

Instituto Tecnológico de Costa Rica

ESCUELA DE COMPUTACIÓN – DISEÑO DE SOFTWARE

EDGAR SOLÓRZANO

200940313

FRANK BRENES

201136757

JOHN LARGAESPADA

201173581

Contenido

Introducción	4
Marco Teórico	5
Ingeniería de Software	5
Métricas para medir la calidad de Abstracción	5
Acoplamiento	5
Cohesión	5
Suficiencia	5
Compleitud	5
Primitividad	5
Escogencia de métodos	5
Representación de Abstracciones	5
Calidad de Software	5
Aseguramiento de la Calidad	5
Factores de Calidad	6
Confiabilidad	6
Eficiencia	6
Mantenibilidad	6
Usabilidad	6
Proceso de Diseño de Software	6
Diseño Top-Dow	6
Diseño Bottom-Up	6
Principios de Diseño	6
Proceso de Diseño de Arquitectura	7
Primer paso: Boceto general de arquitectura	7
Segundo paso: Refinamiento	7
Tercer paso: Descripción de arquitectura	7
Patrones de Diseño	7
Abstracción-ocurrencia	7
Jerarquía general	7
Jugador-rol	7
Singleton	7
Observador	7

Delegado	7
Adaptador.....	7
Fachada	7
Inmutable	7
Interface solo lectura	8
Resumen Ejecutivo	8
Propósito:.....	9
Especificación de los Requerimientos:.....	9
Introducción la Especificación de Requerimientos	9
Propósito	9
Alcance	9
Personal involucrado.....	9
Definiciones, acrónimos y abreviaturas	10
Referencias.....	10
Resumen.....	10
Descripción general.....	10
Perspectiva del producto	10
Funcionalidad del producto	10
Características del usuario	10
Restricciones	11
Suposiciones y dependencias.....	11
Requisitos específicos	11
Requerimientos Funcionales	11
Requerimientos no Funcionales	14
Requisitos comunes de las interfaces	16
Interfaces de usuario.....	16
Interfaces de hardware	16
Interfaces de software	16
Interfaces de comunicación	16
Requisitos funcionales.....	17
Requisito funcional 1.....	17
Requisito funcional 2.....	17
Requisito funcional 3.....	17

Requisito funcional 4.....	17
Requisito funcional 5.....	17
Requisito funcional 6.....	17
Requisito funcional 7.....	18
Requisito funcional 8.....	18
Requisito funcional 9.....	18
Requerimiento funcional 10.....	18
Requerimiento funcional 11.....	18
Requerimiento funcional 12.....	18
Requisitos no funcionales	18
Requisitos de rendimiento	18
Seguridad.....	18
Fiabilidad	19
Mantenibilidad	19
Portabilidad	19
Prioridades (funcionalidades y factores de calidad más importantes).....	19
Descripción del diseño de alto nivel.....	20
Diagrama de arquitectura conceptual	20
Diagrama de paquetes	21
Diagrama de componentes	22
Diagrama de despliegue.....	23
Descripción del diseño de manera detallada	24
Diagrama de Clases	24
Justificación de patrones usados	24
Problemas de diseño	25
Interacción de sistemas externos.....	25
Otros detalles	25
Conclusión	25

Introducción

Para el presente proyecto se pretende el desarrollo de una aplicación web enfocada en la calendarización de la rutina de trabajo para las personas ocupadas de la actualidad, tratando de contemplar todas las actividades y necesidades que puede tener.

La aplicación será colgada en un servidor gratuito que se encargará de permitir el acceso a la página en cualquier momento, diseñada bajo la práctica de desarrollo RWD (Responsive web design), lo que permitirá poder ser apreciada de la mejor manera desde cualquier dispositivo móvil (smartphone y/o tablet) además de un equipo de trabajo mayor como lo es una computadora. De esta manera permitirá una mayor facilidad de acceso y usabilidad para quien quiera aprovechar la herramienta.

Además, el desarrollo de la aplicación estará acompañado de una documentación donde se presentarán los pasos precedentes al desarrollo de la aplicación; se mostrarán los principales puntos y procesos que ayudaron al diseño tanto de la fachada de la aplicación como de la arquitectura interna, puntos que se desarrollarán más adelante.

Marco Teórico

Ingeniería de Software

Disciplina que comprende todos los aspectos de la producción del software desde las etapas iniciales de la especificación del sistema, hasta el mantenimiento de este después de que se utiliza.

Métricas para medir la calidad de Abstracción

Acoplamiento

Medida de la fuerza de una asociación establecida con una conexión entre un módulo y otro.

Cohesión

Nivel de conectividad entre los elementos de un mismo módulo.

Suficiencia

Determina si un módulo incorpora suficientes características de abstracción para permitir interacción significativa y eficiente.

Complejidad

Determina si la interfaz del módulo (o clase) captura todas las características significativas de la abstracción, en otras palabras, determina si una interfaz es suficientemente general para ser usada por cualquier persona.

Primitividad

Determina qué tanto se utilizan las facilidades propias de un lenguaje. Generalmente se realiza mediante una comparación con otro módulo.

Escogencia de métodos

Se realizan al ser el proceso de diseño un método iterativo, se utilizan patrones y técnicas para mejorarlo, donde se destacan la semántica funcional y la semántica de tiempo y espacio.

Representación de Abstracciones

Se sabe de antemano que es imposible capturar un sistema complejo en un solo diagrama por lo que se aprovechan distintos diagramas UML para dar esta representación, como lo son los diagramas estructurales (paquetes, clases y componentes) que se encargan de representar la organización de los elementos dentro de un sistema y los diagramas de comportamiento (casos de uso, actividades, interacción) que representan la forma en la que elementos interactúan entre sí para ejecutar sus funciones.

Calidad de Software

Grado en que un sistema, componente o proceso cumple con un grupo específica de requisitos, expectativas y necesidades de los usuarios.

Aseguramiento de la Calidad

Patrón planificado y sistemático de todas las acciones necesarias para proporcionar confianza adecuada de que in artículo o producto, se ajusta a lo establecido en los requerimientos técnicos.

Factores de Calidad

Confiabilidad

Permite hacer predicciones del comportamiento, verificando la completitud, consistencia y que sea robusto.

Eficiencia

Presentar un uso razonable de los recursos de los dispositivos relacionados con el programa, donde se medirá el tiempo de procesamiento, memoria, persistencia y los recursos de red.

Mantenibilidad

Mide la facilidad de hacer cambios o correcciones en el programa.

Usabilidad

La facilidad de interacción que hay entre la aplicación y el usuario.

Proceso de Diseño de Software

Para este punto se debe tomar en cuenta la información obtenida en la etapa de requerimientos, diseño actual, tecnología y frameworks que se pueden usar, principios de diseño y lecciones aprendidas.

Diseño Top-Dow

Se inicia por la estructura de alto nivel del sistema, detallando las decisiones de diseño hacia abajo.

Diseño Bottom-Up

Se inicia con decisiones de diseño de módulos de bajo nivel de tal manera que sean reutilizables, de tal manera se ensamblan módulos hasta cumplir con los requerimientos.

Principios de Diseño

- Divide y vencerás
- Aumentar la cohesión
- Reducir el acoplamiento
- Abstracción alta
- Incrementar reusabilidad
- Reusar diseño
- Flexibilidad
- Anticipar obsolescencia
- Diseñar para portabilidad
- Diseñar para pruebas
- Diseñar a la defensiva
- Diseño por contratos

Proceso de Diseño de Arquitectura

Primer paso: Boceto general de arquitectura

Se basa en requerimientos y casos de uso, determinar los componentes y escoger el tipo de arquitectura

Segundo paso: Refinamiento

Se identifican las formas de interacción entre componentes, definición de la distribución de funcionalidades y datos entre componentes, definir uso de frameworks existentes, todo este mientras se valida con base en los requerimientos.

Tercer paso: Descripción de arquitectura

Esta etapa se desarrolla mediante los diagramas UML descritos anteriormente.

Patrones de Diseño

Abstracción-ocurrencia

Consiste en mantener en una misma clase la información que comparten y abstraer de ella la información diferente

Jerarquía general

Se representa cuando objetos en una jerarquía pueden tener elementos superiores e inferiores, pero eso no significa que todos los objetos van a tener subordinados.

Jugador-rol

El rol es un conjunto particular de propiedades asociadas a un objeto y este objeto puede tener distintos roles según el contexto.

Singleton

Se representa en las clases que sólo existe una instancia

Observador

Define una dependencia del tipo uno-a-muchos entre objetos, de manera que cuando uno de los objetos cambia su estado, notifica este cambio a todos los dependientes.

Delegado

Determina que cuando se diseña un método de una clase, se requiere de otro método de otra clase para implementarlo.

Adaptador

Se procede a usar cuando se tiene una jerarquía de clases que se quiere usar desde otra clase y esta clase reusada también pertenece a su propia jerarquía de clases.

Fachada

Se da cuando los programadores deben manipular muchas clases de muchos módulos.

Inmutable

Se declara a un objeto como inmutable cuando una vez que es creado éste nunca cambia.

Interface solo lectura

Se desempeña cuando se quiere dar privilegio para que una clase actúe como mutable o inmutable.

Resumen Ejecutivo

La página “VNote” es una página web que funciona como una agenda virtual, cuál va a ser útil para cualquier persona que así la requiera. Ya sea para cursos de estudiantes, profesores y/o proyectos profesionales. La aplicación va a estar dividida en 3 grandes áreas: Contactos, Apuntes y Eventos. Los usuarios pueden registrarse en el sitio (iniciando sesión a través de Facebook) y contar con un respaldo de todos los contactos (previamente ingresados por él), tendrá la facilidad para crear eventos o actividades por realizar en el futuro. La página podrá informarle al usuario sobre los acontecimientos que debe de hacer cada día como tareas por hacer, además de tener la opción de hacer un tweet con los datos o apuntes registrados. La aplicación web será albergada en un servidor de un tercero con hosting gratuito, esto para que sea accesible a través de un dominio público y a la vez cumplir con el requisito de IaaS. Se propone desarrollarla en HTML, con PHP y usando MySQL como base de datos.

- Contactos: los usuarios de la página podrán ingresar y almacenar la información de sus contactos, además de eliminarlos o modificarlos. Esto le permitiría contar con un respaldo de sus contactos. A la vez, el sitio brindaría la funcionalidad de que sus usuarios puedan exportar a un archivo de texto sus contactos y de ésta manera no perder nunca un número telefónico. Los contactos contarán con información como nombre, dirección, teléfono, correo, cuentas de redes sociales, entre otros.

Se podría brindar la opción de realizar búsquedas de contactos por nombre y apellidos.

- Eventos: el usuario podrá crear actividades a realizar en su quehacer cotidiano (reuniones, clases, proyectos, entregas...), además la aplicación mostrará las prioridades de cada uno de los eventos, así como eliminarlos o modificarlos de ser necesario.
- Apuntes: esta parte o funcionalidad de la aplicación, le va a permitir al usuario tener apuntes separados por temas. Por ejemplo, un estudiante está llevando un curso llamado Diseño de Software y desea tomar apuntes sobre el mismo, pero a su vez tiene un proyecto ajeno a la Universidad llamado Desarrollo Comunal del barrio Los Yoses, entonces va a tener la posibilidad de llevar apuntes de ambos sin ningún problema. Los apuntes van a poder llevar un texto descriptivo y contar la opción de adjuntar archivos a los mismos. Además de que van a contar con un atributo de Prioridad, la cual puede ser Alta, Media o Baja. Si luego el usuario desea realizar una consulta sobre los apuntes tomados de un cierto “tema” y filtrarlos por una determinada prioridad, lo va a poder hacer. Así como la opción de mostrar sólo los apuntes que se hayan tomado en un cierto rango de fechas.

Además el proyecto estará respaldado para su desarrollo por las bases de un marco teórico enfocado en los buenos principios del diseño de software, tomando para la construcción del proyecto principio

Propósito:

La aplicación se crea con el propósito de facilitar la rutina de las personas ocupadas, inspirada en las rutinas de clase y los trabajos por hacer, se desea mediante una aplicación llevar de manera ordenada y detallada cualquier tipo de compromiso o quehacer, además de poder contar con la facilidad de guardar apuntes sobre esos compromisos o cualquier otra nota rápida que deba ser tomada, así como enlazar archivos a esos eventos. Además de contar con todos tus contactos al alcance de la mano, para no pasar por la pena de pedir de nuevo el número de los contactos en caso de extraviar el celular o que sea robado.

Especificación de los Requerimientos:

Introducción la Especificación de Requerimientos

Este documento es una Especificación de Requisitos Software (ERS) para el Sistema de información vNote para la gestión de contactos, eventos y notas importantes para la rutina de trabajo o estudio. Esta especificación se ha estructurado basándose en las directrices dadas por el estándar IEEE.

Propósito

El presente documento tiene como propósito definir las especificaciones funcionales, no funcionales para el desarrollo de un sistema de información web que permitirá gestionar distintos contactos, eventos y notas. Éste será utilizado por estudiantes, profesores y/o directivos.

Alcance

Esta especificación de requerimientos está dirigida al usuario del sistema, para continuar con el desarrollo de una aplicación educativa/autodidáctica sobre la automatización del itinerario del estudiante/trabajador actual, la cual tiene por objetivo principal el gestionar las distintas actividades laborales/sociales (Reuniones, Eventos, Clases, Citas e Información al respecto de cada uno) y contactos (Teléfono, cuentas de redes sociales, etc.).

Personal involucrado

Nombre	Edgar Solórzano Pacheco
Rol	Analista, diseñador y programador
Categoría Profesional	Estudiante ATI
Responsabilidad	Análisis de información, diseño y programación de VNote
Información de contacto	edgarspr@gmail.com

Nombre	Frank Brenes Alvarado
Rol	Analista, diseñador y programador
Categoría Profesional	Estudiante ATI
Responsabilidad	Análisis de información, diseño y programación de VNote
Información de contacto	fbrenes7@gmail.com

Nombre	John Largaespada Pérez
Rol	Analista, diseñador y programador
Categoría Profesional	Estudiante ATI
Responsabilidad	Análisis de información, diseño y programación de VNote
Información de contacto	jolape04@gmail.com

Definiciones, acrónimos y abreviaturas

Nombre	Descripción
Usuario	Persona que usará el sistema para gestionar itinerario
ERS	Especificación de Requisitos Software
RF	Requerimiento Funcional
RNF	Requerimiento No Funcional
API	Application Programming Interface
RWD	Técnica de diseño web
NA	No Aplica
VNote	Agenda Virtual

Referencias

Título del Documento	Referencia
Standard IEEE 830 - 1998	IEEE

Resumen

Este documento consta de tres secciones. En la primera sección se realiza una introducción al mismo y se proporciona una visión general de la especificación de recursos del sistema.

En la segunda sección del documento se realiza una descripción general del sistema, con el fin de conocer las principales funciones que éste debe realizar, los datos asociados y los factores, restricciones, supuestos y dependencias que afectan al desarrollo, sin entrar en excesivos detalles.

Por último, la tercera sección del documento es aquella en la que se definen detalladamente los requisitos que debe satisfacer el sistema.

Descripción general

Perspectiva del producto

El sistema vNote será un producto diseñado para trabajar en entornos web, lo que permitirá su utilización de forma rápida y eficaz, además se desarrollará con RWD para lograr una mejor usabilidad.

Funcionalidad del producto

Aquí debería de ir un diagrama de casos de uso

Características del usuario

Tipo de usuario	Propietario
Formación	NA

Actividades	Gestiona la información y se preinscribe en los cursos
--------------------	--

Restricciones

- Interfaz para ser usada con internet a través de una computadora, Tablet o Smartphone.
- Uso de Dominio (X)
- Lenguajes y tecnologías en uso: HTML, PHP y MySQL.
- El sistema se diseñará según un modelo cliente/servidor.
- El sistema deberá tener un diseño e implementación sencilla, independiente de la plataforma.

Suposiciones y dependencias

- Se asume que los requisitos aquí descritos son estables

Requisitos específicos

Requerimientos Funcionales

Identificación del requerimiento:	RF01
Nombre del Requerimiento:	Autenticación y Registro de Usuario.
Características:	Los usuarios deberán identificarse para acceder a cualquier parte del sistema a través de un API con Facebook.
Descripción del requerimiento:	El sistema podrá ser utilizado por cualquier usuario que ingrese mediante su cuenta de Facebook, ingresando su correo (con el que abrió la cuenta) y la contraseña.
Requerimiento NO funcional:	<ul style="list-style-type: none"> • RNF01 • RNF02 • RNF06
Prioridad del requerimiento: Alta	

Identificación del requerimiento:	RF02
Nombre del Requerimiento:	Gestión de Contactos.
Características:	El sistema ofrecerá al usuario la opción de agregar, eliminar y/o modificar los contactos agregados al sistema.
Descripción del requerimiento:	Gestión de Contactos: El usuario podrá agregar, eliminar y/o modificar los contactos de su agenda.
Requerimiento NO funcional:	<ul style="list-style-type: none"> • RNF01 • RNF03 • RNF06
Prioridad del requerimiento: Alta	

Identificación del requerimiento:	RF03
--	------

Nombre del Requerimiento:	Almacenamiento de Contactos.
Características:	Se definen los rubros del sistema para almacenar los contactos.
Descripción del requerimiento:	<u>Almacenamiento de Contactos:</u> A la hora de agregar un nuevo contacto se requiere como mínimo el nombre y teléfono del contacto, pero además se debe tener la opción de agregar más valores al contacto, Por ejemplo cuentas en redes sociales, correo o dirección.
Requerimiento NO funcional:	<ul style="list-style-type: none"> • RNF01 • RNF02 • RNF03
Prioridad del requerimiento: Alta	

Identificación del requerimiento:	RF04
Nombre del Requerimiento:	Consultar Información del Contacto.
Características:	El sistema ofrecerá al usuario la opción de búsqueda por medio de los datos del contacto.
Descripción del requerimiento:	<u>Consultar Contactos:</u> Muestra a los usuarios información referente a los contactos a través de una búsqueda por sus diferentes datos.
Requerimiento NO funcional:	<ul style="list-style-type: none"> • RNF01 • RNF02 • RNF03
Prioridad del requerimiento: Alta	

Identificación del requerimiento:	RF05
Nombre del Requerimiento:	Exportar.
Características:	El sistema permitirá al usuario exportar sus contactos a un archivo de texto.
Descripción del requerimiento:	Permite al usuario exportar todos sus contactos a un archivo de Formato csv. Además de la opción de generar una vCard para un contacto a la vez.
Requerimiento NO funcional:	<ul style="list-style-type: none"> • RNF01 • RNF04
Prioridad del requerimiento: Media	

Identificación del requerimiento:	RF06
Nombre del Requerimiento:	Gestión de Eventos.

Características:	El sistema ofrecerá al usuario la opción de agregar, eliminar y/o modificar eventos agregados al sistema.
Descripción del requerimiento:	Gestión de Eventos: El usuario podrá agregar, eliminar y/o modificar los eventos de su agenda.
Requerimiento NO funcional:	<ul style="list-style-type: none"> • RNF01 • RNF02 • RNF03 • RNF05 • RNF06
Prioridad del requerimiento: Alta	

Identificación del requerimiento:	RF07
Nombre del Requerimiento:	Mostrar eventos más próximos.
Características:	Permite al usuario analizar la proximidad de los eventos por realizar.
Descripción del requerimiento:	Eventos a ocurrir dentro de la siguiente semana: Permitir al usuario ver en pantalla los eventos más próximos en su itinerario
Requerimiento NO funcional:	<ul style="list-style-type: none"> • RNF01 • RNF05 • RNF06
Prioridad del requerimiento: Alta	

Identificación del requerimiento:	RF08
Nombre del Requerimiento:	Publicación en Twitter.
Características:	Dar la posibilidad al usuario de realizar un tweet sobre las actividades que realice y/o datos, apuntes o contactos registrados en su agenda.
Descripción del requerimiento:	Publicación en Twitter: permite al usuario realizar publicar en Twitter sobre los datos registrados en su agenda digital.
Requerimiento NO funcional:	<ul style="list-style-type: none"> • RNF01 • RNF02 • RNF03 • RNF05
Prioridad del requerimiento: Alta	

Identificación del requerimiento:	RF010
Nombre del Requerimiento:	Gestión de Apuntes/Notas.

Características:	El sistema ofrecerá al usuario la opción de agregar, eliminar y/o apuntes/notas agregados al sistema
Descripción del requerimiento:	Gestión de Apuntes/Notas: El usuario podrá agregar, eliminar y/o modificar los apuntes/notas de su agenda.
Requerimiento NO funcional:	<ul style="list-style-type: none"> • RNF01 • RNF03
Prioridad del requerimiento: Alta	

Identificación del requerimiento:	RF11
Nombre del Requerimiento:	Adjuntar Archivos.
Características:	Permite adjuntar archivos a las notas con tal de complementar el comentario que se realice.
Descripción del requerimiento:	Adjuntar Archivos: Permitir la opción al usuario de adjuntar archivos a las notas en caso de ser requerido.
Requerimiento NO funcional:	<ul style="list-style-type: none"> • RNF01 • RNF04
Prioridad del requerimiento: Alta	

Identificación del requerimiento:	RF12
Nombre del Requerimiento:	Consultar Información de Notas/Apuntes.
Características:	El usuario podrá realizar búsquedas en los apuntes sobre temas específicos, con tal de tener a la mano todas las notas referentes a dicho tema.
Descripción del requerimiento:	El usuario contará con un buscador que le permita observar las notas realizadas sobre un tema específico.
Requerimiento NO funcional:	<ul style="list-style-type: none"> • RNF01 • RNF03 • RNF05
Prioridad del requerimiento: Alta	

Requerimientos no Funcionales.

Identificación del requerimiento:	RNF01
Nombre del Requerimiento:	Interfaz del sistema.
Características:	El sistema presentara una interfaz de usuario sencilla en un formato RWD para que sea de fácil manejo a los usuarios del sistema.

Descripción del requerimiento:	El sistema debe tener una interfaz de uso intuitiva y sencilla, que se acople al dispositivo que está utilizando.
Prioridad del requerimiento: Alta	

Identificación del requerimiento:	RNF02
Nombre del Requerimiento:	Usabilidad del sistema.
Características:	La interfaz del usuario deberá de presentar un ambiente de uso entendible para cualquier usuario sin importar su nivel académico.
Descripción del requerimiento:	La interfaz debe estar diseñado de manera simple tomando como base el protocolo KISS.
Prioridad del requerimiento: Alta	

Identificación del requerimiento:	RNF03
Nombre del Requerimiento:	Mantenimiento.
Características:	El sistema deberá de tener un manual de usuario para facilitar los procesos que serán realizados por el usuario.
Descripción del requerimiento:	El sistema debe disponer de una documentación fácilmente actualizable y entendible que permita realizar operaciones de uso básico con el menor esfuerzo posible.
Prioridad del requerimiento: Alta	

Identificación del requerimiento:	RNF04
Nombre del Requerimiento:	Desempeño
Características:	El sistema garantizara a los usuarios un desempeño en cuanto a los datos almacenado en el sistema ofreciéndole una confiabilidad a esta misma.
Descripción del requerimiento:	Garantizar el desempeño del sistema informático a los usuarios. En este sentido la información almacenada podrá ser consultada y actualizada permanente y simultáneamente, sin que se afecte el tiempo de respuesta.
Prioridad del requerimiento:	

Alta

Identificación del requerimiento:	RNF05
Nombre del Requerimiento:	Confiabilidad continua del sistema.
Características:	El sistema tendrá que estar en funcionamiento las 24 horas los 7 días de la semana. Ya que es una página web diseñada para la administración del tiempo en personas activas tanto social como laboralmente.
Descripción del requerimiento:	La disponibilidad del sistema debe ser continua con un nivel de servicio para los usuarios de 7 días por 24 horas, garantizando un esquema adecuado que permita la posible falla en cualquiera de sus componentes, contar con una contingencia, generación de alarmas.
Prioridad del requerimiento: Alta	

Identificación del requerimiento:	RNF06
Nombre del Requerimiento:	Seguridad en información
Características:	El sistema garantizara a los usuarios una seguridad en cuanto a la información que se procede en el sistema.
Descripción del requerimiento:	Garantizar la seguridad del sistema con respecto a la información y datos que se manejan tales sean documentos, archivos y contraseñas.
Prioridad del requerimiento: Alta	

Requisitos comunes de las interfaces

Interfaces de usuario

La interfaz con el usuario consistirá en un conjunto de ventanas con botones, listas, calendario y campos de textos. Ésta deberá ser construida específicamente para el sistema propuesto y, será visualizada desde un navegador de internet desde cualquier dispositivo que tenga acceso a éste.

Interfaces de hardware

Será necesario disponer de dispositivos con acceso a internet.

Interfaces de software

- Sistema Operativo: NA.
- Explorador: NA.

Interfaces de comunicación

Los servidores, clientes y aplicaciones se comunicarán entre sí, mediante protocolos estándares en internet, siempre que sea posible. Por ejemplo, para transferir archivos o documentos deberán utilizarse protocolos existentes (FTP u otros convenientes).

Requisitos funcionales

Requisito funcional 1

- *Autenticación y Registro de Usuarios*: los usuarios deberán identificarse para acceder a cualquier parte del sistema.
- El sistema deberá autenticarse mediante Facebook.

Requisito funcional 2

- *Gestión de Contactos*: El sistema ofrecerá al usuario la opción de agregar, eliminar y/o modificar los contactos agregados al sistema.
 - *Agregar Contacto*: Permite al usuario una vez que haya accedido a su cuenta crear un nuevo contacto.
 - *Modificar Contacto*: Permite al usuario una vez que haya accedido a su cuenta seleccionar un contacto y modificar los datos referentes a él.
 - *Eliminar Contacto*: Permite al usuario una vez que haya accedido a su cuenta seleccionar un contacto y eliminarlo.

Requisito funcional 3

- *Almacenamiento de Contactos*: A la hora de agregar un nuevo contacto se requiere como mínimo el nombre y teléfono del contacto, pero además se debe tener la opción de agregar más valores al contacto, Por ejemplo cuentas en redes sociales, correo o dirección.

Requisito funcional 4

- *Consultar Información*: El sistema ofrecerá al usuario información general acerca de los Contactos, Eventos y/o Apuntes.
 - *Consultar Contactos*: Muestra información disponible sobre los contactos mediante una búsqueda a través del nombre y/o apellidos de éste, así como exportar los contactos.

Requisito funcional 5

- *Exportar*: Permite al usuario exportar todos sus contactos a un archivo de Formato PDF.

Requisito funcional 6

- *Gestión de Eventos*: El sistema ofrecerá al usuario la opción de agregar, eliminar y/o modificar los eventos agregados al sistema.
 - *Agregar Evento*: Permite al usuario una vez que haya accedido a su cuenta crear un nuevo evento.
 - *Modificar Evento*: Permite al usuario una vez que haya accedido a su cuenta seleccionar un evento y modificarlo.

- *Eliminar Evento*: Permite al usuario una vez que haya accedido a su cuenta seleccionar un evento y eliminarlo.

Requisito funcional 7

- *Definición de Eventos*: Permite al usuario determinar el tipo de actividad se pretende asistir, ya sea una reunión, clase, entrevista....

Requisito funcional 8

- *Prioridad de Eventos*: Permitir al usuario ver en pantalla los eventos o actividades más próximos en su itinerario.

Requisito funcional 9

- *Publicación en Twitter*: E permite al usuario realizar publicar en Twitter sobre los datos registrados en su agenda digital.

Requerimiento funcional 10

- *Gestión de Apuntes/Notas*: El sistema ofrecerá al usuario la opción de agregar, eliminar y/o modificar los apuntes agregados al sistema.
 - *Agregar Apuntes*: Permite al usuario una vez que haya accedido a su cuenta crear un nuevo apunte.
 - *Modificar Apuntes*: Permite al usuario una vez que haya accedido a su cuenta seleccionar un apunte y modificarlo.
 - *Eliminar Apuntes*: Permite al usuario una vez que haya accedido a su cuenta seleccionar un apunte y eliminarlo.

Requerimiento funcional 11

- *Adjuntar Archivos*: Permitir la opción al usuario de adjuntar archivos a las notas en caso de ser requerido.

Requerimiento funcional 12

- *Consultar Información de los Apuntes*: El usuario contará con un buscador que le permita observar las notas realizadas sobre un tema y una fecha en particular.

Requisitos no funcionales

Requisitos de rendimiento

- Garantizar que el diseño de las consultas u otro proceso no afecte el desempeño de la base de datos, ni considerablemente el tráfico de la red.

Seguridad

- Garantizar la confiabilidad, la seguridad y el desempeño de la aplicación web a los diferentes usuarios. En este sentido la información almacenada podrá ser consultada y actualizada permanente y simultáneamente, sin que se afecte el tiempo de respuesta.

Fiabilidad

- El sistema debe tener una interfaz de uso intuitiva y sencilla
- La interfaz de usuario debe ajustarse al dispositivo desde donde se acceda.

Mantenibilidad

- El sistema debe disponer de una documentación fácilmente actualizable que permita realizar operaciones de mantenimiento con el menor esfuerzo posible.
- La interfaz debe contar con manual de usuario que sea fácil de actualizar.

Portabilidad

- La aplicación estará albergada en un servidor de una solución de terceros, pero se deberá de contar con los archivos necesarios que permitan una posible migración.

Prioridades (funcionalidades y factores de calidad más importantes)

Dentro de las funcionalidades más importantes se encuentran las siguientes:

- Un inicio de sesión tanto afiliándose a la página como a través de la cuenta de Facebook del usuario.
- Agregar, eliminar y/o modificar contactos.
- Exportar contactos.
- Agregar, eliminar y/o modificar eventos.
- Visualizar los eventos en la bandeja de inicio.
- Seleccionar eventos por búsqueda de fecha.
- Agregar, eliminar y/o modificar apuntes.
- Adjuntar archivos a los apuntes realizados.
- Realizar búsquedas en los apuntes por su tema.

Dentro de los factores de calidad más importantes en éste proyecto, se encuentran los siguientes:

- Usabilidad: se prioriza el uso de la aplicación web enfocándose en un diseño intuitivo que sea amigable con el usuario, y que sin necesidad de haber leído el manual de usuario, sepa cómo utilizar la mayoría de las funcionalidades brindadas sin mayor complicación.
- Eficiencia: resulta de relevancia ya que la administración de recurso de la aplicación va a ser independiente del dispositivo en donde va a ser ejecuta, ya que se va a poder acceder desde una computadora de escritorio hasta un Smartphone. Y deberá de responder con la misma eficiencia en cualquier caso.
- Confiabilidad: la aplicación web deberá de funcionar correctamente e independientemente del dispositivo en donde haya sido ejecutada.

Descripción del diseño de alto nivel

Diagrama de arquitectura conceptual

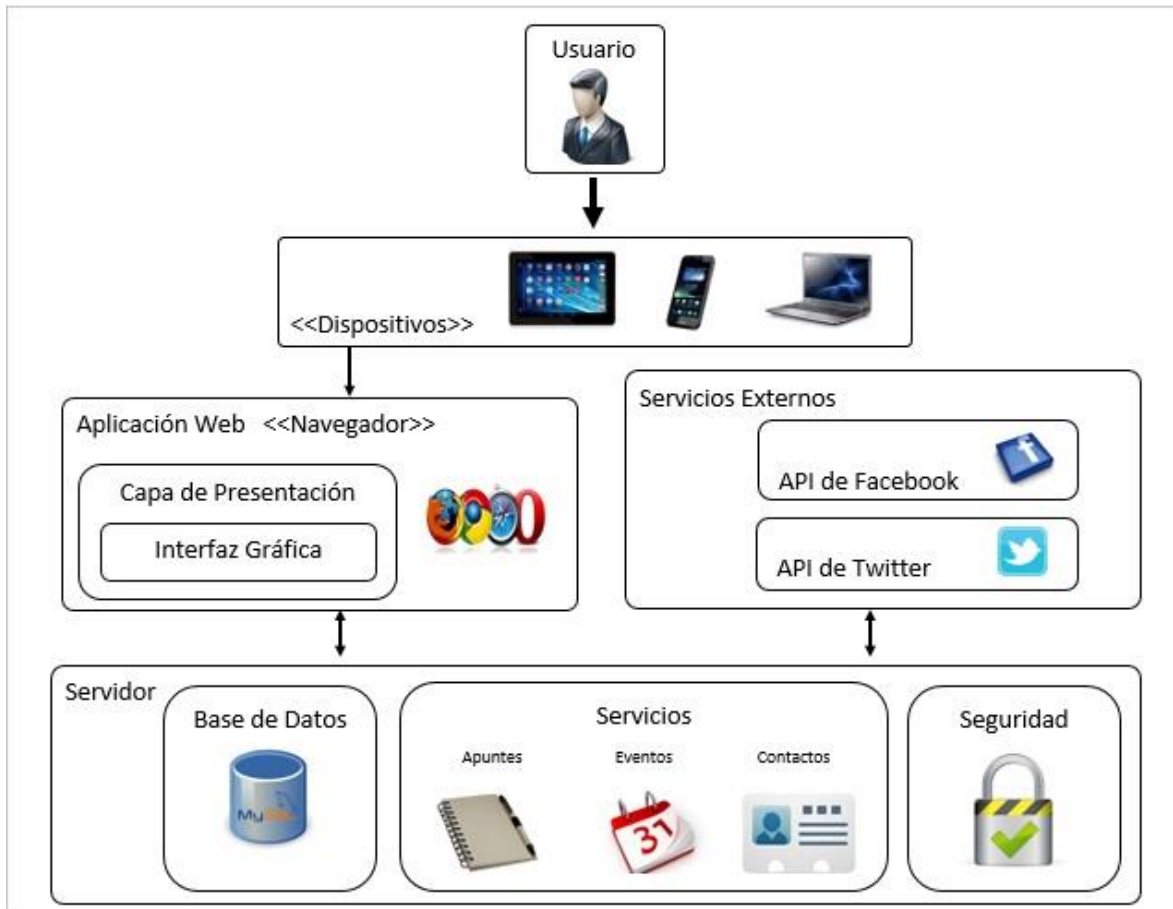


Diagrama de paquetes

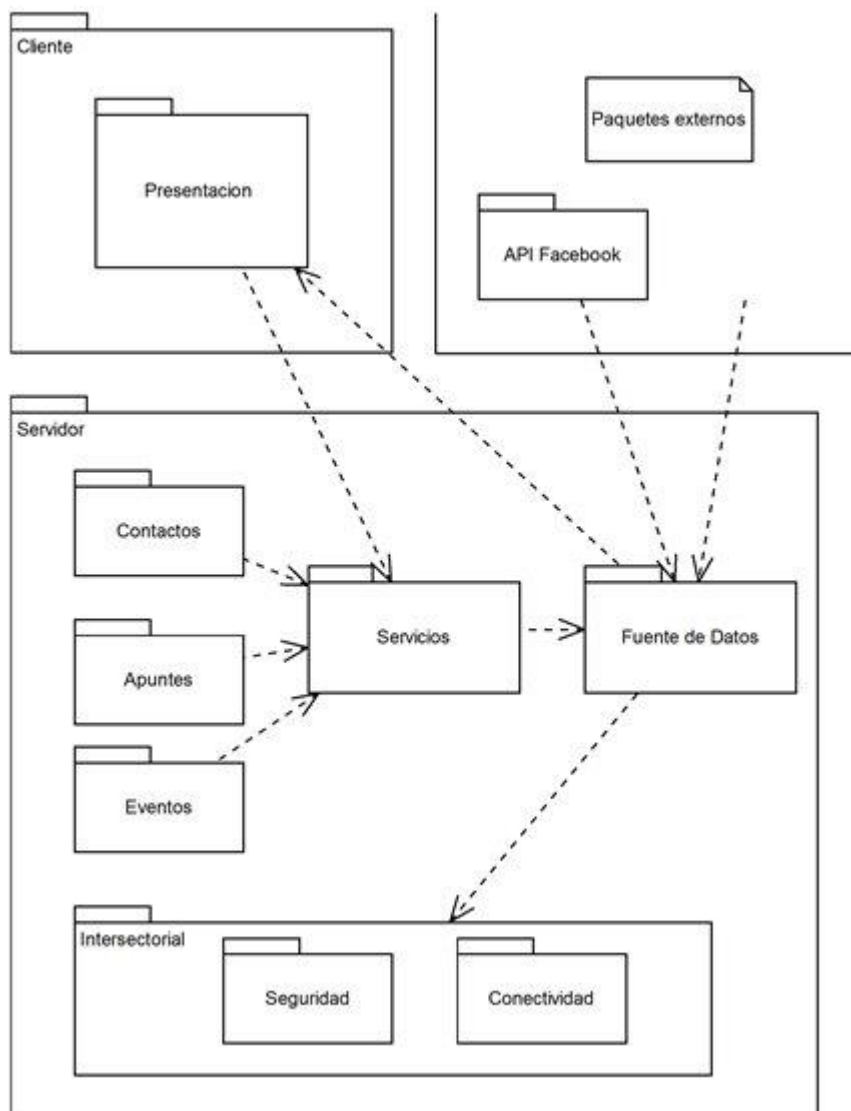


Diagrama de componentes

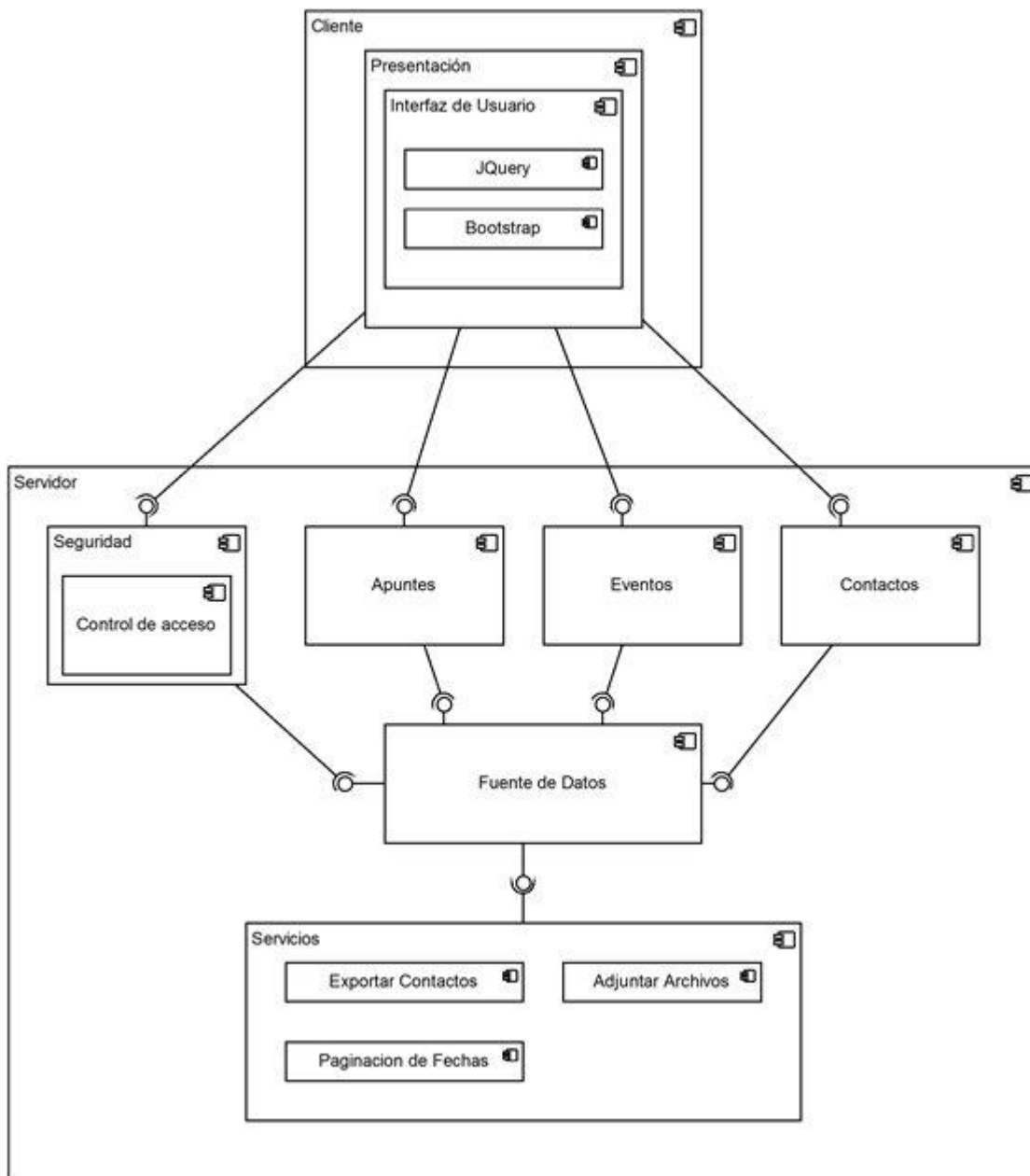
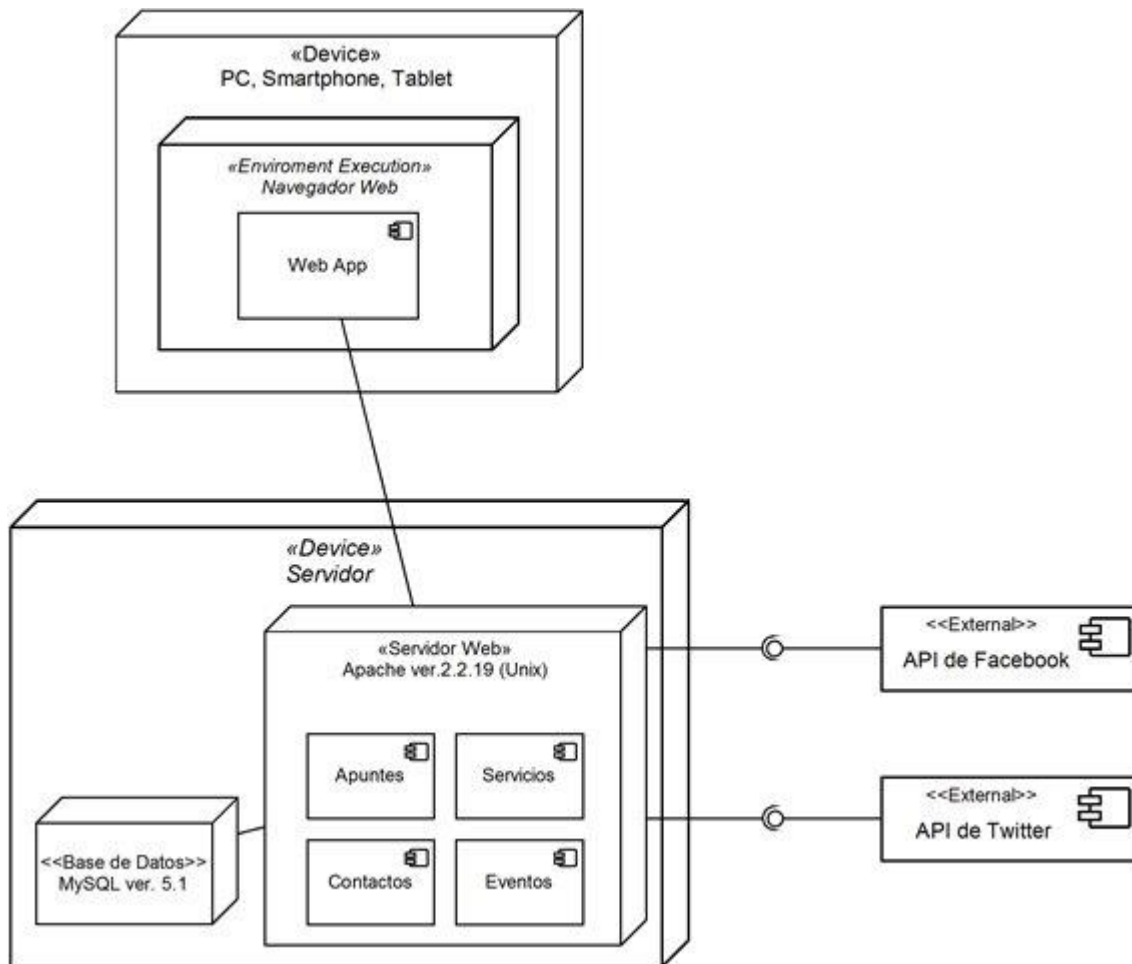
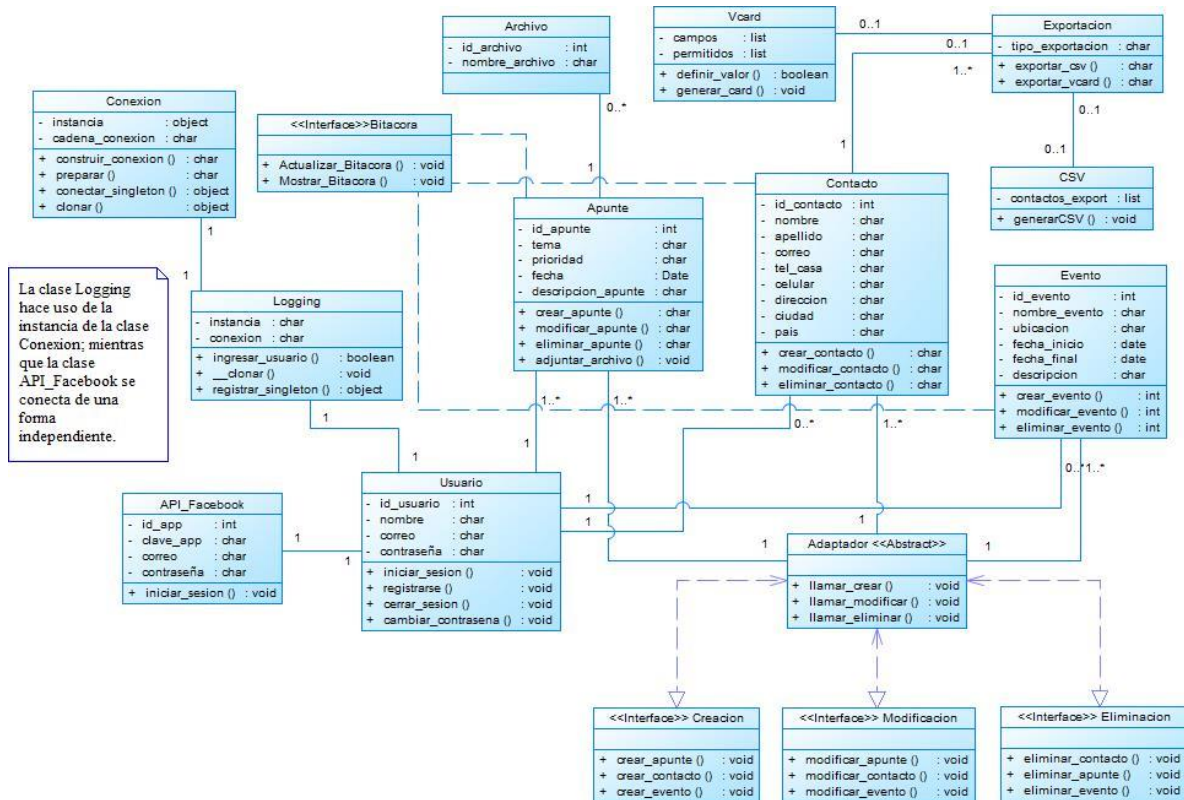


Diagrama de despliegue



Descripción del diseño de manera detallada

Diagrama de Clases



Justificación de patrones usados

Los siguientes van a ser los patrones utilizados en la aplicación web a desarrollar:

1. El Singleton: su uso se sustentará con la implementación del login de usuario. Ya que se contará con una sola instancia de la sesión.
2. Abstracción-ocurrencia: Se ejemplificará el uso de patrón abstracción-ocurrencia en la parte que involucra las notas por usuario. En otras palabras los apuntes, eventos y contactos, van a ser la ocurrencia de la abstracción usuario, ya que se van a abstraer datos de la clase usuario en apuntes, eventos y contactos.
3. Adaptador: Se ejemplifica en el uso de la clase abstracta llamada adaptador, que se va a conectar con las clases interfaces Modificar, Eliminar y Crear; tanto para Eventos, como Notas y Apuntes
4. Delegado: Se evidenciará en el uso de la funcionalidad de Bitácora, ya que los métodos se ejecutan dependiendo de lo que realice en los otros métodos de las otras clases (apuntes, eventos y contactos).

5. Inmutable: ya una vez que el usuario hace inicio de sesión, la instancia de la misma no podrá ser modificada, solo leída o terminada.
6. Fachada: Se implementará mediante el uso de la clase Exportacion, la cual va a llamar métodos de la clase Vcard y CSV.

Problemas de diseño

- Confusión a la hora de hacer los diagramas que los nombres de las clases deben tener son sustantivos y los métodos por su parte deben ser verbos, mientras que los nombres de los componentes deben ser sustantivos.
- Necesidad de incorporar una capa de servicio para evitar el acoplamiento de la capa de presentación con el resto de componentes.
- Falta de experiencia a la hora de desarrollar diagramas de despliegue y arquitectura, ya que se nota al no poder detallarlo de la manera adecuada.
- Determinar el dominio escalable.
- Problemas de herencia múltiple al tratar de utilizar el patrón adaptador en la sección de definición de la clase Adaptador ya que al inicio se definía como subclase de Apunte y Contacto, de tal manera que lo estábamos manejando como herencia múltiple y utilizando composición de una manera absurda.

Interacción de sistemas externos

La aplicación web vNote se conectará a dos API's externos. El primero será el API externo de Facebook llamado Facebook conect el cual le permitirá a los usuarios iniciar sesión en la aplicación web haciendo uso de su cuenta de Facebook y así evitando tener que registrarse o proporcionar información personal adicional. El segundo API externo será de Twitter el cual permitirá compartir mediante la red social cualquier actualización realizada en la aplicación web.

Otros detalles

Para que la aplicación web tenga un dominio público se hará uso de un servidor de aplicaciones web que brinda servicios gratuitos de almacenamiento de aplicaciones y a su vez brinda dominios de manera gratuita.

Conclusión

En esta primera etapa de planificación y diseño de la aplicación web se enfrentaron varias discrepancias con respecto a los diagramas de paquetes, componentes, despliegue, clases y arquitectura conceptual. De la misma manera el uso de patrones de diseño interfirió en la elaboración y modificación del diagrama de clases, teniendo por ende hacer ajustes para disminuir el acoplamiento y aumentar la cohesión de la aplicación web.

Por otro lado el uso de la infraestructura de terceros para el almacenamiento de la versión inicial de la aplicación ha resultado provechosas al tener que configurar el dominio público en un

servidor y utilizar un cliente FTP para conectarse directamente al servidor que alberga la aplicación.