Patrones significativos para la arquitectura:

Patrón Arquitectura en Capas

|  |  |
| --- | --- |
| Capa de presentación Móvil | Capa de presentación Gadgets |
| Capa de Servicios Web | |
| Capa de Lógica de Negocios | |
| Capa de Administración de Base de Datos | |

Consiste en una pila de capas, donde cada capa actúa como máquina virtual de la capa de arriba. En un estilo estratificado simple, las capas solo pueden usar la capa que está directamente debajo. La restricción significa que las capas subsiguientes más abajo están ocultas. Dado que las capas dependen solo de la inmediata inferior las capas subsecuentes pueden ser intercambiadas o emuladas, las capas superiores dan más oportunidades de sustitución frente a un posible problema de performance.

Este patrón se utiliza para organizar la implementación del sistema en capas de servicios auto-contenidos, para lograr un sistema de fácil manutención, de bajo acoplamiento, adaptable y escalable.

Motivaciones:

Independencia antes los cambios: las interfaces estandarizadas entre capas limitan el efecto de cambios de código a la capa a modificar.

Reutilización de servicios: brindados por la interfaz brindada por cada capa.

Mejorar la portabilidad: los cambios de hardware, del sistema operativo, de motor de base de datos, y todo lo que afecté directamente a una capa, se pude modificar sin alterar al resto de las capas.

Funcionamiento de las diferentes capas:

Los clientes móviles tanto de habitantes, contactos de confianza y la administración realizan peticiones a la capa de Servidor Web. Los diferentes Gadget también realizan peticiones para registrar los eventos.

El Servidor Web recibe la petición Web que le llego, y delega la resolución de las peticiones a la capa de aplicaciones.

La capa de aplicaciones resuelve la lógica de la petición, que se le ha realizado y solicita los accesos a la base de datos, cada vez que sea necesario. Esta capa para el caso de la gestión de eventos, tiene que tener en cuenta cuando un evento está por fuera de los valores permitidos, y entonces generar una notificación a los clientes interesados.

Finalmente, la capa de administración de base de datos mantiene la persistencia de los objetos o entidades que manipula la capa de aplicaciones, respondiendo a las peticiones que le realiza la capa adyacente.

Patron N-Tier Client Server

Se utiliza el patrón para implementar los distintos módulos del sistema Home Safe Home. Corresponde a la vista de ejecución, que destaca sobre la estructura de capas del patrón Layered las comunicaciones entre las capas y sienta las bases para la distribución de estas capas en los niveles de hardware de la arquitectura.

Cliente móvil

Administración

Gadget

Cliente móvil

Contacto

Cliente móvil

Habitante

Servidor Web

Servidor de Aplicaciones

Servidor de Base de Datos

Motivaciones:

Es una arquitectura que permite que un Servidor Web se comunique con clientes livianos, tanto clientes móviles como el gadget, ambos no tienen exigencias de hardware a excepción de la conexión a internet.

Comunicaciones síncronas para manejar las transacciones respetando la performance y confiabilidad requerida.

Separación de los distintos intereses en varias capas lógicas, facilitando las modificaciones y extensibilidad del sistema

Patrón Publish – Suscribe

Este patrón es viable de aplicar en os componentes independientes de gestión de notificaciones o gestión avisos que publican notificaciones, avisos respectivamente y los diferentes usuarios se suscriben a ellos.

El principal beneficio es desacoplar los componentes que producen los tópicos de quienes los consumen, lo que hace que la arquitectura posible de mantener y evolucionar. Esta arquitectura a su vez nos permitirá que el sistema sea flexible y adecuada a para mensajes asincrónicos uno a muchos, muchos a uno.

Situaciones donde aplica:

Cuando el administrador desea publicar un aviso que será enviado a los habitantes del edificio como puede ser el cambio de guarda del edificio o cuando por ejemplo se está realizando la recarga de los matafuegos.

Entonces el subsistema de gestión de avisos, realiza la publicación del aviso y esta situación será notificada al dispositivo móvil de los habitantes del edifico. El aviso refleja una ocurrencia que afecte o atente contra la seguridad de los habitantes del edificio.

Componente Ejecución Móvil

Tópico: Cambio de guardia

Subsistema de Avisos

Cuando se produce un evento como la perdida de monóxido de carbono, el módulo de eventos y notificaciones trabajan en conjunto para generar una notificación que será enviada a los usuarios que estén registrados al gadget en particular que detecto el evento, ya sean los habitantes del departamento o usuarios de confianza que se hayan registrado para recibir notificaciones.

Tópico: Perdida de monóxido de carbono

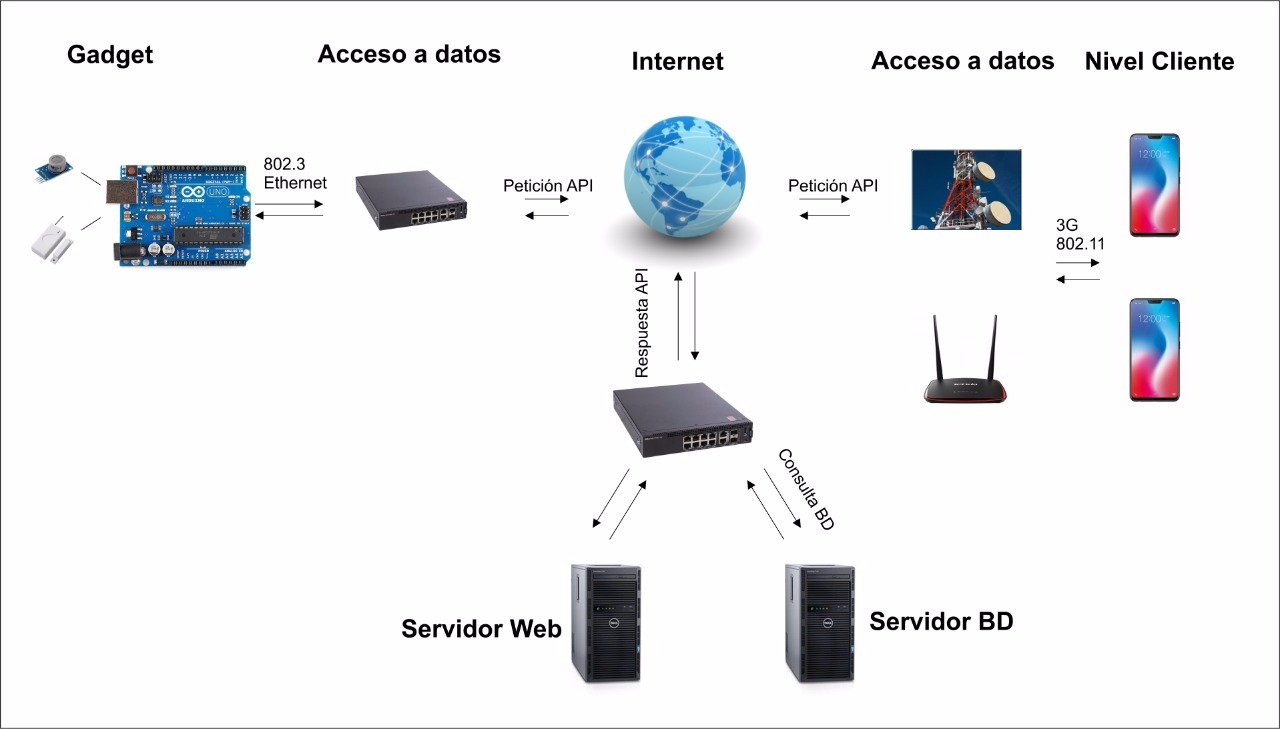
Componente Ejecución Móvil

Subsistema de Notificaciones

Arquitectura del Sistema

Arquitectura de despliegue a nivel de Hardware

En base a los patrones anteriormente nombrados, se realizó el siguiente planteamiento de la arquitectura del sistema desarrollado se tiene en cuenta los dispositivos a utilizar, y los usuarios finales que van a utilizarlo.



En el diagrama se observa la estructura del sistema y la manera como fluyen los datos y la información en el sistema a través de cada una de las capas que se presentan a continuación:

En el nivel Gadget se encuentran la placa y sensores que integran las mediciones realizadas. La placa se encarga de suministrar energía al gadget y ser el enlace con la interfaz de comunicación, mediante el programa cargado en el Arduino.

Capa de acceso a datos, que es la encargada de establecer la comunicación entre los datos que fueron sensados y el Servidor Web de la siguiente capa. El gadget cuenta con un módulo de conexión que facilita la transferencia de datos a la red, un protocolo de comunicaciones y un lenguaje de intercambio que facilita la transferencia de datos.

Capa de transporte, se encarga de interconectar las mediciones recibidas por el servidor web con el resto de los dispositivos que quieran acceder a los datos a través de Internet. Consta de un sistema diseñado en varias capas, como se comentó anteriormente en los patrones que fueron seleccionados, donde la primera capa es un servidor API REST que permite que los clientes móviles le realicen las peticiones, y delega la tarea a las siguientes capas para realizar el registro en la Base de Datos Relacional, donde se almacenan los eventos recibidos.

Capa de acceso a datos, queda explicito que los clientes pueden obtener la información a través de internet, ya sea por una conexión wi fi o a partir de una conexión 3G.

Capa de aplicación, en esta capa se encuentran los clientes front-end que se comunican con el servidor de la capa de transporte, mostrando así, de manera cómoda para los usuarios, los resultados que se obtienen de las solicitudes realizadas al servidor. Permitiéndoles recibir alertas de notificaciones, avisos o reclamos que ocurran en el hogar o edificio.

Diseño del Gadget

A continuación, se describen los dispositivos físicos que se usan para este sistema, clasificándolos como sensores, placas y los diferentes módulos de comunicación.

Sensores

Los sensores a utilizar en el prototipo, se han seleccionado teniendo en cuenta los alcances que fueron definidos en el plan de proyecto, a su funcionamiento, variables de medida, señales de salida, entro otros aspectos. Una breve descripción de los sensores y características se muestran en la tabla a continuación:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Alcance | Sensor | Variables de medida | Unidades de medida | Precisión | Voltaje de funcionamiento | Señal de salida | Imagen |
| Detector humo/incendio |  |  |  |  |  |  |  |
| Monitoreo de temperatura de ambiente | DHT22 | Temperatura y humedad | ºC y % | ±1ºC | 5v | Digital |  |
| Apertura/Cierre de puertas o ventanas | EV1527 | Apertura | Abierto, Cerrado | - | 3v | Binaria |  |
| Pérdidas de monóxido de carbono | MQ7 | Concentración CO | ppm | Depende de la Resistencia de la carga | 5V | Digital |  |