Instituto Tecnológico Colonia CTC

Obligatorio Ingeniería de Software

Matías Erramouspe

Bruno Ríos

2022

Tutores: Andrés Klett y Gustavo Erramouspe

Contenido

[Abstract 4](#_Toc120651912)

[Introducción 5](#_Toc120651913)

[Presentación del Cliente 6](#_Toc120651914)

[Presentación del Problema 7](#_Toc120651915)

[Actores involucrados 8](#_Toc120651916)

[Lista de necesidades 9](#_Toc120651917)

[Lista de Objetivos 9](#_Toc120651918)

[Lista de Requerimientos 9](#_Toc120651919)

[Funcionales 9](#_Toc120651920)

[No Funcionales 10](#_Toc120651921)

[Descripción del entorno 10](#_Toc120651922)

[Análisis de alternativas 12](#_Toc120651923)

[Estudio de factibilidad 13](#_Toc120651924)

[Integrantes y roles 14](#_Toc120651925)

[Descripción y selección de herramientas 15](#_Toc120651926)

[Plan de SQA 16](#_Toc120651927)

[Estándares definidos y convenciones 16](#_Toc120651928)

[Documentación 16](#_Toc120651929)

[Diagrama de clases 17](#_Toc120651930)

[Base de datos 17](#_Toc120651931)

[Casos de Uso 17](#_Toc120651932)

[Desarrollo 17](#_Toc120651933)

[Interfaz de usuario 17](#_Toc120651934)

[Plan de Testing 18](#_Toc120651935)

[Pruebas de caja negra 19](#_Toc120651936)

[Plan de SCM 20](#_Toc120651937)

[Plan de Capacitación 21](#_Toc120651938)

[Elección de metodología 22](#_Toc120651939)

[Modelo espiral 23](#_Toc120651940)

[Metodología Scrum 23](#_Toc120651941)

[Ciclo de vida elegido 24](#_Toc120651942)

[Etapas del ciclo de vida elegido 25](#_Toc120651943)

[Iteraciones de implementación definidas 25](#_Toc120651944)

[Iteración Nro. 1: 25](#_Toc120651945)

[Iteración Nro. 2: 26](#_Toc120651946)

[Iteración Nro. 3: 26](#_Toc120651947)

[Cronograma de trabajo y criticidad 26](#_Toc120651948)

[Diagramas de Grantt 26](#_Toc120651949)

[Diagrama de Grannt ruta critica 27](#_Toc120651950)

# Abstract

El Sistema de administración de seguros es un sistema desarrollado en C# + Windows Form guardando los datos en una planilla de Excel y guardando los datos en la nube. Este sistema da su origen en Erramouspe Seguros para mejorar la organización de los datos de sus Clientes.

El programa consiste de un ejecutable que se instala en el lugar a trabajar, el cual permite ingresar agregar, eliminar, modificar los datos de los clientes, estos datos se proceden a guardar en planilla de Excel que se subirá a Google Drive. Además, el programa permitirá guardar automáticamente los datos de clientes enviados por las compañías. Aparte de eso el programa permitirá organizar los clientes por medio de distintas consultas, ejemplo: número de póliza, apellido, fecha asegurada, etc. También el programa contará con un sistema de aviso de renovación el cual avisará tanto a los corredores como al cliente su fecha de renovación 15 días antes, este aviso se realizará por WhatsApp a los clientes y por Email a los corredores, asi evitando la tarea manual de avisar a cada cliente cuando corresponda.

La interfaz será simple y contará con imágenes de las distintas compañías, las cuales al darle click mostrará los clientes asegurados con esas compañías.

# Introducción

El cliente para el que se realiza el sistema es Erramouspe Seguros una Empresa que se dedica a la venta de seguros de distintas compañías. Opera en Rosario, Nueva Helvecia Y el Ecilda Paullier. Esta empresa vende seguros automovilísticos, de hogar, y de vida.

Este Sistema se ejecutará en la oficina de Rosario dado que es la oficina más grande y relevante, dado que en la cual concurren una mayor cantidad de clientes.

El programa se centraría en el almacenamiento de los datos de los clientes, y la notificación de renovación de la póliza con 15 días de antelación. Esto permitirá optimizar los tiempos de notificación de las póliza y los tiempos de almacenaje de datos local, además de tener toda la información en una sola aplicación, en vez de ir a las distintas paginas Web de cada compañía.

El programa permitirá leer planillas de Excel con los seguros que se están por renovar, si los clientes ya están ingresados en el programa no se guardaran los datos, solo se actualiza la fecha de renovación, si el cliente no existe se crea un nuevo cliente en la planilla de Excel. El programa busca no solo leer datos de Excel si no también llegar a leer los datos de las distintas páginas de las compañías sin necesidad de ir uno por uno.

# Presentación del Cliente

Erramouspe Seguros, es una empresa con una larga trayectoria de experiencia en el mercado asegurador, desarrollando la actividad en todo el país desde hace más de 15 años, esta se define como una empresa dedicada a brindar una gestión completa e integral para contratar la mejor opción en seguros, adaptados a la exigencia digital y a las necesidades del mercado asegurador actual, con la responsabilidad y seriedad que se necesita para proteger su patrimonio.

Esta representa a las principales aseguradoras del mercado, con absoluta independencia y total objetividad, ellos asesoraran de forma responsable para encontrar la opción más conveniente.

Cuentan con 3 oficinas ubicadas en los Departamento de Colonia y San José. En ellas encontrará el asesoramiento que necesita y el profesionalismo que merece.

# Presentación del Problema

Actualmente nuestro cliente cuenta con los siguientes problemas. Uno de los problemas es la organización de los datos de los clientes de cada compañía, dado que por cada una de ellas cuentan con su propia pagina web con los datos de sus clientes, y la manera que tienen ahora de organizarse es descargar en Excel los datos de Porto Seguros, agregar los datos de las otras compañías a este Excel y subirlos a una Nube online. Este problema impide la clara y rápida organización de los datos, el programa busca no solo leer datos de Excel si no también llegar a leer los datos de las distintas páginas de las compañías sin necesidad de ir uno por uno. Otro problema es la notificación de la renovación de la póliza para cada cliente, actualmente se realiza de manera manual, pero esto es una mala optimización de los tiempos, el programa buscara notificara de manera automática a cada cliente, 15 días antes de su renovación de póliza.

# Actores involucrados

Aseguradoras: Empresas destinadas a la venta de pólizas, son los que proveen los servicios al sistema. Las aseguradoras enviaran los datos de los clientes con póliza a renovar y el sistema se encargará de guardar los datos de los clientes a su aseguradora correspondiente.

Corredor de Seguros: Es el encargado de modificar agregar eliminar los datos en el sistema, a su vez es el que les vende los seguros a los clientes, para luego mandárselo a las aseguradoras.

Clientes: Los clientes son los que se dirigen al corredor de seguros para comprar una póliza, sus datos, son enviados a las aseguradoras para luego ser guardados en el sistema.

# Lista de necesidades

* Ordenar toda la información de los clientes en un solo lugar de manera automática.
* Notificar a los clientes la renovación de su póliza con 15 días de antelación.

# Lista de Objetivos

* Mejorar el maneja de la información de cada cliente.
* Optimizar los tiempos a la hora de guardar la información de los clientes y notificación. a la hora del vencimiento de una póliza.
* Notificar a todos los clientes con 15 días de antelación a su renovación de póliza.

# Lista de Requerimientos

## Funcionales

* Poder agregar, baja y modificar clientes en el sistema.
* El sistema podrá listar los datos por empresas aseguradoras.
* El sistema podrá guardar los datos en un Excel.
* El programa deberá notificar a los clientes con 15 días de antelación a su renovación de póliza.
* El programa podrá leer y guardar los datos de los clientes de las distintas páginas web automáticamente.
* El sistema podrá realizar búsquedas de clientes personalizadas (Consultas), por Ejemplo: mostrar todos los nombres similares a “Ju”.
* El sistema permitirá ordenar por nombres, empresa, fecha, apellido, etc.
* El sistema deberá poder leer una plantilla de datos de Excel con los clientes y poder guardarlos en una planilla de Excel ubicado en Google Drive, si el cliente ya existe no se guardan sus datos.

## No Funcionales

* El sistema deberá realizarse en C# Windows Form
* El programa utilizara como base de datos una plantilla de Excel que se guardara en la nube.
* El sistema deberá correr en Windows 8, 10 y 11.
* El sistema notifica a los clientes atreves de WhatsApp o mensajes de texto.

# Descripción del entorno

La descripción a que se muestra a continuación muestra el proceso desde la adquisición de un automóvil hasta el aseguramiento de el mismo, aparte de las contribuciones del programa a este proceso.

Primero el cliente tiene que adquirir un auto, según el artículo 1 de la ley 18412 , se enuncia que se prohíbe todo vehículo que no cuente como mínimo con un seguro contra terceros.

Luego el cliente contacta a la aseguradora, ya sea por vía presencial o online, una vez realizado lo anteriormente descrito, al cliente se le muestran los diferentes tipos de seguros lo que cuben y su costo despidiendo de cada empresa de seguros.

Como siguiente se le revisa al cliente el vehículo y pasa a pagar la póliza del seguro.

Una vez realizada la venta, el/la encargado de realizar la venta, entra en la pagina de la empresa a la cual se vendió su seguro, se ingresan los datos del cliente y se envían.

Cada mes todas las empresas envían los datos de los clientes asegurados a esa aseguradora con la renovación de la póliza, y la empresa aseguradora se encarga de guardar los datos y avisar a los clientes cuando deben renovar su póliza. En el caso de Erramouspe Seguros la forma de guardar los datos es usar una plantilla de Excel de Porto Seguros y añadir manualmente la información de los clientes de las otras empresas, además la forma que tiene de avisar a sus clientes de la renovación de la póliza es de manera manual vía WhatsApp o vía mensajes de texto, el programa se involucrara a la hora de guardar los datos de los clientes de manera casi automática y la notificación de la renovación de la póliza a los clientes.

Como ultimo pasa del proceso es que el cliente vuelva a renovar o no su póliza.

# Análisis de alternativas

Una de las alternativas que se podría utilizar en el proyecto a la hora de guardar los clientes es que en vez de utilizar un Excel que se guarda en Google drive, usar Google sheets, pero se descartó la opción, dado que hay que pagar por un dominio para poder utilizar sus servicios, una mejor idea es utilizar una base local como por ejemplo SQL, pero el dilema que tiene utilizar esta idea es que es una gran pérdida de tiempo y dinero dado que se tiene que configurar el server de tal forma que no solo puedan modificar los datos de los clientes las computadoras de Rosario, sino también las demás computadoras de otras oficinas que están fuera de la red local, además que retrasaría mas el tiempo de desarrollo del sistema y además no trae un beneficio significativo en vez de usar un Excel que se guarde en Google Drive. Una alternativa para Google Drive es OneDrive pero aunque los planes de pago de OneDrive para empresas son superiores dado que dan 1TB y los servicios de Office por 12,50 USD, para la magnitud de este proyecto solo es necesario un par de GB con 15 GB que son los que ofrece Google Drive de manera gratuita son mas que suficientes y 3 veces mayor a la cantidad que ofrece OneDrive de manera gratuita.

Se utilizará Windows form sobre otra alternativa dado que es un método sencillo en el cual programar, permite mayor personalización del sistema, dado que de otra forma se tiene que utilizar los parámetros del navegador, además el cliente pidió un programa ejecutable, agregándole a lo anteriormente mencionado si se quiere desarrollar un programa que se ejecute en un navegador este tiene que estar subido en algún servidor o en otro lugar, lo cual eleva la dificultad del proyecto. Aparte de los puntos mencionados anteriormente los miembros del proyecto ya cuentan con experiencia desarrollando en Windows Form C#.

# Estudio de factibilidad

Para poder realizar este proyecto se debe contar con conocimiento sobre Windows form c#, Excel, Google Drive y como puede el programa guardar información en el Excel guardado en el Google Drive. Como Material necesario para la empresa es una o varias Computadoras conexión a internet, el Excel instalado, sino se puede utilizar como alternativa el Excel online en Office.Com. Los costos del proyecto son solo de desarrollo dado que los materiales anunciados anteriormente la empresa Erramouspe Seguros ya cuentan con los materiales descritos.

La propuesta a implementar no cuenta con ningún tipo de gasto fuera del de desarrollo, ya que el Google drive cuenta con 15gb de uso gratuito, aparte que el Excel si no lo tiene instalado lo pueden usar en Office.com. Aparte la empresa ya contaba con internet, computadoras y el Excel instalado por lo que no son gastos adicionales al proyecto. El lugar donde se desarrolla el sistema es en Windows Form, este es gratis y Open-source por lo que su licencia y su uso es totalmente gratis.

La tecnología con la cual se implementará el sistema es existente y de libre uso Windows form, por lo que no complica el desarrollo, este medio se adapta bien con la idea que el cliente tiene en mente. La tecnología que se usa para guardar los datos es un Excel en un Drive, siendo estos totalmente accesibles para cualquier usuario corriente, aunque Excel de manera normal es de pago, se puede utilizar la manera gratuita en Office.com.

El sistema va a ser utilizado por los oficinistas en cada sucursal de la aseguradora Erramouspe Seguros, le ahorrara el tiempo a la hora de ingresar los clientes y cuando tengan que notificar a los clientes la renovación de la póliza. Se sabe que el proyecto se va a utilizar dado que se realizo en base a las necesidades y herramientas descriptas por los oficinistas y por el dueño de la empresa. El proyecto con una buena organización y dirección puede tomar de desarrollo de 2 a 3 meses o menos. Con lo anteriormente descrito se llegó a la conclusión que el proyecto es totalmente factible.

# Integrantes y roles

Los miembros de este equipo son Matías Erramouspe y Bruno Ríos, estudiantes de la carrera de Analista Programador del Instituto CTC, con la tutela de los profesores Andrés Klett y Gustavo Erramouspe. Ambos estudiantes realizaran el diseño, desarrollo, análisis y testeos del sistema, además de realizar las tareas de documentación requeridas a este proyecto. Cada dos semanas el equipo se reunirá para informar de avances y reajustar las necesidades, tareas, e ideas para seguir el proyecto.

# Descripción y selección de herramientas

En base a lo mencionado anteriormente en el estudio de alternativas, para este proyecto se utilizarán las siguientes herramientas:

* Para la gestión del desarrollo se utilizará Microsoft Project, que es ideal para la construcción y seguimiento del cronograma del trabajo.
* Para el desarrollo del código backend y frontend se utilizará Visual Studio 2019, el frontend se desarrollará utilizando la herramienta Windows form C#, el backend se desarrollará con C#.
* Para el desarrollo y mantenimiento de la base de datos se utilizará una planilla de Excel que se subirá a una nube (Google Drive).
* Todo tipo de documentación se realizará en Microsoft Word. Y la generación de diagramas de clase y casos de uso se realizará en StarUML.

# Plan de SQA

El plan de SQA es crear estándares de calidad que se deben cumplir a lo largo del proyecto, ya sea en la documentación, código, o en los diagramas. Además de lo anteriormente mencionado también se encuentra el plan de Testing para poder asegurarse que el proyecto es de alta calidad

## Estándares definidos y convenciones

### Documentación

La documentación estará basada en los parámetros definidos por el Documento 302 de CTC “Normas para la presentación de trabajos finales de curso".

Para la mayor parte del documento se utilizará Calibri(cuerpo) 11 y para los títulos y subtítulos se utilizará Calibiri Light (titulo) pero de un tamaño mayor (16 y 13 respectivamente) y de color azul

Cada punto estará definido en una hoja o más, además cada hoja u hojas tendrá un punto propio, por ejemplo, cuando se termine de detallar el punto “Estudio de factibilidad” se deja en blanco lo restante de la hoja y se comienza a redactar el siguiente punto de documentación en la siguiente hoja.

Cada punto en el índice estará enlazado con un hipervínculo al página del documento con el punto mencionado en el índice.

### Diagrama de clases

Tanto el diagrama de clases como otros diagramas utilizados como ayuda se construirán usando el estándar de modelado UML

### Base de datos

En la plantilla de Excel cada columna estará in infinitivo, para aumentar la claridad

### Casos de Uso

Los nombres se utilizarán en infinitivo, los actores empiezan con mayúscula, además se tendrá un identificador que comienza con “CU\_” y el número de caso de uso en tres dígitos, por ejemplo, CU\_001.

### Desarrollo

A la hora de asignar el nombre de atributos y métodos se utilizará el estilo lowerCamelCase, los nombres de las clases empiezan con mayúscula y las interfaces tendrán como prefijo la letra “I”.

### Interfaz de usuario

Es fundamental que el programa siga la misma línea de diseño durante todo el programa por lo cual el equipo junto al usuario definirá los colores y estilos a utilizar, para cada formulario generado por el programa se utilizara el patrón de diseño Singleton.

Cuando el programa liste los distintos clientes de las empresas tiene que estar claro y legible cada atributo y a que empresa está asociado el cliente, por ejemplo, con una imagen de la empresa en cuestión en la esquina superior derecha.

A su vez, todas las pantallas tendrán una menú fijo, desde el cual se podrá acceder a todas las funcionalidades disponibles para el usuario. Como norma general, las pantallas deben respetar los principios básicos de usabilidad, siendo estas sencillas y de fácil navegabilidad para cualquier usuario.

# Plan de Testing

Para comprobar que la calidad del producto sea la mayor posible y evitar errores se realizara un Testing, en el cual se busca encontrar fallos en el proyecto, que no se encontraron anteriormente, estos Testing se realizaran regularmente para asegurar que no se escapara ningún error o que se generen nuevos con el código nuevo generado.

Todas las funciones y métodos deben ser puestos a prueba, conociendo de ante mano los parámetros que recibe y que parámetros devuelve. Se debe simular todas las posibles situaciones y usos del programa para poder otorgar una mayor fiabilidad al software desarrollado.

La documentación también será revisada durante el análisis y diseño del proyecto de esta forma reducir los errores en redacción en el documento, y los errores en la programación, además de disminuir las diferencias del código esperado por la documentación y el código que se desarrolló.

### Pruebas de caja negra

La prueba de caja negra es una forma de probar el software desde el punto de vista del usuario teniendo como objetivo encontrar errores. Se comprobarán las clases, las controladoras, el acceso a la base de datos (plantilla de Excel), los formularios, los abms, etc.

Luego de cada iteración se realizarán pruebas de caja negra, o de funcionamiento, para verificar que el sistema siga operando de forma correcta.

Para cada abm se generarán juegos de datos para probar la mayor cantidad de posibilidades de uso del sistema.

Para cada uno de los formularios:

1. Verificar el acceso al formulario.

a. Comprobar si se puede abrir, y como, probar desde todas las referencias.

b. Probar manejo de usuarios

c. Comprobar cómo se cierra, que pasa cuando redirige a otros formularios.

2. Verificar el correcto funcionamiento del formulario.

a. Testear botones, campos de texto, listas, validadores y demás elementos.

3. Verificar entrada de datos.

a. Verificar correcto funcionamiento de la validación frontend y backend.

b. Comprobar funcionamientos (ABM, búsquedas, filtros), verificando validaciones

backend. Esto se debe hacer en conjunto con el punto 4.

4. Verificar salida de datos

a. Probar listados, búsquedas y filtros. Esto se hace en conjunto con el punto 3.

Las pruebas se controlarán mediante una planilla electrónica, en donde se registrará el resultado de cada una de ellas, registrando:

• el número de prueba

• la fecha de la prueba

• la descripción de la prueba

• el resultado esperado

• el resultado obtenido

• la prueba que repite, si corresponden

• un indicador de si la prueba ha sido finalizada con un resultado aceptable

• módulo en que se realiza la prueba

• responsable de la prueba

# Plan de SCM

El fin del plan SCM es controlar los cambios del software. Para la gestión de cambios en la configuración se utilizará la herramienta online GitHub, donde habrá un repositorio en el que se actualizará semanalmente con los cambios realizados en el proyecto. Git permite llevar control de las modificaciones de configuración del software evitando que el proyecto se vea afectado por los cambios que regularmente se ingresan durante el desarrollo Además Git ofrece almacenamiento gratuito para el código fuente.

A partir del main se creará una rama de trabajo para cada iteración en donde se incluirán los cambios. Cuando se termine una iteración, si se superan los datos del Testing, y luego de ser aprobado por el cliente, se incluye en la rama principal, creándose una nueva versión y una nueva base desde donde iniciar la siguiente iteración.

Para deshacer cambios Git presenta el comando git Reset este comando tiene tres formas de invocación: –hard, –mixed y –soft, Siendo Hard la opción que deja el código tal cual estaba en la versión especificada.

# Plan de Capacitación

La interfaz que se diseñará será de rápido aprendizaje e intuitivo, por lo que el plan de capacitación no será tan complejo para el usuario. Se sabe que los usuarios que operaran con el programa tienen conocimientos previos sobre ingresar datos y como navegar en el programa. Aparte de la capacitación del usuario este será acompañada de la implementación de un manual de uso rápido.

Instancian de reuniones en las cuales los programadores irán adaptando el sitio a la comodidad del usuario además de explicar el funcionamiento de las distintas pantallas, despejando cualquier tipo de duda que vaya surgiendo en el camino

# Elección de metodología

Para la planificación de las tareas necesarias para la elaboración de un software de calidad para el cliente, es necesario definir un tipo de metodología que sirva como guía para llevar a cabo el proyecto en tiempo y forma.

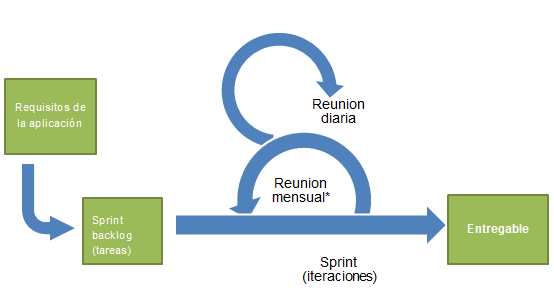
Para este proyecto se han tenido en cuenta dos metodologías:

## Modelo espiral

El desarrollo en espiral es un modelo de ciclo de vida del software definido por primera vez por Barry Boehm en 1986, ​ utilizado generalmente en la ingeniería de software. Las actividades de este modelo se conforman en una espiral, en la que cada bucle o iteración representa un conjunto de actividades.

[Información sacada de Wikipedia](https://es.wikipedia.org/wiki/Desarrollo_en_espiral)

## Metodología Scrum

Scrum es un marco de trabajo para desarrollo ágil de software que se ha expandido a otras industrias.

Es un proceso en el que se aplican de manera regular un conjunto de buenas prácticas para trabajar colaborativamente, en equipo y obtener el mejor resultado posible de proyectos, caracterizado por:

* Adoptar una estrategia de desarrollo incremental, en lugar de la planificación y ejecución completa del producto.
* Basar la calidad del resultado más en el conocimiento tácito de las personas en equipos auto organizados, que en la calidad de los procesos empleados.
* Solapar las diferentes fases del desarrollo, en lugar de realizar una tras otra en un ciclo secuencial o en cascada.

[Información sacada de Wikipedia](https://es.wikipedia.org/wiki/Scrum_(desarrollo_de_software))

# Ciclo de vida elegido

Para este proyecto se decidió utilizar la metodología de desarrollo incremental con un ciclo de vida basado en el modelo espiral.

El modelo elegido permitirá dividir el proyecto del sistema en partes pequeñas, para poder tener un mejor control del avance del proyecto y del cumplimiento de los requerimientos.

El modelo espiral permite la reducción de los riesgos asociados al proyecto debido a que los mismos han de ser considerados para cada etapa el desarrollo. Además, fuerza una estrecha comunicación con el cliente reduciendo también los riesgos asociados a la poca comunicación entre el cliente y los desarrolladores.

Permite la reducción de los riesgos asociados al proyecto ya que deben ser considerados en cada etapa del desarrollo.

## Etapas del ciclo de vida elegido

1. Comunicación con el cliente: Mantener un contacto con el cliente para que valide en cada ciclo los requerimientos o los modifique.
2. Planificación: Planificar y administrar los tiempos y recursos necesarios para el desarrollo del ciclo.
3. Análisis de riesgos: Considerar los riesgos que pudieran existir en la etapa de desarrollo y tomar las medidas necesarias para mitigarlos en caso de que existieran.
4. Programación: Desarrollo y testeo de la versión del sistema correspondiente al ciclo.
5. Testeo del software por parte de los programadores
6. Planificación de la siguiente etapa

# Iteraciones de implementación definidas

Posterior a comenzar el ciclo de iteraciones, se le entregará una copia del Anteproyecto al cliente que podrá validar el inicio del desarrollo. El cliente formará parte de la verificación conjunta de los resultados de cara iteración, lo que permitirá ir realizando los ajustes necesarios.

## Iteración Nro. 1:

Creación de Proyecto y la Base de Datos Excel

Creación de la interfaz gráfica

ABM:

Clientes

Compañías de Pólizas

Testing

## Iteración Nro. 2:

Listados de Clientes y Compañías

Aviso de renovación de pólizas

Consultas

Testing

## Iteración Nro. 3:

Pruebas:

Validaciones finales

Testing

Ajustes finales

Manual de usuario

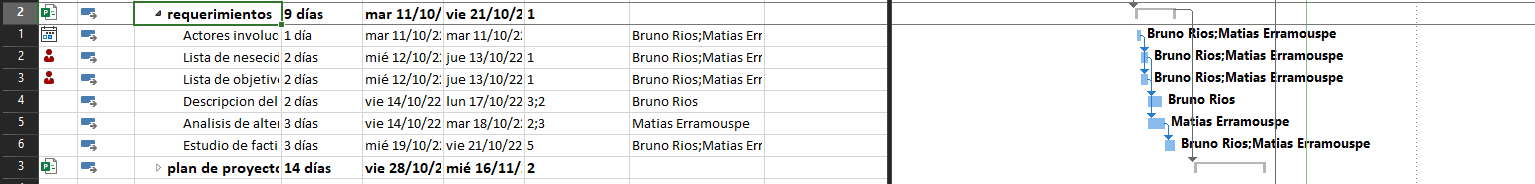
Documentación

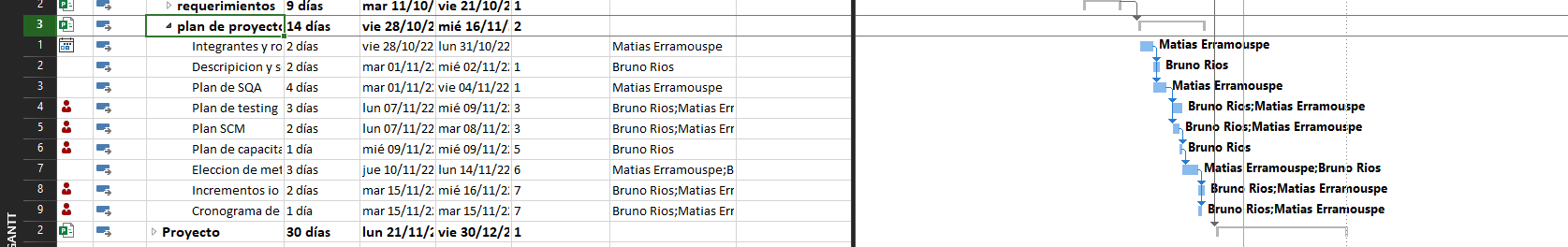
Bibliografía utilizada

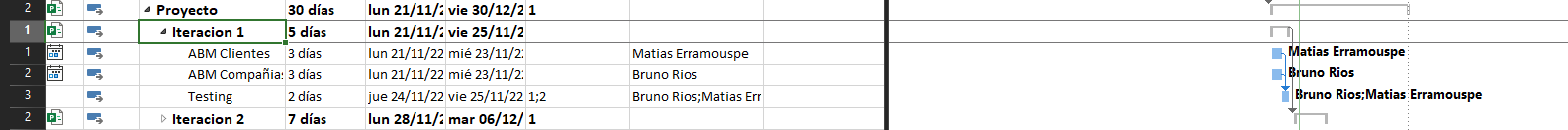
Grado de satisfacción del cliente

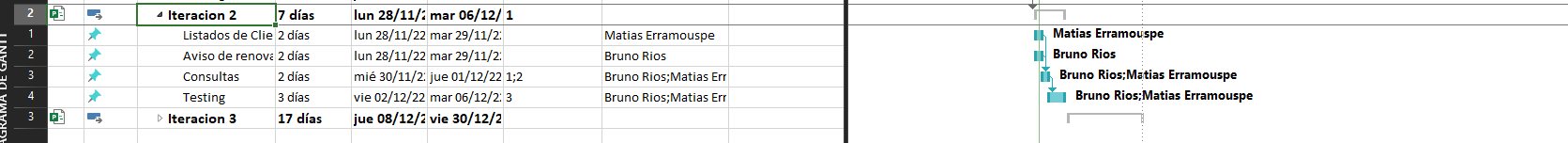
# Cronograma de trabajo y criticidad

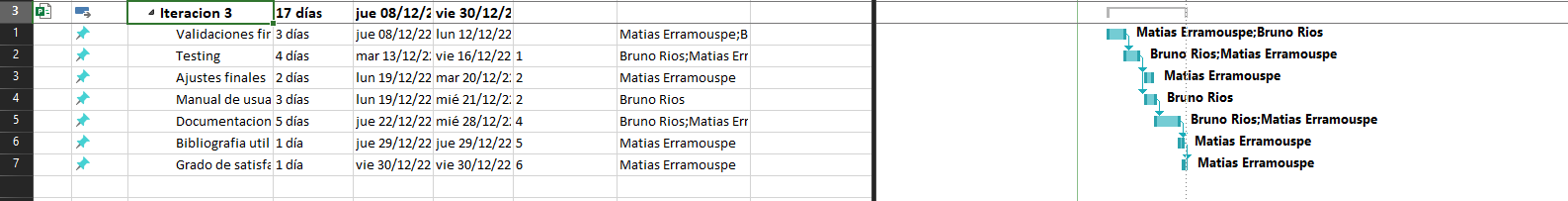
## Diagramas de Grantt











## Diagrama de Grannt ruta critica

