

Ambiente de AutoEvaluación 2.0 (AAEV 2.0)



Plan de Gestión de Proyecto

Versión [1.0]

Contenido

1. INTRODUCCIÓN	3
1.1. ALCANCE DEL PROYECTO	3
1.2. ENTREGABLES DEL PROYECTO	4
1.3. ESTRATEGIA DE EVOLUCIÓN DEL PLAN	9
2. ORGANIZACIÓN DEL PROYECTO	9
2.1. MODELO DE PROCESO	9
2.2. RESPONSABLES	9
3. PROCESO DE GESTIÓN	13
3.1. OBJETIVOS Y PRIORIDADES DE GESTIÓN	13
3.2. CONDICIONES ASUMIDAS, DEPENDENCIAS Y RESTRICCIONES	14
3.3. GESTIÓN DE RIESGOS	14
3.4. MECANISMOS DE CONTROL Y AJUSTE	14
3.4.1. <i>Mecanismos para la Gestión de calidad</i>	14
3.4.2. <i>Mecanismos para la Gestión de configuración</i>	15
3.4.3. <i>Mecanismos para Verificación</i>	15
3.4.4. <i>Mecanismos para la Gestión de proyecto</i>	16
3.5. RECURSOS	16
4. PROCESO TÉCNICO	17
4.1. PROCEDIMIENTOS TÉCNICOS, HERRAMIENTAS Y TECNOLOGÍAS	17
4.2. DOCUMENTACIÓN DE SOFTWARE	17

1 Introducción

Este documento es el plan de Gestión del proyecto Ambiente de AutoEvaluación 2.0 (AAEV 2.0), propuesto por los profesores de la cátedra Proyecto de Software de la carrera Licenciatura en Sistemas de la Universidad Nacional de Lanús, con el fin de crear un espacio web donde docentes crearán exámenes para que los alumnos prueben sus conocimientos sobre las materias de sus respectivas carreras.

El proyecto sigue el ciclo de vida en prototipado evolutivo, teniendo el diseño de una maqueta primero de 1 a 3 iteraciones y luego el prototipo donde el usuario refinará los requisitos del sistema.

1.1 Alcance del proyecto

El objetivo del proyecto es crear un ambiente web de ámbito universitario donde el docente de una o varias materias de una carrera elaborará exámenes a partir del contenido de las mismas para ser luego realizados por los alumnos que se inscriban en ellas mediante el ambiente para así poder lograr una observación de su progreso en la adquisición y entendimiento de conceptos de la materia.

El producto entregado, dependiendo del privilegio asignado, permitirá a los usuarios:

- Gestionar las carreras de la universidad: agregar, modificar, o eliminar una carrera. Acción realizada por el administrador del ambiente.
- Gestionar las materias de cada carrera: agregar, modificar o eliminar una materia. Esta acción es realizada por el administrador del ambiente.
- Gestionar las cuentas del cuerpo docente de cada carrera: agregar, modificar o eliminar un docente. Esta acción es realizada por el administrador del ambiente.
- Gestionar los exámenes: agregar exámenes, modificarlos o eliminarlos. Dentro del proceso de gestión de exámenes encontramos la gestión de opciones, donde dependiendo de la pregunta elegida el docente realizará el alta, baja o modificación de las mismas. Por último, el docente asigna un sistema de puntuación al examen agregado.
- Aceptar solicitudes de alumnos para unirse a la materia: los docentes eligen de la lista de solicitudes que alumnos podrán acceder a determinada materia y cuáles no.
- Registrarse en el sistema: el alumno ingresará sus datos en un formulario y se registrará en el sistema con su usuario y clave
- Inscribirse a materia: el alumno elige una de las materias de la carrera y envía una solicitud de inscripción al/los docente/s de la misma, es cuestión del docente aceptarlo en ella o no
- Realizar examen: el alumno, una vez aceptado en la materia, ingresa al examen y contesta las preguntas definidas previamente por el docente, para al final obtener una nota en base a las respuestas correctas e incorrectas del alumno

Para más detalles de las acciones de los usuarios, ver el documento de requisitos del proyecto.

Los privilegios del sistema serán:

- Alto: Administrador
- Medio: Docente
- Bajo: Alumno

1.2 Entregables del proyecto

Identificación del entregable	Descripción del entregable	Fecha de entrega	Lugar de entrega	Condiciones de satisfacción
Plan de Gestión de Proyecto	Es el presente documento	25/8/16	Campus Virtual de la UNLA: Curso Proyecto de Software	Aprobación del cliente. Aprobación del docente
Mapa de actividades	Mapa donde se indica las actividades de los procesos y en qué etapa de desarrollo se realizan, guía al equipo de desarrollo	25/8/16	Campus Virtual de la UNLA: Curso Proyecto de Software	Aprobación del cliente. Aprobación del docente
Plan de Retiro	Documento donde se planifica el retiro del sistema desde el ambiente donde está instalado	25/8/16	Campus Virtual de la UNLA: Curso Proyecto de Software	Aprobación del cliente. Aprobación del docente
Análisis de Riesgos	Análisis de los riesgos potenciales del proyecto, su probabilidad de ocurrencia e impacto en el mismo	25/8/16	Campus Virtual de la UNLA: Curso Proyecto de Software	Aprobación del cliente. Aprobación del docente
Plan de Contingencias	Documento donde se planifican las medidas a tomar en caso de que uno de los riesgos analizados se materialice en el desarrollo del proyecto	25/8/16	Campus Virtual de la UNLA: Curso Proyecto de Software	Aprobación del cliente. Aprobación del docente
Plan de Garantía de Calidad del Software	Documento donde se indican los factores que	Sin especificar	Campus Virtual de la UNLA: Curso Proyecto	Aprobación del cliente. Aprobación del

	aseguran la calidad del software para satisfacer necesidades dadas		de Software	docente
Estudios de viabilidad	Estudio de requisitos del cliente para analizar y determinar si el proyecto es viable o no	25/8/16	Campus Virtual de la UNLA: Curso Proyecto de Software	Aprobación del cliente. Aprobación del docente
Plan de transición del proyecto	Documento donde se planifica la transición del proyecto del ambiente de desarrollo al ambiente de producción	25/8/16	Campus Virtual de la UNLA: Curso Proyecto de Software	Aprobación del cliente. Aprobación del docente
Modelo de la situación actual	Modelo donde se indica la situación actual del entorno del proyecto (si funciona con otros sistemas, y el funcionamiento de estos)	25/8/16	Campus Virtual de la UNLA: Curso Proyecto de Software	Aprobación del cliente. Aprobación del docente
Modelo del dominio del problema	Modelo conceptual de todos los temas relacionados con un problema específico. En él se describen las distintas entidades, sus atributos, papeles y relaciones, además de las restricciones que rigen el dominio del problema.	25/8/16	Campus Virtual de la UNLA: Curso Proyecto de Software	Aprobación del cliente. Aprobación del docente
Informe preliminar de necesidades	Informe de las necesidades del cliente para el	25/8/16	Campus Virtual de la UNLA: Curso Proyecto	Aprobación del cliente. Aprobación del

	proyecto		de Software	docente
Soluciones alternativas posibles	Documento donde se tienen en cuenta las alternativas posibles del desarrollo del software	25/8/16	Campus Virtual de la UNLA: Curso Proyecto de Software	Aprobación del cliente. Aprobación del docente
Soluciones recomendadas	Documento donde se plasman las soluciones recomendadas de acuerdo a las necesidades del cliente.	25/8/16	Campus Virtual de la UNLA: Curso Proyecto de Software	Aprobación del cliente. Aprobación del docente
Documento de requisitos del software	Documento donde se detallan los requisitos del cliente	No especificada.	Campus Virtual de la UNLA: Curso Proyecto de Software	Aprobación del cliente. Aprobación del docente
-Descripción del diseño de software	Auto explicativo	No especificada.	Campus Virtual de la UNLA: Curso Proyecto de Software	Aprobación del cliente. Aprobación del docente
Descripción de la arquitectura del software	Auto explicativo	No especificada.	Campus Virtual de la UNLA: Curso Proyecto de Software	Aprobación del cliente. Aprobación del docente
Descripción del flujo de información	Auto explicativo	No especificada.	Campus Virtual de la UNLA: Curso Proyecto de Software	Aprobación del cliente. Aprobación del docente
Descripción de la base de datos	Auto explicativo	No especificada.	Campus Virtual de la UNLA: Curso Proyecto de Software	Aprobación del cliente. Aprobación del docente
Descripción de las interfaces	Auto explicativo	No especificada.	Campus Virtual de la UNLA: Curso Proyecto de Software	Aprobación del cliente. Aprobación del docente
-Datos para las pruebas	Datos necesarios para desarrollar los casos de prueba	No especificada.	Campus Virtual de la UNLA: Curso Proyecto de Software	Aprobación del cliente. Aprobación del docente
Documentación de usuario	Manual de usuario	No especificada.	Campus Virtual de la UNLA: Curso Proyecto de Software	Aprobación del cliente. Aprobación del docente
Plan de integración	Plan de integración de	No especificada.	Campus Virtual de la UNLA:	Aprobación del cliente.

	los módulos del sistema		Curso Proyecto de Software	Aprobación del docente
Sistema software integrado	“Materialización” del software diseñado a través de lenguajes de programación	No especificada.	Campus Virtual de la UNLA: Curso Proyecto de Software	Aprobación del cliente. Aprobación del docente
Plan de instalación de software	Documento que brinda pasos al usuario de cómo instalar el producto software (también llamado manual de instalación)	No especificada.	Campus Virtual de la UNLA: Curso Proyecto de Software	Aprobación del cliente. Aprobación del docente
Plan de verificación y validación	Plan cuya función es Identificar los componentes de software y documentos que deben ser verificados y describir las estrategias de verificación que serán usadas.	No especificada.	Campus Virtual de la UNLA: Curso Proyecto de Software	Aprobación del cliente. Aprobación del docente
Plan de Pruebas	Permite tener una planeación de la aplicación de las pruebas y el tipo de pruebas que harán que el sistema funcione correctamente	No especificada.	Campus Virtual de la UNLA: Curso Proyecto de Software	Aprobación del cliente. Aprobación del docente
Especificación de las pruebas	Especificación de cada caso de prueba para verificar el software	No especificada.	Campus Virtual de la UNLA: Curso Proyecto de Software	Aprobación del cliente. Aprobación del docente
Informe resumen de las pruebas	Resumen de los resultados obtenidos en las pruebas de software	No especificada.	Campus Virtual de la UNLA: Curso Proyecto de Software	Aprobación del cliente. Aprobación del docente
Plan de Gestión de	Plan para el control de la	No especificada.	Campus Virtual de la UNLA:	Aprobación del cliente.

Configuración	configuración de los documentos del proyecto		Curso Proyecto de Software	Aprobación del docente
Informe de estado	Informa acerca del estado de los componentes de un producto y de las solicitudes de cambio	No especificada.	Campus Virtual de la UNLA: Curso Proyecto de Software	Aprobación del cliente. Aprobación del docente
Plan de documentación	Establece la estrategia de documentación durante todo el proyecto	No especificada.	Campus Virtual de la UNLA: Curso Proyecto de Software	Aprobación del cliente. Aprobación del docente
Casos de Uso	Documento que muestra las funciones del punto de vista del negocio y los usuarios que las realizan	No especificada.	Campus Virtual de la UNLA: Curso Proyecto de Software	Aprobación del cliente. Aprobación del docente
Especificación de Casos de Uso	Brinda información detallada de cada caso de uso así como los pasos contenidos y la información necesaria para cada uno	No especificada.	Campus Virtual de la UNLA: Curso Proyecto de Software	Aprobación del cliente. Aprobación del docente
Diagrama Entidad-Relación	Diagrama que establece la relación entre los datos del sistema y como se distribuyen los mismos	No especificada.	Campus Virtual de la UNLA: Curso Proyecto de Software	Aprobación del cliente. Aprobación del docente
Diagrama de clases	Diagrama que muestra el conjunto de objetos del sistema, las operaciones que puede realizar cada una de ellas y las dependencias y asociaciones entre los mismos	No especificada.	Campus Virtual de la UNLA: Curso Proyecto de Software	Aprobación del cliente. Aprobación del docente

1.3 Estrategia de evolución del Plan

Por decisión del equipo, Alejandro Canosa es responsable del monitoreo del presente documento, con una frecuencia media prevista de cambios al mismo, en caso de que se presente un posible cambio, el equipo se reunirá y debatirá si el cambio del plan es necesario o no, en caso de que lo sea, los cambios en el documento serán efectuados, se enviará una copia a cada integrante del equipo y se registrará el cambio en la tabla de versiones de documentos, junto con la versión, la fecha, el autor y la descripción del mismo.

Al finalizar el cambio y su registro, el autor del mismo deberá informar por cualquier medio de comunicación la finalización del proceso y esta versión nueva se les deberá ser enviada y subida al repositorio donde se guardan las versiones actuales.

El plan será revisado al inicio de cada fase, modificado de acuerdo a lo necesario, aprobado y distribuido al equipo de proyecto.

2 Organización del proyecto

Esta sección contiene la especificación del modelo de proceso del Proyecto, descripción de la estructura organizacional del proyecto, identificación de interfaces e interacciones y definición de responsables.

2.1 Modelo de proceso

Esta sección contiene la definición de las relaciones entre las actividades más relevantes del proyecto. Ver Diagrama de GANTT del proyecto.

2.2 Responsables

Se identifican las actividades más relevantes en el proyecto, los responsables de dichas actividades y los involucrados.

Identificación de actividad	Descripción de actividad	Responsable/s	Involucrados
Establecer matriz de actividades	Elaboración de mapa de actividades del proyecto	Nazareno Galvan, Agustín Di Stefano, Alejandro Canosa	Alejandro Canosa, Nazareno Galvan, Agustín Di Stefano
Asignar recursos	Se asignan los recursos necesarios para llevar a cabo el proyecto	Alejandro Canosa	Alejandro Canosa, Nazareno Galvan, Agustín Di Stefano
Definir el entorno del proyecto	Definir Hardware y Software necesario para el correcto	Profesores de cátedra	Alejandro Canosa, Nazareno Galvan, Agustín Di Stefano

	desarrollo del proyecto		
Planificar la gestión del proyecto	Planificación del control del proyecto	Alejandro Canosa	Alejandro Canosa, Nazareno Galvan, Agustín Di Stefano
Analizar riesgos.	Análisis de riesgos potenciales que podrían aparecer durante el desarrollo del proyecto	Nazareno Galvan, Agustín Di Stefano	Alejandro Canosa, Nazareno Galvan, Agustín Di Stefano
Realizar la planificación de contingencias	Generar un plan que contenga las contramedidas en caso de que un riesgo se materialice	Nazareno Galvan, Agustín Di Stefano	Alejandro Canosa, Nazareno Galvan, Agustín Di Stefano
Gestionar el proyecto	Mediante el plan de gestión, controlar el desarrollo del proyecto	Alejandro Canosa, Nazareno Galvan, Agustín Di Stefano	Alejandro Canosa, Nazareno Galvan, Agustín Di Stefano
Identificar las ideas o necesidades	Identificación de las ideas o necesidades del producto a desarrollar y su documentación	Alejandro Canosa	Alejandro Canosa, Nazareno Galvan, Agustín Di Stefano
Formular las soluciones potenciales	Formulación de las posibles soluciones al proyecto en base a la identificación de ideas o necesidades	Alejandro Canosa	Alejandro Canosa, Nazareno Galvan, Agustín Di Stefano
Dirigir los estudios de viabilidad	Analizar los requisitos del proyecto y determinar si este es viable o no	Alejandro Canosa	Alejandro Canosa, Nazareno Galvan, Agustín Di Stefano
Planificar la transición del sistema	Generar un documento que contenga el plan de instalación	Alejandro Canosa, Nazareno Galvan, Agustín Di Stefano	Alejandro Canosa, Nazareno Galvan, Agustín Di Stefano
Refinar y finalizar la idea o necesidad	Modificar ideas o necesidades en base a las soluciones encontradas	Alejandro Canosa, Nazareno Galvan, Agustín Di Stefano	Alejandro Canosa, Nazareno Galvan, Agustín Di Stefano
Definir y Desarrollar los Requisitos de	Establecer requisitos funcionales, no	Alejandro Canosa, Nazareno Galvan, Agustín Di	Alejandro Canosa, Nazareno Galvan, Agustín Di

software	funcionales, de interfaz, etc. Del proyecto y documentarlos	Stefano	Stefano
Definir los requisitos de interfaz	Definir requisitos de la interfaz de usuario del proyecto	Alejandro Canosa, Nazareno Galvan, Agustín Di Stefano	Alejandro Canosa, Nazareno Galvan, Agustín Di Stefano
Priorizar e integrar los requisitos del software	Establecer prioridades a cada uno de los requisitos del software a construir y documentarlos en un Documento de Requisitos	Alejandro Canosa	Alejandro Canosa, Nazareno Galvan, Agustín Di Stefano
Realizar el diseño preliminar	Se planifica el proyecto software a través de documentos de diseño según el paradigma a aplicar, en este, se usaran las herramientas UML para el Paradigma Orientado a Objetos	Alejandro Canosa, Nazareno Galvan, Agustín Di Stefano	Alejandro Canosa, Nazareno Galvan, Agustín Di Stefano
Analizar el flujo de información	Análisis del camino que realizara la información a través del sistema	Alejandro Canosa, Nazareno Galvan, Agustín Di Stefano	Alejandro Canosa, Nazareno Galvan, Agustín Di Stefano
Diseñar la base de datos (si se aplica).	Diseño de las tablas de la Base de datos en base a las relaciones y atributos necesarios para satisfacer los requisitos	Alejandro Canosa	Alejandro Canosa, Nazareno Galvan, Agustín Di Stefano
Diseñar las interfaces	Diseño de interfaces de usuario	Agustín Di Stefano	Alejandro Canosa, Nazareno Galvan, Agustín Di Stefano
Realizar el diseño detallado	Realización de implementaciones de código del producto (clases de implementación) según los datos obtenidos en el diseño preliminar		Alejandro Canosa, Nazareno Galvan, Agustín Di Stefano

	y el documento de requisitos		
Crear los datos de prueba	Se crean los datos que diferencien los casos de prueba	Alejandro Canosa, Nazareno Galvan, Agustín Di Stefano	Alejandro Canosa, Nazareno Galvan, Agustín Di Stefano
Crear la documentación de operación	Se realiza el manual de usuario para el uso del producto software	Alejandro Canosa, Nazareno Galvan, Agustín Di Stefano	Alejandro Canosa, Nazareno Galvan, Agustín Di Stefano
Planificar la verificación y validación	Realizar el plan de verificación y validación con el fin de revisar si el producto software es correcto y cumple los requisitos establecidos por el cliente	Alejandro Canosa, Nazareno Galvan, Agustín Di Stefano	Alejandro Canosa, Nazareno Galvan, Agustín Di Stefano
Ejecutar tareas, recoger y analizar los datos	Ejecución de tareas necesarias y recopilación de datos de las mismas para la posterior planificación de prueba	Alejandro Canosa, Nazareno Galvan, Agustín Di Stefano	Alejandro Canosa, Nazareno Galvan, Agustín Di Stefano
Planificar las pruebas	Diseñar el documento denominado Plan de Pruebas en el que se establecerán los casos de prueba de los módulos designados	Alejandro Canosa, Nazareno Galvan, Agustín Di Stefano	Alejandro Canosa, Nazareno Galvan, Agustín Di Stefano
Ejecutar las Pruebas	Ejecución de las pruebas diseñadas para verificar funcionamiento del software	Alejandro Canosa, Nazareno Galvan, Agustín Di Stefano	Alejandro Canosa, Nazareno Galvan, Agustín Di Stefano
Planificar la gestión de la configuración	Realizar el Plan de Gestión de Configuración del Software.	Alejandro Canosa	Alejandro Canosa, Nazareno Galvan, Agustín Di Stefano
Realizar la identificación de la configuración	Identificar los elementos del sistema y registrarlos como Elementos de Configuración de	Alejandro Canosa	Alejandro Canosa, Nazareno Galvan, Agustín Di Stefano

	Software		
Realizar el control de la configuración	Hasta el retiro del producto software, realizar la Gestión de Configuración planificada en el Plan de Gestión de Configuración	Alejandro Canosa, Nazareno Galvan, Agustín Di Stefano	Alejandro Canosa, Nazareno Galvan, Agustín Di Stefano
Realizar la información del estado de la configuración	Documento que indique versiones, solicitudes de cambio, etc.	Alejandro Canosa	Alejandro Canosa, Nazareno Galvan, Agustín Di Stefano

3 Proceso de Gestión

3.1 Objetivos y Prioridades de Gestión

El objetivo de la Gestión es lograr controlar los procesos software que se llevarán a cabo en la realización del proyecto, así como la gestión de calidad, la verificación y validación y los cambios del mismo.

La gestión comprende 4 actividades

- Iniciación
- Planificación
- Ejecución
- Control

El fin de la iniciación es poder definir la propuesta del proyecto, sus objetivos, alcance, calidad y estimar los riesgos del mismo, es una actividad de alta prioridad ya que si este proceso está definido de manera incorrecta llevara a un desarrollo exhaustivo del proyecto, básicamente. En esta actividad se realizarán consultas al cliente por medio de entrevistas sobre los objetivos del proyecto a desarrollar.

La planificación conllevara la realización del diagrama de GANTT y matriz de actividades, la elaboración del plan de Gestión, el análisis de riesgos, el plan de Contingencias y el diseño y codificación del proyecto. La codificación y las pruebas forman parte de esta actividad o etapa de gestión.

La ejecución refiere a la implementación o puesta en marcha del proyecto, consiste en poner en práctica la planificación llevada a cabo previamente. Se realizará mediante el manual de instalación producido durante el proyecto. En esta actividad se llevará a cabo el plan de transición del proyecto, se presentaran los manuales de instalación y de usuario, se revisaran los prototipos y el cliente aprobara o especificará los cambios necesarios para la instalación del proyecto en el entorno.

El fin de las actividades de control es asegurar que los objetivos sean alcanzados en el tiempo y calidad planificada, realizando una buena supervisión y medición del rendimiento de los resultados, con el objetivo de que se puedan tomar acciones correctivas, esto se hace mediante la comparación entre la planificación realizada y los valores incurridos. En esta actividad se llevara a cabo la Gestión de calidad, de configuración y el proceso de verificación y validación.

3.2 Condiciones asumidas, dependencias y restricciones

En este momento el proyecto no contiene restricciones.

Dependencias: aceptación del cliente.

Condiciones asumidas: 4 hitos de entrega con fechas predefinidas:

- Hito 1. Análisis y diseño inicial.
- Hito 2. Prototipo 1. Maqueta.
- Hito 3. Prototipo 2. Prototipo Funcional
- Hito 4. Presentación Global del Proyecto.

Realización de presentación Técnica que describan las decisiones realizadas durante los hitos.

Cada hito y documentación final deberá ser entregado en el formato de la revista Latinoamericana de Ingeniería de Software.

3.3 Gestión de riesgos

Los riesgos se ven en el Documento de Análisis de Riesgos y sus contramedidas en el Plan de Contingencias

3.4 Mecanismos de control y ajuste

Consultas al cliente: via mail, Skype o presencial.

Reportes presenciales al cliente o por medio del aula virtual del campus de la UNLa.

Auditorias: presenciales, teniendo la configuración de los documentos revisados de acuerdo al plan de Gestión de Configuración

3.4.1 Mecanismos para la Gestión de calidad

Las actividades más importantes de la Gestión de Calidad de este proyecto son:

- Pruebas de Software: reunión de información, creación de casos de prueba, registro de los mismos, ejecución de las pruebas y registro de los resultados

- **Gestión de Configuración:** definición de ECS (Elementos de Configuración de Software), registro de ECS, tabla de versiones, control de cambios, registro de cambios.

El monitoreo y control de las pruebas será realizado por el equipo. En caso de posible modificación se notificaran al resto del equipo y se evaluara en conjunto para afirmar la necesidad de actualizar los documentos o descartar el posible cambio. Cada Prueba estará documentada, así como cada caso de prueba y la planificación para llevar a cabo las pruebas de software. Se verificarán los resultados obtenidos en las mismas

La Gestión de Configuración será monitoreada por Nazareno Galvan y Alejandro Canosa, en caso de posible notificación se notificara al resto del equipo y se evaluara en conjunto para afirmar la necesidad de actualizar los documentos o descartar el posible cambio. Cada modificación será registrada en la tabla de versiones indicando la versión nueva, el documento cambiado, y el autor del cambio. Antes de actualizar la tabla de versiones, se notificará al resto del equipo.

3.4.2 Mecanismos para la Gestión de configuración

Las actividades más importantes de la gestión de configuración son:

- Planificar la gestión de la configuración: realización del Plan de Gestión de Configuración.
- Realizar la identificación de la configuración: identificación de los ECS del proyecto
- Realizar el control de la configuración: registrar cambios en una tabla de versiones indicando los datos mencionados previamente

El monitoreo de la Gestión de configuración se centra en asegurarse del registro del cambio de versión en la tabla de versionado junto con la descripción adecuada, además de agregar nuevos ECS conforme el proyecto vaya avanzando y creciendo, y en las revisiones del equipo al documento que se solicita cambiar

3.4.3 Mecanismos para Verificación y Validación

Las actividades más importantes del proceso de Verificación son:

- Planificar la Verificación y Validación: Elaboración del Plan de Verificación y Validación
- Pruebas de Software: conjunto de casos de prueba que indican si un modulo o conjunto de módulos funcionan correctamente o no.

El monitoreo se centra en la elaboración y seguimiento del Plan de Verificación y Validación y los requisitos provistos por el cliente. En caso de cambios el equipo realiza el procedimiento de configuración descrito en los mecanismos anteriores

3.4.4 Mecanismos para la Gestión de proyecto

Las actividades más importantes de la Gestión de Proyecto son:

- Analizar riesgos
- Realizar la planificación de contingencias

La gestión se monitorea con el presente documento, teniendo en cuenta la revisión del mismo al inicio de cada fase, y las posibles modificaciones que este pueda tener.

3.5 Recursos

El equipo de proyecto está compuesto por Alejandro Canosa, Agustín Di Stefano y Nazareno Galvan.

Recursos tecnológicos de los miembros

- Entornos de desarrollo
- Repositorio Git para control de versiones, control de documentación y gestión. Se usa la pagina web GitHub y la versión de escritorio teniendo cada uno un repositorio local en su pc.

Experiencia:

- Codificación (lógica):
 - Alejandro Canosa: Media/alta
 - Nazareno Galvan: Media/alta
 - Agustín Di Stefano: Media/alta
- Codificación (interfaz de usuario):
 - Alejandro Canosa: Media
 - Nazareno Galvan: Baja
 - Agustín Di Stefano: Alta
- Ingeniería de software:
 - Alejandro Canosa: Media
 - Nazareno Galvan: Media
 - Agustín Di Stefano: Media

4 Proceso técnico

4.1 Procedimientos técnicos, herramientas y tecnologías

Requerimientos técnicos:

- Sistema Operativo: Debían Jessie
- Servidor Web: Apache2
- Gestor de Base de datos: MySQL v5.5
- Lenguajes: Python v2.7
- Framework: Django Instalado 1.7.7

Herramientas de diseño:

- Software Microsoft Visio 2007

Estándar especificado: IEEE 1074/97

4.2 Documentación de software

La documentación se realizará de acuerdo a las plantillas de documentos definidas en el Modelo de Proceso para desarrollo de Software del estándar IEEE 1074/97

El formato de las mismas será según la Revista Latinoamericana de Ingeniería de Software