**Introducción**

En este documento se redactan los detalles destacables respecto al proyecto de Ingeniería de Software del grupo 1-1, grado de Inteligencia Artificial.

Integrantes del grupo:

* Antonio Serrano Rodríguez
* Guillermo García Engelmo
* Iago Grandal del Río
* Hugo Bouzón Fernández
* Marcos Güeto Sandá
* Óscar Vilela Rodríguez

El documento se rige por la siguiente estructura:

[1) Explicación de puesta en práctica de Scrum](#_Explicación_de_puesta)

[1.1) Descripción de los roles de cada miembro del equipo](#_Descripción_de_los)

[1.2) Explicación de cómo se ha puesto en práctica Scrum](#_Explicación_de_cómo)

[1.2.1) Calendario de eventos](#_Calendario_de_eventos)

[1.2.2) Problemas encontrados y soluciones halladas](#_Problemas_encontrados_y)

[1.3) Listado de los artefactos de Scrum.](#_Listado_de_los)

[2) Descripción de la gestión del repositorio grupal](#_Descripción_de_la)

[2.1) Flujo y política de creación de ramas](#_Flujo_y_política)

[2.2) Política de nombrado de “commits”](#_Política_de_nombrado)

[3) Explicación del código desarrollado](#_Explicación_del_código)

[3.1)Patrones, aspectos de diseño, etcétera, relevantes](#_Los_patrones,_aspectos)

[3.2) Pruebas y planes de pruebas](#_Pruebas_y_planes)

[3.3) Normas de estilo](#_Normas_de_estilo)

[3.4) Descripción del proceso de despliegue](#_Descripción_del_proceso)

# **Explicación de puesta en práctica de Scrum**

# **Descripción de los roles de cada miembro del equipo**

Para este proyecto no fue necesaria una asignación compleja. Excluyendo al Product Owner (Alberto Alvarellos), los roles en Scrum fueron básicos: un Scrum Master y varios desarrolladores. El Scrum Master designado fue Antonio Serrano, mientras que todos los demás integrantes adoptaron el rol de Developers.

Las tareas del Scrum Master incluyen tomar las riendas de la conversación en el Sprin review, retrospectiva y planning, así como asignar y dirigir las reuniones Daily Scrum.

Adicionalmente, se designó como Git Master a Marcos Güeto, encargado de las tareas de supervisión y moderación del repositorio en GitHub.

# **Explicación de cómo se ha puesto en práctica Scrum**

En el caso particular de nuestro equipo, este proyecto supuso el primer contacto con Scrum, lo cual propició cierto componente de desorden en un inicio, con varios problemas.

# **Calendario de eventos**

Con sprints semanales, los eventos de Scrum se organizaron de manera sencilla. Para la mayoría de los eventos, los únicos por sprint (Sprint review, retrospectiva y planning), el día fijado fue el miércoles. Los Scrum diarios, sin embargo, tomaron lugar tanto los lunes como los viernes, para una mejor coordinación del trabajo durante los fines de semana.

# **Problemas encontrados y soluciones halladas**

La mayor parte de los problemas surgieron durante las fases iniciales del proyecto. Algunos de estos problemas fueron la creación de tareas inadecuadas, asignaciones a múltiples integrantes, distribuciones incorrectas, etc. Uno de los más destacables fue la sobrecarga de trabajo en diversos sprints, sucedido en numerosas ocasiones. Dichos inconvenientes fueron corregidos tras tiempo de puesta en práctica y adaptación, en el que el equipo de desarrollo logró obtener conocimiento acerca de esta metodología y experiencia en la misma. El componente clave para soluciones fue un incremento tanto en la cantidad como en la calidad de la comunicación, acompañado con un mayor dominio y conocimiento de las capacidades del equipo.

# **Listado de los artefactos de Scrum**

Como artefactos, este proyecto ha contado con varias instancias de cada clase. Entre estas, existen 23 historias de usuario del Product Backlog, repartidas en 8 sprints de duración variable. Todas estas historias han sido satisfechas. Sin embargo, en ciertos casos, las historias fueron forzadas a sprints siguientes debido a no ser terminadas en el tiempo establecido. Este problema surge de la inexperiencia del grupo a la hora de juzgar la carga de trabajo factible para cada sprint.

# **Descripción de la gestión del repositorio grupal**

Este proyecto fue el primer contacto del equipo con repositorios grupales, con lo que existen numerosas inconsistencias en este apartado.

La gestión del repositorio se realizó mediante GitHub.

# **Flujo y política de creación de ramas**

Para organizar el trabajo en el repositorio fue necesaria una gestión de diferentes ramas de trabajo. Esto conlleva la necesidad de diseñar una política de gestión de las mismas. En este proyecto existieron dos fases: una inicial en la que se creaban y eliminaban ramas muy ocasionalmente para encargarse de pequeñas tareas (ya que en esta etapa aún se estaban desarrollando módulos nuevos y, por tanto, había conflictos mínimos) y una segunda fase, en la que la creación de ramas era un evento común para cada tarea, y en la que se empleó una rama “develop”, por la cual el código pasaba antes de ingresar a la rama estable “main”.

# **Política de nombrado de “commits”**

El nombrado de “commits” no siguió una política estricta para este proyecto. La convención general consistió en dos puntos importantes: los mensajes deben estar escritos en inglés y deben ser breves y concisos, resumiendo en la menor cantidad de palabras posibles las funcionalidades que implementan.

# **Explicación del código desarrollado**

Para desarrollar el código de este proyecto, se utilizó el entorno de desarrollo Visual Studio Code, que permite interacción rápida y sencilla con repositorios de Git.

# **Patrones, aspectos de diseño, etcétera, relevantes**

Respecto al diseño del proyecto existe un detalle destacable. En contraste con lo solicitado, el proyecto tomó la realización de una aplicación web en lugar de una tradicional.

# **Pruebas y planes de pruebas**

Para realizar las pruebas de eficacia del software, se hizo uso de la librería Pytest, llevando a cabo un módulo “test.py” en el que ejecutar las comprobaciones.

# **Normas de estilo**

En el caso del código, se trató de seguir una política para construir un código coherente. Las medidas principales incluyen redactar comentarios y nombrar variables, funciones y módulos en inglés, utilizar snake\_case para variables, funciones y clases y mixedCase para módulos, y realizar las salidas por pantalla en castellano.

# **Descripción del proceso de despliegue**

El despliegue de este proyecto es sencillo, ya que, a pesar de su naturaleza como aplicación web, no se realizó su publicación en un servidor. Simplemente, se puso a disposición de los usuarios de GitHub, configurando como público el repositorio.