

UNIVERSIDAD DE ALMERIA

ESCUELA SUPERIOR DE INGENIERÍA

Cuadro de mandos
para la producción
hortícola en el
poniente almeriense

Curso 2019/2020

Alumno/a:

Razvan Ioan Stoia

Director/es:

José del Sagrado Martínez

Universidad de Almería

Escuela Superior de Ingeniería

Grado en Ingeniería Informática



Cuadro de mandos para la producción hortícola en el poniente almeriense

Autor:
Razvan Ioan Stoia

Director:
José del Sagrado Martínez

Septiembre 2020

Agradecimientos

Agradezco en primer lugar a mi familia, a mis padres Virgil y Ana y a mi hermano Adrian, por comprenderme y apoyarme siempre, y especialmente durante este largo camino.

Agradecer también a mis amigos, en especial a Mario por acompañarme en los momentos difíciles y a Hugo por compartir conmigo los éxitos y fracasos de la carrera.

Por último, agradezco la ayuda de mi tutor José del Sagrado, y su paciencia con mi mala planificación y mis infinitos correos.

Índice general

1. Introducción.	3
1.1. Motivación	3
1.2. Objetivos	3
1.3. Planificación inicial	4
1.4. Estructura de la memoria	5
1.5. Materiales y métodos	6
2. Conociendo los datos.	7
2.1. Extracción y estudio	7
2.1.1. Datos de producción	8
2.1.2. Datos climatológicos	9
2.1.3. Datos de mercado	11
2.2. Carga y transformación	12
2.2.1. Carga	13
2.2.2. Preprocesamiento	13
2.2.3. Transformación	15
2.2.4. Integración	18
3. Elementos, tecnologías, modelo y metodología de la solución.	19

3.1. Dashboards	19
3.1.1. ¿Qué es un dashboard?	19
3.1.2. Intelindalo	20
3.2. Tecnologías y herramientas	20
3.2.1. Planner	21
3.2.2. iMacros	21
3.2.3. Blasamiq Cloud	21
3.2.4. R y Rstudio	21
3.2.5. Shiny y Shinydashboard	22
3.2.6. Tidyverse	28
3.3. Modelo y metodología de desarrollo	30
4. Solución.	31
4.1. Definición del sistema	31
4.2. Análisis	33
4.2.1. Objetivos	33
4.2.2. Actores	35
4.2.3. Requisitos funcionales	35
4.2.4. Requisitos no funcionales	37
4.3. Validación con prototipos	37
4.4. Implementación	38
4.4.1. Estructura de la aplicación	39
4.4.2. Sección Ranking	39
4.4.3. Sección Resumen	41
4.4.4. Sección Progreso	45

4.5. Despliegue	49
5. Conclusión y líneas de trabajo futuras	51
A. Descarga de los datos de mercado	53
A.1. Página de Fhalmeria	53
A.2. Script en VBScript	54
A.3. Macros	57
B. Requisitos funcionales	59
C. Prototipos iniciales	89
D. Despliegue de la aplicación	95

Índice de figuras

1.1. Cronograma con la planificación provisional	5
2.1. Ubicación de las dos estaciones climáticas	10
2.2. Zona invernada del Poniente almeriense (La Mojonera, El Ejido y alrededores). .	11
2.3. Histograma de la superficie de cultivo.	16
2.4. Histograma de los Kilogramos recolectados.	16
3.1. Barra de navegación en Shiny.	24
3.2. Panel de navegación en Shiny.	25
3.3. Pestañas en Shiny.	25
3.4. Flujo de intercambio de información en Shiny.	26
3.5. Gráfica generada con ggplot2.	29
4.1. Sección ranking de la aplicación.	40
4.2. Sección resumen de la aplicación.	42
4.3. Sección progreso de la aplicación.	46
4.4. Agrupación de variables con la técnica <i>pivot longer</i>	48
A.1. Fhalmeria. Pantalla 1 de la sección del histórico de mercados, con los elementos que intervienen en nuestro script <code>marcados en rojo</code>	53

A.2. Fhalmeria. Pantalla 2 de la sección del histórico de mercados, con los elementos que intervienen en nuestro script marcados en rojo.	54
C.1. Prototipo inicial para la página de ranking (1).	89
C.2. Prototipo inicial para la página de ranking (2).	90
C.3. Prototipo inicial para la página de resumen (1).	90
C.4. Prototipo inicial para la página de resumen (2).	91
C.5. Prototipo inicial para la página de resumen (3).	91
C.6. Prototipo inicial para la página de resumen (4).	92
C.7. Prototipo inicial para la página de progreso (1).	92
C.8. Prototipo inicial para la página de progreso (2).	93
C.9. Prototipo inicial para la página de progreso (3).	93
D.1. Dashboard de Shinyapps.	95

Resumen

En el presente documento se exponen los pasos seguidos para la construcción de un cuadro de inteligencia de negocio con *R* y la librería *Shiny*. El cuadro de mandos contiene información relativa a productos hortícolas e información climática, de la región geográfica del poniente almeriense. Desde la herramienta, se podrán consultar de forma rápida datos que requieren de un cálculo previo. Vamos a describir todos los pasos, desde la extracción de datos de fuentes heterogéneas hasta el desarrollo de la solución sobre la plataforma *R*.

La herramienta se encuentra desplegada en el entorno *shinyapps* y se puede acceder de forma pública.

Palabras clave: KDD, Dashboard, R, Agile.

Abstract

This document contains the steps followed to build a business intelligence chart with R and the Shiny library. The dashboard contains horticultural products and weather information from the western region of Almería. From the tool, data that requires a prior calculation can be quickly consulted. We are going to describe all the steps, starting with data extraction and ending with the development of the solution on the R platform.

The tool is deployed in the shinyapps environment and can be publicly accessed.

Keywords: KDD, Dashboard, R, Agile.

1

Introducción.

1.1. Motivación

Hoy en día conocer y estudiar los datos en el mundo empresarial puede resultar en una gran ventaja competitiva. El constante crecimiento de internet está relacionado de manera directa con el increíble aumento en la cantidad de datos que se generan. Según un informe de la consultora IDC [11], en 2025 el volumen de datos alcanzará los 175 zettabytes ($175 * 10^{21}$ bytes), que son 175 veces más datos que en el año 2011. La informática, mediante sus métodos y herramientas, debe ser capaz de dar solución a la obtención, tratamiento y estudio de datos con el objetivo final de proporcionar inteligencia en el ámbito de negocio.

Por otro lado, la provincia de Almería se ha convertido en la primera exportadora española de hortalizas en la campaña 2016/2017 en la que se obtuvieron las mejores cifras de comercialización y precios de todos los tiempos [7], superándose al cierre del año 2018 [8]. Este hecho ha generado una gran cantidad de datos durante el ciclo de vida completo de los productos hortícolas. Por tanto, nuestra línea de trabajo irá orientada en ese sentido.

Juntando estos dos factores, el objetivo de este trabajo será crear un cuadro de mandos con acceso a una serie de datos de producción, datos climatológicos y datos de subastas en la zona invernada del poniente almeriense. Todos estos datos son relevantes para los procesos de toma de decisiones dentro del sector hortofructícola. Desde el panel se podrán usar, visualizar y comparar datos históricos, usar esos datos en el presente e incluso, intentar realizar predicciones.

1.2. Objetivos

El principal objetivo del presente TFG es el desarrollo de un cuadro de inteligencia de negocio (*dashboard*) desde el cual se pueda consultar de forma dinámica información acerca de la

producción y comercialización de productos hortícolas en el poniente almeriense.

Algunos de los objetivos asociados son:

- Desarrollar un proyecto informático siguiendo una metodología de desarrollo, y con la ayuda de herramientas de la ingeniería del software.
- Adquirir conocimientos de inteligencia de negocio.
- Adquirir capacidades para resolver problemas y para obtener conclusiones a través de análisis previos.

Algunos objetivos específicos son:

- Estudiar y manejar el compilador R, el IDE R-Studio y la librería Shiny para elaborar interfaces gráficas.
- Crear una aplicación web.
- Construir informes profesionales a través de LaTeX.
- Trabajar con repositorios git para el control y la integración de cambios.

1.3. Planificación inicial

Vamos a utilizar un modelo de proceso basado en prototipos como modelo de ciclo de vida del software [15]. Tras el primer contacto con mi tutor, me expone la visión o el objetivo general del producto. Antes de comenzar con la toma de requisitos y el desarrollo, debemos realizar algunas tareas previas que se corresponden con el “conocimiento de los datos” [14]. A todo este proceso le dedicaremos una sección, pero no vamos a exponer aquí la estructura de la memoria, para ello tenemos el siguiente punto.

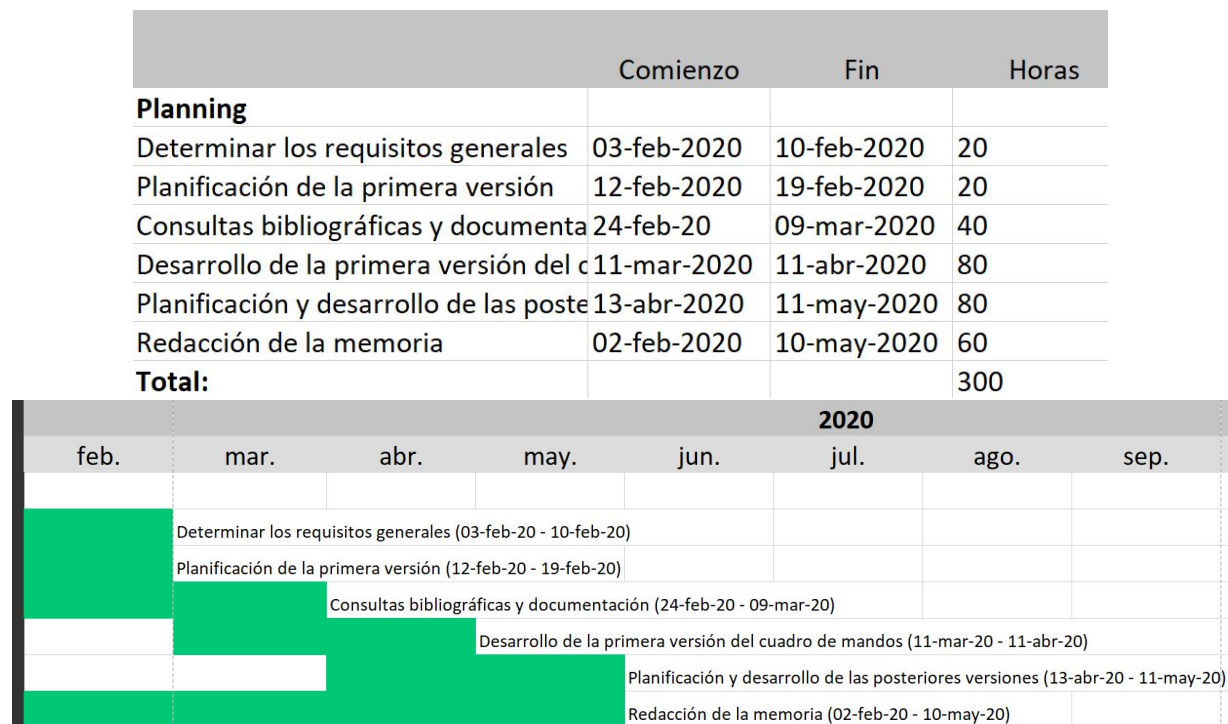
Tras esta primera clasificación de los datos, concertaremos una reunión con el cliente (en este caso mi tutor, José del Sagrado Martínez), de la que debemos obtener un subconjunto de requisitos del producto para así poder desarrollar el primer prototipo. Sucesivamente, tendremos reuniones de revisión y obtención de requisitos adicionales, desarrollando prototipos evolutivos y obteniendo *feedback* continuamente. Para la parte del desarrollo seguiremos una metodología ágil que nos permita desglosar los requisitos en tareas, estimar la duración de estas tareas y por último decidir cuáles de ellas se van a implementar en cada versión del prototipo.

Una vez que hayamos hecho todas las consultas bibliográficas necesarias acerca del lenguaje R y de otros cuadros de mandos similares, comenzaremos a trabajar en el primer prototipo. Tras completarlo, tendremos una reunión con el cliente para mostrarle el producto.

1. INTRODUCCIÓN.

En las etapas siguientes, seguimos el mismo método de trabajo, derivando el desarrollo en tantos prototipos como sea necesario, hasta obtener un producto robusto y fiable.

A modo de esquema mental, hemos elaborado un cronograma con los principales bloques de trabajo. Una posible planificación sería trabajar durante 4 horas diarias, 5 días a la semana, lo que resultaría en 20 horas de trabajo semanales.



Fuente: autoría propia

Figura 1.1: Cronograma con la planificación provisional

1.4. Estructura de la memoria

Esta memoria contiene 5 puntos principales:

- **Introducción:** En este primer capítulo se quiere exponer la naturaleza del proyecto, los objetivos que se persiguen, la planificación general y los materiales y métodos que vamos a utilizar.
- **Conociendo los datos:** En este capítulo se explica la metodología aplicada para conocer los datos, desde su extracción hasta la integración, pasando por la carga y transformación.
- **Elementos, tecnologías, modelo y metodología de la solución:** Este capítulo define las diferentes herramientas y procesos de los que se ha hecho uso en el desarrollo de la solución.

- Solución: Hacemos un recorrido por todas las fases de la solución, hasta el desarrollo del producto final.
- Conclusión y líneas de trabajo futuras: Hacemos una valoración final del proyecto y un breve análisis de las posibles mejoras que se puedan llevar a cabo.

1.5. Materiales y métodos

El lenguaje que vamos a utilizar será R para los análisis estadísticos, que es un lenguaje muy utilizado para el tratamiento de datos. Para los gráficos dinámicos de las interfaces de usuario utilizaremos Shiny, un paquete de R orientado a tales tareas.

Los materiales y herramientas que vamos a utilizar son los siguientes:

- Compilador de R, entorno de desarrollo integrado R Studio y la librería Shiny para interfaces gráficas.
- Repositorio en github.com.
- Overleaf para la redacción de la memoria en LaTeX.
- Desarrollo ágil.

2

Conociendo los datos.

Cuando se lleva a cabo un proyecto de este tipo, es muy importante seguir una serie de pasos antes de comenzar la minería de datos [21]. Estos pasos iniciales consisten en preparar los datos, y para esta preparación primero tenemos que obtenerlos, en un proceso que puede ser simple o puede complicarse en gran medida. En nuestro caso, nos enfrentaremos a las dos caras de la moneda y trabajaremos con técnicas como la automatización de tareas en una web, mediante el uso de macros [23]. Obtenidos los datos, deberemos estudiarlos minuciosamente para dar paso al siguiente punto que consistirá en transformarlos, eliminando el ruido y tratando de unificarlos lo máximo posible, esto es usar la misma terminología en datos de distinta índole y el mismo formato para valores que se pueden presentar de distintas maneras (como por ejemplo las fechas). Por último, deberán pasar por un proceso de agrupación o integración, que consiste en juntar los datos de la forma más compacta posible, para posteriormente comenzar con la carga [14].

2.1. Extracción y estudio

Vamos a trabajar con 3 conjuntos de datos, todos ellos relacionados con el sector hortícola. El objetivo es obtener la siguiente información:

- Datos de producción. Nos han sido cedidos por una entidad anónima del sector. La ubicación de la fuente es el poniente almeriense y el período es del 2013 al 2015. Fue la recepción de estos datos lo que nos motivó a estudiar esta zona geográfica. En principio los recibiremos por correo.
- Datos climatológicos. Tendremos que obtenerlos por nuestros propios medios. Deberemos localizar estaciones agroclimáticas ubicadas en la zona.
- Datos de mercado. También tendremos que obtenerlos por nuestra cuenta. Para ello exploraremos las webs públicas de subastas, tratando de encontrar un histórico de ventas.

Estos datos deben ser de la misma zona geográfica, el área invernada del poniente almeriense y obviamente del mismo período, relativo al trienio 2013-2015. Si no fuese así, no tendría ningún sentido estudiar los tres conjuntos de datos.

2.1.1. Datos de producción

Estos son los primeros datos con los que entramos en contacto. Los recibimos en hojas de cálculo de Excel (.xlsx) y se encuentran en dos documentos separados. El primero de ellos contiene datos de cultivo que se corresponden con el momento de la siembra. El archivo tiene varias páginas con las distintas campañas comprendidas en el período 2013-2015 y consta aproximadamente de 700 registros. Las variables que observamos son:

- Socio, de tipo entero
- Finca, de tipo entero
- Nave, de tipo entero
- Superficie, de tipo entero
- Cultivo, tipo texto
- Variedad, tipo texto
- Fecha, en formato dd/mm/AAAA

Los campos referentes al socio, a la finca y a la nave contienen un código numérico para identificarlos (a priori del número 1 al 18). La superficie entendemos que viene indicada en m^2 , aunque en las fases posteriores deberemos analizar los valores que toman para corroborar que se ajustan a la realidad. El cultivo y la variedad hacen referencia al género del producto. El cultivo indica el tipo de producto de forma más general mientras que la variedad es más específica, llegando a contener nombres científicos.

El segundo archivo tiene datos de producción del momento de la recolecta y los datos vienen en una única página. En este caso observamos aproximadamente 24.400 registros. Contiene las siguientes variables:

- CDPRINCIPAL, de tipo entero
- CDFINCA, de tipo entero
- CDNAVE, de tipo entero
- GÉNERO, de tipo texto

- CAMPAÑA, de tipo entero
- FECHA, en formato dd/mm/AAAA
- Sum(KILOSNETOS), de tipo entero

Los tres primeros atributos se corresponden con los que tenemos en los datos de cultivo. Para llegar a esta conclusión, hemos tenido en cuenta tres factores:

1. Los datos provienen de la misma fuente.
2. Para la finca y la nave, los nombres son casi homogéneos.
3. Los valores que toman Socio y CDPRINCIPAL son los mismos (1-18).

En este caso, el género del producto se indica mediante un único campo, siendo menos descriptivo que en los datos de la siembra. La campaña toma los valores 13, 14 o 15 y el peso viene indicado en Kilogramos.

2.1.2. Datos climatológicos

Llegados a este punto y habiendo estudiado la información de la que disponemos, el siguiente paso consiste en extraer los datos climatológicos del mismo período. Para ello, realizamos una búsqueda de las estaciones climáticas que hay en la zona. De esta forma, ubicamos dos estaciones en el Poniente almeriense. Estas son la estación de la RAIF (Red de Alerta e Información Fitosanitaria) [3] y la Estación Agroclimática [10] ambas de la Junta de Andalucía. En sus respectivas webs ofrecen información sobre sus coordenadas geográficas así como acceso a datos climatológicos históricos. Por tanto, accedemos y nos descargamos los datos necesarios.

Tras formatear los datos y pasarlos todos a Excel, de un vistazo eliminamos aquellas columnas que no contienen información o cuya información es constante. Este proceso de limpieza y filtrado lo seguiremos en el siguiente punto con un análisis más exhaustivo. Las variables resultantes comunes a ambos conjuntos de datos son:

- Temperaturas máxima
- Temperatura mínima
- Temperatura media
- Humedad máxima
- Humedad mínima

- Humedad media
- Radiación solar

Todos los valores son de tipo decimal. Las temperaturas vienen en $^{\circ}\text{C}$, las humedades en % y la radiación solar en MJ/m^2 .

Adicionalmente, la Estación Agroclimática dispone de algunos datos extra:

- Hora de la temperatura máxima ($HH : mm : ss$)
- Hora de la temperatura mínima ($HH : mm : ss$)
- Velocidad del viento media (m/s)
- Dirección del viento media ($0 - 360$)
- Precipitaciones (%)
- Evapotranspiración de referencia (mm/dia)

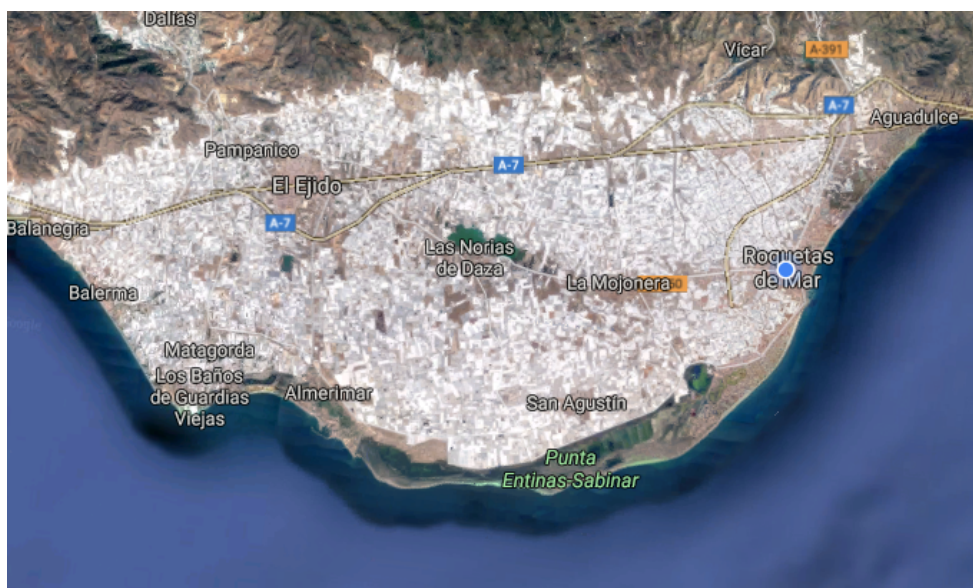


Fuente: autoría propia

Figura 2.1: Ubicación de las dos estaciones climáticas

2.1.3. Datos de mercado

Los datos de mercado los obtenemos de las subastas públicas que se hacen diariamente. La página que mantiene un histórico de estos datos es la de Fhalmería [6]. La web contiene información de mercado de más de 50 entidades agrícolas en distintas ubicaciones geográficas. A nosotros nos interesan las de la provincia de Almería y concretamente aquellas ubicadas en la zona invernada del Poniente.



Fuente: autoría propia

Figura 2.2: Zona invernada del Poniente almeriense (La Mojonera, El Ejido y alrededores).

Tras realizar un filtrado con ayuda de Google Maps, ubicando cada entidad agrícola en el mapa, nos quedamos con las siguientes:

- Agroejido
- Agroponiente Poniente
- Agrupaejido La Costa
- Agrupaejido La Redonda
- Agrupaejido Cortijos de Marín
- Cehorpa
- Costa de Almería

- Femago
- La Unión

El problema que nos encontramos a la hora de extraer los datos, es que están en formato de tabla en HTML. La solución que aplicamos es la automatización web mediante un script en Visual Basic y una herramienta para implementar macros denominada iMacros [9], que se podrán consultar en los anexos. Las columnas o variables que tienen los datos de mercado son:

- Producto
- Media 3 primeros
- Primer corte
- Sucesivos cortes

Producto almacena el género. La media de los 3 primeros se refiere a los 3 primeros cortes. Después hay varias columnas con los precios de todos los cortes. Cada nuevo corte tiene un precio inferior, seguramente porque la calidad también lo es.

2.2. Carga y transformación

Tal y como mencionan Wickham y Grolemund en su libro *R for Data Science* [19], el cual nos ha sido muy útil para toda esta sección, suele ser muy raro obtener los datos en la forma exacta en la que los necesitamos. A menudo se necesita crear nuevas variables, agrupaciones, renombrar aquellas ya existentes o incluso eliminarlas, todo esto con el objetivo final de transformar los datos a un formato con el que sea más fácil trabajar.

Para llevar a cabo todo este proceso de transformación utilizaremos R. Partimos desde el punto donde lo dejamos en la anterior sección, con los datos dispersos en distintos archivos de excel y algunos de ellos divididos en varias hojas dentro del mismo archivo. Hemos considerado que la manera más correcta de proceder en este caso sería creando un script en R para cada tipo de datos, y recoger en cada script las acciones que ejecutamos sobre el *dataframe*. De esta forma, tendríamos un histórico de todos los pasos que vamos dando y no dependeríamos del archivo *.RData* del *workspace*. Además, en cualquier momento podríamos volver a ejecutar el script en su conjunto y el resultado obtenido (*dataframe* resultante) sería el mismo. Esto sería muy útil si por ejemplo queremos deshacer algún paso o insertar alguna acción intermedia una vez que hemos ejecutado varios comandos sobre el *dataframe*. Es cierto que R almacena en un archivo *.RHistory* todos los comandos ejecutados, pero este archivo puede llegar a ser un auténtico caos y encontrar un paso específico puede hacerse muy costoso.

Nuestro objetivo no consiste en documentar paso por paso todas las acciones que hemos realizado sobre cada conjunto de datos, aunque sí que comentaremos alguna función importante. Para ello se pueden consultar los anexos y el repositorio de Git donde vamos a dejar todos los scripts. Lo que vamos a tratar de recopilar en esta sección son los aspectos más importantes que hemos considerado a la hora de trabajar los datos y el objetivo final que buscamos.

2.2.1. Carga

Para cargar los datos en R hemos utilizado 3 formas distintas, en función de cómo estaban organizados:

1. Mediante la función `read_excel()` nativa de R o la librería `xlsx` y la función `read.xlsx()`.
2. Mediante la función `read_excel_allsheets()` creada por nosotros, para los registros de cultivo que estaban dispersos en varias hojas dentro del mismo archivo `.xlsx`.

```
1 read_excel_allsheets <- function(filename, tibble = FALSE) {
2   sheets <- readxl::excel_sheets(filename)
3   x <- lapply(sheets, function(X) readxl::read_excel(filename, sheet = X))
4   if(!tibble) x <- lapply(x, as.data.frame)
5   names(x) <- sheets
6   return(x)
7 }
8
```

3. Mediante la función `lectura_datos_mercado()` también definida por nosotros para cargar los más de 6.500 archivos `.csv` de mercado. Esta función básicamente recorre todos los `.csv` de un directorio y los va almacenando en un `dataframe`.

```
1 lectura_datos_mercado <- function() {
2   setwd("data/merc")
3   archivos = list.files(pattern="*.csv")
4   mercado = read.csv(file=archivos[1], header=TRUE, sep=";")
5   for (i in 2:length(archivos)) {
6     mercado_temp = tryCatch(read.csv(file=archivos[i], header=TRUE, sep=";"),
7       error=function(e) NULL)
8     mercado <- rbind(mercado, mercado_temp)
9   }
10  setwd("../..")
11  return(mercado)
12 }
```

2.2.2. Preprocesamiento

Con los datos ya cargados y almacenados en los correspondientes `dataframes`, comenzamos a trabajar sobre ellos. Lo primero que haremos será limpiarlos y seleccionarlos. Para el proceso de

limpieza, hemos seguido en mayor o menor medida para cada *dataframe* los siguientes pasos:

1. Concentrar el tipo de producto en una única variable. En alguno de los conjuntos de datos, como por ejemplo los de cultivo, el tipo de producto se encontraba disperso en dos variables (Género y Producto). Lo que hemos hecho en estos casos es juntarlo todo en una variable y de esta forma preparamos el terreno para facilitar el filtrado y la integración posteriores.
2. Eliminar las variables que no vayamos a considerar para el problema. Este punto es complicado de abordar ya que es difícil ajustar el problema a las restricciones de tiempo y recursos debido a que casi todas las variables aportan información y con todas podríamos trabajar para extraer conocimiento. Teniendo en cuenta esto, hemos decidido prescindir de la siguiente información:
 - Finca, Socio y Nave en los datos de cultivo y producción. Hemos tomado esta decisión porque no disponemos de esta información en los datos de mercado y porque si en un futuro decidimos medir la rentabilidad de la producción lo vamos a hacer de forma genérica para todo el conjunto de datos.
 - Por otro lado, en los datos de mercado nos hemos quedado únicamente con los precios del primer corte (Corte1) porque la información de los sucesivos cortes está incompleta.
 - En los datos climáticos de la Estación Agroclimática hemos eliminado las horas de las temperaturas máxima y mínima, porque los vamos a cruzar con los datos de la estación de la RAIF, así como la dirección del viento ya que no registraba valores correctos y a menudo estaba incompleta, además tampoco nos aporta información valiosa porque los cultivos son en invernadero.
 - Otras variables residuales de las que se puede prescindir como puede ser la campaña (12, 13, 14...) porque ya viene indicada en la fecha, o el día (1-365) por el mismo motivo.

Es importante conservar solamente las variables que vamos a estudiar, porque nos facilita la integración de los datos.

3. Transformar las variables conservadas. Primero establecemos una nomenclatura homogénea para aquellas variables que hemos decidido conservar y después convertimos cada variable a su tipo de dato correcto (Date, Numeric, Factor...) ya que al importar los datos, algunos tipos no se han conservado.
4. Transformar y filtrar los registros. De partida, tenemos muchos registros con el nombre del producto distinto pero que en realidad pertenecen a la misma variedad o a variedades casi idénticas en precio y características, que es lo que nos interesa. Esto nos complica mucho el problema a la hora de tener que integrar los datos, ya que tenemos demasiados tipos de producto y son muy pocos aquellos que se encuentran en los 3 *dataframes* (cultivo, producción y mercado). Esto se debe a diversos factores, entre los que destacan la presencia de géneros casi idénticos pero con ligeras diferencias a nivel genético o estructural, fuentes de datos dispersas, o errores ortográficos. Para resolver este problema, al no tener contacto con el origen de los datos, hemos actuado de la siguiente forma:

- a) Hemos generado para cada *dataframe* una tabla de frecuencia, en la que se refleja el número de veces que aparece cada tipo de producto.

```
1 #Frecuencia de cada variedad de mercado
2 frec_merc = table(mercado$Producto)
3 frec_merc = as.data.frame(frec_merc)
4 frec_merc = frec_merc[order(-(frec_merc$Freq)),]
5 View(frec_merc)
6
```

- b) Comparando cada tabla, hemos analizado primero aquellos productos más frecuentes.
- c) Con la ayuda del buscador de la web de Agrovademecum [1], hemos investigado acerca de cada producto extraído de la tabla obteniendo el nombre genérico y comprobando como en algunas ocasiones dos productos de la tabla eran en realidad del mismo género.
- d) Hemos creado una nueva columna y le hemos asignado el nombre genérico de esos productos siguiendo la misma nomenclatura para todos.

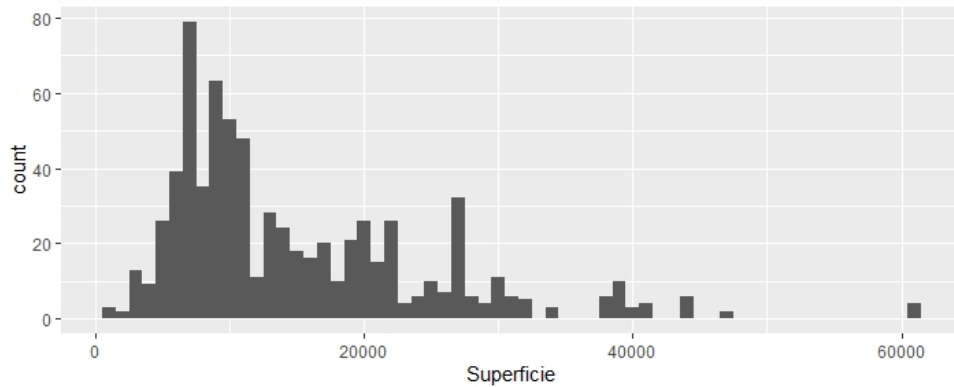
```
1 #Renombrar datos de cultivo (ejecutamos para cada modificación)
2 for (i in 1:nrow(cultivo)){
3   if(grepl("valverde", cultivo[i, 3])){#string contains
4     print(cultivo[i, 3])
5     cultivo[i, 4] <- "Melón Piel de sapo"
6   }
7 }
8
```

- e) Repetimos el proceso para aquellos productos menos frecuentes.
- f) Aquellos registros muy poco frecuentes (<10) y pertenecientes a géneros muy diferentes y específicos los hemos eliminado tras comprobar que disponemos de registros suficientes como para que el problema no se vea afectado.
- g) Hemos conservado en el nombre los indicadores de calidad, que son “E”, “I” y “II”. La “E” se refiere a “Extra” y el “I”y “II” son los niveles de calidad posteriores [4].

2.2.3. Transformación

Después de haber limpiado y seleccionado los datos en los procesos expuestos previamente comenzamos a trabajar en la transformación. Entendemos por transformación la consolidación de los datos en formas apropiadas para la tarea de minería, mediante la realización de operaciones de resumen y/o agregación [14]. Lo que vamos a hacer en este punto será agrupar los datos por semanas para facilitar la integración. Y, aparte de eso, no tiene sentido trabajar con los datos por días, ya que no podemos extraer conclusiones.

Para agrupar por semanas, primero analizamos la distribución de los datos mediante histogramas. En el caso de los datos de cultivo, no observamos valores atípicos.



Fuente: autoría propia

Figura 2.3: Histograma de la superficie de cultivo.

Por tanto, agrupamos los datos por semanas por producto.

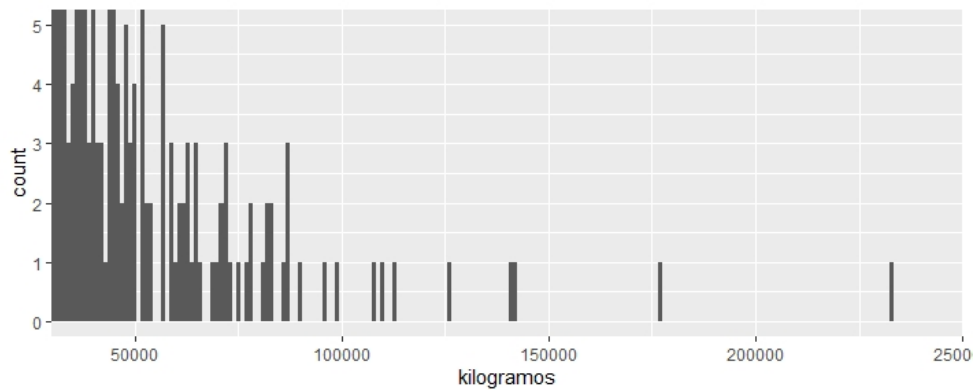
```

1 cultivo_group <- cultivo %>%
2   mutate(fecha_semana = lubridate::floor_date(fecha_plantacion, 'week')) %>%
3   group_by(producto_f, fecha_semana) %>%
4   summarise(superficie_semana = sum(Superficie)) %>%
5   arrange(fecha_semana, desc(producto_f))

```

Destacar que calculamos la superficie total cultivada sumando todas las superficies cultivadas en una semana para cada producto determinado.

Observamos la distribución de los datos de producción con respecto a los Kg recolectados para detectar posibles errores.



Fuente: autoría propia

Figura 2.4: Histograma de los Kilogramos recolectados.

2. CONOCIENDO LOS DATOS.

Esta comprobación nos genera dudas, vemos algunos valores muy altos, valores atípicos. En estos casos las recomendaciones [19] son:

1. O bien eliminar la fila completa.
2. O bien reemplazar la celda por un valor desconocido *N.A.* (*Not available*).

Para eliminar la fila completa tenemos que comprobar que no se pierden otras variables importantes y además tenemos que comprobar que disponemos de registros suficientes para que cualquiera de los métodos seguidos no afecte negativamente al problema. Nosotros finalmente optamos por otra solución.

```
1 produccion_group <- produccion %>%
2   mutate(fecha_semana = lubridate::floor_date(as.Date(fecha_produccion), 'week')) %
   >%
3   group_by(fecha_semana, producto_f) %>%
4   summarise(total_kg_semana = sum(kilogramos),
5             min_kg_semana = min(kilogramos, na.rm = TRUE),
6             q1_kg_semana = quantile(kilogramos, 0.25, na.rm = TRUE),
7             med_kg_semana = median(kilogramos, na.rm = TRUE),
8             mean_kg_semana = mean(kilogramos, na.rm = TRUE),
9             q3_kg_semana = quantile(kilogramos, 0.75, na.rm = TRUE),
10            max_kg_semana = max(kilogramos, na.rm = TRUE)) %>%
11   arrange(fecha_semana, desc(producto_f))
```

Lo que hacemos es, aparte de sumar los Kilogramos de una semana determinada, calcular también el mínimo, el primer cuartil, la mediana, la media, el tercer cuartil y el máximo. Con estos valores conocemos el rango (max - min) donde se encuentran todos los valores, el rango (intercuartílico) donde está la mitad central de los valores Q2-Q3 y, con la mediana (Q2) cuál es el valor que divide en dos mitades los valores (la mitad está por debajo y la otra por encima). La media es sensible a los valores extremos y nos sirve como indicador al compararla con la mediana, de la dispersión de los datos. Los 6 valores anteriores reflejan la dispersión de la variable en una semana y nos dan una idea aproximada de su tendencia.

Lo mismo hacemos con los datos de mercado, que también tienen valores atípicos para el precio, solo que esta vez no sumamos los importes para una semana sino que calculamos directamente la media, además de todas las medidas indicadas anteriormente.

Los datos climáticos, al tenerlos en dos *dataframes* distintos, tendremos que cruzarlos previamente, pero antes analizamos cada *dataframe* por separado para comprobar que no haya errores. Para integrarlos, las variables que solamente están en uno de ellos las mantenemos sin ninguna alteración y para las que están en ambos calculamos la media de los valores cuando disponemos de ambas muestras, en el caso contrario conservamos la muestra de la que disponemos. De esta forma, los datos climáticos se complementan y son más fiables.

```

1 clima_cruzado <- clima_estacion_agro %>%
2   full_join(clima_raif, c("FECHA" = "Fecha")) %>%
3   rowwise() %>%
4   transmute(fecha = FECHA,
5             temperatura_minima = mean(c(Temperatura.mínima.C.x, Temperatura.mí
6             nima.C.y), na.rm=TRUE),
7             temperatura_maxima = mean(c(Temperatura.máxima.C.x, Temperatura.má
8             xima.C.y), na.rm=TRUE),
9             temperatura_media = mean(c(Temperatura.media, Temperatura.media.C),
10            na.rm=TRUE),
11            humedad_minima = mean(c(Humedad.mínima, Humedad.mínima..), na.rm=TRUE)
12            ,
13            humedad_media = mean(c(Humedad.media, Humedad.media..), na.rm=TRUE),
14            humedad_maxima = mean(c(Humedad.máxima, Humedad.máxima..), na.rm=TRUE)
15            ,
16            radiacion_solar = mean(c(Radiación.solar, Radiación.solar.MJ.m2), na.
17            rm=TRUE),
18            velocidad_viento = Velocidad.del.viento,
19            precipitaciones = Precipitaciones,
20            evapotranspiracion_referencia = Evapotranspiración.de.referencia)

```

Después de esto los agrupamos por semanas, al igual que hemos hecho con los otros datos. Para cada variable calculamos todo el rango de valores igual que antes.

2.2.4. Integración

Después de haber preparado todos los *dataframes*, el último paso consiste en integrar todos los datos en un solo conjunto. Este proceso es bastante simple al haber dejado todos los datos bien depurados y agrupados. Lo que hacemos son sucesivos full joins, primero de los datos de cultivo con los de producción. El resultado lo cruzamos con los datos de mercado, y por último el resultado lo cruzamos nuevamente con los datos climáticos. Los joins los hacemos con el producto y la fecha como claves. El proceso es muy parecido a lo que hicimos con los datos climáticos.

3

Elementos, tecnologías, modelo y metodología de la solución.

En el capítulo anterior describimos todo el proceso de tratamiento de los datos, desde la extracción hasta la obtención del *dataset* final. Este capítulo lo vamos a dedicar a describir todas las entidades, desde herramientas hasta metodologías, que nos han servido en el desarrollo del proyecto.

3.1. Dashboards

3.1.1. ¿Qué es un dashboard?

Un *dashboard*, o cuadro de mandos es una interfaz gráfica que nos proporciona indicadores de interés relacionados con un objetivo específico o con un proceso de negocio, de forma visual, para gestionar la inteligencia de negocio. Lo que se busca con la consulta a un *dashboard* es facilitar la toma de decisiones en el ámbito empresarial. Los indicadores que nos proporciona el cuadro de mandos a menudo nos permiten evaluar el éxito en una organización o en alguna de las actividades que desempeña.

La gran ventaja que aporta un cuadro de mandos es el disponer de todos los datos centralizados y sin necesidad de realizar cálculos intermedios. Esto ahorra tiempo y esfuerzo que puede ser invertido en analizar estos datos y definir estrategias de negocio [20]. Algunos beneficios adicionales son:

- Presentar visualmente las medidas de rendimiento.
- Identificar y corregir tendencias negativas.

- Medir eficiencias e ineficiencias.
- Generar informes detallados que muestren tendencias.
- Tomar decisiones respaldadas en datos y basadas en inteligencia empresarial.
- Alinear las estrategias con los objetivos de la organización.
- Identificar datos atípicos y correlaciones.
- Generar informes detallados que muestren tendencias.
- Consolidar los informes en una ubicación.

3.1.2. Intelindalo

El *dashboard* que vamos a construir, Intelindalo, está enfocado al sector agroalimentario del poniente almeriense. Toda la información que se va a mostrar es relevante para la toma de decisiones dentro del sector hortofrutícola. Desde el cuadro de mandos se podrán usar, visualizar y comparar datos históricos, usar esos datos en el presente e incluso intentar realizar predicciones.

Este cuadro de mandos tendrá 3 secciones:

- La sección inicial denominada *Ranking* tendrá una serie de gráficas que expondrán de forma simple los 5 mejores y los 5 peores productos para cada una de las variables de las que disponemos o de las que podemos calcular. Por ejemplo, la superficie cultivada, el rendimiento del producto o el precio.
- La sección *Resumen* tomará una serie de entradas del usuario relacionadas con el producto y el periodo temporal y mostrará un resumen de forma gráfica y en formato de tabla. De nuevo el resumen se podrá ver para cada una de las variables de las que disponemos.
- La tercera sección, *Progreso*, mostrará información evolutiva (crecimiento en el tiempo) acerca de la variable (superficie cultivada, precio, temperatura, etc.) y el producto que escoja el usuario. Se podrá ver la evolución de toda la distribución de los datos.

3.2. Tecnologías y herramientas

Para construir la solución final hemos hecho uso de múltiples herramientas, algunas de ellas en etapas concretas del proyecto y otras a lo largo de todo el ciclo de vida.

3.2.1. Planner

Para la planificación del proyecto hemos utilizado Planner, tanto para planificar las fases del proyecto a grandes rasgos, como para la fase de desarrollo. Planner es una herramienta de Microsoft que permite construir tableros de tipo Kanban y que tiene un alto nivel de personalización. Planner permite crear planes, definir y asignar tareas y recibir actualizaciones de progreso.

3.2.2. iMacros

Para la obtención de los datos de mercado hemos utilizado iMacros, que es una aplicación utilizada para la creación, grabación, edición y reproducción de macros para automatización y pruebas web. Esta aplicación se instala como un *plugin* del navegador. Esta aplicación se puede utilizar de dos formas distintas. Por un lado, se puede hacer uso de la interfaz que tiene para definir una secuencia de acciones sobre una página web. La aplicación en este modo puede ser muy útil para grabar y reproducir acciones simples que no requieren de almacenamiento de datos o de una lógica personalizada. En nuestro caso, hemos optado por la segunda forma, que consiste en construir un script en Visual Basic haciendo llamadas a la librería de la aplicación, consiguiendo una automatización a medida y con una lógica personalizada.

3.2.3. Blasamiq Cloud

Para construir los primeros prototipos, se ha utilizado Balsamiq Cloud, que es una herramienta para construir mockups. Los siguientes prototipos se han construido directamente con *Shiny*.

3.2.4. R y Rstudio

Para hacer el tratamiento de los datos y para construir el cuadro de mandos se ha utilizado R y RStudio. R es un lenguaje de programación enfocado a la computación estadística y los gráficos. Hoy en día se utiliza mucho en campos como las matemáticas o la ciencia de datos. R se encuentra bajo licencia GNU como software libre [17]. Una de las ventajas principales de R es la escalabilidad que tiene, puesto que se le pueden importar una gran variedad de paquetes para extender su funcionalidad.

RStudio es un IDE moderno para programar en R [22]. Dispone de varias secciones: consola, editor de sintaxis, herramientas para el trazado, depuración y gestión del espacio de trabajo. Tiene algunas funcionalidades destacadas como por ejemplo la consulta de documentación desde el propio IDE (apartado *Help*) o la exposición de variables y funciones en el panel lateral (*Environment*).

En la actualidad hay múltiples herramientas para la construcción de cuadros de mandos [5]

pero hemos decidido utilizar R porque cubría con todas las necesidades que teníamos tanto para el preprocesamiento, conocimiento y tratamiento de los datos como para desarrollar el cuadro de mandos.

3.2.5. Shiny y Shinydashboard

Adicionalmente, hemos utilizado múltiples paquetes de R. Los paquetes de R enriquecen el lenguaje aportando librerías para distintos fines. De todos los paquetes que hemos utilizado, los más destacados son Shiny y Tidyverse. Shiny es una librería que podemos definir como un *framework* web para R [2]. Es una herramienta para construir aplicaciones con *bootstrap* sin tener grandes conocimientos de esta tecnología. Shiny genera principalmente código *HTML* aunque existe la posibilidad de añadir scripts en *Javascript* para conseguir una web más dinámica.

Las aplicaciones en *Shiny* separan el *back end* del *front end* en dos archivos: *server.R* y *ui.R*. El archivo *ui.R* genera la interfaz de usuario. Dentro de este archivo se definen todos los elementos visuales, tanto los de entrada como los de salida de datos. Las posibles entradas de datos son:

- Botón: botones de acción
- *Checkbox*: simple o múltiple.
- Fecha: para indicar una fecha.
- Rango de fechas: para indicar una fecha de inicio y una fecha de fin.
- Archivo: opción de adjuntar y subir archivos.
- Valor numérico: campo de entrada de números.
- Texto: campo de entrada de texto.
- Selector de opciones: que puede ser una lista desplegable o una serie de botones circulares.
- *Slider*: tanto para un único valor como para un rango.

Shiny utiliza las entradas para renderizar las posibles salidas de datos. Los elementos que se pueden mostrar en Shiny son los siguientes:

- Imágenes.
- Tablas.
- Texto.
- HTML.

- Gráficas.

Por tanto, la forma de construir la interfaz de usuario consiste en distribuir las distintas entradas y salidas de datos en distintos paneles. Hay una gran variedad de paneles, entre los que destacan los siguientes:

- *absolutePanel()*.
- *conditionalPanel()*.
- *fixedPanel()*.
- *headerPanel()*.
- *inputPanel()*.
- *mainPanel()*.
- *navlistPanel()*.
- *sidebarPanel()*.
- *tabPanel()*.
- *tabsetPanel()*.
- *titlePanel()*.
- *wellPanel()*.

Para colocar los paneles, disponemos de varios tipos de contenedores, que a su vez pueden estar dentro de otros contenedores, resultando en estructuras más complejas. Los tipos principales de contenedores son:

- Contenedor fila (*fluidRow()*): esta forma de organización consiste en definir una o más filas de 12 puntos de tamaño. Dentro de cada fila, se pueden establecer columnas de distintos tamaños. Las columnas deben sumar 12 puntos de anchura para crear una malla de contenedores. Dentro de cada columna se pueden insertar otros contenedores o elementos.

```
1 ui <- fluidPage(fluidRow(column(width = 4), column(width = 2, offset = 3)),  
2                 fluidRow(column(width = 12)))  
3
```

- Contenedor malla regular (*flowLayout()*): este contenedor crea otros contenedores de igual tamaño y los distribuye de manera uniforme.

```
1 ui <- fluidPage(flowLayout(#object 1, #object 2, #object 3))  
2
```

- Contenedor con panel lateral (*sidebarLayout()*): divide la interfaz en dos paneles, uno principal y otro secundario. En el panel principal se suelen colocar las salidas de datos, y en el panel secundario (normalmente de menor tamaño) se distribuyen las entradas de datos.

```
1 ui <- fluidPage(sidebarLayout(sidebarPanel(), mainPanel()))  
2
```

- Contenedor con panel lateral (*sidebarLayout()*): divide la interfaz en dos paneles, uno principal y otro secundario. En el panel principal se suelen colocar las salidas de datos, y en el panel secundario (normalmente de menor tamaño) se distribuyen las entradas de datos.

```
1 ui <- fluidPage(splitLayout(#object 1, #object 2))  
2
```

- Contenedor con dos paneles (*splitLayout()*): este tipo de contenedor tiene dos paneles iguales. Por tanto, la interfaz se divide en dos partes.

```
1 ui <- fluidPage(sidebarLayout(sidebarPanel(), mainPanel()))  
2
```

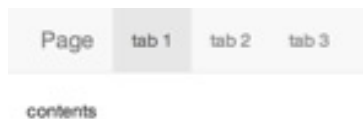
- Contenedor vertical (*verticalLayout()*): los elementos que lo integran se apilan verticalmente.

```
1 ui <- fluidPage( verticalLayout(# object 1, # object 2, # object 3 ))  
2
```

Todos estos contenedores se pueden colocar unos dentro de otros, pero también existen una serie de contenedores más grandes que se utilizan para navegar a través de la aplicación. Existen 3 tipos de contenedores de navegación:

- Barra de navegación (*navbarPage()*).

```
1 ui <- navbarPage(title = "Page", tabPanel("tab", "contents"), tabPanel("tab_2",  
  "contents"), tabPanel("tab_3", "contents"))  
2
```



Fuente: <https://shiny.rstudio.com/>
Figura 3.1: Barra de navegación en Shiny.

- Panel de navegación (*navlistPanel()*).



Fuente: <https://shiny.rstudio.com/>

Figura 3.2: Panel de navegación en Shiny.

```
1 ui <- navlistPanel(tabPanel("tab", "contents"), tabPanel("tab_2", "contents"),  
2   tabPanel("tab_3", "contents"))
```

■ Pestañas (*tabsetPanel()*).

```
1 ui <- fluidPage(tabsetPanel(tabPanel("tab_1", "contents"), tabPanel("tab_2", "  
2   contents"), tabPanel("tab_3", "contents")))
```



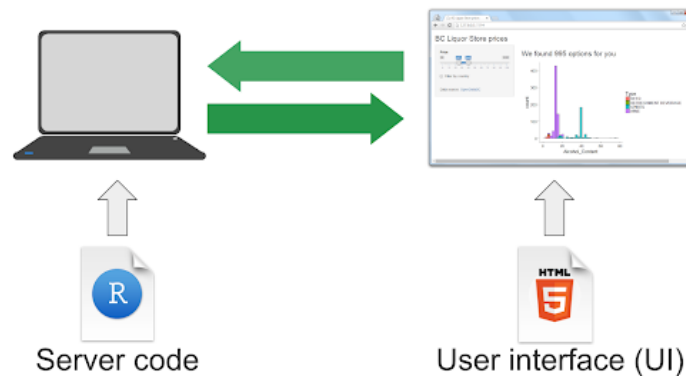
Fuente: <https://shiny.rstudio.com/>

Figura 3.3: Pestañas en Shiny.

El cálculo de los datos se hace dentro del archivo *server.R* que recibe los valores de las entradas, los procesa y realiza los cálculos y genera los elementos de salida.

Para generar los elementos de salida, hacemos llamadas a distintas funciones desde *server.R* dependiendo del tipo de elemento que queremos renderizar:

- *renderImage()*: para renderizar una imagen.
- *renderPlot()*: para renderizar una gráfica.
- *renderPrint()*: para renderizar texto de cualquier elemento.
- *renderTable()*: para renderizar una tabla.
- *DT::renderDataTable()*: para renderizar una tabla con el paquete DT.



Fuente: <https://deanattali.com/>

Figura 3.4: Flujo de intercambio de información en Shiny.

- `renderText()`: para renderizar texto.
- `renderUI()`: para renderizar una interfaz completa.

Para el desarrollo de la aplicación hemos hecho uso en múltiples ocasiones de la función `renderUI()` ya que nos permite crear una interfaz de forma dinámica desde el servidor. Esto ha sido muy útil cuando alguna de las entradas de datos requería de cálculos previos y/o cuando esos cálculos requerían de la interacción del usuario con otras entradas.

A continuación podemos ver un ejemplo de una aplicación simple en Shiny.

```

1 library(shiny)
2
3 # Definición de la IU de una aplicación que muestra un histograma —
4 ui <- fluidPage(
5
6   # Título de la app —
7   titlePanel("Hola Shiny!"),
8
9   # Contenedor con panel lateral con la definición de las entradas y salidas de
   # datos —
10  sidebarLayout(
11
12    # Panel lateral para las entradas —
13    sidebarPanel(
14
15      # Entrada: slider para seleccionar el número de elementos —
16      sliderInput(inputId = "bins",
17                  label = "Número de elementos:",
18                  min = 1,
19                  max = 50,

```

```
20         value = 30)
21     ),
22     # Panel principal para mostrar la salida —
23     mainPanel(
24         # Salida: histograma —
25         plotOutput(outputId = "distPlot")
26     )
27 )
28 )
29 )
30 )
31 )
32 )

1 # Definición de la lógica del servidor para construir un histograma —
2 server <- function(input, output) {
3
4     # Histograma sobre el dataset Old Faithful Geyser —
5     # con el número de elementos como entrada
6     # La expresión que genera el histograma está encapsulada en una llamada
7     # a renderPlot para indicar que:
8     #
9     # 1. Es reactiva y por tanto debería re-ejecutarse
10    #     automáticamente cuando las entradas (input$bins) cambiasen
11    # 2. El tipo de salida es una gráfica
12
13    output$distPlot <- renderPlot({
14
15        x <- faithful$waiting
16        bins <- seq(min(x), max(x), length.out = input$bins + 1)
17
18        hist(x, breaks = bins, col = "#75AADB", border = "white",
19             xlab = "Tiempo de espera para la siguiente erupción (en minutos)",
20             main = "Histograma con los tiempos de espera")
21    })
22 }
23
24 }
```

En el servidor se renderiza la gráfica de la siguiente forma:

```
1 output$distPlot <- renderPlot({})
```

Las funciones de *Shiny* utilizadas para renderizar las salidas son en su mayoría reactivas, por tanto cualquier cambio en las entradas referenciadas dentro de ellas provoca un nuevo renderizado del elemento.

Shiny es un paquete orientado a construir interfaces gráficas que no tienen que tener necesariamente una estructura de *dashboard*. Es por ello que, aunque la librería de elementos de Shiny es muy completa, en ocasiones podemos necesitar componentes o funcionalidades adicionales.

Shinydashboard es un paquete que funciona con Shiny y está enfocado a la creación de *dashboards*. Este paquete, que extiende a Shiny, nos proporciona una gran variedad de componentes visuales adicionales muy utilizados en la creación de cuadros de mando [13]. La estructura de la interfaz de un proyecto con *shinydashboard* cambia ligeramente, agregando una capa superior para la estructura de la página sobre la que se insertarán los componentes.

```
1 ## ui.R ##
2 library(shinydashboard)
3
4 dashboardPage(
5   dashboardHeader(),
6   dashboardSidebar(),
7   dashboardBody()
8 )
```

3.2.6. Tidyverse

Tidyverse es una colección de paquetes de R enfocados a la ciencia de datos que facilita y agiliza la escritura de código, así como la documentación y el mantenimiento [12]. De todos los paquetes de Tidyverse, hemos utilizado principalmente dplyr para el cálculo y la manipulación de los datos y ggplot2 para la presentación gráfica.

Dplyr se define como una gramática de manipulación de datos, proporciona un conjunto consistente de verbos que lo ayudan a resolver los desafíos más comunes a la hora de tratar los datos [19]. Las operaciones que realizamos con dplyr son las siguientes:

- *mutate()*: añade nuevas variables que son funciones de las ya existentes.
- *select()*: selecciona variables en función de sus nombres.
- *filter()*: filtra registros en función de sus valores.
- *summarise()*: reduce múltiples valores a un solo valor aplicando una operación determinada.
- *arrange()*: cambia el orden de las filas.

Todas estas operaciones se pueden combinar con *group_by()* que permite operar con grupos de registros. A continuación podemos ver un ejemplo, extraído de la aplicación, de cómo funcionan todas estas operaciones en conjunto:

```
1 subdata <- agrodata %>%
2   filter(!is.na(total_kg_semana)&!is.na(mean_cortel_semana)) %>%
3   filter(if(product == "Todos") TRUE else producto == product) %>%
4   filter(between(ano, period_f, period_l)) %>%
5   mutate(rentabilidad_semana = total_kg_semana * mean_cortel_semana) %>%
6   group_by(filtro = if(product == "Todos") producto else get(applied_filter)) %
  >%
```

```

7   summarise(rentabilidad = mean(rentabilidad_semana)) %>%
8   mutate_if(is.numeric, round, roundDecimals)

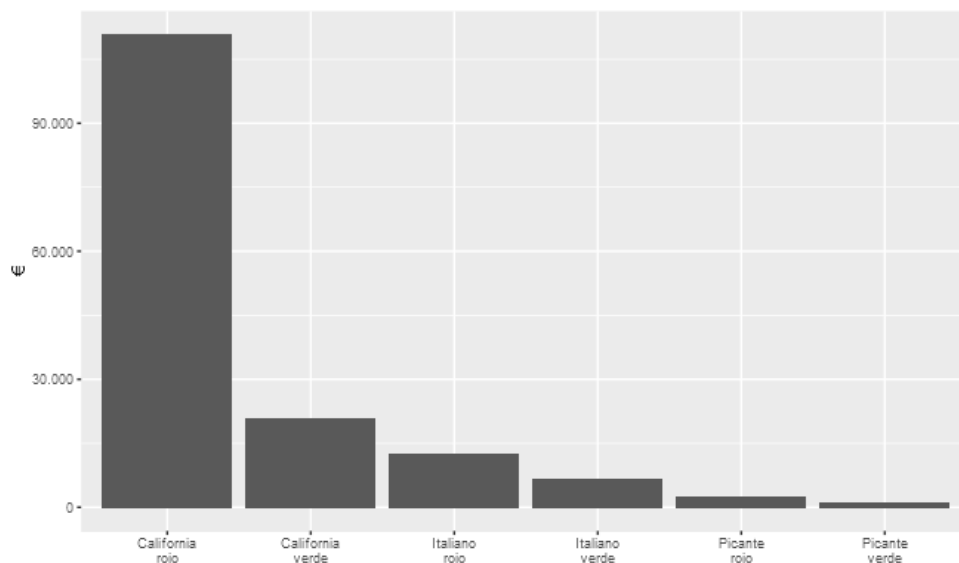
```

Ggplot2 es un paquete para producir gráficos estadísticos o de datos, pero es diferente a la mayoría de paquetes gráficos porque tiene una gramática subyacente profunda. Se compone de un conjunto de elementos independientes que se pueden componer de muchas maneras distintas [18]. La potencia de ggplot2 reside en que no se limita a un conjunto de gráficos preestablecidos sino que permite crear nuevos gráficos que se adapten con precisión a nuestro problema. Ggplot2 está diseñado para funcionar de forma iterativa, lo que quiere decir que podemos comenzar añadiendo una capa que muestre por ejemplo los datos sin procesar y, a continuación (o en cualquier otro momento), agregar capas de anotaciones y resúmenes estadísticos. Podemos ver a continuación un ejemplo junto con el gráfico generado:

```

1 subdata %>%
2   ggplot(mapping = aes(x=reorder(gsub(" ", "\n", filtro), -rentabilidad), y=
   rentabilidad)) +
3     geom_bar(stat = "identity") +
4     labs(x="", y = "€") +
5     scale_y_continuous(
6       labels = scales::number_format(big.mark = ".", decimal.mark = ",")
7     )+
8     {if (nrow(subdata)>=10){coord_flip()}}+
9     {if (nrow(subdata)>=20){scale_x_discrete(labels = abbreviate)}}

```



Fuente: autoría propia

Figura 3.5: Gráfica generada con ggplot2.

3.3. Modelo y metodología de desarrollo

Al ser un proyecto en el que la comunicación y las reuniones con el cliente han sido constantes y habiendo solamente una persona desarrollando, hemos decidido seguir un modelo de proceso basado en prototipos evolutivos. Partiendo de la especificación inicial y del análisis de requisitos se han construido los primeros prototipos como mock-ups en Balsamiq Cloud. Tras revisarlos, se han trasladado a la interfaz de Shiny con las primeras indicaciones por parte del cliente. Sucesivamente la interfaz ha ido evolucionando en los distintos ciclos de desarrollo, hasta llegar a la interfaz final aprobada por el cliente. Los primeros prototipos se pueden consultar en los apéndices.

El uso de este modelo mejora el entendimiento común de los requisitos al principio del desarrollo, evitando que los fallos en la interpretación propaguen errores en el código. También facilita la estimación de los plazos y el esfuerzo necesario para el proyecto. El modelo es muy útil para alcanzar con el cliente lo más pronto posible una visión compartida del proyecto y conforme este avance, será el propio cliente el que guíe el desarrollo [15].

Para el desarrollo de la aplicación se ha utilizado una metodología ágil adaptada a la naturaleza del proyecto. Se han dividido los requisitos en tareas más pequeñas con el objetivo de hacer planificaciones y revisiones semanales. Para organizar las tareas hemos utilizado un tablero construido con Microsoft Planner.

4

Solución.

En este capítulo abordaremos todas las fases del ciclo de vida del producto que vamos a construir. Antes de comenzar con el desarrollo de la solución, vamos a dedicar un par de secciones a todas las actividades previas. Trataremos de definir muy bien el producto mediante una especificación inicial. Haremos un análisis de esta especificación del cual obtendremos los requisitos de partida. Con estos requisitos, esbozaremos los primeros prototipos.

Es importante mencionar que ni mucho menos trataremos de definir una versión final del producto. En cualquier momento puede haber cambios motivados por el *feedback* obtenido de los prototipos, o incluso de nuevas ideas que nos vayan surgiendo. Lo importante es visibilizar esos cambios, es decir, reflejar la evolución del sistema como consecuencia del modelo basado en prototipos.

4.1. Definición del sistema

Se desea construir una aplicación web para exponer información sobre productos hortícolas en la zona del poniente almeriense. Partimos de un conjunto completo de datos en el que podemos encontrar información semanal de cultivo (superficie cultivada), información de producción (kilogramos recolectados), información de mercado (precio de venta en subasta) e información climatológica, todo ello relativo al período 2012-2015.

Para la información de producción, la de mercado y la climatológica disponemos de toda la distribución de los datos agrupados por semana (min, Q1, mediana, Q3, max y media). Para la información de cultivo y la de producción disponemos de la sumatoria total por semanas.

La aplicación constará de 3 secciones. Por un lado, tendremos una sección de resumen en la que se mostrará información de carácter general. La información será la siguiente:

- Información gráfica del total de superficie cultivada por producto.
- Información en forma de tabla acerca del total de superficie cultivada por producto.
- Información gráfica del total de Kg recogidos por producto.
- Información en forma de tabla acerca del total de Kg recogidos por producto.
- Información gráfica del rendimiento de cada producto.
- Información en forma de tabla acerca del rendimiento de cada producto.
- Información gráfica del dinero promedio recaudado por producto.
- Información en forma de tabla acerca del importe recaudado por producto.
- Información gráfica de la rentabilidad de cada producto (€).
- Información en forma de tabla acerca de la rentabilidad de cada producto (€) .
- Información gráfica de la media de precio de cada producto.
- Información en forma de tabla acerca de la media de precio de cada producto.

Esta información se debe poder filtrar por años y disponer de la opción de exportar en cada caso la información completa a formato csv o txt. También tendrá un formulario para poder elegir un producto y poder filtrar por género, calidad, calibre o aspecto, de modo que se pueda explorar cada uno de los productos por separado.

La segunda sección contendrá información con forma de ranking donde podremos ver los géneros más cultivados, menos cultivados, más demandados, los menos demandados, con mejor rendimiento, con peor rendimiento, más caros, más baratos, más rentables, menos rentables, los que más han recaudado y los que menos. Esta información se podrá filtrar por años y estará disponible en la página de inicio.

La tercera sección mostrará información evolutiva. El usuario podrá elegir qué tipo de información evolutiva desea visualizar de entre las opciones disponibles:

- Cultivo.
- Producción.
- Mercado.
- Climatológica.

También podrá elegir un período concreto. Si la información que desea visualizar es relativa a productos, podrá elegir el producto (tomate, pepino, pimiento. . .) y también tendrá la opción de filtrar por distintos atributos, siendo estos el género, el calibre, el aspecto o la calidad. Si la información que desea visualizar es relativa a variables climatológicas, podrá elegir la variable que desea visualizar (temperatura, humedad, etc.). En caso de ver información de mercado o climatológica, en la gráfica podrá habilitar/deshabilitar cualquier valor de la distribución, pudiendo visualizar la evolución de los máximos, Q3, mediana, media, Q1 y mínimos.

Aparte de la información gráfica, podrá ver una tabla resumen de los valores representados. Esta tabla resumen será diferente dependiendo del tipo de dato que esté visualizando. Si los datos son de cultivo o de producción, la tabla reflejará el total cultivado o recolectado en cada etapa. Por el contrario, si los datos son de mercado o climatológicos, dispondremos de su distribución. Al igual que en la otra sección, podrá exportar el conjunto de datos con mayor nivel de detalle a formato .csv o .txt. Tendrá la opción de limpiar los filtros y dejar de nuevo los campos a su valor por defecto.

4.2. Análisis

En esta sección extraeremos los distintos elementos de la especificación.

4.2.1. Objetivos

Identificamos los siguientes objetivos:

OBJ-01	Consulta de información ranking
Versión	1.00
Autor	Razvan Ioan Stoia
Descripción	El sistema permitirá consultar información de ranking de productos
Estabilidad	Alta
Importancia	Alta
Comentarios	

OBJ-02	Consulta de información resumen
Versión	1.00
Autor	Razvan Ioan Stoia
Descripción	El sistema permitirá consultar información resumen de cultivo, producción y mercado
Estabilidad	Alta
Importancia	Alta
Comentarios	Esta información se expondrá tanto de forma gráfica como en forma de tabla.

OBJ-03	Consulta de información evolutiva
Versión	1.00
Autor	Razvan Ioan Stoia
Descripción	El sistema permitirá consultar información evolutiva de cultivo, producción, mercado y climatológica
Estabilidad	Alta
Importancia	Alta
Comentarios	Esta información se expondrá tanto de forma gráfica como en forma de tabla

4.2.2. Actores

La aplicación tendrá un solo usuario con un solo rol, ya que contiene datos generales del poniente almeriense. Si finalmente la aplicación toma como información de entrada datos correspondientes a alguna organización o entidad del sector agroalimentario, se podría adaptar el cuadro de mandos para que mostrase distinta información en función del usuario que accediese. Actualmente el único actor y rol que hay es :

ACT-01	Usuario
Rol	Rol - 01: Usuario
Descripción	Este actor representa a cualquier usuario que acceda a la aplicación
Comentarios	
Rol - 01	Usuario
Versión	1.00
Actor	ACT - 01: Usuario
Descripción	Usa la aplicación para la toma de decisiones en el sector agroalimentario
Requisitos asociados	
Descripción	El sistema tiene un único actor para el que se definen todos los requisitos funcionales
Comentarios	

4.2.3. Requisitos funcionales

Los requisitos funcionales detallados los podemos consultar en los anexos. A continuación vamos a listarlos para cada objetivo:

Objetivo	Requisitos funcionales
Obj-01 Consulta de información de ranking	RF-18 Acceder a página ranking RF-19 Filtrar por año RF-20 Mostrar ranking cultivo RF-22 Mostrar ranking producción RF-24 Mostrar ranking rendimiento RF-26 Mostrar ranking ganancias RF-28 Mostrar ranking precio RF-30 Mostrar ranking rentabilidad

Objetivo	Requisitos funcionales
Obj-02 Consulta de información resumen	RF-01 Acceder a página resumen RF-02 Seleccionar producto RF-03 Seleccionar filtro RF-04 Marcar período temporal RF-05 Mostrar resumen cultivo RF-32 Mostrar resumen producción RF-33 Mostrar resumen rendimiento RF-34 Mostrar resumen ganancias RF-35 Mostrar resumen precio RF-36 Mostrar resumen rentabilidad RF-37 Exportar información RF-38 Limpiar campos

Objetivo	Requisitos funcionales
Obj-03 Consulta de información evolutiva	RF-08 Acceder a página progreso RF-09 Seleccionar tipo de información RF-10 Seleccionar variable climática RF-11 Seleccionar producto RF-12 Seleccionar tipo de filtrado RF-39 Seleccionar filtro RF-13 Marcar período temporal RF-14 Mostrar información RF-15 Habilitar o deshabilitar opciones RF-16 Exportar tabla RF-17 Limpiar campos

4.2.4. Requisitos no funcionales

Aparte de las funcionalidades del sistema, se han definido una serie de requisitos no funcionales, los cuales vamos a listar a continuación:

RNF-01	Disponibilidad
Objetivos asociados	OBJ-01: Consulta de información de ranking. OBJ-02: Consulta de información resumen. OBJ-03: Consulta de información evolutiva.
Requisitos asociados	Ninguno
Descripción	El sistema estará disponible a través de la web las 24 horas.
Comentarios	Ninguno

4.3. Validación con prototipos

Los prototipos iniciales derivados de los requisitos se exponen en los anexos. Hay que tener en cuenta que el sistema está en constante evolución y que los siguientes prototipos se construyeron directamente en Shiny. Por tanto puede haber diferencias significativas entre estos prototipos y

RNF-02	Usabilidad
Objetivos asociados	OBJ-01: Consulta de información de ranking. OBJ-02: Consulta de información resumen. OBJ-03: Consulta de información evolutiva.
Requisitos asociados	Ninguno
Descripción	El sistema tendrá un tiempo corto de aprendizaje.
Comentarios	Ninguno
RNF-03	Eficiencia
Objetivos asociados	OBJ-01: Consulta de información de ranking. OBJ-02: Consulta de información resumen. OBJ-03: Consulta de información evolutiva.
Requisitos asociados	Ninguno
Descripción	El sistema será fluido y sin tiempos de carga excesivos.
Comentarios	Ninguno

el sistema actual.

4.4. Implementación

Para el desarrollo de la aplicación con Shiny, hemos utilizado 4 archivos principalmente. Estos 4 archivos son:

- *config.R*: Contiene todos los parámetros de configuración, tales como el número de decimales, la divisa, el separador decimal, etc. También contiene la carga de los datos dentro de una variable.
- *ui.R*: Contiene la interfaz de usuario. En realidad, no contiene todos los elementos como tal, ya que para la mayoría de ellos únicamente reserva un espacio dentro de la interfaz, y este espacio es ocupado posteriormente por un elemento visual renderizado por el servidor. Esto se ha implementado así ya que muchos elementos visuales requieren de ciertas entradas y no pueden ser mostrados inicialmente.
- *server.R*: Contiene toda la lógica de la aplicación. Dentro del propio archivo se ha separado la lógica de negocio, de la lógica de la aplicación. Esto se ha hecho mediante funciones. Dentro de este archivo se renderiza gran parte de la interfaz, tanto *inputs* como *outputs*.

- *functions.R*: Contiene las funciones comunes a todo el sistema. Funciones para construir gráficas, construir tablas, o utilidades como por ejemplo funciones para obtener valores de un mapa a partir de una clave. Estas funciones facilitan la reutilización de código.

4.4.1. Estructura de la aplicación

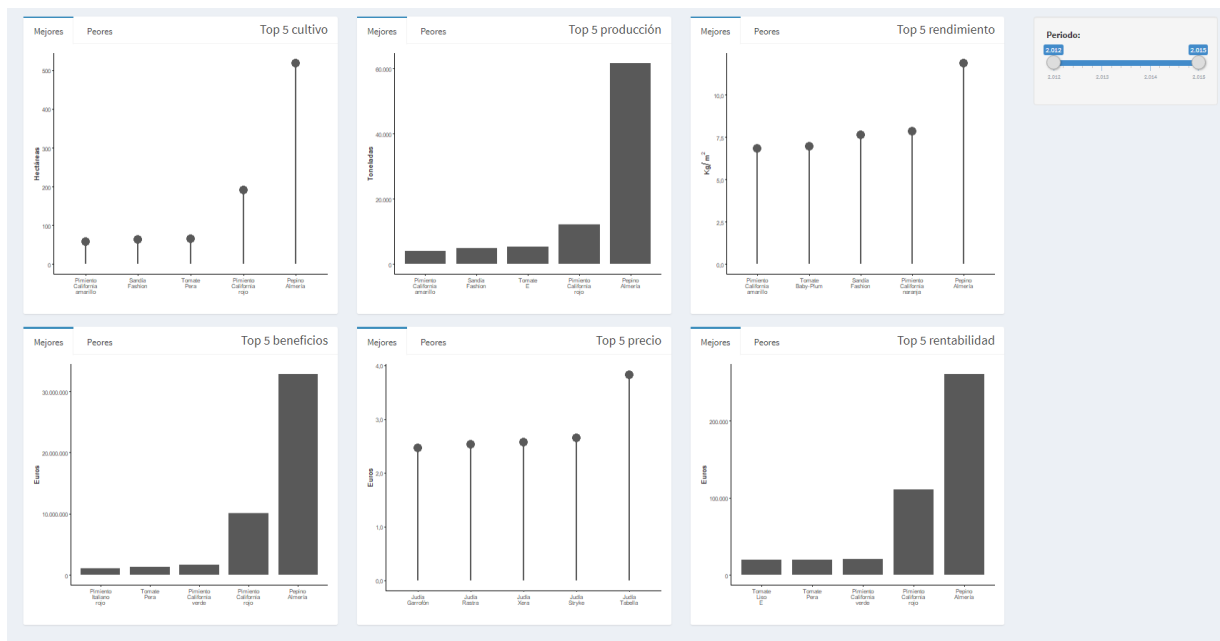
El archivo *ui.R* tiene tres funciones principales. Una función que carga la cabecera, *dashboardHeader()*, y que contiene elementos como el título de la aplicación o el logo. Otra función, *dashboardSidebar()* que carga el menú lateral a través del cual se accede a las distintas secciones de la aplicación. Y por último, la función que se encarga de cargar el contenido, *dashboardBody()*. Esta última función asocia cada elemento del menú con su contenido a través de un *id*, que se corresponde con el parámetro *tabName*.

```
1 library(shiny)
2 library(shinydashboard)
3 library(tidyverse)
4 library(DT)
5
6 # Interfaz
7 dashboardPage(skin = "green",
8   dashboardHeader(
9     title = "Intelindalo"),
10  dashboardSidebar(
11    sidebarMenu(
12      id = "items",
13      menuItem("Ranking", tabName = "ranking", icon = icon("sort-amount-desc")),
14      menuItem("Resumen", tabName = "summary", icon = icon("th")),
15      menuItem("Progreso", tabName = "evolution", icon = icon("line-chart"))
16    )
17  ),
18  dashboardBody(...))
```

Los datos que recibe la aplicación son los mismos que exportamos al final de todo el proceso de tratamiento de datos. Estos se importan en el archivo *config.R*.

4.4.2. Sección Ranking

La sección de ranking contiene varias cajas de contenido, una por cada indicador que se quiere representar, y cada caja tiene dos pestañas, de modo que se puede ver un ranking de los mejores productos o de los peores. Los indicadores que se quieren mostrar son: cultivo, producción, rendimiento, beneficios, precio y rentabilidad. La definición de cada uno de ellos viene detallada en los requisitos. Esta sección tiene un panel lateral en el lado derecho que contiene un slider para poder filtrar por años. El panel principal contiene las cajas con el contenido.



Fuente: autoría propia
 Figura 4.1: Sección ranking de la aplicación.

Lo primero que hicimos fue crear una función que filtrase el *dataframe* y seleccionase, de las 82 variables, solamente aquellas que se iban a utilizar en esta sección. Pero además de eso, también filtramos por el periodo temporal marcado por el usuario. Es por ello que esta función es reactiva, ya que se recalcula cada vez que cambia la entrada.

```

1 ranking <- reactive({
2   req(input$rankPeriod)
3   period_f <- input$rankPeriod[1]
4   period_l <- input$rankPeriod[2]
5   agrodata %>% select(producto_f, anyo, superficie_cultivada, total_kg_semana
6   , mean_cortel_semana) %>%
7   filter(between(anyo, period_f, period_l)) %>%
8   mutate_if(is.numeric, round, roundDecimals)
9 })

```

Esta función alimenta a todas las demás funciones que se encargan de hacer los cálculos específicos de cada indicador. Por ejemplo, la función para calcular el rendimiento promedio es la siguiente:

```

1 yieldRanking <- function(){
2   ranking() %>% select(!mean_cortel_semana) %>%
3   filter(!is.na(superficie_cultivada)|!is.na(total_kg_semana)) %>%
4   group_by(producto_f) %>%

```

4. SOLUCIÓN.

```
5     summarise(recogido = sum(total_kg_semana, na.rm = TRUE), plantado = sum(
6     superficie_cultivada, na.rm = TRUE)) %>%
7     filter((recogido!=0) & (plantado!=0)) %>%
8     mutate(rendimiento = recogido/plantado)
9 }
```

En este punto, ya tendríamos hecho el cálculo para cada indicador. Ahora faltaría extraer los 5 mejores y los 5 peores para representarlos gráficamente. Para ello, primero extraemos la información y después utilizamos una función común definida en el archivo *functions.R* para esbozar la gráfica.

```
1 upYieldRanking <- function() {
2   yieldRanking() %>% top_n(5, rendimiento)
3 }
4
5 downYieldRanking <- function() {
6   yieldRanking() %>% top_n(-5, rendimiento)
7 }
8
9 output$plotYieldRankingUp <- renderPlot({
10  customLollipopChart(upYieldRanking(), "producto_f", "rendimiento", "",
11  expression(Kg/m{2}))
12 })
13 output$plotYieldRankingDown <- renderPlot({
14  customLollipopChart(downYieldRanking(), "producto_f", "rendimiento", "",
15  expression(Kg/m{2}), TRUE)
16 })
```

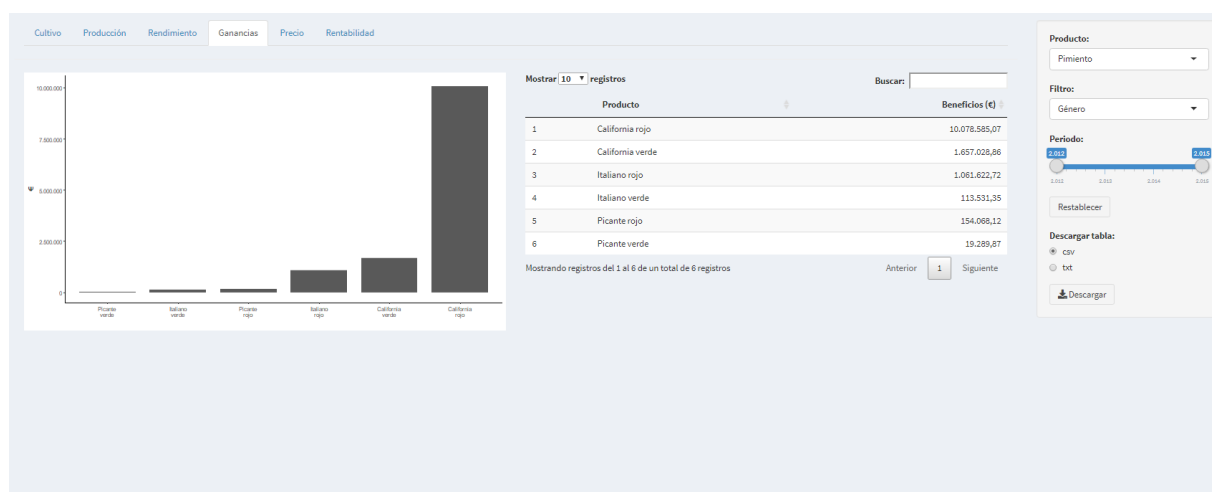
La función *customLollipopChart()* construye un gráfico de líneas con los parámetros que recibe y haciendo uso del paquete *ggplot2*. Faltaría solamente mostrar esta gráfica en la interfaz de usuario.

```
1 tabBox(title = "Top 5 rendimiento", width = 4,
2   tabPanel("Mejores",
3     plotOutput("plotYieldRankingUp")
4   ),
5   tabPanel("Peores",
6     plotOutput("plotYieldRankingDown")
7   )
8 )
```

4.4.3. Sección Resumen

La sección de resumen, al igual que la de ranking, está organizada en dos paneles. Un panel principal con el contenido y un panel secundario con las entradas de datos. El panel principal

contiene 6 pestañas, una para cada indicador del que se va a mostrar un resumen. Dentro de cada pestaña nos encontramos con una gráfica y una tabla. El panel secundario tiene un *select* para elegir un producto, cuyo valor por defecto es “Todos”. Si el usuario elige un producto, podrá seleccionar un filtro a través de otro *select*. Los posibles filtros que puede aplicar son: “género”, “calidad”, “calibre” o “aspecto”, aunque podrá elegir unas u otras en función de si el producto contiene información relacionada con esos atributos o no. También, al igual que en la sección anterior, dispone de un *slider* para filtrar por años, un botón para restablecer todos los campos, y la opción de descargar la tabla que se encuentra activa en ese momento en pantalla.



Fuente: autoría propia

Figura 4.2: Sección resumen de la aplicación.

Los paneles y las pestañas se han definido en el archivo *ui.R*, en el que también se ha reservado espacio para las gráficas, tablas y entradas de datos. Estas últimas se renderizan en *server.R* por ser dinámicas. La filosofía que hemos seguido es la misma que en la anterior sección, con la diferencia de que las funciones parciales que creamos para filtrar los datos progresivamente se utilizan tanto para actualizar las salidas de datos, como para extraer información que nos sirve para renderizar o actualizar las entradas. Lo primero que hacemos es identificar la pestaña en la que se encuentra el usuario, y en función de eso, partimos seleccionando unos datos u otros.

```

1 sumItemsValues <- reactive({
2   actualData <- switch (input$sumTabs,
3     area = sumAreaValues(),
4     kg = sumKgValues(),
5     yield = sumYieldValues(),
6     profits = sumProfitsValues(),
7     price = sumPriceValues(),
8     profitability = sumProfitabilityValues()
9   )
10 })

```

4. SOLUCIÓN.

Cada una de esas funciones, selecciona las variables que se necesitan para el cálculo del indicador en cuestión y filtra los valores ausentes. Por ejemplo, para las ganancias, hacemos lo siguiente:

```
1 sumProfitsValues <- function(){
2   agrodata %>% select(fecha_semana, anyo, producto, genero, aspecto, calidad,
3     calibre, total_kg_semana, mean_corte1_semana) %>%
4   filter(!is.na(total_kg_semana) & !is.na(mean_corte1_semana))
5 }
```

Con esta información, ya podemos pasar a la siguiente función, que filtra los registros que no tienen un producto o ningún filtro. Esta función será la que proporcionará los productos para el *select*.

```
1 sumProductValues <- reactive({
2   sumItemsValues() %>% filter(!is.na(producto)&(!is.na(genero)|!is.na(aspecto)|
3     !is.na(calidad)|!is.na(calibre)))
4 })
```

Y con esto, renderizamos el *select* de productos.

```
1 output$sumProductSelector <- renderUI({
2   products <- sumProductValues() %>%
3     pivot_longer(
4       cols = genero:calibre,
5       names_to = "filtro",
6       values_to = "rank") %>%
7     group_by(producto, filtro) %>%
8     summarise(count = n_distinct(rank)) %>%
9     filter(count > 1) %>%
10    pull(producto)
11   choices = c("Todos")
12   choices <- append(choices, products)
13   selectInput("sumProduct", "Producto:", choices)
14 })
```

Si el usuario selecciona un producto, renderizamos el siguiente *select*, que contiene los filtros disponibles para ese producto. La siguiente función filtra por el producto seleccionado, y será la que utilizemos para extraer los filtros.

```
1 sumFilterValues <- reactive({
2   req(input$sumProduct != "Todos")
3   sumProductValues() %>% filter(producto == input$sumProduct)
4 })
```

Ahora, igual que en el caso anterior, renderizamos el *select* de filtros. Por último, para renderizar el *slider* del periodo y establecer su valor máximo y mínimo, llamaríamos a la siguiente función:

```

1 sumPeriodValues <- reactive({
2   req(input$sumProduct)
3   if(input$sumProduct == "Todos"){
4     sumProductValues() %>% mutate(filtro = producto)
5   }else{
6     req(input$sumFilter)
7     if(input$sumFilter != ""){
8       filter_val <- dictGetKey(filters , input$sumFilter)
9       sumFilterValues() %>% filter(!is.na(.data[[filter_val]])) %>%
10        mutate(filtro = .data[[filter_val]])
11     }
12   }
13 })

```

Es ligeramente distinta ya que el *slider* está siempre presente, la diferencia es que si el usuario ha seleccionado un producto y un filtro, los datos están filtrados y por tanto el rango de valores podría ser distinto, de ahí el condicional que hemos insertado. Con esto renderizamos el *slider*.

```

1 output$sumPeriodSlider <- renderUI({
2   period_f <- min(sumPeriodValues()$anyo)
3   period_l <- max(sumPeriodValues()$anyo)
4   sliderInput("sumPeriod", "Periodo:", min = period_f, max = period_l, value =
5     c(period_f, period_l), step = 1, sep = ".")

```

Y finalmente, aplicamos el filtro temporal.

```

1 sumFinalValues <- reactive({
2   sumPeriodValues() %>% filter(between(anyo, input$sumPeriod[1], input$
3     sumPeriod[2]))

```

Ahora, solo faltaría tomar de nuevo la pestaña activa de la interfaz para hacer los cálculos necesarios, con los datos ya preparados y filtrados.

```

1 sumValues <- reactive({
2   actualData <- switch (input$sumTabs,
3     area = areaSum(),
4     kg = kgSum(),
5     yield = yieldSum(),
6     profits = profitsSum(),

```

4. SOLUCIÓN.

```
7           price = priceSum(),
8           profitability = profitabilitySum()
9       )
10  })
```

Y, siguiendo el ejemplo expuesto con anterioridad, para el cálculo de las ganancias haríamos lo siguiente.

```
1 profitsSum <- function() {
2   sumFinalValues() %>% mutate(beneficios_semana = total_kg_semana * mean_
3     corte1_semana) %>%
4     group_by(filtro) %>%
5     summarise(beneficios = sum(beneficios_semana)) %>%
6     mutate_if(is.numeric, round, roundDecimals)
7 }
```

Y renderizamos gráfica y tabla.

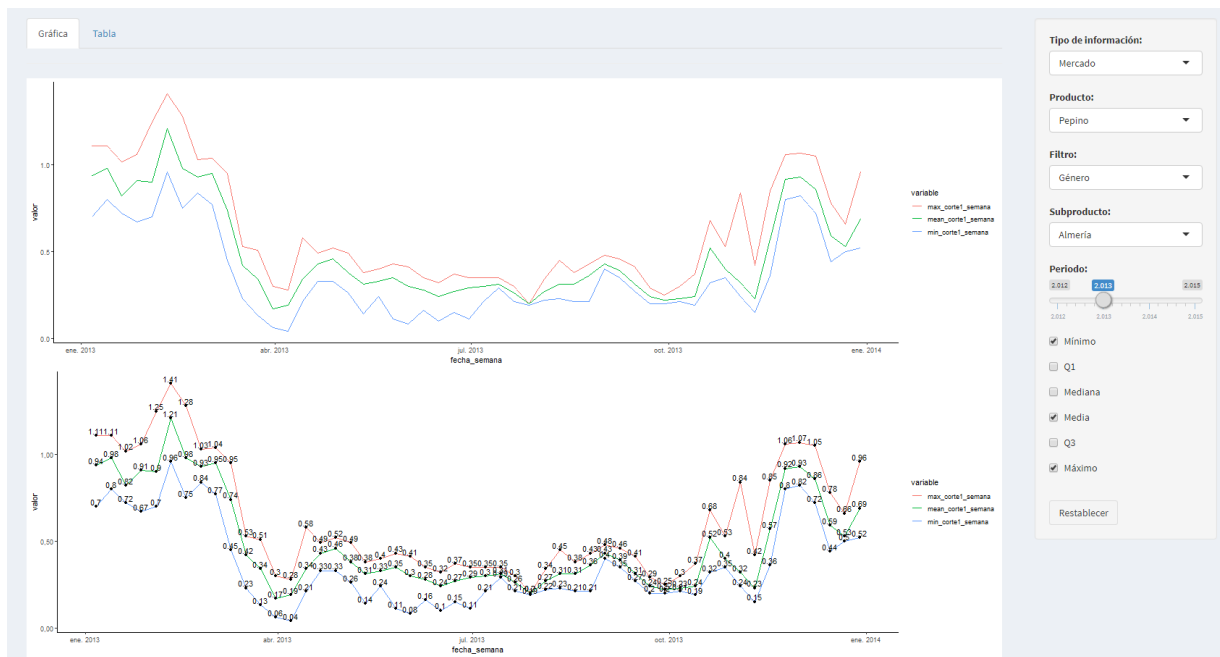
```
1 output$plotProfitsSum <- renderPlot({
2   customBarChart(sumValues(), "filtro", "beneficios", "", "€")
3 })
4
5 output$tableProfitsSum <- DT::renderDataTable({
6   customTable(sumValues(), c('Producto', 'Beneficios (€)'), 10, 2, 2)
7 })
```

Las funciones para renderizar la gráfica y la tabla están definidas en el archivo *functions.R*

4.4.4. Sección Progreso

La sección progreso, siguiendo la misma línea de diseño, también está organizada en dos paneles. El panel principal tiene 2 pestañas, para cambiar entre la vista de gráfica y la de tabla. El panel secundario contiene todos los *inputs*. El primer *input*, que se corresponde con el tipo de información, será el que determine la sucesiva información que se le solicitará al usuario. Si el tipo de información es climática, tendrá que indicar la variable climática a visualizar. En caso contrario, tendrá que elegir un producto, un tipo de filtro y un subproducto para ese filtro. En cualquier caso podrá marcar un periodo temporal, y si el tipo de información es climática o de mercado, podrá elegir que valores de la distribución de los datos quiere visualizar. También dispondrá de los botones de “Restablece” (devolver los campos a su valor por defecto) y “Descargar” (descargar la tabla). Este último solamente estará disponible cuando se visualice la tabla en pantalla.

Para implementar esta sección, hemos seguido la misma metodología que en el apartado de resumen. En función del tipo de información marcado por el usuario, aplicamos una función que



Fuente: autoría propia

Figura 4.3: Sección progreso de la aplicación.

selecciona las variables de partida y filtramos los valores ausentes. Sucesivas funciones nos sirven para renderizar las entradas y a la vez filtrar progresivamente los datos. Al final de todo el filtrado, todas las funciones terminan en dos posibles puntos. Si la información es climática:

```

1 evFinalWeatherValues <- reactive({
2   req(input$evPeriod)
3   evWeatherPeriodValues() %>% filter(between(anyo, input$evPeriod[1], input$
4     evPeriod[2])) %>%
5     mutate_if(is.numeric, round, roundDecimals)
6 })

```

Si es de otro tipo:

```

1 evFinalRestValues <- reactive({
2   req(input$evPeriod)
3   evRestPeriodValues() %>% filter(between(anyo, input$evPeriod[1], input$
4     evPeriod[2])) %>%
5     mutate_if(is.numeric, round, roundDecimals)
6 })

```

Para renderizar los *checkboxes* de la distribución de datos y el botón para descargar la tabla, hemos creado dos funciones simples para comprobar si se cumplen las condiciones necesarias:

4. SOLUCIÓN.

```
1 checkIfRenderCheckboxes <- reactive({
2   req(input$evInformation == "clima" && input$evWeather != "" ||
3     input$evInformation == "corte1" && input$evFilterChoice != "")
4 })
5
6 checkIfRenderDownload <- reactive({
7   req(input$evWeather != "" || input$evFilterChoice != "")
8   req(input$evTab == "table")
9 })
```

De modo que, antes de renderizar esos items, hacemos una llamada a estas funciones, y solamente si se cumplen las condiciones de los *req()* se mostrarán.

De nuevo, mediante una función reactiva, controlamos la selección y cálculo de los datos que representaremos gráficamente y en forma de tabla.

```
1 dataEv <- reactive({
2   actualDataEv <- switch (input$evInformation ,
3     clima = weatherEv() ,
4     superficie_cultivada = areaEv() ,
5     total_kg_semana = kgEv() ,
6     corte1 = priceEv()
7   )
8 })
```

Por ejemplo, la función final para seleccionar los datos climáticos sería:

```
1 weatherEv <- reactive({
2   req(!is.null(input$evMin))
3   req(!is.null(input$evQOne))
4   req(!is.null(input$evMed))
5   req(!is.null(input$evMean))
6   req(!is.null(input$evQThree))
7   req(!is.null(input$evMax))
8
9   subdata <- evFinalWeatherValues() %>% select(!anyo)
10
11   if(!input$evMin){
12     subdata <- subdata %>% select(!starts_with("min"))
13   }
14   if(!input$evQOne){
15     subdata <- subdata %>% select(!starts_with("q1"))
16   }
17   if(!input$evMed){
18     subdata <- subdata %>% select(!starts_with("med"))
19   }
20   if(!input$evMean){
```

```

21   subdata <- subdata %>% select(!starts_with("mean"))
22   }
23   if(!input$evQThree){
24     subdata <- subdata %>% select(!starts_with("q3"))
25   }
26   if(!input$evMax){
27     subdata <- subdata %>% select(!starts_with("max"))
28   }
29   return(subdata)
30 }
31 })

```

Donde ya seleccionamos los valores de la distribución marcados por el usuario.

Por último, para facilitar la representación de los datos, juntamos todos los valores en dos variables, a modo de clave y valor.

```

1 subdata <- dataEv()
2 subdata <- subdata %>% pivot_longer(c(-fecha_semana), names_to = "variable",
  values_to = "valor")

```

Algo parecido a lo que podemos ver en la siguiente figura.

country	year	cases
Afghanistan	1999	745
Afghanistan	2000	2666
Brazil	1999	37737
Brazil	2000	80488
China	1999	212258
China	2000	213766

country	1999	2000
Afghanistan	745	2666
Brazil	37737	80488
China	212258	213766

table4

Fuente: <https://r4ds.had.co.nz/>

Figura 4.4: Agrupación de variables con la técnica *pivot longer*.

Para construir la gráfica inferior, que se utiliza para ampliar la gráfica superior, hemos utilizado un *observer* que comprueba la selección en la gráfica superior y establece unos rangos.

```

1 observe({
2   brush <- input$ev_plot_brush
3   if (!is.null(brush)) {

```

4. SOLUCIÓN.

```
4     evRanges$x <- c(brush$xmin, brush$xmax)
5     evRanges$y <- c(brush$ymin, brush$ymax)
6
7     } else {
8       evRanges$x <- NULL
9       evRanges$y <- NULL
10    }
11  })
```

Y añadimos las siguientes líneas a la función que construye la gráfica con *ggplot2*.

```
1 {if(!is.null(x_lim)) coord_cartesian(xlim = x_lim)}+
2 {if(!is.null(y_lim)) coord_cartesian(ylim = y_lim)}
```

4.5. Despliegue

La aplicación se ha desplegado en *Shinyapps* y se encuentra accesible en el siguiente enlace <https://indalo.shinyapps.io/Intelindalo/>. Los pasos para desplegar la aplicación se pueden revisar en los apéndices.

El código de la aplicación está subido a github, al siguiente repositorio <https://github.com/razvans96/tfg.git>

5

Conclusión y líneas de trabajo futuras

En el presente trabajo se han abordado problemas de distinto ámbito. En primer lugar hemos localizado las distintas fuentes de datos, y una vez localizadas, hemos procedido a la extracción de esos datos desde una base de datos de productos, desde una web de precios y desde una base de datos climática. Esto nos ha llevado a afrontar otro problema, que es la heterogeneidad de estos datos a la hora de unificarlos. Para poder integrarlos, primero hemos pasado por un proceso de carga, preprocesamiento y transformación. Una vez integrados, hemos desarrollado un cuadro de mandos sobre la plataforma R. Desde este cuadro de mandos, se pueden hacer consultas de forma rápida y fácil, con el objetivo de ofrecer una ayuda visual para la consulta de información. Para hacer el cálculo de las consultas, hemos empleado operaciones de agregación y de transformación sobre el *dataset* integrado.

Llegados a este punto, nos podemos preguntar ¿qué nos falta? Bien, pues las opciones son múltiples, ya que tenemos algunos aspectos en los que podemos mejorar y ampliar la aplicación. Podríamos incorporar nuevos datos para ampliar el abanico de información que se ofrece actualmente. La incorporación de nuevos datos debería pasar por todo el flujo ya definido para los datos actuales. También podríamos ofrecer una ayuda visual para la búsqueda de relación entre variables, con el objetivo de facilitar la toma de decisiones en el sector. Para facilitar esa toma de decisiones también se podría emplear algún algoritmo de clasificación y predicción de datos. Por último, también podríamos almacenar los datos en una base de datos, en lugar de utilizar archivos. Para ello deberíamos migrar los datos que ya tenemos a la nueva base de datos y conectarla con la aplicación.

Apéndice A

Descarga de los datos de mercado

A.1. Página de Fhalmeria

Nuestra página objetivo contiene la estructura que podemos observar en las siguientes figuras. Tiene tres entradas de datos, un botón para lanzar la búsqueda y varios botones para desplegar la pizarra correspondiente a cada día. Nuestro objetivo es en cada caso extraer el encabezado con el texto descriptivo, en el cual aparece el nombre de la cooperativa además de la fecha, y la tabla con los datos.

Histórico de pizarras

Selección de la entidad agrícola: Agroejido

Selección del rango de fechas: 01/05/2014 - 01/06/2014

Lanzar búsqueda: Buscar

Pizarras disponibles:

Pizarra del 02/05/2014	Pizarra del 03/05/2014	Pizarra del 05/05/2014	Pizarra del 06/05/2014
Mostrar pizarra	Mostrar pizarra	Mostrar pizarra	Mostrar pizarra
Pizarra del 07/05/2014	Pizarra del 08/05/2014	Pizarra del 09/05/2014	Pizarra del 10/05/2014
Mostrar pizarra	Mostrar pizarra	Mostrar pizarra	Mostrar pizarra
Pizarra del 12/05/2014	Pizarra del 13/05/2014	Pizarra del 14/05/2014	Pizarra del 15/05/2014
Mostrar pizarra	Mostrar pizarra	Mostrar pizarra	Mostrar pizarra
Pizarra del 16/05/2014	Pizarra del 17/05/2014	Pizarra del 19/05/2014	Pizarra del 20/05/2014
Mostrar pizarra	Mostrar pizarra	Mostrar pizarra	Mostrar pizarra
Pizarra del 21/05/2014	Pizarra del 22/05/2014	Pizarra del 23/05/2014	Pizarra del 24/05/2014
Mostrar pizarra	Mostrar pizarra	Mostrar pizarra	Mostrar pizarra
Pizarra del 26/05/2014	Pizarra del 27/05/2014	Pizarra del 28/05/2014	Pizarra del 29/05/2014
Mostrar pizarra	Mostrar pizarra	Mostrar pizarra	Mostrar pizarra

Despliega la tabla en la zona inferior

Fuente: autoría propia

Figura A.1: Fhalmeria. Pantalla 1 de la sección del histórico de mercados, con los elementos que intervienen en nuestro script marcados en rojo.

PIZARRA DE Agroejido DEL 02/05/2014 Encabezado con el nombre y la fecha Tabla objetivo

Producto	Media 3 Primeros																	
Judía Xera	2,12 ▼	2,55	1,70															
Judía Perona larga	1,43 ▲	1,51	1,42	1,37	1,33	1,30	1,26	1,22										
Judía Perona roja	0,96 ▼	1,39	0,77	0,72														
Pepino Francés	0,16 ▼	0,20	0,17	0,13														
Pepino Español	0,17 ▼	0,20	0,17	0,14	0,12	0,08												
Calabacín Fino	0,20 –	0,24	0,20	0,17	0,13	0,10												
Calabacín Gordo	0,21 ▲	0,24	0,22	0,19	0,17	0,16	0,15	0,13										
Berenjena Rayada	0,41 ▼	0,50	0,32															
Berenjena Larga	0,50 ▲	0,55	0,52	0,44	0,37	0,33	0,29	0,26										
Tomate Liso Maduro	0,29 ▼	0,29																
Pimiento Italiano rojo	0,56 ▲	0,60	0,56	0,52	0,48	0,44												
Pimiento Lamuyo verde	0,42 ▼	0,45	0,40															
Pimiento Lamuyo rojo	0,33 ▼	0,40	0,33	0,26	0,22	0,19	0,14	0,13										
Pimiento California verde	0,05 ▼	0,07	0,04															
Pimiento California rojo	0,22 ▲	0,27	0,24	0,16	0,14	0,07												
Pimiento California amarillo	0,04 –	0,04																

Fuente: autoría propia

Figura A.2: Fhalmeria. Pantalla 2 de la sección del histórico de mercados, con los elementos que intervienen en nuestro script marcados en rojo.

A.2. Script en VBScript

El siguiente script en Visual Basic contiene toda la lógica necesaria para automatizar la tarea de obtener y convertir los datos de mercado de un formato de tabla HTML a formato CSV. Para poder declarar los objetos relativos a las macros (las cuales analizaremos en el siguiente punto) previamente hemos instalado unas librerías de la página oficial de la herramienta iMacros [9]. Para procesar y exportar el texto a formato CSV hemos utilizado un componente ActiveX de la web de Chilkat Software [16].

En el script hacemos principalmente lo siguiente:

1. Declaramos las variables e inicializamos los objetos con los que vamos a trabajar. Objetos de entrada y salida de datos, gestión de macros, variables de control, etc.
2. Seleccionamos consecutivamente las entidades agrícolas y para cada una de ellas mediante bucles nos desplazamos por los distintos años y meses. A través de las macros establecemos los valores del rango de fechas y lanzamos las sucesivas búsquedas.
3. Clicamos en los botones correspondientes a cada día y copiamos el encabezado y la tabla HTML a unas variables.
4. Exportamos la tabla a un archivo CSV, con el nombre almacenado previamente.

Para más detalles, en el código se pueden encontrar algunos comentarios.

```

1 ' Script para extraer datos históricos de mercado de la web de Fhalmeria https://
  www.fhalmeria.com/es/historico-mercados.aspx
2 Option Explicit
3 Dim message
4 message = "Este script se encarga de hacer Web Scrapping sobre la pagina de
  fhalmeria para "
5 message = message + "descargar los datos historicos de venta de frutas y
  hortalizas en la provincia de Almeria."
6 MsgBox (message)
7
8 Dim myname, mypath, objFileSystem, objOutputFile
9 ' localizamos la carpeta actual
10 myname = WScript.ScriptFullName
11 mypath = Left(myname, InstrRev(myname, "\"))
12 ' Inicializamos los objetos del sistema
13 Set objFileSystem = CreateObject("Scripting.FileSystemObject")
14 Set objOutputFile = objFileSystem.CreateTextFile("output.txt", True)
15 ' Inicializamos las variables y el objeto de la librería de iMacros para hacer la
  conexión
16 Dim iim1, iret, iplay, control
17 set iim1= CreateObject ("imacros")
18 iret = iim1.iimOpen("")
19 ' Todas las variables que intervienen en el algoritmo
20 Dim id, iterador, anyo, mes, inicio, dfm, count, finicio, ffin,
21 almacen, nombre, fecha, tabla, linea, lineas, column, columns,
22 colcontenido, csv, csvDoc, success
23 ' Las entidades que vamos a recorrer EDIT:de las últimas 2 no hay datos
24 id = Array(1, 3, 4, 6, 7, 9, 10, 11, 16, 43, 52)
25 ' Nos dirigimos a la web de destino
26 iplay = iim1.iimPlay(mypath & "Macros\wsh-destino.iim")
27 For iterador = 0 To UBound(id)-1
28 ' Pasamos parámetros a las macros y lanzamos la búsqueda
29 iret = iim1.iimSet("idalmacen", id(iterador))
30 iplay = iim1.iimPlay(mypath & "Macros\wsh-elegir-almacen.iim")
31 almacen = iim1.iimGetLastExtract(1)
32 If iplay > 0 Then
33   For anyo = 2013 To 2015
34     inicio = 1
35     dfm = 1
36     For mes=1 To 12
37       finicio = inicio & "/" & cstr(mes) & "/" & cstr(anyo)
38       If mes=12 Then
39         ffin = "31/" & cstr(mes) & "/" & cstr(anyo)
40       Else
41         ffin = dfm & "/" & cstr(mes+1) & "/" & cstr(anyo)
42       End If
43       iret = iim1.iimSet("finicio", finicio)
44       iret = iim1.iimSet("ffin", ffin)
45       iplay = iim1.iimPlay(mypath & "Macros\wsh-elegir-fechas.iim")
46       iplay = iim1.iimPlay(mypath & "Macros\wsh-busqueda.iim")
47       If iplay = 1 Then
48         control = 1
49         count = -1
50       ' Clicamos sucesivamente en los botones correspondientes a cada día

```

```

51     While control > 0
52         count = count + 1
53         iret = iim1.iimSet("dia", count)
54         control = iim1.iimPlay(mypath & "Macros\wsh-mostrar-pizarra.iim")
55     If control > 0 Then
56         ' Copiamos el encabezado con el nombre de la entidad y la fecha, que
        será el nombre que daremos al archivo
57         nombre = iim1.iimGetLastExtract(1)
58         ' Separamos la fecha en una variable independiente
59         fecha = Mid(nombre, Len(nombre)-9, 10)
60         ' Transformamos el nombre en un formato de nombre de archivo
correcto, sin / y sin espacios
61         nombre = Replace(nombre, " ", "_")
62         nombre = Replace(nombre, "/", "_")
63         ' Almacenamos la tabla en una variable para procesarla
64         isplay = iim1.iimPlay(mypath & "Macros\wsh-descargar-tabla.iim")

65         tabla = iim1.iimGetLastExtract(1)
66         ' Procesamos la tabla
67         set csv = CreateObject("Chilkat_9_5_0.Csv")
68         csv.HasColumnNames = 1
69         csv.Delimiter = ";"
70         lineas = Split(tabla, "#NEWLINE#")
71         For linea=0 To UBound(lineas)-1
72             columnas=Split(lineas(linea), "#NEXT#")
73             For columna=0 To UBound(columnas)-1
74                 If linea=0 Then
75                     success = csv.SetColumnName(columna, columnas(columna))
76                     If columna=UBound(columnas)-1 Then
77                         success = csv.SetColumnName(UBound(columnas), "Fecha")
78                     End If
79                 Else
80                     colcontenido = columnas(columna)
81                     If columna=2 Then
82                         colcontenido = Mid(colcontenido,1,4)
83                     End If
84                     success = csv.SetCell(linea-1,columna,colcontenido)
85                     If columna=UBound(columnas)-1 Then
86                         success = csv.SetCell(linea-1,UBound(columnas), fecha)
87                     End If
88                 End If
89             Next
90         Next
91         ' Exportamos al archivo en formato .csv
92         csvDoc = csv.SaveToString()
93         objOutputFile.WriteLine(csvDoc)
94         success = csv.SaveFile("Descargas/"& cstr(ano) & "/" & nombre & ".
csv")
95     End If
96     Wend
97     End If
98     inicio = 2
99     Next
100    Next

```

```

101 End If
102 Next
103 iret = iim1.iimClose
104 Set objFileSystem = Nothing
105 MsgBox "El script se ha detenido."
106 WScript.Quit(iret)

```

A.3. Macros

La primera macro la utilizamos para dirigir a la web de destino.

```

1 TAB T=1
2 TAB CLOSEALLOTHERS
3 URL GOTO=https://www.fhalmeria.com/es/historico-mercados.aspx

```

Una vez que estamos en la página de destino, seleccionamos el almacén o la entidad mediante la id que tiene asociada el item en el código fuente (nos llega desde el script) y además extraemos el valor del campo (nombre de la entidad) para devolverlo al script.

```

1 TAG POS=1 TYPE=SELECT FORM=ID:ct100 ATTR=ID:contenedor_informacion_DDLugares
  CONTENT={{idalmacen}}
2 TAG POS=1 TYPE=SELECT FORM=ID:ct100 ATTR=ID:contenedor_informacion_DDLugares
  EXTRACT=TEXT

```

Después completamos los campos con las fechas de inicio y fin, que llegan como parámetros desde el script.

```

1 TAG POS=1 TYPE=INPUT:TEXT FORM=ID:ct100 ATTR=ID:contenedor_informacion_
  TBFechaInicio CONTENT={{finicio}}<SP>
2 TAG POS=1 TYPE=INPUT:TEXT FORM=ID:ct100 ATTR=ID:contenedor_informacion_TBFechaFin
  CONTENT={{ffin}}<SP>

```

Lanzamos la búsqueda mediante la acción de pulsar el botón.

```

1 TAG POS=1 TYPE=INPUT:SUBMIT FORM=ID:ct100 ATTR=ID:contenedor_informacion_
  BBuscar

```

A través del script, con un bucle, hacemos sucesivas llamadas a la macro para pulsar los botones correspondientes a los días, que nuevamente obtenemos como parámetros. También extraemos el encabezado <h3>que contiene el nombre de la entidad y la fecha.

```

1 TAG POS=1 TYPE=INPUT:SUBMIT FORM=ID:ct100 ATTR=ID:contenedor_informacion_
  RBusqueda_BVerPizarra_{{dia}}
2 TAG POS=1 TYPE=H3 ATTR=TEXT:* EXTRACT=TEXT

```

Finalmente, extraemos la tabla.

```

1 TAG POS=1 TYPE=TABLE ATTR=TEXT:* EXTRACT=TEXT
2 'SAVEAS TYPE=EXTRACT FOLDER={{carpeta}} FILE={{archivo}}.csv

```


Apéndice B

Requisitos funcionales

RF-18	Acceder a página Ranking						
Versión	1.00						
Objetivos asociados	OBJ-01 : Consulta de información de ranking						
Descripción	El sistema permitirá al usuario acceder a la sección de Ranking desde el menú.						
Precondición	El usuario debe haber abierto la aplicación y debe haberse completado la carga inicial.						
Secuencia normal	<table border="1"><thead><tr><th>Paso</th><th>Acción</th></tr></thead><tbody><tr><td>P1</td><td>El usuario ejecuta la aplicación.</td></tr><tr><td>P2</td><td>El usuario clica en la sección Ranking.</td></tr></tbody></table>	Paso	Acción	P1	El usuario ejecuta la aplicación.	P2	El usuario clica en la sección Ranking.
Paso	Acción						
P1	El usuario ejecuta la aplicación.						
P2	El usuario clica en la sección Ranking.						
Postcondición	El usuario accederá a la sección de Ranking, desde la cual podrá ejecutar diversas acciones.						
Excepciones							
Comentarios	Ninguno						

RF-19	Filtrar por año								
Versión	1.00								
Objetivos asociados	OBJ-01 : Consulta de información de ranking								
Descripción	El sistema permitirá al usuario filtrar por años la información que visualiza dentro de la sección de Ranking. Por defecto, estaría seleccionado el período completo.								
Precondición	El usuario debe haber abierto la aplicación, debe haberse completado la carga inicial y debe haber accedido a la sección de Ranking.								
Secuencia normal	<table border="1"> <thead> <tr> <th>Paso</th> <th>Acción</th> </tr> </thead> <tbody> <tr> <td>P1</td> <td>El usuario ejecuta la aplicación.</td> </tr> <tr> <td>P2</td> <td>El usuario clica en la sección Ranking.</td> </tr> <tr> <td>P3</td> <td>El usuario selecciona el periodo.</td> </tr> </tbody> </table>	Paso	Acción	P1	El usuario ejecuta la aplicación.	P2	El usuario clica en la sección Ranking.	P3	El usuario selecciona el periodo.
Paso	Acción								
P1	El usuario ejecuta la aplicación.								
P2	El usuario clica en la sección Ranking.								
P3	El usuario selecciona el periodo.								
Postcondición	El usuario visualizará la información filtrada por el periodo marcado.								
Excepciones									
Comentarios	Ninguno								

APÉNDICE B. REQUISITOS FUNCIONALES

RF-20	Mostrar ranking cultivo										
Versión	1.00										
Objetivos asociados	OBJ-01 : Consulta de información de ranking										
Descripción	El sistema permitirá al usuario visualizar de forma gráfica un ranking de los 5 productos más cultivados, y los 5 menos cultivados. El sistema permitirá ver el total de superficie cultivada durante el período seleccionado y se representará el valor en hectáreas.										
Precondición	El usuario debe haber accedido a la sección de Ranking y se debe de haber marcado un período temporal.										
Secuencia normal	<table border="1"> <thead> <tr> <th>Paso</th> <th>Acción</th> </tr> </thead> <tbody> <tr> <td>P1</td> <td>El usuario ejecuta la aplicación.</td> </tr> <tr> <td>P2</td> <td>El usuario clicca en la sección Ranking.</td> </tr> <tr> <td>P3</td> <td>Dentro del top 5 de cultivo, el usuario selecciona la vista de los mejores o de los peores.</td> </tr> <tr> <td>P4</td> <td>El usuario accede a la información de los productos más o menos cultivados.</td> </tr> </tbody> </table>	Paso	Acción	P1	El usuario ejecuta la aplicación.	P2	El usuario clicca en la sección Ranking.	P3	Dentro del top 5 de cultivo, el usuario selecciona la vista de los mejores o de los peores.	P4	El usuario accede a la información de los productos más o menos cultivados.
Paso	Acción										
P1	El usuario ejecuta la aplicación.										
P2	El usuario clicca en la sección Ranking.										
P3	Dentro del top 5 de cultivo, el usuario selecciona la vista de los mejores o de los peores.										
P4	El usuario accede a la información de los productos más o menos cultivados.										
Postcondición	El usuario visualizará la información de forma gráfica.										
Excepciones											
Comentarios	Opcionalmente, podrá seleccionar un periodo temporal.										

RF-22	Mostrar ranking producción											
Versión	1.00											
Objetivos asociados	OBJ-01 : Consulta de información de ranking											
Descripción	El sistema permitirá al usuario visualizar de forma gráfica un ranking de los 5 productos más recolectados o menos recolectados. Se mostrará una sumatoria de la cantidad recolectada por producto durante el período temporal seleccionado. Esta información se mostrará en toneladas.											
Precondición	El usuario debe haber accedido a la sección de Ranking y se debe seleccionar un período temporal.											
Secuencia normal	<table border="1"> <thead> <tr> <th>Paso</th> <th>Acción</th> </tr> </thead> <tbody> <tr> <td>P1</td> <td>El usuario ejecuta la aplicación.</td> </tr> <tr> <td>P2</td> <td>El usuario clica en la sección Ranking.</td> </tr> <tr> <td>P3</td> <td>Dentro del top 5 de recolectados, el usuario selecciona la vista de los mejores o de los peores.</td> </tr> <tr> <td>P4</td> <td>El usuario accede a la información gráfica de los productos más recolectados.</td> </tr> </tbody> </table>		Paso	Acción	P1	El usuario ejecuta la aplicación.	P2	El usuario clica en la sección Ranking.	P3	Dentro del top 5 de recolectados, el usuario selecciona la vista de los mejores o de los peores.	P4	El usuario accede a la información gráfica de los productos más recolectados.
Paso	Acción											
P1	El usuario ejecuta la aplicación.											
P2	El usuario clica en la sección Ranking.											
P3	Dentro del top 5 de recolectados, el usuario selecciona la vista de los mejores o de los peores.											
P4	El usuario accede a la información gráfica de los productos más recolectados.											
Postcondición	El usuario visualizará la información de forma gráfica.											
Excepciones												
Comentarios	Opcionalmente, podrá seleccionar un periodo temporal.											

APÉNDICE B. REQUISITOS FUNCIONALES

RF-24	Mostrar ranking rendimiento											
Versión	1.00											
Objetivos asociados	OBJ-01 : Consulta de información de ranking											
Descripción	El sistema permitirá al usuario visualizar de forma gráfica un ranking de los 5 productos con mejor rendimiento o los 5 con peor rendimiento. El rendimiento se calculará como el total de kilogramos recogidos, dividido entre el total de metros cuadrados cultivados durante el periodo seleccionado. Los valores se mostrarán en Kg/m ³											
Precondición	El usuario debe haber accedido a la sección de Ranking.											
Secuencia normal	<table border="1"> <thead> <tr> <th>Paso</th> <th>Acción</th> </tr> </thead> <tbody> <tr> <td>P1</td> <td>El usuario ejecuta la aplicación.</td> </tr> <tr> <td>P2</td> <td>El usuario clica en la sección Ranking.</td> </tr> <tr> <td>P3</td> <td>Dentro del top 5 de rendimiento, el usuario selecciona la vista de los mejores o de los peores, según la información que desee visualizar.</td> </tr> <tr> <td>P4</td> <td>El usuario accede a la información de los productos con mejor o peor rendimiento.</td> </tr> </tbody> </table>		Paso	Acción	P1	El usuario ejecuta la aplicación.	P2	El usuario clica en la sección Ranking.	P3	Dentro del top 5 de rendimiento, el usuario selecciona la vista de los mejores o de los peores, según la información que desee visualizar.	P4	El usuario accede a la información de los productos con mejor o peor rendimiento.
Paso	Acción											
P1	El usuario ejecuta la aplicación.											
P2	El usuario clica en la sección Ranking.											
P3	Dentro del top 5 de rendimiento, el usuario selecciona la vista de los mejores o de los peores, según la información que desee visualizar.											
P4	El usuario accede a la información de los productos con mejor o peor rendimiento.											
Postcondición	El usuario visualizará la información de forma gráfica.											
Excepciones												
Comentarios	Opcionalmente, podrá seleccionar un periodo temporal.											

RF-26	Mostrar ranking ganancias											
Versión	1.00											
Objetivos asociados	OBJ-01 : Consulta de información de ranking											
Descripción	El sistema permitirá al usuario visualizar de forma gráfica un ranking de los 5 productos con mayores beneficios, o los 5 con menores beneficios. Para cada producto, los beneficios parciales se calcularán como la cantidad de Kg recogidos, multiplicado por el precio de mercado de esa semana. Los beneficios totales se calcularán como la sumatoria de todos los beneficios parciales. El calculo se hará siempre sobre el periodo temporal seleccionado											
Precondición	El usuario debe haber accedido a la sección de Ranking.											
Secuencia normal	<table border="1"> <thead> <tr> <th>Paso</th> <th>Acción</th> </tr> </thead> <tbody> <tr> <td>P1</td> <td>El usuario ejecuta la aplicación.</td> </tr> <tr> <td>P2</td> <td>El usuario clica en la sección Ranking.</td> </tr> <tr> <td>P3</td> <td>Dentro del top 5 de beneficios, el usuario selecciona la vista de los mejores o de los peores.</td> </tr> <tr> <td>P4</td> <td>El usuario accede a la información de los productos con las mayores o menores ganancias.</td> </tr> </tbody> </table>		Paso	Acción	P1	El usuario ejecuta la aplicación.	P2	El usuario clica en la sección Ranking.	P3	Dentro del top 5 de beneficios, el usuario selecciona la vista de los mejores o de los peores.	P4	El usuario accede a la información de los productos con las mayores o menores ganancias.
Paso	Acción											
P1	El usuario ejecuta la aplicación.											
P2	El usuario clica en la sección Ranking.											
P3	Dentro del top 5 de beneficios, el usuario selecciona la vista de los mejores o de los peores.											
P4	El usuario accede a la información de los productos con las mayores o menores ganancias.											
Postcondición	El usuario visualizará la información de forma gráfica.											
Excepciones												
Comentarios	Opcionalmente, podrá seleccionar un periodo temporal.											

APÉNDICE B. REQUISITOS FUNCIONALES

RF-28	Mostrar ranking precio										
Versión	1.00										
Objetivos asociados	OBJ-01 : Consulta de información de ranking										
Descripción	El sistema permitirá al usuario visualizar de forma gráfica un ranking de los 5 productos con mejor precio de venta o los 5 con peor precio de venta. Para ello se calculará la media del precio medio de venta durante el periodo seleccionado.										
Precondición	El usuario debe haber accedido a la sección de Ranking.										
Secuencia normal	<table border="1"> <thead> <tr> <th>Paso</th> <th>Acción</th> </tr> </thead> <tbody> <tr> <td>P1</td> <td>El usuario ejecuta la aplicación.</td> </tr> <tr> <td>P2</td> <td>El usuario clica en la sección Ranking.</td> </tr> <tr> <td>P3</td> <td>Dentro del top 5 de precio, el usuario selecciona la vista de los mejores o peores según le convenga.</td> </tr> <tr> <td>P4</td> <td>El usuario accede a la información de precio de los productos.</td> </tr> </tbody> </table>	Paso	Acción	P1	El usuario ejecuta la aplicación.	P2	El usuario clica en la sección Ranking.	P3	Dentro del top 5 de precio, el usuario selecciona la vista de los mejores o peores según le convenga.	P4	El usuario accede a la información de precio de los productos.
Paso	Acción										
P1	El usuario ejecuta la aplicación.										
P2	El usuario clica en la sección Ranking.										
P3	Dentro del top 5 de precio, el usuario selecciona la vista de los mejores o peores según le convenga.										
P4	El usuario accede a la información de precio de los productos.										
Postcondición	El usuario visualizará la información de forma gráfica.										
Excepciones											
Comentarios	Opcionalmente, podrá seleccionar un periodo temporal.										

RF-30	Mostrar ranking rentabilidad											
Versión	1.00											
Objetivos asociados	OBJ-01 : Consulta de información de ranking											
Descripción	El sistema permitirá al usuario visualizar de forma gráfica un ranking de los 5 productos con mayor rentabilidad o los 5 con rentabilidad más baja. La rentabilidad semanal se calcula como el total de Kg recolectados multiplicado por el precio de venta. La rentabilidad total se define como la media de rentabilidad semanal durante el periodo marcado.											
Precondición	El usuario debe haber accedido a la sección de Ranking.											
Secuencia normal	<table border="1"> <thead> <tr> <th>Paso</th> <th>Acción</th> </tr> </thead> <tbody> <tr> <td>P1</td> <td>El usuario ejecuta la aplicación.</td> </tr> <tr> <td>P2</td> <td>El usuario clica en la sección Ranking.</td> </tr> <tr> <td>P3</td> <td>Dentro del top 5 de rentabilidad, el usuario selecciona la vista de los mejores o peores según le convenga.</td> </tr> <tr> <td>P4</td> <td>El usuario accede a la información de rentabilidad de los productos.</td> </tr> </tbody> </table>		Paso	Acción	P1	El usuario ejecuta la aplicación.	P2	El usuario clica en la sección Ranking.	P3	Dentro del top 5 de rentabilidad, el usuario selecciona la vista de los mejores o peores según le convenga.	P4	El usuario accede a la información de rentabilidad de los productos.
Paso	Acción											
P1	El usuario ejecuta la aplicación.											
P2	El usuario clica en la sección Ranking.											
P3	Dentro del top 5 de rentabilidad, el usuario selecciona la vista de los mejores o peores según le convenga.											
P4	El usuario accede a la información de rentabilidad de los productos.											
Postcondición	El usuario visualizará la información de forma gráfica.											
Excepciones												
Comentarios	Opcionalmente, podrá seleccionar un periodo temporal.											

APÉNDICE B. REQUISITOS FUNCIONALES

RF-01	Acceder a página resumen						
Versión	1.00						
Objetivos asociados	OBJ-02 : Consulta de información resumen						
Descripción	El usuario podrá acceder a una sección donde encontrará información resumen acerca de distintos indicadores sobre los diferentes productos.						
Precondición	El usuario debe abrir la aplicación y se debe completar la carga inicial.						
Secuencia normal	<table border="1"> <thead> <tr> <th>Paso</th> <th>Acción</th> </tr> </thead> <tbody> <tr> <td>P1</td> <td>El usuario ejecuta la aplicación.</td> </tr> <tr> <td>P2</td> <td>El usuario clica en la sección Resumen.</td> </tr> </tbody> </table>	Paso	Acción	P1	El usuario ejecuta la aplicación.	P2	El usuario clica en la sección Resumen.
Paso	Acción						
P1	El usuario ejecuta la aplicación.						
P2	El usuario clica en la sección Resumen.						
Postcondición	El usuario accederá a la sección resumen, desde donde podrá ejecutar diversas acciones.						
Excepciones							
Comentarios	Ninguno						

RF-05	Mostrar resumen cultivo										
Versión	1.00										
Objetivos asociados	OBJ-02 : Consulta de información resumen										
Descripción	El usuario podrá ver un resumen en función de distintos indicadores. Uno de ellos es el cultivo. Podrá ver un resumen de la cantidad total cultivada en hectáreas, agrupada por producto, género, calidad, etc. En función del filtrado aplicado por el usuario.										
Precondición	El usuario debe abrir la aplicación y debe acceder a la sección de resumen.										
Secuencia normal	<table border="1"> <thead> <tr> <th>Paso</th> <th>Acción</th> </tr> </thead> <tbody> <tr> <td>P1</td> <td>El usuario ejecuta la aplicación.</td> </tr> <tr> <td>P2</td> <td>El usuario clica en la sección Resumen.</td> </tr> <tr> <td>P3</td> <td>El usuario selecciona el indicador de cultivo.</td> </tr> </tbody> </table>	Paso	Acción	P1	El usuario ejecuta la aplicación.	P2	El usuario clica en la sección Resumen.	P3	El usuario selecciona el indicador de cultivo.		
Paso	Acción										
P1	El usuario ejecuta la aplicación.										
P2	El usuario clica en la sección Resumen.										
P3	El usuario selecciona el indicador de cultivo.										
Secuencia alternativa	<table border="1"> <thead> <tr> <th>Paso</th> <th>Acción</th> </tr> </thead> <tbody> <tr> <td>P1</td> <td>El usuario ejecuta la aplicación.</td> </tr> <tr> <td>P2</td> <td>El usuario clica en la sección Resumen.</td> </tr> <tr> <td>P3</td> <td>El usuario selecciona el indicador de cultivo.</td> </tr> <tr> <td>P4</td> <td>El usuario selecciona un producto y un atributo por el que agrupar la información.</td> </tr> </tbody> </table>	Paso	Acción	P1	El usuario ejecuta la aplicación.	P2	El usuario clica en la sección Resumen.	P3	El usuario selecciona el indicador de cultivo.	P4	El usuario selecciona un producto y un atributo por el que agrupar la información.
Paso	Acción										
P1	El usuario ejecuta la aplicación.										
P2	El usuario clica en la sección Resumen.										
P3	El usuario selecciona el indicador de cultivo.										
P4	El usuario selecciona un producto y un atributo por el que agrupar la información.										
Postcondición	Se visualizará la información tanto de forma gráfica como en formato tabla.										
Excepciones											
Comentarios	Opcionalmente, podrá seleccionar un periodo temporal.										

APÉNDICE B. REQUISITOS FUNCIONALES

RF-32	Mostrar resumen producción										
Versión	1.00										
Objetivos asociados	OBJ-02 : Consulta de información resumen										
Descripción	El usuario podrá ver un resumen en función de distintos indicadores. Uno de ellos es la producción. Podrá ver un resumen de la cantidad total recolectada en toneladas, agrupada por producto, género, calidad, etc. En función del filtrado aplicado por el usuario.										
Precondición	El usuario debe abrir la aplicación y debe acceder a la sección de resumen.										
Secuencia normal	<table border="1"> <thead> <tr> <th>Paso</th> <th>Acción</th> </tr> </thead> <tbody> <tr> <td>P1</td> <td>El usuario ejecuta la aplicación.</td> </tr> <tr> <td>P2</td> <td>El usuario clicla en la sección Resumen.</td> </tr> <tr> <td>P3</td> <td>El usuario selecciona el indicador de producción.</td> </tr> </tbody> </table>	Paso	Acción	P1	El usuario ejecuta la aplicación.	P2	El usuario clicla en la sección Resumen.	P3	El usuario selecciona el indicador de producción.		
Paso	Acción										
P1	El usuario ejecuta la aplicación.										
P2	El usuario clicla en la sección Resumen.										
P3	El usuario selecciona el indicador de producción.										
Secuencia alternativa	<table border="1"> <thead> <tr> <th>Paso</th> <th>Acción</th> </tr> </thead> <tbody> <tr> <td>P1</td> <td>El usuario ejecuta la aplicación.</td> </tr> <tr> <td>P2</td> <td>El usuario clicla en la sección Resumen.</td> </tr> <tr> <td>P3</td> <td>El usuario selecciona el indicador de producción.</td> </tr> <tr> <td>P4</td> <td>El usuario selecciona un producto y un atributo por el que agrupar la información.</td> </tr> </tbody> </table>	Paso	Acción	P1	El usuario ejecuta la aplicación.	P2	El usuario clicla en la sección Resumen.	P3	El usuario selecciona el indicador de producción.	P4	El usuario selecciona un producto y un atributo por el que agrupar la información.
Paso	Acción										
P1	El usuario ejecuta la aplicación.										
P2	El usuario clicla en la sección Resumen.										
P3	El usuario selecciona el indicador de producción.										
P4	El usuario selecciona un producto y un atributo por el que agrupar la información.										
Postcondición	Se visualizará la información tanto de forma gráfica como en formato tabla.										
Excepciones											
Comentarios	Opcionalmente, podrá seleccionar un periodo temporal.										

RF-33	Mostrar resumen rendimiento										
Versión	1.00										
Objetivos asociados	OBJ-02 : Consulta de información resumen										
Descripción	El usuario podrá ver un resumen en función de distintos indicadores. Uno de ellos es el rendimiento. El rendimiento se define como la cantidad de kilogramos por metro cuadrado. Podrá ver un resumen del rendimiento total agrupado por producto, género, calidad, etc. En función del filtrado aplicado por el usuario.										
Precondición	El usuario debe abrir la aplicación y debe acceder a la sección de resumen.										
Secuencia normal	<table border="1"> <thead> <tr> <th>Paso</th> <th>Acción</th> </tr> </thead> <tbody> <tr> <td>P1</td> <td>El usuario ejecuta la aplicación.</td> </tr> <tr> <td>P2</td> <td>El usuario clica en la sección Resumen.</td> </tr> <tr> <td>P3</td> <td>El usuario selecciona el indicador de rendimiento.</td> </tr> </tbody> </table>	Paso	Acción	P1	El usuario ejecuta la aplicación.	P2	El usuario clica en la sección Resumen.	P3	El usuario selecciona el indicador de rendimiento.		
Paso	Acción										
P1	El usuario ejecuta la aplicación.										
P2	El usuario clica en la sección Resumen.										
P3	El usuario selecciona el indicador de rendimiento.										
Secuencia alternativa	<table border="1"> <thead> <tr> <th>Paso</th> <th>Acción</th> </tr> </thead> <tbody> <tr> <td>P1</td> <td>El usuario ejecuta la aplicación.</td> </tr> <tr> <td>P2</td> <td>El usuario clica en la sección Resumen.</td> </tr> <tr> <td>P3</td> <td>El usuario selecciona el indicador de rendimiento.</td> </tr> <tr> <td>P4</td> <td>El usuario selecciona un producto y un atributo por el que agrupar la información.</td> </tr> </tbody> </table>	Paso	Acción	P1	El usuario ejecuta la aplicación.	P2	El usuario clica en la sección Resumen.	P3	El usuario selecciona el indicador de rendimiento.	P4	El usuario selecciona un producto y un atributo por el que agrupar la información.
Paso	Acción										
P1	El usuario ejecuta la aplicación.										
P2	El usuario clica en la sección Resumen.										
P3	El usuario selecciona el indicador de rendimiento.										
P4	El usuario selecciona un producto y un atributo por el que agrupar la información.										
Postcondición	Se visualizará la información tanto de forma gráfica como en formato tabla.										
Excepciones											
Comentarios	Opcionalmente, podrá seleccionar un periodo temporal.										

APÉNDICE B. REQUISITOS FUNCIONALES

RF-34	Mostrar resumen ganancias										
Versión	1.00										
Objetivos asociados	OBJ-02 : Consulta de información resumen										
Descripción	El usuario podrá ver un resumen de distintos indicadores. Uno de ellos son los beneficios. Los beneficios se obtienen como el total de kilos recolectados multiplicado por el precio de mercado de ese periodo. Podrá ver un resumen de los beneficios totales agrupado por producto, género, calidad, etc. Esto depende del filtrado aplicado por el usuario.										
Precondición	El usuario debe abrir la aplicación y debe acceder a la sección de resumen.										
Secuencia normal	<table border="1"> <thead> <tr> <th>Paso</th> <th>Acción</th> </tr> </thead> <tbody> <tr> <td>P1</td> <td>El usuario ejecuta la aplicación.</td> </tr> <tr> <td>P2</td> <td>El usuario clicla en la sección Resumen.</td> </tr> <tr> <td>P3</td> <td>El usuario selecciona el indicador de beneficios.</td> </tr> </tbody> </table>	Paso	Acción	P1	El usuario ejecuta la aplicación.	P2	El usuario clicla en la sección Resumen.	P3	El usuario selecciona el indicador de beneficios.		
Paso	Acción										
P1	El usuario ejecuta la aplicación.										
P2	El usuario clicla en la sección Resumen.										
P3	El usuario selecciona el indicador de beneficios.										
Secuencia alternativa	<table border="1"> <thead> <tr> <th>Paso</th> <th>Acción</th> </tr> </thead> <tbody> <tr> <td>P1</td> <td>El usuario ejecuta la aplicación.</td> </tr> <tr> <td>P2</td> <td>El usuario clicla en la sección Resumen.</td> </tr> <tr> <td>P3</td> <td>El usuario selecciona el indicador de beneficios.</td> </tr> <tr> <td>P4</td> <td>El usuario selecciona un producto y un atributo por el que agrupar la información.</td> </tr> </tbody> </table>	Paso	Acción	P1	El usuario ejecuta la aplicación.	P2	El usuario clicla en la sección Resumen.	P3	El usuario selecciona el indicador de beneficios.	P4	El usuario selecciona un producto y un atributo por el que agrupar la información.
Paso	Acción										
P1	El usuario ejecuta la aplicación.										
P2	El usuario clicla en la sección Resumen.										
P3	El usuario selecciona el indicador de beneficios.										
P4	El usuario selecciona un producto y un atributo por el que agrupar la información.										
Postcondición	Se visualizará la información tanto de forma gráfica como en formato tabla.										
Excepciones											
Comentarios	Opcionalmente, podrá seleccionar un periodo temporal.										

RF-35	Mostrar resumen precio										
Versión	1.00										
Objetivos asociados	OBJ-02 : Consulta de información resumen										
Descripción	El usuario podrá ver un resumen de distintos indicadores. Uno de ellos es el precio. Podrá ver un resumen de la media del precio agrupado por producto, género, calidad, etc. Esto depende del filtrado aplicado por el usuario.										
Precondición	El usuario debe abrir la aplicación y debe acceder a la sección de resumen.										
Secuencia normal	<table border="1"> <thead> <tr> <th>Paso</th> <th>Acción</th> </tr> </thead> <tbody> <tr> <td>P1</td> <td>El usuario ejecuta la aplicación.</td> </tr> <tr> <td>P2</td> <td>El usuario clic en la sección Resumen.</td> </tr> <tr> <td>P3</td> <td>El usuario selecciona el indicador de precio.</td> </tr> </tbody> </table>	Paso	Acción	P1	El usuario ejecuta la aplicación.	P2	El usuario clic en la sección Resumen.	P3	El usuario selecciona el indicador de precio.		
Paso	Acción										
P1	El usuario ejecuta la aplicación.										
P2	El usuario clic en la sección Resumen.										
P3	El usuario selecciona el indicador de precio.										
Secuencia alternativa	<table border="1"> <thead> <tr> <th>Paso</th> <th>Acción</th> </tr> </thead> <tbody> <tr> <td>P1</td> <td>El usuario ejecuta la aplicación.</td> </tr> <tr> <td>P2</td> <td>El usuario clic en la sección Resumen.</td> </tr> <tr> <td>P3</td> <td>El usuario selecciona el indicador de precio.</td> </tr> <tr> <td>P4</td> <td>El usuario selecciona un producto y un atributo por el que agrupar la información.</td> </tr> </tbody> </table>	Paso	Acción	P1	El usuario ejecuta la aplicación.	P2	El usuario clic en la sección Resumen.	P3	El usuario selecciona el indicador de precio.	P4	El usuario selecciona un producto y un atributo por el que agrupar la información.
Paso	Acción										
P1	El usuario ejecuta la aplicación.										
P2	El usuario clic en la sección Resumen.										
P3	El usuario selecciona el indicador de precio.										
P4	El usuario selecciona un producto y un atributo por el que agrupar la información.										
Postcondición	Se visualizará la información tanto de forma gráfica como en formato tabla.										
Excepciones											
Comentarios	Opcionalmente, podrá seleccionar un periodo temporal.										

RF-36	Mostrar resumen rentabilidad										
Versión	1.00										
Objetivos asociados	OBJ-02 : Consulta de información resumen										
Descripción	El usuario podrá ver un resumen de distintos indicadores. Uno de ellos es la rentabilidad. La rentabilidad semanal se calcula como el total de Kg recolectados multiplicado por el precio de venta. La rentabilidad total se define como la media de rentabilidad semanal durante el periodo marcado. Podrá ver un resumen de la rentabilidad total agrupado por producto, género, calidad, etc. Esto depende del filtrado aplicado por el usuario.										
Precondición	El usuario debe abrir la aplicación y debe acceder a la sección de resumen.										
Secuencia normal	<table border="1"> <thead> <tr> <th>Paso</th> <th>Acción</th> </tr> </thead> <tbody> <tr> <td>P1</td> <td>El usuario ejecuta la aplicación.</td> </tr> <tr> <td>P2</td> <td>El usuario clica en la sección Resumen.</td> </tr> <tr> <td>P3</td> <td>El usuario selecciona el indicador de rentabilidad.</td> </tr> </tbody> </table>	Paso	Acción	P1	El usuario ejecuta la aplicación.	P2	El usuario clica en la sección Resumen.	P3	El usuario selecciona el indicador de rentabilidad.		
Paso	Acción										
P1	El usuario ejecuta la aplicación.										
P2	El usuario clica en la sección Resumen.										
P3	El usuario selecciona el indicador de rentabilidad.										
Secuencia alternativa	<table border="1"> <thead> <tr> <th>Paso</th> <th>Acción</th> </tr> </thead> <tbody> <tr> <td>P1</td> <td>El usuario ejecuta la aplicación.</td> </tr> <tr> <td>P2</td> <td>El usuario clica en la sección Resumen.</td> </tr> <tr> <td>P3</td> <td>El usuario selecciona el indicador de rentabilidad.</td> </tr> <tr> <td>P4</td> <td>El usuario selecciona un producto y un atributo por el que agrupar la información.</td> </tr> </tbody> </table>	Paso	Acción	P1	El usuario ejecuta la aplicación.	P2	El usuario clica en la sección Resumen.	P3	El usuario selecciona el indicador de rentabilidad.	P4	El usuario selecciona un producto y un atributo por el que agrupar la información.
Paso	Acción										
P1	El usuario ejecuta la aplicación.										
P2	El usuario clica en la sección Resumen.										
P3	El usuario selecciona el indicador de rentabilidad.										
P4	El usuario selecciona un producto y un atributo por el que agrupar la información.										
Postcondición	Se visualizará la información tanto de forma gráfica como en formato tabla.										
Excepciones											
Comentarios	Opcionalmente, podrá seleccionar un periodo temporal.										

RF-02	Seleccionar producto											
Versión	1.00											
Objetivos asociados	OBJ-02 : Consulta de información resumen											
Descripción	Se podrá elegir un producto concreto sobre el cual ver la información. El listado de productos contendrá distintos productos en función del indicador que se quiera visualizar.											
Precondición	El usuario debe abrir la aplicación y debe acceder a la sección de resumen. Además debe seleccionar el indicador sobre el que ver un resumen.											
Secuencia normal	<table border="1"> <thead> <tr> <th>Paso</th> <th>Acción</th> </tr> </thead> <tbody> <tr> <td>P1</td> <td>El usuario ejecuta la aplicación.</td> </tr> <tr> <td>P2</td> <td>El usuario clica en la sección Resumen.</td> </tr> <tr> <td>P3</td> <td>El usuario selecciona el indicador que le interesa para ver un resumen.</td> </tr> <tr> <td>P4</td> <td>El usuario selecciona un producto.</td> </tr> </tbody> </table>		Paso	Acción	P1	El usuario ejecuta la aplicación.	P2	El usuario clica en la sección Resumen.	P3	El usuario selecciona el indicador que le interesa para ver un resumen.	P4	El usuario selecciona un producto.
Paso	Acción											
P1	El usuario ejecuta la aplicación.											
P2	El usuario clica en la sección Resumen.											
P3	El usuario selecciona el indicador que le interesa para ver un resumen.											
P4	El usuario selecciona un producto.											
Postcondición	Una vez que selecciona el producto, podrá seleccionar un atributo sobre el que ver el resumen, por ejemplo el género o la calidad.											
Excepciones												
Comentarios	Ninguno											

APÉNDICE B. REQUISITOS FUNCIONALES

RF-03	Seleccionar filtro												
Versión	1.00												
Objetivos asociados	OBJ-02 : Consulta de información resumen												
Descripción	Se podrá elegir un atributo por el que agrupar la información. El listado de atributos contendrá aquellos para los que el producto seleccionado contiene información (género, calidad, calibre o aspecto).												
Precondición	El usuario debe abrir la aplicación y debe acceder a la sección de resumen. Además debe seleccionar el indicador sobre el que ver un resumen y elegir un producto concreto.												
Secuencia normal	<table border="1"> <thead> <tr> <th>Paso</th> <th>Acción</th> </tr> </thead> <tbody> <tr> <td>P1</td> <td>El usuario ejecuta la aplicación.</td> </tr> <tr> <td>P2</td> <td>El usuario clic en la sección Resumen.</td> </tr> <tr> <td>P3</td> <td>El usuario selecciona el indicador que le interesa para ver un resumen.</td> </tr> <tr> <td>P4</td> <td>El usuario selecciona un producto.</td> </tr> <tr> <td>P5</td> <td>El usuario selecciona un filtro o atributo.</td> </tr> </tbody> </table>	Paso	Acción	P1	El usuario ejecuta la aplicación.	P2	El usuario clic en la sección Resumen.	P3	El usuario selecciona el indicador que le interesa para ver un resumen.	P4	El usuario selecciona un producto.	P5	El usuario selecciona un filtro o atributo.
Paso	Acción												
P1	El usuario ejecuta la aplicación.												
P2	El usuario clic en la sección Resumen.												
P3	El usuario selecciona el indicador que le interesa para ver un resumen.												
P4	El usuario selecciona un producto.												
P5	El usuario selecciona un filtro o atributo.												
Postcondición	Una vez que selecciona el filtro, la información se agrupará por el atributo indicado y se visualizará tanto de forma gráfica como en formato de tabla.												
Excepciones													
Comentarios	Ninguno												

RF-04	Marcar periodo temporal														
Versión	1.00														
Objetivos asociados	OBJ-02 : Consulta de información resumen														
Descripción	Se podrá elegir un rango de años para filtrar la información que se muestra.														
Precondición	El usuario debe abrir la aplicación y debe acceder a la sección de resumen. Además debe seleccionar el indicador sobre el que ver un resumen. También podrá seleccionar un producto y un atributo por el que agrupar el resumen.														
Secuencia normal	<table border="1"> <thead> <tr> <th>Paso</th> <th>Acción</th> </tr> </thead> <tbody> <tr> <td>P1</td> <td>El usuario ejecuta la aplicación.</td> </tr> <tr> <td>P2</td> <td>El usuario clica en la sección Resumen.</td> </tr> <tr> <td>P3</td> <td>El usuario selecciona el indicador que le interesa para ver un resumen.</td> </tr> <tr> <td>P4</td> <td>El usuario marca un periodo temporal.</td> </tr> </tbody> </table>	Paso	Acción	P1	El usuario ejecuta la aplicación.	P2	El usuario clica en la sección Resumen.	P3	El usuario selecciona el indicador que le interesa para ver un resumen.	P4	El usuario marca un periodo temporal.				
Paso	Acción														
P1	El usuario ejecuta la aplicación.														
P2	El usuario clica en la sección Resumen.														
P3	El usuario selecciona el indicador que le interesa para ver un resumen.														
P4	El usuario marca un periodo temporal.														
Secuencia alternativa	<table border="1"> <thead> <tr> <th>Paso</th> <th>Acción</th> </tr> </thead> <tbody> <tr> <td>P1</td> <td>El usuario ejecuta la aplicación.</td> </tr> <tr> <td>P2</td> <td>El usuario clica en la sección Resumen.</td> </tr> <tr> <td>P3</td> <td>El usuario selecciona el indicador que le interesa para ver un resumen.</td> </tr> <tr> <td>P4</td> <td>El usuario selecciona un producto.</td> </tr> <tr> <td>P5</td> <td>El usuario selecciona un atributo o filtro por el que agrupar el resumen.</td> </tr> <tr> <td>P6</td> <td>El usuario marca un periodo temporal.</td> </tr> </tbody> </table>	Paso	Acción	P1	El usuario ejecuta la aplicación.	P2	El usuario clica en la sección Resumen.	P3	El usuario selecciona el indicador que le interesa para ver un resumen.	P4	El usuario selecciona un producto.	P5	El usuario selecciona un atributo o filtro por el que agrupar el resumen.	P6	El usuario marca un periodo temporal.
Paso	Acción														
P1	El usuario ejecuta la aplicación.														
P2	El usuario clica en la sección Resumen.														
P3	El usuario selecciona el indicador que le interesa para ver un resumen.														
P4	El usuario selecciona un producto.														
P5	El usuario selecciona un atributo o filtro por el que agrupar el resumen.														
P6	El usuario marca un periodo temporal.														
Postcondición	La información que se muestra corresponde al periodo temporal marcado.														
Excepciones															
Comentarios	Ninguno														

RF-37	Exportar información														
Versión	1.00														
Objetivos asociados	OBJ-02 : Consulta de información resumen														
Descripción	El sistema permitirá exportar la información de la tabla a formato TXT o CSV.														
Precondición	El usuario debe abrir la aplicación y debe acceder a la sección de resumen. Además debe seleccionar el indicador sobre el que ver un resumen. También podrá seleccionar un producto concreto y elegir un atributo por el que agrupar la información.														
Secuencia normal	<table border="1"> <thead> <tr> <th>Paso</th> <th>Acción</th> </tr> </thead> <tbody> <tr> <td>P1</td> <td>El usuario ejecuta la aplicación.</td> </tr> <tr> <td>P2</td> <td>El usuario clica en la sección Resumen.</td> </tr> <tr> <td>P3</td> <td>El usuario selecciona el indicador que le interesa para ver un resumen.</td> </tr> <tr> <td>P4</td> <td>El usuario elige un formato para exportar los datos y pulsa el botón para descargarlos.</td> </tr> </tbody> </table>	Paso	Acción	P1	El usuario ejecuta la aplicación.	P2	El usuario clica en la sección Resumen.	P3	El usuario selecciona el indicador que le interesa para ver un resumen.	P4	El usuario elige un formato para exportar los datos y pulsa el botón para descargarlos.				
Paso	Acción														
P1	El usuario ejecuta la aplicación.														
P2	El usuario clica en la sección Resumen.														
P3	El usuario selecciona el indicador que le interesa para ver un resumen.														
P4	El usuario elige un formato para exportar los datos y pulsa el botón para descargarlos.														
Secuencia alternativa	<table border="1"> <thead> <tr> <th>Paso</th> <th>Acción</th> </tr> </thead> <tbody> <tr> <td>P1</td> <td>El usuario ejecuta la aplicación.</td> </tr> <tr> <td>P2</td> <td>El usuario clica en la sección Resumen.</td> </tr> <tr> <td>P3</td> <td>El usuario selecciona el indicador que le interesa para ver un resumen.</td> </tr> <tr> <td>P4</td> <td>El usuario selecciona un producto.</td> </tr> <tr> <td>P5</td> <td>El usuario selecciona un atributo o filtro por el que agrupar el resumen.</td> </tr> <tr> <td>P5</td> <td>El usuario elige un formato para exportar los datos y pulsa el botón para descargarlos.</td> </tr> </tbody> </table>	Paso	Acción	P1	El usuario ejecuta la aplicación.	P2	El usuario clica en la sección Resumen.	P3	El usuario selecciona el indicador que le interesa para ver un resumen.	P4	El usuario selecciona un producto.	P5	El usuario selecciona un atributo o filtro por el que agrupar el resumen.	P5	El usuario elige un formato para exportar los datos y pulsa el botón para descargarlos.
Paso	Acción														
P1	El usuario ejecuta la aplicación.														
P2	El usuario clica en la sección Resumen.														
P3	El usuario selecciona el indicador que le interesa para ver un resumen.														
P4	El usuario selecciona un producto.														
P5	El usuario selecciona un atributo o filtro por el que agrupar el resumen.														
P5	El usuario elige un formato para exportar los datos y pulsa el botón para descargarlos.														
Postcondición	Los datos se almacenarán en su dispositivo.														
Excepciones															
Comentarios	Ninguno														

RF-38	Limpiar campos												
Versión	1.00												
Objetivos asociados	OBJ-02 : Consulta de información resumen												
Descripción	En cualquier momento, el usuario tendrá la opción de restablecer todos los campos a su valor predeterminado.												
Precondición	El usuario debe abrir la aplicación y debe acceder a la sección de resumen. Además debe seleccionar el indicador sobre el que ver un resumen y modificar alguno de los campos (producto, periodo, etc.).												
Secuencia normal	<table border="1"> <thead> <tr> <th>Paso</th> <th>Acción</th> </tr> </thead> <tbody> <tr> <td>P1</td> <td>El usuario ejecuta la aplicación.</td> </tr> <tr> <td>P2</td> <td>El usuario clic en la sección Resumen.</td> </tr> <tr> <td>P3</td> <td>El usuario selecciona el indicador que le interesa para ver un resumen.</td> </tr> <tr> <td>P4</td> <td>El usuario rellena datos del formulario.</td> </tr> <tr> <td>P5</td> <td>El usuario pulsa el botón para restablecer el valor de los campos.</td> </tr> </tbody> </table>	Paso	Acción	P1	El usuario ejecuta la aplicación.	P2	El usuario clic en la sección Resumen.	P3	El usuario selecciona el indicador que le interesa para ver un resumen.	P4	El usuario rellena datos del formulario.	P5	El usuario pulsa el botón para restablecer el valor de los campos.
Paso	Acción												
P1	El usuario ejecuta la aplicación.												
P2	El usuario clic en la sección Resumen.												
P3	El usuario selecciona el indicador que le interesa para ver un resumen.												
P4	El usuario rellena datos del formulario.												
P5	El usuario pulsa el botón para restablecer el valor de los campos.												
Postcondición	Los campos de entrada se reinician y la aplicación se restablece.												
Excepciones													
Comentarios	Ninguno												

RF-08	Acceder a página progreso						
Versión	1.00						
Objetivos asociados	OBJ-03 : Consulta de información evolutiva						
Descripción	El sistema permitirá al usuario acceder a la sección de progreso desde el menú.						
Precondición	El usuario debe abrir la aplicación y esperar hasta que se procese la carga inicial.						
Secuencia normal	<table border="1"> <thead> <tr> <th>Paso</th> <th>Acción</th> </tr> </thead> <tbody> <tr> <td>P1</td> <td>El usuario ejecuta la aplicación.</td> </tr> <tr> <td>P2</td> <td>El usuario clic en la sección Progreso.</td> </tr> </tbody> </table>	Paso	Acción	P1	El usuario ejecuta la aplicación.	P2	El usuario clic en la sección Progreso.
Paso	Acción						
P1	El usuario ejecuta la aplicación.						
P2	El usuario clic en la sección Progreso.						
Postcondición	El usuario accede a la sección de progreso, desde la cual puede ejecutar diversas acciones.						
Excepciones							
Comentarios	Ninguno						

APÉNDICE B. REQUISITOS FUNCIONALES

RF-09	Seleccionar tipo de información								
Versión	1.00								
Objetivos asociados	OBJ-03 : Consulta de información evolutiva								
Descripción	El usuario podrá elegir el tipo de información sobre la que ver un informe evolutivo. Las opciones que tiene son: climatológica, cultivo, producción y mercado								
Precondición	El usuario debe abrir la aplicación y acceder a la sección de progreso.								
Secuencia normal	<table border="1"> <thead> <tr> <th>Paso</th> <th>Acción</th> </tr> </thead> <tbody> <tr> <td>P1</td> <td>El usuario ejecuta la aplicación.</td> </tr> <tr> <td>P2</td> <td>El usuario clica en la sección Progreso.</td> </tr> <tr> <td>P3</td> <td>El usuario selecciona el tipo de información que desea visualizar.</td> </tr> </tbody> </table>	Paso	Acción	P1	El usuario ejecuta la aplicación.	P2	El usuario clica en la sección Progreso.	P3	El usuario selecciona el tipo de información que desea visualizar.
Paso	Acción								
P1	El usuario ejecuta la aplicación.								
P2	El usuario clica en la sección Progreso.								
P3	El usuario selecciona el tipo de información que desea visualizar.								
Postcondición	En función de la información que seleccione, deberá rellenar unos campos u otros.								
Excepciones									
Comentarios	Ninguno								

RF-10	Seleccionar variable climática											
Versión	1.00											
Objetivos asociados	OBJ-03 : Consulta de información evolutiva											
Descripción	Si el usuario desea ver información evolutiva sobre el clima, podrá elegir la variable climática que desea visualizar.											
Precondición	El usuario debe abrir la aplicación y acceder a la sección de progreso. Además, deberá seleccionar el tipo de información climática.											
Secuencia normal	<table border="1"> <thead> <tr> <th>Paso</th> <th>Acción</th> </tr> </thead> <tbody> <tr> <td>P1</td> <td>El usuario ejecuta la aplicación.</td> </tr> <tr> <td>P2</td> <td>El usuario clica en la sección Progreso.</td> </tr> <tr> <td>P3</td> <td>El usuario selecciona como tipo de información la climática.</td> </tr> <tr> <td>P4</td> <td>El usuario selecciona la variable climática.</td> </tr> </tbody> </table>		Paso	Acción	P1	El usuario ejecuta la aplicación.	P2	El usuario clica en la sección Progreso.	P3	El usuario selecciona como tipo de información la climática.	P4	El usuario selecciona la variable climática.
Paso	Acción											
P1	El usuario ejecuta la aplicación.											
P2	El usuario clica en la sección Progreso.											
P3	El usuario selecciona como tipo de información la climática.											
P4	El usuario selecciona la variable climática.											
Postcondición	El usuario podrá seleccionar uno o más valores de la distribución de datos para que se refleje en la gráfica y en la tabla. Además podrá marcar un período temporal.											
Excepciones												
Comentarios	Ninguno											

APÉNDICE B. REQUISITOS FUNCIONALES

RF-11	Seleccionar producto										
Versión	1.00										
Objetivos asociados	OBJ-03 : Consulta de información evolutiva										
Descripción	Si el usuario desea ver información evolutiva sobre cultivo, producción o mercado, podrá elegir el producto sobre el que desea visualizar esa información.										
Precondición	El usuario debe abrir la aplicación y acceder a la sección de progreso. Además, deberá seleccionar como tipo de información: cultivo, producción o mercado.										
Secuencia normal	<table border="1"> <thead> <tr> <th>Paso</th> <th>Acción</th> </tr> </thead> <tbody> <tr> <td>P1</td> <td>El usuario ejecuta la aplicación.</td> </tr> <tr> <td>P2</td> <td>El usuario clicla en la sección Progreso.</td> </tr> <tr> <td>P3</td> <td>El usuario selecciona como tipo de información cultivo, producción o mercado.</td> </tr> <tr> <td>P4</td> <td>El usuario selecciona el producto.</td> </tr> </tbody> </table>	Paso	Acción	P1	El usuario ejecuta la aplicación.	P2	El usuario clicla en la sección Progreso.	P3	El usuario selecciona como tipo de información cultivo, producción o mercado.	P4	El usuario selecciona el producto.
Paso	Acción										
P1	El usuario ejecuta la aplicación.										
P2	El usuario clicla en la sección Progreso.										
P3	El usuario selecciona como tipo de información cultivo, producción o mercado.										
P4	El usuario selecciona el producto.										
Postcondición	El usuario podrá seleccionar el tipo de atributo por el que quiere filtrar la información.										
Excepciones											
Comentarios	Ninguno										

RF-12	Seleccionar tipo de filtrado													
Versión	1.00													
Objetivos asociados	OBJ-03 : Consulta de información evolutiva													
Descripción	Una vez que el usuario selecciona un producto, podrá elegir un tipo de atributo por el que filtrar la información evolutiva de ese producto.													
Precondición	El usuario debe abrir la aplicación y acceder a la sección de progreso. Además, deberá seleccionar como tipo de información: cultivo, producción o mercado. Deberá elegir un producto.													
Secuencia normal	<table border="1"> <thead> <tr> <th>Paso</th> <th>Acción</th> </tr> </thead> <tbody> <tr> <td>P1</td> <td>El usuario ejecuta la aplicación.</td> </tr> <tr> <td>P2</td> <td>El usuario clic en la sección Progreso.</td> </tr> <tr> <td>P3</td> <td>El usuario selecciona como tipo de información cultivo, producción o mercado.</td> </tr> <tr> <td>P4</td> <td>El usuario selecciona el producto.</td> </tr> <tr> <td>P5</td> <td>El usuario selecciona un tipo de filtro.</td> </tr> </tbody> </table>		Paso	Acción	P1	El usuario ejecuta la aplicación.	P2	El usuario clic en la sección Progreso.	P3	El usuario selecciona como tipo de información cultivo, producción o mercado.	P4	El usuario selecciona el producto.	P5	El usuario selecciona un tipo de filtro.
Paso	Acción													
P1	El usuario ejecuta la aplicación.													
P2	El usuario clic en la sección Progreso.													
P3	El usuario selecciona como tipo de información cultivo, producción o mercado.													
P4	El usuario selecciona el producto.													
P5	El usuario selecciona un tipo de filtro.													
Postcondición	El usuario podrá seleccionar un valor para el filtro seleccionado.													
Excepciones														
Comentarios	Ninguno													

APÉNDICE B. REQUISITOS FUNCIONALES

RF-39	Seleccionar filtro														
Versión	1.00														
Objetivos asociados	OBJ-03 : Consulta de información evolutiva														
Descripción	Una vez que el usuario elija un tipo de atributo para filtrar la información, podrá elegir un valor para ese atributo.														
Precondición	El usuario debe abrir la aplicación y acceder a la sección de progreso. Además, deberá seleccionar como tipo de información: cultivo, producción o mercado. Deberá elegir un producto y un tipo de filtro (género, calidad, calibre o aspecto).														
Secuencia normal	<table border="1"> <thead> <tr> <th>Paso</th> <th>Acción</th> </tr> </thead> <tbody> <tr> <td>P1</td> <td>El usuario ejecuta la aplicación.</td> </tr> <tr> <td>P2</td> <td>El usuario clicla en la sección Progreso.</td> </tr> <tr> <td>P3</td> <td>El usuario selecciona como tipo de información cultivo, producción o mercado.</td> </tr> <tr> <td>P4</td> <td>El usuario selecciona el producto.</td> </tr> <tr> <td>P5</td> <td>El usuario selecciona un tipo de filtro.</td> </tr> <tr> <td>P6</td> <td>El usuario selecciona un valor para el filtro.</td> </tr> </tbody> </table>	Paso	Acción	P1	El usuario ejecuta la aplicación.	P2	El usuario clicla en la sección Progreso.	P3	El usuario selecciona como tipo de información cultivo, producción o mercado.	P4	El usuario selecciona el producto.	P5	El usuario selecciona un tipo de filtro.	P6	El usuario selecciona un valor para el filtro.
Paso	Acción														
P1	El usuario ejecuta la aplicación.														
P2	El usuario clicla en la sección Progreso.														
P3	El usuario selecciona como tipo de información cultivo, producción o mercado.														
P4	El usuario selecciona el producto.														
P5	El usuario selecciona un tipo de filtro.														
P6	El usuario selecciona un valor para el filtro.														
Postcondición	Si la información es de mercado, podrá seleccionar que valores de la distribución de datos mostrar (Min, Q1, med, mean, Q3 o Máx) en la gráfica y en la tabla, y además podrá marcar un periodo temporal.														
Excepciones															
Comentarios	Ninguno														

RF-13	Marcar período temporal																
Versión	1.00																
Objetivos asociados	OBJ-03 : Consulta de información evolutiva																
Descripción	El usuario podrá filtrar la información por años.																
Precondición	Se debe de haber indicado o bien una variable climática, o bien un producto, tipo de filtro y valor del filtro.																
Secuencia normal	<table border="1"> <thead> <tr> <th>Paso</th> <th>Acción</th> </tr> </thead> <tbody> <tr> <td>P1</td> <td>El usuario ejecuta la aplicación.</td> </tr> <tr> <td>P2</td> <td>El usuario clica en la sección Progreso.</td> </tr> <tr> <td>P3</td> <td>El usuario selecciona como tipo de información cultivo, producción o mercado.</td> </tr> <tr> <td>P4</td> <td>El usuario selecciona el producto.</td> </tr> <tr> <td>P5</td> <td>El usuario selecciona un tipo de filtro.</td> </tr> <tr> <td>P6</td> <td>El usuario selecciona un valor para el filtro.</td> </tr> <tr> <td>P7</td> <td>El usuario selecciona un periodo temporal.</td> </tr> </tbody> </table>	Paso	Acción	P1	El usuario ejecuta la aplicación.	P2	El usuario clica en la sección Progreso.	P3	El usuario selecciona como tipo de información cultivo, producción o mercado.	P4	El usuario selecciona el producto.	P5	El usuario selecciona un tipo de filtro.	P6	El usuario selecciona un valor para el filtro.	P7	El usuario selecciona un periodo temporal.
Paso	Acción																
P1	El usuario ejecuta la aplicación.																
P2	El usuario clica en la sección Progreso.																
P3	El usuario selecciona como tipo de información cultivo, producción o mercado.																
P4	El usuario selecciona el producto.																
P5	El usuario selecciona un tipo de filtro.																
P6	El usuario selecciona un valor para el filtro.																
P7	El usuario selecciona un periodo temporal.																
Secuencia alternativa	<table border="1"> <thead> <tr> <th>Paso</th> <th>Acción</th> </tr> </thead> <tbody> <tr> <td>P1</td> <td>El usuario ejecuta la aplicación.</td> </tr> <tr> <td>P2</td> <td>El usuario clica en la sección Progreso.</td> </tr> <tr> <td>P3</td> <td>El usuario selecciona como tipo de información climática.</td> </tr> <tr> <td>P4</td> <td>El usuario selecciona una variable climática.</td> </tr> <tr> <td>P5</td> <td>El usuario selecciona un periodo temporal.</td> </tr> </tbody> </table>	Paso	Acción	P1	El usuario ejecuta la aplicación.	P2	El usuario clica en la sección Progreso.	P3	El usuario selecciona como tipo de información climática.	P4	El usuario selecciona una variable climática.	P5	El usuario selecciona un periodo temporal.				
Paso	Acción																
P1	El usuario ejecuta la aplicación.																
P2	El usuario clica en la sección Progreso.																
P3	El usuario selecciona como tipo de información climática.																
P4	El usuario selecciona una variable climática.																
P5	El usuario selecciona un periodo temporal.																
Postcondición	Se visualiza la gráfica y la tabla filtradas por el periodo seleccionado.																
Excepciones																	
Comentarios	Ninguno																

APÉNDICE B. REQUISITOS FUNCIONALES

RF-14	Mostrar información																
Versión	1.00																
Objetivos asociados	OBJ-03 : Consulta de información evolutiva																
Descripción	El usuario podrá ver una gráfica evolutiva sobre distintas variables e indicadores, así como una tabla.																
Precondición	Se debe de haber indicado o bien una variable climática, o bien un producto, tipo de filtro y valor del filtro.																
Secuencia normal	<table border="1"> <thead> <tr> <th>Paso</th> <th>Acción</th> </tr> </thead> <tbody> <tr> <td>P1</td> <td>El usuario ejecuta la aplicación.</td> </tr> <tr> <td>P2</td> <td>El usuario clica en la sección Progreso.</td> </tr> <tr> <td>P3</td> <td>El usuario selecciona como tipo de información cultivo, producción o mercado.</td> </tr> <tr> <td>P4</td> <td>El usuario selecciona el producto.</td> </tr> <tr> <td>P5</td> <td>El usuario selecciona un tipo de filtro.</td> </tr> <tr> <td>P6</td> <td>El usuario selecciona un valor para el filtro.</td> </tr> <tr> <td>P7</td> <td>El usuario visualiza la información.</td> </tr> </tbody> </table>	Paso	Acción	P1	El usuario ejecuta la aplicación.	P2	El usuario clica en la sección Progreso.	P3	El usuario selecciona como tipo de información cultivo, producción o mercado.	P4	El usuario selecciona el producto.	P5	El usuario selecciona un tipo de filtro.	P6	El usuario selecciona un valor para el filtro.	P7	El usuario visualiza la información.
Paso	Acción																
P1	El usuario ejecuta la aplicación.																
P2	El usuario clica en la sección Progreso.																
P3	El usuario selecciona como tipo de información cultivo, producción o mercado.																
P4	El usuario selecciona el producto.																
P5	El usuario selecciona un tipo de filtro.																
P6	El usuario selecciona un valor para el filtro.																
P7	El usuario visualiza la información.																
Secuencia alternativa	<table border="1"> <thead> <tr> <th>Paso</th> <th>Acción</th> </tr> </thead> <tbody> <tr> <td>P1</td> <td>El usuario ejecuta la aplicación.</td> </tr> <tr> <td>P2</td> <td>El usuario clica en la sección Progreso.</td> </tr> <tr> <td>P3</td> <td>El usuario selecciona como tipo de información climática.</td> </tr> <tr> <td>P4</td> <td>El usuario selecciona una variable climática.</td> </tr> <tr> <td>P5</td> <td>El usuario visualiza la información.</td> </tr> </tbody> </table>	Paso	Acción	P1	El usuario ejecuta la aplicación.	P2	El usuario clica en la sección Progreso.	P3	El usuario selecciona como tipo de información climática.	P4	El usuario selecciona una variable climática.	P5	El usuario visualiza la información.				
Paso	Acción																
P1	El usuario ejecuta la aplicación.																
P2	El usuario clica en la sección Progreso.																
P3	El usuario selecciona como tipo de información climática.																
P4	El usuario selecciona una variable climática.																
P5	El usuario visualiza la información.																
Postcondición																	
Excepciones																	
Comentarios	Opcionalmente, podrá filtrar por un periodo concreto. En algunos indicadores como la información climatológica o información de mercado, podrá marcar distintos valores de la distribución para verlos de forma simultánea tanto en la gráfica como en la tabla.																

RF-15	Habilitar o deshabilitar opciones																
Versión	1.00																
Objetivos asociados	OBJ-03 : Consulta de información evolutiva																
Descripción	El usuario podrá habilitar o deshabilitar los distintos valores de la distribución de datos. Los valores son mínimo, primer cuartil, media, mediana, tercer cuartil y máximo.																
Precondición	Esta opción solamente estará disponible para la información climática y de mercado.																
Secuencia normal	<table border="1"> <thead> <tr> <th>Paso</th> <th>Acción</th> </tr> </thead> <tbody> <tr> <td>P1</td> <td>El usuario ejecuta la aplicación.</td> </tr> <tr> <td>P2</td> <td>El usuario clica en la sección Progreso.</td> </tr> <tr> <td>P3</td> <td>El usuario selecciona como tipo de información mercado.</td> </tr> <tr> <td>P4</td> <td>El usuario selecciona el producto.</td> </tr> <tr> <td>P5</td> <td>El usuario selecciona un tipo de filtro.</td> </tr> <tr> <td>P6</td> <td>El usuario selecciona un valor para el filtro.</td> </tr> <tr> <td>P7</td> <td>El usuario marca aquellos valores de la distribución que desea visualizar.</td> </tr> </tbody> </table>	Paso	Acción	P1	El usuario ejecuta la aplicación.	P2	El usuario clica en la sección Progreso.	P3	El usuario selecciona como tipo de información mercado.	P4	El usuario selecciona el producto.	P5	El usuario selecciona un tipo de filtro.	P6	El usuario selecciona un valor para el filtro.	P7	El usuario marca aquellos valores de la distribución que desea visualizar.
Paso	Acción																
P1	El usuario ejecuta la aplicación.																
P2	El usuario clica en la sección Progreso.																
P3	El usuario selecciona como tipo de información mercado.																
P4	El usuario selecciona el producto.																
P5	El usuario selecciona un tipo de filtro.																
P6	El usuario selecciona un valor para el filtro.																
P7	El usuario marca aquellos valores de la distribución que desea visualizar.																
Secuencia alternativa	<table border="1"> <thead> <tr> <th>Paso</th> <th>Acción</th> </tr> </thead> <tbody> <tr> <td>P1</td> <td>El usuario ejecuta la aplicación.</td> </tr> <tr> <td>P2</td> <td>El usuario clica en la sección Progreso.</td> </tr> <tr> <td>P3</td> <td>El usuario selecciona como tipo de información climática.</td> </tr> <tr> <td>P4</td> <td>El usuario selecciona una variable climática.</td> </tr> <tr> <td>P5</td> <td>El usuario marca aquellos valores de la distribución que desea visualizar.</td> </tr> </tbody> </table>	Paso	Acción	P1	El usuario ejecuta la aplicación.	P2	El usuario clica en la sección Progreso.	P3	El usuario selecciona como tipo de información climática.	P4	El usuario selecciona una variable climática.	P5	El usuario marca aquellos valores de la distribución que desea visualizar.				
Paso	Acción																
P1	El usuario ejecuta la aplicación.																
P2	El usuario clica en la sección Progreso.																
P3	El usuario selecciona como tipo de información climática.																
P4	El usuario selecciona una variable climática.																
P5	El usuario marca aquellos valores de la distribución que desea visualizar.																
Postcondición	El usuario visualiza la información de todos los valores marcados.																
Excepciones																	
Comentarios	Opcionalmente podrá marcar un periodo temporal.																

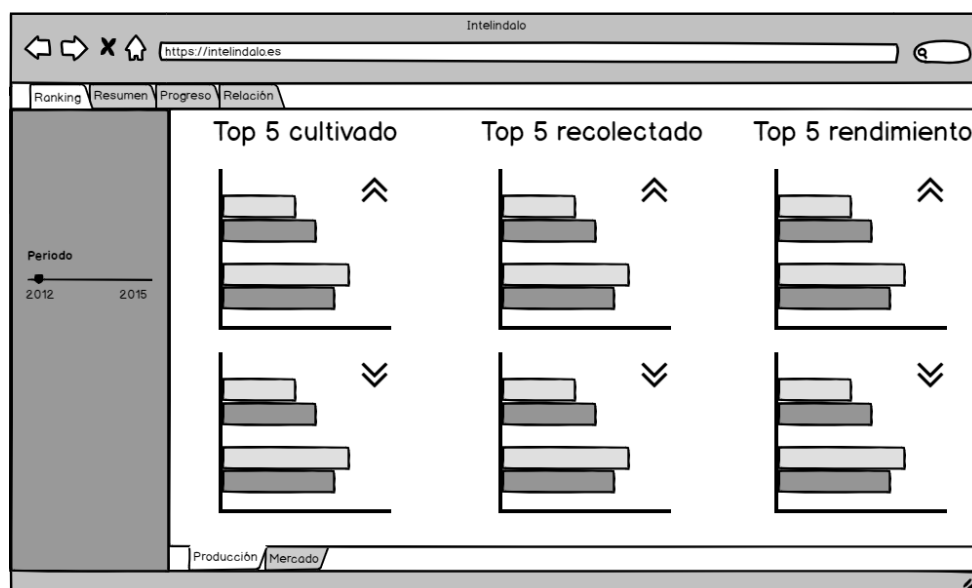
APÉNDICE B. REQUISITOS FUNCIONALES

RF-16	Exportar tabla																		
Versión	1.00																		
Objetivos asociados	OBJ-03 : Consulta de información evolutiva																		
Descripción	El usuario podrá exportar a TXT o CSV la información que visualiza en forma de tabla.																		
Precondición	El usuario debe haber llegado a visualizar la información que necesita y debe seleccionar la vista de tabla.																		
Secuencia normal	<table border="1"> <thead> <tr> <th>Paso</th> <th>Acción</th> </tr> </thead> <tbody> <tr> <td>P1</td> <td>El usuario ejecuta la aplicación.</td> </tr> <tr> <td>P2</td> <td>El usuario clica en la sección Progreso.</td> </tr> <tr> <td>P3</td> <td>El usuario selecciona como tipo de información cultivo, producción o mercado.</td> </tr> <tr> <td>P4</td> <td>El usuario selecciona el producto.</td> </tr> <tr> <td>P5</td> <td>El usuario selecciona un tipo de filtro.</td> </tr> <tr> <td>P6</td> <td>El usuario selecciona un valor para el filtro.</td> </tr> <tr> <td>P7</td> <td>El usuario selecciona la vista de tabla.</td> </tr> <tr> <td>P8</td> <td>El usuario elige el formato y descarga la tabla.</td> </tr> </tbody> </table>	Paso	Acción	P1	El usuario ejecuta la aplicación.	P2	El usuario clica en la sección Progreso.	P3	El usuario selecciona como tipo de información cultivo, producción o mercado.	P4	El usuario selecciona el producto.	P5	El usuario selecciona un tipo de filtro.	P6	El usuario selecciona un valor para el filtro.	P7	El usuario selecciona la vista de tabla.	P8	El usuario elige el formato y descarga la tabla.
Paso	Acción																		
P1	El usuario ejecuta la aplicación.																		
P2	El usuario clica en la sección Progreso.																		
P3	El usuario selecciona como tipo de información cultivo, producción o mercado.																		
P4	El usuario selecciona el producto.																		
P5	El usuario selecciona un tipo de filtro.																		
P6	El usuario selecciona un valor para el filtro.																		
P7	El usuario selecciona la vista de tabla.																		
P8	El usuario elige el formato y descarga la tabla.																		
Secuencia alternativa	<table border="1"> <thead> <tr> <th>Paso</th> <th>Acción</th> </tr> </thead> <tbody> <tr> <td>P1</td> <td>El usuario ejecuta la aplicación.</td> </tr> <tr> <td>P2</td> <td>El usuario clica en la sección Progreso.</td> </tr> <tr> <td>P3</td> <td>El usuario selecciona como tipo de información climática.</td> </tr> <tr> <td>P4</td> <td>El usuario selecciona una variable climática.</td> </tr> <tr> <td>P5</td> <td>El usuario selecciona la vista de tabla.</td> </tr> <tr> <td>P6</td> <td>El usuario elige el formato y descarga la tabla.</td> </tr> </tbody> </table>	Paso	Acción	P1	El usuario ejecuta la aplicación.	P2	El usuario clica en la sección Progreso.	P3	El usuario selecciona como tipo de información climática.	P4	El usuario selecciona una variable climática.	P5	El usuario selecciona la vista de tabla.	P6	El usuario elige el formato y descarga la tabla.				
Paso	Acción																		
P1	El usuario ejecuta la aplicación.																		
P2	El usuario clica en la sección Progreso.																		
P3	El usuario selecciona como tipo de información climática.																		
P4	El usuario selecciona una variable climática.																		
P5	El usuario selecciona la vista de tabla.																		
P6	El usuario elige el formato y descarga la tabla.																		
Postcondición	Los datos se almacenarán en el dispositivo del usuario. Excepciones																		
Comentarios	Opcionalmente podrá cambiar el periodo temporal. Para la información climatológica y la de mercado, podrá seleccionar que valores de la distribución quiere mostrar.																		

RF-17	Limpiar campos										
Versión	1.00										
Objetivos asociados	OBJ-03 : Consulta de información evolutiva										
Descripción	En cualquier momento, el usuario tendrá la opción de restablecer todos los campos a su valor predeterminado.										
Precondición	El usuario debe abrir la aplicación y debe acceder a la sección de progreso. Debe modificar alguno/s de los campos.										
Secuencia normal	<table border="1"> <thead> <tr> <th>Paso</th> <th>Acción</th> </tr> </thead> <tbody> <tr> <td>P1</td> <td>El usuario ejecuta la aplicación.</td> </tr> <tr> <td>P2</td> <td>El usuario clica en la sección Progreso.</td> </tr> <tr> <td>P3</td> <td>El usuario rellena datos del formulario.</td> </tr> <tr> <td>P4</td> <td>El usuario pulsa el botón para restablecer el valor de los campos.</td> </tr> </tbody> </table>	Paso	Acción	P1	El usuario ejecuta la aplicación.	P2	El usuario clica en la sección Progreso.	P3	El usuario rellena datos del formulario.	P4	El usuario pulsa el botón para restablecer el valor de los campos.
Paso	Acción										
P1	El usuario ejecuta la aplicación.										
P2	El usuario clica en la sección Progreso.										
P3	El usuario rellena datos del formulario.										
P4	El usuario pulsa el botón para restablecer el valor de los campos.										
Postcondición	Los campos de entrada se reinician y la aplicación se restablece.										
Excepciones											
Comentarios	Ninguno										

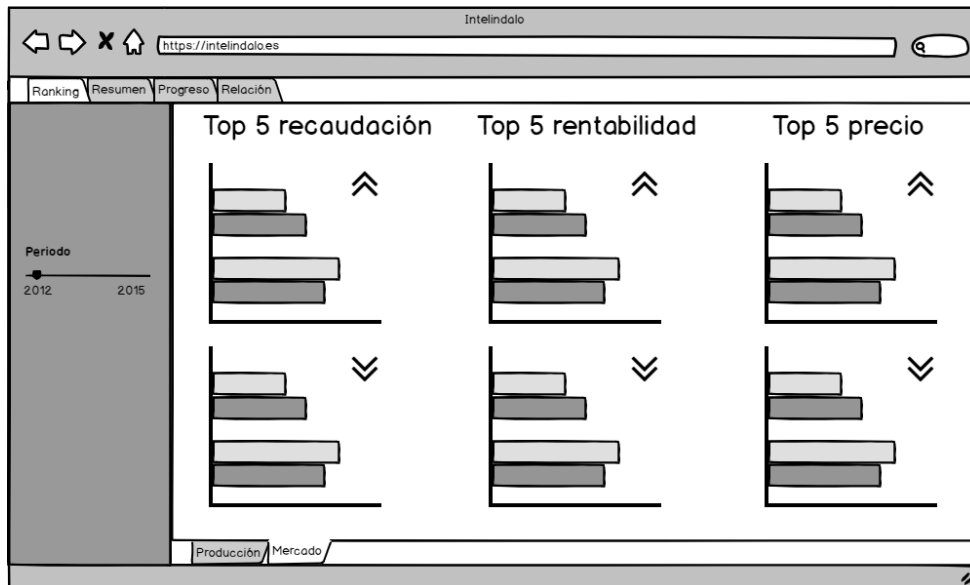
Apéndice C

Prototipos iniciales

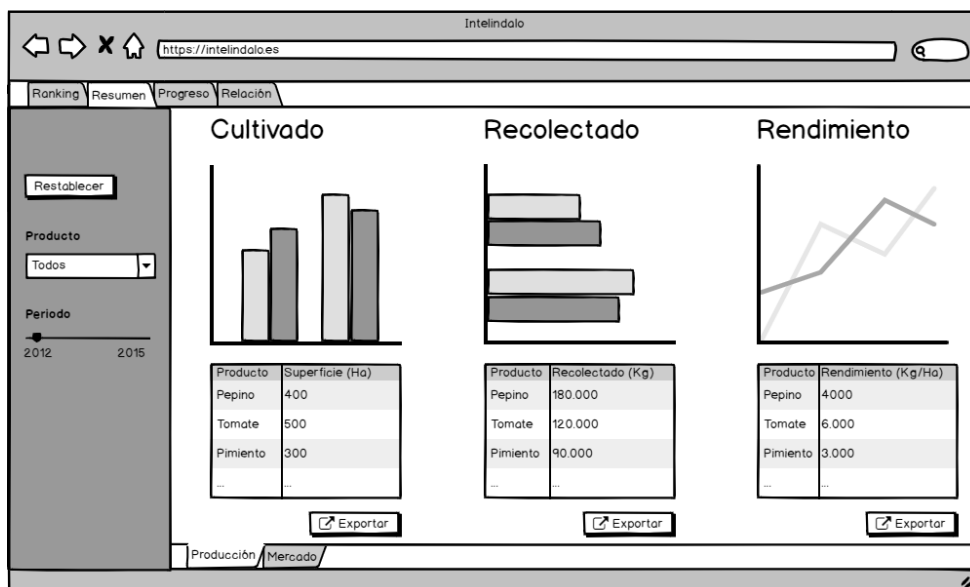


Fuente: autoría propia

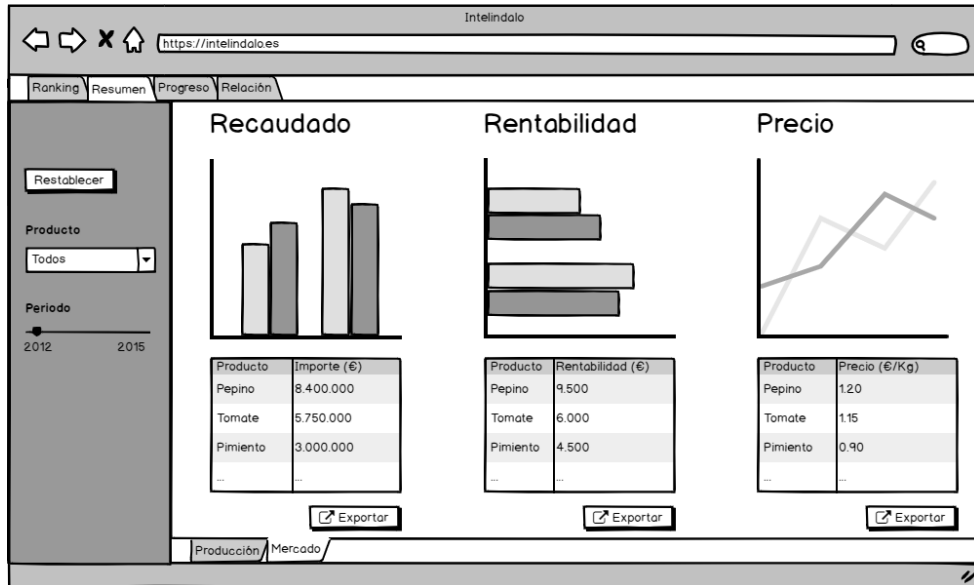
Figura C.1: Prototipo inicial para la página de ranking (1).



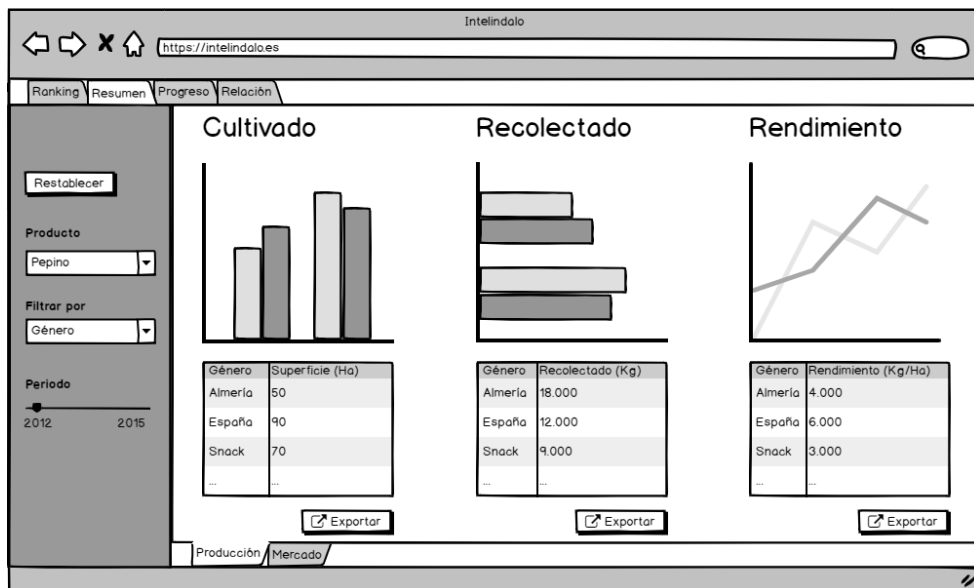
Fuente: autoría propia
 Figura C.2: Prototipo inicial para la página de ranking (2).



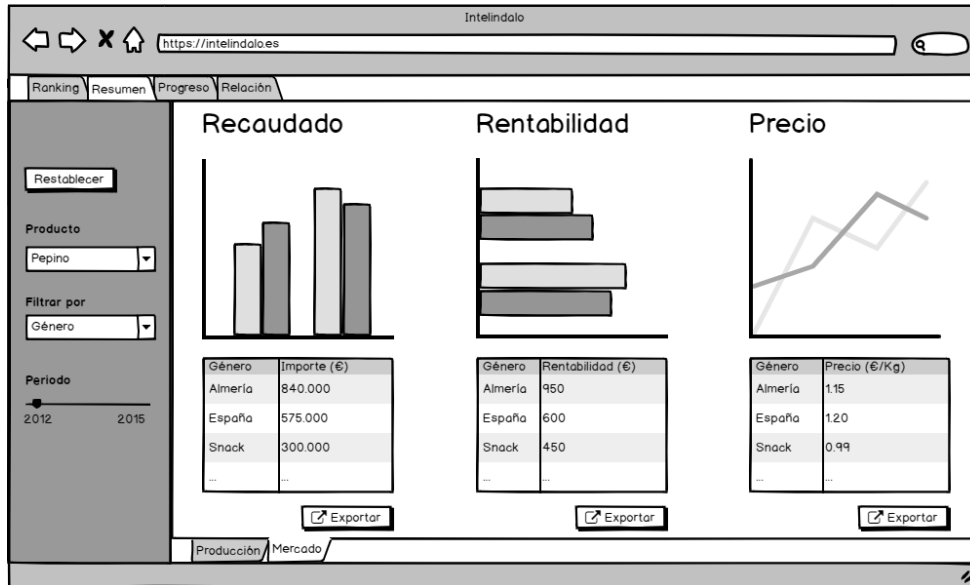
Fuente: autoría propia
 Figura C.3: Prototipo inicial para la página de resumen (1).



Fuente: autoría propia
 Figura C.4: Prototipo inicial para la página de resumen (2).

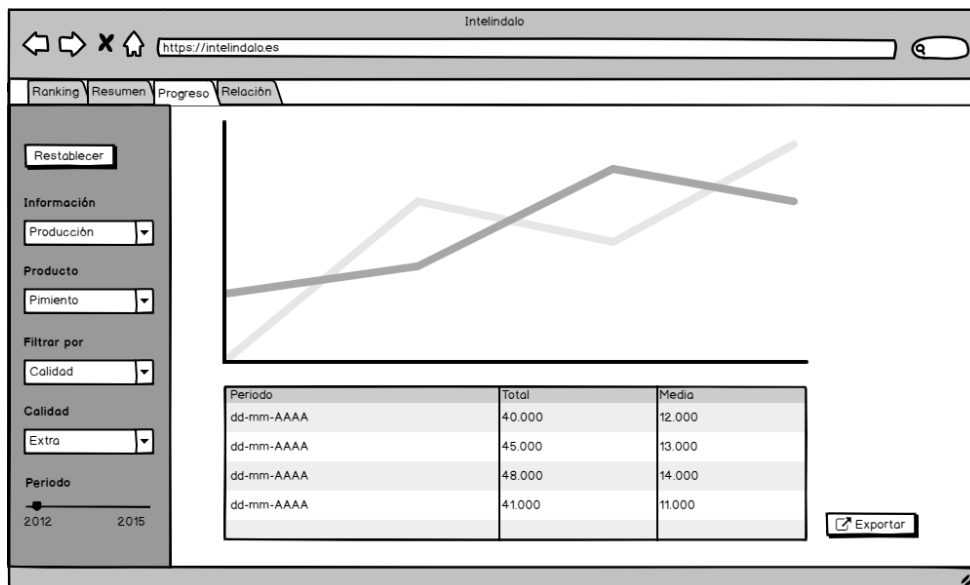


Fuente: autoría propia
 Figura C.5: Prototipo inicial para la página de resumen (3).



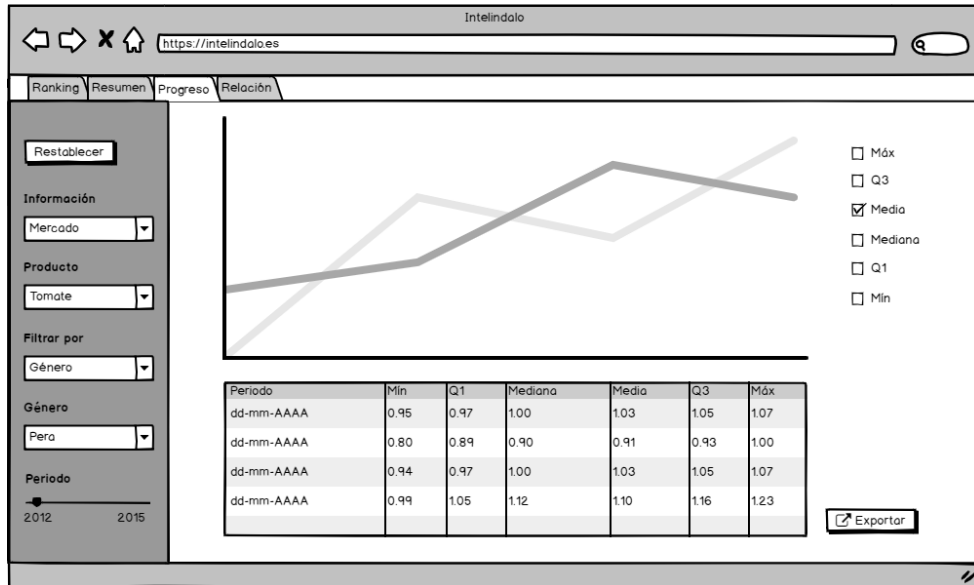
Fuente: autoría propia

Figura C.6: Prototipo inicial para la página de resumen (4).



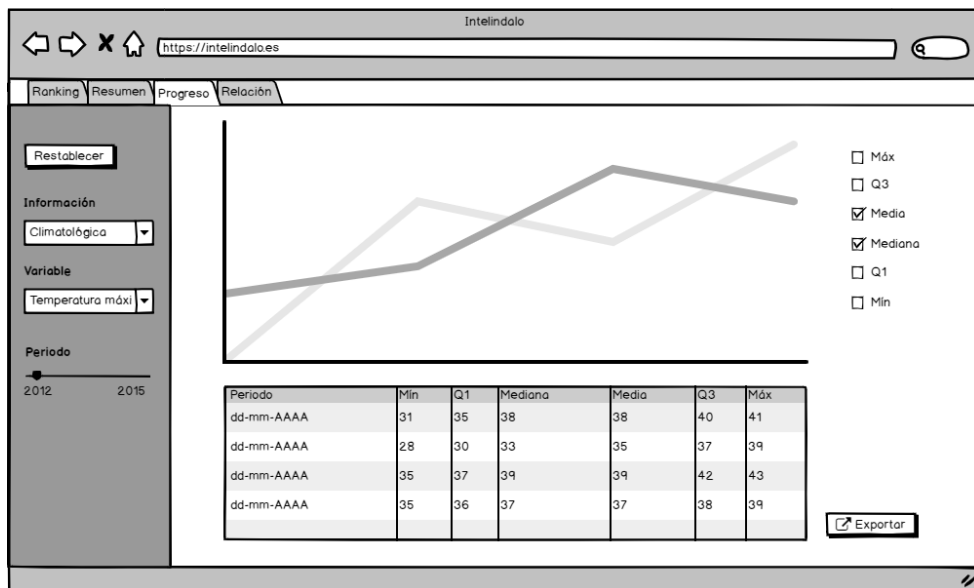
Fuente: autoría propia

Figura C.7: Prototipo inicial para la página de progreso (1).



Fuente: autoría propia

Figura C.8: Prototipo inicial para la página de progreso (2).



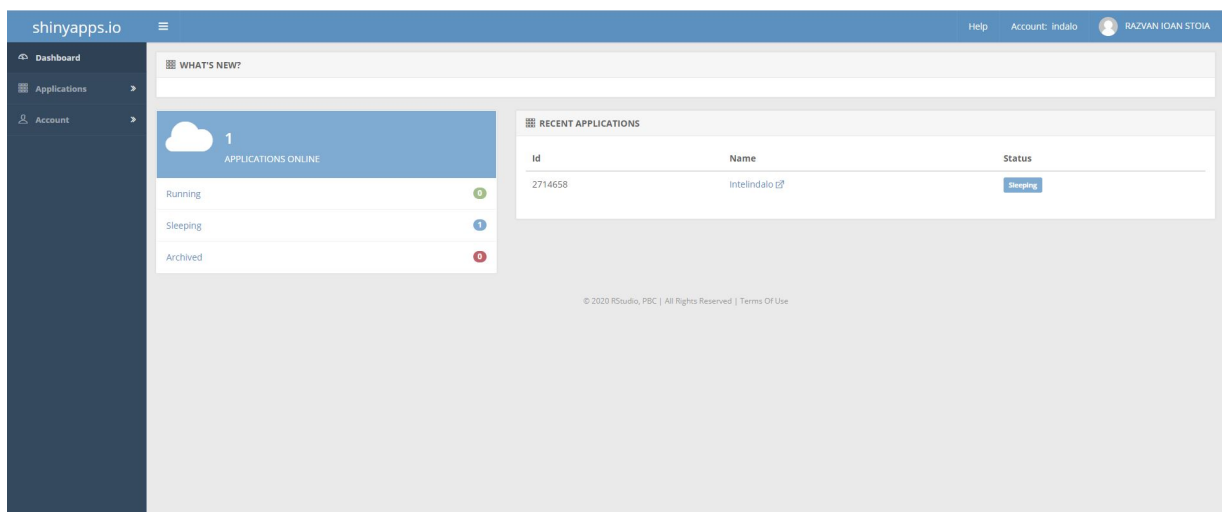
Fuente: autoría propia

Figura C.9: Prototipo inicial para la página de progreso (3).

Apéndice D

Despliegue de la aplicación

El despliegue de la aplicación se ha hecho en el entorno *shinyapps.io*. *Shinyapps* es una plataforma que permite compartir aplicaciones *Shiny* en la web. Ejecuta sus servicios en la nube, en servidores de *RStudio*.



Fuente: autoría propia

Figura D.1: Dashboard de Shinyapps.

Desde *Shinyapps* podemos cambiar el estado de nuestras aplicaciones, analizar métricas interesantes o visualizar los *logs* de la aplicación.

Para hacer el despliegue, hemos utilizado la librería *rsconnect* de *R*. Lo primero que hemos hecho es registrarnos en *Shinyapps*. Una vez registrados, hemos obtenido el *token* desde los ajustes de usuario. Después, desde *R*, hemos configurado la cuenta mediante *rsconnect* [**shinyapps**].

```
1 rsconnect::setAccountInfo(name="<ACCOUNT>", token="<TOKEN>", secret="<SECRET>")
```

Una vez configurada la cuenta, hemos hecho el despliegue de la aplicación.

```
1 deployApp()
```

Es importante mencionar que, para que se instalen todas las librerías que utiliza nuestra aplicación tenemos que cargarlas en el archivo *ui.R*

```
1 library(shiny)
2 library(shinydashboard)
3 library(tidyverse)
4 library(DT)
5 library(caret)
```

Bibliografía

- [1] Agrovademecum. *Buscador de semillas y cultivos*. URL: <https://www.agrovademecum.com/> [Consultado en línea durante el desarrollo del trabajo]. 2019.
- [2] Winston Chang y Joe Cheng. *Package “Shiny”*. URL: <https://cran.r-project.org/web/packages/shiny/shiny.pdf> [Consultado en línea durante el desarrollo del trabajo]. Ago. de 2009.
- [3] *Datos climatológicos en Andalucía*. URL: <https://ws142.juntadeandalucia.es/agriculturaypesca/fit/clima/inicio.do> [Consultado en línea el 17/10/2018].
- [4] Boletín Oficial del Estado. «Reglamento (CE) n^o 790/2000 de la Comisión, de 14 de abril de 2000, por el que se establecen las normas de comercialización de los tomates». En: (). URL: <https://www.boe.es/buscar/doc.php?id=DOUE-L-2000-80643> [Consultado en línea el 07/11/2019].
- [5] June García. «10 Herramientas para crear cuadros de mando y ejemplos». En: *Ambit* (2020). URL: <https://www.ambit-bst.com/blog/10-herramientas-para-crear-cuadros-de-mando-y-ejemplos> [Consultado en línea el 01/07/2020].
- [6] *Histórico de datos de mercado de productos hortícolas*. URL: <https://www.fhalmeria.com/es/historico-mercados.aspx> [Consultado en línea el 10/08/2019].
- [7] Noticia de Hortoinfo. «Almería, 1^a exportadora española de hortalizas». En: *Hortoinfo* (2017). URL: <http://www.hortoinfo.es/index.php/6346-export-hort-prov-301017> [Consultado en línea el 17/10/2018].
- [8] Noticia de Ideal. «Almería cerrará 2018 con las mejores cifras de exportación». En: *Ideal* (2019). URL: <https://www.ideal.es/almeria/almeria/almeria-cerrara-2018-20190124004827-ntvo.html> [Consultado en línea el 01/10/2019].
- [9] iMacros. *Introducing iMacros*. URL: https://wiki.imacros.net/Introducing_iMacros. Oct. de 2018.
- [10] *Red Agroclimática de la Junta de Andalucía. Fuente de datos climatológicos*. URL: https://www.juntadeandalucia.es/agriculturaypesca/ifapa/ria/servlet/FrontController?action=Static&url=coordenadas.jsp&c_provincia=4&c_estacion=1 [Consultado en línea el 17/10/2018].

-
- [11] David Reinsel, John Gantz y John Rydning. *The Digitization of the World, From Edge to Core*. Inf. téc. URL: <https://www.seagate.com/files/www-content/our-story/trends/files/idc-seagate-dataage-whitepaper.pdf> [Consultado en línea el 01/10/2019]. IDC, 2018.
- [12] Bruno Rodrigues. *Modern R with the tidyverse*. URL: https://b-rodrigues.github.io/modern_r/index.html#note-to-the-reader [Consultado en línea el 16/07/2020]. 2020.
- [13] RStudio. *shinydashboard*. URL: <https://rstudio.github.io/shinydashboard/index.html> [Consultado en línea el 30/07/2020]. 2020.
- [14] José del Sagrado Martínez. *Tema 3: Conociendo los Datos*. [Consultado en línea el 25/09/2019].
- [15] Salvador Sánchez, Miguel Ángel Sicilia y Daniel Rodríguez. *Ingeniería del Software, un enfoque desde la guía SWEBOOK*. [Consultado en físico el 17/10/2018]. Garceta, 2019.
- [16] Chilkat Software. *Chilkat ActiveX Component*. URL: <http://www.chilkatsoft.com> [Consultado en línea el 05/08/2019]. 2019.
- [17] W. N. Venables y D. M. Smith. *An Introduction to R*. URL: <https://cran.r-project.org/doc/manuals/r-release/R-intro.pdf>. Jul. de 2019.
- [18] Hadley Wickham. *ggplot2: Elegant Graphics for Data Analysis*. URL: <https://ggplot2-book.org/index.html> [Consultado en línea el 13/05/2020]. 2015.
- [19] Hadley Wickham y Garret Golemund. *R for Data Science*. [Consultado en línea durante el desarrollo del trabajo]. O'Reilly, 2017.
- [20] Wikipedia. *Dashboard (business)*. URL: [https://es.wikipedia.org/wiki/Dashboard_\(business\)](https://es.wikipedia.org/wiki/Dashboard_(business)) [Consultado en línea el 08/07/2020]. 2020.
- [21] Wikipedia. *Minería de datos*. URL: https://es.wikipedia.org/wiki/Mineria_de_datos [Consultado en línea el 18/10/2018]. 2019.
- [22] Wikipedia. *RStudio*. URL: <https://es.wikipedia.org/wiki/RStudio>. Mar. de 2020.
- [23] Wikipedia. *Web scraping*. URL: https://es.wikipedia.org/wiki/Web_scraping [Consultado en línea el 10/08/2019]. 2019.

En el presente documento se exponen los pasos seguidos para la construcción de un cuadro de inteligencia de negocio con R y la librería Shiny. El cuadro de mandos contiene información relativa a productos hortícolas e información climática, de la región geográfica del poniente almeriense. Desde la herramienta, se podrán consultar de forma rápida datos que requieren de un cálculo previo. Vamos a describir todos los pasos, desde la extracción de datos de fuentes heterogéneas hasta el desarrollo de la solución sobre la plataforma R. La herramienta se encuentra desplegada en el entorno shinyapps y se puede acceder de forma pública.

This document contains the steps followed to build a business intelligence chart with R and the Shiny library. The dashboard contains horticultural products and weather information from the western region of Almería. From the tool, data that requires a prior calculation can be quickly consulted. We are going to describe all the steps, starting with data extraction and ending with the development of the solution on the R platform. The tool is deployed in the shinyapps environment and can be publicly accessed.

