

PLANIFICACIÓN Y GESTIÓN DE PROYECTOS INFORMÁTICOS

PRÁCTICA 1

MANUAL DE COORDINACIÓN

Lidia Sánchez Mérida

Fernando Roldán Zafra

Cristina de la Caridad Vaillant Valdéz

CICLO DE VIDA

Tras estudiar y analizar los diversos modelos que existen para especificar el ciclo de vida del *software* hemos decidido representar el de nuestro proyecto mediante un **modelo en espiral**. La posibilidad de dividir el desarrollo del sistema en varias etapas con el fin de planificar cada una así como analizar sus requisitos, riesgos y escoger la mejor alternativa para implementarla, convierte a esta metodología en una buena opción.

METODOLOGÍA DE DESARROLLO

Debido a que el ciclo de vida elegido cuenta con características comunes a las metodologías ágiles de desarrollo hemos optado por aplicar la **eXtreme Programming o Programación Extrema**. La razón de ello se fundamenta en el conjunto de valores que reúne tales como la simplicidad del código con el fin de que cualquier miembro se pueda incorporar al trabajo realizado de otro así como llevar a cabo un mantenimiento sencillo del sistema, el alto grado de comunicación entre los desarrolladores y los clientes, la retroalimentación de estos mediante su integración en el equipo de desarrollo y la valentía necesaria para reestructurar todo el código o ser capaz de desechar aquel que no es útil.

RECURSOS SOFTWARE

El lenguaje de programación que se va a utilizar es **Python** debido a su gran número de ventajas que presenta como su bajo tiempo de ejecución realizando cálculos computacionalmente costosos, su modularidad y su gran conjunto de herramientas y librerías de las que dispone. En particular utilizaremos un *framework* **Pytorch** destinado al desarrollo de redes neuronales mediante *Deep Learning* así como su ejecución en paralelo aprovechando la GPU al máximo.

Para almacenar los datos utilizaremos varias bases de datos NoSQL mediante **MongoDB** puesto que proporciona una gran escalabilidad así como la ventaja de almacenar datos sin una estructura definida. Además también existe en Python una librería denominada **Pymongo** que facilita en gran medida la conexión del sistema con las bases de datos.

ORGANIZACIÓN DEL EQUIPO DE TRABAJO

Será necesario conformar un primer equipo de cinco ingenieros especializados en *Deep Learning* para entrenar y evaluar la capacidad de predicción de la red neuronal. Un segundo equipo de tres programadores para desarrollar el sistema. Asimismo se contratará a un gestor de bases de datos que se encargue del almacenamiento de los

diagnósticos realizados por la red neuronal y un gestor de configuración para que se ocupe del servidor de bases de datos y de la instalación del sistema completamente desarrollado en los centros médicos.

HERRAMIENTAS PARA LA COMUNICACIÓN DEL EQUIPO

En primer lugar se realizarán reuniones a primera hora de todos los equipos para exponer sus progresos y dificultades. No obstante también se utilizarán herramientas tales como **Slack** para crear tantos grupos como equipos de trabajo con el objetivo de que todos los miembros discutan los aspectos comunes así como la comunicación individual con cualquier miembro de otro equipo. Por último se crearán repositorios para la implementación del sistema y se configurará un servidor de bases de datos que permita el acceso a los diagnósticos realizados por la red neuronal.

RELACIONES CON EL CLIENTE

Se integrarán un conjunto de especialistas en enfermedades cardiovasculares en el equipo de desarrollo además de dos nuevos grupos en Slack con el objetivo de que proporcionen una realimentación tanto en los diagnósticos que realice la red neuronal como en el diseño de la interfaz del sistema. Además se realizarán reuniones periódicas para presentarles los prototipos asociados a cada una de las etapas del ciclo de vida.

ESTÁNDARES DE DOCUMENTACIÓN

Basándonos en el estándar de documentación **IEEE 1063-2001** se redactará un documento que encapsule el ciclo de vida del software, un manual para los desarrolladores con información acerca de las operaciones del software y un manual para el cliente con el fin de ayudarle a utilizar el sistema.

ESTÁNDARES DE CÓDIGO

Debido al uso de Python como lenguaje de programación para desarrollar el sistema utilizaremos el estándar de código **PEP 8** que recopila una guía de estilo específico para este lenguaje de programación. Algunos de los aspectos que trata son el tamaño máximo de línea, los saltos de línea, entre otros. Asimismo se definirán las variables con nombres representativos, sencillos y en inglés puesto que es un lenguaje internacional.

CONTROL DE VERSIONES

Para mantener un control de versiones se utilizará la plataforma **GitHub** en la que cada equipo de desarrollo dispondrá de una rama particular. Solo el jefe de cada equipo podrá incorporar los cambios que los demás miembros de su equipo suban a la plataforma. Asimismo será también el responsable de resolver los posibles conflictos que surjan.

GESTIÓN DE CALIDAD

Se realizarán varias reuniones y pruebas del sistema de forma recursiva. De este modo se podrán detectar a tiempo los posibles errores antes de continuar con la siguiente fase. Asimismo estas reuniones también servirán para conocer el estado del proyecto así como su ajuste a las fechas de entrega establecidas.

