# 4. Técnicas y herramientas

Este apartado de la memoria tiene como objetivo presentar las técnicas metodológicas y las herramientas de desarrollo que se han utilizado para llevar a cabo el proyecto.

## 4.1 Metodología

### Scrum.

Scrum es un marco de desarrollo enfocado en crear software funcional en ciclos cuyo principal objetivo es abordar problemas complejos. Esto permite tener un mayor control sobre la versión final pudiendo tener la oportunidad de adaptar o cambiar el desarrollo de forma continua.

La metodología sigue una estructura cíclica que se repite hasta finalizar el proyecto:

* Se selecciona uno o varios problemas que deban ser tratados y se dividen en tareas más asequibles que puedan ser realizadas en un marco temporal muy concreto, también denominados sprint.
* El equipo se reúne al finalizar un sprint para llevar un seguimiento de las tareas realizadas, ajustar expectativas y programar el siguiente sprint.

### Kanban.

## Técnica pomodoro.

Durante todo el desarrollo del proyecto se ha usado una interfaz web[[1]](#endnote-1) para seguir la técnica pomodoro. Su objetivo es mejorar la concentración y tener una administración del tiempo óptima. Para ello divide la carga de trabajo en periodos de 25 minutos de trabajo intenso - denominados *pomodoros* - seguidos por 5 minutos de descanso. Tras 4 pomodoros el tiempo de descanso será de 30 minutos.

## 4.2. Herramientas

Gestion de repositorio.

Git- GitHub.

GitHub es un software de gestión de versiones.

Control de versiones – GitHub

Alternativas.

Motivación.

GitHub cuenta con una aplicación de escritorio (GitHub Desktop) que simplifica considerablemente la gestión del código.

Gestion de proyecto – ZenHub. Jira.

Zenhub tiene mejores funcionalidads.

Desarrollo.

Lenguaje de programación – Python.

Se elegió Python por su versatilidad a la hora de tratar con conjuntos de datos.

Motivación.

Hay mucho desarrollo y facilita el analisis de datos

### Jupyter Notebooks.

Jupyter es un IDE (Integrated Development Environment) basado en web que permite organizar el código en notebooks. Soporta Python y su característica más importante es su diseño basado en celdas que permite outputs de código, HTML, gráficos, imágenes o incluso LaTeX. El diseño modular hace posible crear flujos de trabajo con una alta personalización.

La flexibilidad que otorga Jupyter Notebooks y el hecho de que esta ha sido una herramienta usada en numerosas asignaturas del grado fue lo que hizo que se descartasen otras opciones como PyCharm.

## 4.3 Librerias

### Plotly

Librería gratuita y de código abierto caracterizada por su accesibilidad. Se ha hecho uso de multitud de herramientas que permiten visualizar grandes conjuntos de datos de forma visualmente atractiva para crear gráficos.

### Dash

Dash es un framework de Python. Su aplicación original es crear aplicaciones web, pero su amplia versatilidad ha permitido Su amplia versatilidad permite pero su versatilidad y

Una de las grandes ventajas de Dash es que se puede renderizar directamente en el navegador.

Boostrap

Basado en plotly

openzource

Plotly.

Dash.

Dash es mas flexible. + opciones de botones, gráficos que se salen de lo normal.

Alternativas.

Streamlit, Voila, Panel.

### 4.4 Despliegue.

### Heroku

### 4.5 Documentación.

Zotero.

Gestor de referencias bibliográficas. –

Mendeley

Microsoft Word.

No se necesita aprender latex. Una curva de aprendizaje que no quería asumir.

1. ‘Pomofocus’ <https://pomofocus.io> [accessed 27 April 2022]. [↑](#endnote-ref-1)