* N TIER

Separación de intereses

Comunicaciones sincronicas

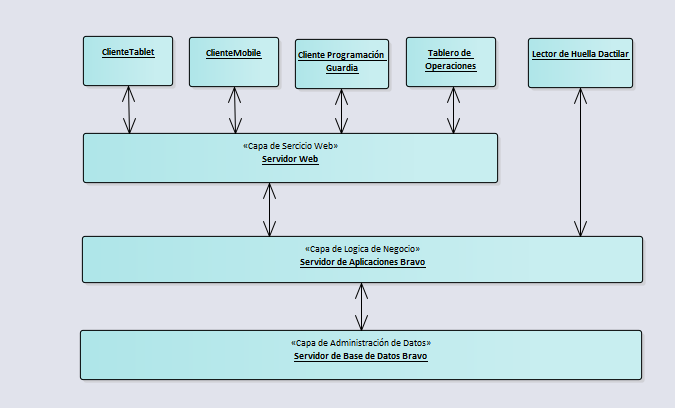
Despliegue flexible

Motivación: La comunicación sincrónica entre capas brinda una mejor performance y confiabilidad en las transacciones.

La separación de intereses en distintas capas lógicas facilitan la futura modificación y extensibilidad del sistema.

Nos permite utilizar clientes web delegados que no requieren demasiado hardware y también la distribución de la capa de datos en distintos servidores facilitando la redundancia ante eventuales catástrofes.

Aplicación: Corresponde a la vista de ejecución runtime que destaca sobre una estructura de capas, las comunicaciones entre las mismas y muestra la distribución de las capas en los niveles de hardware de la arquitectura.



* MESSAGING

Comunicación asincrónica

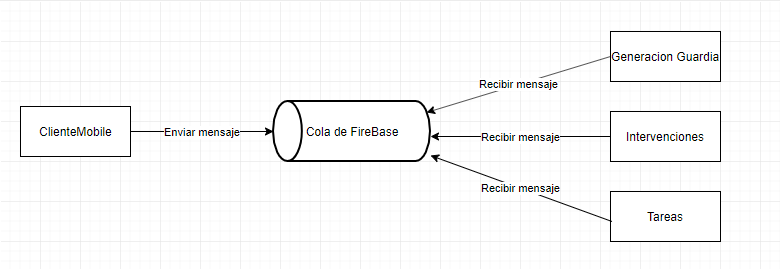
Calidad de servicio configurable

Bajo acoplamiento

Motivación: A medida que el sistema solicita el envio de mensajes, se enviaran los mensajes lo mas rápidamente posible en un caso critico como es la notificación a bomberos de llamado a intervenciones.

(Esto es para mensajes de whatsapp, notificaciones por siniestros y programación de guardias)

Aplicación: Para resolver el envio de mensajes via whatsapp a un grupo de destinatarios aprovechando la potencialidad del patron y las comunicaciones asincrónicas que soporta la falta eventual de comunicación. También se utiliza este patron al implementar un servicio de Google firebase que maneja las colas de mensajes que se enviaran como notificaciones hasta que los mismos sean transmitidos



* PROCESS COORDINATOR:

Encapsulamiento del proceso

Bajo acoplamiento

Comunicación flexible

Motivación: Permite dividir un proceso en subprocesos que no se conocen entre si sino que resuelven lo que les compete únicamente. Podremos así distribuir la carga en uno o mas servidores y en caso de ser necesario modificar el código será mucho mas facil

Aplicación: Dividiremos el proceso de programación de guardias en subprocesos tales como: Disponibilidad del personal, edición de guardias, envio de notificaciones y creación de guardias con bomberos. Este patron nos permite realizar este proceso independientemente de donde se encuentra distribuido cada componente.

