

Memoria

TEMPUS



**Autores:**

Oyarzo, Mariela

Quiroga, Sandra

Márquez, Emanuel

**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 6](#_Toc56460175)

[Objetivos 6](#_Toc56460176)

[Objetivos Generales 6](#_Toc56460177)

[Objetivos Específicos 7](#_Toc56460178)

[Proceso de desarrollo 9](#_Toc56460179)

[Grupo de desarrollo 9](#_Toc56460180)

[Nombre y logo del grupo 9](#_Toc56460181)

[Nombre y logo del sistema 10](#_Toc56460182)

[Comunicación 11](#_Toc56460183)

[Descripción de los procesos actuales 13](#_Toc56460184)

[Descripción del proceso con Tempus 14](#_Toc56460185)

[Planificación 15](#_Toc56460186)

[Estimación 19](#_Toc56460187)

[Gestión de Riesgos 23](#_Toc56460188)

[Categorización 24](#_Toc56460189)

[Proceso 25](#_Toc56460190)

[Fase de Inicio 25](#_Toc56460191)

[Fase Elaboración 27](#_Toc56460192)

[Fase Construcción 28](#_Toc56460193)

[Modelo de casos de uso 30](#_Toc56460194)

[Caso de Uso 01- Ingresar al Sistema 30](#_Toc56460195)

[Caso de Uso 02 - Importar Horario de Cursada 30](#_Toc56460196)

[Caso de Uso 03 - Importar Mesa de Examen 32](#_Toc56460197)

[Caso de Uso 04 - Buscar Horario de Cursada 35](#_Toc56460198)

[Caso de Uso 05 - Buscar Mesa de Examen 36](#_Toc56460199)

[Caso de Uso 06 - Modificar Horario de Cursada 36](#_Toc56460200)

[Caso de Uso 07 - Modificar Mesa de Examen 36](#_Toc56460201)

[Caso de Uso 08 - Crear Horarios de Cursada 36](#_Toc56460202)

[Caso de Uso 09 - Crear Mesa de Examen 37](#_Toc56460203)

[Caso de Uso 10 - Notificar 37](#_Toc56460204)

[Caso de Uso 11 - Informe de Mesa de Examen 37](#_Toc56460205)

[Caso de Uso 12 - Informe de Horario de Cursada 38](#_Toc56460206)

[Caso de Uso 13 - Buscar Aula 38](#_Toc56460207)

[Caso de Uso 14 - Borrar Horario de Cursada 38](#_Toc56460208)

[Caso de Uso 15- Borrar Mesa de Examen 39](#_Toc56460209)

[Caso de Uso 16- Modificar Aula 39](#_Toc56460210)

[Caso de Uso 17- Borrar Aula 39](#_Toc56460211)

[Caso de Uso 18- Generar Horario de Aula 39](#_Toc56460212)

[Modelo de datos 40](#_Toc56460213)

[Implementación 41](#_Toc56460214)

[Pruebas 42](#_Toc56460215)

[Tecnologías 43](#_Toc56460216)

[Hardware 43](#_Toc56460217)

[Oyarzo Mariela 43](#_Toc56460218)

[Quiroga Sandra 43](#_Toc56460219)

[Marquez Emanuel 43](#_Toc56460220)

[Software 44](#_Toc56460221)

[GitHub 44](#_Toc56460222)

[Google Meet 44](#_Toc56460223)

[Ionic Framework 44](#_Toc56460224)

[Netbeans 44](#_Toc56460225)

[Taiga 45](#_Toc56460226)

[Tortoise SVN: 45](#_Toc56460227)

[UARGFlow 45](#_Toc56460228)

[Visual Studio Code 46](#_Toc56460229)

[WhatsApp 46](#_Toc56460230)

[XAMPP 46](#_Toc56460231)

[XDEBUG 46](#_Toc56460232)

[Zoom 47](#_Toc56460233)

[Conclusión 47](#_Toc56460234)

[Experiencia personal 47](#_Toc56460235)

[Oyarzo Mariela 47](#_Toc56460236)

[Quiroga Sandra 48](#_Toc56460237)

[Marquez Emanuel 48](#_Toc56460238)

[Experiencia grupal 49](#_Toc56460239)

[Anexo 49](#_Toc56460240)

Tabla de ilustraciones

[Ilustración 1 - Logo del grupo de desarrollo 10](#_Toc56460241)

[Ilustración 2 - Logo del sistema 11](#_Toc56460242)

[Ilustración 3 - Definición de roles y flujo general de comunicación 12](#_Toc56460243)

[Ilustración 4 - Proceso para la generación de horarios de cursada 13](#_Toc56460244)

[Ilustración 5 - Proceso para la generación de mesas de examen 14](#_Toc56460245)

[Ilustración 6 - Planificación de tareas en el Plan de Iteración Fase Inicio Iteración 1 16](#_Toc56460246)

[Ilustración 7 - Objetivos establecidos en el Plan de Iteración Fase Inicio Iteración 1 17](#_Toc56460247)

[Ilustración 8 - Planificación de tareas en el Plan de Iteración Fase Construcción Iteración 7 18](#_Toc56460248)

[Ilustración 9 - Planificación de Iteraciones 19](#_Toc56460249)

[Ilustración 10 - Diagrama de casos de uso 20](#_Toc56460250)

[Ilustración 11 - Trabajo realizado por Yenú (GitHub) 20](#_Toc56460251)

[Ilustración 12 - Trabajo realizado por Yenú primera etapa 21](#_Toc56460252)

[Ilustración 13 - Trabajo realizado por Yenú segunda etapa 22](#_Toc56460253)

[Ilustración 14 - Trabajo realizado por Yenú tercera etapa 22](#_Toc56460254)

[Ilustración 15 - Trabajo realizado por Yenú cuarta etapa 23](#_Toc56460255)

[Ilustración 16 - Resumen de riesgos Iteración N° 2 Fase de Inicio 26](#_Toc56460256)

[Ilustración 17 - Diagrama de paquetes para el Caso de Uso 02 31](#_Toc56460257)

[Ilustración 18 - Diagrama de secuencia para el Caso de Uso 02 31](#_Toc56460258)

[Ilustración 19 - Diagrama de flujo de eventos para el Caso de Uso 02 32](#_Toc56460259)

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.

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, las tecnologías que se utilizaron, las experiencias personales de los integrantes así como una conclusión grupal. Adicionalmente, se presentan los resúmenes de reunión en el Anexo.

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 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 conocimientos durante el proceso de desarrollo del proyecto Tempus.

# Proceso de desarrollo

Al utilizar el proceso que propone PSI se siguen una serie de fases (Inicio, Elaboración, Construcción y Transición) en las que sucede cada una de las iteraciones planificadas. En el presente documento no se pretende detallar la metodología PSI. Por el contrario, se presenta un análisis 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.



Ilustración 1 - Logo del grupo de desarrollo

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.



Ilustración 2 - Logo del sistema

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 es un aspecto clave en cualquier grupo y es el fundamento del trabajo en equipo. A través de la comunicación el grupo puede unir conocimientos, mejorar habilidades, corregir errores, descubrir debilidades y potenciar las fortalezas. En forma introductoria, se muestra un diagrama general del flujo de comunicación en nuestro proyecto el cual también incluye la definición de los roles de cada miembro.

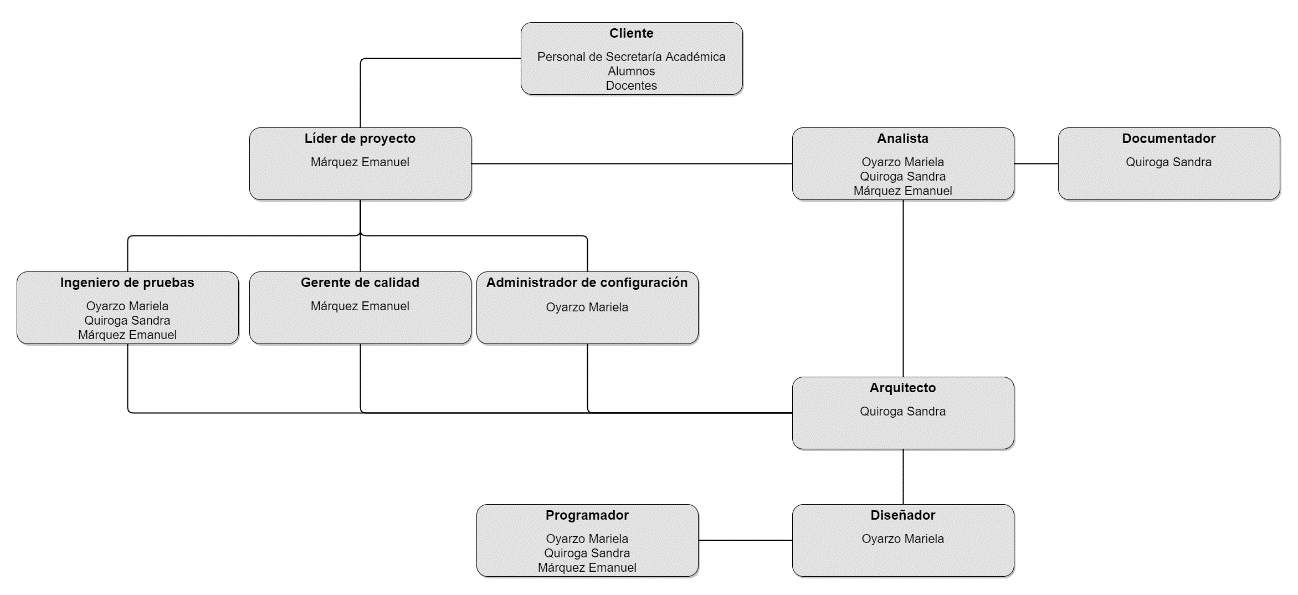


Ilustración 3 - Definición de roles y flujo general de comunicación

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, y Drive, entre otras.

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 electronico 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 horarios de cursada y generación de mesas 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.

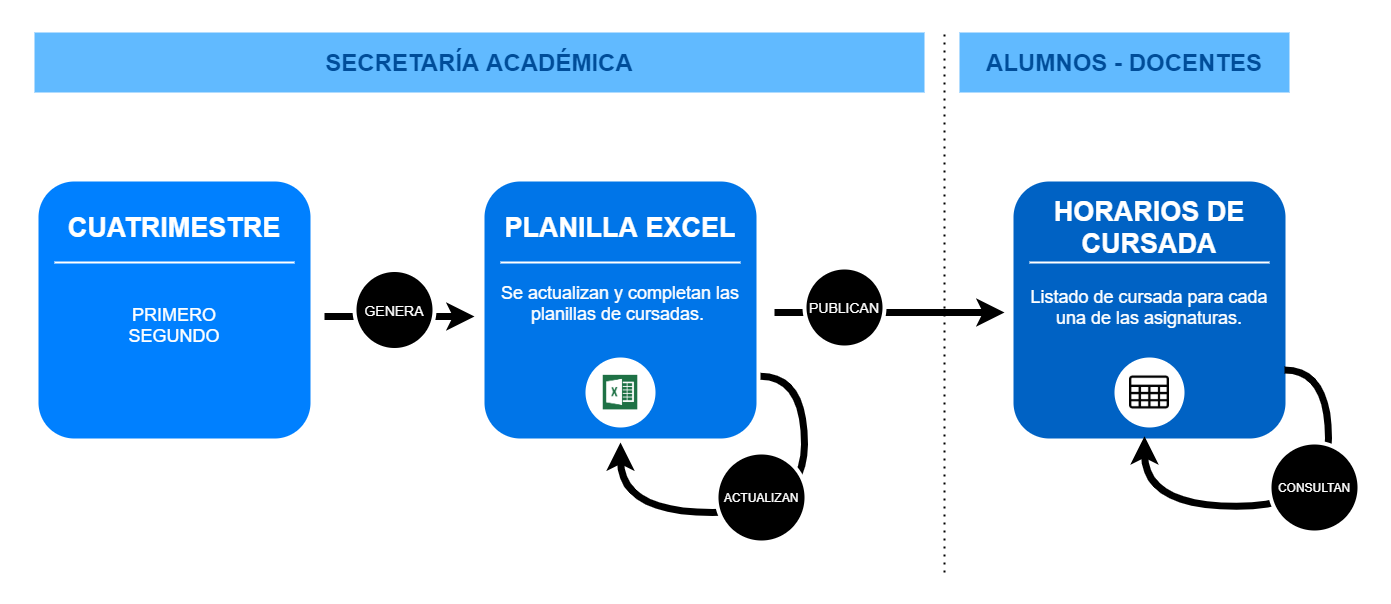


Ilustración 4 - Proceso para la generación de horarios de cursada

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.

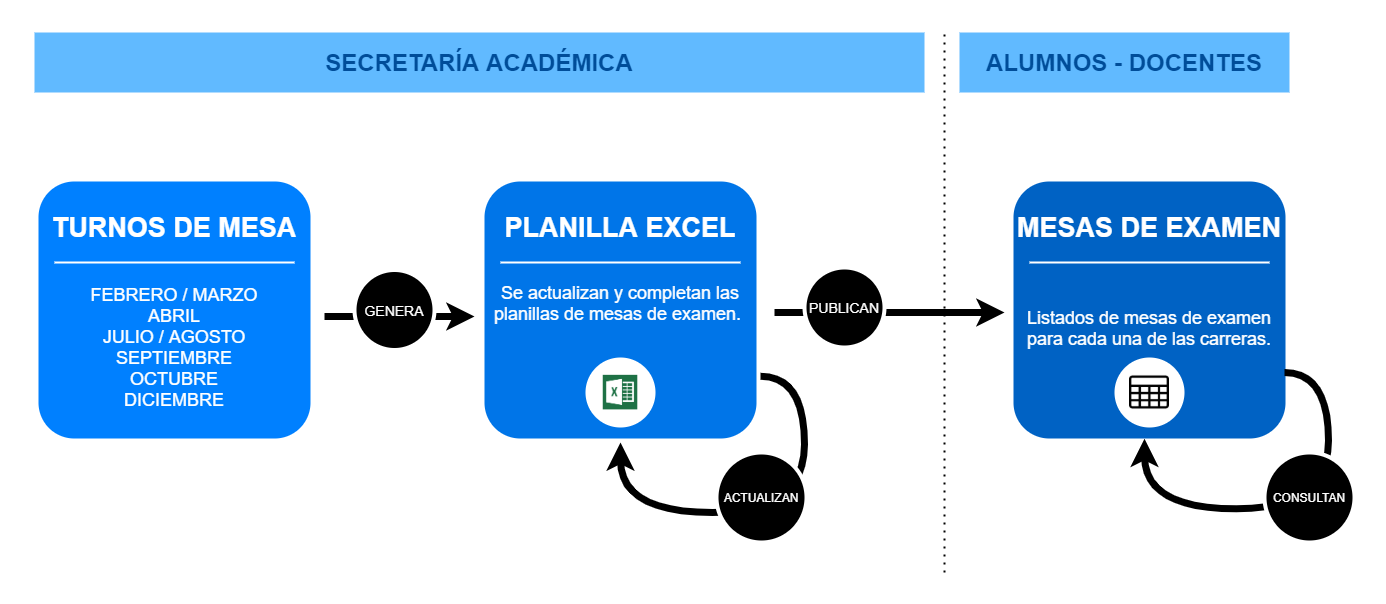


Ilustración 5 - Proceso para la generación de mesas de examen

## 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. Observaremos esto analizando nuestra primera Iteración y comparándola con otra más avanzada en el tiempo.

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:



Ilustración 6 - 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 definió, 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).



Ilustración 7 - 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 las futuras iteraciones:

* 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 que incluye lo mencionado:



Ilustración 8 - Planificación de tareas en el Plan de Iteración Fase Construcción Iteración 7

En forma de resumen, se presenta la siguiente imagen que permite observar todas las planificaciones que se hicieron y de las existe un Plan de Iteración. El periodo de tiempo que se abarca es desde 2017 hasta la actualidad en donde se siguieron las fases establecidas en PSI.

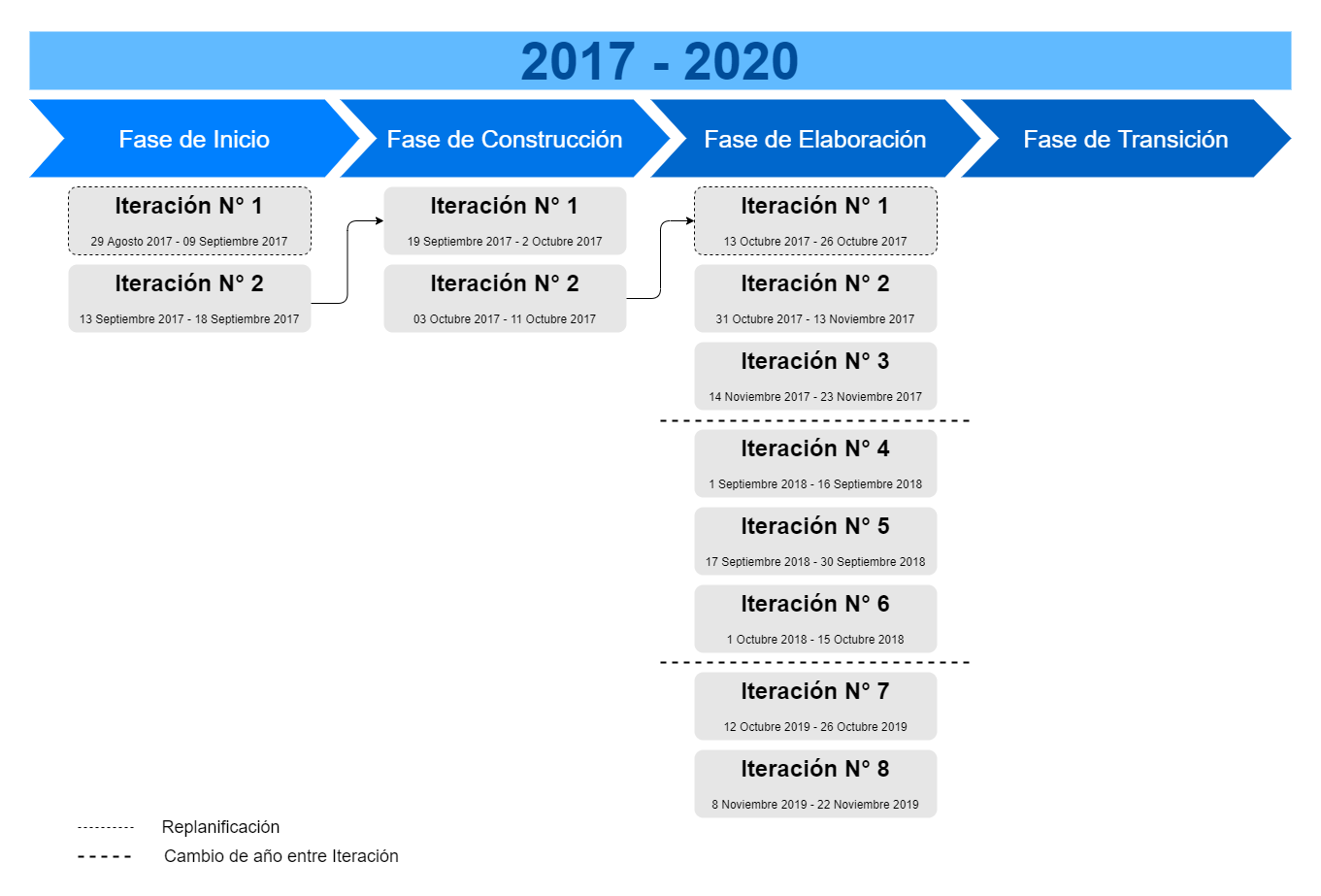


Ilustración 9 - Planificación de Iteraciones

Si bien existieron inconvenientes en la planificación por la falta de experiencia del grupo, se pudo avanzar en forma correcta con la documentación de los planes e incluso se hicieron adaptaciones, como el ordenamiento y priorización de las tareas, que beneficiaron al grupo. Se debe mencionar que en determinados momentos en donde se buscaba avanzar el proyecto, los planes de iteración se omitieron. A corto plazo esto servía para poder invertir el tiempo en otras tareas más prioritarias pero a largo plazo genera más problemas que soluciones porque no queda registro de lo realizado.

## 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 la complejidad del sistema, de la funcionalidad, la 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. Dicha estimación resulta de calcular en términos generales 14 Casos de Uso, sin tener en cuenta la reutilización del módulo de UARGFlow, de los cuales: 2 de ellos son complejos, 4 de tipo promedio y 8 de tipo simple.

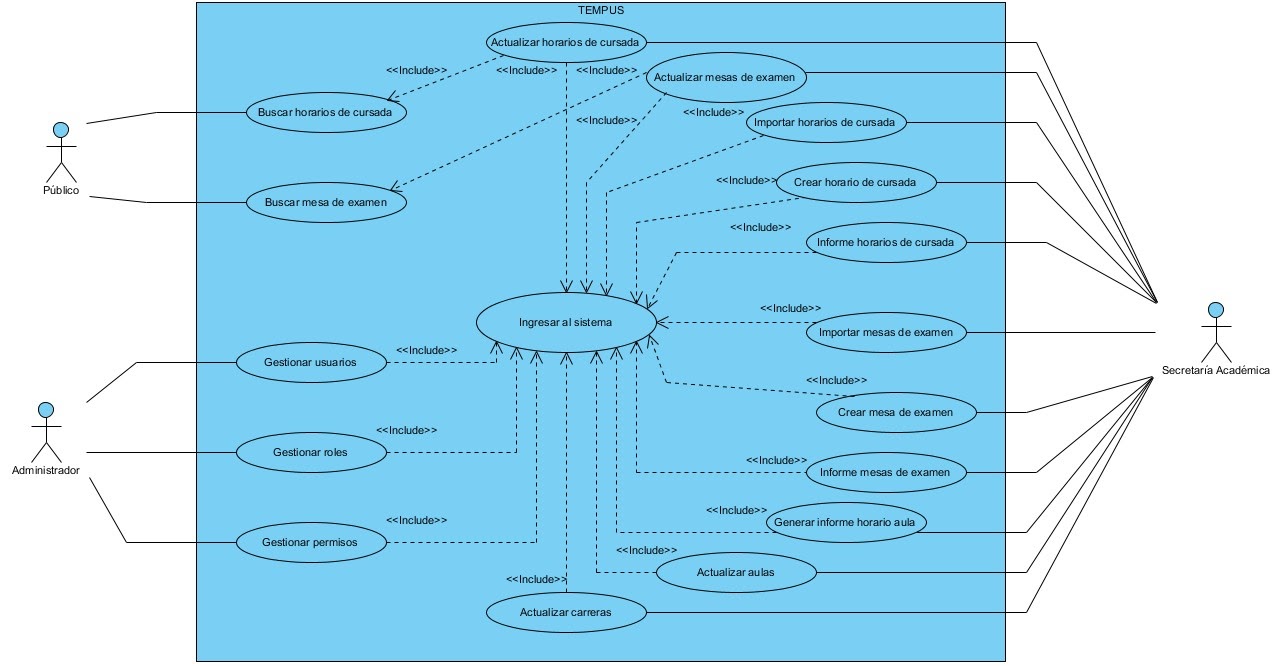


Ilustración 10 - Diagrama de casos de uso

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 los gráficos que nos brinda el repositorio (GitHub), obteniendo como resultado lo siguiente:

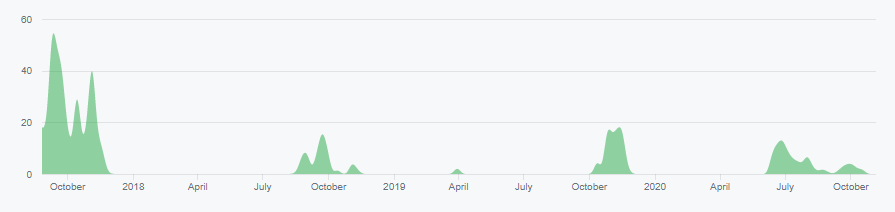


Ilustración 11 - Trabajo realizado por Yenú (GitHub)

Como se mencionó, el gráfico anterior denominado “Trabajo realizado por Yenú” se obtuvo de GitHub y detalla el tiempo de trabajo real del equipo de desarrollo a través de los commits en el repositorio.

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 riesgos de la presente memoria.

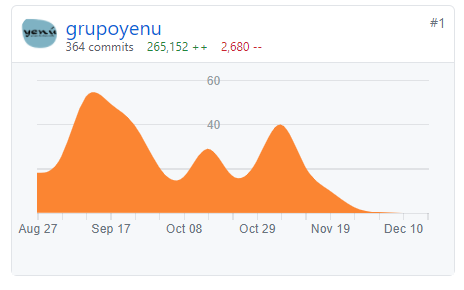


Ilustración 12 - Trabajo realizado por Yenú primera etapa

El  gráfico de la primera etapa, es un acercamiento del grafico general presentado anteriormente. El mismo corresponde al primer periodo y abarca del 27 de Agosto de 2017 al 6 de diciembre del mismo año. Se nota que coincide con el cuatrimestre en el que se cursó la materia Laboratorio de desarrollo del Software.

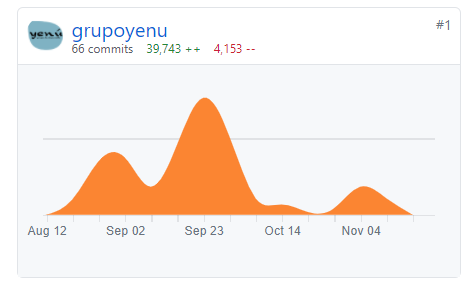


Ilustración 13 - Trabajo realizado por Yenú segunda etapa

El  Gráfico de la segunda etapa abarca el periodo del 12 de Agosto de 2018 al 22 de noviembre del mismo año. Se puede notar una clara disminución del trabajo comparando las 364 confirmaciones de la primera etapa contra los 66 de la segunda.

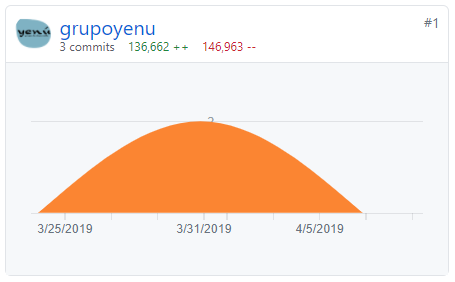


Ilustración 14 - Trabajo realizado por Yenú tercera etapa

El gráfico de la tercera etapa inicia el 24 de Marzo del 2019 y finaliza el 7 de Abril del 2019. En este lapso de tiempo no se aplicaron grandes cantidades de confirmación. Es un número mínimo comparado con las anteriores.

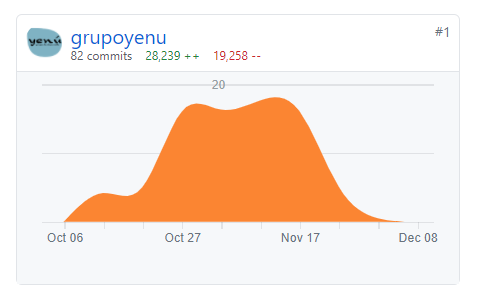


Ilustración 15 - Trabajo realizado por Yenú cuarta etapa

La cuarta etapa corresponde al periodo desde el 6 de Octubre de 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.

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 consideraban cada uno de los integrantes, tanto en lo personal como en las herramientas y la metodología. Para documentar la gestión de los riesgos utilizamos la plantilla Excel del PSI, donde se redactan y contestan 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.

Detectar estos riesgos de manera temprana permite reducir tiempo y costo. Lo que significa que aunque alguno ocurra, el impacto sea mínimo y fácil de subsanar. Sin embargo cabe aclarar  que si bien hay riesgos que se pueden predecir después de una evaluación del plan del proyecto, del entorno técnico y otras fuentes de información fiables, también existen riesgos impredecibles que pueden ocurrir, pero es extremadamente difícil identificarlos por adelantado. Está claro que la experiencia es uno de los aspectos más importantes en la detección de riesgos, algo con lo que los integrantes del grupo no contaban y que derivó en algunos retrasos en la planificación del proyecto.

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. A continuación detallamos  las categorías definidas:

* Cronograma: Esta categoría fue elegida porque considera el tiempo necesario para la realización de las actividades. Es importante ya que abarca todos los aspectos que podrían retrasar el proyecto.
* Experiencia y Capacidad: Esta categoría considera la experiencia o capacidad que tiene cada uno de los integrantes para poder dar avance al proyecto. Es decir que la falta de experiencia  y capacidad sobre las herramientas, tecnología, metodología o lenguaje de programación, trae un retraso al desarrollo.
* 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: Está presente porque toma en cuenta todos aquellos cambios en la tecnología (hardware o software) que puedan impactar en el proyecto.
* Legal y Contractual: Está estrato abarca todos aquellos aspectos de carácter legal o contractual que puedan impactar en el proyecto como por ejemplos restricciones de licencias o uso de datos.

### Proceso

El proceso de Gestión de riesgos debería efectuarse al inicio de cada iteración, siendo ésta una de las primeras tareas a planificar.

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

1. Identificación de los riesgos. Listado de riesgos potenciales agrupados según la categorización.
2. Análisis de riesgos: listados de priorización de riesgos
3. Planificación de riesgos: anulación de riesgos y planes de contingencia
4. Control de riesgos.

### 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 porque en la Iteración N° 1 de la Fase de Inicio se planificó la redacción de un Plan de Riesgos. Se pensó realizar la gestión de riesgo en iteraciones posteriores.

Durante la Iteración N° 2 se realizó la primera Gestión de Riesgos. Inicialmente y por pedido del equipo de cátedra, cada uno de los integrantes hizo una planilla Anexo 1 con los riesgos que consideraba pertinentes para luego unificarlos. Los riesgos que surgieron para un análisis corresponden a las categorías “Cronograma” y “Experiencia y Capacidad”. Se debe a que no teníamos experiencia en desarrollo similares, en la metodología y en el lenguaje de programación, lo que implicaba invertir tiempo en adquirir conocimientos. 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.



Ilustración 16 - Resumen de riesgos Iteración N° 2 Fase de Inicio

Los riesgos detectados son:

**RK01 - ¿Todos los integrantes del grupo de desarrollo cuentan con el tiempo necesario para la realización de las actividades?**

En la categoría “Cronograma” con un factor de 240. Se evaluó el tiempo necesario que dispone cada uno de los integrantes del grupo de desarrollo para la realización de las actividades. El presente riesgo existe porque no todos los integrantes cuentan con el 100% del tiempo para dedicarle al proyecto por: Responsabilidades laborales y tiempo para estudios. La solución al riesgo consiste en coordinar los tiempos que tienen cada uno de los integrantes.

**RK02 - ¿La solución propuesta ha sido implementada con anterioridad?**

En la categoría de “Experiencia y capacidad” con un factor de 320. Se evaluó  que la propuesta no ha sido implementada con anterioridad. Por ello,  el presente riesgo existe dado que Yenú no cuenta con implementaciones de desarrollo anteriores. Como solución se propone  realizar investigaciones sobre lo desconocido, llámese herramientas, tecnología  y  entornos de trabajo.

**RK03 - ¿El equipo de desarrollo tiene experiencia en el desarrollo de soluciones similares?**

En la categoría de “Experiencia y capacidad” con factor 320. Se evalúa la falta de experiencia que tiene el grupo de desarrollo en soluciones similares a la propuesta. El riesgo existe porque ninguno de los integrantes tenía experiencia en el desarrollo de Sitios Web o Aplicaciones móviles, cualquiera sea el dominio del problema. Algunos integrantes del grupo han desarrollado pero no sistemas de este tipo (Por ejemplo: aplicaciones de escritorio en Java). Como solución se realizaron investigaciones, prácticas y consultas a personas que  tenían experiencia en dominios similares.

**RK04 - ¿El equipo de desarrollo tiene experiencia en la metodología?**

Este riesgo pertenece a la categoría de “Experiencia y capacidad” y cuenta con un factor de  240. Se evalúa la experiencia que tiene el grupo de desarrollo con la metodología a utilizar. El presente riesgo existe ya que ninguno de los integrantes posee experiencia en la utilización práctica de la metodología Iterativa e Incremental y nunca utilizaron PSI. Esto se ve reflejado en;  fallas de planificación,  incorrecta identificación de actividades, problemas de estimación, etc. Se buscará reducir el riesgo haciendo uso de PSI y de bibliografía.

**RK05 - ¿El equipo de desarrollo tiene experiencia en el lenguaje de programación?**

El RK05 corresponde a “Experiencia y capacidad” y se evaluó con un factor de 300. Se analiza la falta de experiencia que tienen los programadores en el lenguaje de programación PHP. El presente riesgo existe porque a lo largo de la carrera se trabaja con Java y los integrantes de Yenu nunca utilizaron PHP. Esto ocasiona dificultades en la resolución de problemas y desconocimiento de las características del lenguaje. Como respuesta se debe invertir tiempo en la adquisición de conocimientos del lenguaje de programación. Algunas alternativas son: lectura de libros, hacer ejemplos sencillos o realizar tutoriales.

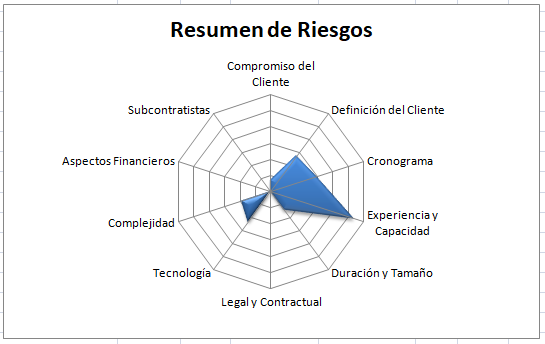
Fase Elaboración

En esta fase no se realizó gestión de riesgos dado que el grupo priorizo otras actividades. Como se mencionó anteriormente, el tiempo con el que se contaba no era suficiente para realizar todas las tareas planificadas y por ello, se debía elegir las de mayor prioridad. Sin embargo, esto no implicó que las soluciones  propuestas de  los riesgos no se llevarán a cabo.

Durante el 2017 todos los participantes del equipo nos encontrábamos; cursando otras materias, debíamos entregar cada semana  avances del proyecto, logrando alcanzar de esta manera los objetivos más importantes, por lo que los tiempos debían distribuirse según las necesidades del momento.

Fase Construcción

En esta fase de Construcción surgieron nuevos riesgos, de los cuales tuvimos que utilizar diferentes  tipos de estrategias para poder eliminarlos, mitigarlos o realizar contingencia. Se debe recordar que durante esta fase sucedieron 8 Iteraciones desde 2017 a la actualidad de manera interrumpida.



Los riesgos nuevos en esta fase son:

**RK06 - ¿El equipo de desarrollo tiene conocimiento en HTML?**

Este riesgo se encuentra dentro de la categoría “Experiencia y capacidad”. Se evalúa la falta de experiencia que tienen los programadores en HTML. El presente riesgo existe porque se tiene poco conocimiento en HTML lo cual es necesario para la creación del sitio web y se debe utilizar también en la aplicación móvil. Como solución se debe reforzar el poco conocimiento adquirido por el grupo de desarrollo con alternativas como: retomar trabajos de Laboratorio de Programación donde se hacían ejemplos sencillos con HTML y JSP, visualizar tutoriales e investigar bibliografía.

**RK07 - ¿El equipo de desarrollo tiene conocimiento en CSS?**

El presente riesgo, encuadra dentro de la categoría “Experiencia y capacidad”. El riesgo surge de la evaluación sobre el poco conocimiento en las hojas de estilo en cascada (CSS). CSS  es la mejor forma de separar los contenidos de su presentación y es imprescindible para crear páginas web complejas.

El principal problema fue la adaptación con UARGFlow. Como solución  se buscó reforzar el poco conocimiento adquirido, durante la cursada de la carrera. Utilizando como fuente de investigación libros y tutoriales. Se debe destacar además  que CSS está estrechamente relacionado con los conocimientos en HTML.

**RK08 - ¿El equipo de desarrollo tiene experiencia en el uso de JavaScript?**

El presente riesgo se encuentra clasificado en “Experiencia y capacidad”.  JavaScript es un lenguaje de programación interpretado y se define como orientado a objetos (OO), basado en prototipos, imperativo, débilmente tipado y dinámico. Se evalúa la poca experiencia que tienen los programadores en el lenguaje. Como se utiliza principalmente del lado del cliente, implementado como parte de un navegador web permitiendo mejoras en la interfaz de usuario y páginas web dinámicas, se deben reforzar todos sus conceptos: utilizando videos tutoriales, ejemplos, foros y libros.

**RK09 - ¿El equipo de desarrollo tiene experiencia utilizando Ionic?**

El presente riesgo se encuadra en la clasificación “Experiencia y capacidad”.  Uno de los requerimientos principales del proyecto es el desarrollo de una aplicación móvil haciendo uso de Ionic. Siendo  Ionic un SDK completo de código abierto para el desarrollo de aplicaciones móviles híbridas multiplataforma que utilizan HTML5, CSS y Cordova.

 El presente riesgo existe porque no se tiene  conocimiento en la utilización de Ionic. Dando como solución la investigación en componentes de empaquetados, las plantillas, el cli, la estructura de archivos y bibliotecas.

**RK10 - ¿Existen factores que pueden alterar el ritmo de trabajo actual?**

El presente riesgo se encuadra en la clasificación “Duración y Tamaño”. Dada la generalidad del riesgo, se pueden enumerar múltiples aspectos que pueden alterar el ritmo de trabajo a  lo largo del proyecto.  Las situaciones que hicieron que se trabaje en forma interrumpida fueron:

* Vacaciones.
* Mesas de exámenes.
* Cursada de materias en diferentes periodos.
* Maternidad.
* Aspectos laborales.
* Pandemia

Cabe destacar que este año surgió un nuevo factor que no se tenía contemplado que fue la pandemia COVID-19. La cual  se produce cuando una enfermedad contagiosa se propaga rápidamente en la población, afectando simultáneamente a un gran número de personas durante un periodo de tiempo concreto.

Este factor nos modificó la manera de poder trabajar en equipo, dado que  estaba prohibido por decreto realizar reuniones presenciales. Como solución buscamos una alternativa para comunicarnos  utilizando el servicio de videoconferencia ZOOM y Google Meet. Esto nos permite realizar reuniones virtuales de acuerdo al tiempo de cada uno.

**RK11 - Dado el avance tecnológico ¿se puede cambiar la tecnología de desarrollo durante el proyecto y afectar lo antes desarrollado?**

El presente riego se encuadra en la categoría “Tecnología”. Se evalúa que dados los cambios y los avances tecnológicos se deban actualizar/modificar funcionalidades del sistema. Se consideran tener que realizar adaptaciones sobre elementos ya existentes en el proyecto. El presente riesgo existe porque se han producido actualizaciones en las herramientas que se utilizaron para el desarrollo de la aplicación móvil de Tempus (de versión 2 a la 3 durante 2017). Esto no había sido considerado en fases previas, por lo cual se analiza esta posibilidad en esta fase de construcción.

Como solución a dicho riesgo, se busca seguir reforzando el conocimiento sobre  la nueva versión de Ionic. Se deben estudiar los cambios para adaptar la versión anterior de nuestra aplicación a la versión actual de Ionic.

Durante  las fases que fuimos realizando en este proyecto, podemos decir que no siempre encontramos todos los riesgos que pudieran surgir, lo que si podemos ir utilizando diferentes tipos de estrategia que nos 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

Los actores que interactúan con este caso de uso son: Administrador y Secretaria Académica.

Para acceder a todas las funcionalidades del sistema primero es necesario autenticarse mediante el correo electrónico institucional. Para un usuario que este habilitado, se le permite el acceso al sistema y se muestra el menú con las opciones que dispone su perfil.

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

Los actores del caso de uso son: Administrador y Secretaría Académica.

Este caso de uso es iniciado por el actor que tiene la opción de importar un archivo con los horarios de cursada. La planilla a cargar debe respetar el formato requerido por el sistema por lo que una vez importada se deben observar aquellas filas que no cumplen alguna validación. El usuario del sistema puede optar por guardar los datos que sean correctos o modificar el documento y repetir el proceso en caso de ser necesario.

Modelar sistemas implica manejar una cantidad considerable de elementos de modelado por lo que es necesario organizar estos elementos en bloques mayores. La mejor forma de comprender un sistema es agrupando las abstracciones en grupos denominados paquetes (UML). Los paquetes son mecanismos de propósito general para organizar elementos en grupos. Se presenta el diagrama de Paquetes para el actual caso de uso:



Ilustración 17 - Diagrama de paquetes para el Caso de Uso 02

En esta sección se utilizan los objetos detectados para indicar su interacción. El detalle de cada uno de los pasos que se realiza en el siguiente diagrama de secuencia se encuentra en el documento Especificación CU02 - Importar horarios de cursada. Lo que se visualiza en el diagrama corresponde al flujo principal del caso de uso.



Ilustración 18 - Diagrama de secuencia para el Caso de Uso 02



Ilustración 19 - Diagrama de flujo de eventos para el Caso de Uso 02

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

Dos son los actores que participan en este caso de uso: Administrador y Secretaría Académica.

Este caso de uso es iniciado por el actor que tiene la opción de importar un archivo con las mesas de examen. La planilla a cargar debe respetar el formato requerido por el sistema por lo que una vez importada se deben observar aquellas filas que no cumplen alguna validación. El usuario del sistema puede optar por guardar los datos que sean correctos o modificar el documento y repetir el proceso en caso de ser necesario.

Al igual que en el caso anterior, se muestra el diagrama de paquetes para el caso de uso:



Imagen - Diagrama de paquetes para el Caso de Uso 03

El detalle de cada uno de los pasos que se realiza en el siguiente diagrama de secuencia se encuentra en el documento Especificación CU03 - Importar mesas de examen. Lo que se visualiza en el diagrama corresponde al flujo principal del caso de uso.



Imagen – Diagrama de secuencia para el Caso de Uso 03



Imagen - Diagrama de flujo de eventos para el Caso de Uso 03

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

Buscar horario de cursada cuenta con los actores: Administrador, Secretaría Académica y Público.

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

Los actores participes aquí son: Administrador, Secretaría Académica y Público.

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

En este caso los actores involucrados son: Administrador y Secretaria Académica.

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

Los actores intervinientes en el caso de uso 07 son: Administrador y Secretaria Académica.

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:

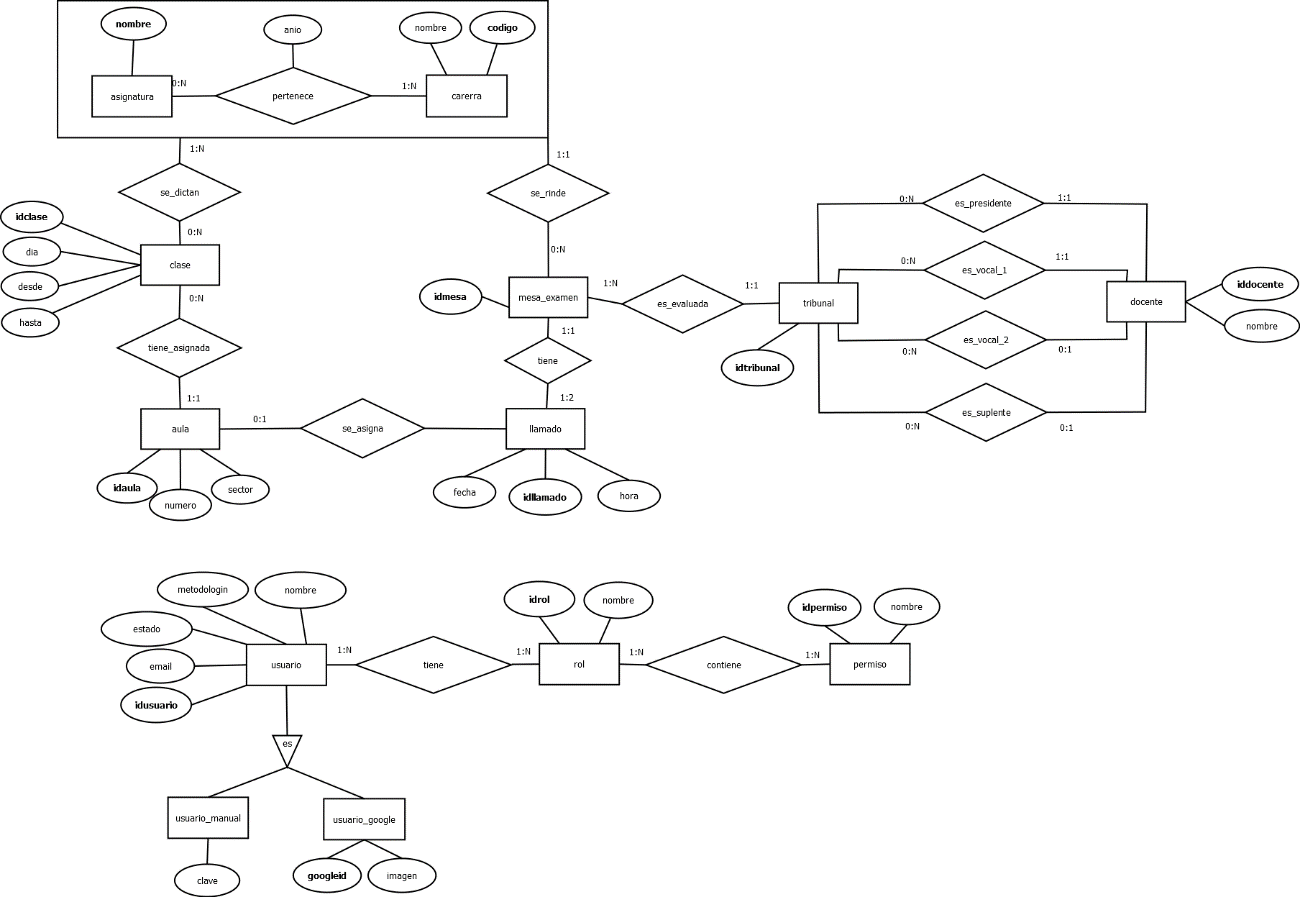


Imagen – Diagrama Entidad-Relación (Notación de Peter Chen)

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:

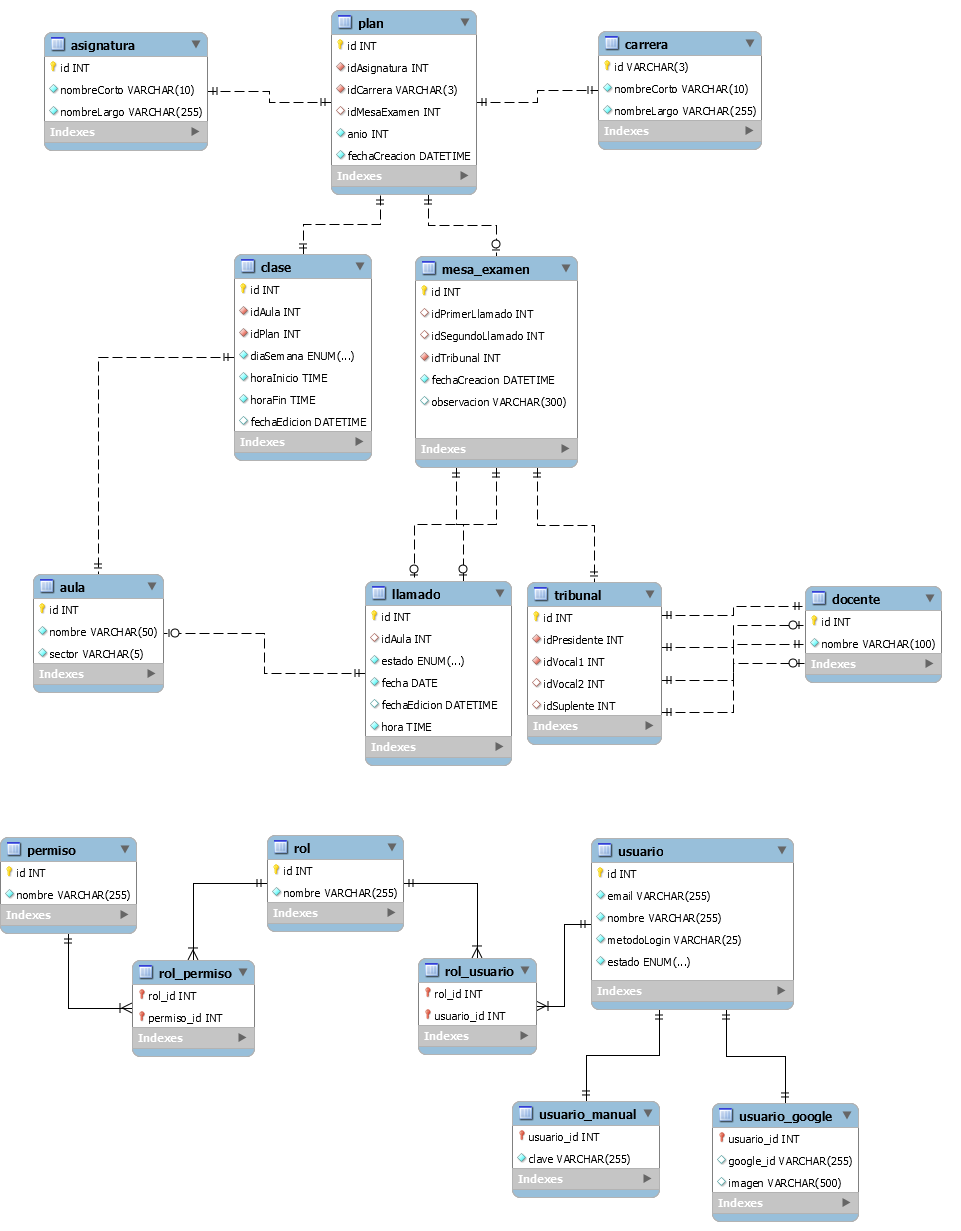


Imagen – Modelo relaciones con MySQL Workbench

## 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.

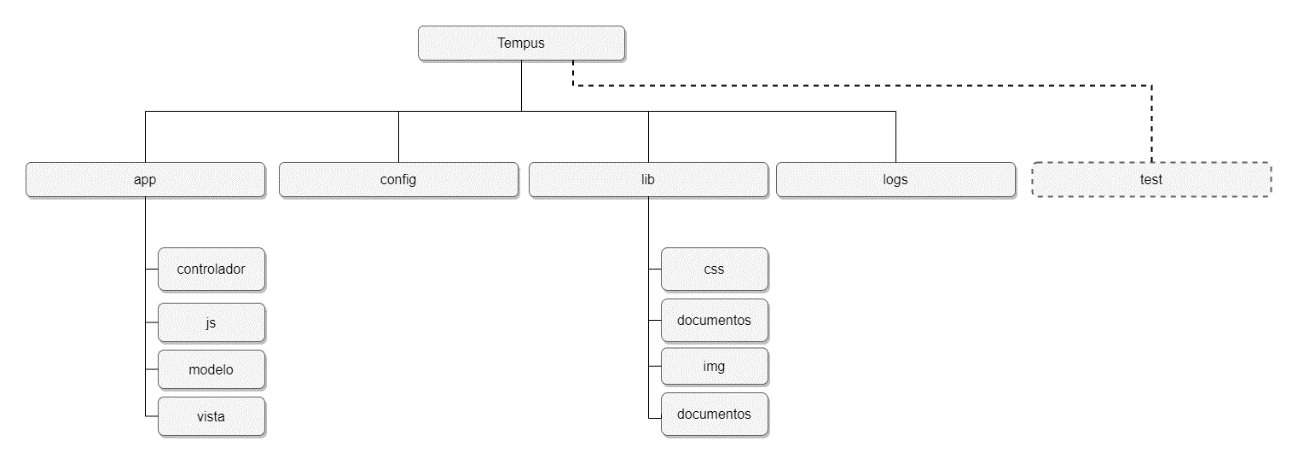


Imagen – Estructura de directorios de la Web

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

En esta sección haremos una descripción de las tecnologías utilizadas, tanto de hardware como de software, durante el desarrollo de proyecto.

## Hardware

### Oyarzo Mariela

Tabla

### Quiroga Sandra

Tabla

### Marquez Emanuel

|  |  |
| --- | --- |
| *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.

### Google Meet

Describir

### 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

Durante prácticamente todo el proceso de desarrollo se efectuaron reuniones, ya sean con periodicidad semanal o diaria, virtual o presencial. Esto se debe a que como grupo entendimos que el proyecto iba a ser más completo y enriquecedor bajo un concepto colaborativo. Dichos encuentros los utilizamos para ver qué cosas funcionaban bien, que aspectos se podían mejorar y definir las tareas a seguir.

Mencionado esto, utilizamos el Anexo para hacer un resumen de las reuniones más importantes que sucedieron a lo largo del proyecto donde la mayoría fueron detalladas en papel.

Reunión N° 1

|  |  |
| --- | --- |
| Fecha | 22/08/2017 |
| Lugar | Campus UARG |
| Temas | Nombre y Logo del grupo.  Nombre y logo del sistema. |
| Resolución | Los integrantes del grupo buscamos alternativas, para darle un nombre tanto al grupo como al sistema, que sea significativo con el desarrollo del mismo.  Nombre del grupo de desarrollo: YENU: Amigo en tehuelche. 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 en el desarrollo del proyecto.  LOGO. La forma no tiene ningún significado en especial. Se ha seleccionado un color de la gama de los azules (#7fb2c3) dado que se asocia con la estabilidad, lealtad y confianza.  Nombre del proyecto: TEMPUS. En latín significa Tiempo. 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. |

Reunión N° 2

|  |  |
| --- | --- |
| Fecha | 26/08/2017 |
| Lugar | Domicilio de Quiroga Sandra |
| Temas | Prototipos desechables de aplicaciones móviles.  Metodología del PSI.  Estilo de la documentación y paleta de colores |
| Resolución | Realizamos prototipos desechables, que permiten al usuario hacerse una idea sobre las interfaces que proveerá el sistema. El uso de prototipos se realiza a través de una serie de imágenes a ser presentadas.  Comenzamos a estudiar la metodología que utilizaríamos para poder realizar la documentación del proyecto.  Buscamos colores de la gama de los azules dado que se asocia con la estabilidad, lealtad y confianza. Tanto en la documentación como en al presentaciones realizadas durante la cursada. |

Reunión N° 3

|  |  |
| --- | --- |
| Fecha | 30/08/2017 |
| Lugar | Vía Whatsapp |
| Temas | Prototipo desechable de la página web |
| Resolución | Planteamos ejemplos de prototipos de la página web. Luego un integrante del grupo se encargaba de realizarlo con el software Pencil. Esto permite al usuario hacerse una idea sobre las interfaces que proveerá el sistema. |

Reunión N° 4

|  |  |
| --- | --- |
| Fecha | 01/09/2017 |
| Lugar | Vía Whastapp |
| Temas | Modelo de negocio.  Glosario de términos. |
| Resolución | Redactar el Modelo de Negocio. Para poder Identificar, describir y analizar la oportunidad de negocio. Analizar su viabilidad técnica, económica y financiera.  Redactar glosario, definimos los principales términos utilizados en el proyecto. Permitiendo establecer la terminología a usar. |

Reunión N° 5

|  |  |
| --- | --- |
| Fecha | 04/09/2017 |
| Lugar | Vía Whatsapp |
| Temas | Plan del proyecto.  Modelo de negocio.  Modelo de casos de uso. |
| Resolución | Redactar el plan del proyecto  Finalizar con el modelo de negocio. Luego de realizar las entrevistas con los docentes logramos evacuar las dudas que teníamos sobre el proyecto en sí.  Iniciar el modelo de caso de uso con priorización, es decir, para cada fase e iteración vamos realizando los casos de uso dependiendo de su complejidad. |

Reunión N° 6

|  |  |
| --- | --- |
| Fecha | 06/09/2017 |
| Lugar | Via Whatsapp |
| Temas | Requerimientos de usuario.  Actores.  Casos de uso.  Diagramas de casos de uso. |
| Resolución | Posibles requerimientos de usuario   * Administrar usuarios. * Administrar roles. * Administrar permisos. * Administrar horarios de cursada. * Administrar mesas de examen. * Administrar aula. * Administrar carrera.   El personal de secretaria académica, debe poder ingresar al sistema, mediante un correo institucional. Ej: [nombre@unpauarg.edu.ar](mailto:nombre@unpauarg.edu.ar)  Posibles actores:   * Administrador: Encargado de administrar usuarios, roles y permisos. Tiene acceso al sistema completo * Secretaria Académica: Encargado de administrar las cursadas y mesas de examen. Además puede acceder a los informes. * Público: Puede consultar horarios de cursada y mesas de examen.   Posibles casos de uso:   * Ingresar al sistema. * Cargar horarios de cursada. * Cargar mesas de examen. * Buscar horarios de cursada. * Buscar mesas de examen. * Modificar horarios de cursada. * Modificar mesas de examen. * Crear horarios de cursada. * Crear mesas de examen. * Informes mesas de examen. * Informes horarios de cursada * Buscar aula * Borrar aula * Modificar aula * Generar horarios de aula. * Buscar carrera. * Borrar carrera. * Modifica carrera. |

Reunión N° 7

|  |  |
| --- | --- |
| Fecha | 11/09/2017 |
| Lugar | Vía Whatsapp |
| Temas | Evaluación final Fase de Inicio Iteración 1  Al finalizar la presente iteración se deberán haber alcanzado los siguientes objetivos establecidos. |
| Resolución | Se detallan los objetivos que se han alcanzado:   * Redactar los planes del proyecto. * Finalizar el Modelo de Negocio. * Iniciar el Modelo de Casos de Uso junto con la priorización de los mismos.   Se detallan los objetivos que no se han alcanzado:   * Redactar el Modelo de Visión. * Redactar Especificación de Requerimientos de Software. * Iniciar la Arquitectura del Sistema. * Gestionar riesgos. * Estimación del proyecto.   Todos aquellos objetivos que no se han concluido, serán planificados para la próxima iteración. |

Reunión N° 8

|  |  |
| --- | --- |
| Fecha | 14/10/2017 |
| Lugar | Domicilio de Quiroga Sandra. |
| Temas | Plan de riesgos.  Estimación de la iteración.  Prototipo.  UARGFlow e Ionic Funcional. |
| Resolución | Mostramos la Gestión de riesgos- Fase Construcción Iteración 1.  Resumen de Riesgo con el grafico que nos permite ver cuáles son los más importantes a tratar.  Estimaciones del proyecto: 1 año y 3 meses.  Estimación 1, considerando los casos de usos realizados hasta el momento: 7 meses.  Estimación 2, idem a lo anterior: 3 meses y 2 semanas.  Prototipo de la App  Prototipos de la Web, teniendo en cuenta la estructura de UARGFLOW |

Reunión N° 9

|  |  |
| --- | --- |
| Fecha | 28/10/2017 |
| Lugar | Domicilio de Quiroga Sandra |
| Temas | Especificación de casos de uso.  Modelo de datos.  Modelos de diseño.  Implementación.  Plan de pruebas.  Casos de prueba. |
| Resolución | Mostraremos las especificaciones de los casos de usos más complejos. Por ejemplo uno de ellos es : Importar mesas de examen  El Modelo de datos que realizamos nos permite describir la estructura de la base de datos que vamos a utilizar en el proyecto. La misma muestra los tipos de datos y la forma que se van a relacionar.  En el Modelo de diseño hicimos visible los diagramas, el de clases con sus respectivos objetos con sus operaciones y atributos.  En la Implementación mostramos la estructura del proyecto.  Plan de Pruebas:   * Unitarias: Comprobar el correcto funcionamiento de una unidad de código. * Integración: Comprobar el correcto funcionamiento de varios módulos en forma conjunta, con el fin de verificar las interfaces entre ellos. * Funcionamiento: Probar es que el programa realice las funciones especificadas por el cliente en el documento de especificación de requerimiento. * Sistemas: Identificar diferencias entre el sistema y sus requerimientos originales(requerimientos funcionales y no funcionales) * Aceptación: Verificar que el producto esté listo para su implementación y utilización, bajo criterios definidos por los usuarios. * Usabilidad: Confirmar que se cumplan las expectativas de los usuarios. Verificar que se pueda operar adecuadamente.   Casos de prueba: realizamos varias pruebas, en especial con los casos de usos más complejos. Por ejemplo Importar mesa de examen y cursada, al igual que buscar mesa y cursada. |

Reunión N° 10

|  |  |
| --- | --- |
| Fecha | 13/11/2017 |
| Lugar | Vía Whatsapp |
| Temas | Cierre de la Iteración |
| Resolución | Se detallan los objetivos que se han alcanzado:   * Finalizar implementación CU02. * Finalizar implementación CU03. * Finalizar implementación CU04. * Finalizar implementación CU05. * Ejecutar casos de prueba CU02. * Ejecutar casos de prueba CU03. * Ejecutar casos de prueba CU04. * Implementar CU04 – Buscar horario de cursada (APP). * Implementar CU08 – Crear horario de cursada. * Implementar CU07 - Modificar mesa de examen. * CU04 – Buscar horario de cursada (APP). * CU05 – Buscar mesa de examen (APP).   Se detallan los objetivos que no se han alcanzado:   * Implementar CU06 – Modificar horario de cursada. * Implementar CU08 - Crear mesa de examen. * CU06 – Modificar horario de cursada. * CU07 – Modificar mesa de examen. * CU08 – Crear horario de cursada. * CU09 – Crear mesa de examen.   Todas aquellas actividades que no fueron terminadas, quedaran en la siguiente iteración. |

Reunión N° 11

|  |  |
| --- | --- |
| Fecha | 23/11/2017 |
| Lugar | Vía Whatsapp |
| Temas | Evaluación de iteración  Estimación  Planificación  Balance  Implementación: Pagina web y App |
| Resolución | Evaluación de la iteración: Tareas que se realizaran según la prioridad (alta, media y baja) junto con el lapso de tiempo que se estima en cada iteración.  Mostramos los objetivos que fueron alcanzados durante la iteración, al igual que aquellos que no fueron alcanzados.  En la estimación hicimos una comparación de la estimación inicial del proyecto siendo este 1año y 3 meses.   * 2 casos de usos complejos * 2 casos de uso promedio * 8 casos de uso simples * Integración de UARGFlow   El tiempo utilizado durante la cursada que fue 3 meses y 1 semana. Entonces la estimación de trabajo restante es de 4 meses y 6 días.  Planificación: En la planificación mostramos la documentación restante la cual serian:   * Manual de instalación * Manual de usuario * Memoria * Actualizaciones sobre documentos actuales.   Implementación restante de los CU10 al CU18.  Balance: Realizamos un balance final sobre aquellas herramientas utilizadas durante la cursada y que utilizaremos a futuro como ser PSI, Taiga, Repositorio Github y WhatsApp. |

Reunión N° 12

|  |  |
| --- | --- |
| Fecha | 17/08/2018 |
| Lugar | Domicilio de Quiroga Sandra |
| Temas | Presentación sobre la continuidad del proyecto.  Descripción general del proyecto.  Estado actual del proyecto.  Planificación de la nueva iteración.  Propuesta sobre notificaciones y favoritos en APP.  Presentación de un video de ejemplo. |
| Resolución | El grupo de desarrollo tomó la decisión de realizar una reunión con el objetivo de reanudar el proyecto Tempus. Durante la misma, se decidió:   * Repasar la documentación actual disponible en el repositorio (Github). * Determinar el estado de la documentación. * Planificar nueva iteración. * Realizar presentación al equipo de cátedra. * Comprometerse con la finalización del trabajo.   Planificación de la nueva iteración: Fase de Construcción Iteración 4 desde 01/09/2018 al 16/09/2018.  Notificaciones y favoritos: Sobre las notificaciones y favoritos en la APP se acordó junto con el equipo de cátedra (durante 2017):   * Dar la opción de tener favoritos para evitar nuevas búsquedas. Esto es, tanto para horarios de cursada como para mesas de examen. * No mostrar notificaciones en la aplicación (al estilo whatsApp) debido a la complejidad. |

Reunión N° 13

|  |  |
| --- | --- |
| Fecha | 16/09/2018 |
| Lugar | Domicilio de Quiroga Sandra |
| Temas | Cierre de Iteración 4 – Fase de construcción. |
| Resolución | A continuación se detallan los objetivos que se han alcanzado:   * Subir la planificación iteración actual a Taiga. * Modificar Especificación de CU06 – Modificar horario de cursada. * Modificar Especificación de CU07 – Modificar mesa de examen. * Modificar Modelo de Datos. * Modificar Base de Datos (tempus.sql disponible).   Los objetivos que no se han alcanzado se debieron a ocupaciones personales de cada uno de los integrantes del grupo.  Las tareas que no se han completado deben realizarse en la siguiente iteración. Las actividades que se han iniciado también deberán finalizarse durante la iteración 5 de la fase de construcción. |

Reunión N° 14

|  |  |
| --- | --- |
| Fecha | 30/09/2018 |
| Lugar | Domicilio de Quiroga Sandra |
| Temas | Cierre de Iteración 5 – Fase de construcción. |
| Resolución | A continuación se detallan los objetivos que se han alcanzado:   * Implementación CU06 – Modificar horario de cursada. * Implementación CU07 – Modificar mesa de examen. * Casos de prueba CU02 – Importar horarios de cursada. * Casos de prueba CU03 – Importar mesas de examen.   A continuación se detallan los objetivos que no se han alcanzado:   * Implementación CU04 – Buscar horario de cursada (APP). * Implementación CU11 – Notificar (APP).   La iteración se da por cerrada, lo que no se realizó se deberá terminar en la siguiente iteración. |

Reunión N° 15

|  |  |
| --- | --- |
| Fecha | 15/10/2018 |
| Lugar | Domicilio de Quiroga Sandra |
| Temas | Cierre de Iteración 6 – Fase de construcción. |
| Resolución | Los objetivos alcanzados en esta iteración fueron:   * Implementar CU04 – Buscar horario de cursada (APP). * Implementar CU08 – Crear horario de cursada. * Implementar CU09 – Crear mesa de examen. * Iniciar implementación CU11-Informe mesa de examen.   Los objetivos no alcanzados fueron:   * Continuar glosario. * Descargar archivo de Mesas de examen Octubre 2018. * Actualizar código para realizar pruebas. * Realizar pruebas CU05 – Modificar horario de cursada. * Realizar pruebas CU06 – Modificar mesa de examen. * Continuar manual de usuario (APP). * Continuar Modelo de Casos de Uso. * Continuar Arquitectura del Sistema. * Continuar Plan de pruebas. * Implementar CU10 – Notificar. * Crear archivo Tempus mesas de examen octubre 2018. * Continuar Modelo de Diseño. |

Reunión N° 16

|  |  |
| --- | --- |
| Fecha | 26/10/2019 |
| Lugar | Domicilio de Quiroga Sandra |
| Temas | Cierre de Iteración 7 – Fase de construcción. |
| Resolución | Estudiar cambios en la versión de Ionic  Crear archivo Tempus mesas de examen octubre 2019.  Crear archivo Tempus Horarios de cursada octubre 2019.  Se realizó la evaluación general del estado del proyecto.  Se realizó la evaluación de los riesgos para la iteración actual. |

Reunión N° 17

|  |  |
| --- | --- |
| Fecha | 17/11/2019 |
| Lugar | Domicilio de Oyarzo Mariela |
| Temas | Descripción general del proyecto  Estado actual del proyecto  Videos de la página web |
| Resolución | Última presentación 31/08/2018.  La última estimación 1 mes y 3 semanas  Reanudación del proyecto 11/10/2019 hasta el 22/11/2019  Total de tiempo trabajado 4 meses y 3 semanas |

Reunión N° 18

|  |  |
| --- | --- |
| Fecha | 15/06/2020 |
| Lugar | Vía Zoom |
| Temas | Recordar todo lo hecho hasta el momento.  Pandemia. |
| Resolución | Hablamos en general sobre lo que teníamos ya realizado de la página y web y la app.  Tuvimos una reunión al principio de la cursada, con los encargados de la secretaria académica.  Actualizar los documentos con los nuevos celulares (hardware) que tenemos ahora.  Un integrante en el lapso de estos años comenzó a trabajar, lo cual es menos cantidad de horas para el proyecto.  Embarazo de una compañera.  No tuvimos en cuenta la actualización de Ionic, si bien ya lo habíamos hablado en otras reuniones. Era importante hacer mención nuevamente para saber dónde nos encontrábamos.  Leer los documentos nuevamente  Pandemia fue un nuevo riesgo que no tuvimos en cuenta, ya que nos limitaba juntarnos de manera presencial. Entonces, nuestra manera de mitigarlo fue a través de la video conferencia (Zoom) |

Reunión N° 19

|  |  |
| --- | --- |
| Fecha | 23/06/2020 |
| Lugar | Vía Zoom |
| Temas | Gestión de riesgos  Página web |
| Resolución | Riesgos que teníamos hasta el momento y eran necesarios hacerle un seguimiento  Mostrar de manera general la página web entre los integrantes, para recordar bien su funcionalidad, ante cualquier duda existente.  Para la próxima reunión cada uno tendría lista sus características de nuevos celulares.  Comenzar con la memoria, buscar ejemplos para comenzar con la misma. |

Reunión N° 20

|  |  |
| --- | --- |
| Fecha | 30/06/2020 |
| Lugar | Vía Zoom |
| Temas | Redacción de algunas secciones de la Memoria. |
| Resolución | Iteraciones (Taiga).  Objetivo principal.  Objetivos específicos. |