

Memoria

TEMPUS



**Autores:**

Quiroga, Sandra

Márquez, Emanuel

Oyarzo, Mariela

**Tutor:**

Mg. Albert Osiris Sofía

Laboratorio de Desarrollo de Software

Analista de Sistemas- Licenciatura en Sistemas

Unidad Académica Río Gallegos

Universidad Nacional de la Patagonia Austral



Esta memoria nos permite mostrar los resultados y el desarrollo del proyecto; a través de la evaluación del mismo, en el cual pudimos elaborar pequeños informes de cada actividad elaborada, y así poder terminar con el proyecto final.



Tabla de contenido

[Introducción 4](#_Toc56280358)

[Objetivos 4](#_Toc56280359)

[Objetivos Generales 4](#_Toc56280360)

[Objetivos Específicos 5](#_Toc56280361)

[Proceso de desarrollo 7](#_Toc56280362)

[Grupo de desarrollo 7](#_Toc56280363)

[Nombre y logo del grupo 7](#_Toc56280364)

[Nombre y logo del sistema 8](#_Toc56280365)

[Comunicación 8](#_Toc56280366)

[Descripción de los procesos actuales 10](#_Toc56280367)

[Descripción del proceso con Tempus 11](#_Toc56280368)

[Planificación 12](#_Toc56280369)

[Estimación 15](#_Toc56280370)

[Gestión de Riesgos 20](#_Toc56280371)

[Categorización 21](#_Toc56280372)

[Fase de Inicio 23](#_Toc56280373)

[Fase Elaboración 24](#_Toc56280374)

[Fase Construcción 24](#_Toc56280375)

[Conclusión 26](#_Toc56280376)

[Modelo de casos de uso 27](#_Toc56280377)

[Caso de Uso 01- Ingresar al Sistema 27](#_Toc56280378)

[Caso de Uso 02 - Importar Horario de Cursada 27](#_Toc56280379)

[Caso de Uso 03 - Importar Mesa de Examen 27](#_Toc56280380)

[Caso de Uso 04 - Buscar Horario de Cursada 27](#_Toc56280381)

[Caso de Uso 05 - Buscar Mesa de Examen 28](#_Toc56280382)

[Caso de Uso 06 - Modificar Horario de Cursada 28](#_Toc56280383)

[Caso de Uso 07 - Modificar Mesa de Examen 29](#_Toc56280384)

[Caso de Uso 08 - Crear Horarios de Cursada 29](#_Toc56280385)

[Caso de Uso 09 - Crear Mesa de Examen 29](#_Toc56280386)

[Caso de Uso 10 - Notificar 30](#_Toc56280387)

[Caso de Uso 11 - Informe de Mesa de Examen 30](#_Toc56280388)

[Caso de Uso 12 - Informe de Horario de Cursada 30](#_Toc56280389)

[Caso de Uso 13 - Buscar Aula 30](#_Toc56280390)

[Caso de Uso 14 - Borrar Horario de Cursada 31](#_Toc56280391)

[Caso de Uso 15- Borrar Mesa de Examen 31](#_Toc56280392)

[Caso de Uso 16- Modificar Aula 31](#_Toc56280393)

[Caso de Uso 17- Borrar Aula 32](#_Toc56280394)

[Caso de Uso 18- Generar Horario de Aula 32](#_Toc56280395)

[Modelo de datos 32](#_Toc56280396)

[Implementación 32](#_Toc56280397)

[Pruebas 32](#_Toc56280398)

[Tecnologías 33](#_Toc56280399)

[Hardware 33](#_Toc56280400)

[Software 34](#_Toc56280401)

[GitHub 34](#_Toc56280402)

[Ionic Framework 34](#_Toc56280403)

[Netbeans 34](#_Toc56280404)

[Taiga 34](#_Toc56280405)

[Tortoise SVN: 35](#_Toc56280406)

[UARGFLOW 35](#_Toc56280407)

[Visual Studio Code 35](#_Toc56280408)

[WhatsApp 36](#_Toc56280409)

[XAMPP 36](#_Toc56280410)

[XDEBUG 36](#_Toc56280411)

[Zoom 37](#_Toc56280412)

[Conclusión 37](#_Toc56280413)

[Experiencia personal 37](#_Toc56280414)

[Oyarzo Mariela 37](#_Toc56280415)

[Quiroga Sandra 37](#_Toc56280416)

[Marquez Emanuel 37](#_Toc56280417)

[Experiencia grupal 38](#_Toc56280418)

[Anexo 39](#_Toc56280419)

[Biblografia 40](#_Toc56280420)

Memoria

**Introducción**

El presente documento es la memoria del proyecto Tempus que propone la Gestión de Horarios de Cursada y Mesas de Examen. Dicho proyecto fue otorgado al Grupo de Desarrollo Yenú en el año 2017, por el equipo docente de la asignatura Laboratorio de Desarrollo de Software perteneciente a las carreras Analista de Sistemas y Licenciatura en Sistemas de la UNPA-UARG.

 Aquí se expresa en forma detallada el proceso de desarrollo del Sistema realizado y su resultado final.  Se describirán las experiencias vividas de cada integrante del grupo durante el desarrollo del proyecto, se hará hincapié en los inconvenientes ocurridos, en  las soluciones encontradas para los mismos y se hablará de los conocimientos adquiridos, durante el desarrollo del sistema.

Tempus contará con un sitio web, que permitirá acceso al personal de Secretaría Académica para cargar los horarios de cursada y mesas de examen. Además, podrán actualizar la información de cursadas, mesas de examen, aulas y carreras junto con la generación de informes. A su vez, se asignará un usuario administrador que podrá realizar la gestión de los usuarios que tengan acceso al sistema.

Tempus contará con una aplicación móvil, que permitirá al público en general consultar sobre los horarios de cursada y mesas de examen. Además, los usuarios podrán recibir notificaciones cuando se genere información de interés.

En las próximas secciones de este documento, se especificarán: los objetivos del proyecto, el proceso de desarrollo del mismo, un resumen de las iteraciones realizadas, las tareas de calidad y gestión del proyecto, las tecnologías utilizadas, algunas features destacadas del sistema y potenciales mejoras a futuro y, finalmente, conclusiones personales y grupales de los miembros del acerca del proyecto, las experiencias vividas y los conocimientos obtenidos.

Objetivos

Objetivos Generales

El objetivo del proyecto es crear un sistema que dé solución a un conjunto de problemas y que beneficie al personal del área de Secretaría Académica, alumnos y docentes de la UNPA UARG.

A continuación se detallan algunos de los problemas detectados:

* No existe actualmente una herramienta hecha a medida que le permita al personal de Secretaría Académica realizar la carga y publicación de los horarios de cursada y mesas de examen.
* No existe actualmente una herramienta hecha a medida que le permita a los alumnos y docentes consultar la información sobre horarios de cursada y mesas de examen.
* No existe una forma de comunicar rápidamente al público en general sobre modificaciones en los horarios de cursada o mesas de examen. Tampoco existe la forma de comunicar que se encuentran disponibles nuevos horarios de cursada o nuevas mesas de examen.
* Existe la incertidumbre sobre la certeza que los horarios de cursada y mesas de examen se encuentran actualizados o son correctos.
* Existen problemas en la asignación de los horarios y aulas para las materias en los horarios de cursada. Este inconveniente se ve reflejado también en las mesas de examen. Esto afecta a los docentes y alumnos que no cuentan con esta información en el momento necesario.

Los inconvenientes actuales son de solucionables a través del desarrollo de un sitio web que permita cargar horarios de cursada y mesas de examen a usuarios que cuenten con un correo institucional, junto con una aplicación móvil que permite consultar dicha información a usuarios públicos.

Para un buen uso de la aplicación creamos un diseño cómodo, agradable y fácil de entender para todos los usuarios que se conecten mediante un sistema móvil. Este es el objetivo principal de la aplicación.

Objetivos Específicos

El área de Secretaría Académica lleva a cabo la generación de horarios de cursada y mesas de examen mediante el uso de planillas de cálculo. Estas tareas exhiben dificultades cuando se presentan cambios, o se realizan malas prácticas, implicando un costo temporal importante. Estos problemas impactan negativamente en las personas que deben utilizar la información generada. En este punto, radican los principales problemas a los cuales se les brinda solución con el desarrollo de un producto de software capaz de garantizar:

* Acceso a usuarios autorizados para la generación de horarios (Cursada y mesas de examen).
* Facilidad de carga de información a través del uso de los archivos existentes (hojas de cálculo).
* Facilidad de acceso a los horarios (Cursada y mesas de examen).

Por lo cual, se especificó:

1. **El sistema permitirá importar horarios de cursada a usuarios autorizados**. El personal de Secretaria Académica o Administrador podrá cargar los horarios de cursada utilizando las planillas de cálculo (Excel) que poseen en la actualidad o alguna variante que no contenga grandes modificaciones.
2. **El sistema permitirá importar mesas de examen a usuarios autorizados**. El personal de Secretaría Académica o el Administrador podrá cargar las mesas de examen utilizando la planilla de cálculo (Excel) que poseen en la actualidad o con alguna variante que no contenga grandes modificaciones.
3. **El sistema permitirá actualizar horarios de cursada a usuarios autorizados**. El personal de Secretaría Académica o el Administrador podrá crear, borrar o modificar los horarios de cursada que se encuentren cargados en el sistema.
4. **El sistema permitirá actualizar mesas de examen a usuarios autorizados**. El personal de Secretaria Académica o Administrador podrá crear, borrar o modificar las mesas de examen que se encuentren cargadas en el sistema.
5. **El sistema permitirá consultar horarios de cursada a usuarios públicos**. Los docentes, alumnos y público en general podrán consultar los horarios de cursada con la aplicación móvil.
6. **El sistema permitirá consultar mesas de examen a usuarios públicos**. Los docentes, alumnos y público en general podrán consultar las mesas de examen con el uso de la aplicación móvil.
7. **El sistema permitirá generar informes de horarios de cursada a usuarios autorizados**. El Administrador o personal de Secretaría Académica podrá  acceder a informes sobre la disponibilidad horaria de aulas durante el período de cursada.
8. **El sistema permitirá generar informes de mesas de examen a usuarios autorizados**. El Administrador o personal de Secretaría Académica podrá  acceder a informes sobre la disponibilidad horaria de aulas durante el periodo de mesas de examen.
9. **El sistema permitirá generar informe sobre horarios de un aula**. Un usuario autorizado podrá generar un informe sobre los horarios de una determinada aula.
10. **El sistema deberá permitir actualizar aulas a usuarios autorizados**. El personal de Secretaría Académica o Administrador podrá borrar o modificar las aulas que se encuentren cargadas en el sistema.
11. **El sistema deberá permitir actualizar carreras a usuarios autorizados**. El personal de Secretaría Académica  o Administrador, podrá borrar o modificar las carreras que se encuentren cargadas en el sistema.
12. **El sistema permitirá al administrador gestionar usuarios**. El administrador podrá gestionar usuarios, los roles que cumplen y los permisos que se le otorgan.
13. **El sistema permitirá notificar a usuarios públicos sobre actualizaciones en los horarios de cursada y mesas de examen**. El usuario que utilice la aplicación móvil de Tempus podrá recibir notificaciones cuando se realicen actualizaciones en la información de horarios de cursada y mesas de examen.

Creemos que lo más significativo del proyecto fue el armado de un equipo de desarrollo conformado por estudiantes que debieron analizar un problema real de la UNPA-UARG y, siguiendo las pautas de trabajo brindadas por la metodología PSI y el framework UARGFlow, llevaron a cabo el desarrollo de una solución informática poniendo en práctica todos los conocimientos obtenidos a lo largo de la carrera en el marco de una asignatura llamada “Laboratorio de Desarrollo de Software”.

El cumplimiento del desarrollo del software que brinda solución a los  problemas presentados, permitirá alcanzar además  el objetivo  académico que implica: conformación de un grupo de desarrollo, la aplicación de los conocimientos adquiridos durante la carrera de Analista de Sistema y  adquisición de nuevos conocimiento durante el proceso de desarrollo del proyecto Tempus.

# Proceso de desarrollo

En esta  sección  vamos a realizar un repaso cronológico del proceso de desarrollo del software. Analizaremos desde las etapas iniciales hasta las finales, haciendo hincapié en los inconvenientes que surgieron y  en la resolución de los mismos.

## Grupo de desarrollo

El primer paso que se realizó fue la elección del grupo, quedando conformado por; Oyarzo Mariela, Quiroga Sandra y Marquez Emanuel.Luego se realizó la definición del nombre del grupo, nombre del sistema, junto con la creación de los logos respectivamente.

### Nombre y logo del grupo

El grupo de desarrollo se denomina Yenú que significa “Amigo” en Tehuelche y está inspirado en los principales valores de una amistad junto con la solidaridad, sinceridad y compromiso, los cuales como grupo consideramos esenciales a la hora trabajar.



El logo del grupo presentado anteriormente, se  define como un símbolo conformado por  una imagen y letras, que sirve para identificar al grupo,  algunos de los aspectos más importantes es que es legible y presenta contraste entre los colores seleccionados, los cuales fueron la gama de los azules (#7fb2c3) dado que se asocia con la estabilidad, lealtad y confianza. Para la letra se ha utilizado la fuente Dragonfly en color negro.

### Nombre y logo del sistema

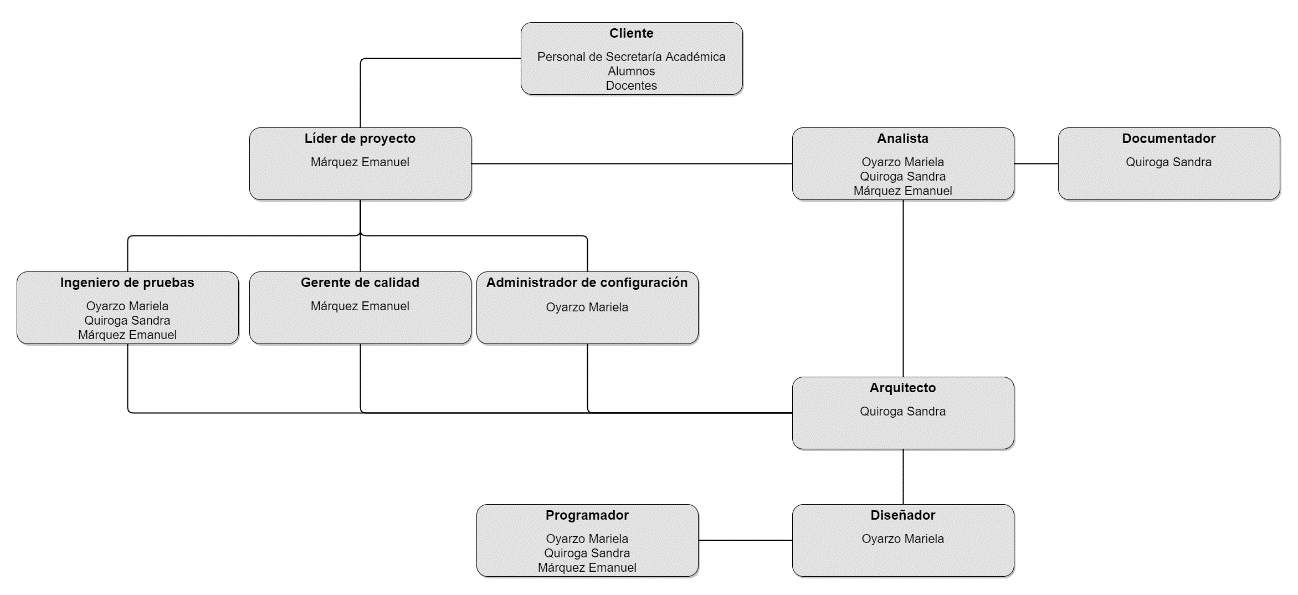
 El sistema se llama Tempus, el cual está inspirado en el tiempo y relacionado con las características principales del sistema (Horarios). Se ha seleccionado ya que consideremos que uno de los principales factores del éxito en cursadas y mesas de examen es justamente, la dedicación de tiempo.



El logo del sistema presentado anteriormente, se  define como un símbolo conformado por  una imágen inspirado en los primeros relojes de sol creados por los egipcios y luego mejorado por los romanos. El mismo, indicaba los momentos del día por la sombra de la aguja con el sol. Algunos de los aspectos más importantes es  que es un diseño simple, sintetizado y fácilmente identificable con un equilibrio entre el diseño y el texto.

Comunicación

La comunicación siempre es clave para despejar cualquier duda, acontecimiento o planificación entre las partes y así evitar posibles errores, y saber a quién preguntar. A continuación, mostraremos un diagrama general de comunicación que utilizamos a la hora de organizarnos o bien mantener una interacción entre los mismos.



Durante el desarrollo del proyecto el equipo mantuvo una comunicación muy activa, fluida, en forma presencial y mediante la utilización de diferentes herramientas tecnológicas.

Podemos diferenciar tres etapas claramente marcadas:

Una primera etapa durante el período de cursada que abarca desde el 17 de Agosto del 2017 hasta el 24 de Noviembre del mismo año. En este espacio de tiempo, las comunicaciones estaban dadas por  reuniones presenciales aprovechando los horarios de clase en el Laboratorio A4, reuniones fuera de horario en el Campus y en los hogares de los integrantes, charlas mediante Whatsapp y utilización de Taiga. Esta forma de trabajo fue beneficiosa para el grupo porque se lograron grandes avances en el desarrollo.

A su vez, la comunicación entre el grupo y el equipo de cátedra era continua. Se realizaban consultas durante las clases, mediante correo electrónico y presentaciones semanales que les permitían evaluar los avances del proyecto. Se debe destacar que ellos cumplían el rol del cliente dado que solo tuvimos una sola reunión con el personal de Secretaría Académica de carácter informativo.

Una segunda etapa que abarca desde el 11 de Octubre del 2019 hasta el 22 de Noviembre del 2019, donde las reuniones presenciales disminuyeron y se hizo uso de charlas mediante Whatsapp. Esto se debía a que cada uno de los integrantes continuaba con el desarrollo de la carrera sin mantener contacto en alguna materia. La productividad y los avances no fueron amplios dado que se tuvieron que hacer tareas de recopilación, repaso de los objetivos del proyecto y actualización de la aplicación a la nueva versión de IONIC. En este punto se descontinuó el uso de la herramienta otorgada por el equipo de cátedra (Taiga) pero se mantuvieron las planificaciones mediante los Planes de Iteración.

Con respecto a la comunicación con los docentes en esta etapa,  se realizó una presentación  con los avances del proyecto, donde el equipo de cátedra nos planteó además cambios en la aplicación móvil.

Una tercera y última etapa que abarca desde mediados de Mayo del 2020 hasta el 9 de Noviembre del mismo año.  En este punto, dada la proximidad de la finalización de la regularidad de la materia, el grupo de desarrollo, decidió continuar avanzando en el proyecto. La particularidad de esta etapa, está dada por la disposición de  un AISLAMIENTO SOCIAL, PREVENTIVO  Y OBLIGATORIO. Por el decreto Número 260 del 12 de Marzo del 2020, se amplió en nuestro País la emergencia pública en materia sanitaria, por el plazo de un año en virtud de la pandemia declarada. Por este motivo no se podían llevar a cabo reuniones de carácter presencial, por lo que nos vimos obligados a realizar reuniones en forma virtual, utilizando las herramientas WathApp, Zoom, Meet, Drive, GitHub.

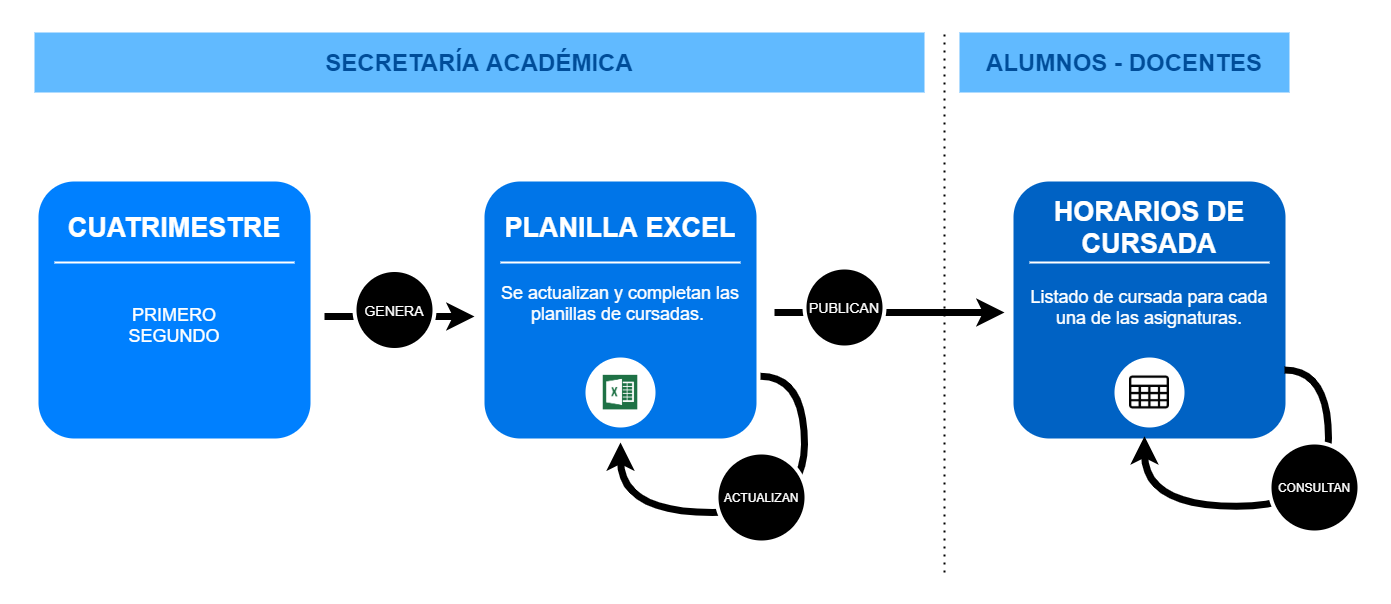
En esta última etapa, no se realizaron reuniones con el equipo de cátedra, sólo se mantuvo contacto para la entrega de este informe, mediante el envío de correo a los docentes.

Se puede decir que las herramientas más utilizadas fueron el Correo grupal, WathApp, Zoom, Meet, Drive, GitHub y Taiga, las cuales se describen en la sección correspondiente.

Descripción de los procesos actuales

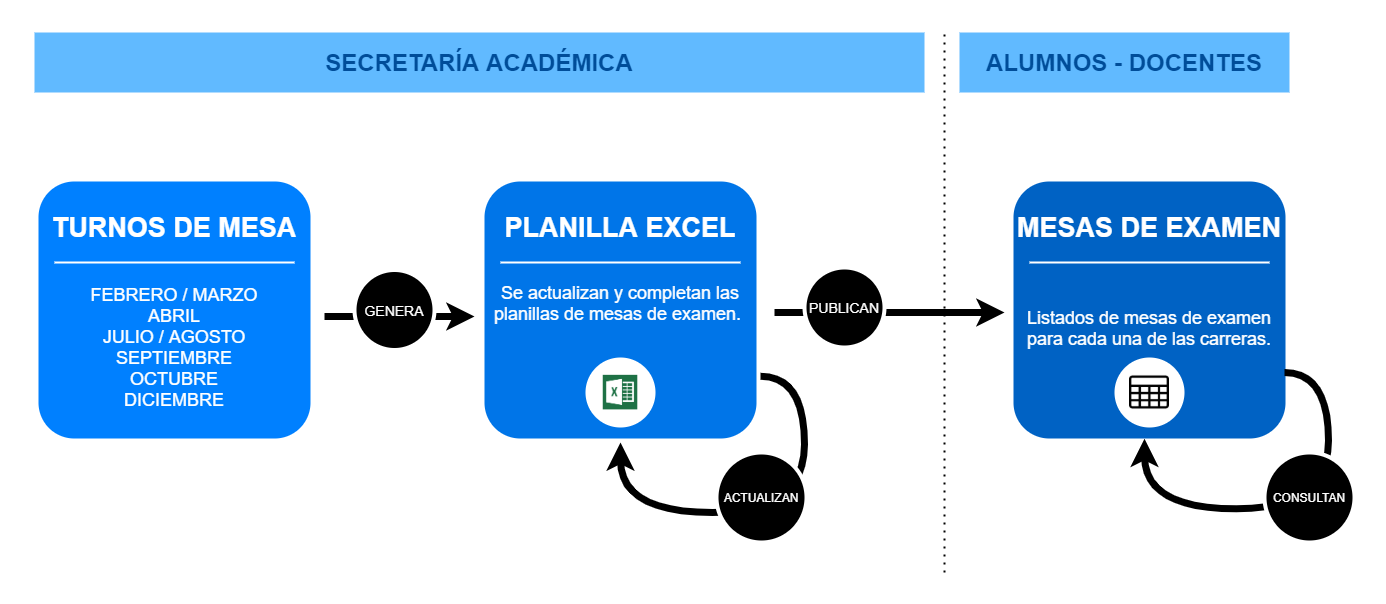
En esta sección se describirán cómo se lleva a cabo actualmente el proceso de  generación de horario de cursada y generación de mesa de examen por Secretaría Académica. Para ello se utilizarán diagramas que faciliten la comprensión del proceso, que muestren los pasos necesarios para la realización del mismo, que  demuestren las interacciones entre las actividades y permitan detectar problemas.

Previo al inicio de cada cuatrimestre (Primero o Segundo) el personal de Secretaría Académica se encarga de generar los horarios de cursada utilizando una planilla de cálculo Excel. Este archivo, posteriormente, se publica en el Portal de la UARG y se imprime para pegarse en el Campus Universitario. Luego, los alumnos y docentes observan los horarios de cursada que fueron generados previamente. Cuando se realiza alguna modificación a los horarios de cursada, el personal de Secretaría Académica actualiza la planilla de cálculo y vuelve a publicarlos. Los docentes y alumnos deben consultar los horarios de cursada regularmente para detectar si se han producido modificaciones.



Proceso para la generación de horarios de cursada

A continuación se explica el proceso para la generación de mesas de examen. Previo al inicio de cada turno de examen, el personal de Secretaría Académica se encarga de generar las mesas de examen utilizando una planilla de cálculo Excel. Este archivo, posteriormente, se publica en el Portal de la UARG y se imprime para pegarse en el Campus Universitario. Luego, los alumnos y docentes observan las mesas de examen que fueron generadas previamente. Cuando se realiza alguna modificación a las mesas, el personal de Secretaría Académica actualiza la planilla de cálculo y vuelve a publicarlos. Los docentes y alumnos deben consultar las mesas de examen regularmente para detectar si se han producido modificaciones.



Proceso para la generación de mesas de examen

## Descripción del proceso con Tempus

Hasta el momento se presentó el problema con sus causas y consecuencias por lo que en esta sección se describirán las soluciones desarrolladas por el grupo Yenu tanto para el  proceso de  generación de horario de cursada como la generación de mesas de examen.

Para dar solución a la problemática, se definieron los perfiles de usuario Administrador, Secretaria Académica y Público en General, se realizó la implementación de una página web que permita cargar horarios de cursada y mesas de examen, junto con la implementación de una aplicación móvil que permite consultar la misma.

 El producto desarrollado, en líneas generales, permite realizar la creación, eliminación, modificación y carga de horarios de cursada y mesas de examen importando las planillas de cálculo que poseen actualmente los empleados de Secretaría Académica de la UNPA-UARG con algunas variantes. Esta tarea debe hacerse accediendo al sistema a través de su sitio web e identificándose mediante un correo institucional. Además, puede acceder a la generación de informes.

A través de un dispositivo móvil que cuente con la aplicación Tempus, se permite al público en general, representado por alumnos y docentes ,  acceder a consultar los horarios de cursada y mesas de examen que han sido previamente cargados por Secretaría Académica.

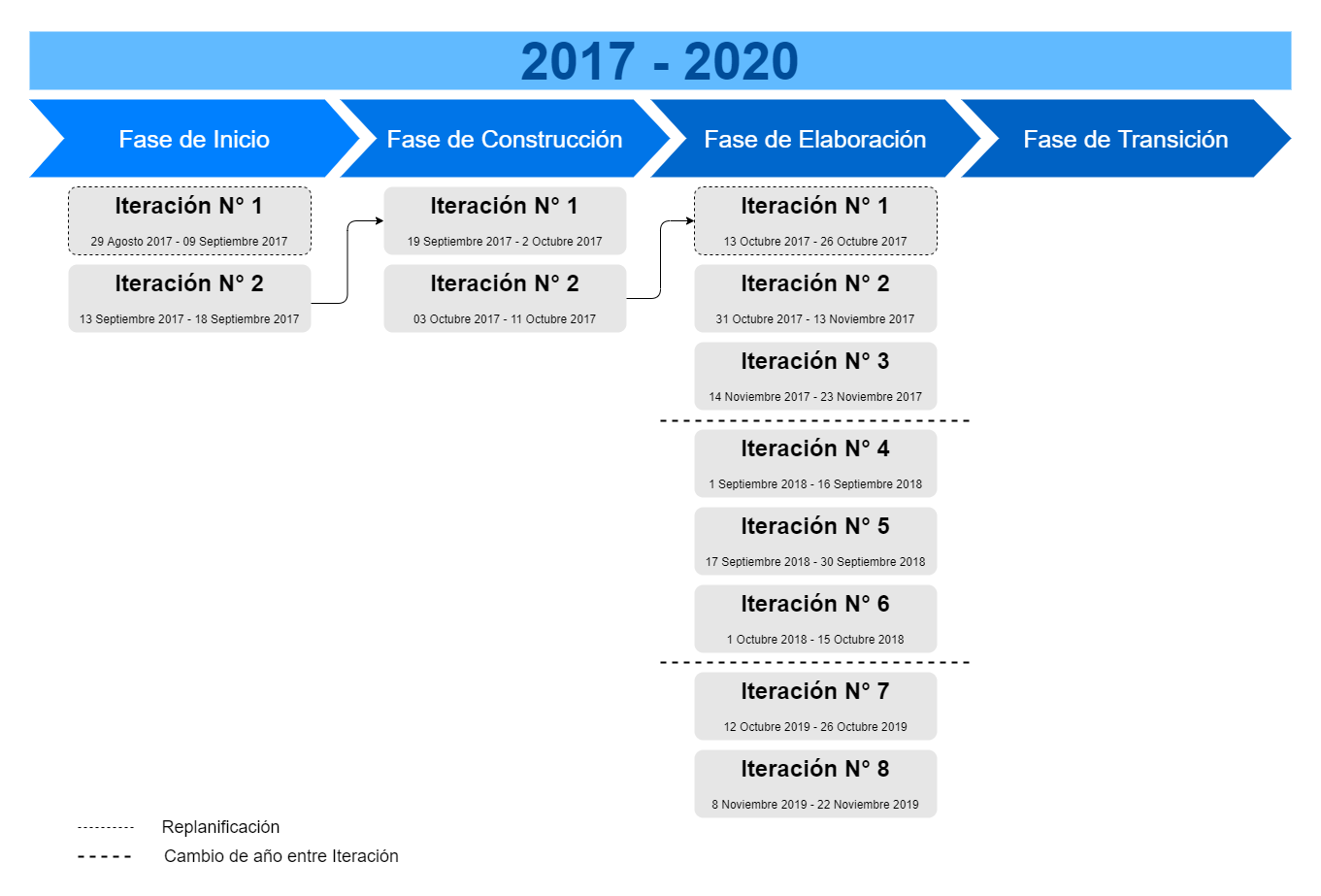
Por último, el usuario de tipo Administrador cuenta con la capacidad de administrar el acceso al sistema junto con todas las demás funcionalidades disponibles.

De aquí en adelante se tratarán temas más específicos de la metodología PSI.

## Planificación

Al iniciar el periodo de cursada de la materia, el equipo docente nos recomienda hacer una primera planificación para dar comienzo al proyecto. En la primera semana no se contaba con la información necesaria con la cual basar una planificación. No obstante, para hacer un plan, primero se tuvo que establecer que es lo que se necesitaba hacer: recopilar información, organizar las tareas, definir las capacidades individuales de los integrantes y analizar como continuar.

En este punto, la metodología PSI fue una herramienta de gran utilidad. Sin embargo, existieron problemas en la forma de entender como planificar las actividades: que se debía hacer, en qué orden debería hacerse o a quien se asignaba las tareas.



Dado que solo contábamos con una descripción genérica de las características del sistema (una página web en PHP y una aplicación móvil con Ionic) el objetivo estaba puesto en ampliar el conocimiento del sistema a desarrollar. En este aspecto se definieron actividades como el desarrollo de un plan general, identificación y clasificación de los requerimientos para detallarlos en los casos de uso y desarrollo de una arquitectura. A continuación listamos las tareas planificadas de la Iteración 1 – Fase de inicio:



*Planificación de tareas en el Plan de Iteración Fase Inicio Iteración 1*

Si se realiza un análisis de esta planificación inicial, se puede observar que:

* Las tareas se redactaban en forma general sin especificar con mayores detalles qué se debía realizar. Por ejemplo: “Redactar Plan del Proyecto” no indica si debe hacer un nuevo documento, utilizar alguno disponible o qué secciones deben completarse.
* Las actividades no tienen asignada una prioridad. Esto implicaba que cada persona iniciaba su actividad en el orden que desease. No se llevó a cabo previo a la planificación qué documento o información era primordial y cual no.
* Si bien la cantidad de actividades está distribuida en forma pareja, la complejidad entre ellas varía. Por lo que, por ejemplo “Cerrar Iteración 1 Fase Inicio” es una actividad que implica mucho menos tiempo que otras como “Redactar Modelo de Visión” o “Redactar Modelo de Diseño”. Esto podía ocasionar que algún integrante del grupo termine todas sus tareas antes de tiempo mientras que otro este con una carga mayor.
* Había tareas que se realizaban y que no se encontraban en la planificación: un ejemplo era la carga de las actividades del documento Plan de Iteración en la herramienta Taiga.

Los aspectos anteriormente mencionados ocasionaron que los objetivos de la Iteración 1 no se hayan cumplido. La cantidad de tareas que se redactaron eran demasiadas para el tiempo asignado (29 de Agosto al 10 de Septiembre).



*Objetivos establecidos en el Plan de Iteración Fase Inicio Iteración 1*

Sin embargo, se concluyó:

“*La iteración no cumplió con los objetivos establecidos para la fecha 12 de Septiembre de 2017. Por tanto se decide crear una nueva iteración (Fase Inicio, Iteración 2) con los objetivos que no se han cumplido. Además, no se tendrá en cuenta el documento “Modelo de Visión”.*”

Atendiendo estos problemas y con recomendaciones del equipo de catedra, se establecieron las siguientes mejoras para la nueva iteración:

* Mantener el periodo de dos semanas como duración de una iteración. Salvo que sea una re-planificación donde la duración debería ser una semana.
* Cerrar las iteraciones un día antes de la siguiente clase para permitir presentar la nueva planificación al equipo de catedra.
* Atomizar las tareas. Reducir el número de actividades generales para planificar actividades más específicas.
* Agregar una sección al plan de iteración donde se encuentre una tabla con las actividades de cada integrante.
* Priorizar las tareas en concordancia con los niveles de prioridad establecidos en Taiga.

Si observamos el plan de iteración de una fase más avanzada del proyecto encontramos una planificación como la siguiente:



*Planificación de tareas en el Plan de Iteración Fase Construcción Iteración 7*

## Estimación

Una parte importante de la toma de decisiones al comenzar un nuevo proyecto de desarrollo de software está dada por el costo que éste tendrá, el tiempo que se tardará en desarrollar el sistema, y cuantas personas se necesitan para desarrollarlo.

Por lo tanto es necesario cuantificar, complejidad del sistema, funcionalidad, complejidad técnica, el nivel de experiencia de los integrantes del proyecto y el tiempo necesario para producir una unidad funcional.

Para ello utilizamos el  método de estimación de puntos en casos de uso, de Gustav Karner (1993), que se adapta bien al enfoque de caso de uso para la descripción de los requisitos. En sus bases yace el concepto de transacción de caso de uso, la unidad más pequeña de medición. El método utiliza los [actores](https://es.wikipedia.org/wiki/Actor_(UML)) y [casos de uso](https://es.wikipedia.org/wiki/Caso_de_uso) relevados para calcular el esfuerzo que significa desarrollarlos. A los casos de uso se les asigna una complejidad basada en transacciones, entendidas como una interacción entre el usuario y el sistema, mientras que a los actores se les asigna una complejidad basada en su tipo, es decir, si son interfaces con usuarios u otros sistemas. También se utilizan factores de entorno y de complejidad técnica para ajustar el resultado.

 Durante el desarrollo del proyecto, se realizaron diferentes estimaciones.

En primer lugar tenemos una estimación inicial, que nos dio como resultado 1 año y 7 meses. Que resulta de calcular en términos generales 14 Casos de Uso, sin tener en cuenta la reutilización del módulo de UARGFLOW, 2 de ellos complejos, 4 de tipo promedio y 8 Casos de Uso de tipo simple.

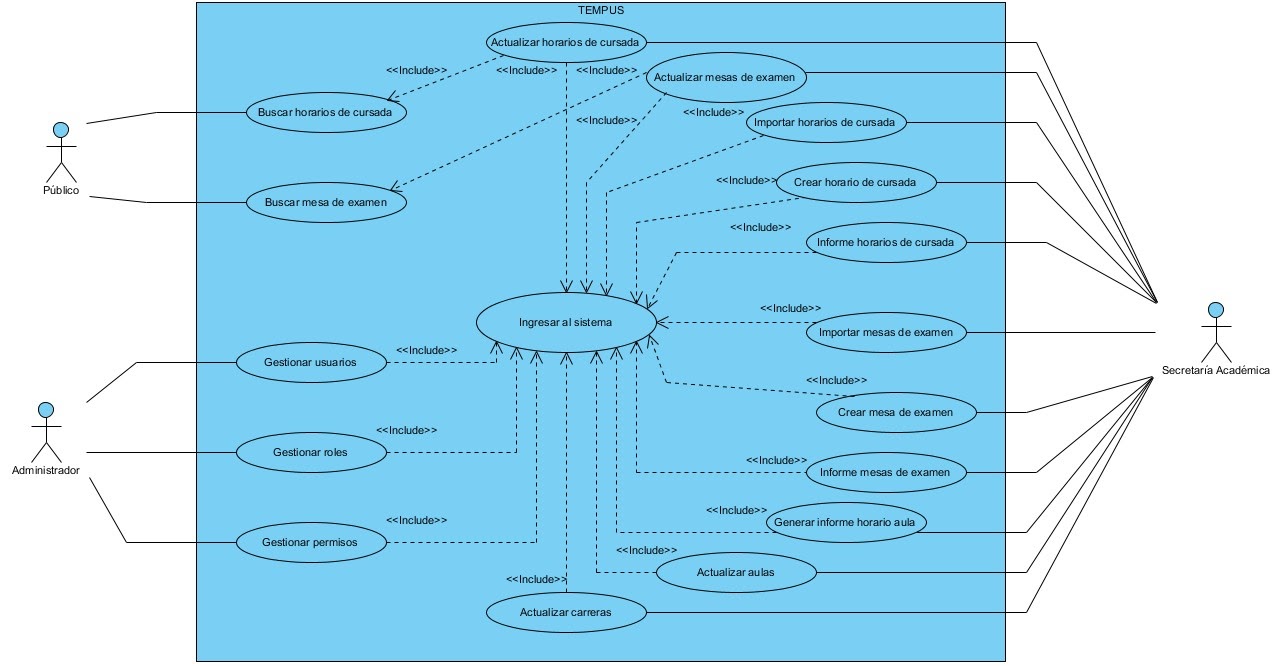
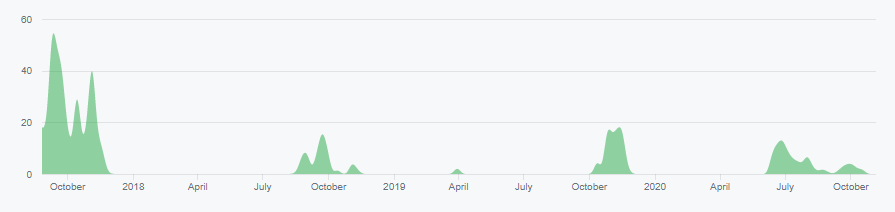


Diagrama de casos de uso

También influyen en este resultado los factores ambientales del equipo, que con el tiempo fueron cambiando gracias a la gestión de riesgos, como ser la familiaridad con el proceso definido, la experiencia en el dominio de la aplicación, capacidad de liderazgo de analista y dificultad con el lenguaje de programación.

La estimación inicial se fue ajustando de acuerdo a los cambios en los factores ambientales nombrados anteriormente, teniendo como resultado una última  estimación  con fecha de realización, 7 de Abril del 2019, la cual nos decía que faltaba 1 mes y 3 semanas para finalizar el proyecto.

Estas estimaciones fueron comparadas con gráficos que nos brinda el repositorio obteniendo como resultado lo siguiente:

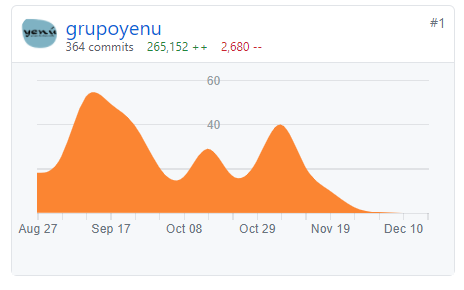


Trabajo realizado por Yenu

El gráfico 1 - Trabajo realizado por Yenu, se obtuvo del repositorio, el cual detalla el tiempo de trabajo real del equipo de desarrollo.

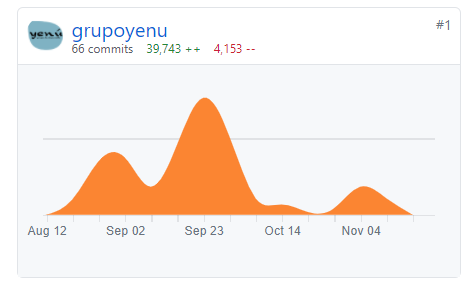
Se puede observar que se trabajó desde el 27 de Agosto del 2017 al 9 de Noviembre del 2020 de manera interrumpida, donde se ven reflejadas cuatro etapas claramente marcadas.

Cabe aclarar que los motivos de las diferentes interrupciones, se verán en forma detallada en la sección de  descripción de los riesgos en esta memoria.



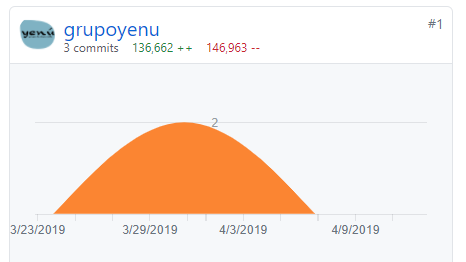
Trabajo realizado por Yenu primera etapa

El  Gráfico 2 - Trabajo realizado por Yenu primera etapa, es una maximización de la primera fase del Gráfico 1. El cual corresponde al primer periodo del 27 de Agosto del 2017 al 6 de diciembre del mismo año. El cual coincide con el cuatrimestre en el que se cursó la materia Laboratorio de desarrollo del Software.



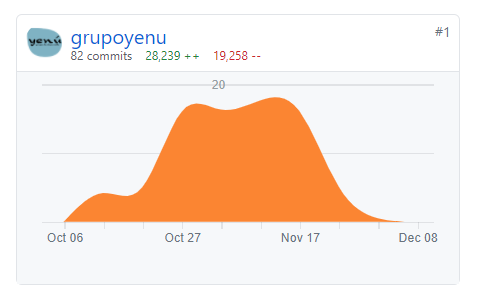
Trabajo realizado por Yenu segunda etapa

El  Gráfico 3 - Trabajo realizado por Yenu segunda etapa, es una maximización de la segunda fase del Gráfico 1. El cual, abarca el periodo del 12 de Agosto del 2018 al 22 de noviembre del mismo año.



Trabajo realizado por Yenu tercera etapa

El  Gráfico 4 - Trabajo realizado por Yenu tercera etapa, es una maximización de la tercera fase del Gráfico 1. La cual, inicia el 24 de Marzo del 2009 y finaliza el 7 de Abril del 2009.



Trabajo realizado por Yenu cuarta etapa

El  Gráfico 5 - Trabajo realizado por Yenu cuarta etapa, es una maximización de la cuarta fase del Gráfico 1. La cual corresponde al periodo del 6 de Octubre del 2020 al 9 de Noviembre del mismo año.

A modo de conclusión, podemos decir que al sumar estas cuatro etapas obtenemos que el desarrollo de nuestro proyecto duró 8 meses, 3 semanas y 6 días.

Comparando este resultado con  el obtenido en la estimación inicial, observamos que hay una diferencia de 10 meses y una semana.

Las fallas en la estimación inicial, se debieron a:

* Poco conocimiento de cómo estimar.
* No contar con experiencia en la realización de estimaciones.
* No poder comparar con proyectos anteriores.
* No contar con estadísticas propias.
* Subestimación del grupo de desarrollo en cuanto a conocimientos previos.

Por otro lado teníamos una última  estimación  con fecha de realización, 7 de Abril del 2019, la cual nos decía que faltaba 1 mes y 3 semanas para finalizar el proyecto. Si observamos el gráfico número 5, vemos que la  cuarta y última etapa del desarrollo del proyecto, corresponde al periodo del 6 de Octubre del 2020 al 9 de Noviembre del mismo año. Comparando estos  resultados, observamos que hay una diferencia de dos semanas y un día. La cual no es significante.



//faltaria grafico comparando estimacion inicial y real

Gestión de Riesgos

Este proceso nos permitió identificar los riesgos mediante análisis cualitativo y cuantitativo, al inicio de cada iteración del proyecto y así planificar respuestas y hacer el seguimiento de los mismos, siendo esta una de las primeras tareas a realizar en este proyecto.

Se trabajó de manera individual y grupal sobre los posibles riesgos que tenían cada uno de los integrantes, tanto en lo personal como en, las herramientas y la metodología utilizada. En el cual utilizamos documentos del psi -una planilla de Excel- donde contestábamos preguntas sobre posibles riesgos que podrían surgir durante el desarrollo del proyecto. Debía indicarse una valoración para el impacto del riesgo, esto permitió obtener un listado resumido de los riesgos más sobresalientes de cada categoría. Una vez analizados pudimos obtener aquellos riesgos de mayor factor, que podían impactar sobre el mismo.

Detectar estos riesgos de manera temprana nos permite reducir el costo a futuro al igual que el tiempo para poder resolverlo. Sin embargo cabe aclarar que hay riesgos imposibles de detectar de esa forma, ya que pueden surgir nuevos durante el desarrollo algo imposible de manejar. Está claro que este es uno de los motivos por el cual se retrasó parte de la planificación del proyecto, junto con la inexperiencia de los integrantes del grupo.

Este análisis de riesgo nos permitirá tener un producto de mayor calidad, el cual al finalizar ayude a resolver las necesidades de los usuarios tal como se espera que funcione.

Categorización

Las categorías de riesgos son utilizadas para generalizar ciertas preguntas sobre posibles riesgos, en la cual podíamos hacer una identificación y Evaluación de las mismas, y así poder encontrar aquellos que podían impactar sobre nuestro proyecto. Las de mayor impacto fueron a las cuales le realizamos un seguimiento durante el desarrollo del mismo. A continuación escribimos por qué impactaron estas categorías:

* Cronograma: Esta categoría fue elegida porque se considera el tiempo necesario que dispone cada uno de los integrantes del grupo de desarrollo para la realización de las actividades. Esta categoría es importante, ya que el tiempo es necesario para la realización de este proyecto su impacto podría retrasar mucho el mismo, modificar la planificación o bien el costo de dicho proyecto.
* Experiencia y Capacidad: Esta categoría fue elegida porque se considera necesario saber el tipo de experiencia o capacidad que tiene cada uno para poder dar avance al proyecto, es decir que la falta de experiencia y capacidad sobre las herramientas, tecnología, metodología o bien el lenguaje de programación que se va a utilizar, trae un retraso al desarrollo, ya que hay que ponerse a estudiar y revisar aquello que desconocemos. Tiempo que podría utilizarse para planificar o realizar actividades de este proyecto.
* Duración y Tamaño: Esta categoría fue considerada por los factores que pueden alterar el ritmo de trabajo. Siendo estos aquellos que se presentan durante los meses o años que llevo la realización de este proyecto.
* Tecnología: Esta categoría está presente porque se han producido actualizaciones en las herramientas que se utilizaron para el desarrollo de la aplicación móvil de Tempus. Esto no había sido considerado previamente, por lo cual se analiza esta posibilidad. Como respuesta al riesgo se debe seguir reforzando el conocimiento sobre Ionic tanto para las versiones anteriores como en la nueva. Se deben estudiar los cambios para adaptar la versión anterior de nuestra aplicación a la versión actual de Ionic.
* Legal y Contractual: Esta categoría se tomó en cuanta, ya que durante el periodo del año 2020, surgió una pandemia que nos limitó en parte nuestro desarrollo del proyecto. Esto se debe a que se implementaron decretos que nos limitó a realizar reuniones presenciales del grupo. Nuestra manera de poder solucionarlo fue utilizar Zoom, pero como bien sabemos no es lo mismo realizar reuniones virtuales que están restringidas por tiempo a tener una reunión presencial en el cual es mucho más fácil evacuar duda o consultas entre los integrantes del grupo.

A continuación se muestra los gráficos de dispersión que se fueron realizando en cada iteración y como se fue tratando cada riesgo según su categoría

Gestión de Riesgo

En la siguiente evaluación de gestión de riesgo se realizaron los siguientes pasos:

Primero el proceso de Identificación y Evaluación de Riesgos el cual debía efectuarse al inicio de cada iteración del proyecto, siendo ésta una de las primeras tareas a planificar. Una vez que se realiza este proceso, se identifican las categorías de riesgos (compromiso del cliente, definición del cliente, cronograma, experiencia y capacidad, duración y tamaño, legal y contractual, tecnología y Contingencia) que tienen mayor impacto, esto a través de las preguntas se que encuentran en el anexo 1. Cada pregunta tendrá una valoración posible de 0 a 3, representando 0 el menor riesgo y 3 el máximo. Cada valor aplicable a una pregunta dada será indicado con una etiqueta representativa del significado de dicha valoración.

Al finalizar la generación de los distintos cuestionarios, se procederá a la puntuación de cada pregunta, en la columna “Puntos”. De manera automática se indicará en la columna siguiente la necesidad de analizar en mayor detalle la pregunta calificada, si la misma posee una valoración igual o mayor a 2.

Una vez concluida la evaluación, se obtendrá la puntuación total y el valor máximo de puntos posibles de obtener, los cuales serán colocados en la parte inferior de cada página.

A continuación deberá indicarse una valoración para el impacto que este grupo de riesgos en su conjunto representa para el proyecto, siendo 5 el más alto impacto y 1 el más bajo. Una vez seleccionado este valor, se obtiene automáticamente un rating (Puntuación/Puntos Posibles x Impacto), el cual servirá para determinar en una etapa posterior el riesgo total del proyecto para la etapa actual de trabajo.

Por último, se cuenta con un espacio para indicar los comentarios que se crean necesarios.

Una vez finalizada con la identificación y el análisis para cada uno de los grupos, se obtendrá una planilla resumen en la hoja “Resumen” del mismo documento “Gestión de Riesgos, Anexo 1”. En este se presenta información acerca de las distintas categorías de riesgos, su peso, impacto y rating. Obteniendo además, datos generales del análisis, conjuntamente de un gráfico tipo radial representativo de la situación.

### Fase de Inicio

En la fase de Inicio el objetivo es comprender el dominio del problema y capturar el conjunto correcto de requerimientos para llegar a un acuerdo con el cliente.

Al principio no se realizó la gestión de riesgo, ya que era uno de los objetivos a realizar en un periodo determinado. Por ello y otros objetivos se tuvo que realizar otra iteración para poder culminar con las tareas que fueron asignadas en esa iteración. Ahora bien, luego de realizar la identificación y evaluación de riesgos pudimos verificar aquellas categorías que tuvieron mayor impacto sobre esta fase las cuales fueron la experiencia y capacidad y el cronograma, ya que este tipo de proyecto no había sido empleado con anterioridad y no teníamos experiencia en desarrollo de soluciones similares, una forma de reducir este riesgo era comenzar a investigar, practicar y consultar a personas que hayan tenido experiencia y así poder disminuirlo. El equipo tenía poca experiencia en la metodología, si bien uno de nosotros tenía algo de conocimiento no era suficiente, entonces comenzamos hacer uso del psi y de bibliografía logrando así tener más conocimiento sobre el mismo, además de estas dificultades teníamos poca experiencia en el lenguaje que se iba a utilizar para el desarrollo, lo que hicimos fue invertir tiempo en la adquisición de conocimientos de dicho lenguaje, consiguiendo de esta manera poder manejarlo y alcanzar dicho objetivo.

En cuanto a la categoría de cronograma, nuestro riesgo se basaba en que contábamos con poco tiempo para la realización de las actividades, entonces para poder mitigar este riesgo lo que hicimos fue evaluar el tiempo de cada integrante, y así poder distribuir las tareas.







Fase Elaboración

En esta fase no se realizó ningún seguimiento de riesgo, ya que los integrantes del grupo consideramos que era necesario realizar otros requerimientos más importantes en ese momento, cabe destacar que como se dijo en la fase anterior, no contábamos con el tiempo suficiente, si bien organizamos los tiempos todavía no era suficiente.

En ese momento nos encontrábamos cursando la materia y debíamos entregar avances cada semana, logrando alcanzar los objetivos más importantes y así de esta manera los tiempos eran distribuidos según la necesidad más urgente.

Fase Construcción

En esta fase de Construcción, surgieron más riesgos de los cuales tuvimos que utilizar tipos de estrategias para poder eliminarlos, mitigarlos o realizar contingencia. Básicamente durante esta etapa se tuvo que utilizar tiempo para obtener conocimiento de HTML, CSS, JavaScript, IONIC y PHP para poder trabajar con ellos y utilizarlos para la realización del proyecto, de esta manera logramos mitigar los riesgos que nos pudieran surgir durante el desarrollo.

Durante el periodo de las iteraciones nos encontrábamos en las mesas de exámenes de la universidad, lo cual nos redujo el tiempo de trabajo que íbamos manejando y distribuyendo para poder terminar con los objetivos planificados, fue entonces que tuvimos que volver a organizar la forma de trabajo para llegar con los más importante en ese momento.

Luego de un largo periodo de no seguir con el proyecto, y comenzar de nuevo hasta donde habíamos quedado surgieron nuevos riesgos, que nos atrasaron más aun y es que dado el avance tecnológico que había surgido durante ese periodo sobre las actualizaciones que se había producido en las herramientas que se utilizaron para el desarrollo de la aplicación móvil TEMPUS, se tuvo que volver para atrás casi todo lo que se había avanzado en la aplicación, lo cual tuvimos que buscar una resolución en la actualización de dicho software, logrando de esta manera reducir el riesgo para que no volviera a ocurrir.

Por último, durante este periodo de construcción nos encontramos con un gran riesgo, que jamás tuvimos en cuenta, y es la pandemia (COVID-19) del año 2020. Este riesgo nos modificó la manera de poder trabajar en equipo, ya que el gobierno saco un decreto que no nos permitió seguir con las reuniones presenciales. Rápidamente buscamos una forma de poder resolver esto en aquel momento y fue el utilizar el servicio de videoconferencia ZOOM, y así poder realizar reuniones virtuales de acuerdo al tiempo de cada uno. Cabe aclarar que siempre es mejor la reunión presencial, el cual te permite organizarte mejor y el tiempo es manejado por el equipo.







### Conclusión

Durante las fases que fuimos realizando en este proyecto iterativo e incremental, podemos decir que no siempre encontramos todos los riesgos que pudieran surgir durante el desarrollo, lo que si podemos ir utilizando diferentes tipos de estrategia que no permita manejarlos. Pero, no sabemos bien con qué riesgo nos podemos encontrar en el camino dependiendo de su gravedad nos puede traer un retraso en el proyecto o bien en el peor de los casos volver hacer alguna tarea ya realizada. Debemos estar atentos a cualquier imprevisto que nos pudiera surgir, para poder resolverlo de la manera más rápida posible y así evitar perder tiempo y dinero si fuese necesario.

## Modelo de casos de uso

En la sección de Estimación se presentó el diagrama de casos de uso inicial. Aquí nos centramos en la definición de los casos de uso específicos y en el diseño de los diferentes diagramas que también son parte del modelo de diseño.

Se brindara una descripción genérica de los casos de uso y se detallaran aquellos que se consideraron con un nivel de prioridad más alto o en los que se hayan detectado inconvenientes.

Los casos de uso que son parte de UARGFlow no son considerados en este documento.

### Caso de Uso 01- Ingresar al Sistema

|  |  |
| --- | --- |
| **Actores** | Administrador, base de datos |
| **Descripción** | Este caso de uso es iniciado por el actor. Tiene la opción de ingresar al sitio web del sistema. |
|  |  |

### Caso de Uso 02 - Importar Horario de Cursada

Actores del CU

Administrador.

Secretaría Académica.

Descripción

Este caso de uso es iniciado por el actor. Tiene la opción de importar un archivo con los horarios de cursada.

### Caso de Uso 03 - Importar Mesa de Examen

Actores del CU

Administrador.

Secretaría Académica.

Descripción

Este caso de uso es iniciado por el actor. Tiene la opción de importar un archivo con mesas de examen.

### Caso de Uso 04 - Buscar Horario de Cursada

Actores del CU

1. Administrador.
2. Secretaría Académica.
3. Público

Descripción

El objetivo del caso de uso es realizar la búsqueda de horario de cursada. Este caso de uso se relaciona con los casos de uso borrar y modificar horario de cursada. Además, puede ser invocado por todos los actores del sistema.

Una característica importante de este caso de uso es que se puede invocar desde la aplicación móvil y el sitio web del sistema Tempus. En el caso del Administrador y Secretaría Académica se otorga el acceso desde el sitio web, mientras que el actor público accede desde la aplicación móvil.

### Caso de Uso 05 - Buscar Mesa de Examen

Actores del CU

1. Administrador.
2. Secretaría Académica.
3. Público.

Descripción

El objetivo del caso de uso es realizar la búsqueda de mesas de examen. Este caso de uso se relaciona con los casos de uso borrar y modificar mesa de examen. Además, puede ser invocado por todos los actores del sistema.

Una característica importante de este caso de uso es que se puede invocar desde la aplicación móvil y el sitio web del sistema Tempus. En el caso del Administrador y Secretaría Académica se otorga el acceso desde el sitio web, mientras que el actor público accede desde la aplicación móvil.

### Caso de Uso 06 - Modificar Horario de Cursada

Actores del CU

1. Administrador
2. Secretaria Académica
3. Base Datos

Descripción

Este caso de uso es iniciado por el actor. Tiene la opción de modificar los horarios de cursada. Se puede modificar la información básica de la asignatura como el nombre y el año al que pertenece en la carrera.

Además se puede modificar los horarios de clase (crear uno nuevo para un día determinado, borrar una clase de un día determinado, modificar una clase de un día determinado). En este caso, cuando se realiza la modificación de un horario de cursada se debe almacenar la fecha actual para permitir a la aplicación Tempus mostrar las actualizaciones en caso que corresponda.

### Caso de Uso 07 - Modificar Mesa de Examen

Actores del CU

1. Administrador
2. Secretaria Académica
3. Base Datos

Descripción

Este caso de uso es iniciado por el actor. Tiene la opción de ingresar al sitio web del sistema. Cuando se realiza una modificación a una determinada mesa de examen, se debe guardar la fecha actual con el objetivo de actualizar los datos correspondientes en los dispositivos móviles.

### Caso de Uso 08 - Crear Horarios de Cursada

Actores del CU

1. Administrador.
2. Secretaría Académica.

Descripción

El objetivo del caso de uso es crear horarios de cursada. Este caso de uso, puede ser invocado por el actor administrador y secretaría académica.

Una característica de este caso de uso es que se puede invocar desde el sitio web del sistema Tempus.

### Caso de Uso 09 - Crear Mesa de Examen

Actores del CU

1. Administrador.
2. Secretaría Académica.

Descripción

Este caso de uso es iniciado por el actor. Mediante este, un actor puede realizar la creación individual de una mesa de examen. Esto se da en casos donde la mesa de examen no ha sido cargada a través del archivo correspondiente.

### Caso de Uso 10 - Notificar

Actores del CU

1. Público.

Descripción

El objetivo del caso de uso es realizar notificaciones. Se relaciona con el caso de uso modificar horario de cursada y mesa de examen. El caso de uso es invocado indirectamente por el actor público cuando este guarda en favoritos algún horario de cursada o mesa de examen.

Una característica de este caso de uso es que el actor público accede a las notificaciones desde la aplicación móvil.

### Caso de Uso 11 - Informe de Mesa de Examen

Actores del CU

1. Administrador.
2. Secretaría Académica.

Descripción

El objetivo del caso de uso es que, el actor genere un informe de las aulas disponibles durante las fechas de examen.

### Caso de Uso 12 - Informe de Horario de Cursada

Actores del CU

1. Administrador.
2. Secretaría Académica.

Descripción

El objetivo del caso de uso es que, el actor genere un informe de las aulas disponibles durante el periodo de cursada.

### Caso de Uso 13 - Buscar Aula

Actores del CU

1. Administrador.
2. Secretaría Académica.

Descripción

El objetivo del caso de uso es realizar la búsqueda de un aula. Este caso de uso se relaciona con los casos de uso borrar y modificar aula. Además, puede ser invocado por todos los actores del sistema.

Una característica importante de este caso de uso es que se puede invocar desde el sitio web del sistema Tempus. En el caso del Administrador y Secretaría Académica se otorga el acceso desde el sitio web.

### Caso de Uso 14 - Borrar Horario de Cursada

Actores del CU

1. Administrador.
2. Secretaría Académica.

Descripción

La funcionalidad que abarca este caso de uso corresponde a la eliminación en la base de datos de un determinado horario de cursada. El actor debe seleccionar el horario de cursada que desea borrar.

### Caso de Uso 15- Borrar Mesa de Examen

Actores del CU

1. Administrador.
2. Secretaría Académica.

Descripción

La funcionalidad que abarca este caso de uso corresponde a la eliminación de la base de datos de una determinada mesa de examen. El actor debe seleccionar la mesa de examen que desea borrar.

### Caso de Uso 16- Modificar Aula

Actores del CU

1. Administrador.
2. Secretaría Académica.
3. Base de Datos.

Descripción

Este caso de uso es iniciado por el actor. Tiene la opción de modificar las aulas.

### Caso de Uso 17- Borrar Aula

Actores del CU

1. Administrador.
2. Secretaría Académica.

Descripción

Este caso de uso es iniciado por el actor. La funcionalidad que abarca este caso de uso corresponde a la eliminación en la base de datos de una determina aula. El actor debe seleccionar el aula que desea borrar.

### Caso de Uso 18- Generar Horario de Aula

Actores del CU

1. Administrador.
2. Secretaría Académica.

Descripción

Este caso de uso es iniciado por el actor. Permite realizar la generación de un informe que muestra los horarios en los que una determinada aula se encuentra ocupada por el dictado de clases.

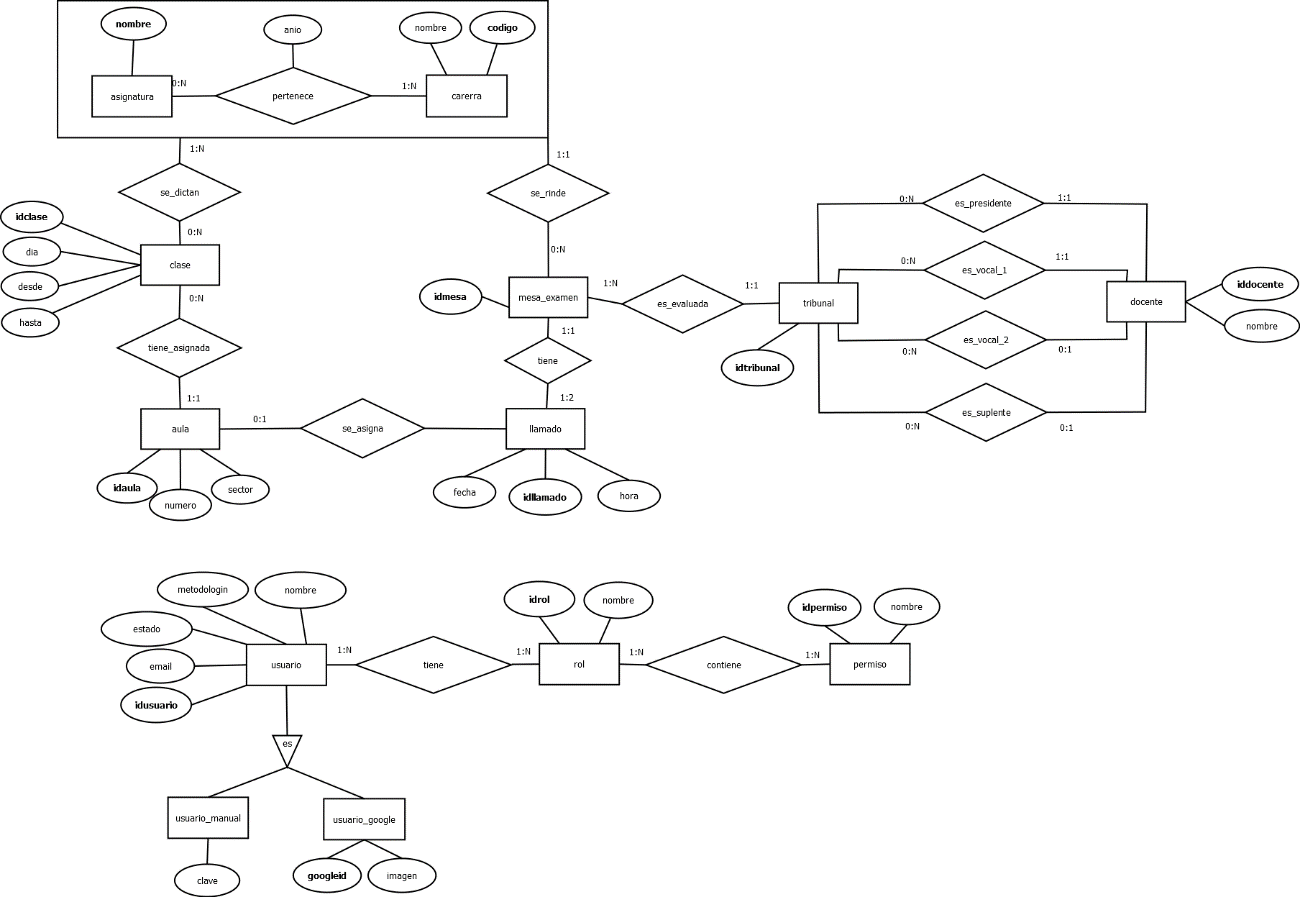
## Modelo de datos

El Modelo de Datos en un documento de gran importancia que nos permite comprender, interpretar, representar y procesar de manera abstracta todos los datos que se encuentran inmersos en el dominio del proyecto.

Para el desarrollo de este modelo se hicieron actividades como la detección de entidades y relaciones, generación del diagrama Entidad-Relación, creación del modelo relacional y la posterior creación de una base de datos en MySQL.

Se debe aclarar que las imágenes que se muestran en esta sección omiten el diseño dado por UARGFlow. Sin embargo, están presentes en los documentos correspondientes.

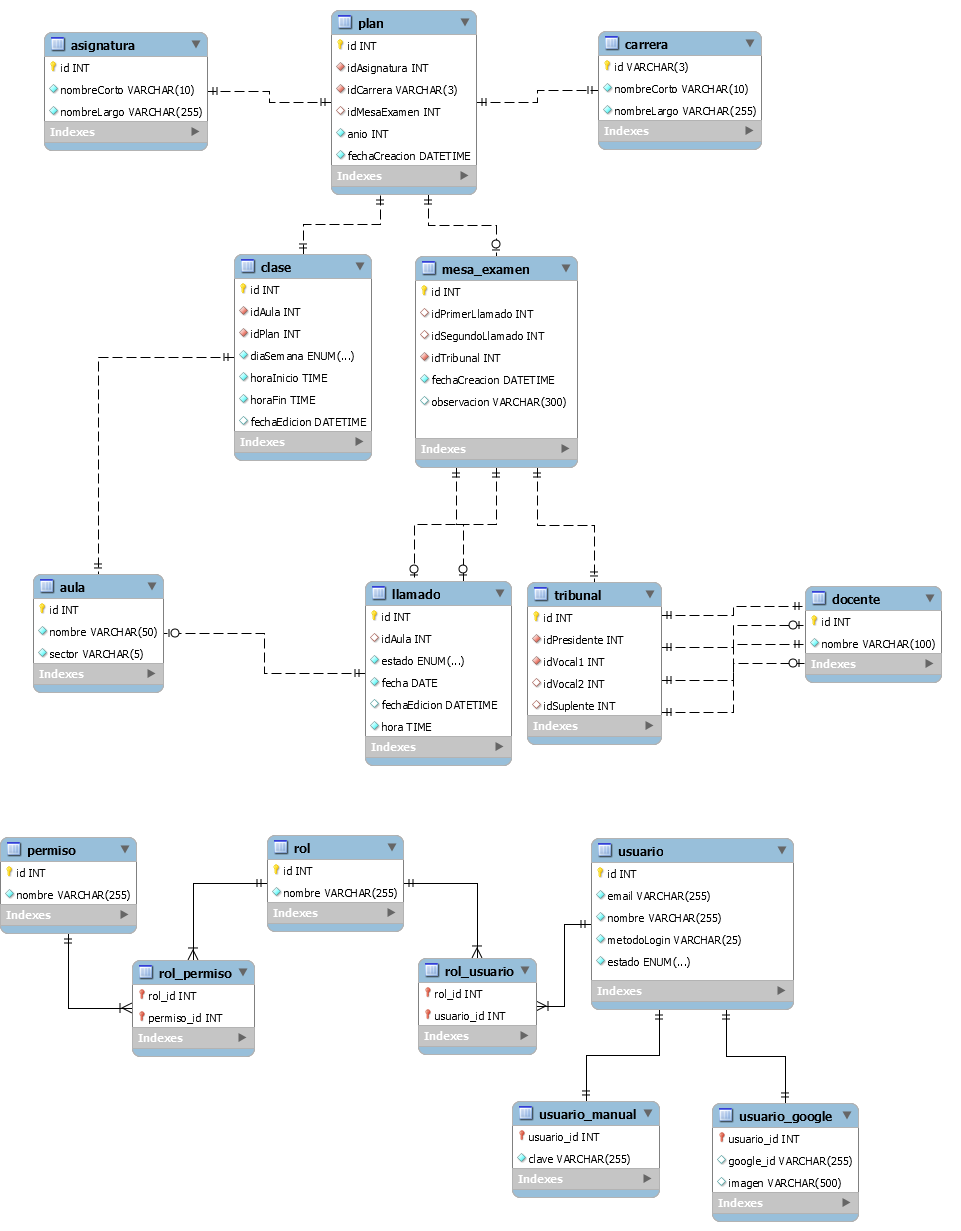
En la siguiente imagen se observa el Diagrama Entidad-Relación generado



Un problema que nos llevó mucho tiempo resolver es la relación entre asignaturas y carreras. Para resolver este problema en forma óptima se tuvo que acudir a información en línea, tutoriales e incluso consultas con profesores de base de datos. El origen del problema estaba dado en relacionar una asignatura a una o más carreras en un determinado año pero a la vez asociarle los horarios de clase y la mesa de examen correspondiente (relaciones n-aria). Luego de varios cambios en la base de datos se pudo solucionar este aspecto creando una entidad (Plan) que engloba la relación anterior.

Este concepto de Agregación se utiliza para construir objetos compuestos a partir de sus objetos componentes. Permite combinar entidades entre las que existe una interrelación y formar una entidad de más alto nivel. Es útil cuando la entidad de más alto nivel se tiene que interrelacionar con otra entidad.

El modelo relacional quedo establecido de la siguiente forma:



## Implementación

Al iniciar el proyecto una de los aspectos más complicados era la codificación porque en la carrera se aprendió a trabajar con Java y ninguno de los integrantes tenía experiencia en otros lenguajes de programación. Sin embargo, una de las primeras decisiones del grupo fue invertir tiempo en aprender cuestiones básicas de PHP y del framework Ionic. Este aspecto, junto con la creación de los prototipos desechables, fue un punto a favor a la hora de iniciar con la programación.

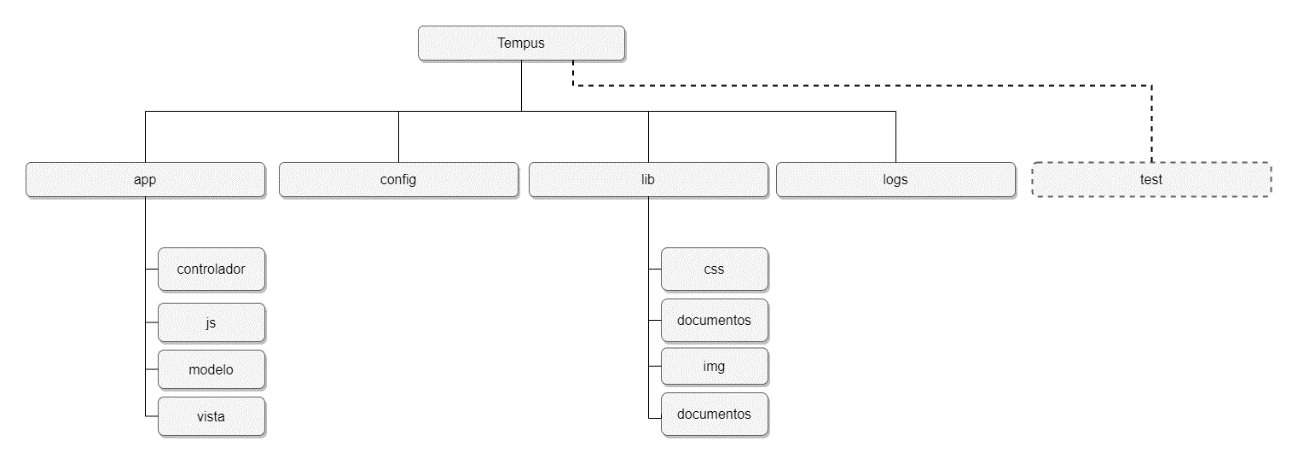
Durante el año 2017 se decidió trabajar con PHP 5.6 porque era la versión más estable (disponible en XAMPP) y con Ionic 2 que era la última versión disponible (lanzada en Febrero de ese año). Con el correr de los años se cambió a la versión 7 de PHP lo que produjo que se modificara la estructura de directorios de la página web para aprovechar nuevas funcionalidades y mejorar el funcionamiento del sistema. En cuanto a Ionic nos encontramos con algo diferente porque se avanzaron dos versiones en este periodo y lo que se encontraba desarrollado no se adaptaba a la arquitectura actual. Si bien la lógica de la aplicación era la misma, se tuvieron que hacer grandes cambios y la documentación disponible para aprender no era amplia.

Una dificultad que se presentó al momento de realizar las pruebas eran los errores que se daban por tener un único directorio versionado con el código. Tanto para la web como para la aplicación se contaba con un directorio en donde se dejaba el código desarrollado para las pruebas. Cuando el desarrollador dejaba disponible el código continuaba trabajando lo que ocasionaba que el tester tuviera inconvenientes en el módulo que debía probar o incluso en otros. Esto se solucionó con la creación de tres directorios y modificando el proceso implementación-pruebas:

* Desarrollo: Este es código fuente en donde el desarrollador trabaja. Cuando se agrega, se quita o se modifica el código lo hace en esta carpeta. Las pruebas informales incluso se pueden dejar aquí (pruebas del codificador).
* Test: En esta carpeta se deja solo el código funcionando y necesario para realizar las pruebas. Este directorio se debe copiar el tester para ejecutar los casos de prueba.
* Producción: El código que funciona correctamente queda disponible en esta carpeta. Solo superadas las pruebas se puede actualizar esta carpeta.

El proceso queda definido de la siguiente manera:

En la imagen que se presenta a continuación se muestra la estructura de directorios del proyecto Tempus para la web.



Este diagrama es de utilidad para observar de forma rápida donde se encuentran cada una de las partes que componen el sistema. En cuanto a la carpeta Test, solo se hizo uso para la etapa de desarrollo y pruebas pero no forma parte de la entrega final.

## Pruebas

Explicar

# Tecnologías

## Hardware

En esta sección haremos una descripción de las tecnologías utilizadas de hardware y software durante el desarrollo de proyecto, la cuales fueron utilizadas para su realización.

|  |  |
| --- | --- |
| *Hardware* |  |
| *Samsung J2 Prime* | *Sistema Operativo:* Android 6.0.1  *Pantalla:* 5 PS *Resolución:* 540 x 960 PX  *RAM:* 1.5 GB  *Almacenamiento interno:* 8 GB  *Tipos de tarjetas de Memoria:* Micro SD  *Cámara principal:* 8 MP  *Cámara secundaria :* 5MP  *Batería:* 2600 mAh  *Procesador:* MediaTek MT6737T de cuatro núcelos a 1.4 GHz |
| *Motorola g (6)play* | *Sistema Operativo:* Android 9  *Pantalla:* 5.7 PS *Resolución:* 1,440x720 HD+  *RAM:* 3GB  *Almacenamiento interno:* 32GB (expandible)  *Tipos de tarjetas de Memoria:* Micro SD  *Cámara principal:* 8 MP  *Cámara secundaria :* 13 MP con PDAF, f/2.0  *Batería:* 4,000 mAh  *Procesador:* Snapdragon 430 (ocho núcleos 1.4GHz) |
| *Samsung Galaxy S9* | *Sistema Operativo:* Android 10  *Pantalla:* 5.8 PS *Resolución:* 2,960 x 1,440  *RAM:* 4GB  *Almacenamiento interno:* 64 GB (expandible)  *Tipos de tarjetas de Memoria:* Micro SD  *Cámara principal:* 12 MP  *Cámara secundaria :* 8 MP  *Batería:* 3,000 mAh  *Procesador:* Snapdragon 845 (ocho núcleos cuatro 2.8 GHz) |
| *Samsung J7 Prime SM-G610M* | *Sistema Operativo:* Android 8.10  *Pantalla:* 5.5 PS *Resolución:* 1080x1920  *RAM:* 3GB  *Almacenamiento interno:* 8GB  *Tipos de tarjetas de Memoria:* Micro SD  *Cámara principal:* 13 MP  *Cámara secundaria :* 8 MP  *Batería:* *batería 3300 mAh*  *Procesador: 1.6Ghz Octa Core ARM Cortex-A53* |
| *Notebook hp pavilion* | *Procesador:* Intel (R) Core(TM) i5-6200u CPU@ 2.30 GHZ 2.40 GHZ  *Memoria Ram:*12 GB |
| *Notebook Toshiba* | *Procesador:* Intel Core i3-3227U 1.90 GHz  *Memoria Ram:* 4 GB |
| *Notebook hp pavilion dv6* | *Procesador:* AMD A8-3500M APU 1.50 GHz  *Memoria Ram:* 8 GB |
|  |  |

## Software

### GitHub

Es una plataforma de desarrollo inspirada en su forma de trabajar. Desde el código abierto hasta el negocio puede alojar y revisar código, administrar proyectos y crear software junto con 50 millones de desarrolladores.

### Ionic Framework

Es una estructura tecnológica (Framework) de código abierto que se utiliza en el desarrollo de aplicaciones móviles hibridas, es decir, se combinan el HTML 5, CSS y JavaScript dando como resultado aplicaciones con una interfaz amigable e intuitiva para el usuario que luego se comercializan o descargan en plataformas como Android o IOs.

### Netbeans

Es un entorno de desarrollo integrado libre, hecho principalmente para el lenguaje de programación Java. Existe además un número importante de módulos para extenderlo. NetBeans IDE es un producto libre y gratuito sin restricciones de uso.

El IDE se usara para desarrollar la aplicación web del proyecto utilizando la extensión de PHP. Además se hará uso de las extensiones necesarias para facilitar el trabajo.

### Taiga

Es una herramienta de gestión de proyectos para equipos ágiles multifuncionales. Tiene un amplio conjunto de funciones y, al mismo tiempo, es muy sencillo comenzar a través de su interfaz de usuario intuitiva.

Scrum es una metodología de desarrollo de software ágil iterativa e incremental para gestionar el desarrollo de productos.

* El trabajo acumulado del producto es lo que finalmente se entregará, ordenado en la secuencia en la que se debe entregar. Los pedidos acumulados de productos se dividen en fragmentos manejables y ejecutables llamados sprints.
* Cada cierto tiempo, el equipo inicia un nuevo sprint y se compromete a entregar una cierta cantidad de historias de usuarios del trabajo acumulado, de acuerdo con sus habilidades, capacidades y recursos. El proyecto avanza a medida que se agota el trabajo atrasado.

La metodología **Kanban** se utiliza para dividir el desarrollo del proyecto (cualquier tipo de proyecto) en etapas.

* Una tarjeta kanban es como una tarjeta de índice o una nota post-it que detalla cada tarea (o historia de usuario) en un proyecto que debe completarse. El tablero Kanban se usa para mover cada carta de un estado de finalización al siguiente y, al hacerlo, ayuda a rastrear el progreso.
* El proyecto avanza tan rápido como el tiempo promedio de ciclo de vida de una tarjeta.

Las actividades planificadas en cada iteración son cargadas en Taiga para que cada uno de los integrantes del grupo pueda monitorear los avances y el equipo docente verificar el avance del proyecto.

### Tortoise SVN:

Es un cliente Apache Subversion, implementado como una extensión de shell de Windows. Es intuitivo y fácil de usar, ya que no requiere que se ejecute el cliente de línea de comandos de Subversion. Y es de uso gratuito, incluso en un entorno comercial.

Se hará uso de Tortoise SVN para administrar los cambios y actualizaciones del repositorio del proyecto.

### UARGFlow

Es un framework desarrollado por alumnos y docentes de la universidad que permite adoptar el login por medio de un correo de Google y que nos marca una línea con respecto a formato de ventanas y estándar de codificación PHP.

### Visual Studio Code

Es un editor de código desarrollado por Microsoft para Windows, Linux y MacOS. Incluye soporte para la [depuración](https://es.wikipedia.org/wiki/Depuraci%C3%B3n_de_programas), control integrado de GIT, resaltado de sintaxis, finalización inteligente de código, fragmentos y refactorización de código. También es personalizable, por lo que los usuarios pueden cambiar el tema del editor, los atajos de teclado y las preferencias. Es gratuito y de código abierto, aunque la descarga oficial está bajo [software privativo](https://es.wikipedia.org/wiki/Software_privativo) e incluye características personalizadas por [Microsoft](https://es.wikipedia.org/wiki/Microsoft).

Es compatible con varios lenguajes de programación y un conjunto de características que pueden o no estar disponibles para un idioma dado. Muchas de las características de Visual Studio Code no están expuestas a través de los menús o la interfaz de usuario. Más bien, se accede a través de la paleta de comandos o a través de archivos.

WhatsApp

Es una [aplicación de mensajería instantánea](https://es.wikipedia.org/wiki/Aplicaciones_de_mensajer%C3%ADa) para [teléfonos inteligentes](https://es.wikipedia.org/wiki/Tel%C3%A9fono_inteligente), en la que se envían y reciben mensajes mediante [Internet](https://es.wikipedia.org/wiki/Internet), así como imágenes, vídeos, audios, grabaciones de audio (notas de voz), documentos, ubicaciones, contactos, gifs, así como llamadas y videollamadas con varios participantes a la vez, entre otras funciones

### XAMPP

**E**s una distribución de Apache completamente gratuita y fácil de instalar que contiene MariaDB, PHP y Perl. El paquete de instalación de XAMPP ha sido diseñado para ser increíblemente fácil de instalar y usar.

El programa se distribuye con la licencia [GNU](https://es.wikipedia.org/wiki/GNU) y actúa como un servidor web libre, fácil de usar y capaz de interpretar páginas dinámicas. A esta fecha, XAMPP está disponible para [Microsoft Windows](https://es.wikipedia.org/wiki/Microsoft_Windows), [GNU/Linux](https://es.wikipedia.org/wiki/GNU/Linux), [Solaris](https://es.wikipedia.org/wiki/Solaris_(sistema_operativo)) y [Mac OS X](https://es.wikipedia.org/wiki/Mac_OS_X).

Se hace uso de XAMPP para desplegar el sitio web y administrar las características de la base de datos.

### XDEBUG

Es una extensión de PHP que proporciona al desarrollador ciertas características para la depuración. Dichas características son:

* Seguimientos de pila. Salida detallada de la ruta que se siguió, incluyendo parámetros enviados a las funciones llamadas.
* Una salida al estilo VAR\_DUMP que produce información estructurada.
* Generador de perfiles para detectar cuellos de botella en la aplicación. Esto se puede visualizar con una herramienta externa como KCacheGrind o WinCacheGrind.
* Un depurador remoto que se puede utilizar para establecer una conexión entre XDebug de forma remota con un cliente.

XDebug le permite al desarrollador de software romper la ejecución del código e inspeccionar todas las variables en el alcance durante una solicitud. Se puede decir que durante una sola iteración es posible encontrar problemas sin llenar el código de alertas o logs. Básicamente, ahorra una gran cantidad de tiempo al momento de rastrear problemas y ayuda a que el flujo de trabajo sea más eficiente.

Esta herramienta encuentra su mayor utilidad al momento de encontrar y solucionar un problema cuya causa es totalmente desconocida para el programador. Muchas veces, es posible saber la causa de un problema y rastrearlo en forma sencilla, pero un comportamiento extraño causa un análisis mayor. Los puntos de interrupción que brinda XDebug permiten pausar el código en cualquier momento haciendo que el seguimiento sea mucho más sencillo.

Para más información se debe observar el Manual de configuración de Xdebug que queda disponible en el repositorio del proyecto.

### Zoom

Es una herramienta de **videoconferencias**. Este programa unifica las videoconferencias en la nube, las reuniones en línea sencillas, la mensajería de grupo y una solución de sala de conferencias definida por software en una plataforma fácil de usar.

Afortunadamente, ofrece la mejor experiencia de video, audio y pantalla inalámbrica para compartir a través de Windows, Mac, iOS, Android, Blackberry, Linux, Salas Zoom y sistemas de salas H. 323/SIP.

El **programa Zoom** no solamente ofrece llamadas a través de video sino además permite escribir mensajes al mismo tiempo y compartir cualquier archivo con los demás usuarios. La interfaz es muy fácil de utilizar y no se necesita unirse a una nueva red social para poder usarla ya que se puede registrar utilizando Gmail o una cuenta de Facebook.

# Conclusión

## Experiencia personal

### Oyarzo Mariela

En este proyecto realizado, hubo acontecimientos tanto buenos como malos, comienzo por los buenos. El grupo que me toco es de amigos, lo cual fue mucho más cómodo y fácil de trabajar, acá pude utilizar todo el conocimiento que he adquirido a lo largo de los años que llevo estudiando la carrera Analista de Sistema, este proyecto me ayudo a entender más y enriquecer aún más dichos conocimiento y poder llevar a cabo el desarrollo a través de toda la documentación que fuimos realizando a lo largo del proyecto, y así poder cumplir con los plazos requeridos en las estimaciones realizadas, que a su vez es costo y calidad. Si bien realizamos la estimación el tiempo no era real, ya que tuvimos ciertos contratiempos durante todos estos años tanto personal como grupal, esto es algo negativo que tuvimos al igual que el tiempo utilizado para aprender algunas herramientas que nunca había utilizado. Pero es más positivo lo que hay que destacar, ya que esta experiencia nos permite formarnos mejor aún como futuros profesionales de esta carrera que elegimos

### Quiroga Sandra

Texto de Sandra.

### Marquez Emanuel

Durante el año 2014 había participado del grupo de desarrollo Koru encargado del proyecto ProMex. Si bien el equipo de desarrollo había logrado la regularización de la materia, a pesar de contar con un integrante menos, no se pudo finalizar el proyecto ni presentar el examen final de la materia. Particularmente esa experiencia me sirvió para tomar este proyecto como un desafío personal tratando de no repetir los errores anteriores.

Teniendo en cuenta lo mencionado, ya había consultado quienes iban a ser parte de la materia antes de iniciar para tratar de formar un grupo de trabajo comprometido. El primer día de cursada ya tenía pensado formar grupo con Oyarzo Mariela y Quiroga Sandra porque con ambas había compartido materias previamente y conocía su forma de ser y su forma de trabajar. También íbamos a compartir otras materias durante ese mismo cuatrimestre (Sistemas Operativos Distribuidos y Gestiona de Proyectos) lo que hizo que la selección de los integrantes del grupo sea realmente sencilla.

Al momento de la definición de los roles me toco ser designado como Líder de Proyecto, tal vez por mi participación previa en la asignatura. Si bien siento que en parte del proyecto pude cumplir con el rol, también es cierto que tanto Mariela como Sandra cumplieron la función en diferentes momentos. Esta situación fue más notoria al finalizar la cursada donde, por diferentes motivos, ellas se pusieron adelante del proyecto para continuar avanzando.

Algo que debo destacar es que desde el comienzo hubo predisposición de mis compañeras para tratar de tomar las mejores decisiones en forma grupal sin anteponer lo personal. Todas las definiciones o decisiones se tomaban entre los tres y creo que ese es uno de los puntos más fuertes del grupo. Esto se veía reflejado en muchas de las presentaciones donde el equipo docente destacaba algunos aspectos de nuestro trabajo en cosas como: la definición de un formato de presentación, la división de los tiempos para hablar o preparar videos por cualquier inconveniente que pudiera ocurrir.

En cuanto a mi opinión, creo la materia tiene como objetivo integrar todos los conocimientos que se obtienen a lo largo de la carrera y aplicarlo en un proyecto, pero un aspecto que considero igual de valioso es la posibilidad de trabajar en grupo. Son pocos los momentos en los que se comparten trabajos en grupo y muchos menos en periodos largos. Esto me sirvió para poder delegar tareas, confiar en mis compañeras e incluso por momentos, relajarme de más.

## Experiencia grupal

Una de los beneficios a nivel grupal es, contar con un integrante que tenía conocimientos y experiencia previa en la asignatura, tanto en la forma de trabajo como en las herramientas dispuestas por el equipo de cátedra.

Se ha desarrollado la cursada sin conflictos ni problemas entre los integrantes del grupo. Por el contrario, se ha generado un buen ámbito de trabajo con compañerismo.

Se trató de aprovechar al máximo cada presentación de los demás grupos, permitiéndonos adoptar algunos puntos necesarios junto con los consejos de docentes, y esto nos ayudaría a mejorar el proyecto.

Se destaca la disposición de los profesores durante el desarrollo de todo el proyecto, ya que a través de cada entrevista podíamos solucionar los problemas que iban surgiendo o bien mejorar algún tema en particular del mismo.

Cada uno de los integrantes pudo adquirir experiencia con respecto al trabajo en grupo en un software real. Se ha comprendido en forma global los conceptos aprendidos durante el desarrollo de la carrera.

# Anexo

***Diagrama de Grantt***

En el siguiente grafico podemos ver que en el año 2017, llegamos a realizar fase de inicio, elaboración y construcción (hasta la iteración 3), cada una con sus respectivas iteraciones, las cuales fueron realizadas durante el periodo de desarrollo del proyecto. Cada iteración tenía una planificación de actividades, estas se realizaban de acuerdo a cierta prioridad alta, media o baja, ya que debíamos darle cierta prioridad para algunas actividades que considerábamos que era más importante para mostrar en ese momento. En la fase de inicio iteración 1, debemos destacar que no cumplimos con el objetivo de realizar la gestión de riesgo. Motivo por el cual no se realizo un seguimiento del mismo, esto se explica en la sección gestión de riesgo. Durante las demás iteraciones estuvimos haciendo documentos que fueron planificado según su fase hasta llegar a la etapa de construcción iteración 2, que se comenzó con las implementaciones de los casos de usos hasta llegar a la iteración 7, donde se realizo una evaluación tanto del proyecto como los riesgos, cabe mencionar que no realizamos un seguimiento del mismo en la etapa de elaboración. Esto permitió encontrar un nuevo riesgo y es que al volver a rever lo hecho hasta el momento, surgió el problemas con la actualización de Ionic que nos hizo volver atrás con el desarrollo de la APP, nos vimos en la necesidad de poder reducir ese riesgo para que no vuelva a suceder en un futuro. En resumen se trabajo más días en el año 2017 siendo estos 80 días y es porque se realizaba el proyecto en paralelo con la cursada de la materia, luego en el 2018 retomamos, con un total de 45 días donde tuvimos que releer todo lo que se había realizado en el proyecto . Continuamos en el año 2019 donde pudimos trabajar 30 días. Por último nos encontramos en el año 2020, el cual nos perjudico en cuanto a las reuniones presenciales y el tiempo de cada uno, ya que tuvimos que seguir el proyecto en forma virtual dependiendo del tiempo de los integrantes y de internet motivo por el cual algunas veces tuvimos que cancelar reuniones a causa de la conexión, de igual manera utilizamos ZOOM, que tiene un tiempo determinado y más de una vez tuvimos que realizar varias reuniones puesto que el tiempo no alcanzaba para solventar las dudas de cada integrante.

Biblografia

<https://jorgesaiz.com/blog/estructura-organizacional-en-proyectos/#:~:text=de%20sus%20implicaciones.-,Estructura%20organizacional%20en%20proyectos,el%20desarrollo%20de%20tus%20proyectos.>

Libro El Proceso Unificado del Desarrollo de Software – Jacobson, Booch y Rumbaugh.