



UNIVERSIDAD DE BURGOS
ESCUELA POLITÉCNICA SUPERIOR
Gº en Ingeniería Informática



TFG Ingeniería Informática:
NetExtractor



Presentado por Jorge Navarro González
en Burgos el 13 febrero de 2020
Tutores D. José Manuel Galán Ordax
y Dña. Virginia Ahedo García

Índice general

Índice general	2
Índice de figuras	1
Índice de tablas	3
Apéndice A. Planificación	1
1.1 Introducción	1
1.2 Planificación temporal	1
Sprint 1 (23/09/2019 – 07/10/2019)	2
Sprint 2 (7/10/2019 – 21/10/2019)	3
Sprint 3 (21/10/2019 – 04/11/2019)	4
Sprint 4 (04/11/2019 – 18/11/2019)	5
Sprint 5 (18/11/2019 – 02/12/2019)	6
Sprint 6 (03/12/2019 – 19/12/2019)	7
Sprint 7 (27/12/2019 – 10/01/2020)	8
Sprint 8 (12/01/2020 – 26/01/2020)	10
Sprint 9 (12/01/2020 – 26/01/2020)	11
1.3 Estudio de viabilidad económica	13
Costes	13
Beneficios	16
1.4 Estudio de la viabilidad legal	16
Apéndice B. Especificación de requisitos	18
2.1 Introducción	18
2.2 Objetivos generales	18
2.3 Catálogo de requisitos	18
Requisitos funcionales	19
Requisitos no funcionales	20
2.4 Especificación de requisitos	20
Diagrama de casos de uso	21
Actores	22
Casos de uso	22
Apéndice C. Especificación de requisitos	34
3.1 Introducción	34
3.2 Diseño de datos	34
Variables de sesión	34
Base de datos temporal	34
Clase personaje	34

Índice general

Clase Modelo.....	35
3.3 Diseño arquitectónico.....	36
Modelo-Vista-Presentador (MVP).....	36
Arquitectura general de la aplicación.....	36
Diagrama de paquetes	38
Diagrama de clases	39
Diseño procedimental.....	44
3.4 Diseño de interfaces	47
Apéndice D. Documentación técnica de programación.....	53
4.1 Introducción.....	53
4.2 Estructura de directorios.....	53
Repositorio principal.....	53
Repositorio despliegue	54
Repositorio MediaWiki	55
4.3 Manual de programador	55
Ejecución en local.....	55
Desplegar aplicación en remoto con Heroku.....	57
Internacionalización	59
Wiki	61
4.4 Realización de pruebas.....	61
4.5 Algunos comentarios adicionales.....	61
Apéndice E. Documentación de usuario	62
5.1 Introducción.....	62
5.2 Requisitos de usuario	62
5.3 Instalación.....	62
5.4 Manual de usuario.....	62
Alertas antes de usar la aplicación	62
Barra de navegación.....	63
Botones de navegación de la aplicación.....	63
Página principal	63
Menú de selección	64
Epub – Introducir ePub	65
Epub – Diccionarios automáticos	65
Epub – Importar diccionario.....	66
Epub – Obtener diccionario.....	67
Película – Introducción del guion	68
Modificar un diccionario.....	70

Índice general

Añadir personaje	70
Borrar personajes	71
Juntar personajes.....	72
Añadir referencias.....	73
Eliminar referencia.....	73
Modificar la id del personaje	74
Modificar etnia	75
Modificar sexo	76
Parámetros que introducir para la creación de la red	77
Generador de la red.....	79
Extracción de datos para el informe.....	80
Visualización del informe	81
Bibliografía	83

Índice de figuras

Figura 1 Burndown sprint 1	2
Figura 2 Burndown sprint 2	3
Figura 3 Burndown sprint 3	4
Figura 4 Burndown sprint 4	5
Figura 5 Burndown sprint 5	6
Figura 6 Burndown sprint 6	7
Figura 7 Burndown sprint 7	9
Figura 8 Burndown sprint 8	10
Figura 9 Burndown sprint 9	11
Figura 10 Impuestos salario	14
Figura 11 Diagrama de casos de uso	21
Figura 12 Modelo-Vista-Presentador	36
Figura 13 Arquitectura de la aplicación.....	37
Figura 14 Diagrama de paquetes	39
Figura 15 Diagrama de clases	40
Figura 16 Clase CreaDict.....	41
Figura 17 Clase TempBD	41
Figura 18 Clase Lectorcsv.....	41
Figura 19 Clase LecturaEpub.....	41
Figura 20 Clase controlador	41
Figura 21 Clase Modelo.....	42
Figura 22 Clase EthneaGenni.....	43
Figura 23 Clase CrearDiccionario	43
Figura 24 Clase Personaje.....	43
Figura 25 Clase PosPersonajes.....	43
Figura 26 Diagrama de secuencia para los algoritmos de creación de diccionarios.....	45
Figura 27 Diagrama de secuencia para los algoritmos de obtención de posiciones y predicción de etnia y sexo	46
Figura 28 Prototipo introducir guion	47
Figura 29 Prototipo selección.....	47
Figura 30 Prototipo visualización de la red.....	48
Figura 31 Prototipo modificar diccionario	48
Figura 32 Prototipo extracción características	49
Figura 33 Prototipo informe.....	49
Figura 34 Seleccion final	50
Figura 35 Introducir guion final	50
Figura 36 Visualización de la red final.....	51
Figura 37 Modificar diccionario final.....	51
Figura 38 Informe final.....	52
Figura 39 Extracción de datos final.....	52
Figura 40 Main.py	56
Figura 41 Debug Mode On	56
Figura 42 Debug Mode Off.....	56
Figura 43 Crear nueva aplicación.....	58
Figura 44 Seleccionar repositorio.....	58
Figura 45 Enlazar GitHub.....	58
Figura 46 Despliegue correcto	59

Índice de figuras

Figura 47 Despliegue.....	59
Figura 48 Traducciones poedit.....	60
Figura 49 Añadir idioma.....	60
Figura 50 Ejemplo escena.....	63
Figura 51 Barra de navegación.....	63
Figura 52 Botones de navegación.....	63
Figura 53 Página principal de NetExtractor.....	64
Figura 54 Selección de opciones NetExtractor	64
Figura 55 Introducción de ePub	65
Figura 56 Creación de diccionarios.....	66
Figura 57 Importar diccionario.....	67
Figura 58 Obtener diccionario.....	68
Figura 59 Introducción guion.....	69
Figura 60 Error en formato	69
Figura 61 Modificar personajes.....	70
Figura 62 Añadir personajes	71
Figura 63 Borrar personajes	72
Figura 64 Juntar personajes.....	72
Figura 65 Añadir referencia	73
Figura 66 Borrar referencia.....	74
Figura 67 Modificar id.....	75
Figura 68 Modificar etnia	76
Figura 69 Modificar sexo.....	77
Figura 70 Parámetros adicionales ePub	78
Figura 71 Parámetros adicionales películas	79
Figura 72 Visualización de la red.....	80
Figura 73 Extracción de datos.....	81
Figura 74 Informe.....	82

Índice de tablas

Tabla 1 Estimaciones de tiempo.....	1
Tabla 2 Distribución tareas sprint 1.....	3
Tabla 3 Distribución tareas sprint 2.....	4
Tabla 4 Distribución tareas sprint 3.....	5
Tabla 5 Distribución tareas sprint 4.....	6
Tabla 6 Distribución tareas sprint 5.....	7
Tabla 7 Distribución tareas sprint 6.....	8
Tabla 8 Distribución tareas sprint 7.....	9
Tabla 9 Distribución tareas sprint 8.....	11
Tabla 10 Distribución tareas sprint 9.....	12
Tabla 11 Costes totales	16
Tabla 12 Licencias de las herramientas utilizadas.....	17
Tabla 13 Caso de uso 1	22
Tabla 14 Caso de uso 2.....	22
Tabla 15 Caso de uso 3.....	23
Tabla 16 Caso de uso 4.....	23
Tabla 17 Caso de uso 5.....	23
Tabla 18 Caso de uso 6.....	24
Tabla 19 Caso de uso 7.....	24
Tabla 20 Caso de uso 8.....	25
Tabla 21 Caso de uso 9.....	25
Tabla 22 Caso de uso 10.....	25
Tabla 23 Caso de uso 11.....	26
Tabla 24 Caso de uso 12.....	26
Tabla 25 Caso de uso 13.....	27
Tabla 26 Caso de uso 14.....	27
Tabla 27 Caso de uso 15.....	28
Tabla 28 Caso de uso 16.....	28
Tabla 29 Caso de uso 17.....	29
Tabla 30 Caso de uso 18.....	29
Tabla 31 Caso de uso 19.....	29
Tabla 32 Caso de uso 20.....	30
Tabla 33 Caso de uso 21.....	30
Tabla 34 Caso de uso 22.....	31
Tabla 35 Caso de uso 23.....	31
Tabla 36 Caso de uso 24.....	31
Tabla 37 Caso de uso 25.....	32
Tabla 38 Caso de uso 26.....	32
Tabla 39 Caso de uso 27.....	32
Tabla 40 Caso de uso 28.....	33

Apéndice A. Planificación

1.1 Introducción

En este apartado se va a tratar cómo se ha distribuido el trabajo a lo largo de los sprints y como se ha planificado el trabajo que se va a realizar a lo largo de cada sprint. Como ya se ha comentado en la memoria, para realizar la planificación del proyecto vamos a usar la herramienta ZenHub, ya que es una extensión de navegador que añade funcionalidades a GitHub, donde tenemos alojado nuestro proyecto, y nos permite su gestión de una forma sencilla y eficiente.

Para empezar con la planificación, en la primera reunión se establecieron los objetivos que tendría el proyecto. Una vez definidos dichos objetivos se añadieron ciertas etiquetas para poder clasificar cada una de las tareas que se irían añadiendo al proyecto de una forma eficiente, ya que las etiquetas que vienen por defecto a veces no son suficientes para cubrir todas las funcionalidades que estamos realizando.

Al emplear la metodología SCRUM, hay que establecer una estimación del tiempo que se va a necesitar para la realización de cada tarea, este tiempo también puede ser modificado y asignado cuando creamos una tarea en nuestro repositorio de GitHub. Para hacernos una idea general y poder asignar los tiempos de una forma más eficiente, al inicio del proyecto se establecieron unos tiempos, acorde con cada uno de los valores que nos deja seleccionar como estimación.

Story Point	Estimación
1	10 minutos
2	20 minutos
3	40 minutos
5	1 hora
8	2 horas
13	5 horas
21	10 horas
40	24 horas

Tabla 1 Estimaciones de tiempo

Como se ha mencionado se ha realizado una metodología SCRUM, no se ha llegado a aplicar en su totalidad debido a que no era un proyecto llevado a cabo por un grupo de trabajo en el que hubiera reuniones diarias. Pero se ha intentado mantener al máximo la forma de trabajar:

- Se ha realizado una reunión a principio de cada sprint para establecer lo que se iba a realizar durante el mismo.
- Se ha realizado una reunión al final de cada sprint para hacer repaso de todo lo realizado durante dicho sprint.
- Se han estimado los tiempos que se iban a necesitar para cada una de las tareas que se realizarían durante los distintos sprints.
- Se iba entregando una parte del proyecto al final de cada sprint como parte del desarrollo incremental.

1.2 Planificación temporal

Para este apartado se van a describir todos los sprints que se han ido realizando a lo largo del proyecto, con todas las tareas y el gráfico *burndown* del sprint obtenido con la herramienta de ZenHub. Para ello se van a describir el número de *storypoints* que hay en el sprint, el nombre del

sprint, las tareas que se han realizado en cada sprint y las fechas de duración de cada sprint. También se detallarán las etiquetas que se ha dado a cada tarea dentro del sprint.

Sprint 1 (23/09/2019 – 07/10/2019)

Primer sprint del proyecto en el que se definen los objetivos que iba a tener el proyecto, cómo se iba a realizar y de manera general los pasos que se iban a seguir.

También se establecieron las tareas que se iban a llevar a cabo en dicho sprint, como era el sprint inicial, las tareas que se establecieron fueron comenzar con la lectura de documentación del proyecto, el *fork* al repositorio anterior, lectura a fondo de código para facilitar su entendimiento una vez que se inicie su modificación, elección de la plataforma donde se iba a realizar la documentación, creación del repositorio dentro de GitHub y se comenzó con algunas instalaciones necesarias para la realización del proyecto (ZenHub, Zotero...).

Enlace al primer sprint del proyecto: [Sprint 1](#).

A continuación, se va a mostrar el gráfico *burndown* del sprint, este gráfico nos muestra de manera general cómo se ha ido avanzando en la realización de tareas dentro de dicho sprint.

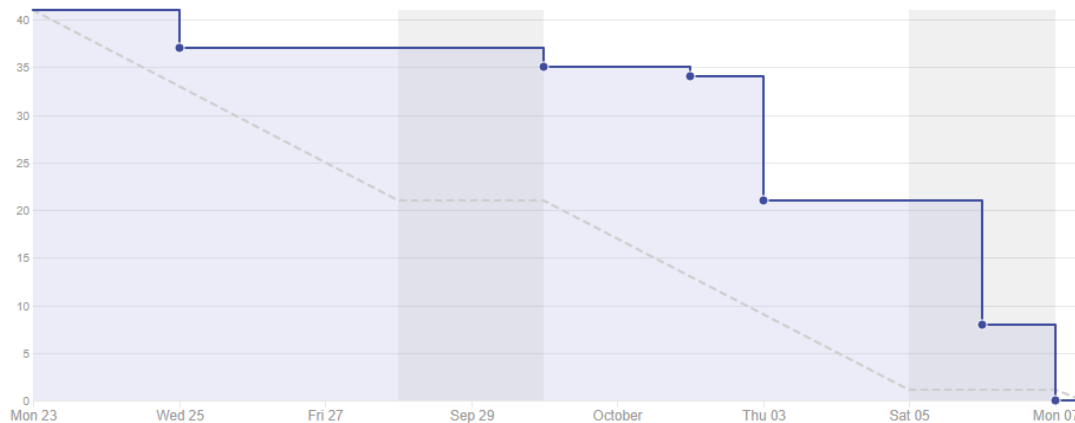


Figura 1 Burndown sprint 1

La línea discontinua muestra cómo habría sido el progreso a lo largo del sprint de una forma idónea, aunque no siempre puede ser así debido a factores externos, es por eso por lo que se puede ver como en ocasiones se avanza mucho en el proyecto y otras veces se ha avanzado poco.

Por último, se va a mostrar una tabla con todas las tareas relacionadas con la etiqueta correspondiente a cada tarea.

Tarea	Nombre	Bug	Decisión	Despliegue	Diseño	Documentación	Instalación	Duplicado	Investigación	Pregunta	Programación	Release	Testeo
#1	Instalación Zotero												
#2	Decidir herramienta documentación												
#3	Decidir herramienta gestión proyecto												
#4	Decidir plataforma repositorio												
#5	Lectura de la memoria												
#6	Lectura de anexos												
#7	Lectura de diagramas												
#8	Realización diagrama secuencias												

Tabla 2 Distribución tareas sprint 1

Sprint 2 (7/10/2019 – 21/10/2019)

En este sprint se empezó a ir más a fondo en la lectura del código, también se empezó a buscar solución al primer problema que existía respecto a las sesiones (la aplicación cerraba la sesión de usuario siempre que se utilizaban los botones de avanzar o retroceder de la propia aplicación, es decir, la aplicación resultaba inutilizable).

Enlace al segundo sprint del proyecto: [Sprint 2](#).

Este sprint tiene menos tareas que otros sprint ya que algunas de las tareas como leer el controlador o solucionar el problema de las sesiones, llevaron más tiempo y por tanto no permitieron avanzar demasiado en este sprint.

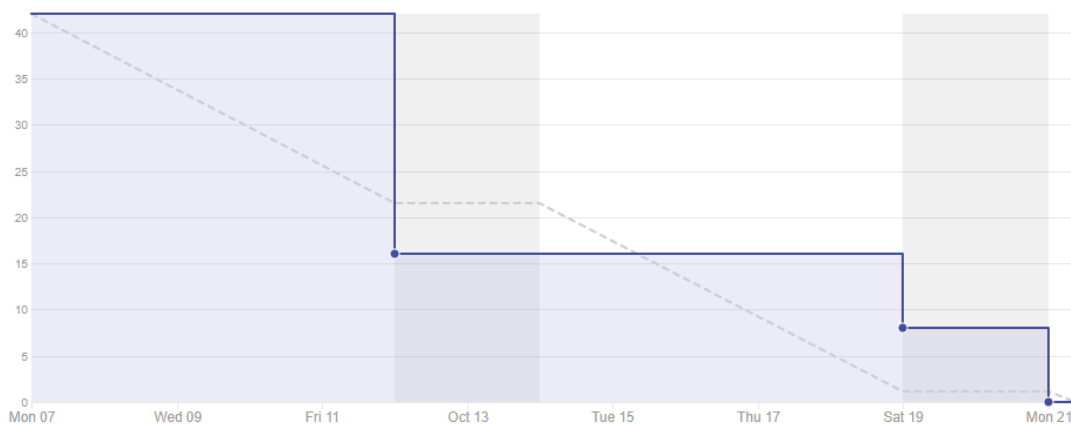


Figura 2 Burndown sprint 2

Como se puede observar en el gráfico *burndown*, las dos tareas que mencionamos anteriormente, son las penalizaron tiempo en este sprint.

A continuación, se encuentra la distribución de tareas del sprint 2.

Tarea	Nombre	Bug	Decisión	Despliegue	Diseño	Documentación	Instalación	Duplicado	Investigación	Pregunta	Programación	Release	Testeo
#9	Analizar el controlador												
#10	Investigar nueva forma de guardar las sesiones												
#11	Implementar la solución a las sesiones												
#12	Realizar documentación												
#13	Testear funcionamiento en distintos navegadores												

Tabla 3 Distribución tareas sprint 2

Sprint 3 (21/10/2019 – 04/11/2019)

En este sprint se empieza a tratar ya el tema de la generación de diccionarios para los guiones de película que se obtenían, es decir, la parte paralela a la obtención de personajes de las novelas. Para este sprint primero se tuvo que investigar en la página <https://www.imsdb.com/>, página en la cual se encuentran los guiones, para comprobar la estructura que tienen los guiones y ver cómo podemos obtener los personajes.

En este sprint se encontró un problema al obtener algunos guiones, no todos los guiones tienen la misma estructura, la mayoría comparten una estructura común de la que obtener los personajes, pero va a haber ciertos guiones que, como ya se explica en la memoria, no van a poder ser obtenidos de forma correcta debido a la estructura.

El enlace al tercer sprint del proyecto: [Sprint 3](#).

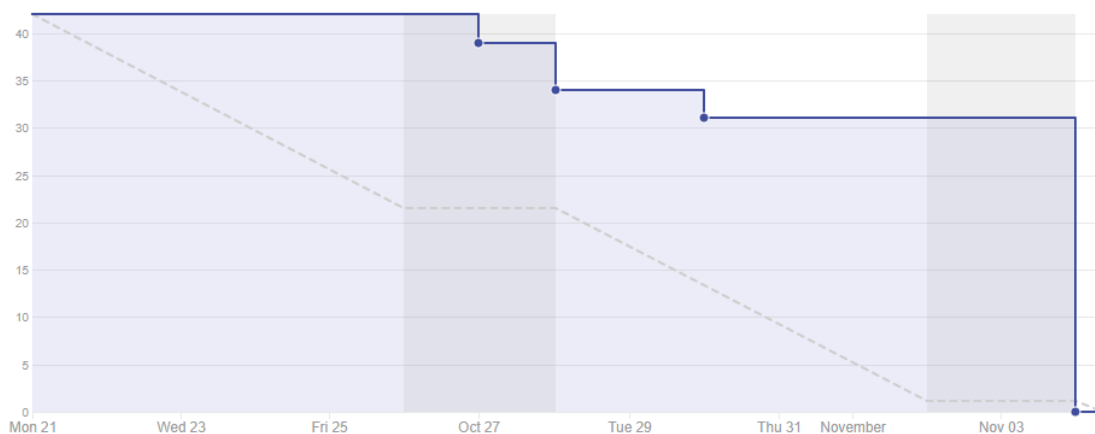


Figura 3 Burndown sprint 3

Y por último esta tabla refleja la distribución que han tenido las distintas tareas dentro del tercer sprint.

Tarea	Nombre	Bug	Decisión	Despliegue	Diseño	Documentación	Instalación	Duplicado	Investigación	Pregunta	Programación	Release	Testeo
#14	Investigar estructura de guiones												
#15	Examinar código ya realizado acerca de diccionarios												
#16	Hacer scraping para obtener personajes												
#17	Detectar número de apariciones												
#18	Obtener enlaces y peso												
#19	Crear diccionario de personajes												
#20	Juntar código realizado												

Tabla 4 Distribución tareas sprint 3

Sprint 4 (04/11/2019 – 18/11/2019)

El sprint 4 se ha tratado como un sprint intermedio, para solución de errores y eliminación de códigos duplicados, por ello este sprint posee un menor número de tareas y las que posee son sobre todo de errores y mejoras.

El enlace al cuarto sprint del proyecto: [Sprint 4](#).

El gráfico *burndown* generado para este sprint va a ser el siguiente.

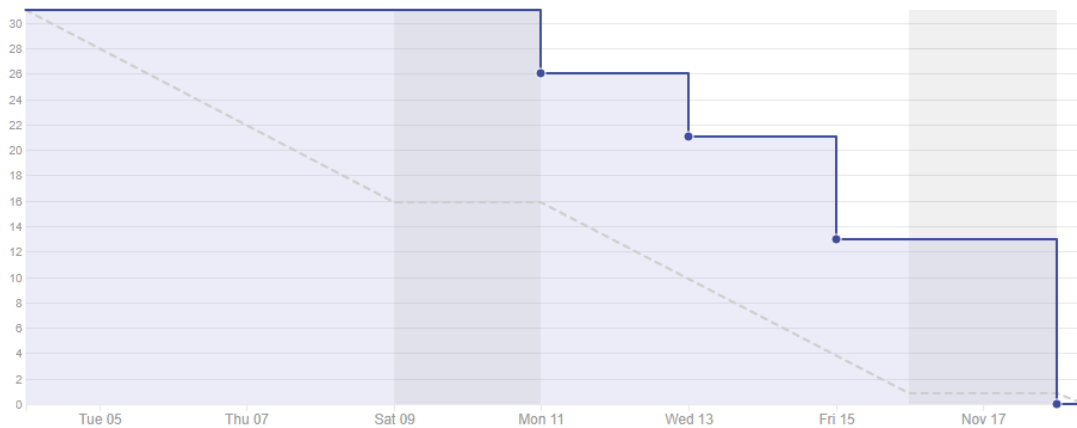


Figura 4 Burndown sprint 4

Como se puede observar en este sprint se realizó primero la tarea de eliminar código duplicado lo que llevo un tiempo importante debido a que las anteriores modificaciones, repetían bastantes ventanas de la aplicación y algunas funciones que se podían combinar en una sola.

La siguiente tabla va a mostrar la distribución que se ha llevado de las tareas a lo largo del sprint.

Tarea	Nombre	Bug	Decisión	Despliegue	Diseño	Documentación	Intalación	Duplicado	Investigación	Pregunta	Programación	Release	Testeo
#21	Realizar documentación												
#22	Eliminar código duplicado												
#23	Mejorar las sesiones de usuario para películas												
#24	Error de acceso a páginas intermedias de la aplicación												

Tabla 5 Distribución tareas sprint 4

Sprint 5 (18/11/2019 – 02/12/2019)

El sprint 5 es un sprint basado sobre todo en la implementación de la interfaz, en la implementación de informes de manera que la aplicación sea más vistosa para el usuario y más sencilla de visualizar la información que aparece en ella.

El enlace al quinto sprint del proyecto es el siguiente: [Sprint 5](#).

La siguiente imagen muestra el gráfico *burndown* generado en el sprint 5.

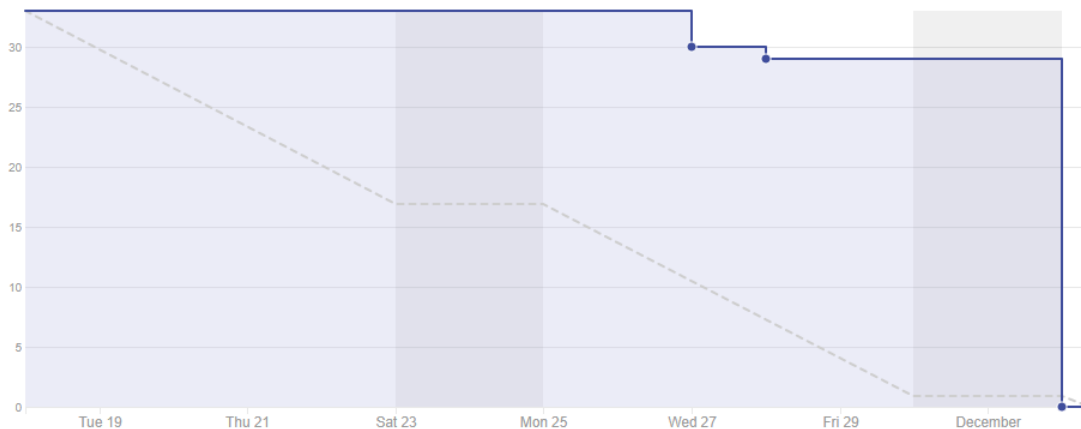


Figura 5 Burndown sprint 5

Y a continuación la distribución de las tareas a lo largo del sprint 5.

Tarea	Nombre	Bug	Decisión	Despliegue	Diseño	Documentación	Intalación	Duplicado	Investigación	Pregunta	Programación	Release	Testeo
#25	Análisis de plantillas css												
#26	Elección de la plantilla												
#27	Implementación de la plantilla												
#28	Observar distintos informes de redes												
#29	Implementación nuevo informe												

Tabla 6 Distribución tareas sprint 5

Sprint 6 (03/12/2019 – 19/12/2019)

En este sprint se continúa un poco con la temática del anterior, mejorar interfaz de usuario y añadir características al informe para que el usuario disponga de mayor cantidad de información. Es un sprint que contiene un gran número de tareas, pero que son de un tamaño reducido, con lo cual la estimación que se las asigno también es muy baja.

El enlace al sexto sprint del proyecto: [Sprint 6](#).

En este sprint se genera el siguiente gráfico *burndown*, como podemos ver, es fácilmente observable la cantidad de tareas que ha habido en este sprint, pero que el tiempo de cada una de ellas es en general bastante reducido.

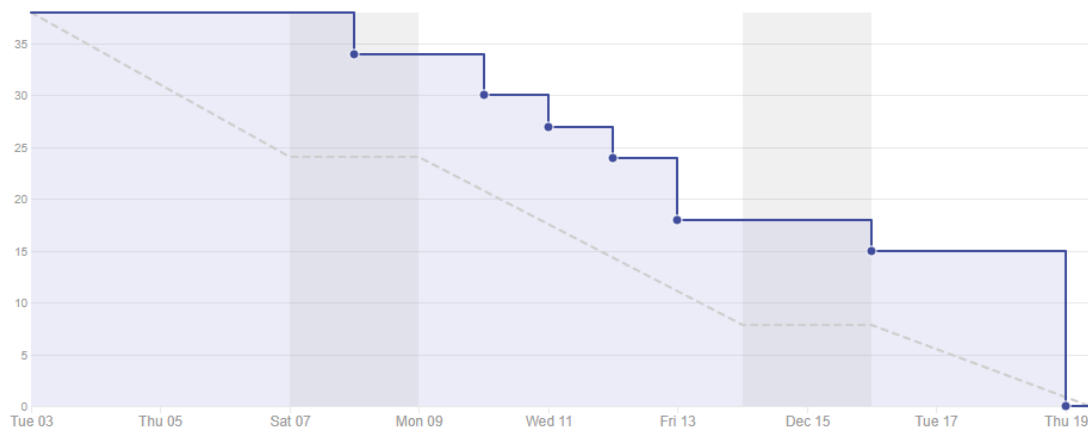


Figura 6 Burndown sprint 6

Para este sprint se ha tenido la siguiente distribución de tareas a lo largo del sprint, como se ha mencionado antes, es un sprint enfocado a la interfaz, por tanto, como podremos observar la mayor parte de las tareas tienen la etiqueta de diseño.

Tarea	Nombre	Bug	Decisión	Despliegue	Diseño	Documentación	Intalación	Duplicado	Investigación	Pregunta	Programación	Release	Testeo
#30	Poner el logo de NetExtractor en grande												
#31	Hacer una página Home												
#32	Crear jerarquía de botones												
#33	Quitar los gets en la selección de características a analizar												
#34	Colocar el escudo de la Ubu en acerca de y cambiar la ventana acerca de												
#35	Poner márgenes a lo largo de la aplicación												
#36	Mejorar el readme del repositorio												
#37	Añadir una imagen de fondo para la portada												
#38	Traducir la página web												
#39	Analizar la detección de roles												
#40	Intentar implementar louvain o un paquete con el que usarlo												

Tabla 7 Distribución tareas sprint 6

Sprint 7 (27/12/2019 – 10/01/2020)

En el sprint 7 ya tenemos una visión bastante acertada de lo que se va a convertir en la aplicación final, es por eso que ya pasamos del diseño a un sprint que busca más ultimar ciertos detalles del informe y empezar ya con el despliegue de la aplicación en algún servidor. Además, en este sprint ya se toma en cuenta el diseño de una wiki y se empiezan a probar opciones para su implementación.

El enlace al séptimo sprint del proyecto: [Sprint 7](#).

Aquí se puede observar el gráfico burndown generado para el sprint 7. En el que la tarea que más tiempo llevo fue la de investigar posibles servidores para alojar la web, que no fuera heroku, debido a sus restricciones en su forma gratuita, aun así, se decidió desplegar en Heroku debido a que todos los servicios de *hosting* gratuito poseían una serie de restricciones y Heroku parecía ser el servicio más sencillo de uso.

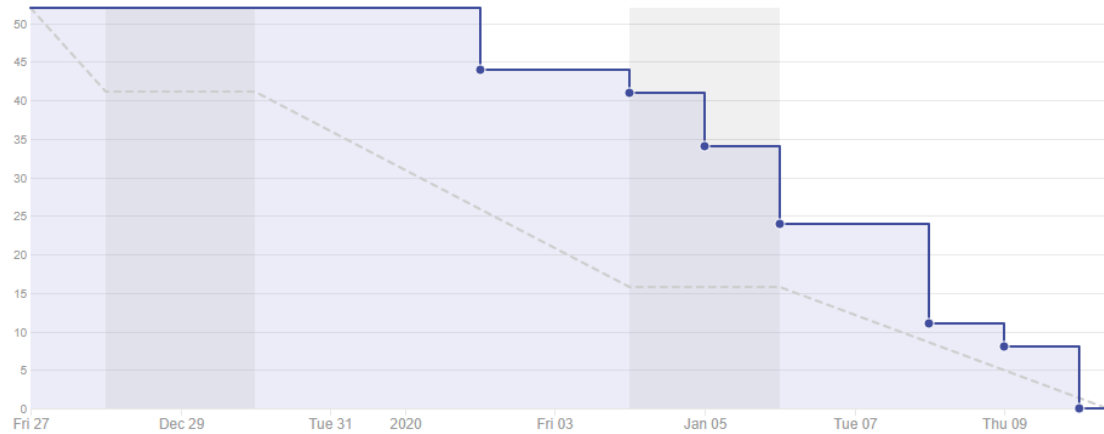


Figura 7 Burndown sprint 7

A continuación, se va a mostrar la distribución de tareas a lo largo del sprint 7, una distribución caracterizada por tener una mayoría de tareas con la etiqueta de “documentación” o de “investigación” ya que esta es la tarea en la cual se estaba buscando un servicio de *hosting* y una posible wiki a implementar.

Tarea	Nombre	Bug	Decisión	Despliegue	Diseño	Documentación	Intalación	Duplicado	Investigación	Pregunta	Programación	Release	Testeo
#41	Implementar nuevos algoritmos de detección de comunidades												
#43	Investigar posibles servidores												
#44	Documentar servidores investigados												
#45	Elección de servidor de hosting												
#46	Intento de despliegue de la aplicación en el servidor elegido												
#47	Elegir Wiki para la aplicación y comprobar validez en Heroku												
#48	Intento de despliegue MediaWiki en Heroku												
#49	Migración a Ubuntu												
#50	Creación de MediaWiki												

Tabla 8 Distribución tareas sprint 7

Como se puede ver en este sprint falta la tarea 42 ya que no se llegó a realizar en este sprint, sino que se movió al siguiente. También podemos observar la tarea de migración a Ubuntu, lo que consistía en mover el proyecto a una imagen de Ubuntu por problemas de compatibilidad con algunas funciones que luego al final fueron resueltas y no hubo que migrar nada.

Sprint 8 (12/01/2020 – 26/01/2020)

En el sprint 8 se van a implementar nuevas funcionalidades para el informe en los apartados de roles y comunidades y se va a implementar un sistema para predecir cuál va a ser el sexo y la etnia del personaje en cuestión. Además, surge un problema con MediaWiki ya mencionado en la memoria con las subidas de imágenes por tanto se decide que se usará wikidot.

El enlace al octavo sprint del proyecto: [Sprint 8](#).

Este es el gráfico burndown que se ha obtenido de las tareas del sprint 8.

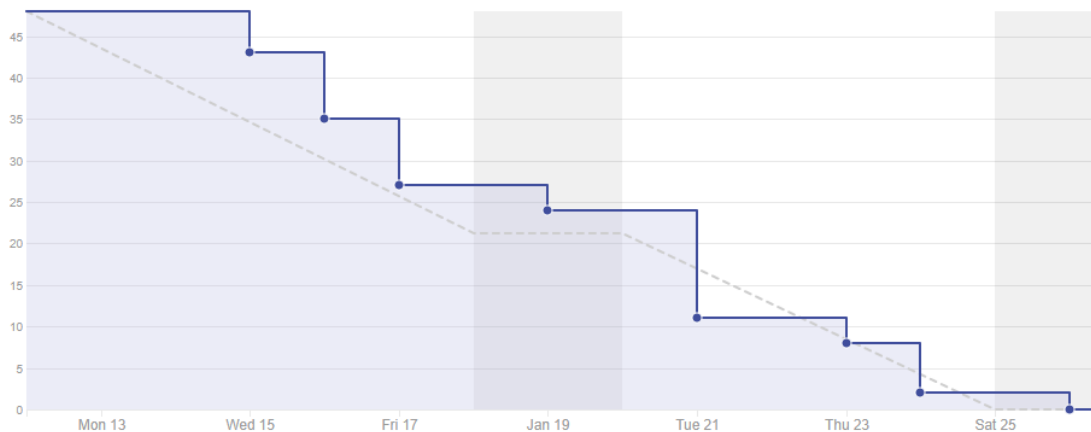


Figura 8 Burndown sprint 8

A continuación, se muestra la distribución de tareas a lo largo de este sprint, ya que es un sprint basado en nuevas implementaciones, la mayor parte de las tareas van a tener la etiqueta “programación”.

Tarea	Nombre	Bug	Decisión	Despliegue	Diseño	Documentación	Intalación	Duplicado	Investigación	Pregunta	Programación	Release	Testeo
#42	Implementar gráfico de detección de roles												
#51	Implementación de Louvain												
#52	Implementar de Ethnea&Genni												
#53	Implementar Genni												
#54	Problema mediawiki usar wikidot												
#55	Implementar detección de roles para cada una de las detecciones de comunidades												
#56	Separar obtención de posiciones de etnia y sexo												
#57	Crear dos redes												
#58	Mensaje de carga cuando se va a generar informe o obtener posiciones, etnia o sexo												
#59	Desplegar la aplicación completa												

Tabla 9 Distribución tareas sprint 8

Sprint 9 (12/01/2020 – 26/01/2020)

Este es el sprint final del proyecto en el cual se van a ultimar todos los detalles de la documentación. Es un sprint con pocas tareas, pero tareas densas.

El enlace al noveno sprint del proyecto: [Sprint 9](#).

El gráfico burndown del sprint.

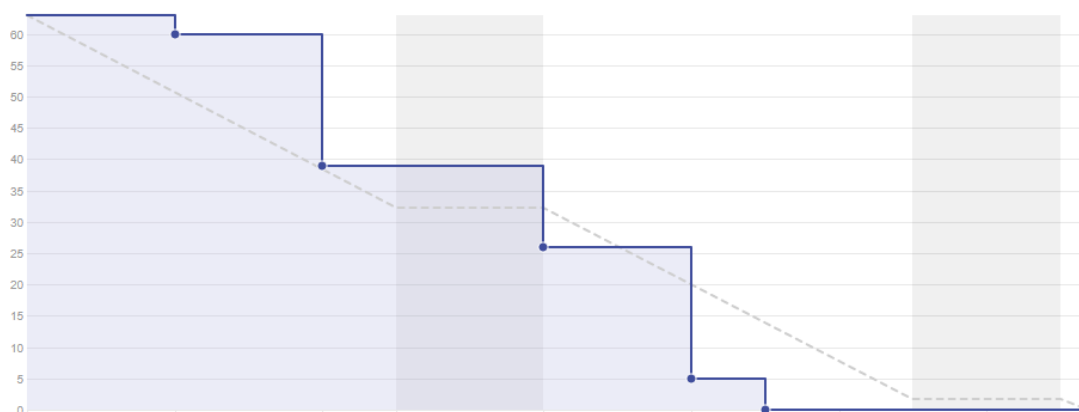


Figura 9 Burndown sprint 9

A continuación, se muestra la distribución de las tareas a lo largo del sprint para lo cual se puede observar que la mayoría de ellas son tareas de documentación.

Tarea	Nombre	Bug	Decisión	Despliegue	Diseño	Documentación	Intalación	Duplicado	Investigación	Pregunta	Programación	Release	Testeo
#60	Adaptar la app a movil												
#61	Crear imagen de la arquitectura de la aplicación												
#62	Realización de la memoria												
#63	Continuar wiki												
#64	Realización de anexos												

Tabla 10 Distribución tareas sprint 9

1.3 Estudio de viabilidad económica

En este apartado se van a calcular los costes y beneficios que se podría obtener con la aplicación en caso de que fuera desarrollada en un marco empresarial real.

Costes

Primero se van a detallar todos los costes:

- Empleados.
- Hardware.
- Software.

Empleados

En este apartado se van a calcular los costes de recursos humanos que supondría este proyecto. Se calcula que se han invertido 300 hora de trabajo repartidas en 6 meses. Esto supone un trabajo de 12,5 horas semanales que redondearemos a 13 horas. Se hace una estimación del salario que va a recibir el alumno de 20€/hora, esto supone que el salario bruto del alumno será el siguiente:

$$13 \frac{\text{horas}}{\text{semana}} \times 20 \frac{\text{€}}{\text{hora}} \times 4 \frac{\text{semanas}}{\text{mes}} = 1040\text{€ al mes}$$

Esto se corresponde con el salario bruto del empleado, a esto se le añaden los impuestos que tiene que pagar la empresa y que pueden ser consultados pulsando [aquí](#)(1). Estos impuestos se corresponden con:

- 26,3% de contingencias.
- 5,5% de desempleo.
- 0.20% de FOGASA.

- 0.60% de formación profesional.

TIPOS DE COTIZACIÓN (%)			
CONTINGENCIAS	EMPRESA	TRABAJADORES	TOTAL
Comunes	23,60	4,70	28,30
Horas Extraordinarias Fuerza Mayor	12,00	2,00	14,00
Resto Horas Extraordinarias	23,60	4,70	28,30

(1) Tipo de contingencias comunes (IT): Trabajadores con 65 años y 0 a 7 meses de edad y 36 años y 9 meses o más de cotización ó 65 años y 8 meses ó más de edad y 35 años y 6 meses o más de cotización: 1,50 por 100 (1,25 por 100 - empresa- y 0,25 por 100 -trabajador-).

(2) En los contratos temporales de duración igual o inferior a cinco días, la cuota empresarial por contingencias comunes se incrementa en un 40 por ciento. No se aplica al Sistema Especial para Trabajadores por Cuenta Ajena Agrario, incluido en el Régimen General.

DESEMPLEO	EMPRESA	TRABAJADORES	TOTAL
Tipo General	5,50	1,55	7,05
Contrato duración determinada Tiempo Completo	6,70	1,60	8,30
Contrato duración determinada Tiempo Parcial	6,70	1,60	8,30

	EMPRESA	TRABAJADORES	TOTAL
FOGASA	0,20		0,20

	EMPRESA	TRABAJADORES	TOTAL
FORMACIÓN PROFESIONAL	0,60	0,10	0,70

Figura 10 Impuestos salario

Por tanto, teniendo en cuenta dichos impuestos, nos sale que la empresa tiene que pagar por este empleado:

$$\frac{1040 \frac{\text{€}}{\text{mes}}}{1 - (0,236 + 0,055 + 0,002 + 0,006)} = 1484\text{€/mes}$$

Además, se dispone de dos profesores tutelando al alumno, con amplios conocimientos en el tema, por tanto, es lógico que el pago por sus servicios sea mayor (40€/hora). Los tutores, van a guiar al alumno a lo largo del proyecto trabajando un total de 2 horas/semana. Por tanto:

$$2 \frac{\text{horas}}{\text{semana}} \times 40 \frac{\text{€}}{\text{hora}} \times 4 \frac{\text{semanas}}{\text{mes}} = 320\text{€ al mes por profesor}$$

Son entonces, 640€ brutos entre los dos profesores que habrá que sumar los impuestos que ello conlleva:

$$\frac{640\text{€}}{\text{mes}} \div (1 - (0,236 + 0,055 + 0,002 + 0,006)) = 913\text{€ al mes entre los dos profesores}$$

Con lo cual, finalmente, la empresa tendrá que pagar un coste mensual de 2397€. El proyecto ha sido realizado a lo largo de seis meses, por tanto, el coste total que la empresa va a tener que pagar será: 14382€

Hardware

Para realizar el proyecto ha hecho falta un ordenador portátil para todo el desarrollo y pruebas en ordenador, y un dispositivo móvil para hacer las pruebas en móvil con el fin de comprobar la compatibilidad con diferentes dispositivos.

El ordenador ha costado un total de 1000€ que se van a amortizar en 4 años, por tanto:

$$\frac{1000\text{€}}{4 \text{ años}} = 250\text{€ por año}$$

Como el proyecto no ha durado un año, sino que ha supuesto 6 meses, medio año, el coste de la amortización será de 125€.

Además, como se ha mencionado, también se dispone de un dispositivo móvil para la realización de pruebas que ha costado 200€ y se amortizará también en 4 años, de tal forma que serán:

$$\frac{200\text{€}}{4 \text{ años}} = 50\text{€ por año}$$

Al igual que antes, se ha trabajado durante 6 meses en el proyecto, por tanto, el coste de amortización va a ser de 25€.

Con esto sale un total de $250\text{€} + 25\text{€} = 275\text{€}$ de costes de hardware.

Software

Para este proyecto todas las herramientas software han sido gratuitas excepto el sistema operativo empleado para ello. Dicho sistema ha sido el Windows 10 Home, valorado en 145,00€.

También se ha empleado el paquete de Office para la realización de la documentación. Este paquete para empresas está valorado en 8,80€ al mes. Ambos productos tienen una amortización de 4 años, por tanto:

- Costes Windows 10 Home:

$$\frac{145\text{€}}{4 \text{ años}} = 36,25\text{€ al año}$$

- Costes Office: el paquete cuesta 8.80€ al mes, el proyecto ha durado 6 meses por tanto ha costado un total de 52.8€:

$$\frac{52.8\text{€}}{4 \text{ años}} = 13,2\text{€ al año}$$

Por tanto, como el proyecto ha durado 6 meses, se dividen estos costes por dos, lo que hace un total de 18,125€ al año el sistema operativo y 6,6€ el paquete Office.

Todo esto junto suponen unos costes de software de 24,725€.

Total

Para calcular los costes totales que va a tener el proyecto, se estima que además hay una serie de costes indirectos, que son los que pertenecen a la luz, internet, infraestructura... Se establece este valor como el 15% de los gastos totales.

De esta forma podemos calcular los costes totales del proyecto como un total de:

Tipos de costes	Total €
-----------------	---------

Costes de empleados	14382€
Costes de hardware	275€
Costes de software	24,725€
Costes indirectos	2202,25€
Total	16884€

Tabla 11 Costes totales

Beneficios

En este apartado van a ser calculados los beneficios que podrían sacarse con la aplicación. Es cierto que la aplicación se ha diseñado de carácter educativo y, por tanto, no se obtiene ningún tipo de beneficio del uso que le den los clientes.

1.4 Estudio de la viabilidad legal

Para este apartado se va a proceder al estudio de las licencias que poseen las librerías y herramientas utilizadas para el desarrollo del proyecto, así como la propia licencia que se establecerá para el proyecto en sí.

Para este proyecto se han utilizado las siguientes herramientas y/o librerías:

Herramienta y/o librería	Versión	Descripción	Licencia
Flask	1.0.2	Framework de Python para crear aplicaciones web de forma rápida y sencilla.	BSD
Flask-babel	0.12.2	Extensión de Flask que permite la internacionalización.	BSD
Ethnea & Genni	-	Librería que permite predecir el sexo y etnia dado un nombre.	Licencia ¹
BeautifulSoup4	4.7.1	Librería que facilita el <i>scraping</i> web.	MIT
Matplotlib	3.0.2	Librería de Python para poder generar gráficos.	PSF
Networkx	2.2	Paquete de Python para la creación, manipulación y estudio de redes.	BSD
Scipy	1.1.0	Biblioteca de Python para implementar herramientas y algoritmos matemáticos.	BSD
Numpy	1.14.3	Extensión de Python, para dar soporte a vectores y matrices, posee funciones matemáticas de alto nivel.	BSD
Ply	3.11	Reimplementación de Lex y Yacc para Python	BSD
Html5lib	1.0.1	Biblioteca de Python con el fin de parsear documentos HTML.	MIT
Lxml	4.3.1	Librería de Python para poder procesar documentos XML y HTML.	BSD
network_styling_with_d3	-	Herramienta que nos permite visualizar redes de una forma más vistosa.	MIT

¹ Se han seguido los procedimientos marcados en la página: <http://abel.lis.illinois.edu/cgi-bin/ethnea/search.py?Fname=Bo&Lname=Svensson> notificando al autor de su uso y esperando 1 segundo entre cada interacción.

Pyhon-louvain	latest	Librería que nos permite implementar el algoritmo de Louvain en Python.	Licencia ²
---------------	--------	---	-----------------------

Tabla 12 Licencias de las herramientas utilizadas

Como podemos observar todas las herramientas empleadas para la realización del proyecto permiten su uso libre, por tanto, no tenemos ninguna clase de restricción por la cual necesitemos poner una licencia más restrictiva a nuestro proyecto.

Es por esto por lo que se ha decidido que el proyecto va a tener una licencia MIT, de tal forma que se permite el uso de la aplicación en todos los sentidos, es decir, se permite el uso de la aplicación libremente, modificación, copia, publicación, venta o distribución.

² Copyright (c) 2009, Thomas Aynaud <thomas.aynaud@lip6.fr> reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* Neither the name of the python-louvain Developers nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Apéndice B. Especificación de requisitos

2.1 Introducción

En este apartado se van a definir tanto los objetivos que se han tenido en la realización de este proyecto como los requisitos que se han definido, casos de uso y en general el comportamiento que va a tener la aplicación.

La estructura que se va a seguir para este apartado va a ser la siguiente:

- Introducción: breve descripción de lo que va a contener este apartado.
- Objetivos generales: los objetivos que se van a cumplir en este proyecto.
- Catálogo de requisitos: donde se van a mencionar todos los requisitos que hemos planteado en el proyecto.
- Especificación de requisitos: casos de uso y su definición.

2.2 Objetivos generales

Los objetivos generales que se establecen a la hora de realizar este proyecto son los siguientes:

- Realizar una aplicación web que permita extraer características de una red introducida.
- Poder elegir entre introducir una novela en formato EPUB o un guion extraído de una página en concreto.
- Realizar un informe estructurado y fácil de comprender por el usuario una vez extraídas las características deseadas.
- Permitir que el usuario pueda modificar los diccionarios de personajes extraídos de tal forma que:
 - Pueda añadir nuevos personajes.
 - Pueda eliminar los personajes existentes.
 - Pueda unir dos personajes en caso de que sean el mismo.
 - Pueda modificar la id de un personaje.
 - Pueda modificar el nombre de un personaje.
 - Pueda borrar el nombre de un personaje.
 - Pueda modificar la etnia de un personaje.
 - Pueda modificar el sexo de un personaje.
- Permitir al usuario visualizar una red y cambiarla a gusto del usuario.
- Permitir que un usuario pueda descargar la red en el formato deseado entre los posibles con el fin de que pueda importarla en otros programas como Gephi, Pajek, Cytoscape, etc.
- Permitir al usuario acceder a una wiki que le proporcione la información necesaria en cada pantalla.

2.3 Catálogo de requisitos

A continuación, se van a definir todos los requisitos derivados de los objetivos que se han marcado para el proyecto.

Requisitos funcionales

- **R.F-1 Selección ePub o Película:** La aplicación tiene que ser capaz de permitir al usuario elegir entre introducir el guion de una película o un libro en formato ePub.
- **R.F-2 Introducción de ePub:** La aplicación tiene que ser capaz de permitir que el usuario introduzca un libro en formato ePub, verifique que es correcto y de acceso a la siguiente pantalla.
- **R.F-3 Crear diccionario de forma automática:** La aplicación debe permitir crear un diccionario de forma automática a partir del ePub introducido.
- **R.F-4 Vaciar diccionario:** La aplicación debe permitir vaciar un diccionario ya existente.
- **R.F-5 Importación de diccionario:** La aplicación debe permitir que el usuario importe un diccionario desde un fichero csv que será validado.
- **R.F-6 Obtención de diccionario:** La aplicación debe permitir que el usuario obtenga un diccionario de una wiki de personajes.
- **R.F-7 Introducción de guion:** La aplicación debe permitir al usuario introducir una película en forma de enlace a una página preestablecida de forma que pueda leer dicha página y extraer sus personajes en un diccionario.
- **R.F-8 Modificar personajes:** La aplicación debe mostrar una lista con los personajes obtenidos y permitir al usuario modificarlos a su gusto.
 - **R.F-8.1 Añadir personaje:** La aplicación debe permitir al usuario añadir los personajes que él quiera.
 - **R.F-8.2 Borrar personaje:** La aplicación debe permitir al usuario borrar los personajes que a él le interese.
 - **R.F-8.3 Juntar personajes:** La aplicación debe permitir al usuario juntar los personajes como sea oportuno.
 - **R.F-8.4 Añadir referencia a un personaje:** La aplicación debe permitir añadir una referencia a la id del personaje que sea.
 - **R.F-8.5 Eliminar referencia a un personaje:** La aplicación debe permitir al usuario eliminar una referencia de una id aportada por el usuario.
 - **R.F-8.6 Modificar id personaje:** La aplicación debe permitir modificar el id de un personaje dado.
 - **R.F-8.7 Modificar etnia:** La aplicación debe permitir modificar la etnia de un personaje existente de la aplicación.
 - **R.F-8.8 Modificar sexo:** La aplicación debe permitir al usuario modificar el sexo del personaje deseado.
 - **R.F-8.9 Exportar diccionario:** La aplicación debe permitir al usuario exportar el diccionario de personajes que ha creado.
 - **R.F-8.10 Obtener posiciones:** La aplicación debe permitir obtener las posiciones en las que se encuentran los personajes dentro del catálogo de personajes.
 - **R.F-8.11 Obtener etnia y sexo:** La aplicación debe permitir obtener la etnia y el sexo de los personajes que se encuentran en el catálogo de personajes.
- **R.F-9 Añadir parámetros ePub:** La aplicación debe recoger parámetros correspondientes a número de apariciones, intervalo de aparición y si se tienen en cuenta los capítulos.
- **R.F-10 Añadir parámetros película:** La aplicación debe recoger el parámetro correspondiente al número de apariciones.

- **R.F-11 Visualización de la red:** La aplicación debe permitir al usuario visualizar la red de forma que pueda cambiarla al gusto que el usuario quiera.
 - **R.F-11.1 Exportación de la red:** La aplicación debe permitir exportar la red en los distintos formatos aportados.
- **R.F-12 Selección de características:** La aplicación debe permitir seleccionar al usuario las características deseadas con el fin de poder visualizarlas en el informe generado.
- **R.F-13 Visualización del informe:** La aplicación debe permitir al usuario visualizar el informe en el que estarán las características seleccionadas previamente y navegar a través del informe.
- **R.F-14 Botones de navegación:** La aplicación debe permitir al usuario el uso de forma correcta de los botones de navegación aportados por la aplicación.
- **R.F-15 Wiki:** La aplicación debe permitir al usuario acceder a la wiki de ayuda que contiene la guía de usuario.
- **R.F-16 Acerca de:** La aplicación debe permitir al usuario moverse a la página de “acerca de” con el fin de consultar la información que allí aparece.
- **R.F-17 Cambio de idioma:** La aplicación debe permitir cambiar el idioma de la página por aquellos idiomas que se encuentren disponibles en la barra de navegación.

Requisitos no funcionales

- **R.N.F-1 Usabilidad:** La aplicación debe ser intuitiva y que permita su navegación y aprendizaje de una forma rápida.
- **R.N.F-2 Escalabilidad:** La aplicación debe estar preparada para escalar, es decir, para que se puedan añadir nuevas funcionalidades de una forma rápida y sencilla.
- **R.N.F-3 Internacionalización:** La aplicación debe estar preparada para cambiar a los idiomas establecidos.
- **R.N.F-4 Seguridad:** Se debe gestionar todos los datos de una forma adecuada y sin mantener información vulnerable de forma innecesaria.
- **R.N.F-5 Mantenibilidad:** Se debe asegurar que la aplicación va a ser mantenible con facilidad, facilitando la labor de escalado.
- **R.N.F-6 Disponibilidad:** La aplicación debe poder estar disponible siempre que el usuario la requiera y mantenga conexión a internet.

2.4 Especificación de requisitos

En este apartado se va a describir un diagrama de caso de uso y se va a definir cada uno de los casos de uso que posee la aplicación. Se especifica que la frecuencia indicada en los casos de uso es la siguiente:

- **Baja:** la frecuencia baja significa que se usa ese caso de uso menos de un 30% de las veces que se usa la aplicación.
- **Media:** la frecuencia media significa que se usa ese caso de uso entre un 30-60% de las veces que se usa la aplicación.
- **Alta:** la frecuencia alta significa que se usa ese caso de uso entre un 60-90% de las veces que se usa la aplicación.
- **Muy alta:** la frecuencia muy alta significa que se usa ese caso de uso más de un 90% de las veces.

Diagrama de casos de uso

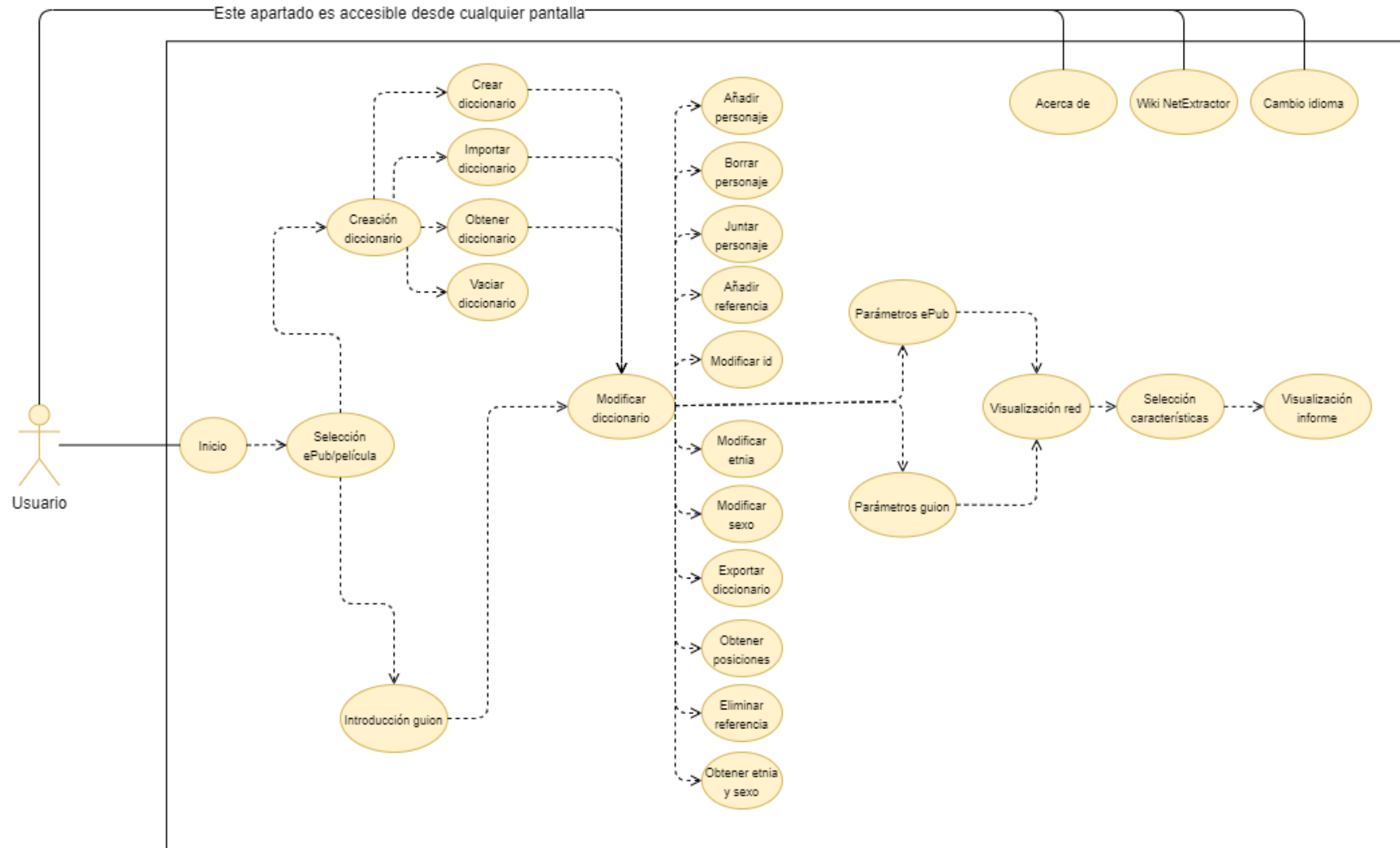


Figura 11 Diagrama de casos de uso

Actores

En esta aplicación vamos a tener un único actor, la diferencia es que cuando el actor introduzca un ePub o un guion, la aplicación creará una sesión temporal para ese actor mientras está navegando por la aplicación, por tanto, un actor que ya haya introducido una novela o un guion tendrá acceso a zonas donde si no has introducido nada no podrías entrar.

Casos de uso

Caso de uso 1	Seleccionar ePub o Película	
Requisitos asociados:	R.F-1	
Descripción:	Permite al usuario, aún sin sesión creada, seleccionar qué va a introducir, ePub o guion de película.	
Precondición:	Haber pulsado el botón de empezar en la pantalla principal.	
Acciones:	Paso	Acción
	1-	El usuario accede a la página de selección.
	2-	El usuario escoge una opción.
Postcondición:	Según la opción que se haya seleccionado, el usuario será dirigido al menú para introducir un ePub o un guion.	
Excepciones:	Esta pantalla no tiene ninguna excepción.	
Importancia:	Alta	
Frecuencia:	Muy alta	

Tabla 13 Caso de uso 1

Caso de uso 2	Introducción de ePub	
Requisitos asociados:	R.F-2	
Descripción:	Permite al usuario introducir un libro en formato ePub.	
Precondición:	Haber seleccionado el botón de ePub de la pantalla anterior.	
Acciones:	Paso	Acción
	1-	El usuario pulsa el botón de seleccionar archivo.
	2-	El usuario escoge el archivo de su explorador de archivos.
	3-	El usuario pulsa el botón de “Cargar”.
Postcondición:	El usuario es redirigido a la pantalla de creación de diccionarios.	
Excepciones:	Número	Excepción
	2-	El fichero introducido no se corresponde con un ePub.
	3-	El fichero introducido no tiene una estructura válida.
Importancia:	Alta	
Frecuencia:	Media	

Tabla 14 Caso de uso 2

Caso de uso 3	Crear diccionario de forma automática	
Requisitos asociados:	R.F-3	
Descripción:	Permite al usuario crear un diccionario de forma automática.	
Precondición:	Haber introducido un ePub de forma correcta.	
Acciones:	Paso	Acción
	1-	El usuario presiona el botón de “Crear diccionario”.
	2-	Se obtienen los personajes y se crea el diccionario.
	3-	Se muestra una alerta diciendo que se ha creado.
Postcondición:	Se crea el diccionario con los personajes, en caso de existir uno, se añaden al diccionario.	
Excepciones:	Este caso de uso no tiene excepción.	
Importancia:	Media	
Frecuencia:	Baja	

Tabla 15 Caso de uso 3

Caso de uso 4	Vaciar diccionario	
Requisitos asociados:	R.F-4	
Descripción:	Permite al usuario vaciar un diccionario existente.	
Precondición:	Haber introducido un ePub de forma correcta.	
Acciones:	Paso	Acción
	1-	El usuario presiona el botón de “Vaciar diccionario”.
	2-	Se vacía el diccionario.
	3-	Se muestra una alerta diciendo que se ha vaciado.
Postcondición:	Vacía el diccionario de personajes actual.	
Excepciones:	Este caso de uso no tiene excepción.	
Importancia:	Media	
Frecuencia:	Baja	

Tabla 16 Caso de uso 4

Caso de uso 5	Importación de diccionario	
Requisitos asociados:	R.F-5	
Descripción:	Permite al usuario importar un diccionario de personajes.	
Precondición:	Haber introducido un ePub de manera correcta.	
Acciones:	Paso	Acción
	1-	El usuario pulsa el botón de “Importar diccionario”.
	2-	El usuario pulsa el botón de seleccionar archivo.
	3-	El usuario escoge el archivo desde su pc.
	4-	El usuario pulsa el botón “Cargar”.
	5-	Se muestra una alerta diciendo que se ha importado.
Postcondición:	Se rellena el diccionario con los personajes importados.	
Excepciones:	Número	Excepción
	4-	El fichero introducido no tiene formato “.csv”
Importancia:	Alta	
Frecuencia:	Media	

Tabla 17 Caso de uso 5

Caso de uso 6	Obtención de diccionario	
Requisitos asociados:	R.F-6	
Descripción:	Permite al usuario obtener un diccionario sacado de una wiki en concreto.	
Precondición:	Haber introducido un ePub de manera correcta.	
Acciones:	Paso	Acción
	1-	El usuario introduce el link de la web correcta.
	2-	El usuario pulsa el botón “Obtener”.
	3-	Se notifica mediante una alerta el éxito en la obtención.
Postcondición:	Se rellena el diccionario con los personajes obtenidos.	
Excepciones:	Número	Excepción
	2-	El usuario no ha introducido la url correcta.
Importancia:	Baja	
Frecuencia:	Baja	

Tabla 18 Caso de uso 6

Caso de uso 7	Introducción del guion	
Requisitos asociados:	R.F-7	
Descripción:	Permite al usuario crear el diccionario a partir del guion obtenido de una página concreta.	
Precondición:	Haber seleccionado el botón “Película”.	
Acciones:	Paso	Acción
	1-	El usuario introduce el link de la web correcta.
	2-	El usuario pulsa el botón “Obtener”.
	3-	Se pasa a la siguiente pantalla de modificar personajes si ha tenido éxito.
Postcondición:	Se crea el diccionario con los personajes obtenidos.	
Excepciones:	Número	Excepción
	2-	El usuario no ha introducido la url correcta.
	2-	El usuario no ha introducido ninguna url.
	2-	El guion no sigue el formato, se redirige a una página de alerta.
Importancia:	Alta	
Frecuencia:	Media	

Tabla 19 Caso de uso 7

Caso de uso 8	Modificar personajes	
Requisitos asociados:	R.F-8, R.F-8.1, R.F-8.2, R.F-8.3, R.F-8.4, R.F-8.5, R.F-8.6, R.F-8.7, R.F-8.8, R.F-8.9, R.F-8.10, R.F-8.11	
Descripción:	Tener un diccionario cargado, es decir, haber introducido un ePub o un guion en la app y estar en la ventana de modificar personajes.	
Precondición:	Haber creado los diccionarios tanto de ePub como de película de forma correcta.	
Acciones:	Paso	Acción
	1-	Se cargan los personajes en una tabla.
	2-	Se muestran los personajes al usuario.
	3-	El usuario podrá elegir distintas opciones de modificación.
Postcondición:	El usuario visualiza los personajes.	
Excepciones:	En este caso de uso no hay excepciones.	
Importancia:	Alta	
Frecuencia:	Alta	

Tabla 20 Caso de uso 8

Caso de uso 9		Añadir personaje
Requisitos asociados:	R.F-8.1,- R.F-14	
Descripción:	Permite al usuario añadir un nuevo personaje al diccionario.	
Precondición:	Tener un diccionario cargado, es decir, haber introducido un ePub o un guion en la app y estar en la ventana de añadir personajes.	
Acciones:	Paso	Acción
	1-	El usuario introduce la id del personaje a introducir.
	2-	El usuario introduce el nombre del personaje.
	3-	El usuario pulsa "Añadir".
	4-	Se notifica al usuario con una alerta que el personaje se añadió.
Postcondición:	Se añade un nuevo personaje en el diccionario.	
Excepciones:	Paso	Excepción
	1-, 2-	El usuario no introduce alguno de los campos.
	2-	El usuario introduce una id ya existente.
Importancia:	Media	
Frecuencia:	Baja	

Tabla 21 Caso de uso 9

Caso de uso 10		Borrar personaje
Requisitos asociados:	R.F-8.2	
Descripción:	Permite al usuario borrar un personaje del diccionario.	
Precondición:	Tener un diccionario cargado, es decir, haber introducido un ePub o un guion en la app y estar en la ventana de borrar personajes.	
Acciones:	Paso	Acción
	1-	El usuario marca el/los personajes que desea eliminar.
	2-	El usuario presiona el botón de "Eliminar"
	3-	Se notifica al usuario con una alerta que el personaje ha sido eliminado.
Postcondición:	Se borra el personaje del diccionario.	
Excepciones:	En este caso de uso no hay excepciones.	
Importancia:	Media	
Frecuencia:	Media	

Tabla 22 Caso de uso 10

Caso de uso 11	Juntar personajes	
Requisitos asociados:	R.F-8.3	
Descripción:	Permite al usuario juntar un personaje con otro.	
Precondición:	Tener un diccionario cargado, es decir, haber introducido un ePub o un guion en la app y estar en la ventana de juntar personajes.	
Acciones:	Paso	Acción
	1-	El usuario marca los personajes que desea juntar.
	2-	El usuario presiona el botón de “Juntar”
	3-	Se notifica al usuario con una alerta que los personajes se han juntado.
Postcondición:	Se juntan los personajes en el diccionario de personajes.	
Excepciones:	En este caso de uso no hay excepciones.	
Importancia:	Media	
Frecuencia:	Media	

Tabla 23 Caso de uso 11

Caso de uso 12	Añadir referencia a un personaje	
Requisitos asociados:	R.F-8.4	
Descripción:	Permite al usuario añadir una nueva referencia a un personaje.	
Precondición:	Tener un diccionario cargado, es decir, haber introducido un ePub o un guion en la app y estar en la ventana de añadir referencias de personajes.	
Acciones:	Paso	Acción
	1-	El usuario introduce la id del personaje al que añadir referencia.
	2-	El usuario introduce el nombre de la referencia a añadir.
	3-	El usuario pulsa “Añadir”.
	4-	Se notifica al usuario con una alerta que la referencia del personaje se añadió.
Postcondición:	Se añade una referencia al personaje en el diccionario.	
Excepciones:	Paso	Excepción
	1-	El usuario introduce una id inexistente
	2-	El usuario introduce un nombre que ya existe para esa id, en cuyo caso simplemente dejará ese personaje como estaba.
Importancia:	Alta	
Frecuencia:	Media	

Tabla 24 Caso de uso 12

Caso de uso 13	Eliminar referencia a un personaje	
Requisitos asociados:	R.F-8.5	
Descripción:	Permite al usuario eliminar una referencia a un personaje.	
Precondición:	Tener un diccionario cargado, es decir, haber introducido un ePub o un guion en la app y estar en la ventana de eliminar referencias de personajes.	
Acciones:	Paso	Acción
	1-	El usuario marca la referencia a eliminar.
	2-	El usuario pulsa el botón eliminar.
	3-	Se notifica al usuario que la referencia fue eliminada.
Postcondición:	Se elimina la referencia del diccionario.	
Excepciones:	No hay excepciones en este caso de uso.	
Importancia:	Baja	
Frecuencia:	Baja	

Tabla 25 Caso de uso 13

Caso de uso 14	Modificar id personaje	
Requisitos asociados:	R.F-8.6	
Descripción:	Permite al usuario modificar la id de un personaje.	
Precondición:	Tener un diccionario cargado, es decir, haber introducido un ePub o un guion en la app y estar en la ventana de modificar id de personajes.	
Acciones:	Paso	Acción
	1-	El usuario introduce la id del personaje actual.
	2-	El usuario introduce la nueva id que quiere.
	3-	El usuario pulsa "Cambiar".
	4-	Se notifica al usuario con una alerta que la id ha sido modificada.
Postcondición:	Se modifica la id del personaje en el diccionario.	
Excepciones:	Paso	Excepción
	1-	El usuario introduce una id inexistente.
	2-	El usuario introduce un id existente como nueva id.
Importancia:	Baja	
Frecuencia:	Baja	

Tabla 26 Caso de uso 14

Caso de uso 15	Modificar etnia	
Requisitos asociados:	R.F-8.7	
Descripción:	Permite al usuario modificar la etnia.	
Precondición:	Tener un diccionario cargado, es decir, haber introducido un ePub o un guion en la app y estar en la ventana de modificar personajes.	
Acciones:	Paso	Acción
	1-	El usuario introduce la id del personaje.
	2-	El usuario introduce la etnia que desee.
	3-	El usuario pulsa “Modificar”.
	4-	Se notifica al usuario con una alerta que la etnia ha sido modificada.
Postcondición:	Se añade una etnia a la clase de personaje.	
Excepciones:	Paso	Excepción
	1-	El usuario introduce una id inexistente.
Importancia:	Media	
Frecuencia:	Media	

Tabla 27 Caso de uso 15

Caso de uso 16	Modificar sexo	
Requisitos asociados:	R.F-8.8, R.F-14	
Descripción:	Permite al usuario modificar el sexo.	
Precondición:	Tener un diccionario cargado, es decir, haber introducido un ePub o un guion en la app y estar en la ventana de modificar personajes.	
Acciones:	Paso	Acción
	1-	El usuario introduce la id del personaje.
	2-	El usuario selecciona el sexo en el desplegable.
	3-	El usuario pulsa “Modificar”.
	4-	Se notifica al usuario con una alerta que el sexo ha sido modificado.
Postcondición:	Se añade el sexo en la clase personaje.	
Excepciones:	Paso	Excepción
	1-	El usuario introduce una id inexistente.
Importancia:	Media	
Frecuencia:	Media	

Tabla 28 Caso de uso 16

Caso de uso 17	Exportar diccionario	
Requisitos asociados:	R.F-8.9	
Descripción:	Permite al usuario exportar el diccionario en formato csv.	
Precondición:	Tener un diccionario cargado, es decir, haber introducido un ePub o un guion en la app y estar en la ventana de modificar personajes.	
Acciones:	Paso	Acción
	1-	El usuario presiona el botón “Exportar Diccionario”.
	2-	Se descarga el archivo en el pc del usuario.
Postcondición:	Se descarga el diccionario y el usuario permanece en la misma pantalla.	
Excepciones:	En este caso de uso no hay excepciones.	
Importancia:	Alta	
Frecuencia:	Media	

Tabla 29 Caso de uso 17

Caso de uso 18	Obtener posiciones	
Requisitos asociados:	R.F-8.10, R.F-14	
Descripción:	Permite al usuario obtener las posiciones, es decir, el número de apariciones.	
Precondición:	Tener un diccionario cargado, es decir, haber introducido un ePub o un guion en la app y estar en la ventana de modificar personajes.	
Acciones:	Paso	Acción
	1-	Marcar la opción “Obtener Posiciones”.
	2-	Pulsar el botón de navegación de siguiente.
Postcondición:	Una vez pasada la pantalla de carga y obtenidas las posiciones, se volverá a la misma pantalla que antes, pero con las posiciones ya calculadas sobre la tabla.	
Excepciones:	En este caso de uso no hay excepciones.	
Importancia:	Alta	
Frecuencia:	Muy alta	

Tabla 30 Caso de uso 18

Caso de uso 19	Obtener etnia y sexo	
Requisitos asociados:	R.F-8.11, R.F-14	
Descripción:	Permite al usuario obtener la etnia y el sexo del personaje.	
Precondición:	Tener un diccionario cargado, es decir, haber introducido un ePub o un guion en la app y estar en la ventana de modificar personajes.	
Acciones:	Paso	Acción
	1-	Marcar la opción “Obtener Etnia y sexo”.
	2-	Pulsar el botón de navegación de siguiente.
Postcondición:	Una vez pasada la pantalla de carga y obtenido el sexo y la etnia, se volverá a la misma pantalla que antes, pero con el sexo y la etnia obtenidos.	
Excepciones:	En este caso de uso no hay excepciones.	
Importancia:	Media	
Frecuencia:	Alta	

Tabla 31 Caso de uso 19

Caso de uso 20	Añadir parámetros ePub	
Requisitos asociados:	R.F-9, R.F-14	
Descripción:	Introducir un filtro indicando el número mínimo de apariciones que se quiere del personaje, el rango de palabras que tiene que haber máximo entre dos personajes para ser considerado relación y si se quiere tener los capítulos en mente.	
Precondición:	Tener un diccionario cargado, es decir, haber introducido un ePub en la app y estar en la ventana de parámetros de novela.	
Acciones:	Paso	Acción
	1-	El usuario introduce el número mínimo de apariciones.
	2-	El usuario introduce el rango máximo de palabras.
	3-	El usuario <i>checkea</i> o no a su gusto la casilla de tener en cuenta capítulos.
	4-	El usuario pulsa el botón de “Obtener red”.
Postcondición:	Después del último paso, se saltará a la ventana de visualización de la red y se obtendrán los enlaces y añadirán los atributos etnia y sexo a cada nodo.	
Excepciones:	Paso	Excepción
	1-	No haber introducido alguno de los campos obligatorios.
Importancia:	Alta	
Frecuencia:	Media	

Tabla 32 Caso de uso 20

Caso de uso 21	Añadir parámetros película	
Requisitos asociados:	R.F-10, R.F-14	
Descripción:	Introducir un filtro indicando el número mínimo de apariciones que se quiere del personaje.	
Precondición:	Tener un diccionario cargado, es decir, haber introducido un guion en la app y estar en la ventana de parámetros de película.	
Acciones:	Paso	Acción
	1-	El usuario introduce el número mínimo de apariciones.
	2-	El usuario pulsa el botón de “Obtener red”.
Postcondición:	Después del último paso, se saltará a la ventana de visualización de la red y se obtendrán los enlaces y añadirán los atributos etnia y sexo a cada nodo.	
Excepciones:	Paso	Excepción
	1-	No haber introducido un número mínimo de apariciones.
Importancia:	Alta	
Frecuencia:	Media	

Tabla 33 Caso de uso 21

Caso de uso 22	Visualización de la red	
Requisitos asociados:	R.F-11, R.F-11.1, R.F-14	
Descripción:	Permite al usuario visualizar la red y modificarla a su gusto.	
Precondición:	Tener un diccionario cargado, es decir, haber introducido un ePub o un guion en la app y estar en la ventana de visualización de la red.	
Acciones:	Paso	Acción
	1-	Se llama a la librería de visualización con el grafo generado por la aplicación.
	2-	Se visualiza la red.
Postcondición:	Se visualiza la red, si se pulsa el botón de “Generar Informe” nos llevará a la pantalla de seleccionar características del informe.	
Excepciones:	En este caso de uso no hay excepciones.	
Importancia:	Alta	
Frecuencia:	Muy alta	

Tabla 34 Caso de uso 22

Caso de uso 23	Exportación de la red	
Requisitos asociados:	R.F-11.1	
Descripción:	Permite al usuario descargarse la red en diferentes formatos para luego poder emplearlos en otras herramientas.	
Precondición:	Tener un diccionario cargado, es decir, haber introducido un ePub o un guion en la app y estar en la ventana de visualización de la red.	
Acciones:	Paso	Acción
	1-	Se pulsa el botón de exportar en el formato deseado.
	2-	Se descarga la red en el formato seleccionado.
Postcondición:	Se queda en la misma pantalla, pero con la red descargada en el formato indicado en el ordenador del usuario.	
Excepciones:	En este caso de uso no hay excepciones.	
Importancia:	Alta	
Frecuencia:	Alta	

Tabla 35 Caso de uso 23

Caso de uso 24	Selección de características	
Requisitos asociados:	R.F-12, R.F-14	
Descripción:	Permite seleccionar al usuario las características que desea extraer de la red.	
Precondición:	Tener un diccionario cargado, es decir, haber introducido un ePub o un guion en la app y estar en la ventana de selección de características.	
Acciones:	Paso	Acción
	1-	Seleccionar las características deseadas.
	2-	El usuario pulsa el botón de “Generar informe”.
Postcondición:	Genera el informe con las características.	
Excepciones:	Paso	Excepción
	2-	Que alguno de los valores obtenidos no sea válido.
Importancia:	Alta	
Frecuencia:	Muy alta	

Tabla 36 Caso de uso 24

Caso de uso 25	Visualización del informe	
Requisitos asociados:	R.F-13	
Descripción:	Permite al usuario visualizar el informe.	
Precondición:	Tener un diccionario cargado, es decir, haber introducido un ePub o un guion en la app y estar en la ventana de visualización del informe.	
Acciones:	Paso	Acción (estas acciones son opcionales)
	1-	Pulsar el botón del menú de navegación que aparece en la parte izquierda de la pantalla para navegar por el informe.
	2-	Visualizar el informe.
Postcondición:	Se visualiza el informe.	
Excepciones:	En este caso de uso no hay excepciones.	
Importancia:	Alta	
Frecuencia:	Alta	

Tabla 37 Caso de uso 25

Caso de uso 26	Ayuda al usuario	
Requisitos asociados:	R.F-15	
Descripción:	Permite al usuario acceder a una wiki con ayuda para manejarte con la aplicación.	
Precondición:	Tener conexión a internet.	
Acciones:	Paso	Acción (estas acciones son opcionales)
	1-	El usuario presiona el botón de “Wiki”.
	2-	Se abre otra pestaña con la wiki de la aplicación.
Postcondición:	-	
Excepciones:	En este caso de uso no hay excepciones.	
Importancia:	Alta	
Frecuencia:	Baja	

Tabla 38 Caso de uso 26

Caso de uso 27	Información sobre la aplicación	
Requisitos asociados:	R.F-16	
Descripción:	Permite al usuario acceder a una ventana de “Acerca de” con información sobre la aplicación.	
Precondición:	Tener conexión a internet.	
Acciones:	Paso	Acción (estas acciones son opcionales)
	1-	El usuario presiona el botón de “Acerca de”.
	2-	Se abre otra pestaña con la información de la aplicación.
Postcondición:	-	
Excepciones:	En este caso de uso no hay excepciones.	
Importancia:	Baja	
Frecuencia:	Baja	

Tabla 39 Caso de uso 27

Caso de uso 28	Cambio de idioma	
Requisitos asociados:	R.F-17	
Descripción:	Permite al usuario cambiar el idioma de la aplicación.	
Precondición:	Tener conexión a internet.	
Acciones:	Paso	Acción (estas acciones son opcionales)
	1-	El usuario presiona la bandera del idioma que desea.
	2-	Se cambia el idioma de la aplicación.
Postcondición:	Se cambia el idioma	
Excepciones:	En este caso de uso no hay excepciones.	
Importancia:	Alta	
Frecuencia:	Baja	

Tabla 40 Caso de uso 28

Apéndice C. Especificación de requisitos

3.1 Introducción

En este apartado se van a definir los datos que van a ser manejados por la aplicación, su estructura, las interfaces que se han definido, detalles procedimentales...

3.2 Diseño de datos

En este apartado se van a explicar los tipos de datos que se están manejando en la aplicación, algunos de ellos, van a ser almacenados en variables de sesión, pero otros al tener un tamaño mucho mayor van a ser guardados en el objeto modelo.

Variables de sesión

Los datos que van a ser guardados en variables de sesión van a ser los siguientes:

- Sesión de usuario: en la que se va a guardar la id que tiene el usuario que entra a la aplicación.
- Sesión de fichero: es en la que se guarda el nombre de la novela o de la película que se ha introducido a la aplicación.
 - Nombre de la novela: en este caso lo que se va a añadir a la sesión del fichero va a ser el nombre del ePub que se ha subido en la aplicación.
 - Nombre de la película: por el contrario, en este caso lo que se hace es recortar la url de forma que solo se obtenga de ella el nombre de la película, es decir, si el enlace es <https://www.imsdb.com/scripts/Anastasia.html> se recorta de forma que en la variable de sesión solo se guarde “Anastasia”.
- Sesión de configuración de la red: en esta variable se va a guardar la configuración de la red que luego será mostrada en la pantalla de visualización.
- Sesión de idioma: en la que se guardará el idioma elegido en la aplicación.

La variable de fichero va a ser la que de nombre a los ficheros que pueden ser exportados a lo largo de la aplicación. La sesión de usuario va a ser la que sirva para enviar estos ficheros exportados a la carpeta de sesión correspondiente.

Base de datos temporal

También existe una clase utilizada para almacenar datos de los usuarios que estén usando la aplicación. En esta base de datos existe un diccionario, que guarda dos datos:

- La clave del diccionario que va a ser considerada la id del usuario.
- El valor de dicho diccionario que es el objeto modelo que lleva asociado dicho usuario.

La id del usuario se va a ir incrementando cada vez que se cree una nueva entrada en el diccionario, es decir, cada vez que un usuario entra en la aplicación.

Una vez que el usuario se sale de la aplicación, es decir, cierra su navegador, esa sesión va a ser borrada, y con ella todos los datos que hay en su interior.

Clase personaje

La forma para almacenar el número de apariciones y los distintos nombres que puede tomar un mismo personaje en el transcurso de una novela o película, va a ser mediante una clase

de Python llamada “Personaje”, en esta clase se almacena toda la información necesaria de un personaje.

La clase Personaje va a contar con los siguientes campos:

- Diccionario de nombres: un diccionario contiene como clave el nombre de un personaje y como valor tiene la posición en la que aparece.
- Diccionario de posiciones: un diccionario que tiene como clave el capítulo de la novela y como valor las posiciones en las que aparece en dicho capítulo.
- Diccionario de referencias: un diccionario que tiene como clave cada uno de los nombres que posee un personaje y como valor las apariciones de ese personaje.
- Número de apariciones: es una variable que posee el número de apariciones de cada personaje.
- Sexo: variable con el sexo del personaje.
- Etnia: variable con la etnia del personaje.
- Diccionario de sexo y etnia: es un diccionario que posee dos claves “Etnia” y “Sexo” y como valor tiene la etnia y sexo del personaje respectivamente. Este diccionario se emplea para añadir estos atributos a los nodos, ya que se pueden añadir como un diccionario directamente los dos en vez de tener que ir añadiendo los atributos uno a uno.

Clase Modelo

La clase modelo es la encargada de albergar la lógica que tiene la aplicación, todos los métodos que aportan la funcionalidad a la aplicación. En esta clase se van a almacenar algunos datos que son usados por la aplicación y que poseen un peso demasiado grande para almacenarse en una variable de sesión.

La clase modelo maneja los siguientes datos:

- Lista de texto: es una lista que contiene todo el texto que contiene la novela por cada capítulo.
- Lista de palabras: es una lista que contiene el número de palabras que tiene cada capítulo de la novela.
- Diccionario personajes: es un diccionario que contendrá todos los personajes de la novela o el guion de forma que la clave es la id del personaje y el valor es el objeto “Personaje”.
- El grafo generado con los atributos de etnia y sexo.
- El mismo grafo mencionado anteriormente, pero sin los atributos para poder ser visualizado de forma correcta.
- El enlace que va a tener la película que se introduzca.
- Diccionario de apariciones: es un diccionario creado simplemente para los algoritmos de obtención de personajes y posiciones en los guiones. Como clave tiene la id del personaje y como valor tiene una lista con los números de las escenas en las que aparece. Se utiliza sobre todo para la obtención de enlaces.
- Variable de cambio: empleada para acceder a las distintas pantallas entre película y novela. Ya que la aplicación tiene una estructura en la cual hay distintas pantallas comunes y otras pantallas no comunes entre novelas y películas, esta variable se emplea para alternar entre estas pantallas.

3.3 Diseño arquitectónico

En este apartado se va a explicar la estructura que tiene el proyecto, cómo se distribuyen los paquetes, como se conectan las clases entre ellas, el diseño procedimental de la aplicación reflejado en un diagrama de secuencias y cómo han sido desarrolladas las interfaces de la aplicación.

Modelo-Vista-Presentador (MVP)

Se ha elegido emplear un patrón que cada vez está cobrando más importancia en el desarrollo de aplicaciones como es el Modelo-Vista-Presentador. Uno de los motivos por los cuales se ha implementado esto es la facilidad que aporta al desarrollador a la hora de realizar modificaciones, ya que podemos cambiar la lógica de la aplicación, sin tener que cambiar ningún requisito de la interfaz y viceversa(2). Además, gracias al desarrollo de la aplicación con Flask, se permite implementar este patrón de una forma muy sencilla.

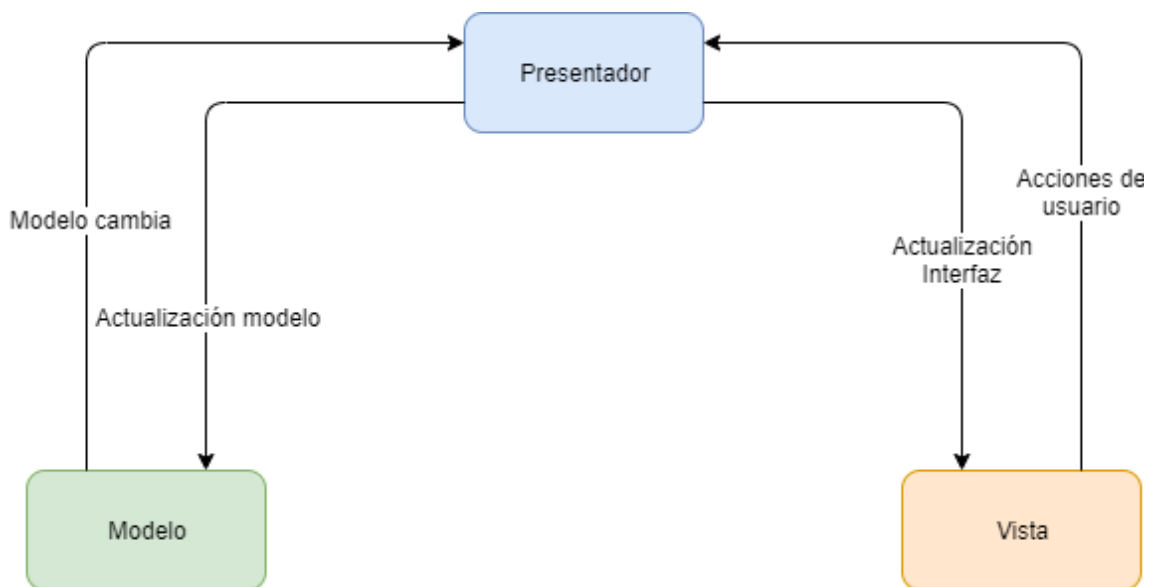


Figura 12 Modelo-Vista-Presentador

Arquitectura general de la aplicación

La arquitectura que se va a mostrar a continuación muestra los inputs que se introducen en la aplicación que se corresponden con la novela o película y el informe generado como output. Además, como es una aplicación de Python, se van a mostrar algunas librerías de Python y herramientas que se han empleado en su desarrollo, también aparecen dos librerías externas que se han utilizado para la funcionalidad añadida de Etnia y Sexo(3) y para la visualización de la red(4).

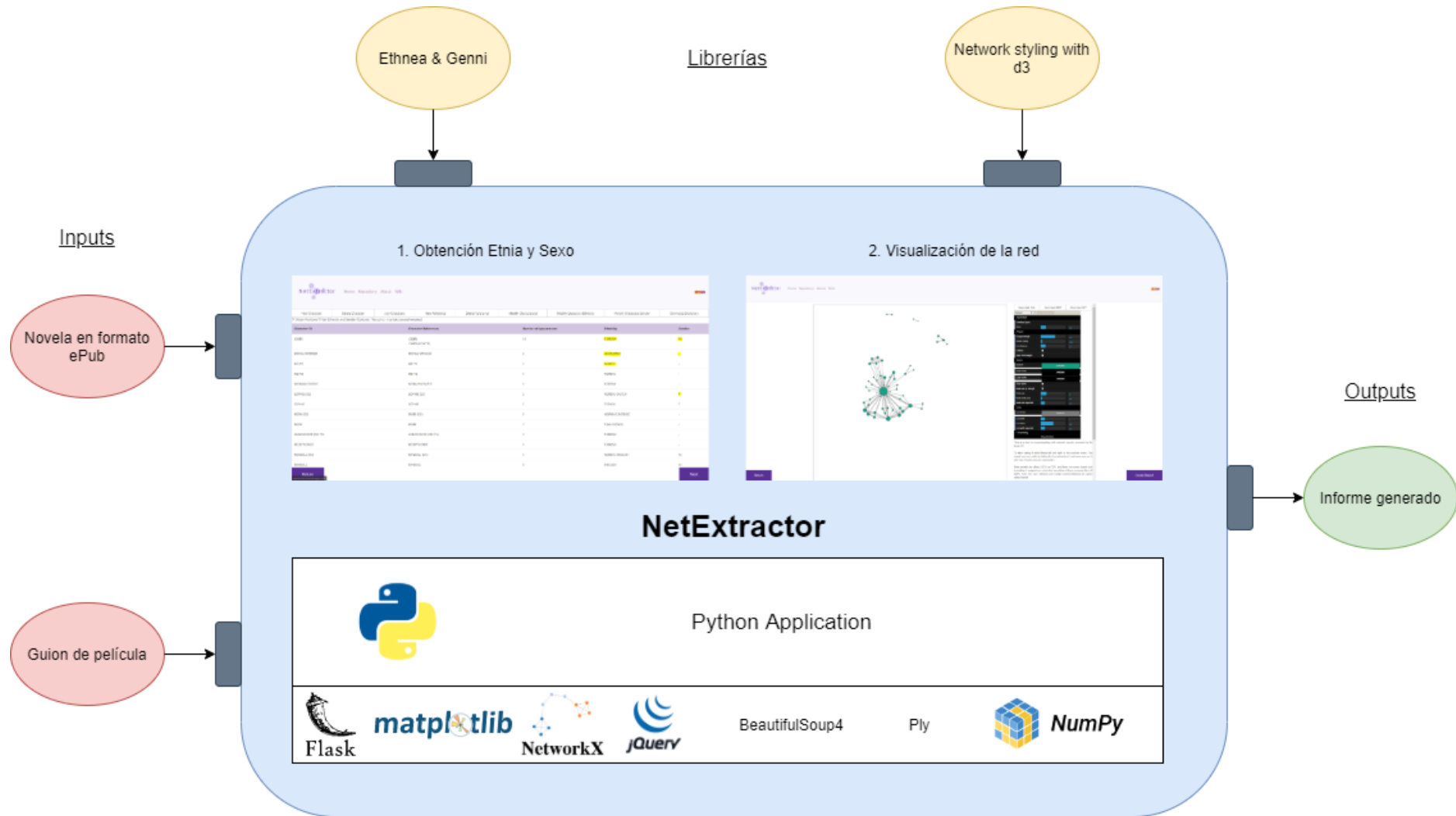


Figura 13 Arquitectura de la aplicación

Diagrama de paquetes

Los paquetes de la aplicación tienen la siguiente estructura, desde el paquete src hasta cada una de las subcarpetas que lo componen:

- src: contiene todos los paquetes del proyecto. Además, posee dos archivos adicionales que se corresponden con el controlador y con el fichero de configuración de Flask Babel.
- src/Modelo: contiene los dos ficheros de Python donde se encuentra la lógica de la aplicación, estos ficheros se corresponden con el modelo y con la clase personaje.
- src/LecturaFicheros: contiene los archivos para la lectura de ePubs y de csv. También contiene archivos para la escritura de los csv cuando el diccionario es exportado.
- src/Lexers: contiene cada uno de los analizadores léxicos empleados para obtener las posiciones de los personajes y para crear el diccionario de personajes a partir de un ePub introducido.
- src/PersistenciaSesiones: contiene la lógica de almacenamiento de sesiones del usuario. Es decir, la id de la sesión y el objeto modelo.
- src/static: contiene todos los archivos que han sido empleados por las plantillas html, es decir, las hojas de estilo, imágenes y archivos javascript.
- src/static/css: contiene las hojas de estilo que se han usado en las plantillas html.
- src/static/img: contiene las imágenes que han sido usadas en las plantillas html.
- src/static/js: contiene los ficheros javascript que se han empleado en las plantillas html.
- src/translations: contiene el catálogo de traducciones de la aplicación.
- src/templates: contiene las plantillas html que han sido usadas en el proyecto.
- src/guiones: contiene el archivo de Python para la obtención del diccionario de personajes al obtener un guion.
- src/PredictorEtniaSexo: contiene el archivo Python que implementa el predictor de etnia y sexo de Ethnea y Genni(3).

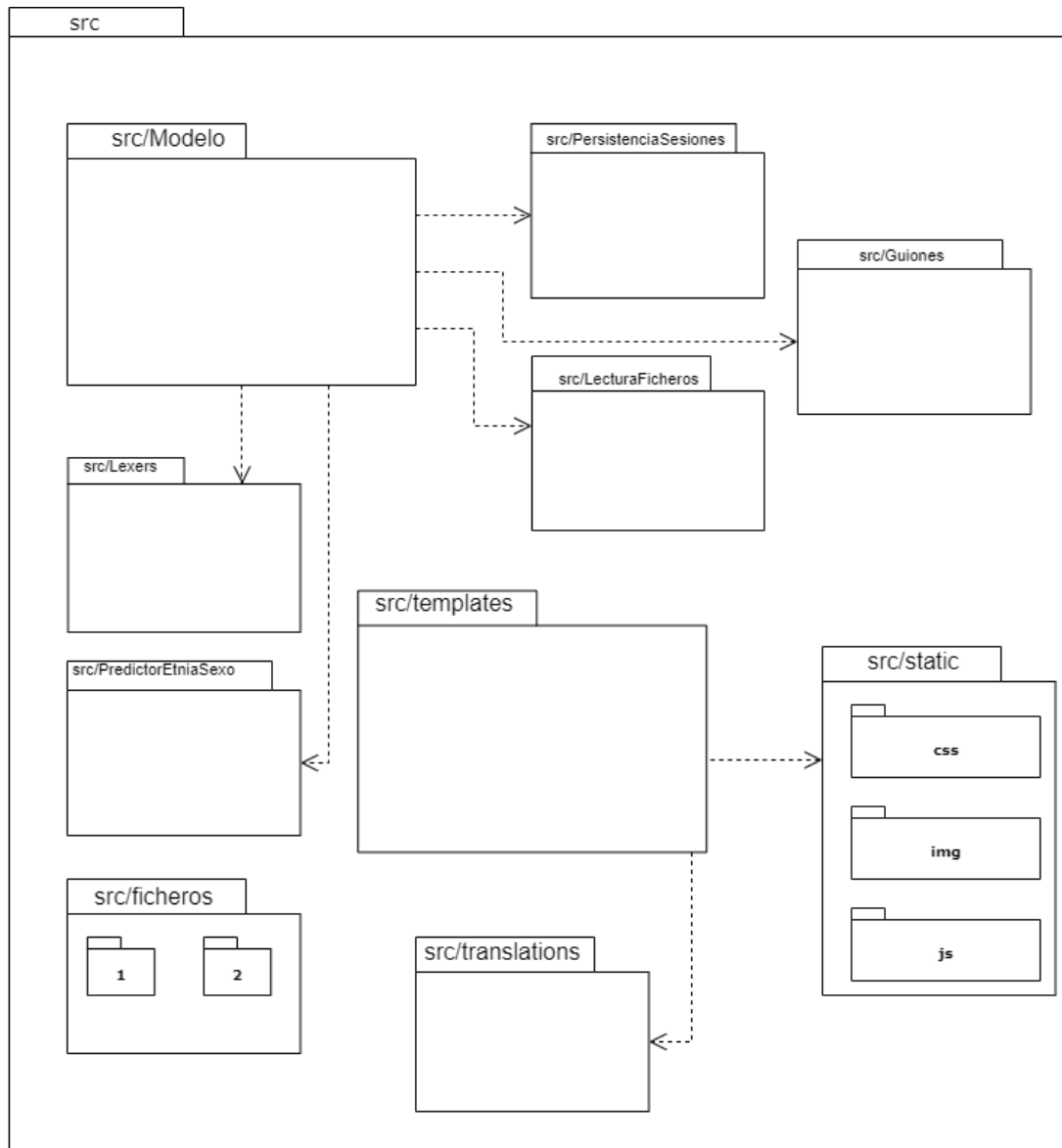


Figura 14 Diagrama de paquetes

Diagrama de clases

En este apartado se va a mostrar el diagrama de clases generado en este proyecto, se mostrará la estructura del sistema y sus clases y atributos, dejando claro las relaciones que se establecen entre ellos. En este diagrama no se van a mostrar los métodos que existen en ellos, sino que se describirán a continuación de forma individual debido a la cantidad de métodos que poseen algunas clases.

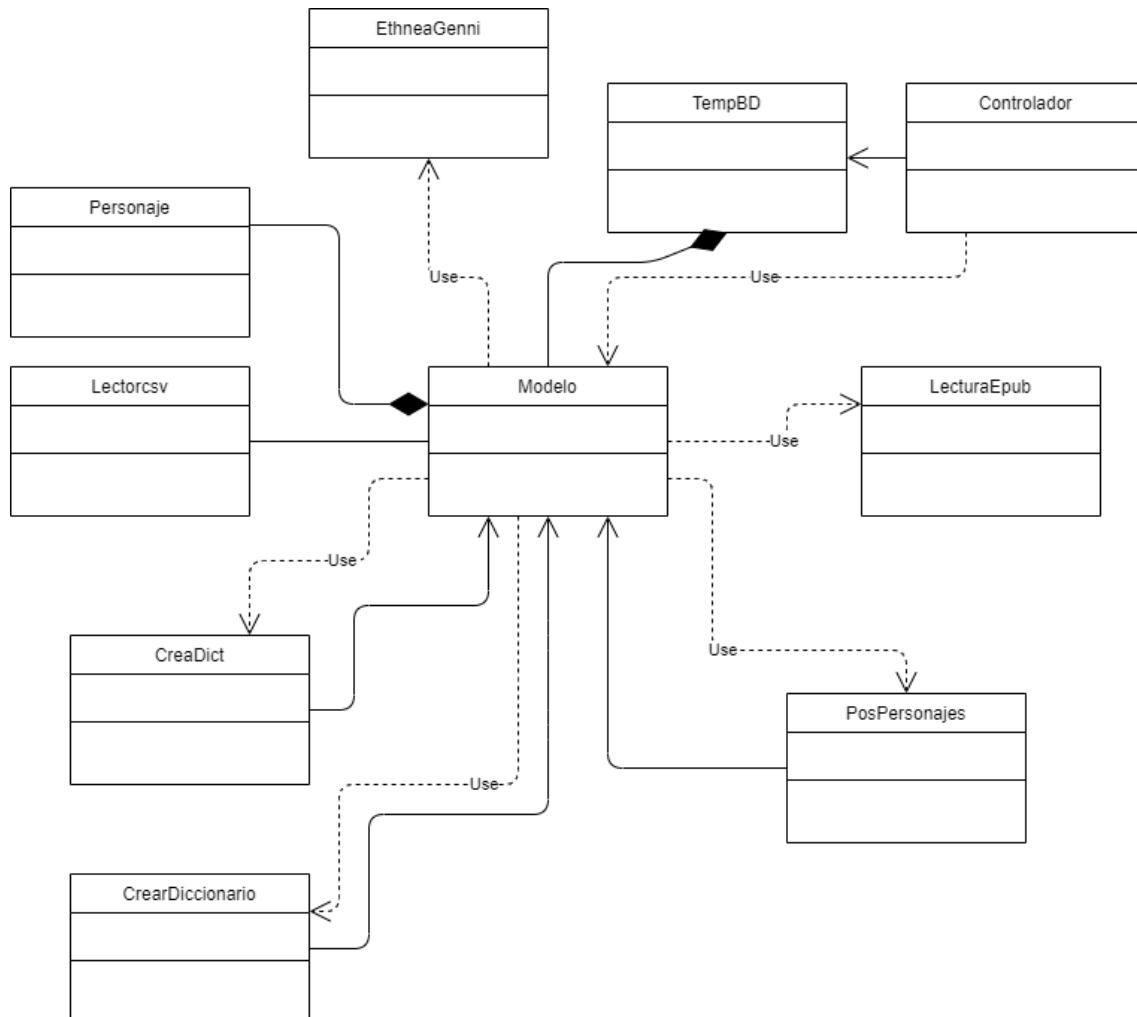


Figura 15 Diagrama de clases

A continuación, se van a describir cada uno de los tipos de relaciones que se van a dar en el siguiente diagrama.

- **Composición:** es la relación que se marca con un diamante relleno. Esta relación indica un todo y cada una de sus partes. Por ejemplo, en nuestro proyecto la clase **Personaje** es una parte de la clase **Modelo**.
- **Asociación:** es la relación marcada con una línea. Una asociación indica la relación entre dos clases de forma estática. La clase **Modelo** se asocia con la clase **Lectorcsv**.
- **Dependencia:** es la relación marcada con una línea discontinua y corresponde con una relación cliente-proveedor, en la cual el cliente usa los servicios del proveedor. Por ejemplo, la clase **Modelo**, usa los servicios de la clase **LecturaEpub**.

- Asociación directa: es la relación marcada con una flecha. Esta relación denota navegación en la dirección hacia la que va la flecha. Por ejemplo, la clase CreaDict y Modelo tienen una asociación directa ya que poseen una relación en la que Modelo llama a CreaDict, por tanto, la dirección de navegación va de CreaDict a Modelo.

Como no cabían todas las variables y métodos en cada una de las clases, se van a poner a continuación todas las clases detalladas más a fondo:

Controlador	CreaDict
+app:Flask +babel:Babel +tbd:TempDB	+lexer:lex +aux:dict() +mod:Modelo
+get:locale():session['lang'] +before_request() +home() +inicio() +formatoIncorrecto() +diccionarioPelicula() +index() +about() +dictaut() +impdict() +obtdict() +moddict() +newpers() +etnipers() +sexopers() +delpers() +joinpers() +newrefpers() +delrefpers() +modidpers() +params() +paramsPeliculas() +red() +informe() +visinforme() +download_file(string filename) +idioma() +guardarConfig() +finSesion()	+__init__(Modelo modusuario) +t_PERSONAJE(token t):token +t_ESPACIOS(token t) +t_PUNTO(token t) +t_CHARACTER(token t) +t_punto_OTRO(token t) +t_punto_ESPACIOS(token t) +t_punto_error(token t) +t_error(token t) +crearDict(string texto)

Figura 16 Clase CreaDict

Lectorcsv
-modelo:Modelo
+__init__(Modelo m) +importDict(string fichero) +exportDict(string fichero)

Figura 18 Clase Lectorcsv

Figura 20 Clase controlador

LecturaEpub	TempBD
+fich: string +epub:zip -orden:list()	-instance:self -sesiones:dict() -nextID:int
+__init__(string fichero) -obtenerOrdenLectura() +siguienteArchivo():string	+__init__() +getInstance():TempBD.__instance +addSesion(Modelo sesionObject):int +delSesion(int sesionID) +replaceObject(int sesionID, Modelo sesionObject) +getObject(int sesionID):Modelo +getSesiones():dict()

Figura 19 Clase LecturaEpub

Figura 17 Clase TempBD

Figura 21 Clase Modelo

Modelo
<pre> -csv:Lectorcsv -texto: list() +personajes: dict() -fincaps: list() -G: nx. Graph() -Gnoatt: nx. Graph() +urlPelícula: string +diccionarioApariciones: dict() +cambio: int +formato: int +__init__() +cambiaPantallas(int cambiopantalla) +getFormato():int +devolverCambio():int +crearDict(): +hayPersonajes():int +obtenerPosPers() +obtenerNumApariciones():dict() +obtenerPosPers() +obtenerEthnea() +getDictParsear():list() +getPersonajes():dict() +vaciarDiccionario():dict() +cambiarEtnia(string etnia,string pers) +cambiarSexo(string sexo,string pers) +anadirPersonaje(string idpers,string pers):string +__eliminarPersonaje(string idPersonaje) +eliminarListPersonajes(list() personajes) +__juntarPersonajes(string idPersonaje1,string idPersonaje2) +juntarListPersonajes(list() lista) +anadirReferenciaPersonaje(string idp, string ref) +__eliminarReferenciaPersonaje(string idp, string ref) +eliminarListRefs(list() lista) +modificarIdPersonaje(string idact, string newid) +juntarPosiciones() +prepararRed() +prepararRedPelículas() +getMatrizAdyacencia():SciPy parse matrix +generarGrafo(int rango, int minapar, boolean caps) +elementosComunes(list() lista, list() lista1):list() +obtenerRed(int apar) +anadirAtributos() +visualizar(): json +scrapeWiki(string url) +scrapeWikiPelícula(string url) +importarDict(string fichero) +exportDict(string fichero) +obtTextoEpub(string fich) +esEpub(string fich): Boolean +exportGML(string filename) +exportPajek(string filename) +writeFile(string filename,string text) +generarInforme(list() solicitud,string direc) +nNodos():int +nEnl(): int +nInt():int +gSin():int +gCon():int +dSin():distribución de grado +dCon():distribucion de grado +dens():nx.density +conComp():list() +exc():dict() +diam():nx.diameter +rad():nx.radius +lonMed():nx.average_shortest_path_length +locClust():nx.clustering +clust():nx.average_clustering +trans():nx.transitivity +centG():nx.degree_centrality +centC():nx.closeness_centrality +centI():nx.betweenness_centrality +ranWal: nx.current_flow_betweenness_centrality +centV():nx.eigenvector_centrality +paRa():nx.pagerank_numpy +ordenarFrozen(partition):list() +louvain():list() +greedyComunidad():list() +kCliPerc(int k):list() +girNew():list() +roles(nx.Graph() resul, string nombre):dict() +rolesLouvain():dict() +rolesGreedy():dict() +roleskclique(int k):dict() +rolesGirvan():dict() +devuelveComunidadesSeparadas(list() resultado, nx.Graph() grafo):nx.Graph() +obtenerZ(nx.Graph() grafo, nx.Graph() resul):dict() +obtenerP(nx.Graph() grafo, nx.Graph() resul):dict() </pre>

Personaje
-nombres: dict() -pospers: dict() -numapar: int +lenombres: dict() -etnia: string -sexo: string -dictSexEt: dict()
+__init__() +getDiccionario():dict() +crearDictSE() +setEtnia(String etnia) +setSexo(String sexo) +getEtnia():string +getSexo():string +getPersonaje(): dict() +getPosicionPers():dict() +setPosicionPers(pospers) +getNumApariciones():int, boolean +sumNumApariciones(apar) +resNumApariciones(apar)

Figura 24 Clase Personaje

PosPersonajes	CrearDiccionario
+lexer:lex +nombres:list() +m:Modelo	+mod:Modelo
+__init__(Modelo modelo) +t_PALABRA(token t) +t_CAPITULO(token t) +t_ESPACIO(token t) +t_coincidencia_PALABRA(token t) +t_error(token t) +t_coincidencia_error(token t) +obtenerPos(string texto, list() nombres) +esSubcadena(string st, list() lsita):list()	+__init__(Modelo modusuario) +obtenerPersPelicula(string url)

Figura 23 Clase CrearDiccionario

EthneaGenni
+replacements:list()
+__init__() +normalize(string s):string +separaNombres(string nombre):string, string +obtenerEtniaSexo(string nombre):string, string

Figura 22 Clase EthneaGenni

Diseño procedimental

En este apartado se muestran los detalles más relevantes de los algoritmos utilizados para la obtención de diccionario, apariciones de personajes y para la predicción de la etnia y sexo del personaje.

Los siguientes diagramas de secuencia representan la interacción de las clases de este proyecto cuando tienen lugar los algoritmos mencionados en el párrafo anterior.

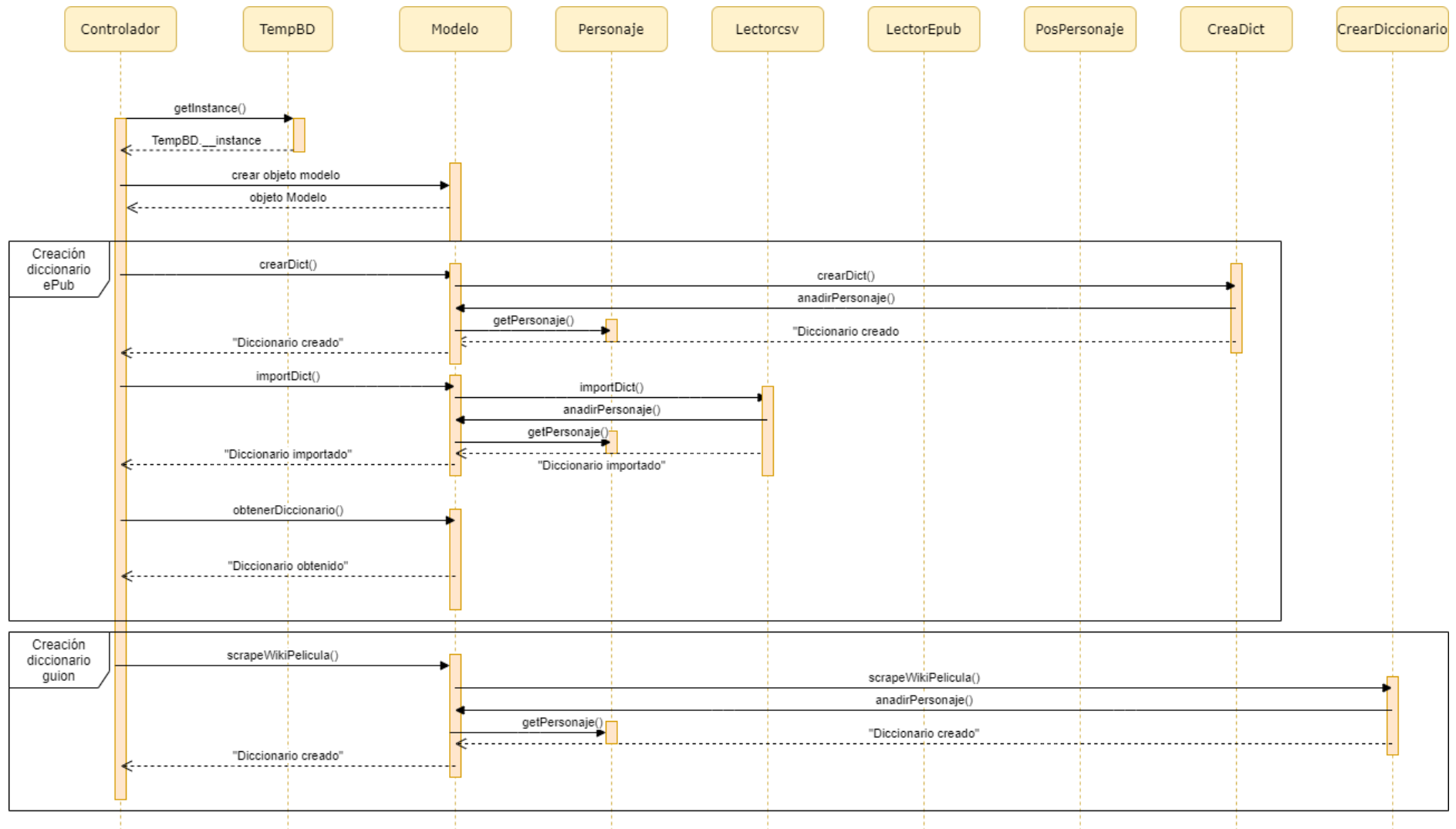


Figura 26 Diagrama de secuencia para los algoritmos de creación de diccionarios

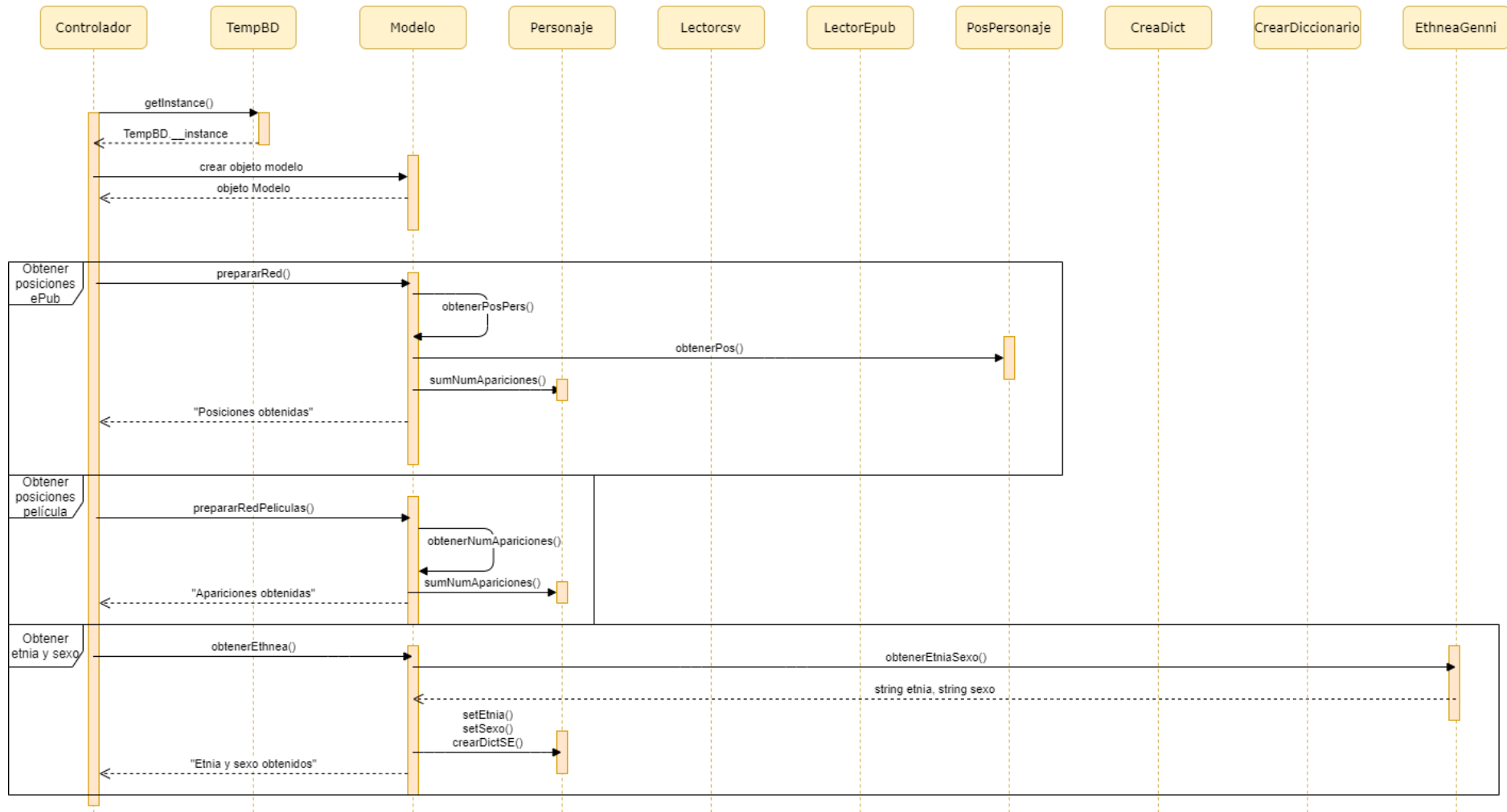


Figura 27 Diagrama de secuencia para los algoritmos de obtención de posiciones y predicción de etnia y sexo

3.4 Diseño de interfaces

Al empezar el proyecto, se decidió que la interfaz iba a ser una de las cosas que habría que cambiar, sustituirla por una más vistosa y que pueda ser más entendible al usuario.

Para ello se desarrollaron una serie de prototipos sobre algunas pantallas de la aplicación. Estos prototipos han sido realizados con la herramienta de Adobe XD(5), una herramienta que permite la realización de prototipos, tanto móviles como web. Permite además exportar los prototipos en distintos tipos de formato de imagen. Algunos de los prototipos que se realizaron fueron los siguientes:

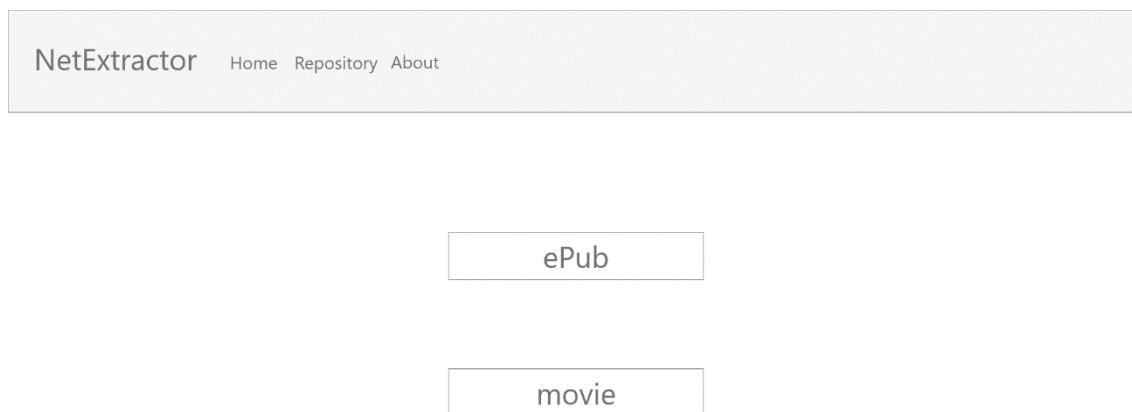


Figura 29 Prototipo selección

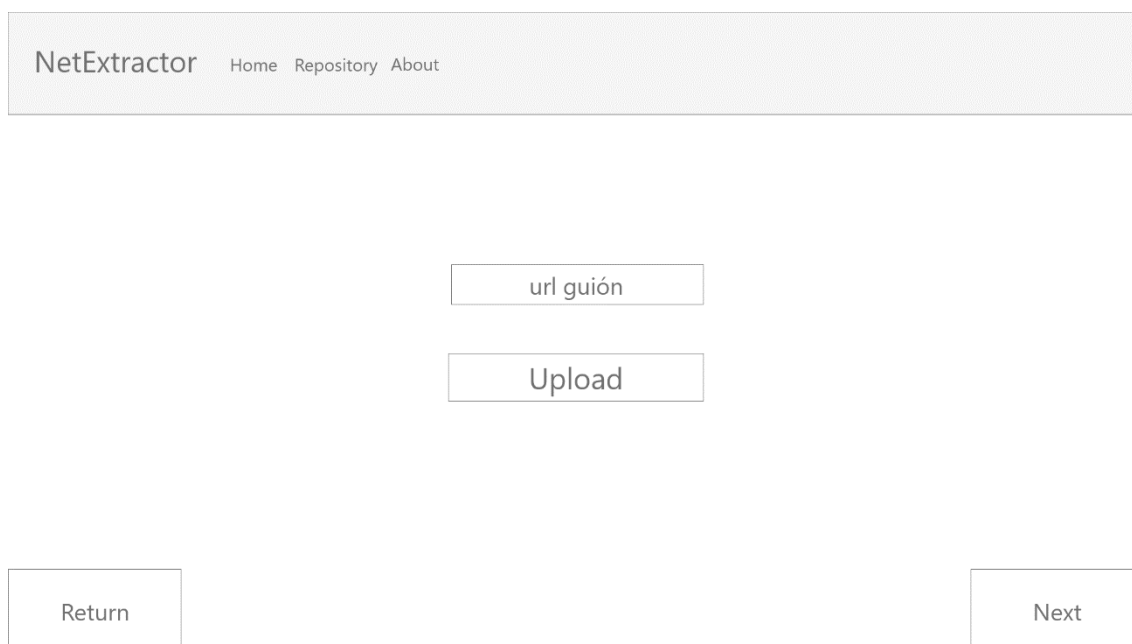


Figura 28 Prototipo introducir guion

NetExtractor					
Home Repository About					

Id Personaje	Referencias	Apariciones
PERSONAJES		

Return

Next

Figura 31 Prototipo modificar diccionario

NetExtractor	
Home Repository About	

VISUALIZACIÓN DE LA RED

MODIFICAR RED

Return

Next

Figura 30 Prototipo visualización de la red

NetExtractor

HomeRepositoryAbout

Características que extraer

☐ Característica

☐ Característica

☐ Característica

☐ Característica

☐ Característica

☐ Característica

☐ Característica

☐ Característica

☐ Característica

☐ Característica

☐ Característica

☐ Característica

☐ Característica

☐ Característica

☐ Característica

☐ Característica

Return

Next

Figura 32 Prototipo extracción características

NetExtractor

HomeRepositoryAbout

Navigation menu

Apartado 1

Apartado 2

Apartado 3

Apartado 4

Report

Apartado 1

Texto donde sale la información que tiene que reflejarse en el informe.

Gráfica

Apartado 2

Return

Figura 33 Prototipo informe

Estos fueron los primeros prototipos, una ventaja de su realización es que al ser una evolución de un proyecto anterior (Ububooknet), ya existía una interfaz anterior, por lo tanto, se sabía como podría desarrollarse la nueva interfaz conociendo la anterior.

Al final, a lo largo del proyecto, se fueron cambiando los colores y diseños de la interfaz hasta conseguir los finales que son los siguientes:

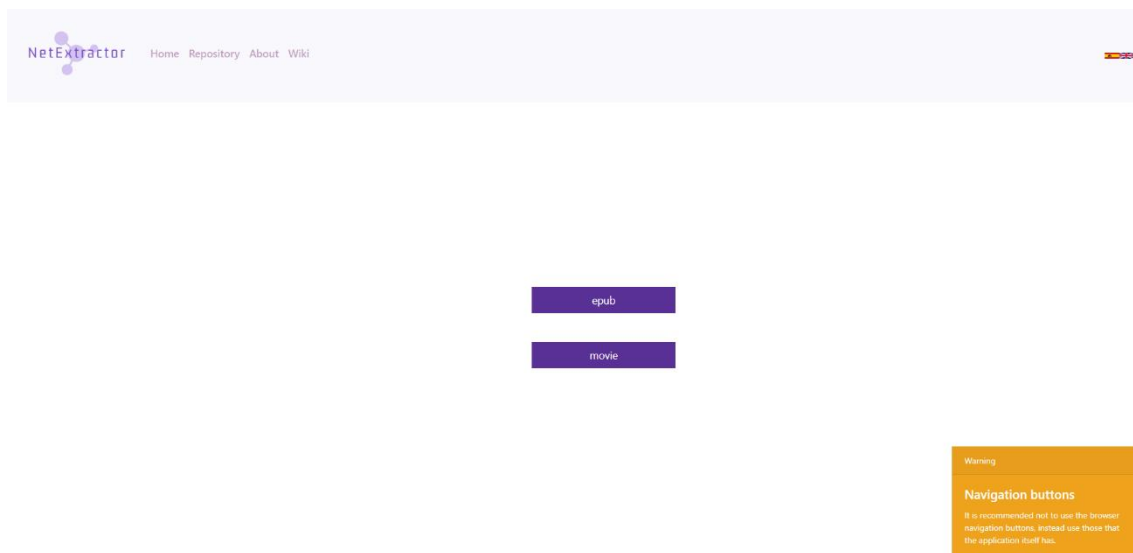


Figura 34 Seleccion final

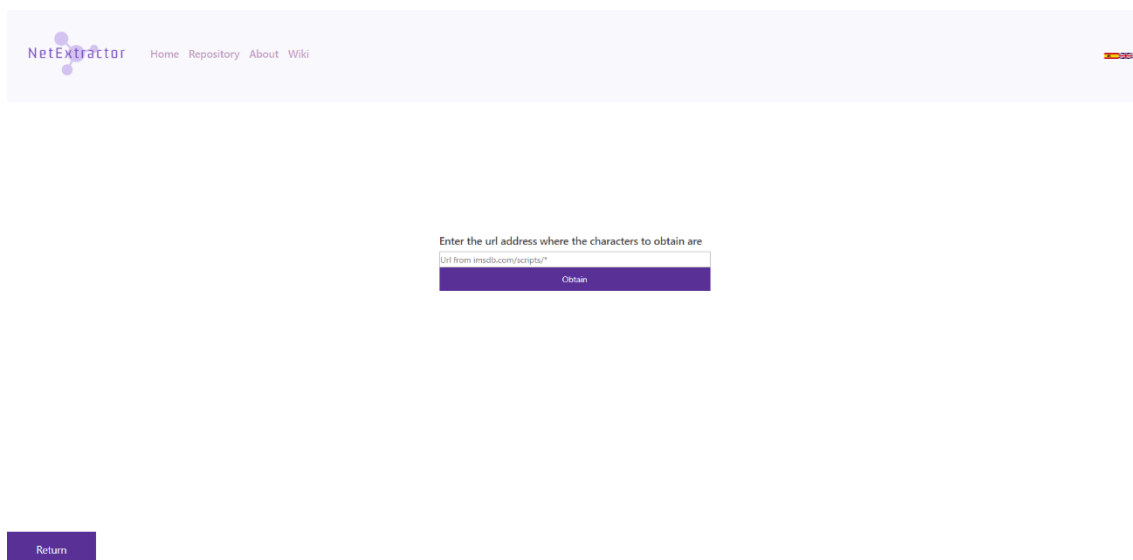




Figura 35 Introducir guion final


[Home](#) [Repository](#) [About](#) [Wiki](#)


Report data extraction

Basic metrics

- ☐ Number of Nodes
- ☐ Number of Edges
- ☐ Number of Interactions

Degree metrics

- ☐ Nodes Degree Unweighted
- ☐ Nodes Degree Weighted
- ☐ Degree Distribution Unweighted
- ☐ Degree Distribution Weighted
- ☐ Density

Distance Metrics

- ☐ Connected Components
- ☐ Eccentricity
- ☐ Diameter
- ☐ Radius
- ☐ Average Distance

Clustering Metrics

- ☐ Local Clustering Coefficient
- ☐ Average Local Clustering Coefficient
- ☐ Transitivity

Centrality Metrics

- ☐ Degree Centrality
- ☐ Closeness Centrality
- ☐ Betweenness Centrality
- ☐ Random Path Betweenness Centrality
- ☐ Eigenvector Centrality
- ☐ Pagerank

Communities and groups

- ☐ K-clique Communities

K-clique k value for communities: (Default will be 2)

- ☐ Girvan-Newman Communities
- ☐ Clauset-Newman-Moore Communities
- ☐ Louvain communities

Role Detection



- ☐ K-clique roles

K-clique k value for roles: (Default will be 2)

- ☐ Girvan-Newman roles
- ☐ Clauset-Newman-Moore roles
- ☐ Louvain roles

[Return](#)
[Create Report](#)

Figura 39 Extracción de datos final


[Home](#) [Repository](#) [About](#) [Wiki](#)


Report

Navigation Menu

- Role Detection**
- K-clique roles
- Girvan-Newman roles
- Clauset-Newman-Moore roles
- Louvain roles

Role Detection

K-clique roles:

Non Ultra Peripheral Hubs (R1):

SOCIAL WORKER | (MORE) | JOKER (CONT'D) | WOMAN ON BUS | SOPHIE (OS) | SOPHIE | MOM (OS) | MOM | ANNOUNCER (ON TV) | RECEPTIONIST | RANDALL (OS) | RANDALL | GARY | HOYT | BRANCH MANAGER | HOYT (OVER PHONE) | WALL STREET #1 | WALL STREET #2 | WALL STREET #3 | YOUNG WOMAN | BOBBY SHORT (SINGING) | THOMAS WAYNE (ON TV) | EMCEE (OS) | MURRAY FRANKLIN (ON TV) | MOM (OS) (CONT'D) | HA-HA CLOWN #1 (OS) | HA-HA CLOWN #2 | MAGICIAN | RANDALL (CONT'D) | 53 | ALFRED (OS) | ALFRED | ER NURSE | ER RESIDENT | YOUNG DOCTOR (OS) | NURSE #1 (OS) | THOMAS WAYNE | 77 | CLERK | CLERK (CONT'D) | SHOW BOOKER (ON MACHINE) | SHOW BOOKER (OVER PHONE) | THE FUCK-- | GARY (OS) | GENE FRANKLIN (ON MONITORS) | POLICE OFFICER #1 (OS) | POLICE OFFICER #2 (OS) |

Non Peripheral Hubs (R2):

Non Hub Connectors (R3):

Non Hub Kinless Nodes (R4):

Provincial Hubs (R5):

JOKER |

Connector Hubs (R6):

Kinless Hubs (R7):

Isolated Nodes:

JOKER (PRE-LAP) | 65 | MOM (VO) | JOKER (OS) | BURKE | 103 | PUNK |

[Return](#)
[Navigation Menu](#)

Figura 38 Informe final

Apéndice D. Documentación técnica de programación

4.1 Introducción

En este apartado se va a describir toda la estructura de archivos que tenemos en el repositorio del proyecto y que serán necesarios si vamos a desplegar la aplicación. También se van a describir todos los aspectos técnicos de la aplicación y cómo se debe instalar todos los archivos para poder lanzar la aplicación tanto en local como en el servidor de Heroku.

También se mostrará cómo crear la wiki en Wikidot y aunque no se haya implementado finalmente, se mostrará cómo desplegar MediaWiki en Heroku también.

4.2 Estructura de directorios

En este proyecto existen tres repositorios que hemos empleado durante su realización.

- Repositorio principal: es donde se aloja todo el proyecto y por tanto todos los archivos que lo componen.
- Repositorio despliegue: como ya se mencionó en la memoria, el *main* del despliegue difiere un poco del *main* necesario para su ejecución en local, por tanto, se creó otro repositorio para albergar el despliegue de la aplicación.
- Repositorio MediaWiki: aunque no se llegó a emplear MediaWiki para la realización de la wiki debido a problemas aparecidos mencionados en la memoria, sí que se llegó a desplegar MediaWiki en Heroku, por tanto, este repositorio también figurará en el proyecto.

Repositorio principal

Se mostrará a continuación la estructura de directorios que vamos a tener en el repositorio principal de [GitHub](#).

- /Imágenes Wiki/: Carpeta donde están contenidas las imágenes que van a ser usadas en la wiki de la aplicación.
- /ReadMeImágenes/: Carpeta para guardar las imágenes usadas en el *readme* del repositorio.
- /doc/: Carpeta que contiene toda la documentación del proyecto.
- /doc/Imágenes/: Carpeta que contiene las imágenes que se han empleado en la documentación.
- /src/: Carpeta que contiene todos los archivos necesarios para el funcionamiento del proyecto.
- /src/LecturaFicheros/: Carpeta que contiene los ficheros que van a ser usados para leer los distintos archivos subidos a la aplicación (ePub, csv importados...).
- /src/Lexers/: Carpeta que contiene todos los archivos necesarios para realizar los distintos análisis léxicos a lo largo de la aplicación.
- /src/Modelo/: Carpeta que contiene dos ficheros, uno llamado *Modelo.py* que contiene todas las funcionalidades que posee la aplicación; y otro llamado *Personaje.py* que contiene la clase con la lógica que posee cada personaje detectado.

- /src/PersistenciaSesiones/: Carpeta que contiene las sesiones de usuario, la información perteneciente a cada usuario será eliminada una vez que la aplicación se cierre.
- /src/ficheros/: Carpeta que contiene los ficheros empleados a lo largo del transcurso de la aplicación, en esta carpeta se guardan, por ejemplo, las imágenes usadas cuando se genera el informe.
- /src/Guiones: Carpeta que contiene el fichero de obtención de personajes para un guion.
- /src/PredictorEtniaSexo/: Carpeta que contiene el fichero que usa Ethnea y Genni(3) para predecir la etnia y el sexo.
- /src/static/: Carpeta que contiene ficheros pertenecientes a la web de la aplicación.
- /src/static/img: Carpeta que contiene las imágenes que van a ser usadas en la aplicación, estas se corresponden con las banderas del idioma, el logo de la aplicación, escudos de universidad y politécnica y gif de pantalla de carga.
- /src/static/css: Carpeta que contiene el css para la interfaz de la web.
- /src/static/js: Carpeta que contiene ficheros js para poder realizar ciertas funcionalidades de la web.
- /src/templates/: Carpeta que contiene todos los ficheros html de cada pantalla que existe en la aplicación.
- /src/translations/: Carpeta que contiene todos los ficheros de traducciones que tiene la aplicación.
- /src/Controlador.py/: Es el fichero que conecta el modelo con la vista de la aplicación en la arquitectura MVP.
- /src/Config.py/: Es el fichero de configuración empleado por Flask y Babel.
- /LICENSE: Es el fichero con la licencia del proyecto.
- /README.md: Es el fichero del *readme* del repositorio.
- /babel.cfg: Es un fichero para la configuración de babel.
- /requirements.txt: Es el fichero utilizado para poder instalar todos los paquetes y herramientas necesarios para este proyecto.
- /main.py: Es el fichero con el cual se va a poder ejecutar el proyecto.
- /tst/: Carpeta que contiene una clase de pruebas unitarias.

Repositorio despliegue

En este apartado se va a describir la estructura de directorios del repositorio despliegue, muchos de los ficheros son iguales a los mencionados en el apartado anterior, con lo cual, dichos archivos no se volverán a definir de nuevo. Esta estructura se puede encontrar en el repositorio de GitHub correspondiente para el despliegue [aquí](#).

- /src/: Misma estructura que el apartado anterior.
- /main.py: Distinto al del apartado anterior con el fin de que pueda ser leído por Heroku a la hora de desplegar la aplicación.
- /README.md: Mismo que el apartado anterior.
- /requirements.txt: Fichero necesario para que Heroku pueda leer lo que necesita instalar, mismo que en el apartado anterior.
- /Procfile: Fichero necesario para que Heroku lea como se debe ejecutar la aplicación.
- /runtime.txt: Fichero para que Heroku detecte el lenguaje de programación que se emplea en la aplicación.

- /requirements.txt: Mismo que en el apartado anterior, también empleado por Heroku para poder instalar todas las herramientas necesarias.
- /LICENSE: Mismo documento que en el apartado anterior.
- /babel.cfg: Mismo documento que en el apartado anterior.

Repositorio MediaWiki

En este repositorio se encuentran todos los ficheros que se obtienen cuando descargamos MediaWiki de la página oficial [aquí](#). Además contiene un fichero composer.lock necesario para que Heroku pueda leer la aplicación que estamos desplegando.

Para generar este fichero, se tuvo que descargar MediaWiki en una imagen de Linux, en este caso Ubuntu con el fin de instalar composer, ya que su instalación en Linux ofrecía menos resistencias. Para instalar composer podemos seguir el siguiente tutorial:

- [Cómo instalar y utilizar Composer en Ubuntu 18.04](#)(6).

Una vez instalado, simplemente debemos movernos a la carpeta donde se encuentran todos los ficheros descargados de MediaWiki, abrir una terminal en esta carpeta y escribir el siguiente comando:

```
“$ composer update”
```

Cuando hagamos esto, se nos habrá creado el fichero composer.lock, con esto ya tendríamos todos los ficheros necesarios para su despliegue.

4.3 Manual de programador

En este apartado se van a describir los procesos de instalación de la aplicación, tanto en local como en remoto, también se mostrará cómo se ha realizado la internacionalización de la aplicación y la wiki correspondiente, tanto la de Wikidot, como la de MediaWiki.

Ejecución en local

Para realizar la ejecución de la aplicación en local, debemos primero descargarnos el proyecto que está alojado en el [repositorio principal](#) mencionado anteriormente.

Una vez que lo hemos descargado, lo primero que se debe hacer es descargar cada una de las herramientas que van a ser usadas. Para ello simplemente debemos situarnos en la carpeta del proyecto y ejecutar el siguiente comando:

- pip install -r requirements.txt

Componentes instalados

Estos son los componentes que van a ser instalados al ejecutar el comando anterior:

- Flask: v1.0
- Flask Babel: v0.12
- Matplotlib: v3.0
- Numpy: v1.15
- Ply: v3.11
- BeautifulSoup4: v4.7
- lxml: v4.3
- html5lib: v1.0
- Networkx: v2.2
- Scipy: v1.1

- Python-louvain: versión sin especificar.
- Werkzeug: v0.16.1

Para poder ejecutar nuestra aplicación también debemos tener instalado Python en su versión 3.7.4.

Una vez que se hayan instalado los anteriores componentes ya podemos ejecutar la aplicación, para ello, tenemos que movernos hasta la carpeta donde se encuentra el fichero “main.py” y ejecutar el siguiente comando:

- “python main.py”

Cuando hayamos realizado esto nos aparecerá una serie de líneas indicando que se está ejecutando la aplicación en local, pueden darse dos casos distintos dependiendo de si estemos en modo debug o no, para entrar en modo debug debemos simplemente abrir el main.py con un editor de texto y poner la variable marcada a continuación como “True”.

```
# -*- coding: utf-8 -*-

from tst import testUnitarios
from src import Controlador
import unittest

#runner = unittest.TextTestRunner()
#result = runner.run(unittest.makeSuite(testUnitarios.testUnitarios))

Controlador.app.run(threaded=True, debug=True)
```

Figura 40 Main.py

Con este *main* obtendremos lo siguiente cuando ejecutemos la aplicación.

```
* Serving Flask app "src.Controlador" (lazy loading)
* Environment: production
  WARNING: Do not use the development server in a production environment.
  Use a production WSGI server instead.
* Debug mode: on
* Restarting with stat
* Debugger is active!
* Debugger PIN: 223-373-641
* Running on http://127.0.0.1:5000/ (Press CTRL+C to quit)
```

Figura 41 Debug Mode On

Por el contrario, si no queremos activar el “*debug mode*” simplemente debemos poner la variable marcada con anterioridad a “*False*”, con lo que obtendremos las siguientes líneas.

```
* Serving Flask app "src.Controlador" (lazy loading)
* Environment: production
  WARNING: Do not use the development server in a production environment.
  Use a production WSGI server instead.
* Debug mode: off
* Running on http://127.0.0.1:5000/ (Press CTRL+C to quit)
```

Figura 42 Debug Mode Off

Con esto ya podríamos utilizar la aplicación en nuestro dispositivo de forma local simplemente abriendo nuestro navegador y accediendo a la dirección marcada en la anterior imagen (<http://127.0.0.1:5000/>).

Desplegar aplicación en remoto con Heroku

Para desplegar la aplicación en Heroku se puede hacer de dos formas distintas, por la línea de comandos, de forma que se puede visitar un tutorial con el cual desplegar la aplicación.

- [Getting Started on Heroku](#).

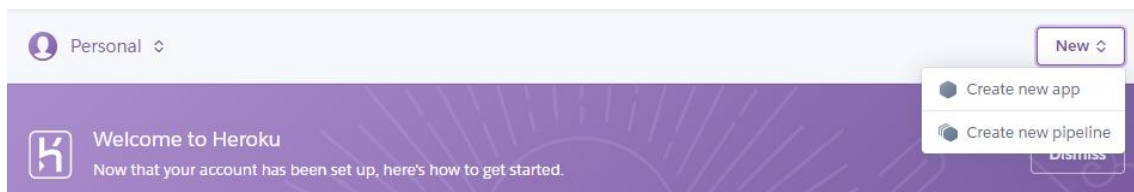
En esta página podemos seleccionar el lenguaje de programación que queramos, y desplegar nuestra aplicación, en nuestro caso seleccionaríamos Python.

Aquí se explicará cómo realizar el despliegue a través de la web de Heroku, ya que es el método que se ha empleado para el despliegue.

Lo primero que debemos saber antes de explicar el despliegue es la necesidad de tener en nuestro proyecto tres ficheros, que van a ser necesarios a la hora de que Heroku lea nuestra aplicación y sea capaz de desplegarla:

- Requirements.txt: Es el fichero donde tendremos que especificar todas las herramientas que van a tener que ser utilizadas e instaladas, de esta forma, en el despliegue Heroku lee este fichero e instala los requisitos necesarios.
- Runtime.txt: Es el fichero en el que se especifica el lenguaje de programación empleado y la versión en la que se está utilizando.
- Procfile: Es el fichero donde se encuentra el comando que Heroku tendrá que utilizar para ejecutar la aplicación.

Una vez tenemos nuestro proyecto en GitHub con los tres ficheros mencionados anteriormente creados, debemos crearnos una cuenta gratuita en Heroku, para ello simplemente tenemos que acceder a la página oficial de [Heroku](#). Cuando nos hayamos *logueado*, en la parte superior de la pantalla nos aparecerá lo siguiente:



Debemos seleccionar donde dice “Create new app”, entonces, nos aparecerá otra pantalla en la cual nos permite seleccionar el nombre que queremos darle a nuestra aplicación y donde queremos que se encuentre.

Cuando pulsemos en “Create app” tendremos ya la aplicación desplegada y nos aparecerá una pantalla parecida a la siguiente:

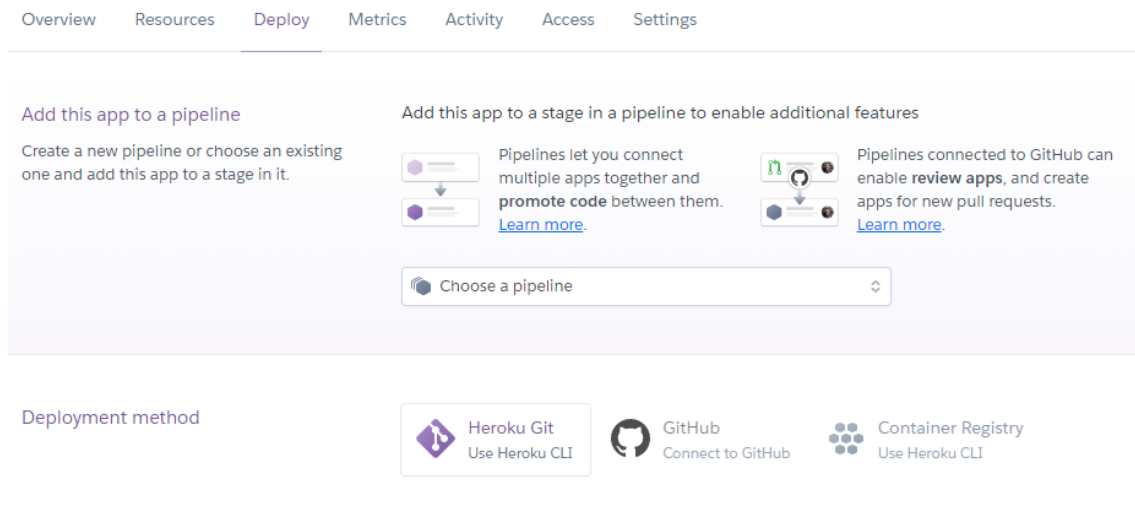


Figura 45 Enlazar GitHub

Figura 43 Crear nueva aplicación

Para que nuestra aplicación se corresponda con la que tenemos en nuestro repositorio de

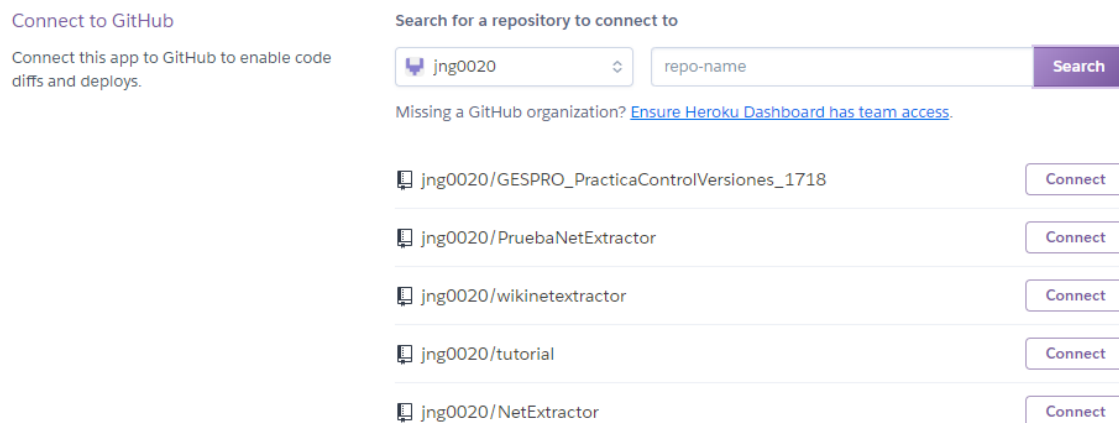


Figura 44 Seleccionar repositorio

GitHub simplemente tenemos seleccionar en el apartado “Deployment method” mostrado en la imagen la opción de GitHub, de esta forma nos pedirá que iniciemos la sesión de nuestra cuenta de GitHub. Una vez que lo tengamos debemos seleccionar el repositorio donde tenemos la aplicación haciendo clic en “Connect” en el repositorio correspondiente.

Una vez hecho esto ya tendremos nuestra aplicación seleccionada, simplemente debemos hacer el despliegue, para ello debemos ir hasta la parte inferior de la web donde nos encontraremos lo siguiente, podemos establecer que si hay cambios en el repositorio se cambie de forma automática en la aplicación habilitando lo despliegues automáticos en la primera opción “Enable Automatic Deploys”. En el apartado de “Manual deploy” debemos de establecer qué rama del

repositorio queremos desplegar y pulsar en “*Deploy Branch*”, después tenemos que esperar a que todo cargue correctamente y podremos acceder ya a nuestra aplicación de forma remota.

Automatic deploys

Enables a chosen branch to be automatically deployed to this app.

Enable automatic deploys from GitHub

Every push to the branch you specify here will deploy a new version of this app. **Deploys happen automatically:** be sure that this branch is always in a deployable state and any tests have passed before you push. [Learn more.](#)

Choose a branch to deploy

master

☐ Wait for CI to pass before deploy

Only enable this option if you have a Continuous Integration service configured on your repo.

Enable Automatic Deploys

Manual deploy

Deploy the current state of a branch to this app.

Deploy a GitHub branch

This will deploy the current state of the branch you specify below. [Learn more.](#)

Choose a branch to deploy

master

Deploy Branch

Figura 47 Despliegue

Receive code from GitHub

Build master 5db5a14c

Release phase

Deploy to Heroku

✓

✓

✓

✓

Your app was successfully deployed.

[View](#)

Figura 46 Despliegue correcto

Cuando todo haya cargado aparecerá algo como lo siguiente:

Internacionalización

Otro aspecto que tratar en este manual es la internacionalización de la aplicación, se puede actualizar el catálogo de traducciones de una manera muy cómoda realizando dos comandos en caso de que se hayan añadido nuevas palabras para traducir. Los textos que se van a traducir en nuestra aplicación van a ser todos aquellos que estén en la carpeta `/src/templates/`, en los archivos html que se encuentren de esta forma `{{ _('texto a traducir') }}`. Si se han añadido nuevas traducciones debemos ejecutar los siguientes comandos:

Cuando hayamos hecho esto podemos entrar en los catálogos de traducciones, por ejemplo en la traducción inglesa, podemos entrar en `/src/translations/en/LC_MESSAGES/` y acceder al fichero “messages.po” abriéndolo con un editor como Poedit, de tal forma que nos aparecerán dos columnas, una con el texto encontrado con el formato mencionado anteriormente (`{{ _('texto a traducir') }}`) y otra con la traducción que queramos hacer:

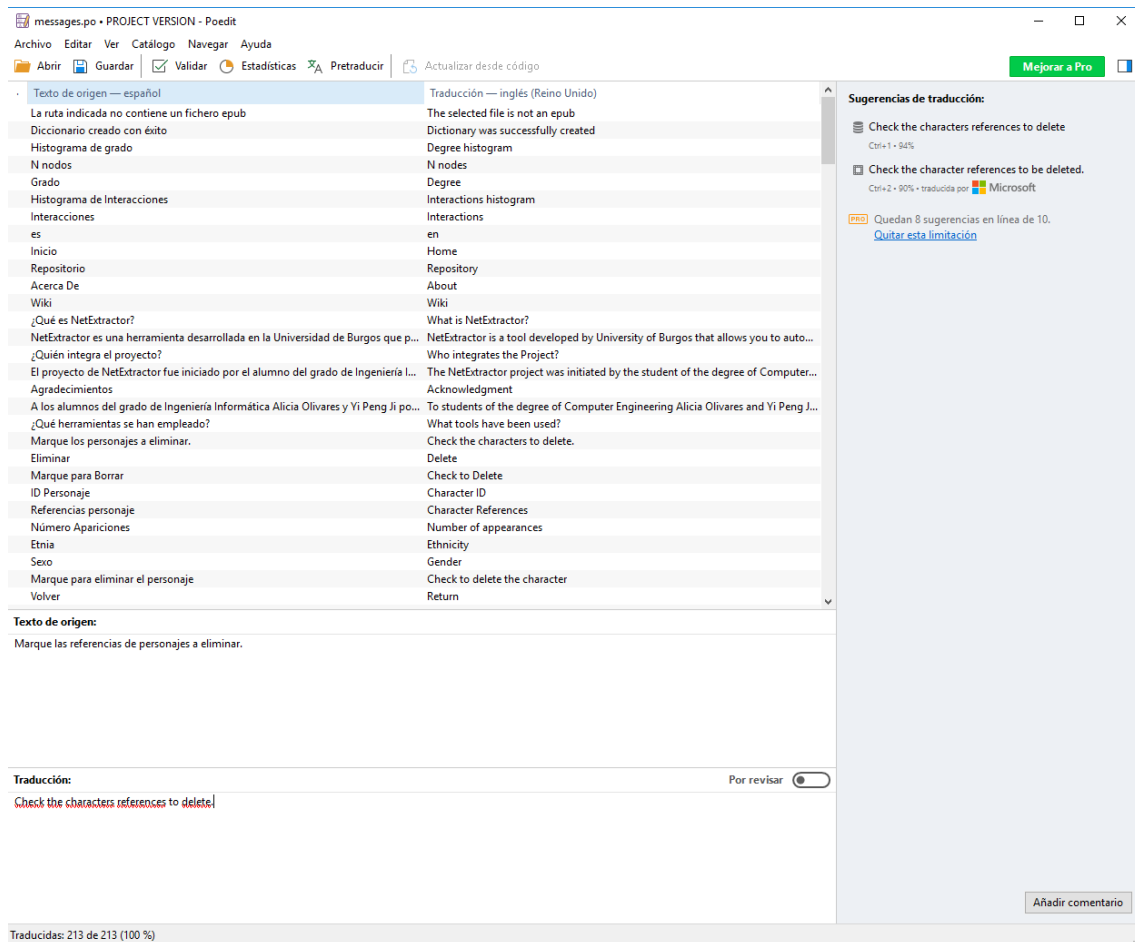


Figura 48 Traducciones poedit

Una vez establecidas las traducciones simplemente debemos guardar los cambios.

También se podría añadir nuevos idiomas para la aplicación web, bastaría con ejecutar un comando y cambiar un fichero que se encuentra dentro de la carpeta /src/. Pasos para añadir un nuevo idioma:

- Ejecutar el siguiente comando: “pybabel init -i messages.pot -d src/translations -l idioma
- Modificar el fichero “Config.py” que se encuentra dentro de la carpeta /src/ de tal manera que figure el nuevo idioma que se ha añadido.

```
# -*- coding: utf-8 -*-
import os
import secrets
#Idiomas permitidos en internacionalización
LANGUAGES = {
    'en': 'English',
    'es': 'Español'
}
#Carpeta donde subir archivos
upload_folder = os.path.join(os.path.dirname(os.path.realpath(__file__)), "ficheros")
#Carpera donde se guardan las traducciones
translations_folder = os.path.join(os.path.dirname(os.path.realpath(__file__)), "translations")
#Clave secreta
secretkey = secrets.token_urlsafe(16)
```

Figura 49 Añadir idioma

Wiki

Finalmente, aunque ya se ha comentado anteriormente que hay una MediaWiki desplegada en un repositorio, se ha elegido Wikidot como plataforma para albergar la wiki.

Para crear dicha wiki simplemente debemos crearnos una cuenta gratuita en [Wikidot](https://www.wikidot.com) y crear el sitio que deseemos. Por cada cuenta gratuita tendremos cinco sitios disponibles para crear, simplemente cuando lo vayamos a crear tendremos que rellenar los siguientes campos:

- Título.
- Posibles tags de la wiki.
- Dirección web.
- Idioma.
- La plantilla que queramos.
- Acceso a las modificaciones por parte de usuarios que queramos dar.
- Aceptar los términos.

Una vez rellenados estos campos podremos acceder a la wiki y modificarla a nuestro gusto. Para acceder a la wiki de la aplicación, se deberá pulsar en el siguiente enlace: <https://wikinetextractor.wikidot.com>.

4.4 Realización de pruebas

En la carpeta `/tst/` existe una clase en la cual existen una serie de tests que el desarrollador podrá ejecutar para comprobar el correcto funcionamiento de algunas de las funcionalidades que posee la aplicación. Sobre todo, en esta clase se encuentran las funcionalidades correspondientes a la modificación del personaje. Para ejecutar estas pruebas, hay que comentar en el *main* la última línea, y quitar como comentario las dos líneas anteriores.

Otras pruebas que se han realizado en la aplicación con el fin de comprobar su correcto funcionamiento han sido a través del modo *debug*, que permite realizar cambios sobre la aplicación sin la necesidad de reiniciar la misma, de tal forma que podemos ir realizando cambios y actualizando la página visualizando los nuevos cambios.

4.5 Algunos comentarios adicionales

En este apartado se van a tratar algunos temas adicionales a tener en cuenta a la hora de utilizar y modificar la aplicación. Hay algunas funcionalidades de esta que se han obtenido mediante el *scraping* de una web, es decir, la recogida de información de esta página mediante la lectura de su código.

Se deberá tener en cuenta esto debido a los posibles cambios que puedan surgir en dichas páginas y que puedan impedir el correcto funcionamiento de la aplicación, debido a que un cambio en el código de la página o de su estructura, podría impedir la realización de alguna de estas funcionalidades, por tanto, se debe tener en cuenta a la hora de tener que adaptar el código a dichos cambios o incluso proceder a la búsqueda de una nueva web con un patrón más legible.

Apéndice E. Documentación de usuario

5.1 Introducción

En este apartado se van a tratar temas relativos a la ejecución de la aplicación por parte del usuario, de tal forma que se detallaran los requisitos que hacen falta al usuario para poder utilizar la aplicación y también se detallarán los procesos de instalación y una guía de usuario con todas las pantallas que posee la aplicación, de tal forma que facilite el entendimiento al usuario.

5.2 Requisitos de usuario

La aplicación va a ser accesible por parte del usuario de dos formas, remota o local. Para acceder de forma remota a la aplicación el usuario tendrá que introducir en su navegador el siguiente enlace <https://netextractor.herokuapp.com>, si por el contrario desea acceder de forma local, se deberá acceder a <http://127.0.0.1:5000/>, acceder de forma local permite al usuario poseer la aplicación sin límites, ya que la versión gratuita de Heroku posee ciertas restricciones.

Al tratarse de una aplicación web, el usuario simplemente necesita un navegador web para poder acceder a la página. Por tanto, los únicos requisitos para poder utilizar la aplicación van a ser poseer las siguientes versiones de navegador:

- Safari 9 o superior.
- Opera 28 o superior.
- Google Chrome 41 o superior.
- Mozilla Firefox 40 o superior.
- Microsoft Edge 12 o superior.

5.3 Instalación

En cuanto a la instalación de la aplicación, no es necesario que el usuario instale nada acerca de NetExtractor, simplemente alguno de los navegadores citados en el apartado anterior.

Para su uso en local sí que es necesario instalar ciertos requisitos que quedan detallados en la sección de “[Manual del programador](#)” dentro de este documento.

5.4 Manual de usuario

En este apartado se van a describir cada una de las pantallas que va a visitar el usuario a lo largo de la aplicación, con la finalidad de que facilite su aprendizaje y entienda todos los conceptos que se tratan en cada una de las pantallas.

Para acceder a la aplicación debemos entrar, como mencionamos anteriormente, al enlace de <http://netextractor.herokuapp.com> o bien si estamos en local a <http://127.0.0.1:5000/>.

Alertas antes de usar la aplicación

Se recomienda a los usuarios que no empleen los botones del navegador para avanzar retroceder y actualizar la página, puede causar problemas con las sesiones de usuario.

Se recomienda que los guiones estén introducidos con el formato general de escenas que tienen la mayor parte de guiones de la página <https://www.imsdb.com/> es decir, separación de escenas por ‘INT’ y ‘EXT’. En caso contrario, puede que los diccionarios no se creen de forma correcta o incluso que no se lleguen a crear.

INT. DEPT. OF HEALTH, OFFICE - MORNING

Figura 50 Ejemplo escena

Barra de navegación

En cada una de las pantallas de la aplicación, se va a disponer de una barra de navegación, la cual nos va a permitir acceder a diversas opciones. Las opciones a las que podemos acceder son las siguientes:

- Home: Con este botón podremos volver al inicio.
- Repositorio: Permite el acceso al repositorio de la aplicación, esto se abrirá en una nueva pestaña.
- Acerca de: Permite acceder a la información acerca de la aplicación en una nueva pestaña.
- Wiki: Permite acceder a la wiki de la aplicación alojada en Wikidot. Para ello, se abrirá una nueva pestaña.
- Idiomas: Habrá dos banderas situadas en la parte derecha de la barra, si se pulsa sobre una bandera, cambiará el idioma al que se corresponda.

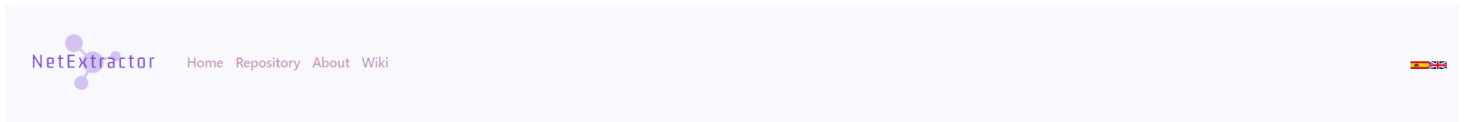


Figura 51 Barra de navegación

Botones de navegación de la aplicación

A lo largo de la aplicación vamos a tener una barra en la parte inferior de cada pantalla en la que vamos a tener un máximo de dos botones un mínimo de un botón. Cuando hay dos botones va a significar que tendremos un botón de retroceso y otro de avance en la parte izquierda y derecha respectivamente, la excepción es en la visualización del informe en la cual el botón derecho sirve para acceder al menú de navegación.

Cuando tenemos simplemente un botón va a ser porque dicho botón nos va a servir para retroceder por la página, hay una excepción y es la página principal en la cual solo hay un botón de avance.



Figura 52 Botones de navegación

Página principal

Esta será la página principal de la aplicación, donde nos aparecerá una serie de párrafos explicando por encima en lo que consiste la aplicación y cómo se extraen algunos datos.

Para pasar a la siguiente página, nos dará la opción de pulsar un botón de “Empezar” el cual nos llevará a la siguiente pantalla.

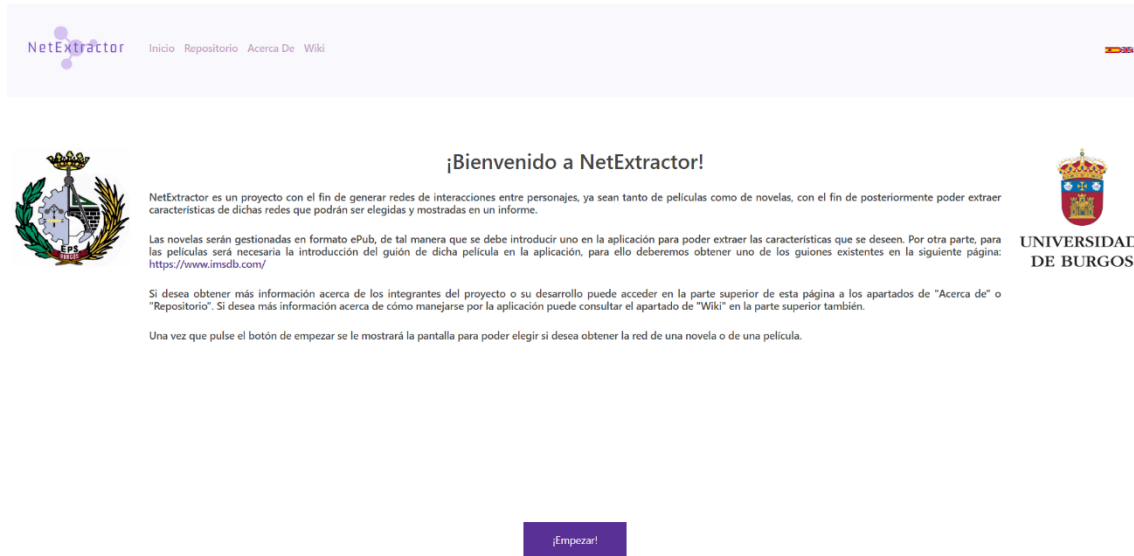


Figura 53 Página principal de NetExtractor

Menú de selección

La siguiente pantalla una vez hemos pulsado el botón de empezar va a ser la de selección, de tal forma que podemos seleccionar entre tres opciones, introducir un ePub en la aplicación,

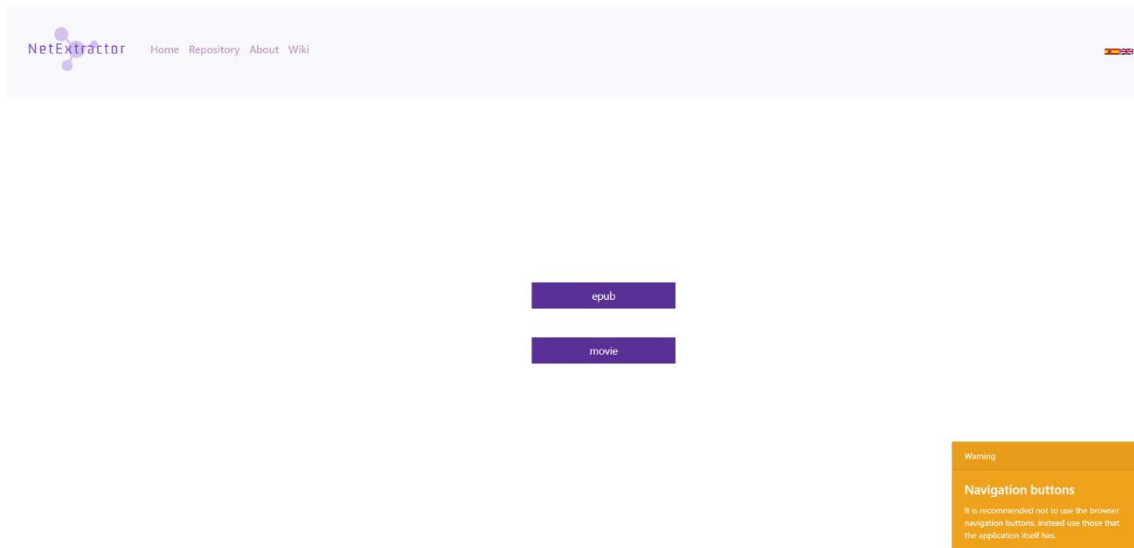


Figura 54 Selección de opciones NetExtractor

introducir un guion o usar la prueba de la aplicación, la cual generará un diccionario automático para que el usuario pueda probar la aplicación con un diccionario ya creado de prueba.

También se muestra en la parte inferior derecha un mensaje de aviso, recomendando al usuario no usar los botones de navegación.

Epub – Introducir ePub

Esta pantalla será mostrada una vez que hemos seleccionado el ePub, de tal forma que nos muestra una pantalla en la que introducir el ePub extraído de nuestro dispositivo. Simplemente debemos hacer clic en “Seleccionar archivo” y seleccionar el ePub que queramos. Más tarde, tendremos que presionar sobre el botón de “Cargar” para que el ePub sea introducido y cargado y pasaremos a la siguiente pantalla.

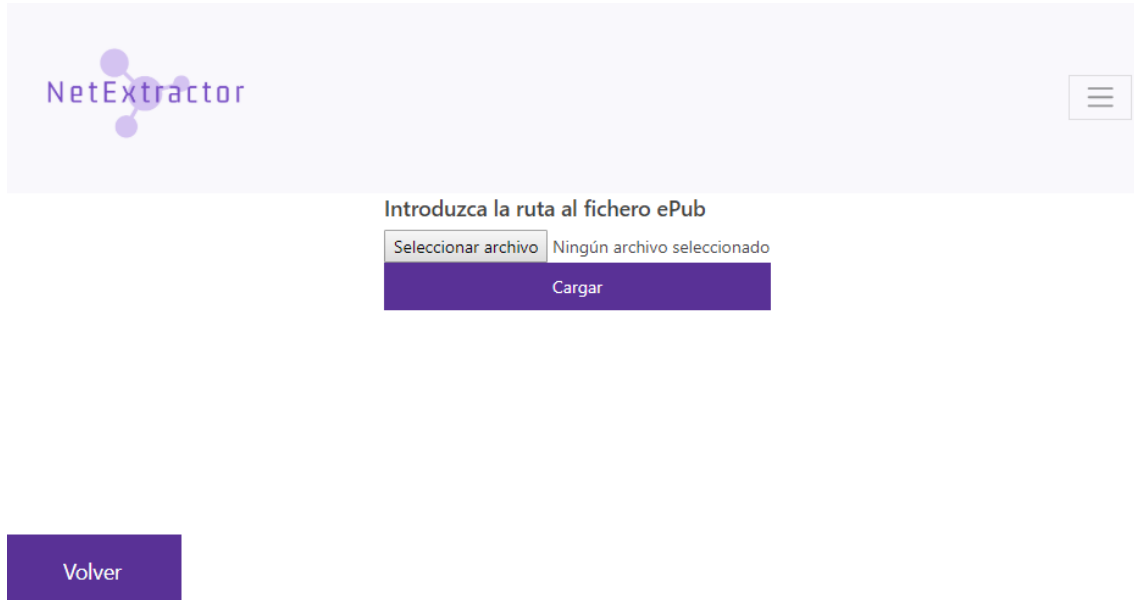


Figura 55 Introducción de ePub

Epub – Diccionarios automáticos

Si hemos introducido un ePub, nos va a llevar a la creación de diccionarios, en esta parte de la aplicación vamos a poder crear el diccionario de personajes para una novela, tenemos diversas opciones con las cuales crear un diccionario:

- Crear diccionario automáticamente: esta opción sirve para poder crear un diccionario de forma automática, puede ser que cause falsos positivos al examinar toda la novela.
- Importar un diccionario: esta opción nos llevará a otra pantalla explicada a continuación. Para acceder a ese apartado haz clic aquí.
- Obtener diccionario: esta opción nos lleva a otra pantalla para obtener un diccionario de una wiki de personajes, para poder acceder a este apartado puede hacer clic aquí.

Estas son las distintas formas que tenemos para crear un diccionario, puede que el usuario quiera combinar alguna de ellas para conseguir un abanico de personajes mucho más concreto y preciso. Hay otra opción en esta pantalla y sirve para poder vaciar un diccionario, es decir, si se ha creado un diccionario, pero no estamos de acuerdo o preferimos simplemente cambiar de opciones, podemos pulsar en la opción de vaciar diccionario, de tal forma que el diccionario quede vacío. No se podrá acceder a la siguiente página si no se ha introducido o creado ningún diccionario.

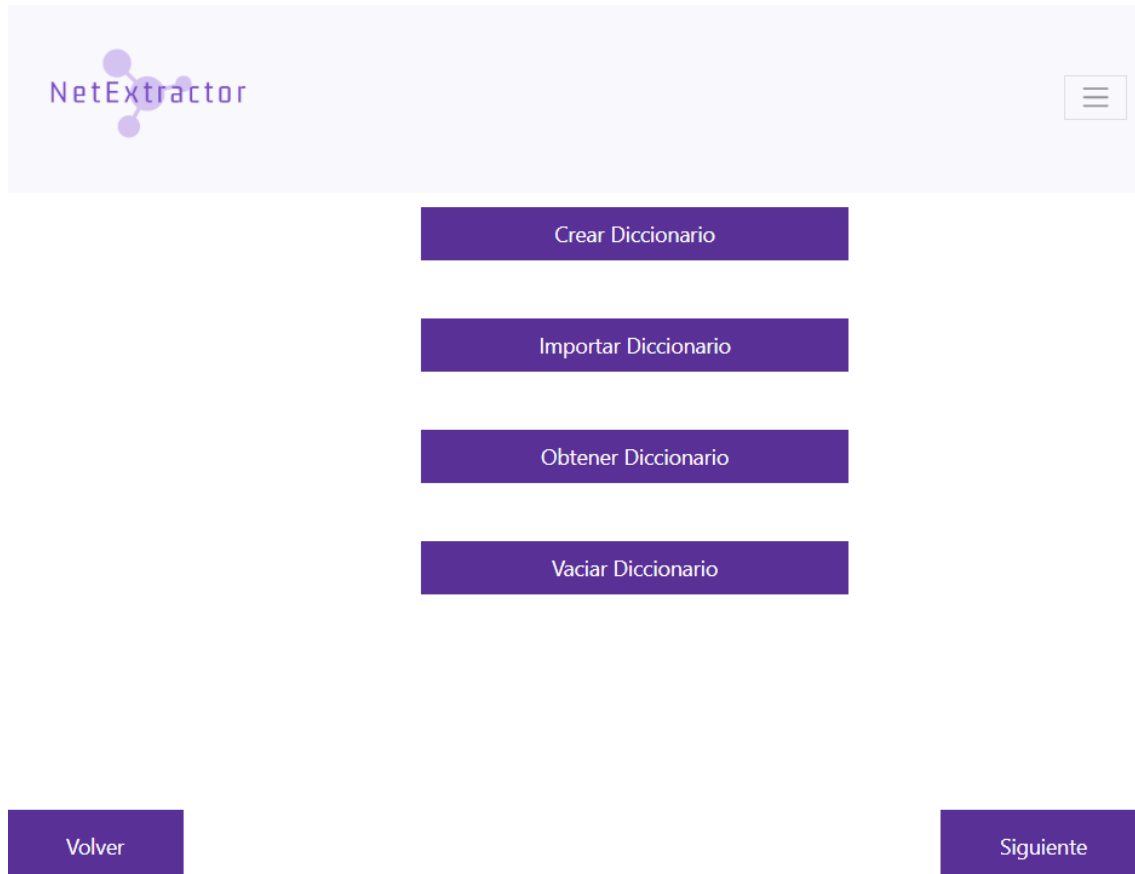


Figura 56 Creación de diccionarios

Epub – Importar diccionario

En esta pantalla se permite al usuario importar un fichero en formato csv, de tal forma que el usuario simplemente tiene que pulsar en “Seleccionar archivo” y seleccionar el csv que quiera, después pulsar en el botón de “Cargar” para subir el csv a la aplicación. No se podrá acceder a la siguiente página si no se ha introducido o creado ningún diccionario.

Figura 57 Importar diccionario

El formato que debe tener el csv tiene que ser el siguiente:

- La primera fila del csv va a ser ignorada.
- A continuación de esta primera fila vendrán las filas divididas en pares de la siguiente forma:
 - La primera fila contendrá la id que va a tener el personaje y lo que figurará como el primer nombre que tendrá el personaje.
 - La segunda fila será para dar lugar al resto de referencias que pueda contener el personaje. Hay ciertos factores a tener en cuenta en esta segunda fila:
 - Esta puede quedar vacía y entonces se tomará como nombre la id del personaje en la primera fila.
 - Puede repetirse el nombre de la id que pusimos en la primera fila sin estar duplicado.
 - Se pueden añadir más de un nombre separados por comas.

```

1  Diccionario
2  Pedro Ro
3  Pedro, Pedro Rodriguez, Pedro R
4  María
5  María
6  Jose
7  Josema
8  Pedrope
9  Pedro Pérez
10 Ana
11 Ana
12
    
```

Epub – Obtener diccionario

También se puede obtener el diccionario accediendo a la pantalla de obtener diccionario, en esta pantalla te da la opción de introducir un enlace que debe ser obtenido de una página en

concreto, la cual pertenece a una wiki de personajes en la cual obtenerlos. El enlace al que debemos acceder para poder obtener es el siguiente: <https://www.fandom.com/>.

En esta wiki podemos buscar la novela que queramos para poder obtener sus personajes de un enlace tipo: [https://\"nombrenovela\".fandom.com/wiki/Category:Characters](https://\). No se podrá acceder a la siguiente página si no se ha introducido o creado ningún diccionario.

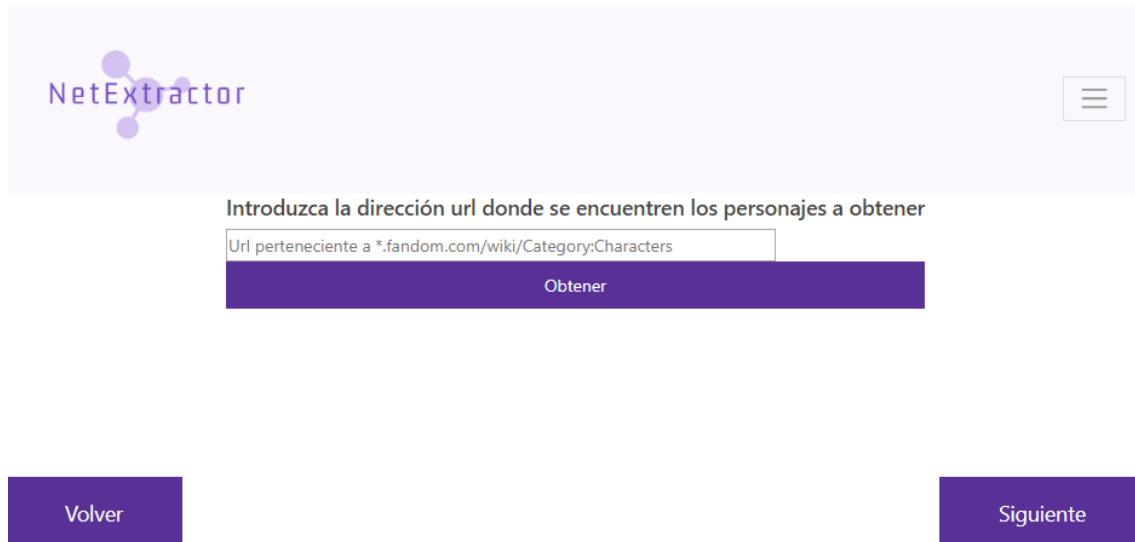
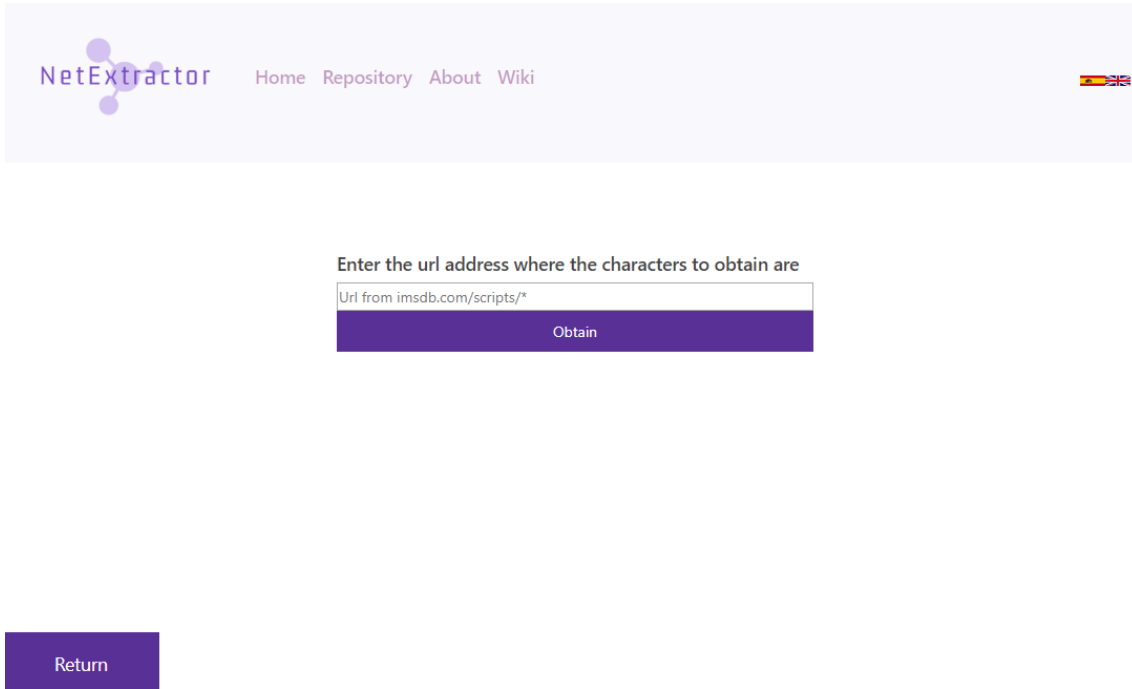


Figura 58 Obtener diccionario

Película – Introducción del guion

En esta pantalla se obtendrá el guion de una película que se encuentre en la página <https://www.imsdb.com/>, la pantalla es muy similar a la de obtener diccionario explicada anteriormente, pero esta vez el enlace a introducir va a ser distinto.

Para obtener los personajes de la película se examina la página buscando todos los personajes que existen, los personajes obtenidos tienen que encontrarse en negrita en el guion ya que es el patrón común que se ha encontrado en la mayor parte de los guiones de esta web, en caso de que la película no siga con la estructura que poseen la mayoría de los guiones, puede haber problemas a la hora de obtener sus personajes e incluso llegar a no crear el diccionario de personajes. En este caso prueba con otro guion que cumpla las condiciones.



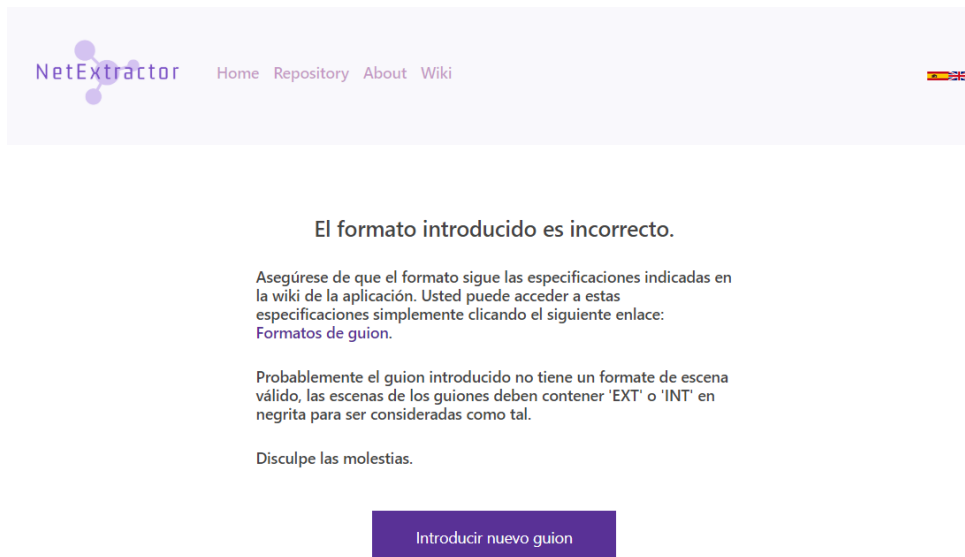
The screenshot shows the NetExtractor application interface. At the top, there is a navigation bar with the logo 'NetExtractor' and links for 'Home', 'Repository', 'About', and 'Wiki'. A small flag icon is visible on the right. Below the navigation bar, there is a form with the label 'Enter the url address where the characters to obtain are'. The input field contains the text 'Url from imdb.com/scripts/*'. Below the input field is a purple button labeled 'Obtain'. At the bottom left of the form, there is a purple button labeled 'Return'.

Figura 59 Introducción guion

Debido al gran número de guiones, se ha tenido que establecer un determinado formato y estructura para que los guiones sean aceptados. Esta estructura ha sido elegida después de examinar un gran número de guiones con el fin de comprobar las características que por lo general tenían. Así, han acabado definiéndose las siguientes especificaciones:

- Los guiones deben contener escenas, no sólo diálogos entre personajes.
- Las escenas deben estar marcadas por el texto 'EXT ', 'INT ', 'EXT. ' o 'INT. '. Esto es debido a que algunos guiones distinguen las escenas por el nombre de la escena, sin aportar ninguna estructura especial que permita diferenciarlas.
- Los personajes deben estar en negrita. Entre etiquetas de html.

Si alguna de las anteriores características no se cumple, se redirigirá al usuario a la siguiente pantalla, explicando el por qué y dando la opción de que vuelva a introducir un guion.



The screenshot shows the NetExtractor application interface displaying an error message. The navigation bar at the top is the same as in Figure 59. The main content area has a title 'El formato introducido es incorrecto.' followed by a paragraph: 'Asegúrese de que el formato sigue las especificaciones indicadas en la wiki de la aplicación. Usted puede acceder a estas especificaciones simplemente clicando el siguiente enlace: [Formatos de guion.](#)' Below this, another paragraph states: 'Probablemente el guion introducido no tiene un formate de escena válido, las escenas de los guiones deben contener 'EXT' o 'INT' en negrita para ser consideradas como tal.' At the bottom, there is a line 'Disculpe las molestias.' and a purple button labeled 'Introducir nuevo guion'.

Figura 60 Error en formato


Modificar un diccionario

Cuando se hayan creado los guiones en alguna de las dos opciones que proporciona la aplicación, accederemos a la pantalla de modificar diccionario en la cual tenemos diversas opciones que podemos hacer con los distintos personajes. Estas opciones son las siguientes:

- Añadir personajes.
- Borrar personajes.
- Juntar personajes.
- Añadir referencias.
- Borrar referencias.
- Modificar la id del personaje.
- Modificar la etnia del personaje.
- Modificar el sexo del personaje.
- Exportar diccionario.

También vamos a tener dos opciones en forma de *checks* con las cuales podemos obtener las posiciones de los personajes, esta opción viene marcada por defecto nada más que se accede a la pantalla de modificar diccionario de manera que siempre tenga que calcular las posiciones de los personajes. Y la otra casilla se corresponde con obtener etnia y sexo, es decir, usar Ethnea y Genni con el fin de poder predecir que etnia y sexo poseen los personajes que hemos obtenido.

Para cada una de las dos opciones anteriores, se deberá marcar la casilla y pulsar en el botón de siguiente, una vez hecho esto aparecerán las posiciones o la etnia y sexo.



ID Personaje	Referencias personaje	Número Apariciones	Etnia	Sexo
JOKER	JOKER	N/A	None	None
SOCIAL WORKER	SOCIAL WORKER	N/A	None	None

Figura 61 Modificar personajes

Añadir personaje


Si se ha accedido a esta pantalla es porque se quiere añadir un personaje a nuestro diccionario de personajes, añadir un personaje se puede hacer de una forma muy simple y se explicará con detalle a continuación, además también se explicarán las posibles alertas que nos pueden surgir a la hora de introducir un personaje.

- Para introducir un personaje debemos hacer lo siguiente.
- Introducir la id del personaje que queremos crear.
- Introducir la referencia o el nombre asignado a esa id que acabamos de introducir.
- Presionar el botón de "Añadir".

Una vez que hemos pulsado en el botón de añadir, pueden darse una serie de alertas dependiendo de si lo que hemos introducido se ha introducido de una forma correcta o incorrecta, las posibles alertas son las siguientes:

- La primera alerta surge cuando se ha introducido un personaje de forma correcta, indicando que el personaje se ha introducido correctamente en el diccionario.
- Otra posibilidad es no haber introducido alguno de los campos o ninguno, en este caso nos mostrará una alerta indicando que debemos introducir todos los campos.
- La última opción es que la id del personaje ya esté en el diccionario en cuyo caso se nos notificará diciendo que ya está dicha id en el diccionario.

Para salir de esta página simplemente debemos presionar el botón de la parte inferior izquierda que dice "Atrás".



NetExtractor

Introduzca la id del nuevo personaje:

Introduzca el nombre del nuevo personaje:

ID Personaje	Referencias personaje	Número Apariciones	Etnia	Sexo
JOKER	JOKER JOKER (CONT'D) JOKER (ON TV) JOKER (OS)	N/A	None	None
SOCIAL WORKER	SOCIAL WORKER	N/A	None	None
KID #1	KID #1	N/A	None	None
KID #2	KID #2	N/A	None	None

Figura 62 Añadir personajes

Borrar personajes

Si se ha accedido a esta página es porque se desea eliminar alguno de los personajes, opción muy frecuente en caso de haber creado los diccionarios de una forma automática, tanto de películas como de novelas, ya que se van a dar cierto número de falsos positivos. Para borrar un personaje el proceso es de los más sencillo que tiene la aplicación. Se explica a continuación.

- Debemos marcar primero el personaje o personajes que queremos eliminar.
- Una vez marcados debemos presionar sobre el botón de borrado y los personajes se borrarán del diccionario.
- Se alertará al usuario que el personaje ha sido borrado cuando la acción se lleve a cabo.

Para volver atrás simplemente debemos pulsar el botón de "Atrás" en la parte inferior izquierda.



Figura 63 Borrar personajes

Juntar personajes

En esta página se pueden juntar distintos personajes en caso de que un personaje sea igual a otro, pero por cuestiones de nombre la aplicación les haya creado por separado. El proceso de unión es muy parecido al de borrado, simplemente lo que tenemos que hacer es seleccionar los personajes que deseamos juntar y presionar el botón de "Juntar", de tal manera que los personajes se cuenten como uno solo.

Esta opción es muy frecuente cuando se generan diccionarios automáticos ya que al detectar un nombre distinto los considera como un personaje distinto. La aplicación nos genera una alerta diciendo que los personajes se han juntado de forma exitosa, a partir de ahora, todos los enlaces que tuvieran cada personaje individualmente, los tendrán de forma conjunta.

Para volver atrás simplemente debemos pulsar el botón de "Atrás" en la parte inferior izquierda.



Figura 64 Juntar personajes

Añadir referencias

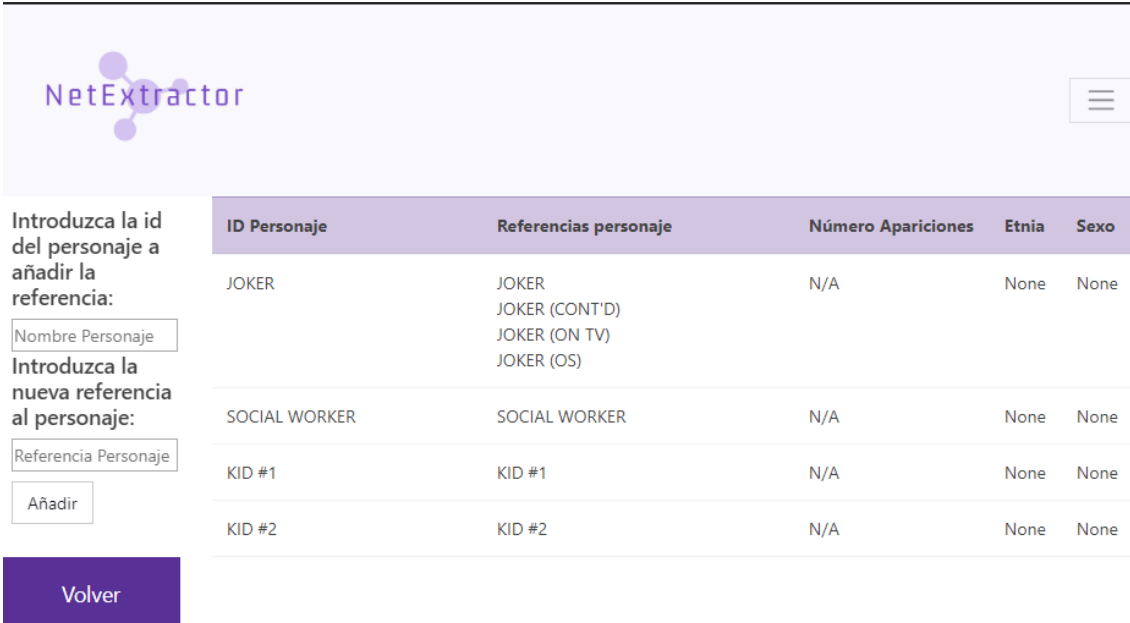
Esta es la página que va a ser empleada para añadir nuevas referencias a un personaje, en caso de que tengamos un personaje del cual conocemos más nombres que se le dan, pero no figuran en la creación del diccionario, lo podremos añadir de forma sencilla accediendo a esta página en la que simplemente debemos realizar los siguientes pasos.

- Introducir la id del personaje al que deseamos añadir una nueva referencia.
- Introducir la referencia que queremos añadir.
- Presionar sobre el botón de añadir.

Una vez hecho esto la aplicación nos puede mostrar distintos tipos de alertas dependiendo de si la introducción de la referencia se ha realizado de una forma satisfactoria o por el contrario a ocurrido algún error. Los mensajes posibles son los siguientes.

- Referencia añadida de forma correcta en caso de que la todos los parámetros introducidos sean correctos.
- Id inexistente en caso de intentar añadir una referencia en una id que no exista.
- Si no se introduce alguno de los campos o ninguno también se nos mostrará una alerta comentando que debemos introducir todos los campos para poder añadir una referencia.

Para volver atrás simplemente debemos pulsar el botón de "Atrás" en la parte inferior izquierda.



NetExtractor

Introduzca la id del personaje a añadir la referencia:

Nombre Personaje

Introduzca la nueva referencia al personaje:

Referencia Personaje

Añadir

Volver

ID Personaje	Referencias personaje	Número Apariciones	Etnia	Sexo
JOKER	JOKER JOKER (CONT'D) JOKER (ON TV) JOKER (OS)	N/A	None	None
SOCIAL WORKER	SOCIAL WORKER	N/A	None	None
KID #1	KID #1	N/A	None	None
KID #2	KID #2	N/A	None	None

Figura 65 Añadir referencia

Eliminar referencia

En esta página podremos borrar alguna referencia que tenga un personaje en caso de que no queramos tener dicha referencia. Para ello los pasos son muy parecidos a los seguidos cuando queremos borrar un personaje solo que ahora deberemos marcar la casilla que queramos eliminar en la parte de referencias y no de id de personaje. Una vez que hemos marcado la casilla

correspondiente a la referencia que deseamos borrar simplemente debemos hacer clic en el botón de borrado para que se borre la referencia. Cuando la referencia ha sido eliminada se mostrará un mensaje al usuario avisándole de que la referencia ha sido borrada con éxito.

Para volver atrás simplemente debemos pulsar el botón de "Atrás" en la parte inferior izquierda.



Figura 66 Borrar referencia

Modificar la id del personaje

Si el usuario ha accedido a la siguiente página es porque se quiere modificar la id de un personaje, para ello debemos de introducir parámetros parecido a cuando se quiere añadir un personaje con la diferencia de que en vez de introducir el nombre que deseamos que tenga el personaje después de introducir la id, aquí debemos realizar lo siguiente.

- Introducir la id del personaje que se desea modificar.
- Introducir la nueva id que se quiere asignar al personaje.
- Pulsar el botón de cambiar.

Este proceso puede generar diversas alertas a la hora de pulsar el botón de cambiar.

- Puede que el proceso se realice de forma correcta en cuyo caso nos aparecerá un mensaje por pantalla que nos indicará que la id ha sido cambiada con éxito.
- Otra opción es que alguno de los parámetros no se haya introducido o ningún parámetro se haya introducido, en ese caso se alertará al usuario de que debe introducir todos los parámetros.
- La última opción es que la id actual introducida no exista o que la id por la que se quiere cambiar ya exista, en este caso se notificará al usuario que la id no existe o que se quiere cambiar el id por uno ya existente.

Para volver atrás simplemente debemos pulsar el botón de "Atrás" en la parte inferior izquierda.



ID Personaje	Referencias personaje	Número Apariciones	Etnia	Sexo
JOKER	JOKER JOKER (CONT'D) JOKER (ON TV) JOKER (OS)	N/A	None	None
SOCIAL WORKER	SOCIAL WORKER	N/A	None	None
KID #1	KID #1	N/A	None	None
KID #2	KID #2	N/A	None	None

Figura 67 Modificar id

Modificar etnia


En esta pantalla podremos modificar la etnia, en caso de que la etnia predicha no sea correcta o no sea la que el usuario deseaba. También se puede cambiar la etnia sin haber predicho las etnias de los personajes, pero si cambiamos la etnia y luego hacemos la predicción, la etnia predicha sobrescribirá la anterior. Para modificar la etnia de un personaje debemos realizar lo siguiente.

- Introducir la id del personaje al que queremos cambiar la etnia.
- Introducir la etnia que queramos asignarle.
- Presionar en el botón de modificar.

Estas opciones nos pueden mostrar las siguientes alertas.

- La id introducida no existe, en este caso nos notificará que la id es inexistente.
- Si no se ha introducido alguno de los campos también se notificará al usuario de ello.
- Si todo se ha introducido con éxito nos aparecerá una alerta diciendo que la introducción de los datos ha sido correcta y la etnia ha sido modificada.

Para volver atrás simplemente debemos pulsar el botón de "Atrás" en la parte inferior izquierda.



NetExtractor

Introduzca la id del personaje

ID Personaje

Introduzca la etnia del personaje

Etnia Personaje

Modificar

Volver

ID Personaje	Referencias personaje	Número Apariciones	Etnia	Sexo
JOKER	JOKER JOKER (CONT'D) JOKER (ON TV) JOKER (OS)	N/A	None	None
SOCIAL WORKER	SOCIAL WORKER	N/A	None	None
KID #1	KID #1	N/A	None	None
KID #2	KID #2	N/A	None	None

Figura 68 Modificar etnia

Modificar sexo

En esta pantalla podremos modificar el sexo, en caso de que el sexo predicho no sea correcto o no sea el que el usuario deseaba. También se puede cambiar el sexo sin haber predicho los sexos de los personajes, pero si cambiamos el sexo y luego hacemos la predicción, el sexo predicho sobrescribirá el anterior. Para modificar el sexo de un personaje debemos realizar lo siguiente.

- Introducir la id del personaje al que queremos cambiar el sexo.
- Elegir en el menú desplegable la opción que deseamos para el sexo.
- Presionar en el botón de modificar.

Estas opciones nos pueden mostrar las siguientes alertas.

- La id introducida no existe, en este caso nos notificará que la id es inexistente.
- Si no se ha introducido alguno de los campos también se notificará al usuario de ello.
- Si todo se ha introducido con éxito nos aparecerá una alerta diciendo que la introducción de los datos ha sido correcta y el sexo ha sido modificado.

Para volver atrás simplemente debemos pulsar el botón de "Atrás" en la parte inferior izquierda.



NetExtractor

Introduzca la id del personaje

ID Personaje

Introduzca el sexo del personaje

Seleccione opción ▼

Modificar

Volver

ID Personaje	Referencias personaje	Número Apariciones	Etnia	Sexo
JOKER	JOKER JOKER (CONT'D) JOKER (ON TV) JOKER (OS)	N/A	None	None
SOCIAL WORKER	SOCIAL WORKER	N/A	None	None
KID #1	KID #1	N/A	None	None
KID #2	KID #2	N/A	None	None

Figura 69 Modificar sexo

Parámetros que introducir para la creación de la red

Hay dos opciones de parámetros adicionales dependiendo de si lo que se ha introducido es una película o un guion, estas dos posibilidades van a ser explicadas a continuación.

Parámetros adicionales para los ePub

En caso de que hayamos accedido a esta pantalla es porque en la selección hemos elegido introducir un ePub. En este caso tendremos unos parámetros que debemos introducir para pasar a la pantalla de visualización. Estos parámetros van a afectar en la creación de la red, dependiendo de que pongamos unos u otros la red tendrá una forma u otra. Los pasos para pasar a la siguiente pantalla serán los siguientes.

- Introducir el número de apariciones mínimo que debe tener el personaje, este paso es igual que si hubiéramos introducido un guion.
- Introducir el número de palabras máximo que debe haber entre dos personajes para que se cuente como relación. Es decir, si introducimos un 10, la aplicación comprobará 10 palabras adelante y atrás de cada personaje buscando otros personajes con los que establecer la relación.
- Marcar si queremos tener los capítulos en cuenta o no, es decir, si queremos que si hay un personaje en otro capítulo con menos del valor introducido anteriormente de palabras entre ellos, queremos que lo cuente como relación o consideramos que al estar en un nuevo capítulo no necesitamos que se cuente como relación.

Al igual que anteriormente, cuando pulsamos al botón de siguiente, nos puede notificar la aplicación con varios tipos de alertas que serán los siguientes.

- Si algún parámetro o los dos quedan sin introducir, nos notificará de que debemos introducir dichos parámetros.
- Si todo se ha introducido correctamente no notificará que todo está correcto y nos dirigirá a la siguiente pantalla.



NetExtractor

Introduzca el número mínimo de apariciones en el libro:

Número mínimo de a

Introduzca la distancia máxima de palabras entre personajes para que cuente como relación:

Número de palabras

ID Personaje	Referencias personaje	Número Apariciones	Etnia	Sexo
Pedro Pérez	Pedro Pérez	2	None	None
Pedro	Pedro	2	None	None
Josema	Josema	1	None	None
Pedro Rodríguez	Pedro Rodríguez	1	None	None
Ana	Ana	1	None	None

Volver

Obtener red

Figura 70 Parámetros adicionales ePub

Parámetros adicionales para los guiones

Si hemos accedido a esta pantalla va a ser porque en la selección elegimos la película y por tanto en nuestra aplicación hemos introducido el guion de una película. Para esta pantalla simplemente tenemos que introducir un parámetro que afectará a la creación de la red. Los pasos que debemos seguir para introducir dicho parámetro son los siguientes.

- Introducir el número mínimo de apariciones que queremos que el personaje tenga para que aparezca en la red que se generará a continuación.

Una vez introducido el número de apariciones mínimo se deberá pulsar en el botón de siguiente. Hecho esto, habrá dos alertas que nos pueden aparecer por pantalla.

- Si no se ha introducido ningún valor nos aparecerá la primera alerta diciendo que debemos introducir un valor.
- Si todo se ha introducido de forma correcta avanzará hasta la siguiente pantalla de visualización.

Introduzca el número mínimo de apariciones en la película:

Número mínimo de a

ID Personaje	Referencias personaje	Número Apariciones	Etnia	Sexo
JOKER	JOKER JOKER (CONT'D) JOKER (ON TV) JOKER (OS)	55	None	None
MOM (OS)	MOM (OS) MOM MOM (OS) (CONT'D) MOM (VO) MOM (V0)	14	None	None

Volver

Obtener red

Figura 71 Parámetros adicionales películas

Generador de la red

En esta página se va a mostrar la visualización de la red a través de la siguiente herramienta presente en [GitHub](#):

En esta pantalla se le va a permitir al usuario que pueda modificar algunas características de la red como:

- Node size: Tamaño de los nodos.
- Link size: Tamaño de los enlaces.
- Zoom: con el que se quiere visualizar la red.
- Link distance: La distancia que deben tener los enlaces.
- Charge strength o Center gravity: La gravedad o la fuerza de la red que afecta a que la red se vea más compacta o más separada.
- Node fill, Node stroke y Link stroke: Color de nodos y enlaces.
- Label stroke: Color de las etiquetas de nodo.
- Show labels: Mostrar o no las etiquetas del nodo.
- Node size by strength: Si queremos que el nodo salga más grande o más pequeño dependiendo del grado.
- Show singleton nodes: Mostrar los nodos separados.

Se pueden modificar más características pero las anteriores son las más importantes. En esta pantalla también podemos descargar la red en distintos formatos para poder ser usada en otras herramientas que acepten dichos formatos. Formatos disponibles:

- Descargar NET: para poder introducir la red en NetworkX o Pajek, por ejemplo.
- Descargar GEXF: para poder usar la red en Gephi por ejemplo.
- Descargar GML.

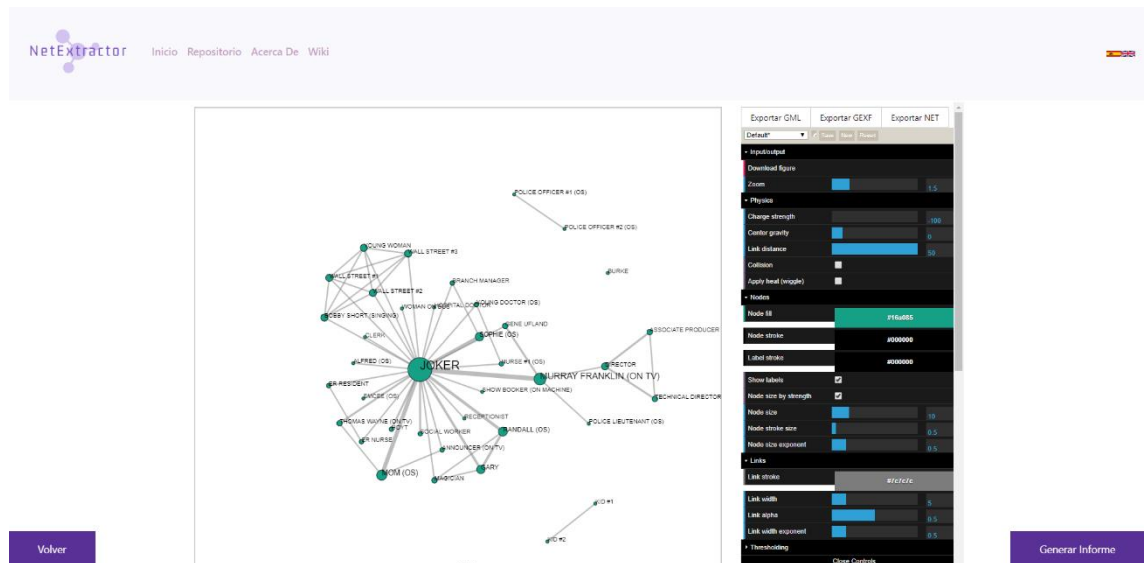


Figura 72 Visualización de la red

Extracción de datos para el informe

En esta pantalla se podrán elegir todas las características que vamos a tener a la hora de obtener la red, todas ellas van a ser generadas en el informe final una vez que se avance a la siguiente página. Las opciones que se van a encontrar en esta página van a ser las que aparecen en la siguiente lista, con la posibilidad de ver lo que hace cada medida siguiendo el enlace con las que están marcadas.

Medidas básicas:

- Número de nodos.
- Número de enlaces.
- Número de interacciones.

Medidas de grado (el grado si contar el peso de un nodo es el número de enlaces que tiene, si hay peso en los enlaces el grado es el sumatorio de los pesos de sus enlaces):

- Grado de los nodos sin contar con el peso de los enlaces.
- Grado de los nodos teniendo en cuenta el peso de los enlaces.
- Distribución de grado sin tener en cuenta el peso.
- Distribución de grado teniendo en cuenta el peso.
- Densidad.

Medidas de distancia:

- Componentes conectados.
- Excentricidad: distancia máxima que va a existir entre ese nodo y cualquier otro nodo de la red, siguiendo caminos de distancia mínima.
- Diámetro: la máxima de las excentricidades.
- Radio: la mínima de las excentricidades.
- Distancia media.

Medidas de clustering:

- Coeficiente de clustering local.
- Coeficiente de clustering local medio.
- Transitividad.

Medidas de centralidad:

- Centralidad de grado.
- Centralidad de cercanía.
- Centralidad de intermediación.
- Centralidad de intermediación de camino aleatorio.
- Centralidad de valor propio.
- PageRank.

Grupos y comunidades:

- Comunidades K-clique.
- Comunidades Girvan-Newman.
- Comunidades Clauset-Newman-Moore.
- Comunidades Louvain.

Detección de roles:

- Roles K-clique.
- Roles Girvan-Newman.
- Roles Clauset-Newman-Moore.
- Roles Louvain.

Una vez seleccionadas todas las opciones que el usuario quiera obtener, simplemente se debe presionar el botón de "Generar Informe" y se accederá a la última pantalla de la aplicación en la cual se podrá visualizar el informe final.

The screenshot shows the 'Extracción de datos para el informe' screen of the NetExtractor application. The interface is organized into four columns of options, each with a header and a list of checkboxes:

- Medidas básicas:**
 - ☐ Número Nodos
 - ☐ Número Enlaces
 - ☐ Número Interacciones
- Medidas de grado:**
 - ☐ Grado Nodos Sin Peso
 - ☐ Grado Nodos Con Peso
 - ☐ Distribución Grado Sin Peso
 - ☐ Distribución Grado Con Peso
 - ☐ Densidad
- Medidas de Distancia:**
 - ☐ Componentes Conectados
 - ☐ Excentricidad
 - ☐ Diámetro
 - ☐ Radio
 - ☐ Distancia Media
- Medidas de Clustering:**
 - ☐ Coeficiente de Clustering Local
 - ☐ Coeficiente de Clustering Local Medio
 - ☐ Transividad
- Medidas de Centralidad:**
 - ☐ Centralidad Grado
 - ☐ Centralidad Cercanía
 - ☐ Centralidad Intermediación
 - ☐ Intermediación Camino Aleatorio
 - ☐ Centralidad Valor Propio
 - ☐ Pagerank
- Grupos y comunidades:**
 - ☐ Comunidades K-clique
 - Valor k del K-clique para comunidades: (Por defecto será 2)
 -
 - ☐ Comunidades Girvan-Newman
 - ☐ Comunidades Clauset-Newman-Moore
 - ☐ Comunidades Louvain
- Detección de roles:**
 - ☐ Roles K-clique
 - Valor k del K-clique para roles: (Por defecto será 2)
 -
 - ☐ Roles Girvan-Newman
 - ☐ Roles Clauset-Newman-Moore
 - ☐ Roles Louvain

At the bottom of the screen, there are two buttons: 'Volver' on the left and 'Generar Informe' on the right.

Figura 73 Extracción de datos

Visualización del informe

En esta pantalla se puede visualizar el informe que se ha obtenido después de seleccionar las características deseadas en el apartado anterior. En este informe se van a mostrar además un menú en la parte izquierda con todas las opciones que ha seleccionado el usuario con el fin de que se pueda navegar a través del informe de una forma mucho más sencilla. Para una mejor visualización de las imágenes el usuario debe pasar el ratón por encima de la

imagen y esta imagen será ampliada automáticamente. En cuanto el usuario quite el ratón de encima de la imagen, la imagen volverá a su tamaño original.

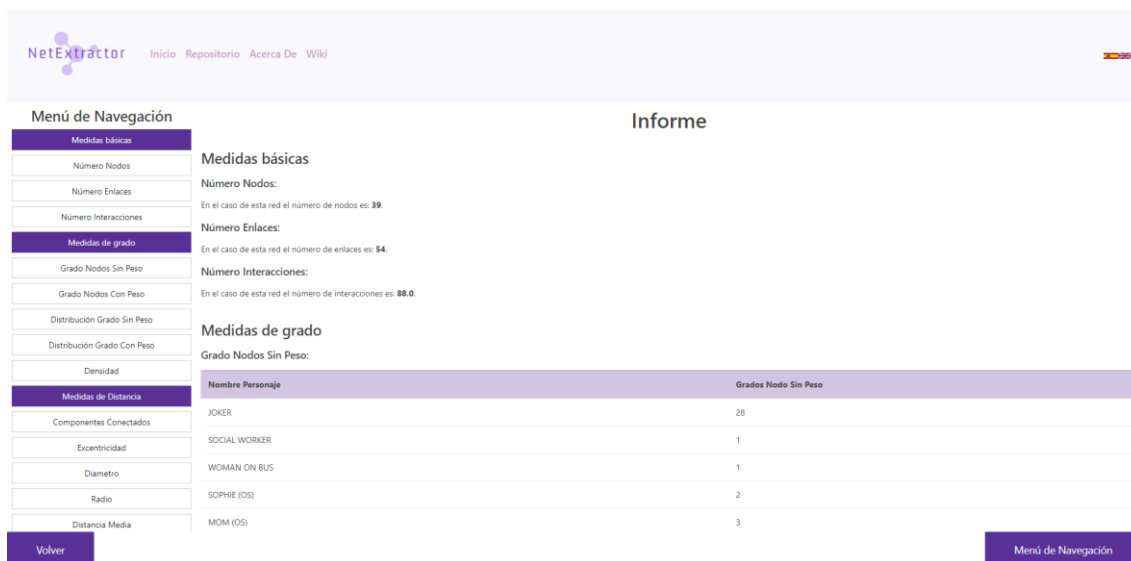


Figura 74 Informe

Bibliografía

1. Seguridad Social: Cotización / Recaudación de Trabajadores [Internet]. [citado 5 de febrero de 2020]. Disponible en: <http://www.seg-social.es/wps/portal/wss/internet/Trabajadores/CotizacionRecaudacionTrabajadores/36537?changeLanguage=es>
2. Modelo–vista–presentador. En: Wikipedia, la enciclopedia libre [Internet]. 2019 [citado 4 de febrero de 2020]. Disponible en: <https://es.wikipedia.org/w/index.php?title=Modelo%E2%80%93vista%E2%80%93presentador&oldid=119127836>
3. Torvik V, Agarwal S. Ethnea --an instance-based ethnicity classifier based on geo-coded author names in a large-scale bibliographic database. En 2016.
4. Aslak U. ulfaslak/network_styling_with_d3 [Internet]. 2020 [citado 28 de enero de 2020]. Disponible en: https://github.com/ulfaslak/network_styling_with_d3
5. Crea y comparte diseños, maquetas y prototipos rápidamente | Adobe XD [Internet]. [citado 5 de febrero de 2020]. Disponible en: <https://www.adobe.com/es/products/xd/details.html>
6. Cómo instalar y utilizar Composer en Ubuntu 18.04 [Internet]. DigitalOcean. [citado 3 de febrero de 2020]. Disponible en: <https://www.digitalocean.com/community/tutorials/como-instalar-y-utilizar-composer-en-ubuntu-18-04-es>