

S.I.G.D.

Análisis y Diseño de Aplicaciones

Panthercode

Rol	Apellido	Nombre	C.I	Email	Tel/Cel.
Coordinador	Irigoin	Juan Pedro	5.423.370-8	pedroirigoin@gmail.com	092292101
Sub-Coordinador	Pacheco	Nicolás	5.387.574-3	nicolaspacheco1a@gmail.com	095258122
Integrante 1	Cardozo	Santiago	5.476.192-5	santicar041@gmail.com	094650373
Integrante 2	Ramos	Rainer	6.351.063-4	rainerizasiga@gmail.com	097133432

Docente: Barboza, Gabriel

Fecha de culminación:

19/07/2022

PRIMERA ENTREGA

I.S.B.O.

3°BC



ÍNDICE

ÍNDICE	1
1- INTRODUCCIÓN:	2
2- ANTEPROYECTO:	3
2.1- RELEVAMIENTO:	3
2.2- ANÁLISIS F.O.D.A.:	6
2.3- LÓGICA DEL SISTEMA:	7
2.4- ESTUDIO DE FACTIBILIDADES:	10
2.5- DEFINICIÓN DE ROLES DE USUARIO:	16
2.6- PLANIFICACIÓN:	17
2.7- ESPECIFICACIÓN DE REQUERIMIENTOS:	19
2.7.1- REQUERIMIENTOS FUNCIONALES:	19
2.7.2- REQUERIMIENTOS NO FUNCIONALES:	20
2.7.3- ALCANCE:	21
2.7.4- LIMITACIONES:	21
3- ELECCIÓN CICLO DE VIDA:	22
4- CASOS DE USO DE FORMA NARRATIVA	23



1- INTRODUCCIÓN:

El objetivo de Panthercode en este proyecto es el desarrollo de un sistema que permita la gestión de todo a lo que refiere la realización de actividades deportivas con énfasis en Fútbol, Basquetbol y Handball, pero permitiendo su uso en otros deportes desde un punto de vista administrativo. El nombre de este sistema es Sistema Informático de Gestión Deportiva o S.I.G.D por sus siglas.

El enfoque de este desarrollo sería la creación de múltiples herramientas contenidas en una sola aplicación para su uso en la educación media, como ayuda a la organización en los cursos de deporte de UTU, tanto como en su potencial uso en clubes deportivos. Estas herramientas proporcionarán la capacidad para:

- Visualizar y registrar partidos con sus estadísticas.
- Crear y administrar equipos de forma exhaustiva, con la capacidad de controlar cada detalle que hace al equipo deportivo, incluyendo:
 - o Manejo de jugadores y sus estadísticas.
 - o Estadísticas generales del equipo (partidos ganados, perdidos, jugados, etc.)



2- ANTEPROYECTO:

2.1- RELEVAMIENTO:

El relevamiento se utiliza para recolectar información y entender la situación actual, para esto fueron utilizadas varias técnicas, además de la lectura y análisis de la letra como método de “revisión de registros”, fue utilizada la técnica de entrevista con el coordinador de informática para tratar las siguientes cuestiones:

- ¿Qué quiere el cliente que haga nuestro sistema?
- ¿Cómo se lleva actualmente el registro de los deportes?
 - En caso de existir ¿Es posible ver un ejemplo?
 - ¿El sistema actual se comunica con otro sistema del que dependa?
 - En caso de no existir ¿Cómo quisieran que se registrara?
- ¿Qué gente participa de estos registros?
- ¿Qué datos se almacenan o almacenarán en estos registros?
- ¿Cada cuánto se registran estos datos?
- ¿Cómo se respaldan los datos?



ENTREVISTA:

ENTREVISTADO:

Luís E. Fagúndez – Coordinador de Informática

ENTREVISTADORES:

- Santiago Cardozo
- Rainer Ramos
- Juan Pedro Irigoin

PREGUNTAS FORMULADAS:

1. ¿Qué quiere el cliente que haga nuestro sistema?
2. ¿Cómo se lleva actualmente el registro de los deportes?
 - 2.1. En caso de existir ¿Es posible ver un ejemplo?
 - 2.1.1. ¿El sistema actual se comunica con otro sistema del que dependa?
 - 2.2. En caso de no existir ¿Cómo quisieran que se registrara?
3. ¿Qué gente participa de estos registros?
4. ¿Qué datos se almacenan o almacenarán en estos registros?
5. ¿Cada cuánto se registran estos datos?
6. ¿Cómo se respaldan los datos?
7. Un profesor mencionó la ampliación del área de funcionamiento del sistema para que incluya la plaza frente al ISBO. ¿Está de acuerdo con eso?



RESPUESTAS:

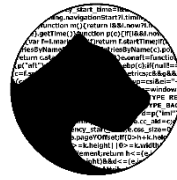
- 1- “Que cumpla con los principios actuales de cualidades de software”.
- 2- “En un Excel, tienen un profesor de tallerista de deporte que se encarga de registrar los partidos”.
- 3- “No gurises, pero no hay mucho que ver”.
- 4- “No, no se comunica con nada”.
- 5- “Solo se registran los jugadores que son los alumnos y algunos profesores como jueces”.
- 6- “La idea es que en cada partido se registre el nombre, apellido para los jugadores de cada equipo. Luego de cada partido, los usuarios podrían acceder a un registro e imprimirlo para archivar de manera física”.
- 7- “Se registran todos los participantes de cada partido”.
- 8- “Se le saca capturas que se guardan en el portafolio docente, y se guardan en papel, se imprime, y todo se archiva en UTU central.”
- 9- “No es necesario, pero si lo quieren hacer pueden usar un tp-link de largo alcance conectado a la red para crear una red privada accesible desde la plaza, donde los profesores puedan entrar con las laptops a su aplicación y registrar desde ahí”.

Nota. Las respuestas no son las palabras exactas del entrevistado, se cambiaron o se redujeron para representar de forma clara la idea expresada.

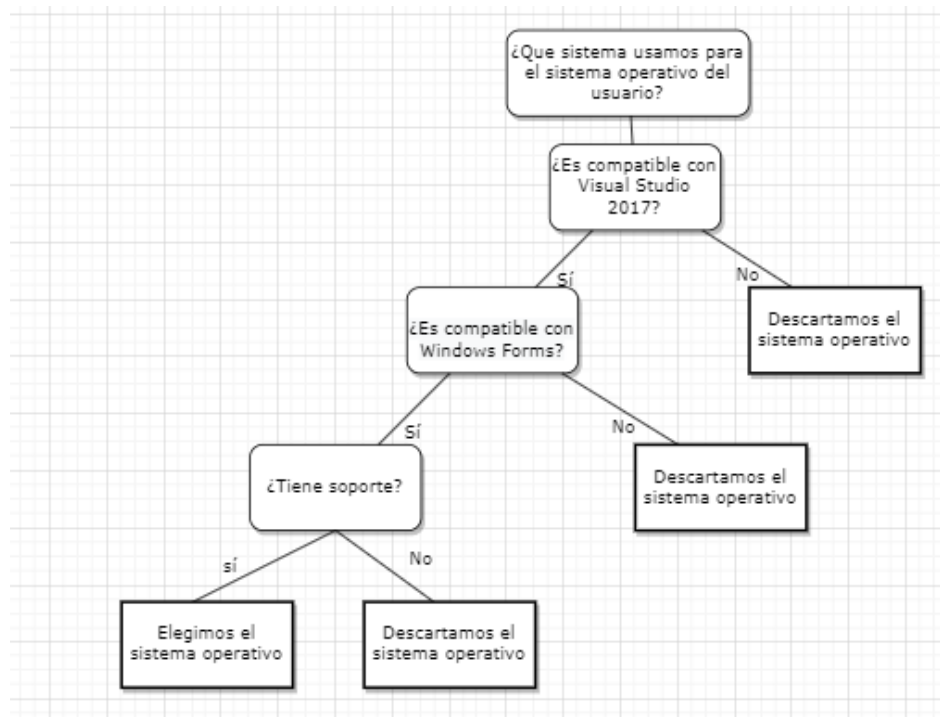
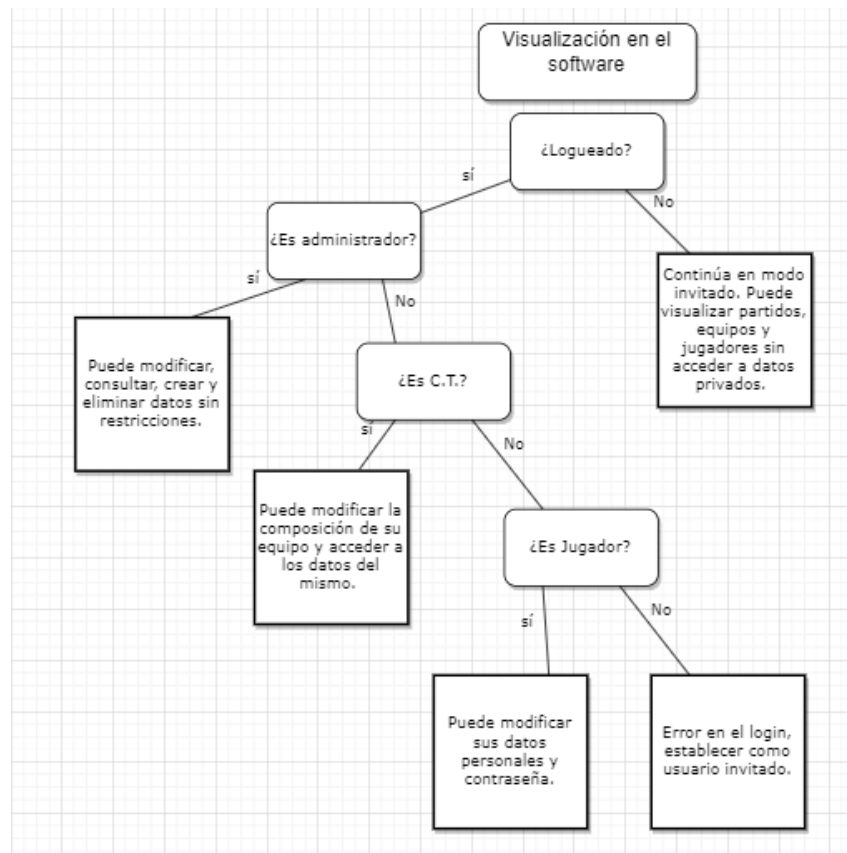


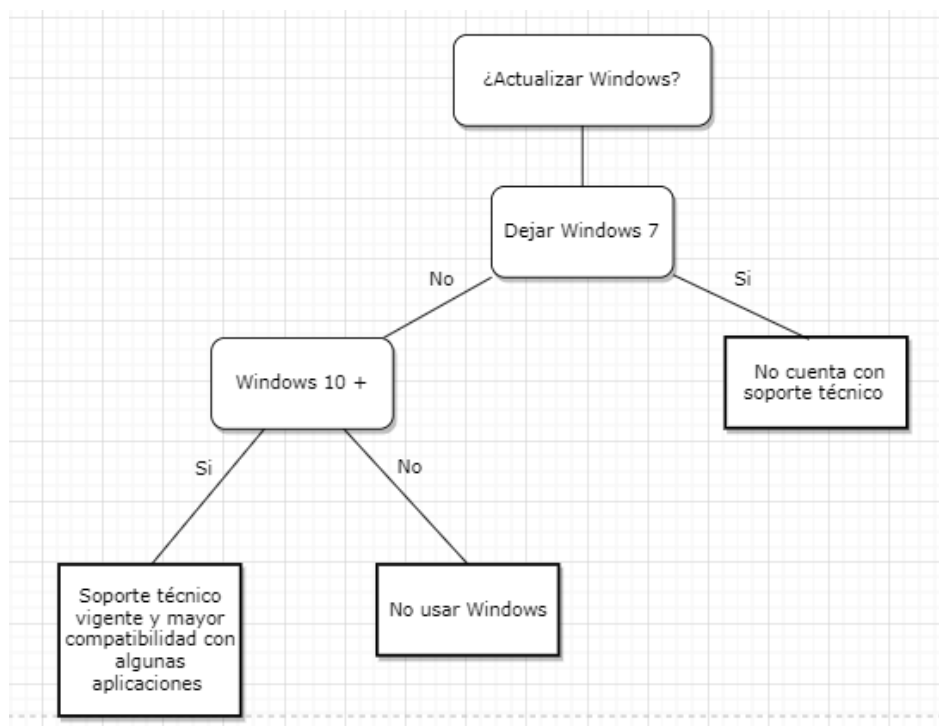
2.2- ANÁLISIS F.O.D.A.:

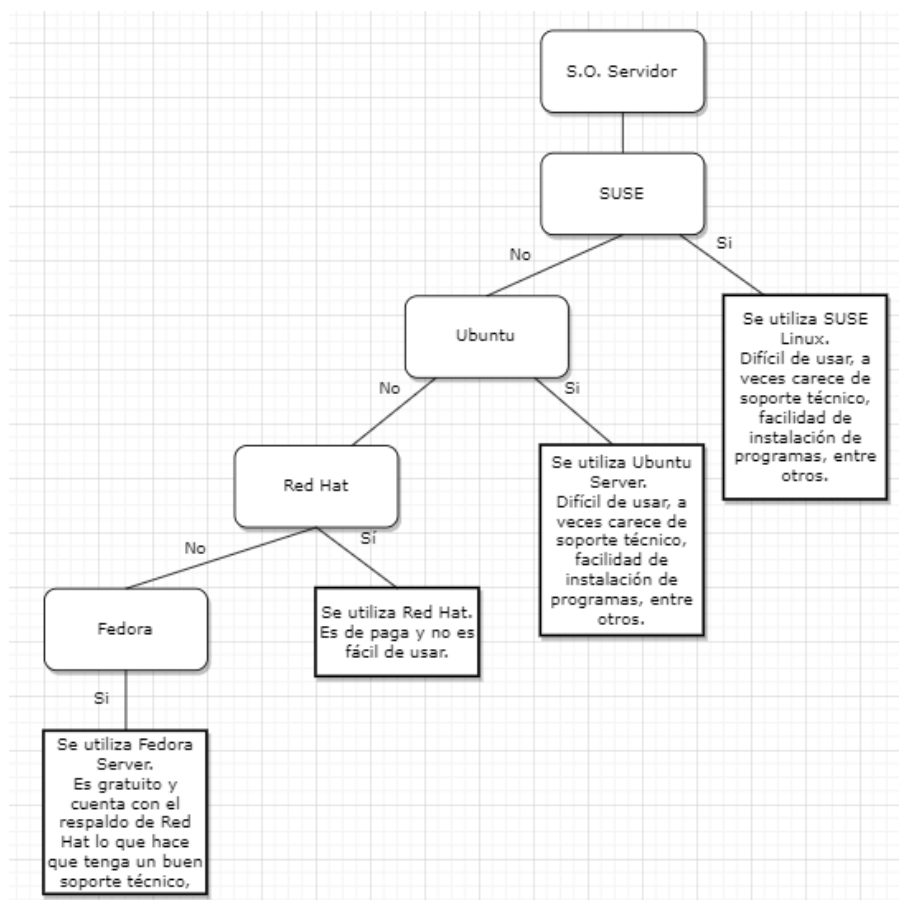
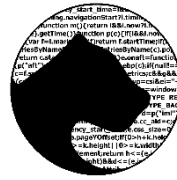
Fortalezas:	Debilidades:
<ul style="list-style-type: none"> - Horarios flexibles - Alta motivación por parte de los miembros - Creatividad 	<ul style="list-style-type: none"> - Otras responsabilidades fuera de la empresa - Poca experiencia - Escasez de personal - Poco conocimiento empresarial
Oportunidades:	Amenazas:
<ul style="list-style-type: none"> - Desarrollar un negocio estable - Creciente aumento en la demanda de aplicaciones de gestión. - Generar experiencia para el trabajo 	<ul style="list-style-type: none"> - Otras empresas en el mismo rubro - Recursos tecnológicos limitados tanto en software como hardware



2.3- LÓGICA DEL SISTEMA:









2.4- ESTUDIO DE FACTIBILIDADES:

Para el éxito del proyecto, es necesario determinar un estudio de factibilidad, el cual es uno de los resultados más esperados e importantes en la etapa del anteproyecto. Para ello, dicho estudio estará dividido en 6 particiones: factibilidad técnica, operativa, económica, comercial, legal y de cronograma. En este caso solo estudiaremos cuatro, la factibilidad técnica, operativa, económica y la legal.

2.4.1- FACTIBILIDAD TÉCNICA:

Esta factibilidad pretende establecer si el sistema propuesto puede desarrollarse con los recursos técnicos con los que cuenta el equipo de desarrollo; esto se hace considerando la disponibilidad de los recursos existentes (hardware, software y recurso humano), es decir, la existencia de tecnología y conocimiento necesarios para establecer que sea factible técnicamente el desarrollo del proyecto.

2.4.1.1- SISTEMA OPERATIVO PARA EL DESARROLLO:

Este elemento debe cumplir con las características de estabilidad, velocidad, facilidad de uso, seguridad y compatibilidad tanto con Visual Studio 2017 como con Windows Forms para soportar el correcto desarrollo del sistema informático, además de brindar seguridad a los usuarios.

Se presentan a continuación diferentes sistemas operativos que cumplen con las características necesarias e indispensables para el desarrollo del sistema propuesto:

- Windows 10 versión 1507 o posteriores: Home, Professional, Education y Enterprise (LTSC y S no se admiten).
- Windows Server 2016: Standard y Datacenter.
- Windows 8.1 (con la actualización 2919355): Core, Professional y Enterprise.
- Windows Server 2012 R2 (con la actualización 2919355): Essentials, Standard y Datacenter.



- Windows 7 SP1 (con las actualizaciones más recientes de Windows): Home Premium, Professional, Enterprise y Ultimate.

2.4.1.2- LENGUAJE DE DESARROLLO:

Es necesario que el lenguaje de desarrollo cumpla con las siguientes características:

- Compatibilidad con el entorno de desarrollo integrado para Windows y macOS Visual Studio 2017.
- Sintaxis sencilla.
- Integración con otros lenguajes.
- Un mínimo de bibliotecas disponibles para usar.
- Compatibilidad con Windows Forms.

Los lenguajes de desarrollo que cumplen con las características anteriormente mencionadas son:

- C#
- C++

2.4.1.3- SISTEMA GESTOR DE BASE DE DATOS:

Determinará la manera en la que se guarda la información, la velocidad de procesamiento de datos, respaldo de los mismos y la seguridad.

El sistema debe cumplir los siguientes requisitos:

- Seguro
- Escalable
- Estable
- Gratuito
- Veloz

En relación a lo descrito anteriormente se presentan las bases de datos que logran cumplir los requisitos.

- MySQL (La versión Community Edition es gratuita pero no es recomendable para su uso en una empresa)



- SQL Server (La versión 2019 Express es gratuita pero no es escalable)
- Maria DB

2.4.1.4- SISTEMA OPERATIVO DEL SERVIDOR

Debe poder soportar la instalación del sistema informático, satisfacer los requisitos de:

- Estabilidad
- Correcta administración
- Velocidad (por ejemplo, de conexión a las bases de datos)
- Facilidad de uso
- Seguridad de y a los usuarios
- Multiusuario
- Escalabilidad

A continuación se presenta una lista de S.O. que cumplen las características requeridas.

- Red Hat Linux.
- Fedora Server.
- Windows Server.
- CentOS.

2.4.1.5- HARDWARE PARA EL DESARROLLO:

El buen estado y funcionamiento de los equipos es necesario para lograr de manera correcta el desarrollo del sistema propuesto, por lo tanto a continuación se definen los requisitos mínimos y del entorno de desarrollo Visual Studio 2017 (datos oficiales de Microsoft):

- Procesador de 1,8 GHz o superior. Doble núcleo o superior recomendado.
- 2 GB de RAM; 4 GB de RAM recomendado
- Espacio en disco duro: hasta 130 GB de espacio disponible, en función de las características instaladas; las instalaciones típicas requieren entre 20 y 50 GB de espacio libre.
- Velocidad del disco duro: para mejorar el rendimiento, instale Windows y Visual Studio en una unidad de estado sólido (SSD).



- Tarjeta de vídeo que admita una resolución de pantalla mínima de 720p (1280 x 720); Visual Studio funcionará mejor con una resolución de WXGA (1366 x 768) o superior.

Se presentan a continuación el hardware con el que se cuenta para el desarrollo:

PC 1	CPU:	Intel(R) Core (TM)i5-3330 CPU-3.00GHz
	RAM:	4096 MB RAM
	Almacenamiento:	Disco Duro 600GB
	GPU:	Intel(R) HD Graphics
PC 2	CPU:	Intel(R) Core (TM)i5-3330 CPU-3.00GHz
	RAM:	4096 MB RAM
	Almacenamiento:	Disco Duro 600GB
	GPU:	Intel(R) HD Graphics
PC 3	CPU:	Intel(R) Core (TM)i5-4570 CPU-3.00GHz
	RAM:	8196 MB RAM
	Almacenamiento:	Disco Duro 1TB
	GPU:	Intel(R) HD Graphics 4600
PC 4	CPU:	Intel(R) Core (TM)i5-4570 CPU-3.00GHz
	RAM:	8196 MB RAM
	Almacenamiento:	Disco Duro 1TB
	GPU:	Intel(R) HD Graphics 4600



2.4.1.6- RECURSO HUMANO:

Esta factibilidad determinará en forma de lista los diferentes conocimientos necesarios para el correcto desarrollo del sistema:

Experiencia en:

- Administración de proyectos informáticos.
- Programando sistemas orientados a objetos.
- Desarrollo de sistemas.
- Análisis y diseño de sistemas.

Conocimientos:

- Lenguajes de programación.
- Manejo de bases SQL.

2.4.1.7-CONCLUSION:

Se cuenta con los requisitos necesarios para la elaboración del proyecto tanto en hardware como en software o recurso humano.

2.4.2- FACTIBILIDAD OPERATIVA:

Para esta factibilidad se estudia los posibles problemas que puedan tener los usuarios o el cliente con el sistema, es decir, si utilizaran el sistema correctamente una vez esté en uso. Por esto, la empresa, una vez esté terminado, presentará un pequeño curso de capacitación para toda persona que lo necesite, además contará con un manual de uso para las personas que no quieran participar del curso y también una ayuda técnica, todo esto para lograr comprender y hacer un uso correcto del sistema.

2.4.3- FACTIBILIDAD ECONÓMICA:

En esta factibilidad se hace un análisis de la relación de costo-beneficio del proyecto. En este proyecto, en cuanto a lo económico, se espera que lo invertido en y los costos no superen a los beneficios y ganancias que se obtengan por el



sistema, y que al menos logre un porcentaje de ganancia del 50% . Esto siempre y cuando se cumpla con los tiempos establecidos por el cliente.

2.4.5- FACTIBILIDAD LEGAL:

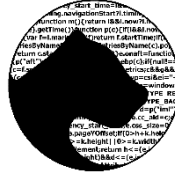
Para esta factibilidad se realiza un análisis y evaluación para saber si el proyecto cumple con las normativas y marcos legales. En este caso se debe tener en cuenta las licencias del sistema operativo que va a ser implementado, además de cumplir con las leyes internacionales de protección de datos personales y las nacionales de delitos informáticos.



2.5- DEFINICIÓN DE ROLES DE USUARIO:

Para el S.I.G.D se contará con los siguientes roles de usuarios:

- **Rol Invitado:** El rol invitado solo permite al usuario tener un modo “solo lectura” del sistema, existirán datos a los que no tendrá permitido acceder ya que será necesario ingresar a una cuenta para obtenerlos.
- **Rol Administrador:** Permite a un usuario con este rol crear y modificar equipos y usuarios, así como dar de baja a ambos, todo esto sobre las capacidades del invitado.
- **Rol Árbitro:** Le permitirá ver la información de la página, ingresar las incidencias a lo largo del partido y modificar algunos de sus propios datos personales.
- **Rol Jugador:** Tiene permitido editar su información personal pero no sus estadísticas además de los permisos básicos de invitado.
- **Rol C.T.:** Le permitirá visualizar y modificar los datos de sus jugadores, además de tener los permisos básicos de invitado.



2.6- PLANIFICACIÓN:

2.6.1- SCRUMBAN

Scrumban, como su nombre indica, es la combinación de dos metodologías de organización distintas, Kanban y Scrum.

Kanban es una metodología que divide un trabajo en pequeñas partes que pasan por varios estados: historias de usuario (Stories), por hacer (To Do), en progreso (In Progress), en proceso de análisis y prueba (Testing) y listo (Done).

El modelo Kanban aplicado al Scrum permite visualizar el progreso dentro de cada Sprint de manera accesible y dinámica utilizando aplicaciones como Miro, Trello o usando un pizarrón real.

2.6.2- IMPLEMENTACIÓN:

La forma elegida por el grupo para el uso de Scrumban es el registro de todas las capacidades del sistema en historias de usuarios. Usando Miro como pizarra elegida, se obtiene este resultado:

<https://miro.com/app/board/uXjVOvU9cB4=/>



2.6.3- CONFORMACIÓN DEL SCRUM:

Panthercode está compuesta por cuatro miembros que actúan tanto como dueños como fuerza de trabajo. Estos miembros son:

- Santiago Cardozo.
- Rainer Ramos.
- Nicolás Pacheco.
- Juan Pedro Irigoin.

A su vez cada integrante presenta un rol funcional de organización en la empresa pero que no representa superioridad o responsabilidad sobre otros. Estos roles son:

- Coordinador: Juan Pedro Irigoin.
- Subcoordinador: Nicolás Pacheco.
- Integrante 1: Santiago Cardozo.
- Integrante 2: Rainer Ramos.

Dentro de Scrum los roles de los miembros son:

- **Product Owner:** Profesores del área específica del Bachillerato Informático de EMT junto a los Profesores de Sociología, Inglés y Formación Empresarial.
- **Scrum Master:** Profesores de A.D.A, Gestión de Proyecto y el Coordinador de Proyecto.
- **Development Team:**
 - Santiago Cardozo.
 - Rainer Ramos.
 - Nicolás Pacheco.
 - Juan Pedro Irigoin.



2.7- ESPECIFICACIÓN DE REQUERIMIENTOS:

2.7.1- REQUERIMIENTOS FUNCIONALES:

- El usuario deberá ingresar credenciales para ingresar a su perfil en el sistema.
- El sistema permitirá a los usuarios con rol de Árbitro ingresar las incidencias de los partidos.
- El sistema permitirá a los usuarios con rol de Administrador ingresar las anotaciones, sanciones y las incidencias, ingresadas anteriormente por el usuario Árbitro, a la base de datos
- El sistema permitirá la modificación, alta y baja de los equipos.
- El sistema permitirá la modificación, alta y baja de los jugadores.
- El sistema permitirá al usuario de rol de Administrador crear usuarios. Les brindará un rol a cada usuario dependiendo de su función y con sus respectivos permisos.
- Existirá un rol de invitado para permitir la visualización de los datos de equipos o de un jugador en base a su cédula, además de datos generales de las competencias.
- El sistema permitirá elegir el deporte el cual desea el usuario visualizar la información. Estos deportes son, por el momento: fútbol, básquetbol y handball.
- A cada equipo se le asignará un identificador único, que será utilizado para identificarlos en los procesos que se realicen sobre ellos.
- El sistema permitirá a los usuarios con rol de CT modificar sus equipos asignados.
- El sistema avisará a los usuarios CT los jugadores que estén inhabilitados para próximos encuentros.
- El sistema permitirá guardar información de los partidos en cualquier deporte. Esta información incluye: resultados, participantes, lugar, fecha, etc.



2.7.2- REQUERIMIENTOS NO FUNCIONALES:

- El sistema tendrá la opción de elegir el idioma. Los idiomas serán inglés y español.
- El usuario tendrá 3 intentos para ingresar su contraseña, una vez utilizados los tres intentos, el usuario tendrá que esperar 30s para volver a ingresar la contraseña de nuevo. Si el usuario no recuerda su contraseña le puede pedir ayuda a un superior para resetear su contraseña.
- El sistema presentará un apartado de ayuda donde el usuario escribirá sus problemas, dudas, etc. a la bandeja de entrada de un mail de contacto de la empresa, y dentro de un lapso de tiempo determinado se le solucionara o responderá su mensaje.
- El sistema guardará los datos de los usuarios en la base de datos.
- El sistema proporcionará validación de datos al momento de iniciar sesión para controlar o restringir el acceso a diferentes funciones del software, dependiendo el rol del usuario.
- El diseño del sistema será amigable e intuitivo con el usuario, facilitando la navegación y la interacción con el mismo.



2.7.3- ALCANCE:

El Sistema Informático de Gestión Deportiva (S.I.G.D), a desarrollarse; proveerá la función de gestionar, controlar y brindar un seguimiento a las futuras actividades deportivas que se realicen en la escuela o en cualquier otra institución. Dichos deportes serán fútbol, basketball y handball, además de permitir agregar otro deporte al gusto de los clientes.

2.7.4- LIMITACIONES:

Software: S.O windows 10, MySQL 8.0.29, Fedora Server 36 , Microsoft Visual Studio 2017, GitHub.com.

Tiempo: 8 de noviembre (6 meses y 6 días).

Costo: Licencia de MySQL Enterprise.

Estándares: Normas ISO, IEEE y CMM.



3- ELECCIÓN CICLO DE VIDA:

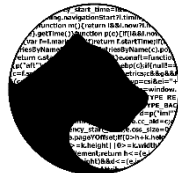
El ciclo de vida de un proyecto, más específicamente el modelo de ciclo de vida, se refiere a la forma en que se estructuran las distintas fases de un proyecto en el tiempo. Cada fase con su propósito y resultado, con el objetivo de llegar al mejor resultado posible de la manera que mejor complementa al proyecto, en este caso un desarrollo de software.

Dentro de los distintos ciclos de vida que son usados hoy en día, el grupo decidió utilizar una mezcla entre el Modelo Incremental y Scrum.

El Modelo de Ciclo de Vida Incremental se estructura de forma que el desarrollo crea, como su nombre indica "Incrementos". Estos incrementos son fases de desarrollo donde cada una actúa como su propio ciclo de vida, trabajando sobre distintas subdivisiones del sistema, a veces sobre los resultados de incrementos anteriores. Esto culmina juntando todas las funcionalidades del software para llegar a un producto terminado en mayor velocidad que lo que podrían conseguir otros modelos.

Scrum, por su parte, se estructura en forma de "Sprints", cortos incrementos en donde se trabaja sobre lo que hay que hacer, obteniendo conocimientos para utilizar en el siguiente incremento. Estos Sprints comienzan en el Product Backlog que engloba todo lo que debe realizarse para completar el proyecto, luego se planifica un Sprint Backlog, es decir, todo lo se planea hacer durante el periodo de tiempo del Sprint. Al final del Sprint se llega a la Revisión y a la Retrospectiva del Sprint, la revisión siendo el análisis mencionado previamente que permite ver a que se llegó y la retrospectiva siendo las formas en que se puede mejorar el desarrollo.

El plan es utilizar el Modelo Incremental para planificar la estructura macro del proyecto y utilizar Scrum para el trabajo diario en cada incremento. El optar por esto en este proyecto viene de las cualidades de tanto el software como de los desarrolladores. El sistema a desarrollar es uno relativamente complejo, pero que está formado por un gran número de pequeñas funcionalidades que individualmente



son a la vez fáciles de planificar y fáciles de desarrollar. Por esto, el grupo encontró esta forma más efectiva y cómoda porque, en términos de tiempo y responsabilidades externas al proyecto, resulta más sencillo trabajar en periodos cortos de trabajo intenso, con resultados visibles, que al combinarse resultan en avances grandes para el conjunto.

4-CASOS DE USO DE FORMA NARRATIVA:

Nombre:	Ingresar un usuario	
Autor:	Santiago Cardozo	
Fecha	23/5/2022	
Descripción:	Permite a un usuario loguearse al sistema	
Actores:	Usuario invitado/no logueado	
Precondiciones:	1. Que esté registrado 2. Tener el software instalado.	
Flujo Normal:	1. El actor pulsa el botón para ingresar. 2. El sistema muestra varias cajas de texto para introducir CI y contraseña. 3. El actor ingresa todos los datos correspondientes. 4. El sistema comprueba la validez de los datos y que existan en la base de datos.	
Flujo Alternativo:	4.1 El sistema comprueba la validez de los datos, si los datos no son correctos, se avisa al actor de ellos permitiéndole que los corrija. 4.2 El sistema comprueba que los datos si no existen en la base de datos, se avisa al actor de ello permitiéndole registrar o dar la opción de contactar a soporte o a un administrador.	
Flujo de excepción:	1. Se cortó la comunicación con el servidor. No se loguea el usuario. 2. Las credenciales del usuario no son admitidas por el sistema.	
Poscondiciones:	El usuario ha sido logueado en el sistema.	



Nombre:	Asignar un jugador	
Autor:	Santiago Cardozo	
Fecha	23/5/2022	
Descripción:	Permite al D.D oCT asignar un jugador en su equipo	
Actores:	Usuario C.T(Cuerpo Técnico).	
Precondiciones:	1.Tener el software instalado 2.Que esté logueado en el sistema 3.Que el usuario tenga rol de C.T	
Flujo Normal:	1.El actor pulsa el botón para agregar jugador al equipo. 2.El sistema muestra varias cajas de texto para introducir CI del jugador y posición que se le asignará. 3.El actor ingresa todos los datos correspondientes. 4.El sistema comprueba la validez de los datos, que el usuario esté registrado en la base de datos y sea un jugador.	
Flujo Alternativo:	4.1 El sistema comprueba la validez de los datos, si los datos no son correctos, se avisa al actor de ellos permitiéndole que los corrija. 4.2 El sistema comprueba que los datos existan en la base de datos, en caso de que no existan o no sea un jugador se avisa al actor de ello permitiéndole cambiar los datos o dar la opción de contactar a soporte o a un administrador.	
Flujo de excepción:	1.Se cortó la comunicación con el servidor. No se asigna el jugador al equipo. 2.Las credenciales del usuario no son admitidas por el sistema. 3.El usuario no tiene permisos para realizar la asignación del jugador.	
Poscondiciones:	El usuario ha asignado un jugador a su equipo.	



Nombre:	Registrar un usuario	
Autor:	Juan Pedro Irigoin	
Fecha	7/7/2022	
Descripción:	Permite crear un usuario en el software	
Actores:	Administrador y Usuario no Ingresado	
Precondiciones:	1.Tener el software instalado 2.Que el usuario no esté ingresado en el sistema	
Flujo Normal:	1.El administrador pulsa el botón para registrar. 2.El sistema muestra varias cajas de texto para introducir nombres, apellidos, contraseña, mail, CI, teléfono y rol del usuario a ingresar. 3.El actor ingresa todos los datos correspondientes. 4.El sistema comprueba la validez de los datos y los almacena en la base de datos.	
Flujo Alternativo:	4.1 El sistema comprueba la validez de los datos, si los datos no son correctos, se avisa al actor de ellos permitiéndole que los corrija. 4.2 El sistema comprueba que los datos ya existen en la base de datos, se avisa al actor de ello permitiéndole cambiar los datos o contactar a soporte.	
Flujo de excepción:	1.Se cortó la comunicación con el servidor. No se registra el usuario. 2.Las credenciales del usuario no son admitidas por el sistema.	
Poscondiciones:	El usuario ha sido registrado en el sistema.	



Nombre:	Eliminar un Usuario	
Autor:	Juan Pedro Irigoin	
Fecha	7/7/2022	
Descripción:	Permite borrar un usuario en el software	
Actores:	Administrador y Usuario Ingresado	
Precondiciones:	1.Tener el software instalado 2.Que el usuario esté ingresado en el sistema	
Flujo Normal:	1.El administrador pulsa el botón para eliminar un usuario. 2.El sistema muestra una caja de texto para introducir la CI del usuario a eliminar. 3.El actor ingresa todos los datos correspondientes. 4.El sistema busca el usuario de la CI correspondiente en la base de datos. 5.El sistema elimina los datos de ese usuario.	
Flujo Alternativo:	4.1 El sistema comprueba la validez de la CI y busca el usuario correspondiente en la base de datos. 4.2 El sistema comprueba que la CI no está registrada en la base de datos, se avisa al actor de ello permitiéndole cambiar los datos o contactar a soporte.	
Flujo de excepción:	1.Se cortó la comunicación con el servidor. No se elimina el usuario. 2.Las credenciales del usuario no son admitidas por el sistema.	
Poscondiciones:	El usuario ha sido borrado del sistema.	

Nombre:	Cierre de sesión	
Autor:	Santiago Cardozo	
Fecha	18/7/2022	
Descripción:	Permite cerrar la sesión del usuario en el software	
Actores:	Usuario Ingresado	
Precondiciones:	1.Tener el software instalado 2.Que el usuario esté ingresado en el sistema	
Flujo Normal:	1.El Usuario pulsa el botón para cerrar sesión. 2.El sistema cierra su sesión en el sistema.	
Flujo Alternativo:	2.1 El sistema no puede cerrar su sesión, sale un mensaje diciendo inténtelo nuevamente	
Flujo de excepción:	1.Se cortó la comunicación con el servidor. No se registra el usuario.	
Poscondiciones:	El usuario ha cerrado su sesión del sistema.	



Nombre:	Registrar un equipo	
Autor:	Santiago Cardozo	
Fecha	18/7/2022	
Descripción:	Permite crear un equipo en el software	
Actores:	Administrador	
Precondiciones:	1. Tener el software instalado 2. Que el equipo no esté ingresado en el sistema	
Flujo Normal:	1. El administrador pulsa el botón para registrar. 2. El sistema muestra varias cajas de texto para introducir nombre del equipo, deporte, jugadores y C.T. 3. El actor ingresa todos los datos correspondientes. 4. El sistema comprueba la validez de los datos y los almacena en la base de datos.	
Flujo Alternativo:	4.1 El sistema comprueba la validez de los datos, si los datos no son correctos, se avisa al actor de ellos permitiéndole que los corrija. 4.2 El sistema comprueba que los datos ya existen en la base de datos, se avisa al actor de ello permitiéndole cambiar los datos o contactar a soporte.	
Flujo de excepción:	1. Se cortó la comunicación con el servidor. No se registra el usuario. 2. Las credenciales del equipo no son admitidas por el sistema.	
Poscondiciones:	El equipo ha sido registrado en el sistema.	



Nombre:	Eliminar un Equipo	
Autor:	Santiago Cardozo	
Fecha	18/7/2022	
Descripción:	Permite borrar un equipo en el software	
Actores:	Administrador	
Precondiciones:	1.Tener el software instalado 2.Que el equipo esté ingresado en el sistema	
Flujo Normal:	1.El administrador pulsa el botón para eliminar un usuario. 2.El sistema muestra una caja de texto para introducir la id del equipo a eliminar. 3.El actor ingresa todos los datos correspondientes. 4.El sistema busca el equipo de la id correspondiente en la base de datos. 5.El sistema elimina los datos de ese equipo.	
Flujo Alternativo:	4.1 El sistema comprueba la validez de la id y busca el usuario correspondiente en la base de datos. 4.2 El sistema comprueba que la id no está registrada en la base de datos, se avisa al actor de ello permitiéndole cambiar los datos o contactar a soporte.	
Flujo de excepción:	1.Se cortó la comunicación con el servidor. No se elimina el equipo. 2.El id del equipo no son admitidas por el sistema.	
Poscondiciones:	El equipo ha sido borrado del sistema.	