

Memoria

TEMPUS



**Autores:**

Quiroga, Sandra

Márquez, Emanuel

Oyarzo, Mariela

**Tutor:**

Mg. Albert Osiris Sofía

Laboratorio de Desarrollo de Software

Analista de Sistemas- Licenciatura en Sistemas

Unidad Académica Río Gallegos

Universidad Nacional de la Patagonia Austral



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



Tabla de contenido

[Introducción 4](#_Toc257017457)

[Alcance 4](#_Toc257017458)

[Documentos relacionados 4](#_Toc257017459)

[Definición, Acrónimos y Abreviaturas 4](#_Toc257017460)

[Descripción 4](#_Toc257017461)

[Subsistemas 4](#_Toc257017462)

[Estructura 4](#_Toc257017463)

[Construcción 4](#_Toc257017464)

[Prueba 4](#_Toc257017465)

[Evaluación 4](#_Toc257017466)

Memoria

**Introducción**

En el siguiente proyecto se propone un sistema, que permitirá innovar la carga de horarios de cursada y mesa de examen de la Unidad Académica UNPA. El objetivo es mejorar el flujo de información entre los alumnos, profesores y Administrativos, a través de la app. Como así también los administrativos de la Secretaria Académica a través de la página web podrán mejorar el acceso a usuarios autorizados para la generación de horarios, facilidad de carga de información a través del uso de los archivos existentes (hojas de cálculo) y el acceso a los horarios tanto de cursada como el de mesa de examen de dicha universidad.

Antes los usuarios hacían sus consultas a través de planillas de Excel que se pegaban en los pizarrones de la entrada del hall o bien en el Portal UARG para su descarga, tanto los horarios de cursada como las mesas de examen. En lo que respecta los administrativos de la Secretaria Académica realizaban estas operaciones antes mencionadas a mano, como se dijo anteriormente en planillas Excel. 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. Por ello es que se va a realizar dicho proyecto.

Se desarrollo los documentos a través del psi, el cual utilizamos como guía de los documentos realizados. Este proyecto se realizo a través de iteraciones

Objetivos

Esto lo encontré en la arquitectura

Objetivos Generales

El objetivo general del Sistema de Gestión de horarios de cursada y mesas de examen, es permitir la realización de la carga de horarios de cursada y mesas de examen haciendo uso de las planillas de calculo que poseen actualmente los empleados de Secretaria Académica de la UNPA-UARG. Esta tarea debe hacerse accediendo al sistema a través de su sitio web e identificándose mediante un correo institucional.

También se permite a usuarios públicos que accedan desde una aplicación móvil consultar los horarios de cursada y mesas de examen que han sido previamente cargados por Secretaría Académica. Recibiendo además notificación por cambios ocurridos en las mesas de examen.

Objetivos Específicos

Para lograr el objetivo mencionado, se describen a continuación metas específicas que se proponen alcanzar.

Integrar uargflow basado en la arquitectura cliente-servidor. Usando el patrón de diseño mvc con POO y PHP para nuestro sitio web.

Además se debe establecer la comunicación entre el sitio web y la app con la base de datos mysql

Investigar técnicas para poder importar un archivo Excel a la base de datos. El esquema de base de datos debe tener en cuenta las características de los archivos que se deben importar al sistema pero sin desperdiciar las características de MER.

Con respecto a la seguridad se debe considerar el acceso a información privada.

Portabilidad por la aplicación.

Integrar los conocimientos mencionados en una solución de software que se convierta en una herramienta práctica para los docentes, alumnos y personal de Secretaria Académica de la UNPA-UARG.

Comunicación en grupo

El proyecto fue realizado a través de un seguimiento, donde utilizamos diferentes tipos de herramientas para poder comunicarnos, generando reuniones sobre los avances ya realizados y los que debíamos seguir o bien consultas sobre el mismo. Dicho esto, las herramientas utilizadas fueron taiga, whatsApp, Zoom y reuniones presenciales.

Taiga fue utilizado únicamente durante la cursada de la materia de laboratorio de proyecto del software, el cual nos fue de mucha utilidad para hacer un alcance de las fases e iteraciones que fuimos realizando, y así poder organizarnos en las actividades que proponíamos para cada integrante del grupo. Allí podíamos agregar las tareas nuevas, en curso y las cerradas que se iban a realzar en cada iteración. Lo negativo de esto fue no haber continuado con la utilización de dicha herramienta, ya que nos permitía organizarnos mejor con esta, y cumplir con los tiempos establecidos en cada iteración.

Reuniones presenciales fueron realizadas algunas en el aula A4 (Laboratorio) durante la cursada de la materia y otras en los hogares de los integrantes. Donde realizábamos todo tipo de actividad, ya sea planificación de las actividades de cada iteración o bien hacer consultas que nos permitían eliminar las dudas que se presentaban en la realización de actividades.

WhatsApp lo utilizamos hasta el último momento del proyecto, nos sirvió para coordinar reuniones, para mostrar o consultar sobre algún avance realizado. Fue una vía de comunicación muy satisfactoria para mantenernos informado de todo.

Correo electrónico del grupo YENU fue un servicio muy importante a la hora de comunicarnos con los profesores de la cátedra, ya que nos permitía tener contacto con ellos, ya sea para realizar consultas sobre el proyecto o bien para obtener una entrevista con ellos. Esto nos permitía evacuar cualquier duda o consejo de los mismos.

Zoom fue una herramienta esencial en tiempos de pandemia, ya que al no poder realizar reuniones en persona por dicho problema nos vimos obligados a manejarnos de esa forma. Esta nos fue de mucha utilidad, ya que nos permitió llevar a cabo las resoluciones o consultas que eran necesarias ver y comentar con el grupo de manera directa.

Especificaciones de Requerimientos

La Especificación de Requerimientos de Software tiene como propósito definir las especificaciones funcionales, no funcionales y del sistema para 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 permita consultar dicha información.

Funciones del Producto

El producto a desarrollar, en líneas generales, permitirá realizar la carga de horarios de cursada y mesas de examen haciendo uso de las planillas de calculo que poseen actualmente los empleados de Secretaria Académica de la UNPA-UARG. Esta tarea debe hacerse accediendo al sistema a través de su sitio web e identificándose mediante un correo institucional.

También se permite a usuarios públicos que accedan desde una aplicación móvil consultar los horarios de cursada y mesas de examen que han sido previamente cargados por Secretaría Académica.

El sistema Tempus, en términos generales, permitirá realizar las siguientes funciones:

* Brindar acceso al personal de Secretaria Académica al sitio web del sistema con la utilización de un correo institucional.
* Importar planilla de cálculo con los horarios de cursada para ser almacenados en el sistema.
* Importar planilla de cálculo con las mesas de examen para ser almacenadas en el sistema.
* Brindar acceso al público en general para consultar horarios de cursada accediendo desde la aplicación móvil del sistema.
* Brindar acceso al público en general para consultar horarios de mesas de examen accediendo desde la aplicación móvil del sistema.

Característica del Usuario

En este apartado se realiza la descripción de las características generales de los usuarios identificados y que harán uso del sistema.

Tempus contendrá tres tipos de usuarios que interactuaran y lo administraran: Administrador, Secretaría Académica y Público. A continuación se brida una breve descripción de cada uno de los usuarios mencionados:

* Administrador: Encargado de administrar el sistema con responsabilidad. Tiene acceso al sistema completo pero tiene la responsabilidad de administrar a los usuarios que tendrán acceso a las tareas privadas. Es responsable de crear usuarios, asignar roles y asignar permisos. Puede ser una persona perteneciente al área de Secretaría Académica. Para esto, el administrador deberá poseer un correo institucional que le permita iniciar sesión en Tempus.
* Secretaría Académica: Su rol en el sistema consiste en realizar la importación de los archivos de horarios de cursada y mesas de examen. Este usuario representa a una persona del área de Secretaría Académica y podrá realizar la creación, eliminación y modificación de horarios de cursada y mesas de examen. Además puede acceder a la generación de informes. Para ello se debe poseer un correo institucional que le permita iniciar sesión en Tempus.
* Público: Representa a alumnos, docentes y público en general. Su rol en el sistema consiste en consultar horarios de cursada y mesas de examen mediante el uso de la aplicación móvil de Tempus. No tiene ningún permiso especial, solo accede a las opciones públicas que brinda el sistema.

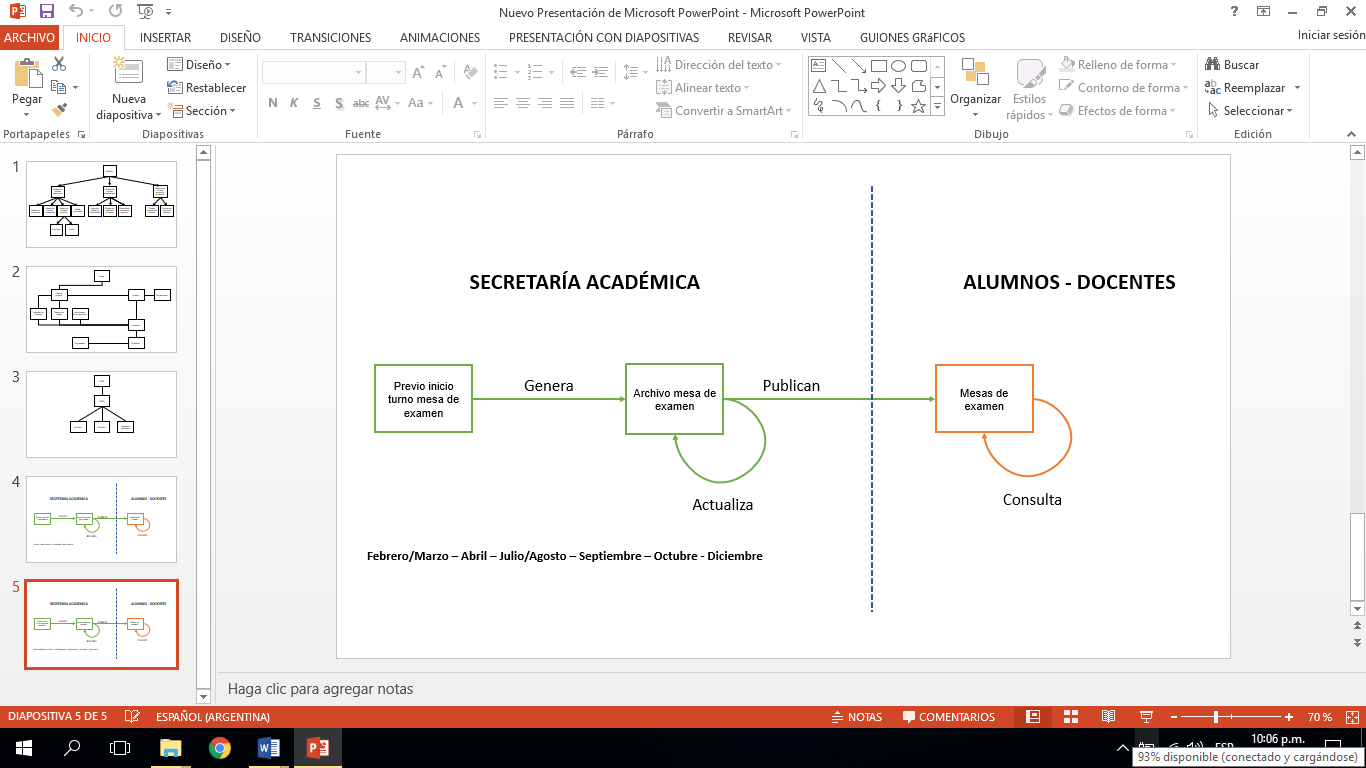
Enunciado del Problema del Cliente

El presente trabajo fue otorgado al Grupo de Desarrollo Yenú 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.

El problema presentado surge ante la necesidad de contar con un sistema que brinde al personal de Secretaria Académica una forma rápida para cargar un conjunto de horarios de cursada y mesas de examen. Además, se debe proporcionar a los alumnos, docentes y público en general una forma rápida y cómoda para consultar dichos horarios de cursada y mesas de examen.

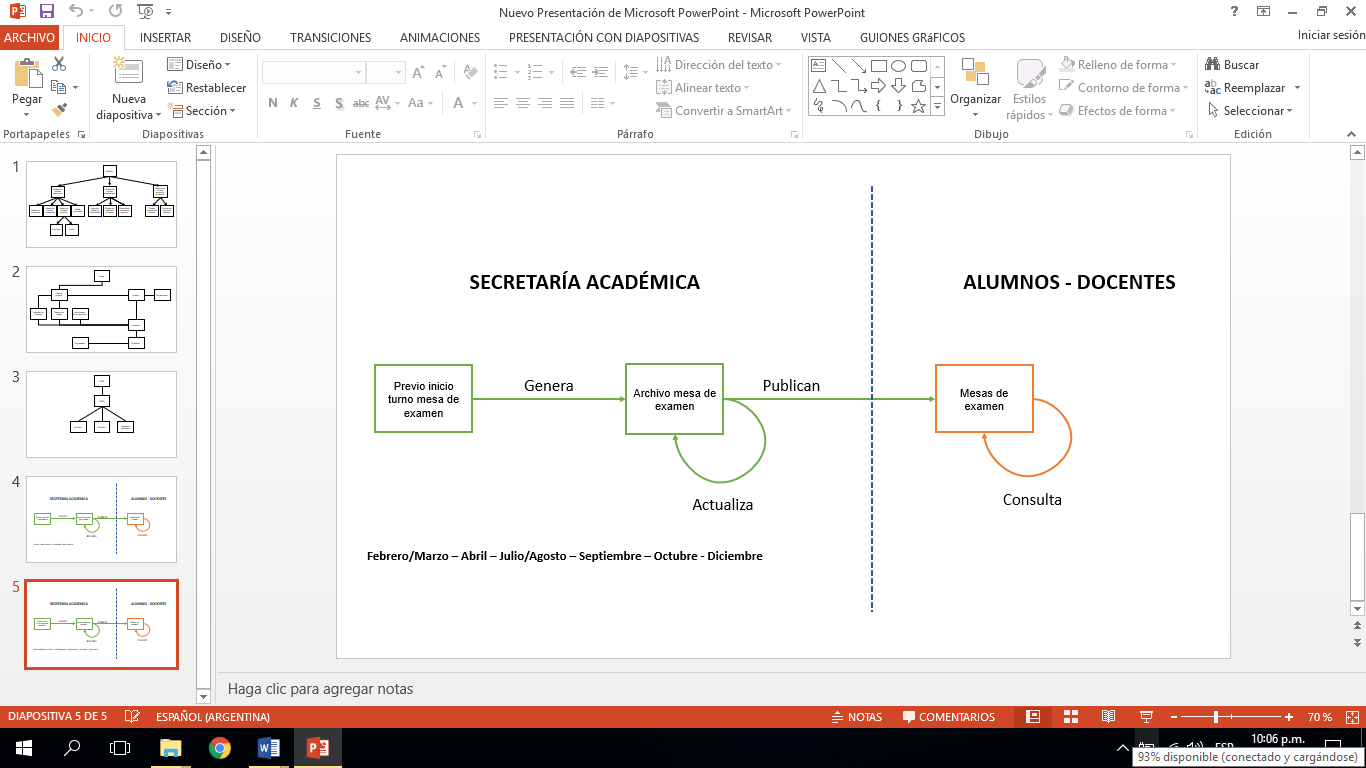
Descripción de los procesos actuales

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



Proceso para la generación de horarios de cursada

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



Proceso para la generación de mesas de examen

A continuación se listan una serie de problemas que se asocian con el enunciado del cliente:

* 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.
* No 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 sean 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.

**Requerimientos**

Requerimientos Funcionales

Los requerimientos funcionales de un sistema describen la funcionalidad o los servicios que se espera que éste provea. En este apartado se debe describe lo que el sistema Tempus tendrá que hacer, los factores que afectan al producto y satisfacen los requerimientos.

A continuación se listan los requerimientos funcionales detectados. Posteriormente se detalla cada uno en una tabla:

1. **El sistema permitirá importar horarios de cursada a usuarios autorizados**. El personal de Secretaria Académica o Administradorpodrá 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 Secretaria 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 Secretaria 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 periodo 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 sobre 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 Secretaria Académica 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.

Requerimientos No Funcionales

Estos requerimientos no funcionales tienen que ver con las características que de una u otra forma pueden limitar al sistema como son: rendimiento, confiabilidad, interfaces, fiabilidad, mantenimiento, seguridad, portabilidad, estándares, etc.

Usabilidad

En este apartado se incluyen todos los requerimientos que afecten la usabilidad. Esto debe incluir: el tiempo que se tomará un usuario en aprender a utilizar el sistema y se podría explicar por qué debe ser rápido el aprendizaje, los tiempos medibles de tarea para las tareas típicas y los requerimientos para concordar con estándares.

1. **El sistema debe seguir el estilo de diseño utilizados definidos para la UARG (UARGFlow).** Se deben respetar los estilos de diseños que son utilizados en las aplicaciones desarrolladas en la UNPA-UARG. Se hace una descripción detallada en la sección Interfaces de Usuario.
2. **El sistema deberá brindar acceso móvil.** Se debe brindar un acceso mediante aplicación móvil a los usuarios públicos para consultar horarios de cursada y mesas de examen.
3. **El sistema deberá brindar acceso web**. Se debe brindar un acceso mediante sitio web a los usuarios autorizados.
4. **La aplicación móvil deberá desarrollarse con el uso del framework Ionic**. Se debe utilizar el framework para el desarrollo de la aplicación móvil multiplataforma.
5. **La aplicación móvil deberá ser multiplataforma**. El sistema Tempus debe estar disponibles para los principales sistemas operativos en dispositivos móviles (Android - IOs).

Confiabilidad

El Requerimientos de confiabilidad del sistema que debe cumplir es:

1. **El sistema debe ejecutarse con normalidad en el entorno operativo del cliente**.

Seguridad

A continuación se detallan los requerimientos de seguridad del sistema.

1. **El sistema deberá permitir iniciar sesión con correo institucional**. Es necesario dado que hay tareas que deben estar disponible solo para personal de Secretaría Académica y el correo institucional sirve para validar la identidad del solicitante.
2. **El sistema deberá identificar al usuario según su tipo**. Es necesario que el sistema identifique al tipo de usuario para que solo se permita el acceso a las tareas que tenga permitido realizar.

Eficiencia

Las características de eficiencia del sistema.

1. **El sistema debe procesar los grandes volúmenes de información de una planilla de cálculo**.

Interfaces

En este apartado se definen las interfaces que debe apoyar la aplicación, como son: las interfaces de usuario o interfaces de software.

Interfaces de Usuario

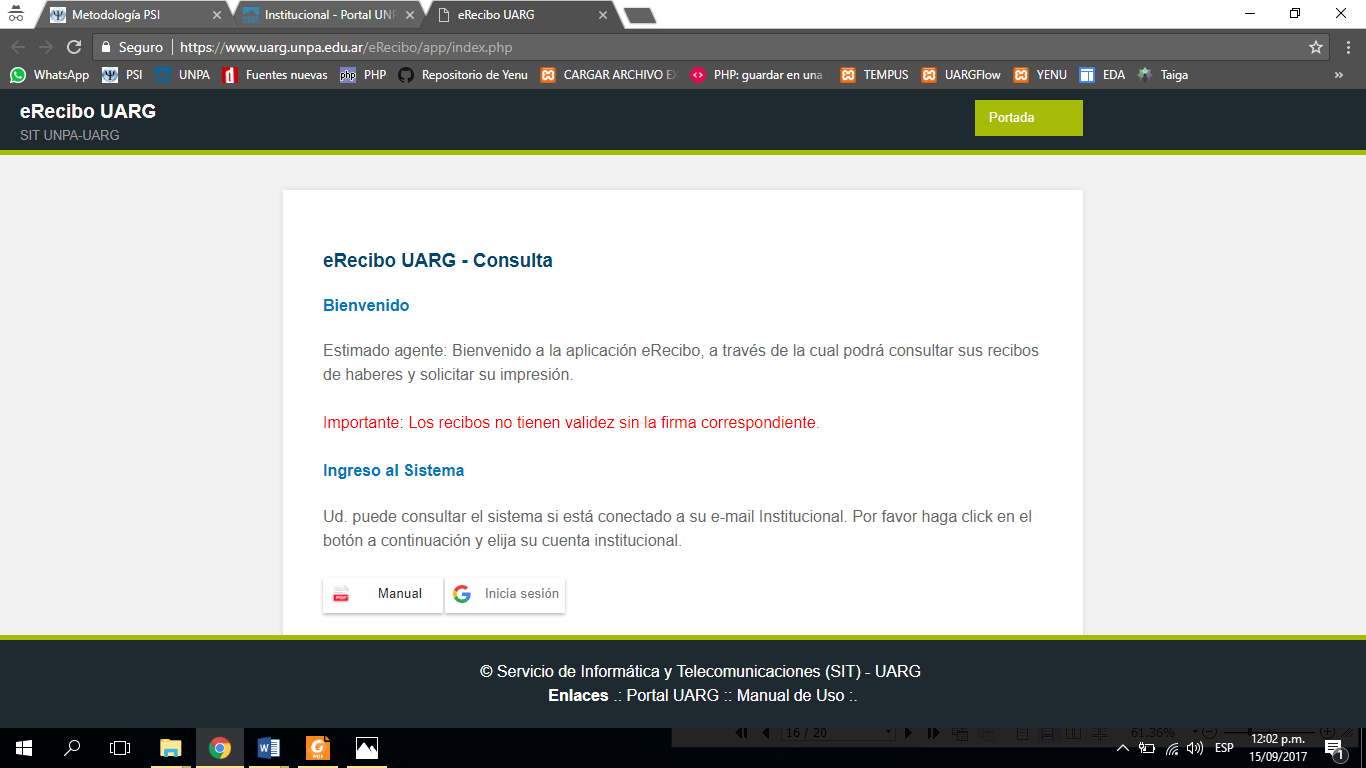
Las interfaces de usuario que se generen para el sitio web deben estar en lengua castellana, seguir las características y estilos que son utilizados para las aplicaciones de la UNPA-UARG.

En forma general se puede dividir a las interfaces web en tres partes: encabezado, contenido y pie. A continuación se hace una pequeña descripción de los componentes que tienen cada una de las partes que se han mencionado.

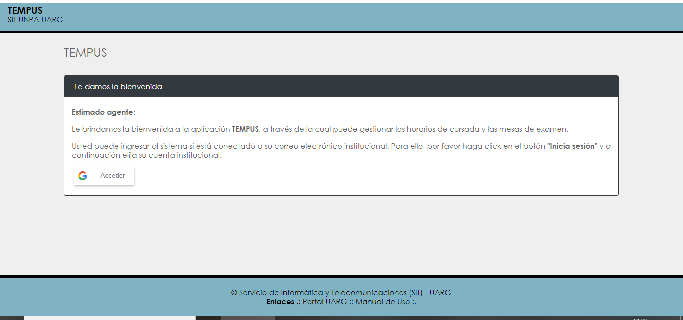
1. Encabezado: El encabezado se encuentra en la parte superior de la pantalla. Sobre el margen izquierdo podrá contener el nombre del sistema y/o el logo del sistema. Sobre el margen derecho podrá contener el logo de la UNPA. En el caso de las interfaces no públicas, se podrá observar en el centro del encabezado el tipo de usuario (Administrador, Secretaria Académica) junto con el menú de opciones correspondiente.
2. Contenido: El contenido se encuentra inmediatamente debajo del encabezado y sobre el pie. En las interfaces públicas se observa una bienvenida junto con una descripción del sistema y una descripción sobre el ingreso al mismo. En las interfaces no públicas se observaran los formularios correspondientes a cada una de las tareas presentadas en el menú de usuario.
3. Pie: El pie se encuentra inmediatamente debajo del contenido. Este podrá contener el logo del grupo de desarrollo junto con una pequeña reseña o información institucional.

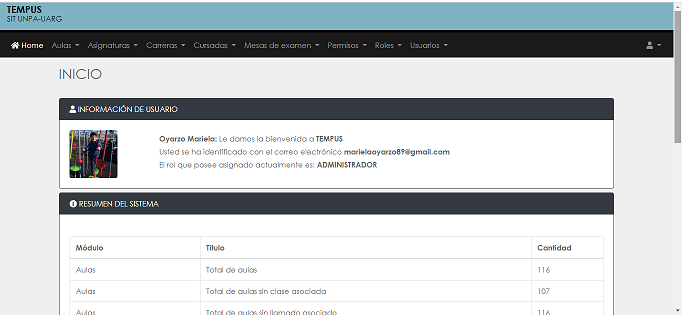
Las interfaces deberán contener una tipografía legible tanto en color como en tamaño.

En cuanto a la aplicación móvil, se deben presentar las tablas de horarios y cursada en un formato que sea observable claramente en cualquier dispositivo.



Ejemplo aplicación eRecibo de la UARG







Interfaces de Software

No se han identificado interfaces de software.

Interfaces de Hardware

No se han identificado interfaces de Hardware para el sistema.

Interfaces de Comunicaciones

[Se debe definir las interfaces de comunicaciones a los demás sistemas o dispositivos como:

Redes LAN y dispositivos seriales remotos, etc.]

Restricción de Diseño

1. **El sistema debe funcionar bajo el servidor disponible para la carrera**. [**CARACTERISTICAS DEL SERVIDOR**]
2. **El sistema no mantendrá almacenado horarios de cursada a modo de historial**. Cuando algún usuario realice la carga de horarios de cursada con la planilla de cálculo, el sistema eliminara todos los horarios que se encuentren cargados hasta ese momento.
3. **El sistema no mantendrá almacenado mesas de examen a modo de historial**. Cuando algún usuario autorizado realice la carga de mesas de examen con la planilla de cálculo, el sistema eliminara todas las mesas de examen que se encuentren cargadas hasta ese momento.

Requerimientos de Licencias

1. **El sistema a desarrollar será un software libre bajo los términos de licenciamiento GNU v3**.

Requerimientos de Documentación

A continuación se especifica el tipo de documentación que se requiere, el contenido y el formato.

Manual de Usuario

El manual de usuario tiene como propósito brindar una explicación completa del sistema para permitir a los usuarios un aprendizaje que requiera poco esfuerzo.

Preferiblemente el manual de usuario contendrá un máximo de 50 hojas con un alto nivel de detalle en las características más importantes y detalle medio en las demás. Dicho documento debe contar con un índice, glosario de términos y las siguientes secciones:

1. Inicio de sesión: Una pequeña introducción y ejemplificación de como iniciar sesión en Tempus siendo usuario registrado.
2. Menú principal: Un presentación del menú principal junto con las opciones disponibles para cada tipo de usuario.
3. Carga y actualización de horarios de cursada: Se hace una explicación y ejemplificación detallada de cómo realizar cada una de las actividades sobre horarios de cursada y sus derivaciones.
4. Carga y actualización de mesas de examen: Se hace una explicación y ejemplificación detallada de cómo realizar cada una de las actividades sobre mesas de examen y sus derivaciones.
5. Consulta de horarios de cursada: Se explica con un nivel medio de detalle sobre cómo realizar la consulta de horarios de cursada.
6. Consulta de mesas de examen: Se explica con un nivel medio de detalle sobre cómo realizar la consulta de mesas de examen.
7. Consulta y actualización de aulas y carreras. Se explica brevemente cada una de las tareas que corresponden a carreras y aulas. Se realiza énfasis en la generación de informe para aulas.
8. Anexo.

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

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

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

Categorización

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

Cronograma: Esta categoría fue elegida porque se considera el tiempo necesario que dispone cada uno de los integrantes del grupo de desarrollo para la realización de las actividades. Esta categoría es importante, ya que el tiempo es necesario para la realización de este proyecto su impacto podría retrasar mucho el mismo, modificar la planificación o bien el costo de dicho proyecto.

Experiencia y Capacidad: Esta categoría fue elegida porque se considera necesario saber el tipo de experiencia o capacidad que tiene cada uno para poder dar avance al proyecto, es decir que la falta de experiencia y capacidad sobre las herramientas, tecnología, metodología o bien el lenguaje de programación que se va a utilizar, trae un retraso al desarrollo, ya que hay que ponerse a estudiar y revisar aquello que desconocemos. Tiempo que podría utilizarse para planificar o realizar actividades de este proyecto.

Duración y Tamaño: Esta categoría fue considerada por los factores que pueden alterar el ritmo de trabajo. Siendo estos aquellos que se presentan durante los meses o años que llevo la realización de este proyecto.

Tecnología: Esta categoría está presente porque se han producido actualizaciones en las herramientas que se utilizaron para el desarrollo de la aplicación móvil de Tempus. Esto no había sido considerado previamente, por lo cual se analiza esta posibilidad. Como respuesta al riesgo se debe seguir reforzando el conocimiento sobre Ionic tanto para las versiones anteriores como en la nueva. Se deben estudiar los cambios para adaptar la versión anterior de nuestra aplicación a la versión actual de Ionic.

Legal y Contractual: Esta categoría se tomo en cuanta, ya que durante el periodo del año 2020, surgió una pandemia que nos limito en parte nuestro desarrollo del proyecto. Esto se debe a que se implementaron decretos que nos limito a realizar reuniones presenciales del grupo. Nuestra manera de poder solucionarlo fue utilizar Zoom, pero como bien sabemos no es lo mismo realizar reuniones virtuales que están restringidas por tiempo a tener una reunión presencial en el cual es mucho más fácil evacuar duda o consultas entre los integrantes del grupo.

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

Gestión de Riesgo

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

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

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

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

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

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

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

Fase de Inicio

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

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

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







Fase Elaboración

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

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

Fase Construcción

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

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

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

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







Conclusión

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

Diagrama de Grantt

Este diagrama fue realizado durante la cursada de la materia Laboratorio de Desarrollo de Software para poder realizar el seguimiento y control del progreso de cada una de las etapas del proyecto (lapso que duro la cursada) teniendo así gráficamente las tareas, su duración y secuencia.



Estructura Organizacional

En una empresa la estructura organizacional da respuesta a una cuestión: ¿Quién es el líder del proyecto y quién toma las decisiones?

La respuesta a esa pregunta puede afectar de manera decisiva al éxito en el proyecto. Este puede alcanzarse de acuerdo a los tres pilares que son el costo, tiempo y calidad. Es decir, que el desarrollo del proyecto se ajuste a la planificación temporal que se ha establecido. Que económicamente no existan desviaciones respecto al presupuesto aprobado. Y que el resultado obtenido alinee con los criterios de calidad solicitados.

Ahora bien en el siguiente cuadro mostraremos los roles necesarios para la realización del proyecto, al cual a cada integrante se le dio uno para poder cumplir con las actividades.

Poner en la presentacion

|  |  |
| --- | --- |
| **Rol** | **Responsable** |
| Administrador de configuración | Oyarzo Mariela. |
| Analista | Todos. |
| Arquitecto | Quiroga Sandra. |
| Diseñador | Oyarzo Mariela. |
| Documentador | Quiroga Sandra. |
| Gerente de calidad | Márquez Emanuel. |
| Ingeniero de pruebas | Todos. |
| Líder del proyecto | Márquez Emanuel. |
| Programador | Todos. |
| Rol general | Todos. |
|  |  |

A continuación, mostraremos el diagrama general de comunicación que utilizamos a la hora de organizarnos o bien mantener una interacción entre los mismos.

La comunicación siempre es clave para cualquier duda, acontecimiento o planificación entre las partes y así evitar cometer posibles errores, y saber a quién preguntar.



Tecnología

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

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

**Estimación del proyecto**

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 método de puntos en casos de uso utilizado, es un método de estimación prometedor 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.

Los casos de uso por sí mismos no permiten efectuar una estimación del tamaño que tendrá el sistema, ni del esfuerzo y el tiempo necesario para implementarlo. Estos permiten documentar los requerimientos del software de una manera compacta y precisa, luego con los puntos de función se puede estimar el tamaño del software a partir de los requerimientos obtenidos de los casos de uso.

Puntos de función de casos de uso consiste en evaluar la complejidad de un sistema de software por medio de una técnica en la que se le asigna una cantidad de puntos de peso, que califican diferentes elementos que componen el sistema de software así como algunos factores del entorno, para obtener una aproximación del tiempo requerido y la cantidad de esfuerzo necesario para la implementación del mismo.

Este proceso se lleva a cabo mediante una serie pasos que como se mencionó anteriormente evalúan cada factor, empezando por ponderar los casos de uso sin ajustar. Esto quiere decir que únicamente son tomados en cuenta los actores (UAW) y los casos de uso (UUCW). Dicho paso se lleva a cabo dejando por el momento los factores técnicos (TCF) y los factores ambientales (EF), para evaluarlos más tarde. Con el fin de multiplicarlos por el resultado final de los casos de uso sin ajustar. Así, se da el resultado de los casos de uso ajustados, que caracteriza la complejidad del sistema y es usado para obtener una idea del número de horas-persona para un proyecto.

Durante el desarrollo de un proyecto de software es necesario estimar, cuánto tiempo se tardará en desarrollar el sistema, cuantas personas se necesitarán para desarrollarlo. Por lo tanto es necesario cuantificar, complejidad del sistema, funcionalidad, complejidad técnica, el nivel de experiencia de los integrantes del proyecto, el tiempo necesario para producir una unidad funcional.  
 Utilizamos un método de estimación desarrollado por Gustav Karner, perteneciente a Rational Software Corporation, el cual caracteriza la complejidad de un sistema a través de Puntos de Casos de Uso.

Se había realizado considerando los casos de uso descriptos en el documento Modelo de Casos de Uso que no habían sido implementados hasta el momento.

Durante la cursada se realizaron 5 estimaciones, donde se tomaban en cuenta los Casos de Uso descriptos en el documento Modelo de Casos de Uso que no habían sido implementados hasta el momento.

La primera estimación inicial, nos dio como resultado 1 año y 3 meses de trabajo. El resultado proviene de calcular:

14 Casos de Usos,

3 actores (administrador, público en general y secretaria académica)

|  |  |
| --- | --- |
| **Información** | **Detalle** |
| Estimación inicial | 1 año y 3 meses |
| Tiempo trabajado | 4 meses y 3 semana (18-Ago-17 al 24-Nov-17)(11-Oct-19 al 22-Nov-19) |
| Ultima estimación (N° 5) | 1 mes y 3 semanas |
| Ultima presentación | 31-Agos-18 |
| Fase | Construcción |
| Iteración | N° 8 (8-Nov-19 al 22-Nov-19) |
| Ultima revisión del repositorio | 511(22-Nov-19) |
| Reanudación del proyecto | 11-Oct-19 |

Casos de Uso

Caso de Uso 01- Ingresar al Sistema

Actores del CU

1. Administrador
2. Base de Datos

Descripción

Este caso de uso es iniciado por el actor. Tiene la opción de ingresar al sitio web del sistema.

Caso de Uso 02- Importar Horario de Cursada

Actores del CU

1. Administrador.
2. Secretaría Académica.



Descripción

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

Caso de Uso 03- Importar Mesa de Examen

Actores del CU

1. Administrador.
2. Secretaría Académica.



Descripción

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

Caso de Uso 04- Buscar Horario de Cursada

Actores del CU

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

Descripción

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

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

Caso de Uso 05- Buscar Mesa de Examen

Actores del CU

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

Descripción

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

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

Caso de Uso 06- Modificar Horario de Cursada

Actores del CU

1. Administrador
2. Secretaria Académica
3. Base Datos

Descripción

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

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

Caso de Uso 07- Modificar Mesa de Examen

Actores del CU

1. Administrador
2. Secretaria Académica
3. Base Datos

Descripción

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

Caso de Uso 08- Crear Horarios de Cursada

Actores del CU

1. Administrador.
2. Secretaría Académica.

Descripción

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

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

Caso de Uso 09- Crear Mesa de Examen

Actores del CU

1. Administrador.
2. Secretaría Académica.

Descripción

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

Caso de Uso 10- Notificar

Actores del CU

1. Público.

Descripción

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

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

Caso de Uso 11- Informe de Mesa de Examen

Actores del CU

1. Administrador.
2. Secretaría Académica.

Descripción

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

Caso de Uso 12- Informe de Horario de Cursada

Actores del CU

1. Administrador.
2. Secretaría Académica.

Descripción

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

Caso de Uso 13- Buscar Aula

Actores del CU

1. Administrador.
2. Secretaría Académica.

Descripción

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

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

Caso de Uso 14- Borrar Horario de Cursada

Actores del CU

1. Administrador.
2. Secretaría Académica.

Descripción

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

Caso de Uso 15- Borrar Mesa de Examen

Actores del CU

1. Administrador.
2. Secretaría Académica.

Descripción

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

Caso de Uso 16- Modificar Aula

Actores del CU

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

Descripción

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

Caso de Uso 17- Borrar Aula

Actores del CU

1. Administrador.
2. Secretaría Académica.

Descripción

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

Caso de Uso 18- Generar Horario de Aula

Actores del CU

1. Administrador.
2. Secretaría Académica.

Descripción

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

Caso de Uso 19- Crear Usuario

Actores del CU

1. Administrador.

Descripción

Este caso de uso es iniciado por el actor. Permite crear un nuevo usuario.

Caso de Uso 20- Buscar Usuario

Actores del CU

1. Administrador.

Descripción

Este caso de uso es iniciado por el actor, el cual permite buscar un usuario. Este caso de uso se relaciona con los casos de uso borrar y modificar usuario.

Caso de Uso 21- Modificar Usuario

Actores del CU

1. Administrador.

Descripción

En este caso de uso el actor busca el usuario, para luego este ser modificado

Caso de Uso 22- Borrar Usuario

Actores del CU

1. Administrador.

Descripción

En este caso de uso el actor busca el usuario que desea borrar.

Caso de Uso 23- Crear Rol

Actores del CU

1. Administrador.

Descripción

El actor planifica crear un nuevo tipo de usuario del sistema, de acuerdo a sus funciones en la estructura administrativa de la UARG.

Caso de Uso 24- Buscar Rol

Actores del CU

1. Administrador.

Descripción

En este caso de uso el actor busca un rol de usuario.

Caso de Uso 25- Modificar Rol

Actores del CU

1. Administrador.

Descripción

El actor busca el rol de usuario que va hacer modificado. Ahora el actor planifica modificar un rol de usuario.

Caso de Uso 26- Borrar Rol

Actores del CU

1. Administrador.

Descripción

En este caso de uso se busca el rol de usuario que va a borrar.

Caso de Uso 27- Crear Permiso

Actores del CU

1. Administrador.

Descripción

En este caso de uso el actor planifica crear un nuevo permiso del sistema.

Caso de Uso 28- Buscar Permiso

Actores del CU

1. Administrador.

Descripción

En este caso de uso el actor planifica buscar un permiso del sistema.

Caso de Uso 29- Borrar Permiso

Actores del CU

1. Administrador.

Descripción

En este caso de uso el actor busca el permiso del sistema para luego borrarlo.

***Diagrama de Grantt***

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

Conclusión personal Mariela Oyarzo

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 mas 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 mas y enriquecer aun mas dichos conocimiento y poder llevar a cabo el desarrollo a traves 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 aun como futuros profesionales de esta carrera que elegimos

Anexo

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.

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.

### 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.

 IONIC: 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.

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.

 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.

En este proyecto se hace uso de esta herramienta para desarrollar la aplicación móvil.

### XAMPP: es 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.

**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.

 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

**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.

Reuniones

|  |  |
| --- | --- |
| Reunión Nº 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 Nº 2 | |
| Fecha : 26/08/2017 | |
| Lugar: Domicilio de Sandra | |
| Temas | * Prototipos desechable de aplicaciones móvil * 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 Nº 3- Fase de Inicio Iteración 1 | |
| Fecha : 30/08/2017 | |
| Lugar: Vía WhatsApp | |
| Temas | * Prototipos desechables de la pagina 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 Nº 4- Fase de Inicio Iteración 1 | |
| Fecha : 01/09/2017 | |
| Lugar: Vía WhatsApp | |
| Temas | Modelo de Negocio.  Glosario |
| 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 Nº 5 Fase de Inicio Iteración 1 | |
| Fecha : 04/09/2017 | |
| Lugar: Via WhatsApp | |
| Temas | * Plan del proyecto * Modelo de negocio * Modelo de Caso 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 si. * 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 Nº 6 Fase de Inicio Iteración 1 | |
| Fecha : 06/09/2017 | |
| Lugar: Vía WhatsApp | |
| Temas | * Requerimientos de usuario * Requerimientos de usuario * Actores * Casos de uso * Diagrama de Casos Uso |
| Resolución | * Posibles requerimientos de usuario * Administrar usuarios: crear, buscar, borrar y modificar usuarios. * Administrar roles: crear, buscar, borrar y modificar roles. * Administrar permisos: crear, buscar y borrar permisos. * Administrar horarios de cursada: cargar, buscar, modificar, crear, informe horarios de cursada. * Administrar mesas de examen: cargar, buscar, modificar, crear, informe mesas de examen. * Administrar aula: buscar, borrar, modificar, generar horarios de aula. * Administrar carrera: buscar, borrar y modificar 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 * Crear usuario. * Buscar usuario. * Borra usuario. * Modificar usuario. * Crear rol. * Buscar rol. * Modificar rol. * Borrar rol. * Crear permiso. * Buscar permiso. * Borrar permiso. |

|  |  |  |
| --- | --- | --- |
| Reunión Nº 7 Fase de Inicio Iteración 1 | | |
| 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:   Redactar los planes del proyecto.  Finalizar el Modelo de Negocio.  Iniciar el Modelo de Casos de Uso junto con la priorización de los mismos.  Redactar el Modelo de Visión.  Redactar Especificación de Requerimientos de Software.  Iniciar la Arquitectura del Sistema.  Gestionar riesgos.  Estimación del proyecto. |
| 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 Nº 8 Evaluación Todo hasta Fase Construcción Iteración 1 | |
| Fecha : 14/10/2017 | |
| Lugar: Domicilio de 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 ha 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 Nº 9 Fase Construcción Iteración 1 | |
| Fecha : 28/10/2017 | |
| Lugar: Domicilio de Sandra | |
| Temas | * Especificación de Caso de Uso * Modelo de Datos * Modelo 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. Luego en el diagrama de despliegue.(que ponemos?) * 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 Nº 10 Fase Construcción Iteración 2 (Evaluación de la iteración concluida) | |
| Fecha : 13/11/2017 | |
| Lugar: Vía WhatsApp | |
| Temas | Al finalizar la presente iteración se deberán haber alcanzado los siguientes objetivos:  Finalizar implementación:  CU02 – Importar horarios de cursada.  CU03 – Importar mesas de examen.  Implementar los siguientes casos de uso:  CU04 – Buscar horario de cursada (APP).  CU05 – Buscar mesa de examen (APP).  CU06 – Modificar horario de cursada.  CU07 – Modificar mesa de examen.  CU08 – Crear horario de cursada.  CU09 – Crear mesa de examen.  Redactar casos de prueba para:  CU04 – Buscar horario de cursada (APP).  CU05 – Buscar mesa de examen (APP).  CU06 – Modificar horario de cursada.  CU07 – Modificar mesa de examen.  CU08 – Crear horario de cursada.  CU09 – Crear mesa de examen.  Ejecutar los casos de pruebas para los casos de uso:  CU02 – Importar horarios de cursada.  CU03 – Importar mesas de examen.  CU04 – Buscar horario de cursada.  CU05 – Buscar mesa de examen. |
| Resolución | * Objetivos alcanzados:   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). * Objetivos no alcanzados:   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 Nº 11 Fase Construcción Iteración 3 | |
| 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. * Aspectos generales: * Un integrante que tenía conocimiento y experiencia previa a la asignatura,, en la forma de trabajo y en las herramientas dispuestas por el equipo de cátedra. * Los integrantes tuvimos un ambiente de trabajo excelente. * Aprovechamos al máximo cada presentación de nuestros compañeros. * Los profesores siempre tuvieron buena pre disposición. * Conclusión: 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. * Se han superado las expectativas. * Se mostro los casos de uso que fueron implementados hasta el momento, tanto de la pagina web como los de la App. |

|  |  |
| --- | --- |
| Reunión Nº 12 Presentación sobre la continuidad del proyecto | |
| Fecha : 17/08/2018 | |
| Lugar: Domicilio de Sandra | |
| Temas | * Objetivo de la presentación. * 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: * Comprometerse con la finalización del trabajo. * Repasar la documentación actual disponible en el repositorio (Github). * Determinar el estado de la documentación. * Planificar nueva iteración. * Realizar presentación al equipo de cátedra. * Descripción general:   En forma resumida:   * Cargar archivo con horarios de cursada (Cada cuatrimestre). * Cargar archivo con mesas de examen (Cada turno). * Generar informes. * Consultar horarios de cursada (APP). * Consultar mesas de examen (APP). * Recibir notificaciones sobre horarios de cursada y mesas de examen ante modificaciones (APP). * Estado del proyecto: * Estimación inicial: 1año y 3 meses. * Tiempo trabajado: 3 meses y 1 semana. * Última estimación: 4 meses y 3 semanas. * Documentación restante: * Manual de instalación * Manual de usuario * Memoria * Actualizaciones sobre las actuales: Se considera actualizaciones a los documentos actuales ante eventuales modificaciones. Abarca documentos como: Especificaciones de casos de uso, Modelos (De datos, de casos de uso, de diseño) o base de datos. * Los casos de uso (cu10 al cu18) se irán completando en el orden que se indica. * Planificación de la nueva iteración: * Fase de Construcción Iteración 4: * 01/09/2018 al 16/09/2018 * Las actividades se encuentran sujetas a las conclusiones obtenidas durante esta presentación por lo que pueden ser modificada posteriormente. * 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. * Realizar notificaciones a correos electrónicos ante modificaciones. Respecto a: * Mesas de examen: asignación de aula, cambio de día, hora o aula. * Horarios de cursada: cambio de día, hora o aula. * El grupo de desarrollo presenta una solución: * Permitir el ingreso de un correo electrónico. * Tener la opción de almacenar favoritos solo a aquellos usuarios que presenten un correo electrónico. Las notificaciones se hacen sobre los favoritos almacenados en la aplicación al correo indicado. * Limitar la cantidad de favoritos para evitar usuarios maliciosos. * Se presenta un prototipo de la App. |

|  |  |
| --- | --- |
| Reunión Nº 13 Evaluación de Fase de Construcción Iteración 4 | |
| Fecha : 16/09/2018 | |
| Lugar: Domicilio de Sandra | |
| Temas | * Hicimos una evaluación de los objetivos que habíamos planificado en la iteración y fueron los siguientes: * Continuar con el glosario * Descargar archivos Excel del año 2018 * Completar los casos de prueba para CU02 * Completar los casos de prueba para CU03 * Completar CU11 informe de mesa de examen * Iniciar manual de usuario (App). * Modificar CU10- Notificar * Modificar Modelo de Caso de Uso * Continuar Arquitectura del sistema * Continuar Plan de pruebas * Finalizar CU04- Buscar horario de cursada (APP). * Implementar CU10- Notificar * Modificar CU06- Modificar horario de cursada * Modificar CU07- Modificar Mesa de Examen * Modificar Modelo de Datos * Continuar Modelo de diseño * Implementar modificaciones CU06- Modificar horario de cursada. * Implementar modificaciones CU07- Modificar horario de mesa de examen * Planificar nueva Iteración. |
| Resolución | * Objetivos Alcanzados a continuación se e 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). * Objetivos No Alcanzados a continuación se detallan los objetivos que no se han alcanzado: * El resto de las actividades no se han finalizado. * 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 Nº 14 Evaluación de Fase de Construcción Iteración 5 | |
| Fecha : 30/09/2018 | |
| Lugar: Domicilio de Sandra | |
| Temas | * Hicimos una evaluación de los objetivos que habíamos planificado en la iteración y fueron los siguientes: * Continuar con el glosario * Descargar archivos Excel del año 2018 * Completar los casos de prueba para CU02 * Completar los casos de prueba para CU03 * Completar CU11 informe de mesa de examen * Iniciar manual de usuario (App). * Modificar CU10- Notificar * Realizar documento “Formato Archivo de Mesa de Examen” * Continuar Modelo de diseño * Implementar modificaciones CU06- Modificar horario de cursada. * Implementar modificaciones CU07- Modificar horario de mesa de examen * Cierre de planificación actual. * Planificar nueva Iteración. |
| Resolución | * Objetivos Alcanzados 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. * Objetivos No Alcanzados 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 realizo se deberá terminar en la siguiente iteración. |

|  |  |
| --- | --- |
| Reunión Nº 15 Evaluación de Fase de Construcción Iteración 6 | |
| Fecha : 15/10/2018 | |
| Lugar: Domicilio de Sandra | |
| Temas | * Luego de realizar una evaluación general de las actividades que fueron planificadas para esta iteración, fueron las siguientes : * 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 CU04 – Buscar horario de cursada (APP). * Implementar CU10 – Notificar. * Crear archivo Tempus mesas de examen octubre 2018. * Continuar Modelo de Diseño. * Implementar CU08 – Crear horario de cursada. * Implementar CU09 – Crear mesa de examen. * Iniciar implementación CU11-Informe mesa de examen * Cierre de la planificación actual * Planificar nueva iteración. |
| 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. * Cierre de la planificación actual * Planificar nueva iteración. * Aquello que no se termino será realizado en la siguiente iteración. |

|  |  |
| --- | --- |
| Reunión Nº 16 Evaluación Fase de construcción iteración 7 | |
| Fecha : 26/10/2019 | |
| Lugar: Domicilio de Sandra | |
| Temas | * Luego de realizar una evaluación general de las actividades que fueron planificadas para esta iteración, fueron las siguientes : * Continuar glosario. * Descargar archivo de Mesas de examen Octubre 2019. * Descargar archivo de Horarios de Cursada Octubre 2019. * Analizar riesgos * Actualizar código para realizar pruebas * Continuar manual de usuario (APP). * Continuar manual de usuario (Rol administrativo) * Continuar manual de usuario (Rol administrador) * Continuar Modelo de Casos de Uso. * Continuar Arquitectura del Sistema. * Continuar Plan de pruebas. * Estudiar cambios en la versión de Ionic * Crear archivo Tempus mesas de examen octubre 2019. * Crear archivo Tempus Horarios de cursada octubre 2019. * Continuar el modelo de diseño * Cierre de planificación actual * Planificar nueva iteración |
| Resolución | * Los objetivos alcanzados fueron:   Se realizó la evaluación general del estado del proyecto.  Se realizó la evaluación de los riesgos para la iteración actual. |

|  |  |
| --- | --- |
| Reunión Nº 17 Evaluación general hasta la fecha | |
| Fecha : 17/11/2019 | |
| Lugar: Domicilio de Mariela | |
| Temas | * Descripción general del proyecto * Estado actual del proyecto * Videos de la pagina web |
| Resolución | * Descripción general:Objetivo crear un sistema web y una aplicación (App) para la gestión de horarios de cursada y mesas de examen. * Destinado principalmente a: * S*ecretaria académica UNPA-UARG:* quienes se encargarían de administrar las cursadas y mesas de examen. Además de acceder a los informes. Esta tarea debe hacerse accediendo al sistema a través de su sitio web e identificándose mediante un correo institucional. * *Administrador: Encargado de administrar usuarios, roles y permisos. Tiene acceso al sistema completo*. * Se muestra una captura de pantalla de la pagina web Tempus –Búsqueda de horarios de cursada * Aplicación móvil(app): * Se permite a usuarios públicos que accedan desde una aplicación móvil consultar los horarios de cursada y mesas de examen que han sido previamente cargados por Secretaría Académica. * Se podrá guardar en favoritos aquellos horarios de cursada y mesas de examen de interés para el alumno o docente. Resaltando en favoritos cuando ocurra algún cambio. * Gestión de riesgos: Se analiza el nuevo riesgo que surgió siendo este RK11: “Dado el avance tecnológico se puede cambiar la tecnología de desarrollo durante el proyecto y afectar lo antes desarrollado. “. Este surgió porque cambio la versión antigua del Framework Ionic, lo cual trajo inconvenientes a la hora de seguir con la app. Una solución fue seguir reforzando el conocimiento sobre Ionic tanto para las versiones anteriores como en la nueva. Se deben estudiar los cambios para adaptar la versión anterior de nuestra aplicación a la versión actual de Ionic. * Estado del proyecto: * Ultima presentación 31/08/2018 y 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 Nº 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 donde 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 Nº 19 | |
| Fecha : 23/06/2020 | |
| Lugar: Vía Zoom | |
| Temas | * Gestión de riesgos * Pagina web * Características de los nuevos celulares * Memoria |
| 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 Nº 20 | |
| Fecha : 30/06/2020 | |
| Lugar: Zoom | |
| Temas | * Iteraciones (Taiga) * Objetivo principal * Objetivos específicos |
| Resolución | * Usamos durante la cursada una herramienta llamada Taiga, que nos permitía tener una guía de las actividades que teníamos cada uno de los integrantes como así también aquellas que habíamos finalizado o las que teníamos en curso, para ser terminada en la siguiente iteración * **Objetivo principal:** es permitir la realización de la carga de horarios de cursada y mesas de examen haciendo uso de las planillas de cálculo que poseen actualmente los empleados de Secretaria Académica de la UNPA-UARG. Esta tarea debe hacerse accediendo al sistema a través de su sitio web e identificándose mediante un correo institucional.   También se permite a usuarios públicos que accedan desde una aplicación móvil consultar los horarios de cursada y mesas de examen que han sido previamente cargados por Secretaría Académica. Recibiendo además notificación por cambios ocurridos en las mesas de examen.   * Objetivos Específicos:   Integrar uargflow basado en la arquitectura cliente-servidor. Usando el patrón de diseño mvc con POO y PHP para nuestro sitio web.   * El esquema de base de datos debe tener en cuenta las características de los archivos que se deben importar al sistema pero sin desperdiciar las características de MER. * se debe establecer la comunicación entre el sitio web y la app con la base de datos mysql * Portabilidad por la aplicación.   Con respecto a la seguridad se debe considerar el acceso a información privada. |

|  |  |
| --- | --- |
| Reunión Nº 14 Fase de Inicio Iteración 1 | |
| Fecha : 11/09/2017 | |
| Lugar: Vía Whatsapp | |
| Temas |  |
| Resolución |  |

|  |  |
| --- | --- |
| Reunión Nº 14 Fase de Inicio Iteración 1 | |
| Fecha : 11/09/2017 | |
| Lugar: Vía Whatsapp | |
| Temas |  |
| Resolución |  |

Conclusión General:

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 trato 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.

Biblografia

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

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