RFID enviando información cada segundo a cuatro antenas ubicadas en diferentes posiciones de la casa u oficina. Estas antenas a su vez consolidan la información y la envían a la central.

Cuando se detecta que un tag RFID ha violado una de las reglas definidas para su comportamiento, se debe generar una alarma indicando a todas las porterías y a los responsables de la vivienda u oficina lo ocurrido.

* 1. **Identificación de patrones**

Para el desarrollo de este sistema se tuvieron en cuenta los siguientes patrones:

*Acceso a recursos – temprana*

Se aplica en la parte de obtención de la conexión a la base de datos para la generación de reportes.

*Acceso a recursos – caché*

Es aplicada para el manejo de y la información de los usuarios y las reglas que son configuradas para cada casa u oficina.

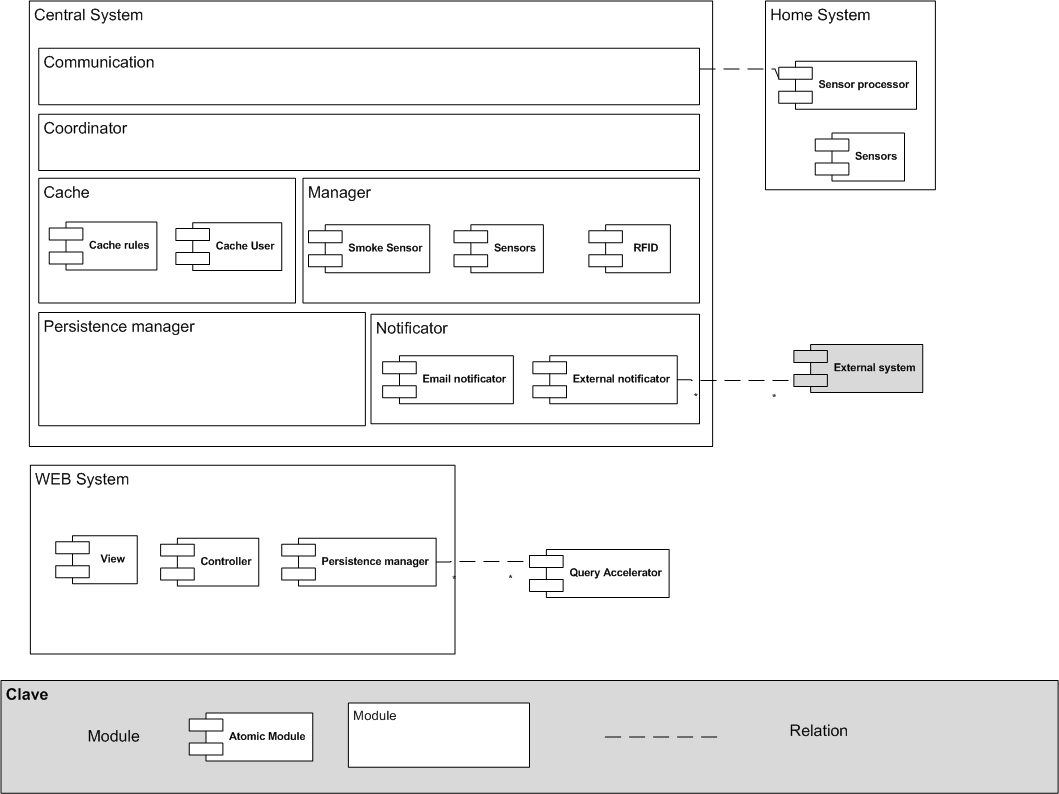
*Manejo de eventos – respuesta reactiva a eventos*

Al llegar un evento al sistema se delega la responsabilidad de manejar el evento a un hilo que se encargue de procesarlo de acuerdo a las reglas configuradas.

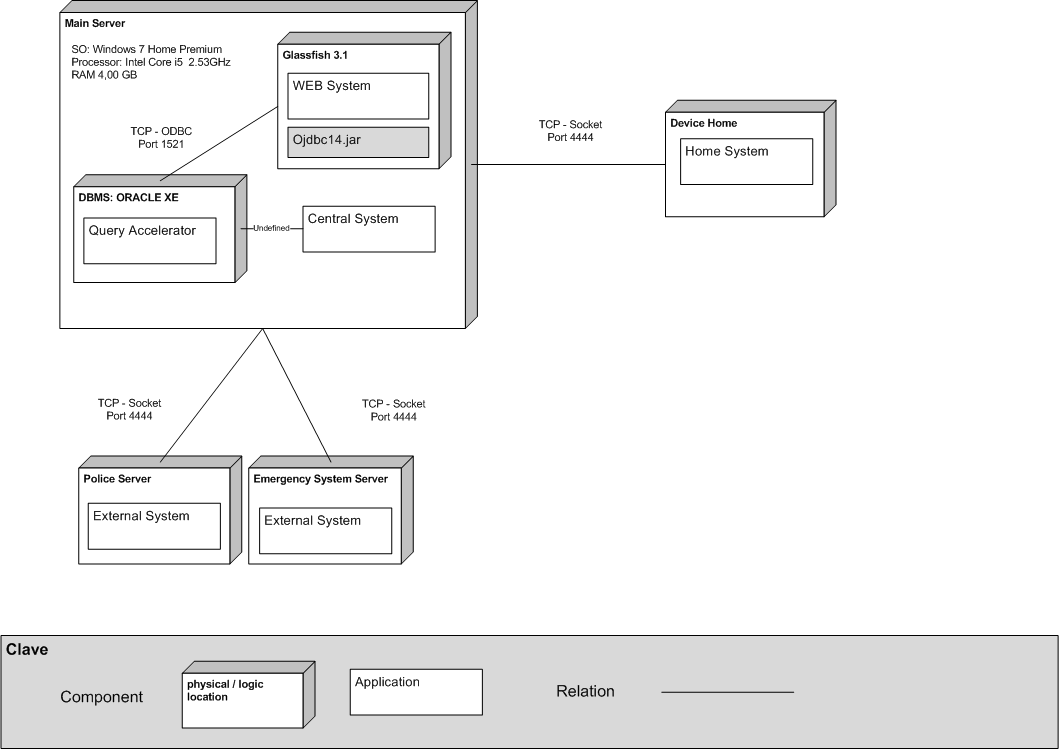
*Concurrencia y sincronización – coordinador*

Existe un componente receptor de todos los eventos que se encarga de crear una instancia de un manejador que procesa el evento de acuerdo a su tipo.

* 1. **Punto de vista funcional**
     1. **Diagrama de descomposición**



* + 1. **Diagrama de despliegue**



1. **Experimento A – Latencia**

El objetivo de este experimento es determinar si las decisiones de diseño satisfacen los escenarios de calidad relacionados con la latencia de la aplicación.

* 1. **A1 – Registro de Alarma**

Tabla 1. Descripción del Experimento A1

|  |  |
| --- | --- |
| **Descripción del Experimento** | |
| ***Título*:** Registro de Alarma | ***ID*:** A1 |
| ***Descripción*:** Este experimento busca verificar la latencia del sistema | ***Responsable*:** Ingenium |
| ***Propósito:***  ( ) Reparación, actualizar, clarificar  ( X ) Obtener Información técnica  ( ) Obtener información de negocio  ( ) Otros: | |
| ***Propósito:***  Verificar las decisiones de diseño asociadas a la latencia del sistema. | |
| ***Descripción del experimento:***  El sistema debe estar en capacidad de analizar la información enviada por los sensores de humo, ventanas y puertas de las oficinas y viviendas monitoreadas.  Se debe tener en cuenta lo siguiente:   * Una vez se detecta humo en una posición específica de la vivienda u oficina, se procede a notificar a los servicios de control de emergencias. * Si se detecta que la vivienda está habitada, se informa a los bomberos y servicios de ambulancias, el número estimado de personas que están en ese momento en el lugar, así como las posibles ubicaciones. * Si se trata de un sensor de ventana o puerta abierta sin autorización, se debe notificar a la policía sobre el evento, enviando como información adicional, la localización de la ventana dentro del inmueble y el número estimado de personas que están en la vivienda o los sectores de la vivienda en los que se ha detectado movimiento. | |
| ***Artefactos Creados:***  ASHCentralSystem -> Es el proyecto principal donde se encuentra toda la lógica de negocio para procesar los eventos recibidos por cada casa u oficina.  ASHCommunication -> Librería que administra la comunicación entre el sistema central y el concentrador ubicado en cada casa u oficina.  ASHHomeModule -> Es el proyecto con la lógica para la simulación de los eventos generados por la casa u oficina.  ASHExternalSistem -> Sistema que emula la policía y los sistemas de emergencia. | |
| ***Criterio de terminación:***  El tiempo para la notificación de la información, una vez recibida la alarma, no debe ser superior a 1 segundo. | |
| ***Recursos Requeridos:***  Recurso humano -> Se necesita el trabajo de 4 personas con conocimientos en java, sockets, estilos y tácticas de arquitectura.  Recursos técnicos -> NetBeans IDE 7.0, Microsoft Visio, JUnit, JUnitPerf | |
| ***Duración estimada:***  32 horas | |

Tabla 2. Resultados del Experimento A1

|  |
| --- |
| **Resultados del Experimento** |
| ***Resumen de los resultados:*** |
| ***Duración Real:***  50 horas |
| ***Recursos Reales:***  Usted debe hacer un estimativo de la escalabilidad de su sistema, mostrando una curva en la que se compare el tiempo de respuesta contra número de usuarios concurrentes, a los que su sistema está en capacidad de servir simultáneamente. |
| ***Recomendaciones:*** |

* 1. **A2 – Consulta de Información Consolidada**

Tabla 3. Descripción del Experimento A2-1

|  |  |
| --- | --- |
| **Descripción del Experimento** | |
| ***Título*:** Latencia | ***ID*:** A2-1 |
| ***Descripción*:** Este experimento busca verificar el desempeño del sistema en el generación de reporte consolidado de eventos generados. | ***Responsable*:** Ingenium |
| ***Propósito:***  ( ) Reparación, actualizar, clarificar  ( X ) Obtener Información técnica  ( ) Obtener información de negocio  ( ) Otros: | |
| ***Propósito:***  Verificar las decisiones de diseño asociada a la latencia del sistema. | |
| ***Descripción del experimento:***  Verificar el tiempo necesario para procesar la solicitud de un reporte consolidado de los 400 eventos registrados (ordenadas por tipo de evento) de forma cronológica (hora, día, mes, año). | |
| ***Artefactos Creados:***  ASHWebModule -> Sistema Web que permite el procesamiento para generar el reporte consolidado de eventos presentados ante el sistema central. Posee una conexión con la base de datos de donde obtiene la información de los eventos. | |
| ***Criterio de terminación:***  Este reporte no debe tomar más de 2 segundos en ser presentado al usuario. | |
| ***Recursos Requeridos:***  Recurso humano -> Se necesita el trabajo de 2 personas con conocimientos en java, estilos y tácticas de arquitectura, bases de datos y diseño Web.  Recursos técnicos -> Base de datos Oracle XE, NetBeans IDE 7.0, Microsoft Visio, Navegador, editor SQL Depeloper | |
| ***Duración estimada:***  16 horas | |

Tabla 4. Resultados del Experimento A2-1

|  |
| --- |
| **Resultados del Experimento** |
| ***Resumen de los resultados:***  Se realizaron pruebas por medio de la Suite de pruebas de Microsoft Visual Studio, generando el reporte con 400 eventos registrados en base de datos, en un tiempo de 0,442 segundos. |
| ***Duración Real:***  19 horas |
| ***Recursos Reales:***  Recurso humano -> Se necesita el trabajo de 3 personas con conocimientos en java, estilos y tácticas de arquitectura, bases de datos y diseño Web.  Recursos técnicos -> Base de datos Oracle XE, NetBeans IDE 7.0, Microsoft Visio, Navegador, editor SQL Depeloper |
| ***Recomendaciones:***  Verificar el experimento A2-2 |

Tabla 3. Descripción del Experimento A2-2

|  |  |
| --- | --- |
| **Descripción del Experimento** | |
| ***Título*:** Latencia | ***ID*:** A2-2 |
| ***Descripción*:** Este experimento busca verificar el desempeño del sistema en el generación de reporte consolidado de eventos generados. | ***Responsable*:** Ingenium |
| ***Propósito:***  ( ) Reparación, actualizar, clarificar  ( X ) Obtener Información técnica  ( ) Obtener información de negocio  ( ) Otros: | |
| ***Propósito:***  Verificar las decisiones de diseño asociada a la latencia del sistema. | |
| ***Descripción del experimento:***  Verificar el tiempo necesario para procesar la solicitud de un reporte consolidado de los 4800 eventos registrados (ordenadas por tipo de evento) de forma cronológica (hora, día, mes, año). | |
| ***Artefactos Creados:***  ASHWebModule -> Sistema Web que permite el procesamiento para generar el reporte consolidado de eventos presentados ante el sistema central. Posee una conexión con la base de datos de donde obtiene la información de los eventos. | |
| ***Criterio de terminación:***  Este reporte no debe tomar más de 2 segundos en ser presentado al usuario. | |
| ***Recursos Requeridos:***  Recurso humano -> Se necesita el trabajo de 2 personas con conocimientos en java, estilos y tácticas de arquitectura, bases de datos y diseño Web.  Recursos técnicos -> Base de datos Oracle XE, NetBeans IDE 7.0, Microsoft Visio, Navegador, editor SQL Depeloper | |
| ***Duración estimada:***  2 horas | |

Tabla 5. Resultados del Experimento A2-2

|  |
| --- |
| **Resultados del Experimento** |
| ***Resumen de los resultados:***  Se realizaron pruebas por medio de la Suite de pruebas de Microsoft Visual Studio, generando el reporte con 4800 eventos registrados en base de datos, en un tiempo de 4,165 segundos. |
| ***Duración Real:***  3 horas |
| ***Recursos Reales:***  Recurso humano -> Se necesita el trabajo de 3 personas con conocimientos en java, estilos y tácticas de arquitectura, bases de datos y diseño Web.  Recursos técnicos -> Base de datos Oracle XE, NetBeans IDE 7.0, Microsoft Visio, Navegador, editor SQL Depeloper |
| ***Recomendaciones:***  Para mejorar el rendimiento del tiempo en la generación del reporte es recomendable realizar una paginación en presentación y realizar paginación por base de datos, es decir, utilizar el patrón Acceso a recursos temprana (Lazy Acquisition). Esta recomendación puede tomar valor en el crecimiento de registros en base de datos de los eventos generados. |

1. **Experimento B – Escalabilidad**

El objetivo de este experimento es determinar la escalabilidad del sistema. Para ello usted deberá utilizar una herramienta para análisis de desempeño y escalabilidad como JUnitPerf, para simular la ocurrencia de múltiples eventos concurrentemente.

* 1. **B1 – Escalabilidad del Sistema**

Tabla 6. Descripción del Experimento B1

|  |  |
| --- | --- |
| **Descripción del Experimento** | |
| ***Título*:** Escalabilidad | ***ID*:** B1 |
| ***Descripción*:** Este experimento busca verificar la escalabilidad del sistema | ***Responsable*:** Ingenium |
| ***Propósito:***  ( ) Reparación, actualizar, clarificar  ( X ) Obtener Información técnica  ( ) Obtener información de negocio  ( ) Otros: | |
| ***Propósito:***  Verificar las decisiones de diseño asociadas a la escalabilidad del sistema. | |
| ***Descripción del experimento:***  En este experimento debe simular la recepción de información proveniente de los tags RFID que se encuentran en cada uno de los inmuebles. Suponga que por cada inmueble u oficina se tienen en promedio 50 tags RFID enviando información cada segundo a cuatro antenas ubicadas en diferentes posiciones de la casa u oficina. Estas antenas a su vez consolidan la información y la envían a la central. | |
| ***Artefactos Creados:***  ASHCentralSystem -> Es el proyecto principal donde se encuentra toda la lógica de negocio para procesar los eventos recibidos por cada casa u oficina.  ASHCommunication -> Librería que administra la comunicación entre el sistema central y el concentrador ubicado en cada casa u oficina.  ASHHomeModule -> Es el proyecto con la lógica para la simulación de los eventos generados por la casa u oficina.  ASHExternalSistem -> Sistema que emula la policía y los sistemas de emergencia. | |
| ***Criterio de terminación:***  Cuando se detecta que un tag RFID ha violado una de las reglas definidas para su comportamiento, se debe generar una alarma indicando a todas las porterías y a los responsables de la vivienda u oficina lo ocurrido. Esto debe suceder en un tiempo inferior a un segundo. | |
| ***Recursos Requeridos:***  Recurso humano -> Se necesita el trabajo de 3 personas con conocimientos en java, sockets, estilos y tácticas de arquitectura.  Recursos técnicos -> NetBeans IDE 7.0, Microsoft Visio, JUnit, JUnitPerf | |
| ***Duración estimada:***  8 horas | |

Tabla 7. Resultados del Experimento B1

|  |
| --- |
| **Resultados del Experimento** |
| ***Resumen de los resultados:***  Usted debe hacer un estimativo de la escalabilidad de su sistema, mostrando una curva en la que se compare el tiempo de respuesta contra número de usuarios concurrentes, a los que su sistema está en capacidad de servir simultáneamente. |
| ***Duración Real:***  12 horas |
| ***Recursos Reales:***  Recurso humano -> Se necesita el trabajo de 2 personas con conocimientos en java, sockets, estilos y tácticas de arquitectura.  Recursos técnicos -> NetBeans IDE 7.0, Microsoft Visio, JUnit, JUnitPerf |
| ***Recomendaciones:*** |