Funcionalidades:

**Backend**

**FASE I**

* **Método Sign in:**
  + Se le pide al usuario user\_name y password como input en la terminal.
  + Se conecta con selenium a la web Buzzerbeater y mediante scraping se localiza el html input de username y password donde se escribe y se la da a submit para entrar a la pagina principal del usuario.
  + Usamos time.sleep para ralentizar el proceso y hacerlo parecer más humano.
* **Método get\_players\_to\_csv:**
  + A partir de las nacionalidades seleccionadas (en este momento hemos creado un diccionario fijo con todas las selecciones que nos interesan) donde se recorren una a una scrapeando la siguiente info de cada player del equipo:
    - Nacionalidad
    - Nombre
    - ID
    - Forma
    - DMI (Valoración)
  + Una vez tenemos esta info la convertimos a listas que posteriormente mediante pandas lo convertimos a .csv, creando un fichero llamado “players\_analysis.csv” en el mismo directorio de proyecto.
* **Selenium en segundo plano:**
  + Investigar cómo podemos usar selenium sin el navegador abierto y que el proceso se ejecute de forma automática.
  + Crear un popup una vez terminado el proceso que avise del dato scrapeado o de los errores que se han producido.
* **Crear nuevas tablas BD para:**
  + Tabla Países 🡪 Pais, ID Pais
  + Tabla enlace 🡪 ID Pais, ID temporada, ID jugador (hasta aquí PK), uid\_enlace
  + Tabla Jugadores 🡪 ID Jugador, Nombre, Edad
  + Tabla Performance 🡪 uid\_enlace (FK), semana (PK), DMI, forma
* **Sacar Forma con decimales:** *(Aplazado 🡪 Necesitamos encontrar el calculo ideal)*
  + Sacar valor mínimo y máximo (más reciente, ya que es absoluto) y buscar un algoritmo. (Regla de 3 sacando el valor máximo)
* **Método exportar info equipo:** 
  + Lista jugadores con sus formas a lo largo de las semanas de la temporada en cuestión
  + Cálculo aproximado de la forma actual.

**FASE II**

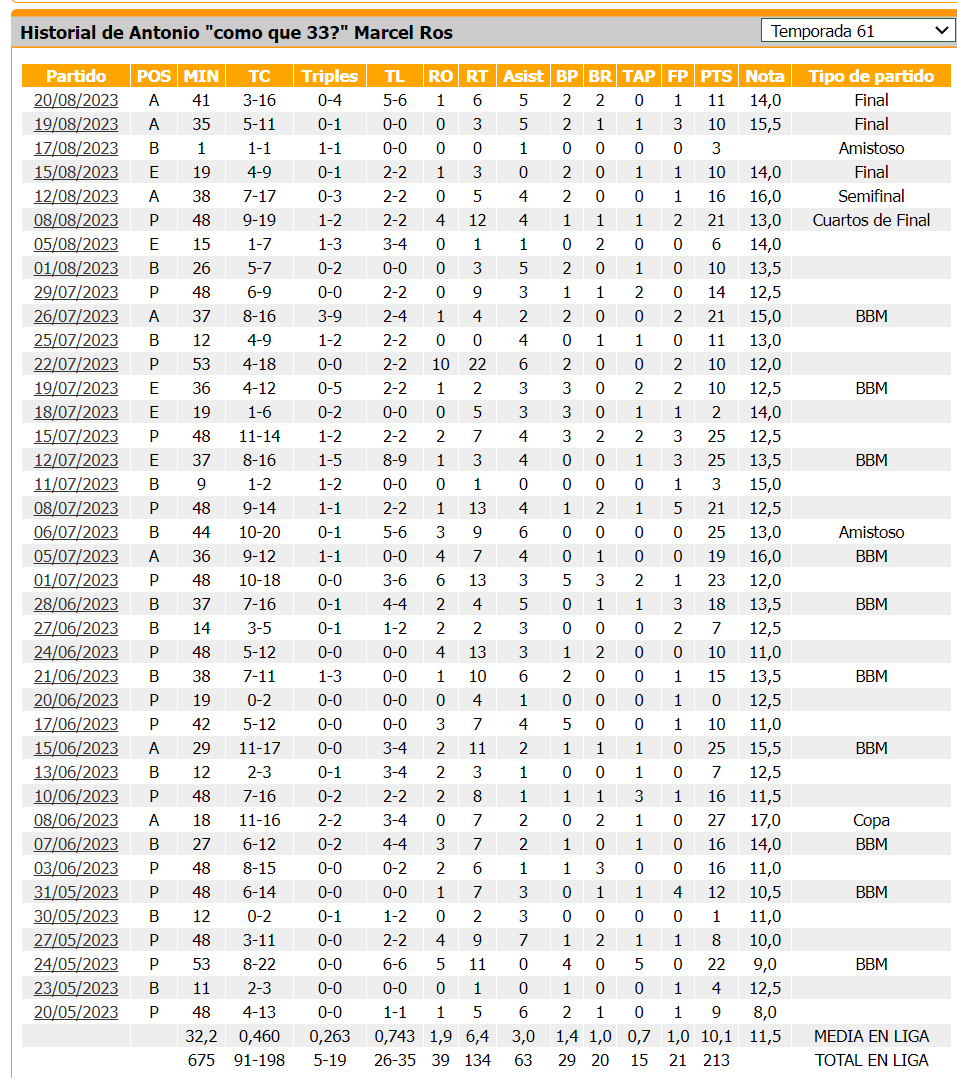
* **Frontend de la FASE I:**
  + HTML/CSS/BOOTSTRAP

**Objetivo Web**

* **Poder ver las formas (Historico), DMI (histórico), entrenos (histórico) y habilidades por jugador (actuales)**

1. **Pantalla Menú**
   1. **Botón Jugadores**

**FASE III**

* **Añadir tablas BD:**
  + Tabla temporada 🡪 ID temporada, semana, fecha inicio, fecha fin
  + Tabla Entrenos 🡪 ID temporada (FK), semana (FK), posición (PK), minutos
    - Crear nuevo método dentro de menú (Selección número 4).
      * Añadir manualmente seasons anteriores.
      * Seleccionar un país y extraer la info por jugador por temporada de la siguiente tabla:
      * Excluir “tipo de partido” = “BBM” y “Liga Privada”
      * NOS INTERESA:
        + Agrupado por Numero de semana
        + Numero de minutos por posición
      * A tener en cuenta:
        + Si el jugador tiene 21 años y estamos en la temporada 62, quiere decir que con 18 años (edad minima) era la temporada 59 (Hacer cálculo en codigo).
        + Puede haber jugadores que empezaron a jugar con 19 años (Si ese jugador no tiene datos para temporada con 18a aparecerá todo en 0 (Partido sin fechas)).
* <https://nt.buzzerbeater.com/>
  + ***Añadir al método de menú 1***
  + ***Crear código Python para cambiar nombres equipos y adecuarlos a los nombres de buzzerbeater***
  + ***Manipular metodo get\_players\_info para aceptar nuevos ids.***
  + Mirar como poder mover los slicers con selenium (o elegir valores sin necesidad de usar el slicer). (Mirar <span style="left:0%;width:100%" class="**MuiSlider-track css-1t2bqnt**"></span>)
  + Los filtros deseados serían
    - salario > 17.5K hasta final
    - Edad 21
    - Potencial >6 hasta final
  + Fijarnos en la lista de jugadores en “plantilla” y matchearla con los jugadores que nos aparecen en nt.
    - Los que no aparecen, de la url del jugador (esta en el nombre del jugador) extraer su id.
    - Con la id añadirlo al listado de players a extraer.
    - Extraer info con el metodo get\_players\_info
* Añadir campos: