

Propuesta de Desarrollo

Kairos

NexTech

Centurión Valeria, Escalante Guillermo, Maldonado Agustina, Mendez Florencia, Ulloa Gonzalo.

**

**



**Índice**

[**Introducción 4**](#_kg9jvkkhxu5d)

[*Propósito 4*](#_3d022d4vkn3v)

[*Alcance 4*](#_7wp4e8n8oavl)

[*Panorama General 4*](#_klvprnu54g30)

[**Información General 4**](#_8wzl7t6co72y)

[*Grupo de desarrollo: 4*](#_v8cxcc8a2ohh)

[*Actividad Principal del Equipo de Trabajo: 4*](#_b919g2gnd7kl)

[*Antecedentes Vinculados a la Temática 5*](#_n0p9m9lpd0hj)

[**Propuesta 5**](#_2trtkbmpe1i1)

[*Análisis Técnico 5*](#_y6qzyxtbfrc0)

[Definición de problema: 5](#_r9t6ppu4f7pz)

[Características actuales del sistema 5](#_68otjuogggaw)

[Descripción de procesos actuales 5](#_x7htdotgsm3o)

[Requisitos detectados 5](#_pnvm71g4v6ha)

[*Propuesta Técnica 6*](#_ujv0vk1d9t4w)

[Casos de uso 6](#_zzklo1r1j76)

[*Planificación Estimada 6*](#_q3ahti7eu14y)

[**Propuesta Económica 6**](#_plh4azno0pu9)

[**Anexos 6**](#_5q4cq9tpwnjs)

Propuesta de Desarrollo

**Introducción**

El presente documento tiene como objetivo describir la propuesta de desarrollo de un sistema de Planificación y Registro de Tiempos, cuyo propósito es dar apoyo a los equipos de desarrollo conformado por estudiantes con una herramienta sencilla de utilizar, que permite organizar tareas, crearlas, estimar y registrar tiempos, así como también generar reportes y planificar proyectos **Propósito**

El propósito de esta propuesta de desarrollo es presentar de manera clara y estructurada la necesidad, justificación, alcance, plan técnico y económico del sistema Kairos, con el fin de obtener la validación y aprobación para llevar adelante su implementación.

**Alcance**

El alcance del presente documento abarca la propuesta de desarrollo del sistema de Planificación y Registro de Tiempos, incluyendo un análisis técnico acompañado de una propuesta técnica, así como también establecer las características principales del producto a desarrollar, los beneficios, las metas a lograr, requisitos funcionales y no funcionales identificados.

Finalmente, una vez redactada y presentada la propuesta de desarrollo, se espera establecer las bases que servirán para la validación y aprobación del proyecto.

**Panorama General**

El contenido y la organización de este documento será el siguiente:

**Introducción**: En primer lugar se describe el propósito y alcance de la propuesta de desarrollo, las características generales del producto, los antecedentes y requisitos detectados.

**Propuestas**: Se procede a establecer la propuesta técnica, la planificación estimada y la propuesta económica.

**Anexos**: Finalizando, se incluirá la sección de anexos con referencias a entrevistas (si las hubiese), referencias y comentarios (opcionales) para el desarrollo del sistema.

**Acrónimos**

**UARGFlow**: Sistema de autenticación utilizado por la Universidad Nacional de la Patagonia Austral (UNPA) UARG.

**Definiciones**

* **Dashboard:** tablero visual con información clave del proyecto.
* **PSI**: El Proceso PSI es un marco de trabajo basado en el Proceso Unificado para el Desarrollo de Software. Se caracteriza por estar dirigido por Casos de Uso, centrado en la arquitectura y por ser iterativo e incremental.

**Información General**

**Grupo de desarrollo:**

El grupo que se encargará de desarrollar el sistema sobre planificación y registro de tiempos está compuesto por:

* **C**enturión Valeria, este miembro estará a cargo del rol de Analista y Programador.
* **E**scalante Guillermo, este miembro estará a cargo del rol de Analista y Programador.
* **M**aldonado Agustina, este miembro estará a cargo del rol de Diseñador y Documentador.
* **M**endez Florencia, este miembro estará a cargo del rol de Diseñador, Documentador y Gerente de Calidad.
* **U**lloa Gonzalo, este miembro estará a cargo del rol de Líder de Proyecto, Administrador de configuración y Tester (Gestor de Pruebas).

**Actividad Principal del Equipo de Trabajo:**

Las actividades en las que el grupo de trabajo se encuentra abocado en la actualidad es el desarrollo del sistema Kairos, un proyecto realizado durante la cursada de la asignatura Laboratorio de desarrollo de software en las carreras Analista de Sistemas y Licenciatura en Sistemas de la Universidad Nacional de la Patagonia Austral con sede en la localidad de localidad de Río Gallegos, el cual consiste en desarrollo de una herramienta para la planificación y registros de tiempos en los proyectos académicos (o de otra área que le resulte relevante). El propósito del proyecto Kairos es poder aplicar las metodologías adquiridas en la carrera sobre la ingeniería de software para construir un sistema capaz de facilitar y optimizar las etapas de planificación que surgen al inicio de un Proceso de Desarrollo de Software, así cómo también permitir el registro en tiempo real de una tarea que se encuentre dentro de una etapa, fase o iteración de la planificación, considerando estas dos actividades como partes fundamentales para poder obtener una buena Gestión de Proyectos y lograr productos de calidad.

El equipo de desarrollo está utilizando el PSI para garantizar la calidad del producto final. En esta se realizan etapas como, relevamiento de requerimientos, entendimiento del modelo de negocio, definición de la arquitectura del sistema, etc

**Antecedentes Vinculados a la Temática**

Durante el desarrollo del proyecto, se pide a los alumnos de la asignatura ‘Laboratorio de Desarrollo de Software’, planificar las actividades que deben realizarse en las distintas etapas del mismo, de manera que puedan organizarse, distribuir tareas, realizar estimaciones y registrar los tiempos que lleva cumplir con cada actividad planificada. Al no haber un método definido para hacerlo, los estudiantes utilizan una variedad de herramientas de software que no cuentan con todas las funcionalidades necesarias de manera unificada, como lo son algunas de ellas: Toggl Track, Clockify, Notion, Jira, entre otras.

**Propuesta**

La presente propuesta consiste en el desarrollo de un sistema de Planificación y Registro de Tiempos, orientado a equipos de estudiantes que participan en proyectos de software en el marco de la materia Laboratorio de Desarrollo. El sistema permitirá planificar tareas, registrar tiempos estimados y reales, visualizar reportes comparativos y generar alertas ante retrasos, brindando así un soporte integral tanto para los integrantes como para el líder del equipo.

El proyecto será llevado adelante por el grupo de desarrollo **NexTech**, conformado por Centurión Valeria, Escalante Guillermo, Maldonado Agustina, Méndez Florencia y Ulloa Gonzalo, quienes se encargarán de todas las etapas: análisis, diseño, implementación y pruebas, así como también toda documentación complementaria.

El entorno del sistema a desarrollar será principalmente académico, con posibilidad de adaptarse a otros tipos de proyectos. Se priorizará la facilidad de uso, la generación de reportes exportables y la versatilidad para que el software pueda aplicarse en distintos equipos y contextos de desarrollo.

**Análisis Técnico**

Actualmente, los equipos que desarrollan proyectos de software en la materia de Laboratorio de Desarrollo carecen de una herramienta unificada para la planificación y el registro de tiempos. Cada integrante suele organizar y registrar sus tareas de manera individual y con distintos criterios y herramientas (Notion, Jira, Clockify, Toggl Track), lo que genera falta de homogeneidad, dificultades para consolidar la información y problemas para evaluar el progreso real.

**Definición de problema:**

El proyecto ‘Kairos’ busca resolver la falta de una herramienta común y estandarizada para registrar los tiempos de las tareas realizadas por los estudiantes en proyectos de desarrollo de software. Esta carencia genera dificultades para estimar esfuerzos, evaluar avances y detectar retrasos, lo que impacta negativamente en la planificación y el control de las distintas etapas del proyecto.

**Características actuales del sistema**

No existe un sistema informático específico en uso actualmente.

**Descripción de procesos actuales**

El proceso de planificación y registro de tiempos actualmente, para los estudiantes de la carrera AdeS/LeS, están dados de manera manual, con la utilización de herramientas externas cómo las plataformas de registro de tiempos (ej: ‘Toggl Track’), o plataformas para planificar y gestionar tareas, (ej: Trello), entre otras. Una vez registrado los tiempos, se procede a utilizar plantillas de cálculos cómo hojas de excel para registrar estimaciones de esfuerzo.

**Requisitos detectados**

**Gestión de usuarios y roles**

1. El sistema **deberá permitir** al líder de proyecto registrar el nombre y correo electrónico de los usuarios que tendrán acceso.
2. El sistema **deberá permitir** al líder de proyecto asignar roles a los usuarios.
3. El sistema **deberá permitir** la creación de un perfil de usuario con los siguientes atributos: nombre, rol y correo electrónico.
4. El sistema **deberá permitir** la asignación de múltiples roles a un mismo usuario.
5. El sistema **deberá contar** con autenticación de usuarios para garantizar el acceso controlado, utilizando UARGFlow y su cuenta de Google.

**Gestión de proyectos**

1. El sistema **deberá permitir** a los usuarios crear proyectos.
2. El sistema **deberá permitir** el ingreso de múltiples usuarios a un mismo proyecto.
3. El sistema **deberá permitir** al líder de proyecto asignar los usuarios participantes de cada proyecto.

**Planificación**

1. El sistema **deberá permitir** planificar las etapas de un proyecto.
2. El sistema **deberá permitir** planificar iteraciones dentro de cada etapa.

**Gestión de tareas**

1. El sistema **deberá permitir** realizar estimaciones sobre las tareas creadas.
2. El sistema **deberá permitir** añadir nuevas tareas a la lista de tareas de cada iteración.
3. El sistema **deberá permitir** a los líderes de proyecto añadir, modificar y eliminar categorías de tiempos (por ejemplo, *Codificación*).
4. El sistema **deberá permitir** categorizar tareas fuera de planificación.
5. El sistema **deberá permitir** que los miembros propongan tareas fuera de la planificación oficial. Estas tareas deberán quedar en estado “pendiente de aprobación” hasta que el líder de proyecto las acepte o rechace.
6. El sistema **deberá permitir** que solo las tareas aprobadas pasen a formar parte del backlog y puedan ser ejecutadas.
7. El sistema **deberá permitir** asignar dependencias entre tareas.
8. El sistema **no deberá permitir** el inicio de una tarea que dependa de otra que aún no se encuentre finalizada.
9. El sistema **deberá permitir** modificar el estado de una tarea.
10. El sistema **deberá permitir** visualizar los diferentes estados de cada tarea.
11. El sistema **deberá permitir** realizar comentarios sobre una tarea.

**Registro de tiempos y seguimiento**

1. El sistema **deberá permitir** registrar el tiempo real trabajado en las actividades en el momento en que se realizan.
2. El sistema **deberá permitir** al usuario ingresar manualmente el tiempo trabajado en una tarea cuando este se haya realizado fuera del sistema.
3. El sistema **deberá permitir** que, además del responsable asignado, otros miembros del proyecto puedan registrar horas en una tarea.
4. El sistema **deberá mostrar** al finalizar la iteración el desglose del tiempo aportado por cada miembro que trabajó en dicha tarea.
5. El sistema **deberá brindar** una visualización global de los avances alcanzados durante el desarrollo del proyecto.
6. El sistema **deberá permitir** realizar exportaciones de información en distintos formatos (PDF, Excel).

**Propuesta Técnica**

**Alcance**

* Planificación de tareas con asignación de responsables y tiempos estimados.
* Registro de tiempos reales de ejecución.
* Comparación entre tiempos planificados y reales.
* Panel de control con reportes (tablas, gráficos, dashboards).
* Notificaciones y alertas ante retrasos.
* Gestión de usuarios y roles (multiusuario, multiproyecto, multirol).
* Exportación e importación de datos en formatos estándar (CSV, PDF, Excel).

**Características**

* Arquitectura cliente-servidor con base de datos centralizada.
* Interfaz simple, clara y de uso rápido (criterio de usabilidad).
* Registro de tareas en pocos segundos, sin interrumpir el trabajo.  
  Posibilidad de integración con *UARGFlow* para autenticación.

**Participantes del desarrollo (Grupo NexTech):**

* Centurión Valeria
* Escalante Guillermo
* Maldonado Agustina
* Mendez Florencia
* Ulloa Gonzalo

**Casos de uso**

Los actores detectados son:

* **Miembro** (miembro del proyecto) → actor complejo (interfaz con el sistema).
* **Líder de proyecto** → actor complejo (interacciones más complejas).
* **Docente/Admin** → actor complejo (interactúa con el sistema y registra usuarios).

**Los casos de uso detectados son los siguientes:**

### 1. Gestión de usuarios y autenticación

* El usuario quiere autenticarse en el sistema para acceder de manera segura a sus proyectos.
* El usuario quiere iniciar sesión con su cuenta de Google para entrar al sistema sin crear nuevas credenciales.
* El Administrador quiere registrar usuarios con nombre y email para darles acceso al sistema.
* El líder de proyecto quiere asignar roles a los usuarios para definir permisos y responsabilidades.
* El usuario quiere gestionar su perfil (nombre, email) para mantener actualizada su información personal.

### 2. Gestión de proyectos e iteraciones

* El líder de proyecto quiere crear proyectos para organizar el trabajo del equipo.
* El líder de proyecto quiere gestionar múltiples proyectos para coordinar distintos desarrollos en paralelo.
* El usuario quiere ingresar a múltiples proyectos para participar en distintos desarrollos.
* El líder de proyecto quiere asignar usuarios a un proyecto para formar el equipo de trabajo.
* El líder de proyecto quiere planificar etapas para estructurar el desarrollo en fases.
* El líder de proyecto quiere planificar iteraciones por etapa para organizar entregas parciales de trabajo.

### 3. Gestión de tareas

* El líder de proyecto quiere crear tareas en una iteración para planificar el trabajo del equipo.
* El miembro quiere proponer tareas fuera de planificación para registrar actividades imprevistas, quedando pendientes de aprobación del líder.
* El líder de proyecto quiere aprobar o rechazar tareas propuestas para mantener el control sobre la planificación.
* El líder o miembro quiere estimar tiempos para tareas para planificar el esfuerzo necesario.
* El líder de proyecto quiere asignar dependencias entre tareas para respetar el orden lógico de ejecución.
* El sistema quiere restringir el inicio de tareas dependientes para evitar que se inicien sin cumplir condiciones previas.
* El líder o miembro quiere modificar el estado de una tarea para indicar su avance (pendiente, en curso, finalizada).
* El líder o miembro quiere visualizar el estado de las tareas para hacer seguimiento del progreso.
* El líder o miembro quiere registrar horas en una tarea ya asignada a otro miembro para contribuir a su finalización, de modo que el sistema registre el tiempo aportado por cada participante.
* El miembro o líder quiere comentar una tarea para compartir notas, aclaraciones o incidencias.
* El líder de proyecto quiere categorizar tareas fuera de planificación para registrar actividades no previstas.

### 4. Gestión de tiempos

* El líder de proyecto quiere añadir, modificar y eliminar categorías de tiempo para organizar el registro de horas por tipo de actividad.
* El miembro quiere registrar tiempos reales de ejecución para reflejar cuánto se trabajó en una tarea.
* El miembro quiere registrar tiempos tomados externamente para documentar horas trabajadas fuera del sistema.
* El miembro quiere asignar el tiempo real aplicado a una tarea para contrastar el esfuerzo planificado vs. real.

### 5. Seguimiento y métricas

* El líder o docente quiere visualizar métricas globales del proyecto para evaluar avances y desempeño del equipo.
* El líder o miembro quiere visualizar el avance por iteración y etapa para controlar el cumplimiento de los objetivos parciales.
* El líder de proyecto quiere exportar reportes en PDF o Excel para compartir y analizar la información registrada.

**Planificación Estimada**

Se realizó una estimación del tiempo necesario para el desarrollo del sistema Kairos utilizando la metodología de Puntos de Caso de Uso (PCU) y considerando los casos de uso definidos previamente. Dado que a la fecha actual la definición detallada de todas las tareas aún no está completamente cerrada, esta estimación constituye una primera aproximación para orientar la planificación y permitirá ajustarse a medida que avance el desarrollo.

* Fin de la etapa de inicio: Este hito se da una vez definidas las bases para el proyecto de desarrollo de software.
* Fin de la etapa de elaboración – Iteración 1: Una vez refinados los detalles del proyecto para asegurar que se cumplen los requisitos y expectativas del cliente a medida que se avanza hacia las próximas iteraciones.
* Fin de la etapa de elaboración – Iteración 2: Definidos todos los aspectos necesarios para avanzar a la etapa de construcción.
* Fin de la etapa de construcción – Iteración 1: Finalizada la implementación de los casos de uso de mayor prioridad, y se realizan las pruebas correspondientes.
* Fin de la etapa de construcción – Iteración 2: Finalizada la segunda tanda de casos de uso y pruebas, y correcciones que hayan sido necesarias.
* Fin de la etapa de construcción – Iteración 3: Terminada la última tanda de casos de uso y pruebas.

**Propuesta Económica**

El desarrollo del sistema Kairos – Planificación y Registro de Tiempos se estima con un costo total de $12,5 millones ARS (aprox. USD 9.600), calculado en base al esfuerzo obtenido mediante la estimación por Puntos de Caso de Uso (PCU)

**Forma de pago:**

El pago podrá realizarse en tres etapas, vinculadas a hitos de avance:

* **30%** al finalizar la etapa de Elaboración.
* **40%** al finalizar la etapa de Construcción – Iteración 2.
* **30%** al entregar la versión final con documentación.

**Entregas del producto:**

* La entrega del Primer Prototipo Funcional tiene una fecha estimada el 05/10/25.
* La entrega final tiene como fecha estimada el 21/11/25.

**Soporte post-entrega:**

El grupo de desarrollo NexTech se compromete a brindar un período de soporte de 30 días posteriores a la entrega final, destinado a:

* Corrección de errores no detectados durante las pruebas.
* Asistencia en la instalación y despliegue del sistema.
* Capacitación básica para el uso de la aplicación (manual de usuario y guía rápida).

**Anexos**

Se anexa con el documento un resumen de la entrevista que fue realizada el día 26/08/2025.