Tabla de contenido

[**Plan de Desarrollo de Software** 3](#_Toc83574519)

[**Introducción** 3](#_Toc83574520)

[**Propósito** 3](#_Toc83574521)

[**Alcance** 3](#_Toc83574522)

[**Resumen** 4](#_Toc83574523)

[**VISTA GENERAL DEL PROYECTO** 4](#_Toc83574524)

[**Objetivo l** 4](#_Toc83574525)

[ **Gestionar partidos de la Liga** 4](#_Toc83574526)

[ **Gestionar calendario de partidos de la Liga.** 4](#_Toc83574527)

[ **Generar un historial de encuentros realizados.** 4](#_Toc83574528)

[ **Gestionar información de cada club.** 4](#_Toc83574529)

[ **Mostrar la ubicación de las distintas canchas.** 5](#_Toc83574530)

[ **Administrar galería de fotos por club.** 5](#_Toc83574531)

[**Suposiciones y restricciones** 5](#_Toc83574532)

[**Suposiciones:** 5](#_Toc83574533)

[**Restricciones:** 5](#_Toc83574534)

[**ENTREGABLES DEL PROYECTO** 5](#_Toc83574535)

[**Fase 1** 5](#_Toc83574536)

[**Fase 2** 5](#_Toc83574537)

[**Fase 3** 5](#_Toc83574538)

[**Evolución del plan del software** 6](#_Toc83574539)

[**ARQUITECTURA EN CAPAS (2, 3 Y N)** 6](#_Toc83574540)

[**Los principales beneficios del estilo de arquitectura basado en capas son** 6](#_Toc83574541)

[ **Abstracción** 6](#_Toc83574542)

[ **Aislamiento** 6](#_Toc83574543)

[ **Rendimiento** 6](#_Toc83574544)

[ **Mejoras en Pruebas** 6](#_Toc83574545)

[ **Independencia** 6](#_Toc83574546)

[**Información:** 6](#_Toc83574547)

[**Logística de presentación** 7](#_Toc83574548)

[**Tecnologías** 8](#_Toc83574549)

[**Herramientas** 9](#_Toc83574550)

[**Logística de Negocio** 14](#_Toc83574551)

[**Presentación distribuida** 14](#_Toc83574552)

[**Presentación remota** 14](#_Toc83574553)

[**Capa de negocios** 15](#_Toc83574554)

[**Lógica distribuida** 16](#_Toc83574555)

[**Ventajas** 16](#_Toc83574556)

[**Desventajas** 16](#_Toc83574557)

[**Capa de datos: Back end** 17](#_Toc83574558)

[**Administración de datos remota** 17](#_Toc83574559)

[**Base de datos distribuida** 18](#_Toc83574560)

[**Diseño interfaz web** 20](#_Toc83574561)

[**Conclusión** 21](#_Toc83574562)

# **Plan de Desarrollo de Software**

## **Introducción**

El proyecto en general tiene como fin desarrollar un sistema de web.

Teniendo en cuenta que la realidad organizacional podría ser mejorada considerablemente, el objetivo es dar solución a aquellas tareas de mayor importancia y que soportan una gran cantidad de datos a ser controlados y de esta manera agilizar los procesos dentro de la institución.

Es por ello por lo que se ha considerado un diseño modular y en base a una metodología iterativa-incremental para implementar las tareas de administración y difusión de información de los campeonatos de la institución, pensando en que a futuro este sistema pueda continuar creciendo hasta abarcar a todas las funciones de la asociación nacional de futbol de la liga nacional.

## **Propósito**

Crear un sistema que pueda controlar automáticamente las operaciones para una buena gestión de inventario y el excelente manejo de control, ya que será de gran ayuda para que los usuarios puedan gestionar digitalmente el entorno del sistema.

## **Alcance**

El sistema debe incluir lo siguiente:

* De las ventas se requiere poder ver las ventas realizadas y ejecutar una nueva venta
* En un apartado de productos se solicita registrador un nuevo producto y poder ver los registros de los productos (también en este apartado se pueda eliminar, actualizar y editar)
* Se requiere registrar nuevos clientes, también poder ver los clientes que ya fueron registrados (en este apartado se debe poder editar y eliminar los clientes)
* Se solicita poder registrar proveedores, también que se pueda observar los proveedores que se encuentran en el sistema (se debe de poder editar y eliminar)
* Se debe de poder registrar usuarios, y de ello se debe poder ver los usuarios que se registraron (De ellos se debe de poder editar y eliminar usuarios)

**Resumen**

En estos tiempos la información se maneja de una forma rápida y confiable, por lo cual tener herramientas que ofrezcan estas dos cualidades se hacen más indispensables, es por esto que los sistemas web son cada día más necesarios para empresarios, comerciantes y para el público en general.

Como un aporte más presentamos una un sistema para gestión de inventario que le permita al público en general realizar sus pedidos desde su hogar o desde cualquier parte de la ciudad capital o cualquiera de los diferentes departamentos del país, teniendo como única exigencia que los usuarios se encuentren registrados en la base de datos para datos de las compras he información.

# **VISTA GENERAL DEL PROYECTO**

## **Objetivo l**

Desarrollar un sistema que satisfaga las necesidades al usuario y en el que podamos realizar un seguimiento de la correcta gestión de inventario con el fin de crear orden y facilitar la gestión de los usuarios que serán los encargados de ser beneficiados con dicho sistema.

## **Suposiciones y restricciones**

### **Suposiciones:**

* Capacitar a los administradores para el manejo correcto del sistema y que su uso sea lo más descifrable para el buen manejo del sistema.
* La información almacenada del sistema será totalmente resguardada por el sistema para poder esquivar el rapto de información y la huida de datos.
* El sistema será proceso eficiente en la gestión para la liga nacional.

### **Restricciones:**

* Solo los administradores pueden realizar cambios.
* Solo los administradores pueden acceder a la información de la base de datos.

# **ENTREGABLES DEL PROYECTO**

## **Fase 1**

* Calendarios de trabajo
* Manual de usuario
* Documento del software
* Presupuesto al Usuario

## **Fase 2**

* Recopilación de información.
* Especificaciones de los diagramas.

## **Fase 3**

* Presentación de los nuevos avances del software.
* Desarrollo de la base de datos.
* Verificación del sistema en el caso de que se presente fallas y modificaciones.

## **Evolución del plan del software**

El plan del desarrollo del sistema se revisará semanalmente testeando y verificando la integridad de la base de datos y ver si el software necesita cambios y modificaciones.

Se verifica la integridad del proyecto para poder refinar detalles del ya mencionado programa.

# **ARQUITECTURA EN CAPAS (2, 3 Y N)**

**Los principales beneficios del estilo de arquitectura basado en capas son**:

* + **Abstracción.** Las capas permiten cambios que se realicen en un nivel abstracto. Usted puede incrementar o disminuir el nivel de abstracción usado en cada capa de la “pila” jerárquica.
  + **Aislamiento**. El estilo de arquitectura de capas permite asilar los cambios en tecnologías a ciertas capas para reducir el impacto en el sistema total.
  + **Rendimiento**. Distribuir las capas entre múltiples sistemas (físicos) puede incrementar la escalabilidad, la tolerancia a fallos y el rendimiento.
  + **Mejoras en Pruebas.** La capacidad de realizar pruebas se beneficia de tener unas interfaces bien definidas para cada capa, así como de la habilidad para cambiar a diferentes implementaciones de las interfaces de cada capa.
  + **Independencia.** El estilo de arquitectura basado en capas el requerimiento de considerar el hardware y los problemas de instalación, así como las dependencias de interfaces externas.

## **Información:**

* De las ventas se requiere poder ver las ventas realizadas y ejecutar una nueva venta
* En un apartado de productos se solicita registrador un nuevo producto y poder ver los registros de los productos (también en este apartado se pueda eliminar, actualizar y editar)
* Se requiere registrar nuevos clientes, también poder ver los clientes que ya fueron registrados (en este apartado se debe poder editar y eliminar los clientes)
* Se solicita poder registrar proveedores, también que se pueda observar los proveedores que se encuentran en el sistema (se debe de poder editar y eliminar)
* Se debe de poder registrar usuarios, y de ello se debe poder ver los usuarios que se registraron (De ellos se debe de poder editar y eliminar usuarios)

## **Logística de presentación**

Aquí se muestro la presentación logística del software a desarrollar, es decir todas las formas, del sistema o menús de dialogo, con los cuales interactúa el usuario.

Los servicios de presentación proporcionan la interfaz necesaria para presentar información y reunir datos.

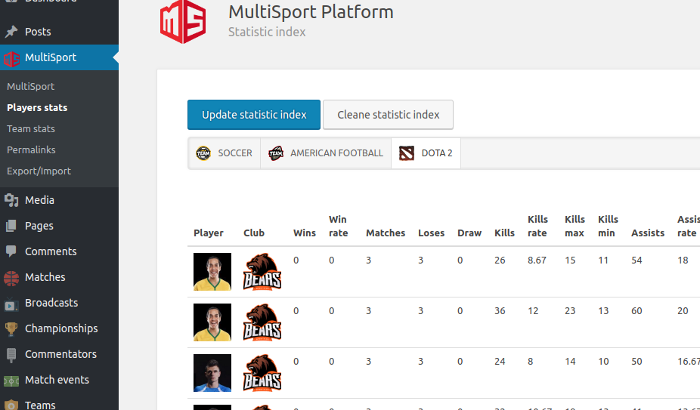
Los servicios de presentación son identificados con la interfaz de usuario, residen en un programa ejecutable localizado en la estación de trabajo del usuario. Aun así, existen oportunidades para identificar servicios que residen en componentes separados.

**Diseño UI**

El Diseño que se aplicará en el sistema será UI ya que las interfaces de la aplicación web, se estará ocupando css, y Bootstrap, para darle mejor vista a la aplicación y que el diseño de interfaz sea más dinámico y creativo para el usuario, y que se vea profesional.



Muestras del Diseño implementado en el sistema.



### **Tecnologías**

En esta sección se definen brevemente las tecnologías a utilizar durante el proceso de desarrollo del proyecto.

**PHP**: Es la sigla que representa Hypertext Pre-Processor. Es un lenguaje gratuito y multiplataforma para programar script del lado del servidor, que se incrustan en el código

**HTML**: Es la sigla que representa HyperText Markup Language. Es un lenguaje de marcado que permite la elaboración de páginas webs.

**JavaScript**: Es un lenguaje de programación interpretado, se utiliza principalmente en el cliente, implementado como parte de un navegador web permitiendo mejoras en la interfaz de usuario y páginas web dinámicas.

**CSS**: Es la sigla que representa Cascading Style Sheet. Es un lenguaje de hojas de estilos creado para controlar el aspecto o presentación de los documentos electrónicos definidos con HTML y XHTML. CSS es la mejor forma de separar los contenidos, su presentación y es imprescindible para crear páginas web complejas.

### **Herramientas**

En esta sección se definen brevemente las herramientas a utilizar durante el proceso de desarrollo del proyecto.

**Visual Studio Code:** es un editor de código fuente desarrollado por Microsoft para Windows, Linux y macOS. Incluye soporte para la depuración, control integrado de Git, resaltado de sintaxis, finalización inteligente de código, fragmentos y refactorización de código.

**MySQL Workbench**: Es un software gratuito distribuido por Oracle Corporation.

Permite la gestión e implementación de un servidor local de base de datos MySQL que servirá para la fase de pruebas.

**MySQL**: Es sistema de gestión de base de datos relacional gratuito que es distribuido y mantenido por Oracle.

**PhpMyAdmin**: Es una plataforma gratuita y multiplataforma para la gestión de base de datos MySQL. Será utilizado para la implementación de base de datos remota.

**Bootstrap**: es una biblioteca multiplataforma o conjunto de herramientas de código abierto para diseño de sitios y aplicaciones web.

**La capa de servicios de presentación es responsable de:**

* Obtener información de usuario.
* Enviar la información del usuario a los servicios de negocios para su procesamiento
* Recibir los resultados del procesamiento de los servicios de negocios
* Presentar estos resultados al usuario

Interfaz de usuario gráfica

Descripción generada automáticamente con confianza baja

Interfaz de usuario gráfica

Descripción generada automáticamente con confianza media

**Usuario**

**Actores primarios:**

* Liga de Futbol, Usuario, Admin
* Registrarse en el campeonato
* Registro de calendarios
* Listar Campeonato, incluir equipos para el campeonato de liga
* Resultados

Interfaz de usuario gráfica

Descripción generada automáticamente

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

**Modelo Vista Controlador**: es un patrón de arquitectura de software que separa los datos de una aplicación, la interfaz de usuario y la lógica de control en tres componentes distintos:

* Modelo: datos.
* Vista: muestra la información del modelo al usuario.
* Controlador: gestiona las entradas del usuario e implementa la lógica de la
* aplicación.

Captura de pantalla de computadora

Descripción generada automáticamente

Interfaz de usuario gráfica

Descripción generada automáticamente

Interfaz de usuario gráfica, Gráfico, Gráfico en cascada

Descripción generada automáticamente

## **Logística de Negocio**

### **Presentación distribuida**

Se distribuye la interfaz entre el cliente y la plataforma servidora.

La aplicación y los datos están ambos en el servidor. Similar a la arquitectura tradicional de un Host y Terminales. El PC se aprovecha solo para mejorar la interfaz gráfica del usuario.

#### **Ventajas**

* Revitaliza los sistemas antiguos.
* Bajo costo de desarrollo.
* No hay cambios en los sistemas existentes.

#### **Desventajas**

* El sistema sigue en el Host.
* No se aprovecha la GUI y/o LAN.
* La interfaz del usuario se mantiene en muchas plataformas.

### **Presentación remota**

* La interfaz para el usuario está completamente en el cliente.
* La aplicación y los datos están en el servidor.

#### **Ventajas**

* La interfaz del usuario aprovecha bien la GUI y la LAN.
* La aplicación aprovecha el Host.
* Adecuado para algunos tipos de aplicaciones de apoyo a la toma de decisiones.

#### **Desventajas**

* Las aplicaciones pueden ser complejas de desarrollar.
* Los programas de la aplicación siguen en el Host.
* El alto volumen de tráfico en la red puede hacer difícil la operación de aplicaciones muy pesadas.

Interfaz de usuario gráfica, Diagrama

Descripción generada automáticamente

## **Capa de negocios**

Esta es la encargada de procesar y validar las reglas del negocio, actúa como un servidor para la capa de presentación y traslada las solicitudes de esta a la capa de datos actuando como cliente de esta. Así mismo también funciona de la misma manera de la forma inversa.

Captura de pantalla de computadora

Descripción generada automáticamente

## **Lógica distribuida**

* La interfaz está en el usuario.
* La base de datos está en el servidor.
* La lógica de la aplicación está distribuida entre el usuario y el servidor.

### **Ventajas**

* Arquitectura más simple que puede manejar todo tipo de aplicaciones.
* Los programas del sistema pueden distribuirse al nodo más apropiado.
* Pueden utilizarse con sistemas existentes.

### **Desventajas**

* Es difícil de diseñar.
* Difícil prueba y mantenimiento si los programas del cliente y el servidor están hechos en distintos lenguajes de programación.

Diagrama

Descripción generada automáticamente

## **Capa de datos: Back end**

Esta capa incluye bases de datos y programas que maneja la lectura y escritura sobre las mismas. En esta capa es donde se deben definir los métodos de acceso y políticas de seguridad utilizadas sobre los datos.

## **SISTEMAS DE INFORMACION**

Es el conjunto formal de procesos que, operando sobre una colección de datos estructurada de acuerdo con las necesidades de una empresa, recopila, elabora y distribuye la información necesaria para la operación de dicha empresa y para las actividades de dirección y control correspondientes, apoyando, al menos en parte, los procesos de toma de decisiones necesarios para desempeñar las funciones de negocio de la empresa de acuerdo con su estrategia.

Interfaz de usuario gráfica

Descripción generada automáticamente

**El nivel de servicios de datos es responsable de:**

* Almacenar los datos
* Recuperar los datos
* Mantener los datos
* La integridad de los datos

### **Administración de datos remota**

* En el cliente residen tanto la interfaz como los procesos de la aplicación.
* Las bases de datos están en el servidor.
* Es lo que comúnmente imaginamos como aplicación cliente servidor

#### **Ventajas**

* Configuración típica de la herramienta GUI 4GL.
* Muy adecuada para las aplicaciones de apoyo a las decisiones del usuario final.
* Fácil de desarrollar ya que los programas de aplicación no están distribuidos.
* Se descargan los programas del Host.

#### **Desventajas**

* No maneja aplicaciones pesadas eficientemente.
* La totalidad de los datos viaja por la red, ya que no hay procesamiento que realice el Host.

### **Base de datos distribuida**

* La interfaz, los procesos de la aplicación, y parte de los datos de la base de datos están en el cliente.
* El resto de los datos están en el servidor.

#### **Ventajas**

* Configuración soportada por herramientas GUI 4GL.
* Adecuada para las aplicaciones de apoyo al usuario final.
* Apoya acceso a datos almacenados en ambientes heterogéneos.
* Ubicación de los datos es transparente para la aplicación.

#### **Desventajas**

* No maneja aplicaciones grandes eficientemente.
* El acceso a la base de datos distribuida es dependiente del proveedor del software administrador de bases de datos.

Tabla

Descripción generada automáticamente con confianza baja

### **Diseño interfaz web**

**Donde cada bloque representa específicamente**:

1. Login.
2. Barra menú.
3. Barra escudos club.
4. Escudo de la Liga.
5. Noticias de Asociación.
6. Noticias club primera división.
7. Noticias club segunda división.
8. Publicidad.
9. Buscar jugador.
10. Calendario de partidos.
11. Reporte del clima.
12. Reglamento de la Liga.

# **ESTRUCTURA DE MICROSERVICIOS**

## **Estructura del sistema**

Es un software donde se llevará el registro de las ventas emitiendo de esa manera las facturas correspondientes,

A continuación, estructura de lo base de nuestro software:

Interfaz de usuario gráfica

Descripción generada automáticamente

**Estructura de sistema aplicándole arquitectura de microservicios:**

Cuando hablamos de microservicios decimos que es una aproximación para el desarrollo de software que consiste en construir una aplicación como un conjunto de pequeños servicios, los cuales se ejecutan en su propio proceso y se comunican con mecanismos ligeros.

Es decir a la hora que de tener los servidores montados en un solo lugar se nos cae un servidor el sistema web queda paralizado, entonces ahí es donde entra la arquitectura de multiservicios aplicando y distribuyendo cada servidor y cada funcionalidad por aparte y siempre por medio de la base de datos aplicando cambios como registro, modificaciones, actualizaciones y eliminaciones, de esa manera si está distribuido cada cosa con su servidor por aparate se evita que a la hora de caer un servidor por ejemplo pagos, los demás servidores siguen trabajando normal, y los desarrolladores se enfocan solo donde está el problema, por eso en este análisis se toma en cuenta la importancia de los microservicios para nuestro software ya que de esa manera construimos una aplicación en conjuntos pequeños los cuales se ejecutan en sus propios procesos y se comunican con mecanismos ligeros por medio de todo los datos que se estarán almacenando en la base de datos.

A continuación, miramos la estructura del sistema ya aplicándole la arquitectura de microservicios:

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

Aquí podemos observar que nuestra sistema cuenta con varios servidores y cada uno de los servidores manejando diferentes procesos y siempre almacenándolos en nuestra base de datos y de esa manera estar comunicados, por ejemplo el usuario a través del servidor del área de ventas el puede entrar al inventario de bodega para realizar una Venta, de esa manera también se lleva el registro en la base de dato de todo los movimientos que se hace y se maneja registros de las facturas emitidas, control de inventario, control de clientes, inventario en bodega.

# **Estrategias de pruebas de software**

# **Conclusión**

En el presente informe se desarrollaron los procesos de planificación, diseño y desarrollo del proyecto “Desarrollo de un Sistema de Gestión de la liga de Futbol Nacional, el cual trata como objetivo principal solucionar los problemas habituales más significativos importantes como la administración y difusión de la información de la liga, eventos que ésta realiza, además de, mejorar la comunicación de sus clubes con sus seguidores(Hinchada, Jugadores, Dirigentes), ya que, no todos conocen la información, historia, dirigentes y ubicación de sus canchas, lo cual mediante perfiles de administración de información de clubes establecidos en el sistema, podrán dar a conocer la información mencionada y necesaria para el conocimiento de todos los que conforman los clubes.

Debido al desarrollo de este sistema se lograron reducir los tiempos de ingreso de datos y la entrega de información a cada plantel o dirigentes que se encuentran asociadas a la institución, para así poder brindar de forma rápida, segura y consistente la información obtenida durante el transcurso de la liga que se realiza.

Con la utilización del sistema se mejoran las relaciones entre los encargados que administran el campeonato, debido a la dimensión del tiempo dedicado, a los cuales el sistema le da solución, obteniendo así tiempo adicional para el empleado en otras tareas, reduciendo las cargas y presión ejercidas sobre el personal.

La metodología y el modelo de datos utilizados permitió llevar a cabo el planeamiento del desarrollo y la decisión de como almacenar los datos de manera ordenada e incremental. Lo que permitió presentar avances de la documentación y del sistema, permitiendo, además, ir realizando correcciones de acuerdo con las necesidades del cliente, como a las supervisiones de parte de la universidad, para lograr cumplir con los objetivos propuestos al inicio del proyecto.

En lo referente a las tecnologías utilizadas, los patrones de diseño y enfoques implementados para llevar a cabo el sistema, éstos sin duda ofrecieron un apoyo importante al desarrollo del proyecto desde principio a fin, siendo el principal desafío el dominarlas por completo y combinarlas de manera efectiva, partiendo de la base de que algunas de ellas no estaban dentro del conocimiento del equipo desarrollador, y no se tenía experiencia utilizándolas como lo fueron JavaScript, JQuery, entre otros. El proyecto se desarrolló de forma ordenada, debido al uso de estas tecnologías, que otorgaron una estructura y reglas de funcionamiento definidas a seguir para la implementación, pruebas y depuración respectivamente.