

Universidad Internacional de La Rioja

ESIT

**Máster universitario en Ingeniería de Software y
Sistemas Informáticos**

TuPaladar: aplicación para sugerencia de menús basada en predicción de gustos

Trabajo Fin de Máster

Tipo de trabajo: Desarrollo práctico

Presentado por: Sánchez Romero, Raúl

Director/a: Padilla Zea, Natalia

Ciudad: Badajoz

Fecha: 19/09/2019

Resumen

El auge de Internet y de las aplicaciones en el que nos encontramos actualmente en nuestra sociedad nos proporciona un sinnúmero de aplicaciones disponibles para utilizar en nuestros teléfonos móviles. TuPaladar es una aplicación móvil, que integra mecanismos de Machine Learning mediante el algoritmo de Clasificación de Bayes, para la predicción de gustos alimenticios personalizados y sugerencia de platos ideales a los usuarios. Además, incorpora información detallada de contenidos alérgenos en toda la carta de los restaurantes.

Para el desarrollo de la aplicación se ha utilizado la metodología de trabajo RUP y el modelado y diseño de sus componentes a través de UML. El aplicativo tiene una arquitectura basada en lenguaje web, Angular, mediante el uso del Framework Ionic, que incorpora las ventajas tecnológicas del desarrollo de aplicaciones móviles híbridas permitiendo, entre otras características, generar aplicaciones portables para cualquier arquitectura móvil.

Palabras Clave: Machine Learning, Predicción alimenticia, Clasificador de Bayes, Ionic, Ingeniería del Software.

Abstract

Currently we are living in a digital society which uses apps on the internet every tiny detail in our life. Moreover, it does that we have a lot of possibilities to improve our digital uses mostly in mobile phone. TuPaladar is a mobile application, which integrates Mechanisms of Machine Learning through the Bayes Classification algorithm, for the prediction of personalized food tastes and give advice of ideals choice for users in order to have a good restaurant experience. In addition, it incorporates detailed information of allergen contents throughout the menu of the restaurants.

This application has been developed using the RUP working methodology and the modeling and design of its components through UML. TuPaladar has a web language-based architecture, Angular, using the Ionic Framework, which incorporates the technological advantages of developing hybrid mobile applications, like to generate applications portable for any mobile architecture.

Keywords: Machine Learning, Food Prediction, Bayes Classifier, Ionic, Software Engineering.

Índice de contenidos

1. Introducción.....	11
1.1. Justificación	11
1.2. Planteamiento del trabajo	11
1.3. Estructura de la memoria	12
2. Contexto y estado del arte.....	13
2.1. Machine Learning y aprendizaje supervisado	13
2.2. Tecnologías de desarrollo para aplicaciones móviles	19
2.3. Estudio de app de hostelería	23
2.4. Conclusiones del estado del arte	29
3. Objetivos concretos y metodología de trabajo	31
3.1. Objetivo general.....	31
3.2. Objetivos específicos	31
3.3. Metodología del trabajo	31
3.3.1. Metodología de desarrollo software RUP	32
4. Aplicación TuPaladar	35
4.1. Análisis	35
4.1.1. Propósito inicial	35
4.1.2. Alcance	36
4.1.1. Identificación de requisitos	36
4.1.2. Aplicación de Algoritmo de Bayes	42
4.3. Diseño del sistema.....	54
4.3.1. Diseño de la arquitectura.....	54
4.3.2. Diseño de casos de uso	54
4.3.3. Modelo entidad-relación	73

4.3.4. Diseño físico de datos	75
4.3.5. Diseño de clases	82
4.4. Desarrollo	83
4.4.1. Aplicación.....	83
4.4.2. Evaluación del sistema.....	85
5. Conclusiones y trabajo futuro	91
5.1. Conclusiones	91
5.2. Líneas de trabajo futuro	92
6. Bibliografía	94
Anexos	97
Anexo I. Artículo	97
Anexo II. Diagrama de casos de usos.....	103
Anexo III. Modelo relacional de base de datos.....	104
Anexo IV. Diagrama de clases.....	105
Anexo V. Aplicación.....	106
Anexo VI. Pruebas de funcionalidad	114
Anexo VII. Pruebas de usabilidad	126

Índice de tablas

Tabla 1. Aplicaciones nativas y características.....	19
Tabla 2. Comparativa aplicaciones Nativas vs Híbridas.	22
Tabla 3. Comparativa funcional de aplicaciones móviles.....	29
Tabla 4. R.F. 01 - Registro	36
Tabla 5. R.F. 02 - Validación de formulario	37
Tabla 6. R.F. 03 - Login.....	37
Tabla 7. R.F. 04 - Modificación de información usuario	37
Tabla 8. R.F. 05 - Consultar restaurantes.....	38
Tabla 9. R.F. 06 - Sugerencia de plato por restaurante	38
Tabla 10. R.F. 07 - Pedir comida a domicilio	38
Tabla 11. R.F. 08 - Reservar mesa	38
Tabla 12. R.F. 09 - Valoración de restaurante	39
Tabla 13. R.F. 10 - Logout.....	39
Tabla 14. R.F. 11 - Registro de restaurante.....	39
Tabla 15. R.F. 12 - Modificación de restaurante	39
Tabla 16. R.F. 13 - Envío de sugerencias	40
Tabla 17. R.F. 14 - Actualización de la base de conocimiento.....	40
Tabla 18. R.N.F. 01 - Usabilidad	40
Tabla 19. R.N.F. 02 - Rendimiento	41
Tabla 20. R.N.F. 03 - Portabilidad	41
Tabla 21. R.N.F. 04 - Privacidad	41
Tabla 22. R.N.F. 05 - Seguridad.....	41
Tabla 23. R.N.F. 06 - Disponibilidad	42
Tabla 24. Herramientas y tecnologías	54
Tabla 25. Características del actor usuario	55
Tabla 26. Características del actor usuario restaurante.....	55

Tabla 27. Caso de uso 001 - Login.....	56
Tabla 28. Caso de uso 002 - Registrar	57
Tabla 29. Caso de uso 003 - Gestionar información personal	58
Tabla 30. Caso de uso 004 - Gestionar valoraciones platos.....	59
Tabla 31. Caso de uso 005 - Gestionar valoraciones restaurantes.....	60
Tabla 32. Caso de uso 006 - Enviar sugerencias	62
Tabla 33. Caso de uso 007 - Realizar encuestas	63
Tabla 34. Caso de uso 008 - Valorar platos	64
Tabla 35. Caso de uso 009 - Valorar restaurantes	65
Tabla 36. Caso de uso 010 - Reservar restaurante	65
Tabla 37. Caso de uso 011 - Pedir comida.....	66
Tabla 38. Caso de uso 012 - Consultar carta restaurante.....	67
Tabla 39. Caso de uso 013 - Localizar restaurantes.....	68
Tabla 40. Caso de uso 014 - Buscar restaurantes.....	68
Tabla 41. Caso de uso 0015 - Alta plato.....	69
Tabla 42. Caso de uso 0016 - Alta restaurantes.....	70
Tabla 43. Caso de uso 0017 - Gestionar platos.....	71
Tabla 44. Caso de uso 0018 - Gestionar restaurantes	72
Tabla 45. Descripción Campos de auditoría	75
Tabla 46. Descripción tabla Alérgenos	75
Tabla 47. Descripción tabla Edades	75
Tabla 48. Descripción tabla Grupos ingredientes	76
Tabla 49. Descripción tabla Grupos usuarios	76
Tabla 50. Descripción tabla Habitat.....	76
Tabla 51. Descripción tabla Sexo	76
Tabla 52. Descripción tabla Alergenos_x_plato.....	77
Tabla 53. Descripción tabla Alergias_x_usuario	77
Tabla 54. Descripción tabla Clasificaciones_base_usuarios.....	77

Tabla 55. Descripción tabla Factores_base_ingrediente_x_edad.....	77
Tabla 56. Descripción tabla Factores_base_ingrediente_x_habitat	78
Tabla 57. Descripción tabla Factores_base_ingrediente_x_sexo	78
Tabla 58. Descripción tabla Ingredientes.....	78
Tabla 59. Descripción tabla Ingredientes_x_platos	78
Tabla 60. Descripción tabla Platos_x_restaurantes	79
Tabla 61. Descripción tabla Puntuaciones_base_usuarios.....	79
Tabla 62. Descripción tabla Puntuaciones_usuarios_x_ingredientes	79
Tabla 63. Descripción tabla Puntuaciones_usuarios_x_plato	79
Tabla 64. Descripción tabla Puntuaciones_usuarios_x_restaurante.....	80
Tabla 65. Descripción tabla Restaurantes	80
Tabla 66. Descripción tabla Sugerencias_x_usuario	80
Tabla 67. Descripción tabla Ubicaciones.....	80
Tabla 68. Descripción tabla Usuarios	81
Tabla 69. Descripción tabla Valoraciones_usuarios_x_platos	81
Tabla 70. Descripción tabla Valoraciones_usuarios_x_restaurantes	82
Tabla 71. Correspondencia de Tipos BBDD - Lógica de Negocio.....	82
Tabla 72. Plantilla prueba funcional.....	85
Tabla 73. Prueba funcional 001 - Login.....	114
Tabla 74. Prueba funcional 002 - Registrar	114
Tabla 75. Prueba funcional 003 - Gestionar información personal.....	115
Tabla 76. Prueba funcional 004 - Gestionar valoraciones platos	116
Tabla 77. Prueba funcional 005 - Gestionar valoraciones restaurantes.....	117
Tabla 78. Prueba funcional 006 - Enviar sugerencias.....	117
Tabla 79. Prueba funcional 007 - Realizar encuesta	118
Tabla 80. Prueba funcional 008 - Valorar plato.....	118
Tabla 81. Prueba funcional 009 - Valorar restaurantes.....	119
Tabla 82. Prueba funcional 010 - Reservar restaurante	120

Tabla 83. Prueba funcional 011 - Pedir comida	120
Tabla 84. Prueba funcional 012 - Consultar carta restaurante	121
Tabla 85. Prueba funcional 013 - Localizar restaurantes	121
Tabla 86. Prueba funcional 014 - Buscar restaurantes	122
Tabla 87. Prueba funcional 015 - Alta plato	122
Tabla 88. Prueba funcional 016 - Alta restaurantes	123
Tabla 89. Prueba funcional 017 - Gestionar platos	123
Tabla 90. Prueba funcional 018 - Gestionar restaurantes	124

Índice de figuras

Figura 1. Ejemplo árbol de decisión [6].....	15
Figura 2. Ecuación del Teorema de Bayes.....	18
Figura 3. Ecuación del Teorema de Bayes hipótesis máxima a posteriori	18
Figura 4. Ecuación del Teorema de Bayes hipótesis de máxima probabilidad.....	18
Figura 5. Icono Ionic.....	20
Figura 6. Icono Just Eat	23
Figura 7. Aplicación Just Eat.....	24
Figura 8. Icono El Tenedor.....	24
Figura 9. Aplicación El Tenedor.....	25
Figura 10. Icono Deliveroo	25
Figura 11. Aplicación Deliveroo.....	26
Figura 12. Icono Yelp	26
Figura 13. Aplicación Yelp.....	27
Figura 14. Icono Glovo	27
Figura 15. Aplicación Glovo.....	28
Figura 16. Representación de iteraciones y fases metodología RUP [14]	34
Figura 17. 14 alérgenos a especificar en la carta [19]	44
Figura 18. Frecuencia de consumo de carne según sexo, edad y habitat [20].....	45
Figura 19. Frecuencia de consumo de pescado según sexo, edad y habitat [21]	46
Figura 20. Consumo de los tipos de carne por sexo [20].....	47
Figura 21. Consumo de tipos de carne por edad [20]	47
Figura 22. Consumo de tipos de carne por habitat [20]	48
Figura 23. Ejemplo hipótesis de máxima probabilidad.....	51
Figura 24. Ecuaciones actualización base de conocimiento	52
Figura 25. Ejemplo actualización base de conocimiento vacuno	53
Figura 26. Ejemplo 2 actualización base de conocimiento vacuno	53

Figura 27. Caso de uso - Login y Registrar	56
Figura 28. Caso de uso - Gestionar información personal	58
Figura 29. Caso de uso 006 - Enviar sugerencias	62
Figura 30. Caso de uso 007 - Realizar encuestas	63
Figura 31. Caso de uso del actor Usuario con los Restaurantes.	64
Figura 32. Caso de uso Alta de restaurante	69
Figura 33. Caso de uso Gestión de restaurante	71
Figura 34. E/R Tipos de usuario	73
Figura 35. E/R Usuario Restaurante.....	74
Figura 36. E/R Usuario Particular	74
Figura 37. Diagrama de casos de uso	103
Figura 38. Modelo relacional de base de datos	104
Figura 39. Diagrama de clases.....	105
Figura 40. Pantalla login.....	106
Figura 41. Pantalla registro y modificación de datos usuario	107
Figura 42. Pantalla principal y menú	108
Figura 43. Pantalla listado de restaurantes	108
Figura 44. Pantalla detalle restaurante	109
Figura 45. Pantalla listado de platos.....	109
Figura 46. Pantalla reservar mesa.....	110
Figura 47. Pantalla valorar restaurante.....	110
Figura 48. Pantalla valorar plato.....	111
Figura 49. Pantalla mis valoraciones.....	111
Figura 50. Pantalla inicio usuario restaurante y mis restaurantes	112
Figura 51. Pantalla alta / detalle restaurante	112
Figura 52. Pantalla alta / detalle del plato	113
Figura 53. Pantalla danos tu opinión	113

1. Introducción

Hoy en día la estrategia comercial para muchas organizaciones pasa por la implementación de aplicaciones corporativas para dispositivos móviles, como teléfonos, relojes inteligentes, coches, etc., ya sea para ofrecer soluciones en esos dispositivos, promovidos por el marketing y de esta manera aumentando las opciones para darse a conocer o para permitir una gestión más accesible y fácil para sus clientes.

Además, según afirma *Gartner* desde el estudio *Market Guide for Mobile Application Testing Services*, “El 70% de las interacciones que se llevarán a cabo desde 2022 serán desde un dispositivo móvil” [1]. En el caso de España, “El móvil es el dispositivo más utilizado para acceder a Internet y usado ya por el 94,6%” [2].

En este Trabajo Fin de Master se desarrolla una aplicación híbrida móvil, *TuPaladar*, enfocada en ofrecer servicios de comida, añadiendo una funcionalidad innovadora en cuanto al sector para la sugerencia personalizada de comidas.

1.1. Justificación

La idea para llevar a cabo este Trabajo de Fin de Máster surgió de la necesidad de información de contenidos de alérgenos, en los platos que se ofrecen en las aplicaciones móviles y web, que se encargan de ofrecer servicios de comida a domicilio o reserva de mesa en restaurantes. Esto es un verdadero problema para aquellas personas que tienen algún tipo de intolerancia que, por miedo a consumir alguna comida que contenga algún ingrediente perjudicial para su salud, opta por no pedir comida a domicilio.

La segunda idea está basada en la necesidad de sugerir comida ideal para cada comensal en función de sus preferencias. Si pensamos en servicios de streaming de contenido audiovisual, como Netflix, podemos observar que, sugiere que películas o series pueden ser interesantes de visualizar por el usuario en base a sus gustos o preferencias previas dentro de la aplicación.

1.2. Planteamiento del trabajo

Este trabajo va a dar solución a estos problemas mediante el desarrollo de una aplicación móvil encargada de ofrecer servicios de comida. La aplicación va a estar provista de un procedimiento de predicción que permita sugerir platos ideales a las personas en base a sus preferencias y gustos, cuya predicción no contenga contenidos intolerantes para ellos de manera individualizada.

1.3. Estructura de la memoria

El resto de la memoria se estructura como sigue:

En el capítulo 2 se sitúa al lector en el contexto y estado del arte que se ha tomado como base para la realización de este trabajo. Para ello, se realiza un estudio sobre las diferentes alternativas móviles disponibles actualmente en el mercado de la hostelería, un estudio, investigación y propuesta de algoritmos de inteligencia artificial y mejora continua, y un estudio de las diferentes alternativas tecnológicas para el desarrollo de aplicaciones móviles.

En el capítulo 3 se definen los objetivos e hitos que se van a alcanzar con este trabajo y la metodología de trabajo que se ha llevado a cabo para su realización.

En el capítulo 4 se aporta todo el contenido desarrollado de este trabajo, detallándose las fases del ciclo de vida de software de la aplicación móvil implementada, y concretando la planificación del proyecto, análisis, diseño, desarrollo y evaluación bajo la metodología expuesta en el capítulo 3.

En el capítulo 5 se describen las conclusiones obtenidas mediante el desarrollo y finalización de los objetivos, se describen los problemas encontrados, las decisiones tomadas y las posibles líneas de trabajos futuras que se pueden abordar para complementar este proyecto.

En el capítulo 6 se listan las diferentes bibliografías que se han tomado como sustento de los argumentos y justificaciones desarrollados en este trabajo.

Por último, en el apartado de anexos, se encuentra todo el contenido adicional, ilustraciones y tablas, así como un resumen de esta memoria.

2. Contexto y estado del arte

2.1. Machine Learning y aprendizaje supervisado

El *Machine Learning* (en español *Aprendizaje automático*) es una disciplina dentro de la Ingeniería de la Computación Informática y un subconjunto de conocimiento dentro de la rama de Inteligencia Artificial basados en la creación de sistemas software capaces de llevar a cabo un proceso de aprendizaje de manera autónoma. Para ello, el sistema o máquina *aprende* mediante el uso de un algoritmo que procesa y cataloga una serie o conjunto de datos de entrada con el objetivo de predecir comportamiento o situaciones futuras. [3]

En *Machine Learning* se pueden definir 3 conceptos principales [3] [4]:

1. **Base de conocimiento.** Es el conjunto de información que contiene el sistema y que es utilizado en el proceso de predicción del algoritmo. Se distinguen dos grupos, **conjunto de entrenamiento** que se utilizan para identificar los parámetros de clasificación y **conjunto de pruebas** usado para el cálculo de la estimación de error con el objetivo que sea pequeño.
2. **Clasificador.** Es la relación entre los datos almacenados en la base de conocimiento, que son usados y procesados para la predicción. Esta relación implica realizar un proceso de abstracción mediante la combinación de los datos para su interpretación en base al objetivo concreto del estudio, es decir, el clasificador es el procedimiento usado para relacionar la experiencia que tiene el algoritmo, base de conocimiento, con la nueva experiencia, no catalogada aún en la base de conocimiento, para predecir el resultado objetivo final.
3. **Aprendizaje (en inglés *learner*).** Es el procedimiento usado para la construcción de un clasificador partiendo de la base de conocimiento. Además, este procedimiento se complementa con un mecanismo de evaluación y retroalimentación para la medición del nuevo conocimiento aprendido que puede conllevar en ajustes del clasificador. También se puede definir este concepto como “*Cambios adaptivos en el sistema para hacer la misma tarea de la misma población de una manera más eficiente y efectiva la próxima vez*”.

Por tanto, para realizar el proceso de *Machine Learning*, el sistema recibe una nueva experiencia o datos, mediante el algoritmo procesa y clasifica esa experiencia con las procesadas con anterioridad y que forman parte de su base de conocimiento, para predecir el

estado final de esa experiencia concreta. Además, la experiencia se introduce como un nuevo aprendizaje para retroalimentar la base de conocimiento del sistema.

Tipos de algoritmos de Machine Learning

Existen tres categorías principales de algoritmos de aprendizaje automático según su propósito de uso [3] [4].

- Algoritmos de **aprendizaje supervisado**. Este tipo de algoritmos *predictivos* se caracterizan por tener un procedimiento claro sobre qué necesitan aprender y cómo hacerlo. Trabajan con datos *etiquetados*, donde dadas las variables de entrada, intentan encontrar una función para asignar la más adecuada etiqueta de salida. Estos algoritmos se basan en un histórico de datos que utilizan para su entrenamiento y aprendizaje, y, por tanto, para la ponderación predictiva de la etiqueta de salida. Los algoritmos de aprendizaje supervisado se suelen utilizar principalmente para problemas de *Clasificación*, cuyo objetivo de salida es la predicción de una etiqueta/clase de salida de manera categórica, y problemas de *Regresión* donde la predicción de salida es del tipo numérico.
- Algoritmos de **aprendizaje no supervisado**. A diferencia de los supervisados, estos algoritmos no se basan en un modelo predictivo sino en un modelo *descriptivo*, donde al no haber una única característica, no hay un objetivo concreto para aprender. Esto implica que no tienen una base de datos previamente etiquetados, solo conocen los datos de entrada, pero no existe una catalogación de salida correspondiente a la entrada. Suelen utilizarse en problemas donde no hay un objetivo definido, basándose en la exploración y la agrupación de los datos.
- Algoritmos de **aprendizaje por refuerzo**. Estos algoritmos no tienen asociada una tarea de aprendizaje específica, sino que más bien se centran en aprender el cómo aprender de la manera más eficiente. Dicho con otras palabras, estos algoritmos se basan en mejorar la respuesta del modelo usando un proceso de retroalimentación. El sistema aprende en base ensayo-error.

El contexto de *Machine Learning* de este trabajo se centra en el aprendizaje supervisado, ya que el problema a resolver se basa en la *predicción* de comidas a *personas* en función de sus *características*. Es decir, estamos ante un problema de *Clasificación* donde el objetivo es la predicción de la salida de manera categórica en base al conocimiento previo. A continuación, se detallan dos algoritmos de aprendizaje supervisado para la resolución de problemas clasificatorios.

Árboles de decisión

Los árboles de decisión son métodos usados para el aprendizaje supervisado, principalmente diseñados para la solución de problemas de clasificación. Están basados en una representación gráfica de las posibles soluciones, en forma de árbol, que se componen mediante decisiones basadas en condiciones previamente establecidas. Esta representación permite la legibilidad del resultado del algoritmo, donde el proceso de clasificación comienza en el nodo raíz del árbol, baja por la rama correspondiente al valor del atributo que se procesa, y se toma la decisión o rama de salida. El proceso se repite hasta que se llega a algún nodo hoja. [5]

En Figura 1 se puede visualizar un ejemplo de árbol de decisión, donde, en función de las características del tiempo: soleado, nublado o lluvia; se puede predecir la humedad o el viento.

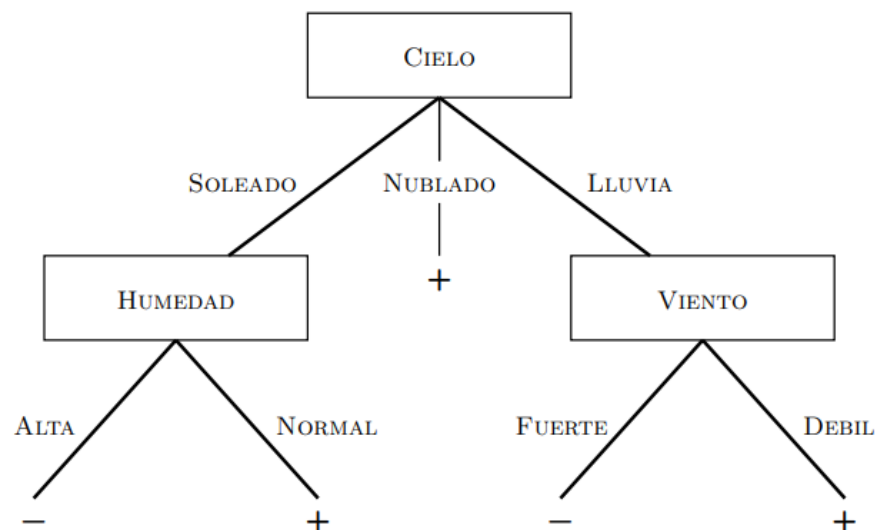


Figura 1. Ejemplo árbol de decisión [6]

En general, los árboles de decisión representan un conjunto de restricciones correspondientes con los valores de los atributos, de las instancias o valores de entrada. Cada camino desde el nodo raíz del árbol hasta un nodo hoja corresponde con una toma de decisión del conjunto de restricciones. [5]

Hay diferentes métodos basados en aprendizaje de árboles de decisión, aunque generalmente son aplicables a problemas con las siguientes características: [5]

- Las instancias son representadas mediante pares atributo-valor, es decir, son descritas por un conjunto fijo de atributos. La situación más fácil es cuando cada atributo adquiere un número pequeño de valores. Como en el ejemplo de la Figura 1, el cielo puede ser: soleado, nublado o lluvia.

- La función de salida de cada restricción tiene valores de salida discretos. En el ejemplo de la Figura 1, cada nodo tiene una condición booleana, si o no. Aunque los árboles de decisión son aplicables a funciones de aprendizaje con más de dos posibles valores de salida, no es común su uso.
- Los datos de entrenamiento pueden contener errores, aunque los árboles de decisión se caracterizan por ser robustos a errores, ya sean errores en la clasificación de los datos ejemplos o errores en los valores de los atributos.
- Son aplicables en las situaciones en las que los datos de entrenamiento pueden no contener la suficiente información para la toma de decisiones, o incluso en situaciones donde los datos de entrenamiento son desconocidos.

Algoritmo de Naive Bayes

El algoritmo o *clasificador de Naive Bayes* se trata de una técnica de clasificación y de predicción supervisada que se caracteriza por su simplicidad y rapidez. Además, el método de Bayes es considerado uno de los más importantes clasificadores, debido a que proporciona una perspectiva intuitiva y útil para la comprensión de algoritmos que no manipulan explícitamente las probabilidades. Es un algoritmo que basa su conocimiento en un modelo para la predicción de probabilidades de posibles resultados. [7]

El algoritmo de aprendizaje de Bayes tiene las siguientes características [5]:

- Cada muestra o ejemplo, que participa en el entrenamiento del algoritmo, puede aumentar o disminuir incrementalmente la probabilidad de estimación de que la hipótesis resultado sea correcta. Esta característica aporta flexibilidad en contraposición con los algoritmos de aprendizaje que descartan la hipótesis es inconsistente, en comparación con los demás ejemplos de la base de conocimiento del algoritmo.
- El conocimiento previo que sirve como experiencia en el algoritmo se combina con los datos de la observación para el cálculo de la hipótesis resultante. Además, este método realiza predicciones probabilísticas de la hipótesis.
- Las nuevas muestras observadas se pueden clasificar combinando las probabilidades de las hipótesis.
- Es un método que puede proporcionar un estándar para la toma de decisiones, aunque la aplicación del método sea computacionalmente imposible. Este algoritmo requiere un conocimiento inicial de muchas probabilidades. Un ejemplo con dificultad

para la aplicación del algoritmo de Bayes es que no se conozcan de antemano las probabilidades, y en vez de ser calculadas para enriquecer el conocimiento inicial sean estimadas.

Una vez planteados dos algoritmos de aprendizaje supervisado, el estudio de este trabajo se centra en el algoritmo de *Naive Bayes* para su aplicación al problema de predicción a resolver.

Teorema de Bayes [5]

Como se ha comentado hasta ahora, el objetivo de *machine learning* es determinar la hipótesis más probable dado un conjunto de datos previamente observados y que han servido como entrenamiento al proceso.

Supongamos una hipótesis h que pertenece al conjunto de hipótesis H y un conjunto de entrenamiento D , vamos a calcular la hipótesis h más probable, dados los datos de entrenamiento D , junto con cualquier otro conocimiento inicial previo sobre las probabilidades de H .

El Teorema de Bayes proporciona un método directo para el cálculo de las probabilidades de una hipótesis basada en su probabilidad previa, las probabilidades de la observancia de varios datos dada la hipótesis y los datos observados en sí. La definición del Teorema de Bayes es la siguiente:

- Se define $P(h)$ como la probabilidad inicial o probabilidad previa de la hipótesis h antes de haber observado los datos de entrenamiento. En el caso de no tener el conocimiento previo de $P(h)$ el teorema posibilita que simplemente se asigne una probabilidad equitativa entre cada hipótesis candidata.
- Se define $P(D)$ como la probabilidad previa de que se observen los datos de entrenamiento o, dicho de otra manera, la probabilidad de que D no tenga conocimientos sobre la hipótesis.
- Se define $P(D|h)$ como la probabilidad de observar los datos de D dado que la hipótesis h exista.
- Se define $P(h|D)$ como la probabilidad que tiene la hipótesis h dados los datos de entrenamiento observados D . También es denominada como probabilidad posterior de la hipótesis h , porque muestra la confianza de la hipótesis h después de haber observado los datos de entrenamiento D . Cabe destacar que la probabilidad $P(h|D)$ muestra la influencia de los datos de entrenamiento D , mientras que, de manera

contraria, la probabilidad $P(h)$ es independiente del conjunto de entrenamiento D dentro del Teorema de Bayes.

La ecuación del Teorema de Bayes es la siguiente:

$$P(h|D) = \frac{P(D|h)P(h)}{P(D)}$$

Figura 2. Ecuación del Teorema de Bayes

Como se observa de la ecuación de Bayes, Figura 2, la probabilidad de $P(h|D)$ aumenta cuanto mayor sean las probabilidades $P(D|h)$ y $P(h)$, al igual que disminuye cuanto mayor sea la probabilidad de $P(D)$, ya que cuanto más probable es que no se observe la hipótesis h en el conjunto de entrenamiento D , menos evidencia hay de h en D .

Existen numerosos escenarios de aprendizaje donde poder aplicar el Teorema de Bayes, pero uno de los más comunes, es dado un conjunto de hipótesis candidatas H , estar interesado en encontrar la hipótesis más probable h que pertenece a H dados los datos observados D . Si hubiera diferentes hipótesis, el resultado sería la hipótesis máxima a posteriori. Cualquier hipótesis máxima probable o también denominada *MAP* se puede determinar con el Teorema de Bayes mediante la siguiente ecuación, Figura 3:

$$\begin{aligned} h_{MAP} &= \operatorname{argmax}_{h \in H} P(h|D) \\ &= \operatorname{argmax}_{h \in H} \frac{P(D|h)P(h)}{P(D)} \\ &= \operatorname{argmax}_{h \in H} P(D|h)P(h) \end{aligned}$$

Figura 3. Ecuación del Teorema de Bayes hipótesis máxima a posteriori

Se observa que el divisor $P(D)$ ha desaparecido de la ecuación, ya que es una constante que no tiene dependencia con la hipótesis.

Por último, la ecuación del Teorema de Bayes se puede simplificar aún más, Figura 4, bajo el supuesto de asumir que cada hipótesis de H es igualmente probable a priori. Es decir, dado h_i y h_j que pertenecen al conjunto de H , suponemos que $P(h_i) = P(h_j)$, ha esta hipótesis también se la denomina, hipótesis de máxima probabilidad h_{ML} :

$$h_{ML} = \operatorname{argmax}_{h \in H} P(D|h)$$

Figura 4. Ecuación del Teorema de Bayes hipótesis de máxima probabilidad

2.2. Tecnologías de desarrollo para aplicaciones móviles

Para el desarrollo e implementación de app móviles existen dos soluciones principales: implementación mediante lenguaje nativo en función de la plataforma, las dos principales son iOS y Android; o bien, mediante lenguaje híbrido basado en tecnología web.

La elección depende de numerosos factores como pueden ser el presupuesto, destinatarios de la aplicación, plazos, funcionalidad a implementar, etc. [8]

Aplicaciones nativas

Son, en esencia, archivos binarios ejecutables que se descargan directamente en el dispositivo móvil y el usuario puede realizar el proceso de instalación. Este proceso se suele realizar mediante la búsqueda y descarga de aplicación en una tienda, como App Store de Apple o Marketplace/Play Store en Android, por ejemplo. Una vez instalada la aplicación en el terminal, esta se ejecuta como un servicio más, pudiendo acceder a todas las APIs que el sistema operativo ponga a su disposición. [8] [9]

Para crear una aplicación nativa es necesario implementar la aplicación mediante código fuente específico para esa plataforma. Una vez implementado, el código es compilado y generado un ejecutable que contendrá la aplicación empaquetada y lista para su instalación. Al conjunto de herramientas que incluyen el acceso a recursos y archivos suministrado para realizar el desarrollo se denomina SDK (*Software Development Kit*). [9]

En la siguiente Tabla 1 se pueden visualizar las principales plataformas o sistemas operativos móviles y sus características.

Tabla 1. Aplicaciones nativas y características



	Apple iOS	Android	Blackberry OS	Windows Phone
Lenguajes	Objective-C, C, C++	Java (algunos C, C++)	Java	C#, VB.NET, etc.
Herramientas	Xcode	Android SDK	BB Java Eclipse Plug-in	Visual Studio, Windows Phone
Formato	.app	.apk	.cod	.xap
Tiendas	Apple App Store	Google Play	Blackberry App World	Windows Phone Marketplace

Tabla extraída de [8]

Ventaja de usar aplicaciones nativas

Las ventajas de desarrollar aplicaciones nativas son las siguientes: [8]

- Rendimiento: El código nativo es más rápido que JavaScript y HTML.
- Gran librería de recursos: Usar el SDK nativo permite usar las últimas características diseñadas para esas plataformas.
- No depende de librerías externas o de terceros y, por tanto, la actualización de estas a las últimas características nativas.

Inconvenientes de usar aplicaciones nativas

Crear una aplicación móvil implica crearla como mínimo en iOS y en Android (por su volumen de mercado) y su implicación o consecuencia radica en lo siguiente: [8]

- Los tiempos de desarrollo son elevados, ya que hay que desarrollar de manera independiente una aplicación por cada sistema operativo nativo.
- Se requiere de desarrolladores especializados en ambas tecnologías, código nativo para desarrollar aplicaciones móviles para diferentes sistemas, Swift para desarrollar la aplicación en iOS y Java para desarrollarla en Android.
- Tanto el mayor tiempo como la necesidad de personal especializado implica una menor agilidad a la hora de lanzar actualizaciones.
- Tanto el mayor tiempo como el mayor volumen de desarrollo, así como la necesidad de desarrolladores especializados implica mayor coste total en el desarrollo.

Aplicaciones Híbridas: Ionic



Figura 5. Icono Ionic

Son, en esencia, aplicaciones nativas que pueden ser publicadas en las respectivas tiendas o app store de los diferentes sistemas. Además, en la mayoría de los casos, las aplicaciones nativas como Ionic, son irreconocibles de una aplicación nativa y los usuarios no

suelen darse cuenta de la diferencia. [10]

Su mayor potencial es que apuesta por lenguaje de tecnologías estandarizadas web, que son caracterizadas por ser multiplataforma y que están basado en mantener las mejores prácticas, características y patrones de desarrollo para cada plataforma. [9]

La base del desarrollo de aplicaciones mediante Ionic es el uso del componente llamado Web-View, el cual nos permite la ejecución de nuestra aplicación basándose en lenguajes web HTML, CSS y JavaScript. Este componente está desarrollado en C#, Java y Objective-C que permite a Ionic y a Cordova usarlo de manera interna para la generación del empaquetado y versión de la aplicación para cada sistema. De esta manera automatizada, permite que los desarrolladores se centren en la creación de la aplicación sin necesidad de controlar esta gestión interna. [9] [10]

Además, este framework está diseñado para que las aplicaciones puedan acceder, a los mismos recursos y a las mismas librerías, como ocurre en el lenguaje nativo. Tienen un rendimiento acelerado por hardware al igual que el usado en el SDK nativo permitiendo que las animaciones y las transiciones sean fluidas y llamativas con la diferencia de que en realidad se está ejecutando código en HTML, CSS y JavaScript. [9] [10]

Ventajas de las aplicaciones híbridas

Seleccionar aplicaciones híbridas para el desarrollo de aplicaciones tiene los siguientes beneficios. [8] [9]

- Mejora los tiempos de desarrollo, ya que la aplicación va a ser desarrollada mediante una única base de código y por tanto disminuir los tiempos de desarrollo en torno al 200-300%.
- Este tipo de tecnología y framework tienen un mejor soporte ya que para la solución de dudas o problemas que puedan surgir, no es necesario de personal especializado en código nativo.
- Solo es necesario conocer una única manera para el desarrollo de aplicaciones ya que se pueden ejecutar en diferentes sistemas y entornos donde se pueda ejecutar tecnología web, ya sea como aplicaciones de escritorio, aplicaciones para móviles, PWA, coches inteligentes, etc. Además, según las previsiones de *Gartner* para el 2022, el 70% de las interacciones en la web serán a través de dispositivos móviles.
- A nivel de recursos, encontrar desarrolladores conocedores de tecnologías web es más fácil que encontrar o enseñar desarrolladores especializados en tecnología más específicas nativas, y como consecuencia fomenta en menor coste.
- Las aplicaciones híbridas permiten tener una mejor interfaz de usuario entre las diferentes plataformas, ya que el desarrollo de una única aplicación permite una mayor consistencia hacia la experiencia del usuario en diferentes entornos.

Inconvenientes de Aplicaciones Híbridas

Sin embargo, la utilización de arquitectura nativa para el desarrollo de aplicaciones móviles tiene una serie de peculiaridades negativas a tener en cuenta: [8] [9]



- Mayor consumo de recursos en comparación con las aplicaciones nativas debido al uso del componente WebView, aunque este inconveniente cada vez tiene menor importancia en comparación con el incremento de potencia de los dispositivos cada año.
- Debido al mayor consumo de recursos, no es buena opción desarrollar aplicaciones híbridas que requieran de unas elevadas exigencias de procesamiento o gráficos, como por ejemplo juegos 3D.
- Aunque las aplicaciones nativas pueden acceder a casi todas las características disponibles en las nativas para los dispositivos, el uso de plugins de terceros, añade una capa de complejidad al desarrollo de aplicaciones.
- La elección de aplicaciones híbridas implica confiar y utilizar un framework de desarrollo. En el caso de Ionic, es un framework que está comprometido a estar actualizado conforme a los cambios que se realicen en las nuevas funcionalidades de las versiones nativas.







Comparativa entre aplicaciones nativas y aplicaciones híbridas

Como reflexión, podemos decir que actualmente conocemos las diferentes arquitecturas y plataformas disponibles, pero pensando en el futuro, ese número y diversidad de soluciones puede ser más amplio y esto puede dar un punto a favor en la utilización de la tecnología web, ya que es la más utilizada para aplicaciones, que requieren ese punto de portabilidad e independencia de la plataforma.

En la siguiente Tabla 2 podemos ver un resumen comparativo entre el desarrollo de aplicaciones nativas e híbridas.

Tabla 2. Comparativa aplicaciones Nativas vs Híbridas.

<i>Característica</i>	<i>Aplicación Nativa</i>	<i>Aplicación Híbrida</i>
<i>Tecnología necesaria</i>	Objective-C, IOS SDK Java, Android SDK C#, Windows SDK	HTML, CSS, JS + framework
<i>Distribución</i>		

Tiempos desarrollo		
Costes	€ € €	€
Rendimiento		
Funcionalidad disponible		

(Elaboración propia)

2.3. Estudio de app de hostelería

La tendencia tecnológica en la que vivimos actualmente está basada en un dispositivo de suma importancia para todas las personas: el teléfono móvil. Acciones básicas como llamar por teléfono o mandar mensajes hasta acciones más complejas, como la búsqueda de información en Internet, uso de redes sociales o gestión de correo electrónico, son ejemplos cotidianos que se pueden realizar con estos terminales. También para salir a comer a algún restaurante o pedir la cena a domicilio. Cuestiones que nos surgen en nuestro día a día como “¿Dónde vamos a comer?”, “¿Qué restaurantes tenemos cerca?” o “¿Te apetece pedir comida china?”, tiene una fácil solución: realizar una búsqueda con nuestro teléfono móvil ya sea en un buscador de Internet o mediante alguna app móvil.

En este ámbito, el hostelero, podemos encontrar numerosas aplicaciones para móviles que tiene por objetivo facilitarnos información de restaurante, bares y establecimientos de comidas para ofrecernos servicios, como la reserva de mesa en un restaurante o pedir comida a domicilio. Algunas apps móviles que lideran este sector y que se engloban dentro de este mercado son las siguientes.

Just Eat



Figura 6. Icono Just Eat

Es una plataforma online (servicio web¹ y aplicación móvil), Figura 6, utilizada para expandir las ofertas gastronómica de los restaurantes dando a conocer su negocio a través de sus clientes. Su principal objetivo es hacer que los usuarios puedan elegir entre una amplia variedad de restaurantes desde cualquier lugar. Fundada en 2001 en Dinamarca, se ha convertido en una de las compañías que lidera el

¹ <https://just-eat.es/>

mercado de comida a domicilio a nivel internacional estando presente en 13 países [11].

Just Eat basa su estrategia e idea de negocio en dos pilares. El primero, potenciando las características de los restaurantes y dando a conocer sus servicios y, en segundo lugar, dotándoles de un ecosistema eficiente para la realización de pedidos.

En la Figura 7 se muestra el aspecto de la aplicación Just Eat para iOS.

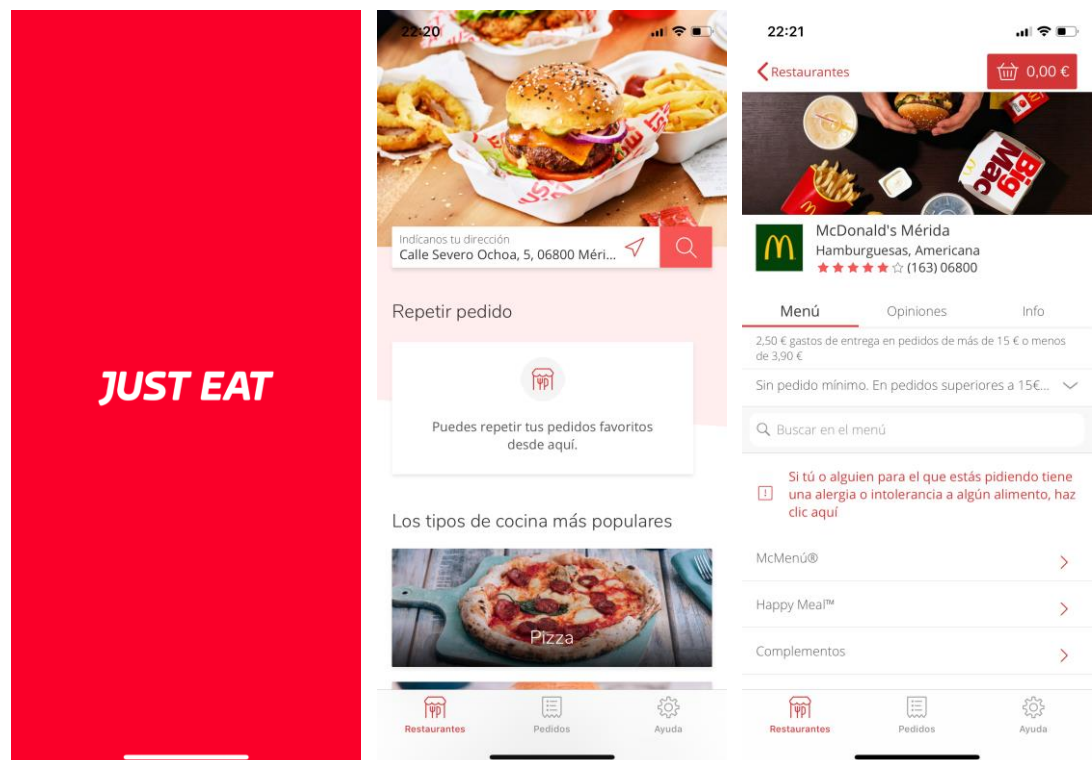


Figura 7. Aplicación Just Eat

El Tenedor



Figura 8. Icono El Tenedor

TripAdvisor.

Es una de las aplicaciones móviles líder en reservas de restaurantes ofreciendo sus servicios en un gran número de países de Europa, Australia y Sudamérica. Es un servicio perteneciente a TripAdvisor, y es una de las soluciones más completas para realizar búsquedas de restaurantes casi en cualquier ciudad. Para los restaurantes es una grandísima manera de ofrecer su propuesta degustativa de manera muy completa y apoyada bajo la red de opiniones y reseñas de

Además, El Tenedor², Figura 8, destaca por su programa de descuentos mediante acumulación de “Yums”, facilidad para la realización de reservas al instante y por información, asistencia y atención al cliente de las reservas realizadas.

En la Figura 9 se muestra el aspecto de la aplicación El Tenedor para iOS.

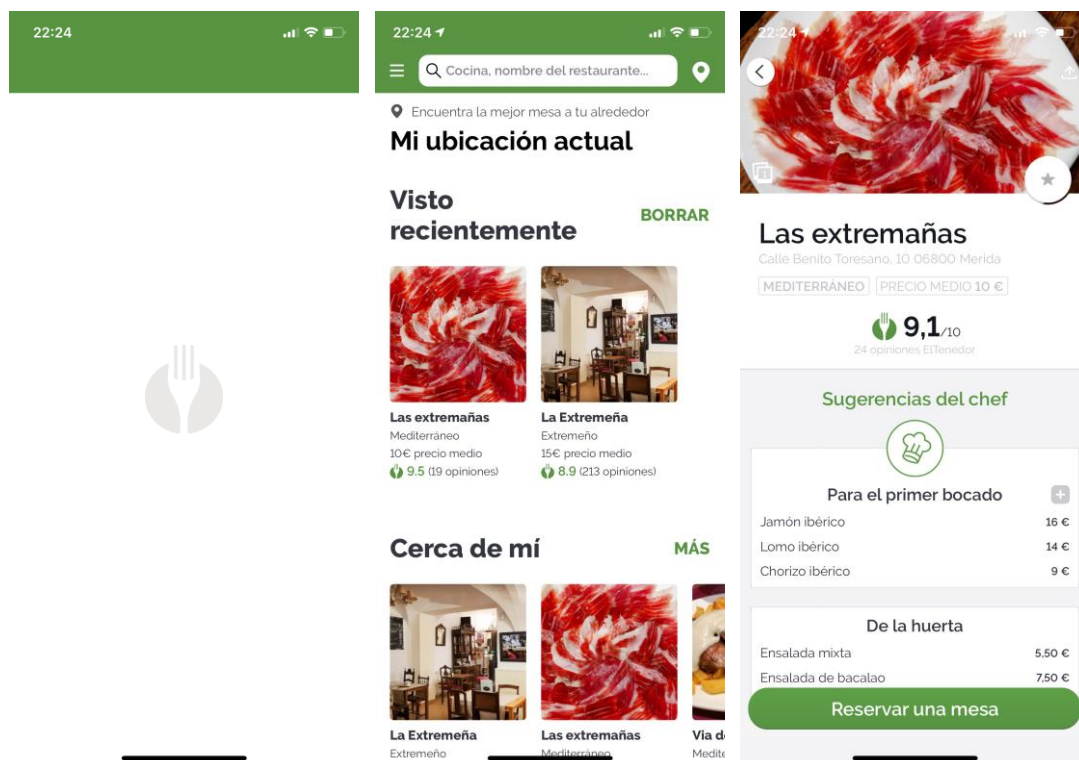


Figura 9. Aplicación El Tenedor

Deliveroo



Figura 10. Icono Deliveroo

Es una aplicación móvil que ofrece un servicio enfocado al reparto de comida a domicilio. Aunque parece propuesta sencilla, su fundador observó que la mayoría de los restaurantes no ofrecía servicio a domicilio y de ahí nace Deliveroo³, Figura 10, una solución centralizada para dotar a los restaurantes de servicio a domicilio. Actualmente, ofrece a los restaurantes asociados un modelo de reparto de comida basado en una entrega rápida y transparente, con la utilización de una

aplicación móvil.

² <https://www.eltenedor.es/>

³ <https://deliveroo.es/>

Tiene cierta similitud con Glovo en cuanto a sus repartidores o “riders” mediante la flexibilidad y libertad para realizar las jornadas de trabajo.

En la Figura 11 se muestra el aspecto de la aplicación Deliveroo para iOS.

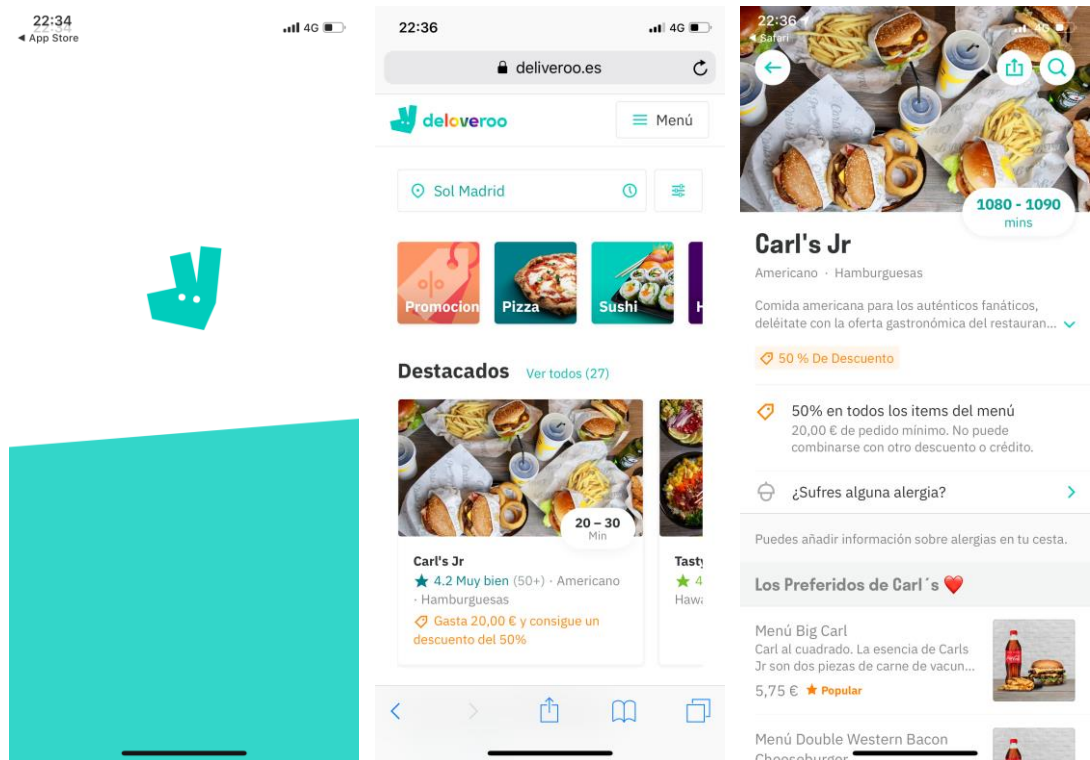


Figura 11. Aplicación Deliveroo

Yelp



Figura 12. Icono Yelp

Es una red social dedicada mejorar la visibilidad de restaurantes y comercios a sus usuarios mediante el uso de reseñas. Se basa en mantener las opiniones, experiencias y valoraciones de usuarios para los diferentes negocios con el objetivo de servir de guía para futuros clientes. De esta manera cada usuario puede buscar, por ejemplo, restaurantes, y puede visualizar cuales son los que tienen mejor puntuación o si tienen críticas negativas.

Además, Yelp⁴, Figura 12, tiene un valor añadido a la información que muestra ya que, ante cualquier información errónea, cualquier usuario podrá complementar o subsanar esa deficiencia para mantener datos actualizados. Cabe destacar que, ante cualquier cambio

⁴ <https://www.yelp.es/>

solicitado por un usuario, el equipo de Yelp se encarga de cotejar su veracidad para tramitar o no esa la modificación.

En la Figura 13 se muestra el aspecto de la aplicación Yelp para iOS.

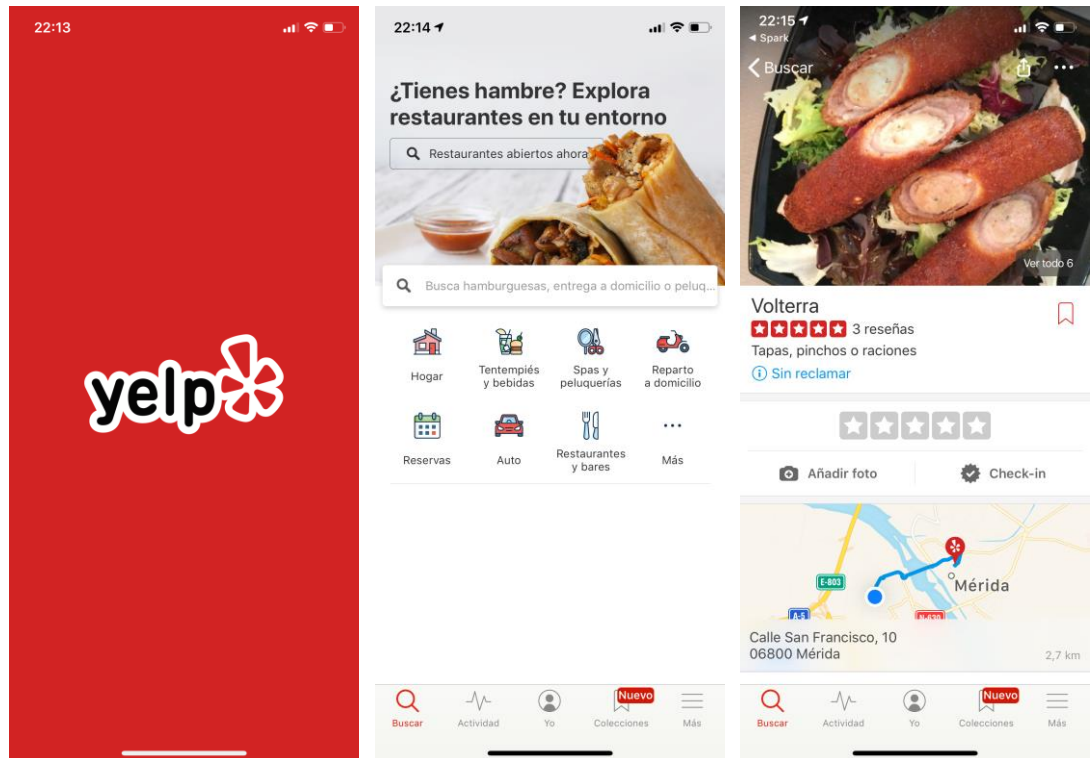


Figura 13. Aplicación Yelp

Glovo



Figura 14. Icono Glovo

Es una aplicación móvil, Figura 14, con soporte y complemento de aplicación web⁵, que se ha ganado una reputación e importancia dentro del mercado de entregas a domicilio bajo su novedosa idea: prometer la entrega de cualquier cosa pequeña, aproximadamente de un tamaño de 40x40cm que permita su transporte en bicicleta o motocicleta, en menos de 60 minutos, al lugar de entrega indicado. Se pueden, por tanto, realizar pedidos de comida, flores o productos de farmacia [12].

Este servicio es realizado por repartidores o “Glovers” empleados de Glovo encargados de realizar el pedido, como, por ejemplo, la compra de flores en una floristería y, posteriormente, entregar el pedido al cliente. Estos repartidores son personas independientes que pueden

⁵ <https://glovoapp.com/es/>

trabajar para Glovo, necesitando únicamente su teléfono y su vehículo. De esta manera, reciben los pedidos en su teléfono de los clientes de Glovo y llevan a cabo el pedido hasta su entrega.

En la Figura 15 se muestra el aspecto de la aplicación Glovo para iOS.

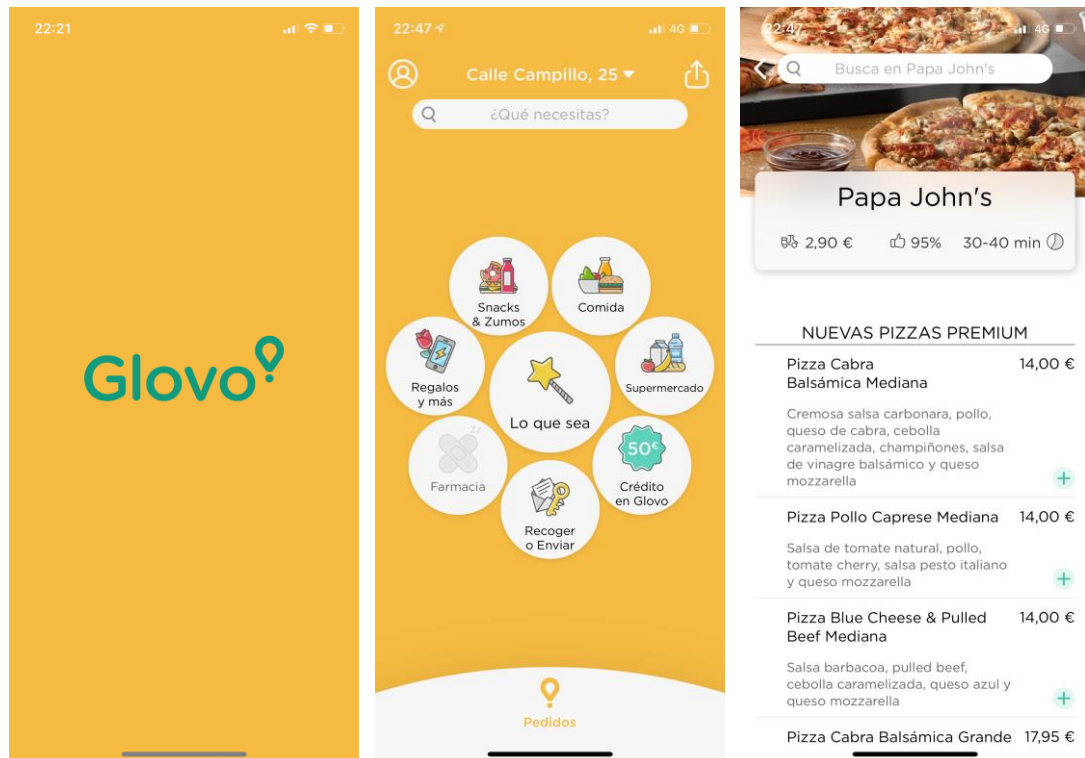


Figura 15. Aplicación Glovo

Comparativa de aplicaciones del estudio de mercado

Una vez realizado el estudio de mercado sobre las diferentes aplicaciones móviles ofrecen servicios de comidas, se va a realizar una comparativa funcional, Tabla 3, bajo los siguientes criterios a evaluar:

- **Búsqueda de restaurantes:** Indica si contiene un buscador de restaurantes y si contiene geolocalización.
- **Carta de los restaurantes:** Indica si contiene la carta y platos con precios del restaurante.
- **Información detallada del restaurante:** Indica si se muestra toda la información relevante de restaurante como ubicación, contacto y horario.
- **Reservas de restaurantes:** Indica si permiten la reserva de mesa en los restaurantes.

- **Opiniones de usuarios:** Indica se muestra información de opiniones de otros clientes para los restaurantes.
- **Programa de incentivos:** Indica si contiene información sobre promociones, programa de puntos o de algún sistema de incentivo mediante el uso.
- **Ingredientes e información de alérgenos:** Indica si por cada plato del restaurante se muestran los ingredientes y la composición de alérgenos alimenticios.
- **Búsqueda avanzada:** Indica si contiene una búsqueda avanzada de platos, ingredientes o contenido no alérgico.
- **Sugerencias por restaurantes:** Indica si proporciona información personalizada a los usuarios sobre platos ideales con forme a sus gustos culinarios.
- **Precios o cuotas:** Indica si el servicio tiene coste para los restaurantes.

Tabla 3. Comparativa funcional de aplicaciones móviles

Aplicación	Globo	Just Eat	El Tenedor	Deliveroo	Yelp
Búsqueda	✓	✓	✓	✓	✓
Carta	✓	✓	✓	✓	✗
Información	✗	✓	✓	✓	✓
Reservas	✗	✗	✓	✗	✗
Opiniones	✗	✓	✓	✗	✓
Incentivos	✓	✓	✓	✗	✗
Alérgenos	✗	✗	✗	✗	✗
Avanzada	✗	✓	✗	✗	✗
Sugerencias	✗	✗	✗	✗	✗
Coste	✗	✓	✓	✓	✗

(Elaboración propia)

2.4. Conclusiones del estado del arte

En este capítulo se han estudiado los contextos en los que va a estar basado este trabajo.

Para empezar, se ha introducido la disciplina de *Machine Learning* y en concreto se ha dado importancia a los algoritmos de aprendizaje supervisado, ya que el problema a solventar en este trabajo está basado en un problema de clasificación.

Además, dentro del aprendizaje supervisado, se han presentado dos opciones de algoritmos: árboles de decisión y el algoritmo de *Naïve Bayes*. Debido a las características del problema,

la predicción de alimentos, se toma la decisión de profundizar en el algoritmo de *Naïve Bayes* y no en los árboles de decisión, como método para solucionar el problema. Esta decisión radica principalmente en las características de aplicabilidad de los árboles de decisión, los cuales son idealmente aplicables en problemas donde la función de salida tiene un número de valores discretos y pequeños, afirmación, que a priori, no corresponde con las premisas del problema.

Posteriormente, se han valorado las dos principales opciones para el desarrollo de aplicaciones móviles, como son la implementación nativa y la implementación híbrida, cada una con sus ventajas y desventajas. De este estudio se destaca la versatilidad en el desarrollo de aplicaciones híbridas, con beneficios importantes como, el menor coste o en el menor tiempo de desarrollo. Al igual que, la implementación nativa, es la opción ideal para el desarrollo de aplicaciones muy específicas, o dedicadas a un mayor procesamiento, o un mejor rendimiento, como pueden ser los videojuegos.

Por último, se ha realizado un estudio de mercado de las principales aplicaciones móviles, enfocadas a ofrecer servicios de comidas, y se han comparado en función de una serie de criterios funcionales. De este estudio podemos destacar dos principales carencias en las aplicaciones analizadas: información de alérgenos en los platos y sugerencias personalizadas de menús a los usuarios. Estos puntos pueden ser un factor novedoso y de impacto para el desarrollo de una aplicación dentro de este mercado.

3. Objetivos concretos y metodología de trabajo

3.1. Objetivo general

El objetivo principal de este trabajo es desarrollar una aplicación móvil, *Tu paladar*, que proporcione información detallada de restaurantes y contenidos alérgenos a los usuarios, que esté dotada de funcionalidades como consultar la carta y precios de los restaurantes, reservar mesas o valorar los restaurantes y platos, y cuya finalidad innovadora es la predicción de comida o plato “ideal” para cada usuario / restaurante mediante la utilización de técnicas de inteligencia artificial y mejora continua.

3.2. Objetivos específicos

- Proporcionar información detallada de la carta y platos de los restaurantes, así como del contenido de ingredientes alérgenos.
- Proporcionar sugerencias de platos ideales para los comensales por restaurantes mediante un sistema de aprendizaje personalizado y automatizado por usuario mediante la combinación, por un lado, de sus gustos, alergias y opiniones y, por otro lado, la información de los platos, de los restaurantes y de las opiniones de otros usuarios.
- Determinar que parámetros se pueden tener en cuenta de los usuarios de la aplicación, para utilizarlos en la búsqueda de predicción de sugerencia de platos.
- Identificar que algoritmo de Inteligencia Artificial puede ofrecer una solución factible para la predicción de sugerencia de platos.
- Desarrollar una única aplicación híbrida móvil para las dos principales arquitecturas de móviles, Android e iOS con el objetivo de reducir costes derivados del desarrollo de dos aplicaciones.

3.3. Metodología del trabajo

La metodología de trabajo seguida en el ciclo de desarrollo de software de la aplicación es RUP (del inglés *Rational Unified Process*), que corresponde con una metodología de desarrollo e Ingeniería de Procesos Software [13] [14], variante del proceso unificado de desarrollo de software (USDP, del inglés *Unified Software Development Process*) o de manera simplificada proceso unificado (UP, del inglés *Unified Process*) [15].

3.3.1. Metodología de desarrollo software RUP

Es un producto comercial de IBM basado en la utilización de UML (*Unified Modeling Language*) para la representación y diseño del sistema de software y define un modelo de proceso de software que puede ser representado y descrito bajo tres perspectivas diferentes: sean Perspectiva estática, Perspectiva dinámica y Perspectiva práctica, a diferencia de los modelos de proceso convencionales que presentan una única perspectiva de todo el proceso de desarrollo [13] [16].

La estrategia que propone la metodología RUP está basada en el modelo de proceso iterativo e incremental. El ciclo de vida del sistema se divide en iteraciones donde cada una de ellas se desarrollan en el tiempo de manera consecutiva. Dentro de cada iteración o ciclo se descomponen una serie de fases que son ejecutadas. Mediante esta interpretación, cada ciclo es una iteración de desarrollo del sistema que concluye en su resultado con una generación del producto, incluyendo el resultado de todas las iteraciones anteriores más un incremento del resultante de la iteración [14] [16].

Bajo esta estrategia, cada iteración se encarga del desarrollo de una parte funcional del sistema. Al finalizarse una iteración, se realiza un estudio y análisis del resultado obtenido, donde se realiza una valoración del cumplimiento de los hitos marcados. Esta conclusión parcial permite retroalimentar el sistema y valorar el estado de la planificación del proyecto. Además, permite la evaluación de la posible aparición de nuevos requisitos y riesgos en el sistema, o incluso cambios en los ya existentes, con el fin de actuar en consecuencia y planificarlos en la posterior iteración [13] [14].

Fases

Cada iteración dentro del ciclo de vida RUP está compuesta por 4 fases. Cada fase tiene como finalización objetivo un hito claramente definido en el tiempo, en el cual se deben realizar las acciones y tomar las decisiones oportunas para alcanzar las tareas marcadas y así poder pasar a la siguiente fase [13] [17].

Las fases son las siguientes [13] [14] [17]:

- **Fase de inicio (Inception Phase).** Es la primera fase del proceso. Se comienza en base a la idea inicial para elaborar el análisis del producto mediante el modelo de negocio, el ámbito del proyecto y sus límites. Además, en esta fase se realiza un estudio de los riesgos potenciales del sistema, se identifican los requisitos para los

casos de usos más importantes, se establecen los costes de recursos y tiempo, y se realiza la planificación.

- **Fase de elaboración (Elaboration Phase).** En esta fase se pretende modelar y diseñar la arquitectura de sistema y desarrollar el plan de proyecto para que sirva como base en los trabajos de implementación de la fase de construcción. En base a esta arquitectura se realiza una descripción del producto final y refinan los requisitos críticos del sistema para clarificar los casos de usos del sistema. Además, se realiza la planificación de la implementación técnica de la fase de construcción y se eliminan los riesgos potenciales
- **Fase de Construcción (Construction Phase).** En esta tercera fase del proceso, realiza la construcción e implementación del software y se produce una versión ejecutable del mismo siguiendo las pautas marcadas en la línea de base y arquitectura diseñadas en las fases anteriores. Por tanto, el objetivo es alcanzar la capacidad operacional y los objetivos de negocio marcados del proyecto mediante los requerimientos y funcionalidades del sistema, reevaluando este proceso de manera sucesiva e incremental. Al final de esta fase se obtiene un producto estable para poder ser evaluado por el usuario.
- **Fase de Transición (Transition Phase).** También puede ser denominada fase de evaluación, tiene como objetivo la puesta en marcha del sistema y a disposición de los usuarios finales para realizar todas las pruebas y test posibles y, de este modo, mitigar todos los errores y bugs que pueda contener el software. En esta fase se pueden generar distintas versiones del producto o diferentes versiones betas.

En la siguiente Figura 16 se muestra el esquema en dos dimensiones de las iteraciones y fases de la metodología RUP: el eje vertical para visualizar la organización, desarrollo y crecimiento del producto y el eje horizontal, la temporalidad de este a lo largo del tiempo.

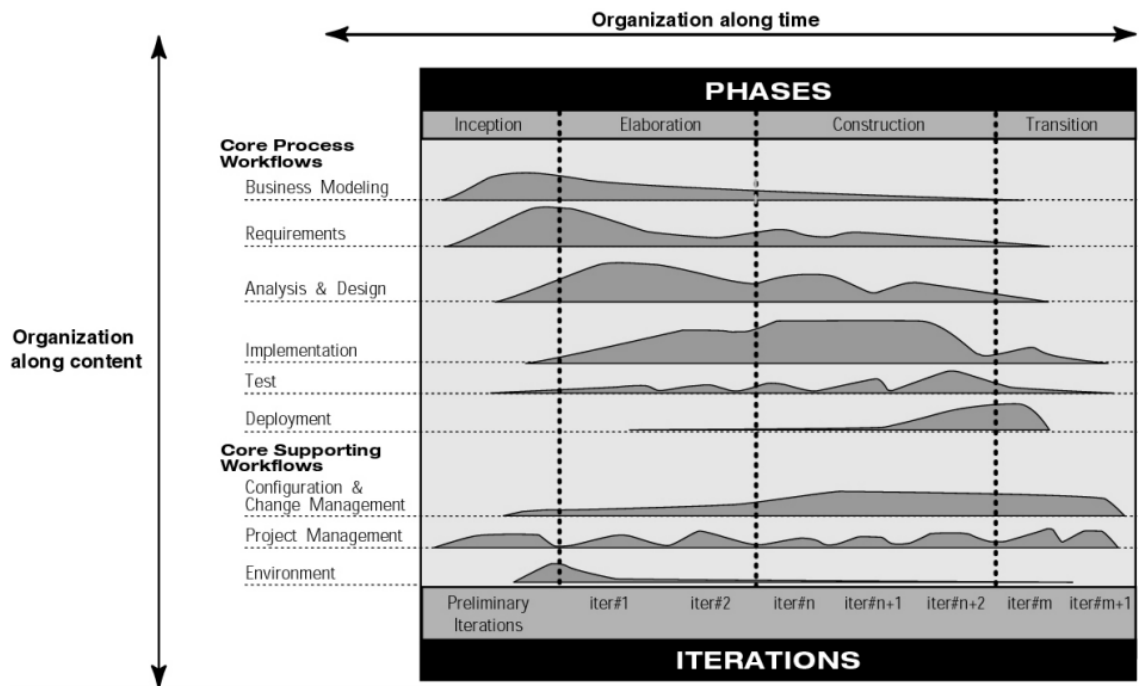


Figura 16. Representación de iteraciones y fases metodología RUP [14]

Por último, se pueden enumerar algunas de las principales ventajas de la utilización de la metodología RUP en la Ingeniería del Software [14]:

- La principal ventaja es que el **software** se desarrolla de manera **iterativa e incrementalmente**.
- Está compuesta de amplia información sobre la **gestión de requisitos** en modelos de negocio.
- Está basada en una **arquitectura orientada a componente** que permite software más flexible, adaptativo, fácil ampliación y reutilización.
- El software, requerimientos y funcionalidades son diseñados y **modelados** de manera **visual** para facilitar su comprensión.
- Por sus características, permite desarrollar **software** y productos de mayor **calidad**.
- Software desarrollado bajo una **gestión, control** y versionado de **cambios**.

4. Aplicación TuPaladar

4.1. Análisis

El objetivo principal de este apartado es la definición de las características del sistema a desarrollar, mediante la descripción general del sistema y su alcance, la identificación de requerimientos que debe tener el sistema, tanto funcionales como no funcionales, la base teórica adaptada y la aplicación del Clasificador de Bayes en el sistema.

4.1.1. Propósito inicial

En este apartado inicial a la fase de análisis, se pretenden motivar las siguientes cuestiones **claves** para el desarrollo de la aplicación y por tanto la finalidad de este trabajo.

Sustento tecnológico

Como se ha comentado anteriormente en el apartado 2.2. Tecnologías de desarrollo para aplicaciones móviles, tenemos dos opciones para el desarrollo de la aplicación. Por un lado, podemos optar por el desarrollo de una aplicación nativa y en este caso elegir una tecnología concreta, Android o iOS por su mayor popularidad, por otro lado, podemos optar por el desarrollo de una aplicación híbrida mediante el framework de Ionic. La elección y decisión de utilizar lenguaje híbrido en lugar de lenguaje nativo está basado en el objetivo de la aplicación y por tanto es una decisión basada en las pretensiones de negocio del producto.

La opción elegida es el desarrollo de una aplicación híbrida ya que permite un grado de portabilidad entre diferentes arquitecturas y requiere el conocimiento de tecnología web menos específica, en comparación con la opciones de aplicación nativa. Cabe destacar que, debido a su complejidad infraestructural, las híbridas pueden tener un peor rendimiento en comparación con las nativas, sobre todo si se tienen requerimientos gráficos en la aplicación, pero no corresponde con el objetivo de *TuPaladar*.

Solución a un problema real

En cuanto al problema a abordar en este trabajo, en la Tabla 3 se ha observado que ninguna de las aplicaciones enumeradas realiza una distinción o identificación de contenido para alérgenos en los ingredientes que componen los platos de los restaurantes, ni proporciona una orientación culinaria a sus usuarios, como es la propuesta y objeto de este trabajo. Por tanto, podemos concluir que el desarrollo de esta aplicación va a solucionar un problema latente dentro de las aplicaciones de servicios de comida.

4.1.2. Alcance

Para el desarrollo de la aplicación, se definen los siguientes entregables para su finalización:

- Requerimientos funcionales.
- Requerimientos no funcionales.
- Adaptación del algoritmo de Bayes al problema.
- Diseño de casos de uso.
- Diseño modelo entidad-relación.
- Diseño físico de datos.
- Diseño de clases.
- Descripción funcional de la aplicación.
- Pruebas de funcionalidad.
- Pruebas de usabilidad.

4.1.1. Identificación de requisitos

Los requisitos o requerimientos de un sistema son la descripción de los servicios proporcionados por el sistema y sus restricciones operativas. [18]

El propósito de este apartado es identificar y definir las especificaciones tanto funcionales como no funcionales para el desarrollo de la app TuPaladar.

Para la especificación de los requisitos se tiene en cuenta la siguiente categoría de datos para la **Importancia**, con valores posibles: **Alta**, **Media**, **Baja**.

Requisitos funcionales

Los requisitos funcionales del sistema son las descripciones detalladas del comportamiento que debe tener el software y como debe manejar la información. [18]

Tabla 4. R.F. 01 - Registro

Requisito funcional	R.F. 01
Nombre	Registro
Descripción	El sistema debe permitir el registro de un nuevo usuario (ya sea usuario particular o usuario restaurante). En el caso de usuario particular, el registro debe de incluir datos personales, preferencias alimenticias y

	datos de alérgicos. En el caso de usuario restaurante, el registro debe de incluir datos personales.
Requerimiento no funcional	R.N.F. 01, R.N.F. 04, R.N.F. 05, R.N.F. 06
Importancia	Alta

(Elaboración propia)

Tabla 5. R.F. 02 - Validación de formulario

Requisito funcional	R.F. 02
Nombre	Validación de formulario
Descripción	El sistema debe validar y controlar que la información introducida en los campos del formulario sea correcta, y que no haya campos vacíos. El proceso de validación incluye comprobación de estructura de email, formato correcto de teléfonos, comprobación de direcciones y selección de contraseña segura.
Requerimiento no funcional	R.N.F. 04, R.N.F. 05, R.N.F. 06
Importancia	Alta

(Elaboración propia)

Tabla 6. R.F. 03 - Login

Requisito funcional	R.F. 03
Nombre	Login
Descripción	El sistema debe permitir el acceso al sistema de los usuarios previamente registrados mediante un proceso de inicio de sesión.
Requerimiento no funcional	R.N.F. 01, R.N.F. 04, R.N.F. 05, R.N.F. 06
Importancia	Alta

(Elaboración propia)

Tabla 7. R.F. 04 - Modificación de información usuario

Requisito funcional	R.F. 04
Nombre	Modificación de información usuario
Descripción	El sistema debe permitir que los usuarios puedan consultar su información personal y sus credenciales de acceso a la aplicación, y poder modificarla. Además, en el caso de usuario particular podrá consultar y modificar tanto sus preferencias culinarias como sus preferencias de alérgenos.
Requerimiento no funcional	R.N.F. 01, R.N.F. 04, R.N.F. 05, R.N.F. 06
Importancia	Media

(Elaboración propia)

Tabla 8. R.F. 05 - Consultar restaurantes

Requisito funcional	R.F. 05
Nombre	Consultar restaurantes
Descripción	Los usuarios particulares podrán visualizar un listado de y podrán acceder a cada uno de ellos para obtener más información del restaurante y su carta.
Requerimiento no funcional	R.N.F. 01, R.N.F. 02, R.N.F. 06
Importancia	Alta

(Elaboración propia)

Tabla 9. R.F. 06 - Sugerencia de plato por restaurante

Requisito funcional	R.F. 06
Nombre	Sugerencia de plato por restaurante
Descripción	El sistema, mediante la ejecución del algoritmo de predicción, sugerirá a los usuarios que hayan accedido al detalle de un restaurante, el plato ideal para él en ese restaurante.
Requerimiento no funcional	R.N.F. 02, R.N.F. 06
Importancia	Alta

(Elaboración propia)

Tabla 10. R.F. 07 - Pedir comida a domicilio

Requisito funcional	R.F. 07
Nombre	Pedir comida a domicilio
Descripción	Los usuarios que hayan accedido al detalle de un restaurante podrán seleccionar los platos que desean pedir a domicilio. Una vez seleccionados comenzará el proceso de verificación de pedido, dirección de entrega y método de pago.
Requerimiento no funcional	R.N.F. 01, R.N.F. 02, R.N.F. 04, R.N.F. 05, R.N.F. 06
Importancia	Alta

(Elaboración propia)

Tabla 11. R.F. 08 - Reservar mesa

Requisito funcional	R.F. 08
Nombre	Reservar mesa
Descripción	Los usuarios que hayan accedido al detalle de un restaurante podrán reservar mesa si hay disponibilidad en el restaurante. Para ello, deberán de rellenar un formulario con las personas y hora de la reserva y el sistema confirmara dicha reserva.

Requerimiento no funcional	R.N.F. 01, R.N.F. 04, R.N.F. 06
Importancia	Alta

(Elaboración propia)

Tabla 12. R.F. 09 - Valoración de restaurante

Requisito funcional	R.F. 09
Nombre	Valoración de restaurante
Descripción	El sistema permitirá y sugerirá a los usuarios valorar restaurantes y platos mediante la cumplimentación de un formulario.
Requerimiento no funcional	R.N.F. 01, R.N.F. 06
Importancia	Baja

(Elaboración propia)

Tabla 13. R.F. 10 - Logout

Requisito funcional	R.F. 10
Nombre	Logout
Descripción	El sistema permitirá que los usuarios que hayan iniciado sesión en la aplicación puedan salir de la aplicación mediante el cierre de la sesión.
Requerimiento no funcional	R.N.F. 04, R.N.F. 05, R.N.F. 06
Importancia	Baja

(Elaboración propia)

Tabla 14. R.F. 11 - Registro de restaurante

Requisito funcional	R.F. 11
Nombre	Registro de restaurante
Descripción	El sistema permitirá a los usuarios restaurantes realizar el registro de un nuevo restaurante, para ello el usuario restaurante debe rellenar un formulario con los datos del restaurante, y el registro de los platos de la carta. Cada plato contendrá datos de información, ingredientes principales y contenido alérgeno.
Requerimiento no funcional	R.N.F. 01, R.N.F. 04, R.N.F. 05, R.N.F. 06
Importancia	Media

(Elaboración propia)

Tabla 15. R.F. 12 - Modificación de restaurante

Requisito funcional	R.F. 12
Nombre	Modificación de restaurante

Descripción	El sistema permitirá a los usuarios restaurantes gestionar sus restaurantes, modificación de la información de cada restaurante, modificación de la información de los platos, alta de nuevos platos, eliminación de platos y eliminación del propio restaurante.
Requerimiento no funcional	R.N.F. 01, R.N.F. 04, R.N.F. 05, R.N.F. 06
Importancia	Media

(Elaboración propia)

Tabla 16. R.F. 13 - Envío de sugerencias

Requisito funcional	R.F. 13
Nombre	Envío de sugerencias
Descripción	El sistema permitirá a los usuarios de la aplicación la cumplimentación de un formulario general, para el envío de opiniones y sugerencias sobre el funcionamiento de la aplicación.
Requerimiento no funcional	R.N.F. 01, R.N.F. 04, R.N.F. 06
Importancia	Baja

(Elaboración propia)

Tabla 17. R.F. 14 - Actualización de la base de conocimiento

Requisito funcional	R.F. 14
Nombre	Actualización de la base de conocimiento
Descripción	El sistema actualizará la base de conocimiento utilizada por el algoritmo predictivo por cada registro de usuario particular en base a sus preferencias culinarias bajo los parámetros de su información personal.
Requerimiento no funcional	R.N.F. 02, R.N.F. 06
Importancia	Alta

(Elaboración propia)

Requisitos no funcionales

Estos requisitos son aquellos requerimientos que no se refieren a las funciones específicas proporcionadas por el sistema, sino que representan características generales y restricciones de la aplicación. [18]

Tabla 18. R.N.F. 01 - Usabilidad

Requisito funcional	R.N.F. 01
Nombre	Usabilidad

Descripción	<ul style="list-style-type: none"> El sistema responsive se adapta a los diferentes formato de pantalla, tanto de terminales iOS, Android, Backberry OS y Windows Phone. El sistema está desarrollado siguiendo pautas de usabilidad como fácil uso y acceso, diseño estético agradable y fácil aprendizaje. El sistema implementa un procedimiento de retroalimentación o feedback para permitir que los usuarios envíen opiniones sobre la aplicación y su funcionamiento.
Importancia	Alta

(Elaboración propia)

Tabla 19. R.N.F. 02 - Rendimiento

Requisito funcional	R.N.F. 02
Nombre	Rendimiento
Descripción	<ul style="list-style-type: none"> El sistema debe responder correctamente a las demandas de solicitud de información de los usuarios. El sistema en la ejecución del proceso de sugerencias de platos debe responder correctamente y en un tiempo adecuado.
Importancia	Alta

(Elaboración propia)

Tabla 20. R.N.F. 03 - Portabilidad

Requisito funcional	R.N.F. 03
Nombre	Portabilidad
Descripción	<ul style="list-style-type: none"> La aplicación es portable por su diseño y arquitectura tecnológica a diferentes entornos: iOS, Android, Windows Phone y PWA.
Importancia	Alta

(Elaboración propia)

Tabla 21. R.N.F. 04 - Privacidad

Requisito funcional	R.N.F. 04
Nombre	Privacidad
Descripción	<ul style="list-style-type: none"> El sistema implementa unas Políticas y Condiciones de Privacidad de la Información que almacena.
Importancia	Alta

(Elaboración propia)

Tabla 22. R.N.F. 05 - Seguridad

Requisito funcional	R.N.F. 05
Nombre	Seguridad
Descripción	<ul style="list-style-type: none"> El sistema implementa unas Políticas de Seguridad y Condiciones de uso.

	<ul style="list-style-type: none"> • El sistema realiza el proceso de identificación y autenticación de los usuarios en el sistema mediante un proceso de login usuario/contraseña. • El sistema implementa mecanismos de validación de los datos de entrada introducidos. • El sistema implementa un mecanismo para la encriptación de contraseñas por algoritmo de autenticación digest SHA-256. • El sistema implementará mecanismos de auditorías sobre todos los procesos que persistan información. • El sistema realiza un envío y recepción de datos mediante el cifrado de la información punto a punto. • El sistema implementa un sistema de backup de Base de Datos para salvaguardar la información de manera periódica. • El sistema implementa medidas para preservar los R.N.F de Privacidad y Disponibilidad de la información.
Importancia	Alta

(Elaboración propia)

Tabla 23. R.N.F. 06 - Disponibilidad

Requisito funcional	R.N.F. 06
Nombre	Disponibilidad
Descripción	<ul style="list-style-type: none"> • El sistema necesita estar siempre conectado al centro de centralizado de datos. Debe estar disponible en cualquier momento.
Importancia	Alta

(Elaboración propia)

4.1.2. Aplicación de Algoritmo de Bayes

En este apartado se van a aplicar los conceptos aprendidos del aprendizaje automático supervisado y del Clasificador de Bayes, al problema a solventar por esta aplicación.

Este método, tal como vimos en el apartado 2.1. Machine Learning y aprendizaje supervisado, es una técnica de predicción probabilística, donde dados una serie de parámetros de entrada, son procesados por la función del algoritmo, para obtener como resultado una clase o catalogación concreta en función del histórico de datos o experiencias que contiene.

Por un lado, necesitamos un conjunto de entrenamiento inicial, en adelante **base de conocimiento inicial**, que sirva como base de información al método. Este concepto es una característica del aprendizaje supervisado.

Por otro lado, el método necesita un conjunto de hipótesis previamente preestablecidas, a estas hipótesis las denominaremos **clases**. El Algoritmo de Bayes calcula las probabilidades de que una clase ocurra, en base a la información contenida en su base de conocimiento.

Además, y aunque parezca una afirmación trivial, los valores o **atributos etiquetados** de los datos contenidos en la base de conocimiento tiene relación con las clases, ya que esa base de conocimiento se ha generado como aprendizaje previo al propio método.

Por tanto, los datos que maneja el algoritmo son objetos cuyos atributos pertenecen a diferentes clases, donde el objetivo del algoritmo es predecir una clase dados una serie de valores de atributos.

Formulación del problema

El problema para solventar en este trabajo se formula con las siguientes cuestiones:

¿Qué platos de la carta de un restaurante es el más indicado para sugerir a una persona?
¿Cuál sería el plato ideal para sugerir a un comensal en un restaurante concreto?

Para solucionar este problema vamos a aplicar el *Clasificador de Bayes*, pero, nos encontramos con la problemática de, ¿Qué datos contendrá nuestra base de conocimiento? ¿Qué atributos etiquetados utilizará el algoritmo? ¿Y qué clases?

Base de conocimiento inicial

Para enriquecer inicialmente la base de conocimiento del algoritmo vamos a analizar qué información necesitamos procesar para dar solución al problema:

Lo primero que se necesita saber, es información sobre las **personas**. Una serie de parámetros que se puedan utilizar para predecir, en base ellos, si le gustará un plato o no. Los parámetros a tener en cuenta sobre las personas son los siguientes:

- **Según sexo.** Con valores: Hombre o mujer.
- **Según edad.** Con valores: Entre 20 y 35 años, entre 36 y 45 años, entre 46 y 55 años, entre 56 y 65 años, o más de 65 años.
- **Según hábitat.** Con valores: Menos de 10.000 habitantes, entre 10.001 y 50.000 habitantes, entre 50.001 y 100.000 habitantes, o más de 100.000 habitantes.

En cuanto a los platos, necesitaremos conocer la carta de los restaurantes. Por tanto, de cada plato de cada restaurante necesitaremos la siguiente información:

- **Ingredientes principales.** Divididos en dos grupos. Carnes: Vacuno, Porcino, Pollo, Ovino/Caprino, Conejo, Pavo y otras aves. Pescados: Blanco y Azul.

Además, otro punto importante a la hora de decidir si, un plato se debe sugerir o no a un comensal, es el contenido de algún alimento alérgeno. De esta manera, hay que mantener información de alimentos con contenidos alérgenos, para evitar que los comensales que tengan algún tipo intolerancia o alergia consuman de manera inconsciente ese alérgeno por desinformación del plato. Los datos de alérgenos que maneja la base de conocimiento del sistema se muestran en la siguiente Figura 17.



Figura 17. 14 alérgenos a especificar en la carta [19]

Por tanto, necesitamos dos datos nuevos:

- Conocer previamente las posibles **intolerancias de las personas**.
- Conocer los contenidos de **alérgenos en los platos**.

Fuente de datos

Según los estudios [20] y [21] realizados por el *Observatorio del Consumo del Ministerio de Medio Ambiente y Medio Rural y Marino*, en España en los años 2008 y 2009 respectivamente, podemos observar la tendencia de consumo tanto de *Carne* como de *Pescado*. Para ello este estudio se realizó en base a diferentes parámetros de las personas encuestadas.

A continuación, se muestran los monográficos resultantes del estudio estadístico de con qué frecuencia se consume carne, Figura 18, y pescado, Figura 19, el estudio al completo de ambos grupos de alimentos se puede visualizar en las referencias [20] y [21].

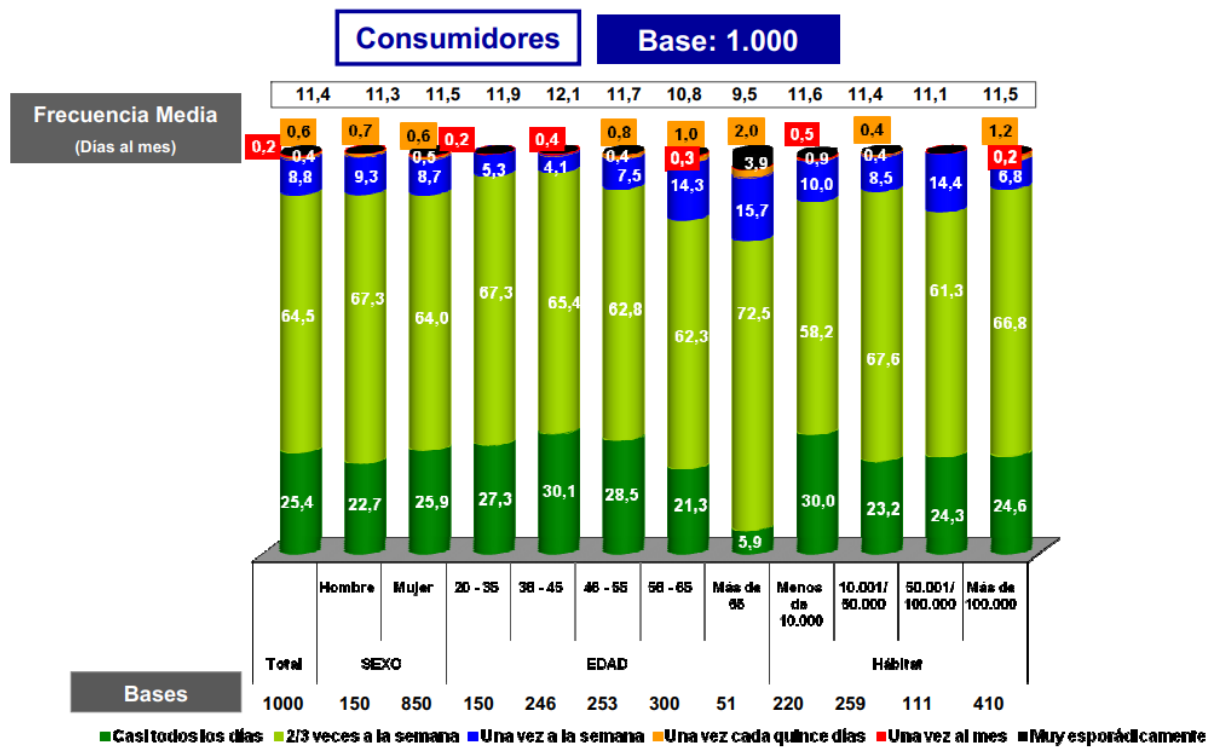


Figura 18. Frecuencia de consumo de carne según sexo, edad y habitat [20]

En la Figura 18, se puede observar que el estudio realizado indica los porcentajes de frecuencia de consumo de carne al mes, en una prueba muestral de 1.000 personas, y en base a su sexo, rango de edad y población de la ciudad de residencia.

Por ejemplo, se puede observar que, de esas 1.000 personas, 150 son hombre y 850 son mujeres, o que 300 estaban comprendidas entre los 56 y 65 años.

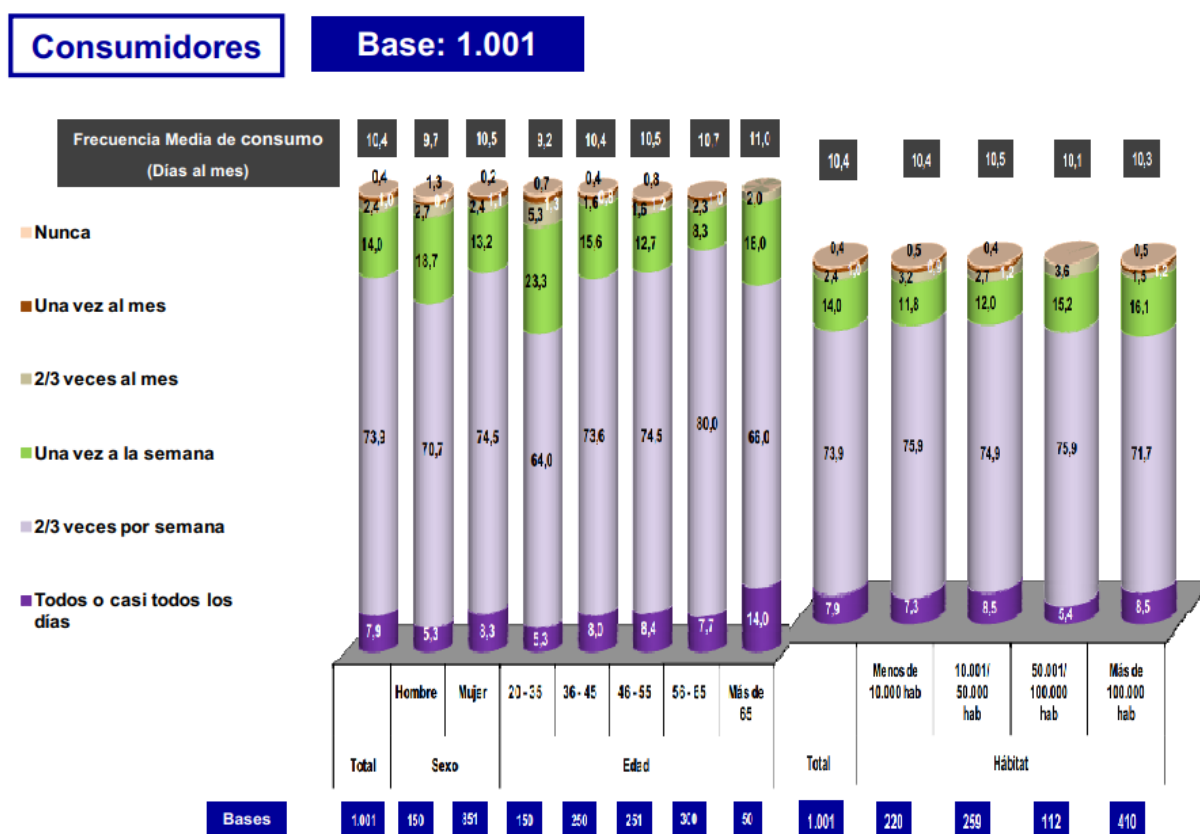


Figura 19. Frecuencia de consumo de pescado según sexo, edad y habitat [21]

En cambio, en la Figura 19, se puede observar que el estudio realizado indica los porcentajes de frecuencia de consumo de pescado al mes, en una prueba muestral de 1.001 personas, y en base a su sexo, rango de edad y población de la ciudad de residencia.

Además, para el caso de carne, el estudio complementa concretamente los diferentes tipos de carne que existen con los diferentes parámetros de las personas encuestadas, obteniendo los siguientes resultados mostrados en la Figura 20, Figura 21 y Figura 22:



Figura 20. Consumo de los tipos de carne por sexo [20]

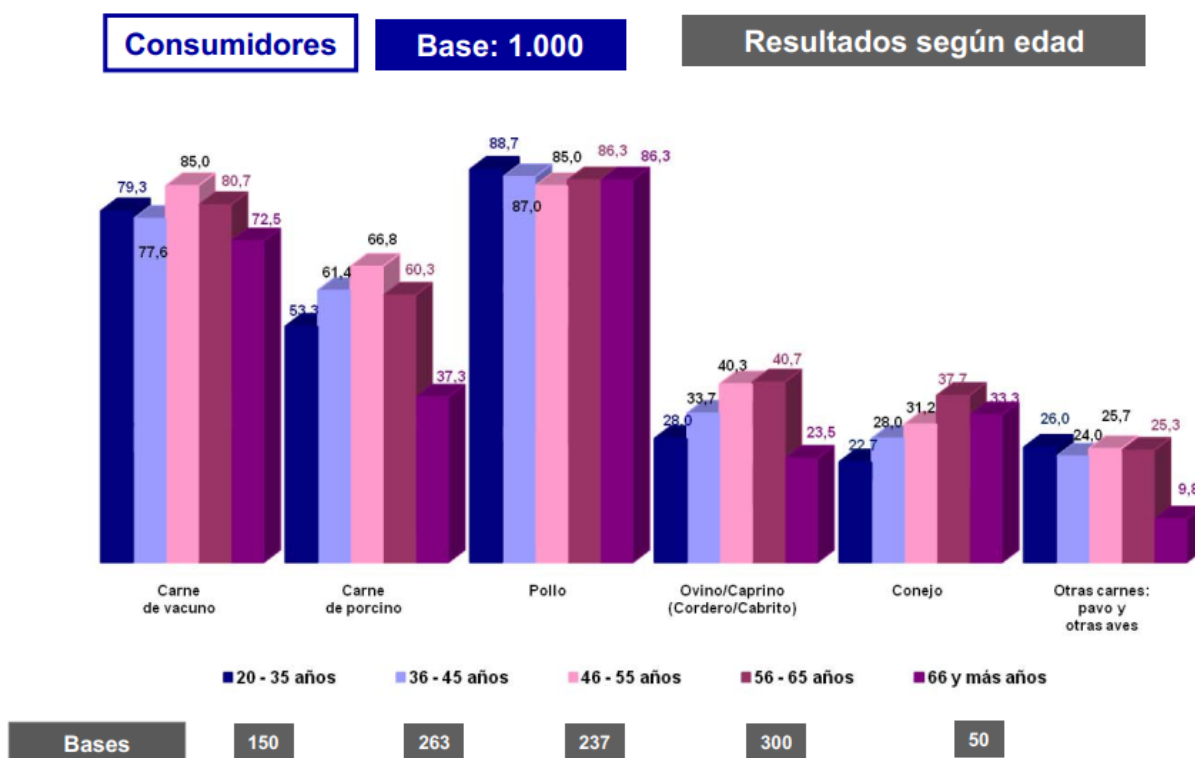


Figura 21. Consumo de tipos de carne por edad [20]

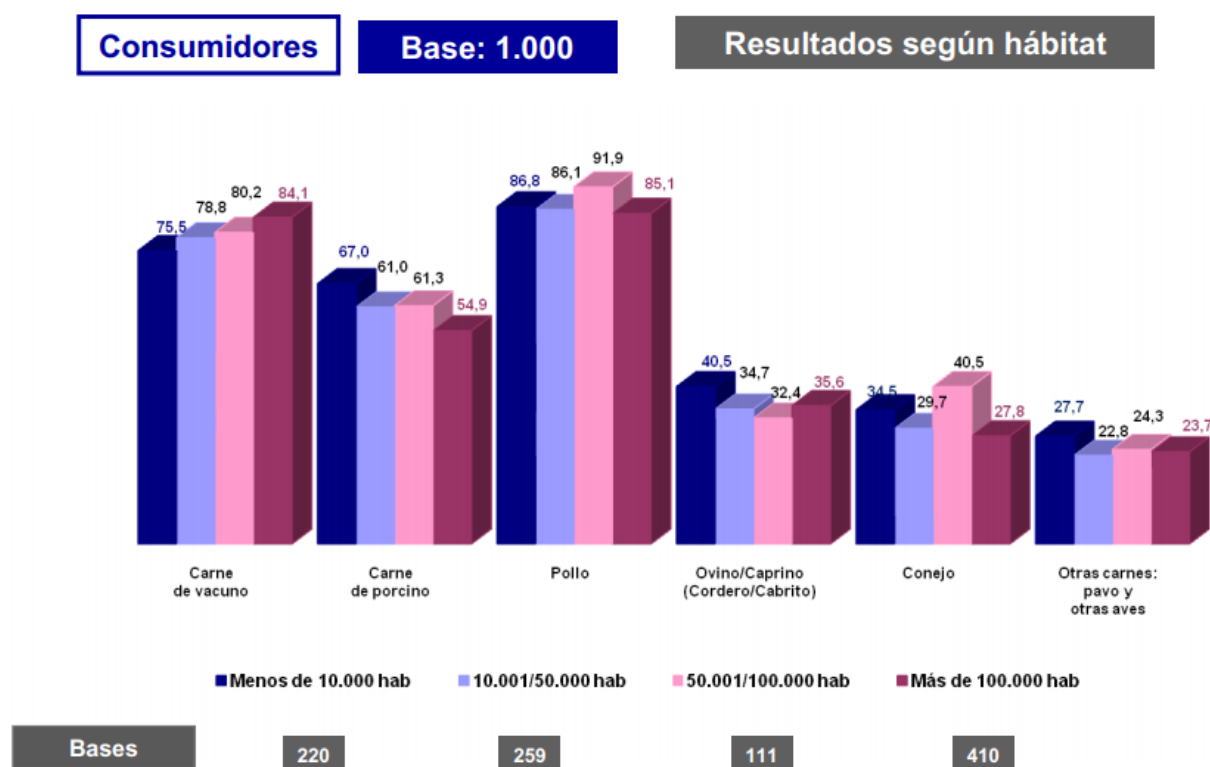


Figura 22. Consumo de tipos de carne por hábitat [20]

Para la base de conocimiento inicial de este trabajo se va a utilizar este estudio, por tanto, vamos a tener la frecuencia de consumo de carne y pescado para personas según su sexo, edad y hábitat, y además tenemos información de consumo concreto de diferentes tipos de carne para personas según su sexo, edad y hábitat.

Por otro lado, y en base a este estudio, podemos sacar las etiquetas de los grupos de ingredientes principales en las comidas, es decir, carne y pescado.

Para el caso de carne, podemos realizar la catalogación con los valores: carne de vacuno, carne de porcino, carne de pollo, carne de ovino/caprino, carne de conejo, y carne de pavo y otras aves.

En cuanto al pescado, se diferencian dos grupos: pescado azul y pescado blanco. Estos dos subgrupos comenzarán sin datos en la base de conocimiento, por lo tanto, se ponderarán equitativamente con los datos de la frecuencia de consumo de pescado, hasta que el algoritmo con el tiempo vaya aprendiendo y afinando estos datos.

Datos del entrada al algoritmo

Para que el algoritmo pueda predecir de manera individualizada las sugerencias a proponer a los usuarios de la aplicación, el sistema debe de almacenar la siguiente información que, además, será utilizada como parámetros del entrada del algoritmo:

- Sexo del usuario.
- Edad del usuario.
- Información de su domicilio.
- Valoración o ponderación de sus gustos sobre los diferentes tipos de carnes y pescados.
- Información de alergias.

La única información que falta para que el algoritmo pueda predecir el plato a sugerir, es la información del menú del restaurante:

- Platos de la carta.
- De cada plato su contenido en ingredientes principales.
- De cada plato el contenido alérgeno.

Hipótesis o clases del algoritmo

En cuanto a las clases a ponderar y predecir por el algoritmo para poder sugerir a un comensal un plato, se va a utilizar los ingredientes principales, por ello, vamos a contemplar 8 clases diferentes:

- Carne de vacuno.
- Carne de porcino.
- Carne de pollo.
- Carne de ovino/caprino.
- Carne de conejo.
- Carne de pavo y otras aves.
- Pescado blanco.
- Pescado azul.

Aplicación del algoritmo

Una vez que se tiene toda la información, el proceso o lógica para dar la solución final de la predicción de un plato se divide en dos fases.

Fase Uno

En esta primera fase, se realiza la aplicación del el método de Bayes, recordando el problema que se quiere solucionar:

*¿Qué platos de la carta de un restaurante es el más indicado para sugerir a una persona?
¿Cuál sería el plato ideal para sugerir a un comensal en un restaurante concreto?*

El método de Bayes va a permitir el cálculo probabilístico de afirmaciones como la siguiente:

Sabiendo la probabilidad de que, un hombre, de 29 años, que vive en una ciudad de menos de 50.000 habitantes, coma carne de vaca, se podría saber la probabilidad de que coma carne de vaca, dado que es hombre de 29 años y vive en una ciudad de menos de 50.000 hab.

Al aplicar el método sobre la base de conocimiento, y partiendo de que se tienen 8 clases o hipótesis, el resultado será una de las 8 clases, es decir, sugerir un plato de carne de vaca, por ejemplo.

Según se ha especificado en el apartado 2.1. Machine Learning y aprendizaje supervisado y en concreto mediante la ecuación de la Figura 3, necesitamos encontrar la clase cuyo resultado del método sea el máximo, h_{MAP} .

Además, se puede suponer que estamos en una situación donde cada clase a priori tiene la misma probabilidad de ser elegida, es decir, todas las clases son equiprobables de ser elegidas.

Por tanto, se concluye que, para aplicar el clasificador de Bayes al problema de predicción de platos en restaurantes, se debe usar la ecuación expuesta en la Figura 4.

A continuación, se realiza un ejemplo de la aplicación de método hasta el momento.

Supongamos que se ha registrado un nuevo usuario en la aplicación con las siguientes características:

- Mujer de 39 años, residente en Madrid.

Se quiere predecir la sugerencia de plato, y por simplificar vamos a suponer que tenemos dos hipótesis objetivo: carne de vacuno o carne de pollo. Además, se tiene en la base de conocimiento la siguiente información, en base a las etiquetas de sexo, edad y hábitat del usuario:

- Valor carne vacuno / Mujer : 0,78 ; Valor carne vacuno / 36-45 años : 0,89 ; Valor carne vacuno / Más 100.000 hab. : 0,82

- Valor carne pollo / Mujer : 0,90 ; Valor carne pollo / 36-45 años : 0,80 ; Valor carne pollo / Más 100.000 hab. : 0,83

Aplicamos la ecuación de la Figura 4 para ambas clases:

$$P(\text{Carne vacuno})$$

$$= P(\text{Mujer}|\text{Carne vacuno}) P(36 - 45 \text{ años}|\text{Carne vacuno}) P(100.000\text{hab}|\text{Carne vacuno})$$

$$P(\text{Carne vacuno}) = 0,78 * 0,89 * 0,82 = 0,569$$

$$P(\text{Carne pollo})$$

$$= P(\text{Mujer}|\text{Carne pollo}) P(36 - 45 \text{ años}|\text{Carne pollo}) P(100.000\text{hab}|\text{Carne pollo})$$

$$P(\text{Carne pollo}) = 0,90 * 0,80 * 0,83 = 0,597$$

$$h_{ML} = \text{Carne pollo}$$

Figura 23. Ejemplo hipótesis de máxima probabilidad

(Elaboración propia)

Fase Dos

En esta segunda fase se realiza un proceso de lógica de negocio exclusivo del problema, ajeno a la aplicación del algoritmo. El resultado de la fase uno nos muestra el ingrediente principal predicho por el algoritmo para sugerir de un plato.

El siguiente paso sería discriminar de la lista de platos disponibles del restaurante los platos cuyo ingrediente principal no se corresponda con el sugerido por el algoritmo.

A continuación, hay que realizar otro paso discriminatorio, correspondiente con los platos que contengan algún tipo de alérgeno y que sea intolerante para el usuario.

Para ello, dada la lista de alérgicos del usuario, la lista de platos exclusivamente con el ingrediente principal predicho por el algoritmo, y la lista de contenido alérgeno de cada plato, se eliminan de la lista de platos aquellos que contengan algún alérgeno de la lista del usuario.

Para finalizar, tendremos una posible lista de platos del restaurante, de los cuales vamos a sugerir al usuario, **únicamente**, un plato, bajo los siguientes criterios:

- Si la **lista resultante es vacía**, (ya que no había ningún plato cuyo ingrediente principal sea el predicho por el algoritmo, o lo había, pero contenida algún tipo de alérgeno incompatible con el usuario), no se sugiere ningún plato al usuario.
- Si la lista **resultante contiene un plato**, este plato se sugiere al usuario.

- Si la lista **resultante contiene más de un plato**, se sugiere, como criterio de modelo de negocio, uno de ellos al usuario.

Actualización de la base de conocimiento

Una vez el método en la aplicación esté activo, la base de conocimiento se va a enriquecer de los usuarios de la aplicación. Para ello es necesario conocer las preferencias de cada usuarios para las hipótesis.

En el momento del registro de un usuario en la aplicación, se le solicitara los siguientes datos (solo se indican los relevantes al algoritmo):

- Sexo, edad y ciudad (para conocer el habitat).
- Se le indicará que valore sus preferencias en base a las 8 clases.

Con esta información y acabado el registro se ejecutará un proceso para actualizar la base de conocimiento en base a los valores etiquetados del usuario y sus preferencias alimenticias de clases.

Esta nueva experiencia será una nueva muestra para nuestra base de conocimiento y aprenderá de ella mediante las siguientes ecuaciones (Figura 24):

$$NVM_k = \frac{(VAM_k * TM_k) + NVI}{TM_k + 1}$$

$$NTM_k = TM_k + 1$$

Figura 24. Ecuaciones actualización base de conocimiento

(Elaboración propia)

Donde:

- k : es la Clase cuyos valores se van a actualizar en la base de conocimiento.
- NVM : es el Nuevo Valor Muestral que contendrá la base de conocimiento.
- VAM : es el Valor Actual Muestral que contiene la base de conocimiento.
- TM : corresponde con el Tamaño Muestral actual que contiene la base de conocimiento.
- NVI : es el Nuevo Valor Individual que corresponde con la valoración del usuario registrado en la aplicación.
- NTM : es el Nuevo Tamaño Muestral que contendrá la base de conocimiento.

Cabe destacar que las ecuaciones mostradas en Figura 24, se deben de procesar por cada valoración del usuario en cada clase. Además, estas ecuaciones son propias del modelo de negocio de la aplicación, es decir, no corresponden con el algoritmo original.

Para clarificar la actualización de la base de conocimiento, se incluye el siguiente un ejemplo:

Supongamos que se ha registrado un nuevo usuario en la aplicación con las siguientes características:

- Mujer de 39 años, residente en Madrid.
- Con una valoración de gusto alimenticio: Carne de vacuno 8/10.

Y que en la base de conocimiento tenemos la siguiente información, en base a las etiquetas de sexo, edad y hábitat del usuario:

- Valor Actual Muestral Vacuno: 0,88
- Tamaño Muestral Vacuno: 850

El proceso de actualización a realizar es el siguiente, Figura 25:

$$NVM_{vacuno} = \frac{(VAM_{vacuno} * TM_{vacuno}) + NVI}{TM_{vacuno} + 1}$$

$$NTM_{vacuno} = TM_{vacuno} + 1$$

Figura 25. Ejemplo actualización base de conocimiento vacuno
(Elaboración propia)

Y el resultado sería el siguiente, Figura 26:

$$NVM_{vacuno} = \frac{(0,88 * 850) + 0,8}{850 + 1} = 0,8799$$

$$NTM_{vacuno} = 850 + 1 = 851$$

Figura 26. Ejemplo 2 actualización base de conocimiento vacuno
(Elaboración propia)

4.3. Diseño del sistema

Una vez analizadas las características de la aplicación, en este apartado se define el proceso para la definición de la arquitectura, la definición de los casos de uso y los actores involucrados, la elaboración del modelo físico de datos y la estructura de objetos clase.

4.3.1. Diseño de la arquitectura

En este apartado se describe la arquitectura tecnológica y los IDE de desarrollo que se van a utilizar en la implementación de la aplicación. En la Tabla 24 podemos visualizar el entorno de desarrollo.

Tabla 24. Herramientas y tecnologías

Tipo	Tecnología / Herramienta
Ionic framework	<ul style="list-style-type: none"> Angular: https://cli.angular.io/ TypeScript: https://www.typescriptlang.org/ Nodejs: https://nodejs.org/es/ Ionic: https://ionicframework.com/docs/
Web	<ul style="list-style-type: none"> CSS3 HTML5 JavaScript
Base de datos	<ul style="list-style-type: none"> MySQL: https://dev.mysql.com/
IDE desarrollo	<ul style="list-style-type: none"> Visual Studio Code, para la implementación con Ionic: https://code.visualstudio.com/ MySQL Workbench, para la gestión de BBDD: https://www.mysql.com/products/workbench/
Simulador aplicación	<ul style="list-style-type: none"> Navegador Google Chrome: https://www.google.com/chrome/ Desde dispositivo físico iOS, Ionic DevApp: https://ionicframework.com/docs/appflow/devapp/

(Elaboración propia)

4.3.2. Diseño de casos de uso

Los casos de uso son una técnica dentro de la Ingeniería del Software para el desarrollo de productos software utilizada para captar y representar los requisitos del sistema enfocados en su importancia para los usuarios, ampliando así los términos puramente funcionales. Se puede afirmar que un Caso de Uso es una representación de un requisito funcional centralizado en su interacción con los usuarios o actores del sistema. [15]

Contemplar y diseñar el sistema a desarrollar mediante Casos de Uso tiene importancia en la metodología RUP, no solo por especificación de los requisitos del sistema, sino porque los casos de uso están integrados en las fases de Diseño, Implementación y Pruebas. [14]

Definición de actores del sistema

“Un actor representa un rol desempeñado por agentes externos que interactúan con el sistema” [22], pudiendo ser una persona, máquinas o sistemas. Según el análisis realizado del sistema, centralizando la interacción con la aplicación móvil, se describen los siguientes actores:

- **Usuario particular:** Corresponde con los usuarios consumidores y por tanto el público objetivo de la aplicación móvil. El rol Usuario particular se describe en la Tabla 25.

Tabla 25. Características del actor usuario

ID	ACT. 1
Nombre	Usuario particular
Descripción	Corresponde a cualquier usuario de la aplicación consumidor de los servicios de hostelería ofrecidos.
Responsabilidades	<ul style="list-style-type: none"> Ingreso seguro al sistema. Custodia adecuada de su información personal. Registro, consulta, rectificación y cancelación de su información personal y sus preferencias. Aceptación de las políticas de uso y privacidad.

(Elaboración propia)

- **Usuario restaurante:** Corresponde con los usuarios clientes encargados de gestionar sus restaurantes. El rol Usuario Restaurante se describe en la Tabla 26.

Tabla 26. Características del actor usuario restaurante

ID	ACT. 2
Nombre	Usuario Restaurante
Descripción	Corresponde a cualquier usuario de la aplicación dueño de algún restaurante o establecimiento.
Responsabilidades	<ul style="list-style-type: none"> Ingreso seguro al sistema. Custodia adecuada de su información personal. Registro, consulta, rectificación y cancelación de su información personal y sus preferencias. Aceptación de las políticas de uso y privacidad.

(Elaboración propia)

Casos de uso de alto nivel

Para la especificación de los requisitos del aplicativo se van a realizar los Casos de uso de alto nivel. Para ello se tienen en cuenta las siguientes consideraciones y categorías de datos.

- Los casos de uso están enfocados a las requisitos funcionales de los actores del aplicativo.
- La categoría de **Importancia** tiene como valores posibles: **Alta, Media, Baja**.
- La categoría de **Urgencia** tiene como valores posibles: **No, Inmediata**.

El diagrama de casos de uso completo del sistema se puede visualizar en el Anexo II. Diagrama de casos de uso. Se puede observar que los casos de uso están realizados a partir de dos actores del sistema, Usuario y Usuario Restaurante. También se puede apreciar que hay casos de uso que pueden ser realizados por ambos actores y por supuesto casos de uso que son exclusivos por actor.

A continuación, se desglosan individualizados los casos de usos, comenzando con los casos de uso interactuados por ambos actores.

➤ Caso de uso Login y Registrar

En la Figura 27 se observa el diagrama del caso de uso de Login.

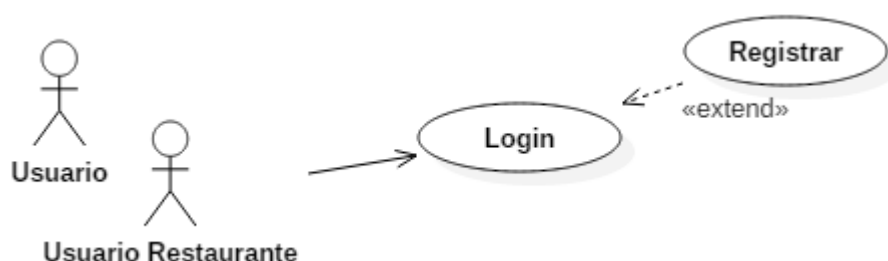


Figura 27. Caso de uso - Login y Registrar

Y en la Tabla 27 se detalla la especificación del caso de uso Login.

Tabla 27. Caso de uso 001 - Login

Caso de Uso	C.U. 001	
Nombre	Login	
Descripción	El actor realiza el proceso de identificación para acceder al sistema.	
Actores	Usuario Particular, Usuario Restaurante	
Precondición	El actor está previamente registrado y dado de alta en el sistema.	
Postcondición	El actor accede al sistema identificado y autenticado correctamente.	
Secuencia normal	Paso	Acción

	1.	El sistema detecta que el actor va a entrar en el sistema y muestra la pantalla de Login.
	2.	El actor cumplimenta el formulario con su usuario y contraseña de acceso al sistema.
	3.	El sistema identifica y autentica al actor, buscando sus credenciales y validando que son correctas.
	4.	El actor accede al sistema y accede al área privada en cuanto a su rol.
Secuencia alternativa	Paso	Acción
	2.	El actor no está registrado en el sistema, puede acceder al registro. C.U. 002
Secuencia error	Paso	Acción
	3.	El sistema detecta que el usuario introducido no existe y lanza un aviso al actor. Se vuelve al Paso 3 de Secuencia normal.
Secuencia error 2	Paso	Acción
	3.	El sistema detecta que la contraseña introducida no es correcta para el usuario y lanza un aviso al actor. Se vuelve al Paso 2 de Secuencia normal.
Importancia	Alta	
Urgencia	Inmediata	
Observaciones	-	

(Elaboración propia)

Y en la Tabla 28 se detalla la especificación del caso de uso Registrar usuario.

Tabla 28. Caso de uso 002 - Registrar

Caso de Uso	C.U. 002	
Nombre	Registrar	
Descripción	El actor realiza el proceso de registro en el sistema, para poder acceder con posterioridad al mismo.	
Actores	Usuario Particular , Usuario Restaurante	
Precondición	El actor no está registrado en el sistema.	
Postcondición	El actor está registrado en el sistema.	
Secuencia normal	Paso	Acción
	1.	El sistema detecta que el actor va a entrar en el sistema y muestra la pantalla de Login.
	2.	El actor, al no tener usuario en el sistema, selecciona la opción de registro de un nuevo usuario.
	3.	El sistema redirige al actor al formulario de alta de un nuevo usuario.

	4.	El actor cumplimenta el formulario de alta del sistema, mediante su información personal, sus preferencias culinarias y sus datos de alergias.
	5.	El sistema valida la información introducida por el actor y guarda los datos para finalizar el proceso.
Secuencia alternativa	-	
Secuencia de error	Paso	Acción
	5.	El sistema detecta que algún campo del formulario no es correcto o es vacío y muestra un mensaje al actor, para que subsane el error. Se vuelve al Paso 4 de Secuencia normal.
Importancia	Alta	
Urgencia	Inmediata	
Observaciones	-	

(Elaboración propia)

➤ **Caso de uso Gestionar información personal**

En la Figura 28 se observa el diagrama del caso de uso de Gestionar información personal.

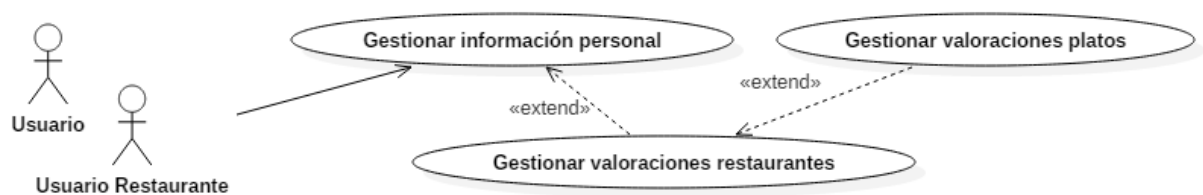


Figura 28. Caso de uso - Gestionar información personal

Y en la Tabla 29, Tabla 30 y Tabla 31 se detalla la especificación de los casos de uso Gestionar información personal.

Tabla 29. Caso de uso 003 - Gestionar información personal

Caso de Uso	C.U. 003	
Nombre	Gestionar información personal	
Descripción	El actor consulta/modifica la su información contenida en el sistema. Además, el actor puede darse de baja en el sistema.	
Actores	Usuario Particular , Usuario Restaurante	
Precondición	El actor ha iniciado sesión en el sistema.	
Postcondición	El actor a consultado/modificado su información personal o se ha dado de baja en el sistema.	
Secuencia normal	Paso	Acción
	1.	El actor accede a su área de información personal.

	2.	El sistema presenta la información personal que tiene almacenada del actor.
	3.	El actor consulta su información personal.
Secuencia alternativa 1	Paso	Acción
	4.	El actor modifica algunos datos de su información personal.
	5.	El sistema valida la información introducida por el actor, persiste los datos y muestra un mensaje al actor del estado de la operación.
Secuencia alternativa 2	Paso	Acción
	4.	El actor selecciona la opción de darse de baja en el sistema.
	5.	El sistema muestra al actor un mensaje de confirmación de la operación de baja en el sistema.
	6.	El actor confirma la operación de baja en el sistema.
	7.	El sistema da de baja al actor, muestra al actor un mensaje de operación realizada correctamente y redirige al actor a la pantalla de login.
Secuencia alternativa 2	Paso	Acción
	4.	El actor accede a la gestión de valoraciones de restaurantes. C.U. 005
Secuencia de error Secuencia alternativa 1	Paso	Acción
	5.	El sistema detecta que algún campo del formulario no es correcto o es vacío y muestra un mensaje al actor, para que subsane el error. Se vuelve al Paso 4 de Secuencia alternativa 1.
Importancia	Media	
Urgencia	No	
Observaciones	-	

(Elaboración propia)

Tabla 30. Caso de uso 004 - Gestionar valoraciones platos

Caso de Uso	C.U. 004	
Nombre	Gestionar valoraciones platos	
Descripción	El actor consulta/modifica la información de valoración de un plato contenida en el sistema. Además, el actor puede darse de baja esa valoración del sistema.	
Actores	Usuario Particular	
Precondición	El actor ha iniciado sesión en el sistema. El actor está gestionando la valoración de un restaurante. C.U. 005	
Postcondición	El actor a consultado/modificado la valoración de un plato o ha dado de baja la valoración del plato en el sistema.	
Secuencia normal	Paso	Acción

	1.	El actor accede a la lista de valoraciones de platos realizada para el restaurante.
	2.	El sistema presenta la lista de las valoraciones de plato del restaurante del actor.
	3.	El actor accede al detalle de una valoración de un plato.
	4.	El sistema presenta la información de la valoración de un plato.
Secuencia alternativa 1	Paso	Acción
	5.	El actor modifica algunos datos de la valoración del plato.
	6.	El sistema valida la información introducida por el actor, persiste los datos, muestra un mensaje al actor del estado de la operación y redirige al actor al detalle de la valoración del restaurante.
Secuencia alternativa 2	Paso	Acción
	5.	El actor selecciona la opción de dar de baja del sistema la valoración del plato.
	6.	El sistema muestra al actor un mensaje de confirmación de la operación de baja en el sistema.
	7.	El actor confirma la operación de baja en el sistema.
	8.	El sistema da de baja la valoración del plato, muestra al actor un mensaje de operación realizada correctamente y redirige al actor al detalle de la valoración de restaurantes.
Secuencia de error Secuencia alternativa 1	Paso	Acción
	6.	El sistema detecta que algún campo del formulario no es correcto o es vacío y muestra un mensaje al actor, para que subsane el error. Se vuelve al Paso 5 de Secuencia alternativa 1.
Importancia	Media	
Urgencia	No	
Observaciones	-	

(Elaboración propia)

Tabla 31. Caso de uso 005 - Gestionar valoraciones restaurantes

Caso de Uso	C.U. 005
Nombre	Gestionar valoraciones restaurantes
Descripción	El actor consulta/modifica la información de valoración de un restaurante contenida en el sistema. Además, el actor puede darse de baja esa valoración del sistema.
Actores	Usuario Particular
Precondición	El actor ha iniciado sesión en el sistema. El actor se encuentra en la gestión de información personal C.U. 003
Postcondición	El actor a consultado/modificado la valoración de un restaurante o ha dado de baja la valoración del restaurante en el sistema.

Secuencia normal	Paso	Acción
	1.	El actor accede a la lista de sus valoraciones a restaurantes.
	2.	El sistema presenta la lista de las valoraciones de restaurantes al actor.
	3.	El actor accede al detalle de una valoración de un restaurante.
Secuencia alternativa 1	Paso	Acción
	5.	El actor modifica algunos datos de la valoración del restaurante.
	6.	El sistema valida la información introducida por el actor, persiste los datos, muestra un mensaje al actor del estado de la operación y redirige al actor al listado de valoraciones de restaurante.
Secuencia alternativa 2	Paso	Acción
	5.	El actor selecciona la opción de dar de baja del sistema la valoración del restaurante.
	6.	El sistema muestra al actor un mensaje de confirmación de la operación de baja en el sistema.
	7.	El actor confirma la operación de baja en el sistema.
Secuencia alternativa 3	Paso	Acción
	5.	El actor accede a la gestión de valoraciones de platos. C.U. 004
Secuencia de error Secuencia alternativa 1	Paso	Acción
	6.	El sistema detecta que algún campo del formulario no es correcto o es vacío y muestra un mensaje al actor, para que subsane el error. Se vuelve al Paso 5 de Secuencia alternativa 1.
Importancia	Media	
Urgencia	No	
Observaciones	-	

(Elaboración propia)

➤ **Caso de uso Enviar sugerencias**

En la Figura 29 se observa el diagrama del caso de uso de Enviar sugerencias.

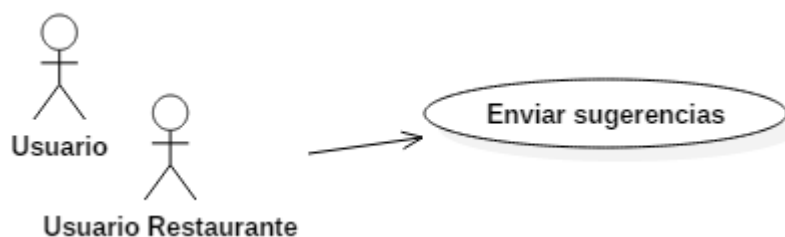


Figura 29. Caso de uso 006 - Enviar sugerencias

Y en la Tabla 32 se detalla la especificación del caso de uso Enviar sugerencias.

Tabla 32. Caso de uso 006 - Enviar sugerencias

Caso de Uso	C.U. 006	
Nombre	Enviar sugerencias	
Descripción	El actor envía sugerencias o feedback con la aplicación, problemas o incidencias que haya tenido al sistema.	
Actores	Usuario Particular , Usuario Restaurante	
Precondición	El actor ha iniciado sesión en el sistema.	
Postcondición	El actor ha enviado su sugerencia al sistema.	
Secuencia normal	Paso	Acción
	1.	El actor accede al apartado de sugerencias y opiniones.
	2.	El sistema presenta al usuario el formulario de sugerencias y opiniones.
	3.	El actor cumplimenta el formulario para el envío de sugerencias y opiniones.
	4.	El sistema valida la información introducida por el actor, persiste los datos y muestra un mensaje al actor del estado de la operación.
Secuencia alternativa	-	
Secuencia de error	Paso	Acción
	4.	El sistema detecta que algún campo del formulario no es correcto o es vacío y muestra un mensaje al actor, para que subsane el error. Se vuelve al Paso 3 de Secuencia normal.
Importancia	Baja	
Urgencia	No	
Observaciones	-	

(Elaboración propia)

➤ Caso de uso Realizar encuestas

En la Figura 30 se observa el diagrama del caso de uso de Realizar encuestas.

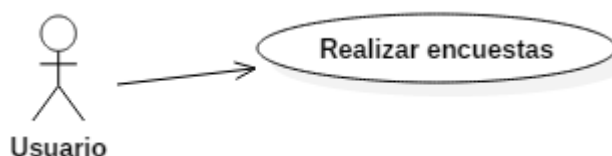


Figura 30. Caso de uso 007 - Realizar encuestas

Y en la Tabla 33 se detalla la especificación del caso de uso Realizar encuestas.

Tabla 33. Caso de uso 007 - Realizar encuestas

Caso de Uso	C.U. 007	
Nombre	Realizar encuestas	
Descripción	El actor realiza el envío de una encuesta en el sistema con el objetivo de la mejora de la base de conocimiento del algoritmo predictivo.	
Actores	Usuario Particular	
Precondición	El actor ha iniciado sesión en el sistema.	
Postcondición	El actor formaliza la encuesta en el sistema.	
Secuencia normal	Paso	Acción
	1.	El actor accede a la opción de realizar encuestas.
	2.	El sistema redirige al usuario al formulario de encuesta.
	3.	El actor cumplimenta el formulario en base a sus preferencias.
	4.	El sistema valida la información introducida por el actor, persiste los datos y muestra un mensaje al actor del estado de la operación.
Secuencia alternativa	-	
Secuencia de error	Paso	Acción
	4.	El sistema detecta que algún campo del formulario no es correcto o es vacío y muestra un mensaje al actor, para que subsane el error. Se vuelve al Paso 3 de Secuencia normal.
Importancia	Baja	
Urgencia	No	
Observaciones	El resultado de estas encuestas puede servir tanto para la retroalimentación de algoritmo predictivo para abrir alguna nueva rama de especialización en el mismo.	

(Elaboración propia)

➤ Casos de uso del actor Usuario con los Restaurantes

En la Figura 31 se observa el diagrama de los casos de uso con los requisitos y acciones que puede realizar el actor Usuario con relación a los Restaurantes.

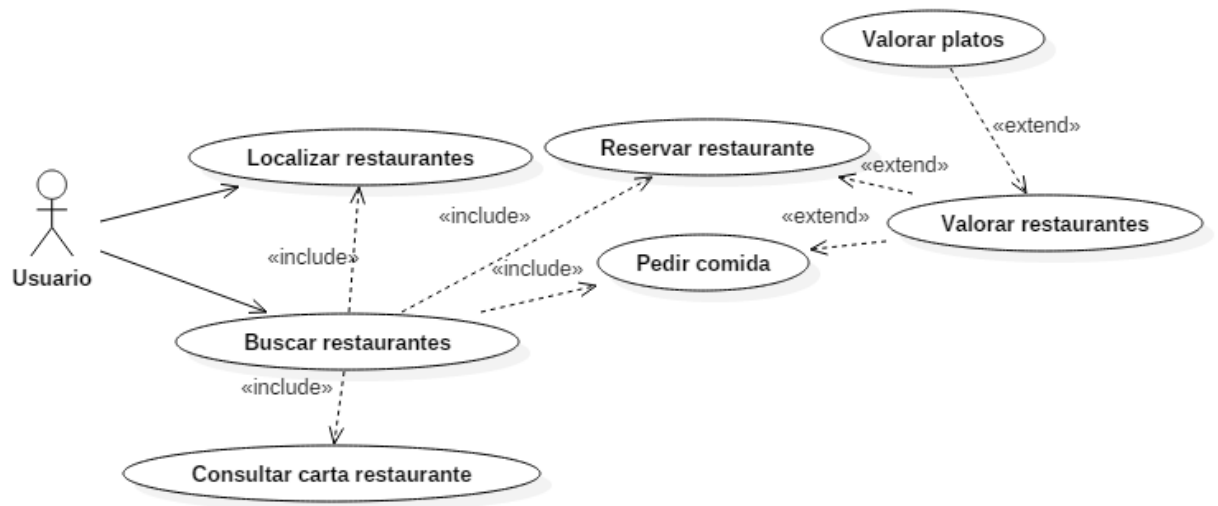


Figura 31. Caso de uso del actor Usuario con los Restaurantes.

En las siguientes Tabla 34, Tabla 35, Tabla 36, Tabla 37, Tabla 38, Tabla 39 y Tabla 40, se detalla la especificación de los casos del actor Usuario con los Restaurantes.

Tabla 34. Caso de uso 008 - Valorar platos

Caso de Uso	C.U. 008	
Nombre	Valorar platos	
Descripción	El actor puntúa su valoración en base a su satisfacción del plato degustado de un restaurante.	
Actores	Usuario Particular	
Precondición	El actor está valorando un restaurante. C.U. 009	
Postcondición	El actor ha valorado un plato degustado de un restaurante.	
Secuencia normal	Paso	Acción
	1.	El actor selecciona de la carta el plato que ha degustado del restaurante.
	2.	El sistema redirige al usuario al formulario de valoración de platos degustados.
	3.	El actor cumplimenta el formulario de valoración de platos degustados.
	4.	El sistema valida la información introducida por el actor, persiste los datos, muestra un mensaje al actor del estado de la operación y redirige al actor a la valoración del restaurante C.U. 009 .
Secuencia alternativa	-	
Secuencia de error	Paso	Acción

	4.	El sistema detecta que algún campo del formulario no es correcto o es vacío y muestra un mensaje al actor, para que subsane el error. Se vuelve al Paso 3 de Secuencia normal.
Importancia	Media	
Urgencia	Inmediata	
Observaciones	-	

(Elaboración propia)

Tabla 35. Caso de uso 009 - Valorar restaurantes

Caso de Uso	C.U. 009	
Nombre	Valorar restaurantes	
Descripción	El actor puntúa y valora un restaurante en base a su satisfacción del servicio prestado por el mismo, ya sea en degustación en el propio restaurante o mediante comida a domicilio.	
Actores	Usuario Particular	
Precondición	El actor ha iniciado sesión en el sistema. C.U. 010 o C.U. 011	
Postcondición	El actor ha valorado un restaurante.	
Secuencia normal	Paso	Acción
	1.	El sistema muestra al actor un mensaje para que valore el restaurante donde ha estado/ha pedido comida recientemente.
	2.	El actor accede a la valoración del restaurante.
	3.	El sistema redirige al actor al formulario de valoración de restaurantes.
	4.	El actor cumplimenta el formulario de valoración de restaurantes.
	5.	El sistema valida la información introducida por el actor, persiste los datos y redirige al actor al formulario de valoración de platos.
	6.	Bucle: El actor valora los platos degustados. C.U. 008
Secuencia alternativa	-	
Secuencia de error	Paso	Acción
	5.	El sistema detecta que algún campo del formulario no es correcto o es vacío y muestra un mensaje al actor, para que subsane el error. Se vuelve al Paso 4 de Secuencia normal.
Importancia	Media	
Urgencia	Inmediata	
Observaciones	-	

(Elaboración propia)

Tabla 36. Caso de uso 010 - Reservar restaurante

Caso de Uso	C.U. 010
Nombre	Reservar restaurante

Descripción	El actor reserva mesa en un restaurante.	
Actores	Usuario Particular	
Precondición	El actor se encuentra en el detalle de un restaurante.	
Postcondición	El actor ha reservado mesa en un restaurante.	
Secuencia normal	Paso	Acción
	1.	El actor selecciona la opción de reservar mesa en el restaurante.
	2.	El sistema redirige al usuario al formulario de reserva de mesa precargando en el formulario los días y horas disponibles del restaurante.
	3.	El actor cumplimenta el formulario de reserva de mesa.
	4.	El sistema valida la información introducida por el actor, persiste los datos, muestra al actor un mensaje de operación correcta y redirige al actor al detalle del restaurante.
	5.	El sistema gestiona la reserva con el restaurante.
Secuencia alternativa	-	
Secuencia de error	Paso	Acción
	4.	El sistema detecta que algún campo del formulario no es correcto o es vacío y muestra un mensaje al actor, para que subsane el error. Se vuelve al Paso 3 de Secuencia normal.
Importancia	Alta	
Urgencia	Inmediata	
Observaciones	-	

(Elaboración propia)

Tabla 37. Caso de uso 011 - Pedir comida

Caso de Uso	C.U. 011	
Nombre	Pedir comida	
Descripción	El actor pide comida a domicilio de un restaurante.	
Actores	Usuario Particular	
Precondición	El actor está en el detalle del restaurante.	
Postcondición	El actor ha pedido comida a domicilio.	
Secuencia normal	Paso	Acción
	1.	El actor selecciona la opción de pedir comida a domicilio.
	2.	El sistema comienza el proceso de pedido redirigiendo al actor al formulario a cumplimentar y con la lista de platos disponibles de restaurante.
	3.	El actor selecciona todos los platos a pedir a domicilio y cumplimenta el formulario.

	4.	El sistema valida la información introducida por el actor, y comienza el proceso de pago mediante TPV virtual.
	5.	El actor realiza el proceso de pago mediante TPV virtual.
	6.	El sistema valida el pedido, muestra al usuario un mensaje de operación correcta con los dato de pedido y envía el pedido al restaurante.
Secuencia alternativa	-	
Secuencia de error 1	Paso	Acción
	4.	El sistema detecta que algún campo del formulario no es correcto o es vacío y muestra un mensaje al actor, para que subsane el error. Se vuelve al Paso 3 de Secuencia normal.
Secuencia de error 1	Paso	Acción
	6.	El sistema no puede validar el pago correcto para finalizar el pedido, muestra un mensaje de error de pago al actor. Se vuelve al Paso 3 de Secuencia normal.
Importancia	Alta	
Urgencia	Inmediata	
Observaciones	Para realizar pedidos a domicilio es necesario el pago a través de TPV virtual.	

(Elaboración propia)

Tabla 38. Caso de uso 012 - Consultar carta restaurante

Caso de Uso	C.U. 012	
Nombre	Consultar carta restaurante	
Descripción	Para cada restaurante el actor puede consultar la información y carta de platos del restaurante.	
Actores	Usuario Particular	
Precondición	El actor ha iniciado sesión en el sistema. El actor está en el detalle de un restaurante.	
Postcondición	El actor ha consultado la información de la carta y platos del restaurante.	
Secuencia normal	Paso	Acción
	1.	El actor selecciona la opción de visualizar la carta del restaurante.
	2.	El sistema muestra un listado con la información de los platos del restaurante.
Secuencia alternativa	-	
Secuencia de error	-	

Importancia	Media
Urgencia	Inmediata
Observaciones	-

(Elaboración propia)

Tabla 39. Caso de uso 013 - Localizar restaurantes

Caso de Uso	C.U. 013	
Nombre	Localizar restaurantes	
Descripción	El actor puede localizar restaurantes en un mapa mediante geolocalización.	
Actores	Usuario Particular	
Precondición	El actor ha iniciado sesión en el sistema.	
Postcondición	El actor ha visualizado y localizado restaurantes en un mapa.	
Secuencia normal	Paso	Acción
	1.	El actor accede a la opción de localización de restaurantes.
	2.	El sistema muestra un mapa con geolocalización, mostrando los restaurantes cercanos al actor.
	3.	El usuario puede acceder al detalle de los restaurantes que aparecen en el mapa.
Secuencia alternativa	-	
Secuencia de error	-	
Importancia	Baja	
Urgencia	No	
Observaciones	-	

(Elaboración propia)

Tabla 40. Caso de uso 014 - Buscar restaurantes

Caso de Uso	C.U. 014	
Nombre	Buscar restaurantes	
Descripción	El actor visualiza restaurantes disponibles.	
Actores	Usuario Particular	
Precondición	El actor ha iniciado sesión en el sistema.	
Postcondición	El actor ha visualizado restaurante disponibles.	
Secuencia normal	Paso	Acción
	1.	El actor accede a la opción de buscar restaurantes.
	2.	El sistema redirige al usuario a la pantalla de listado de expedientes mostrado por defecto los de la localidad del actor.
	3.	El actor visualiza los restaurantes disponibles.

Secuencia alternativa 1	Paso	Acción
	4.	El actor puede Localizar restaurantes en un mapa mediante geolocalización. C.U. 013.
Secuencia alternativa 2	Paso	Acción
	4.	El actor puede Reservar mesa en un restaurante. C.U. 010.
Secuencia alternativa 3	Paso	Acción
	4.	El actor puede Pedir comida seleccionando un restaurante. C.U. 011.
Secuencia alternativa 4	Paso	Acción
	4.	El actor puede Consultar la carta de un restaurante. C.U. 012.
Secuencia de error	-	
Importancia	Alta	
Urgencia	Inmediata	
Observaciones	-	

(Elaboración propia)

➤ Casos de uso Alta de restaurante

En la Figura 32 se observa el diagrama de los casos de uso para el alta de un nuevo restaurante en el sistema por parte del actor Usuario Restaurante.

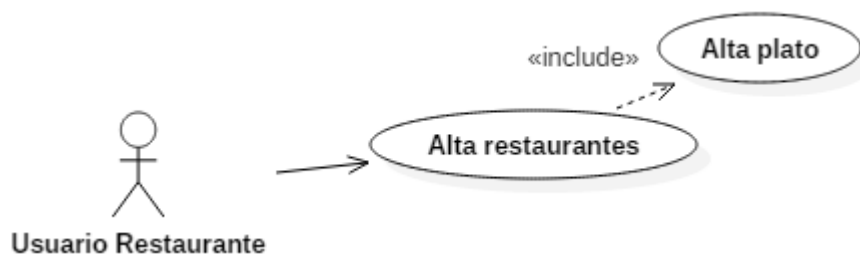


Figura 32. Caso de uso Alta de restaurante

En las siguientes Tabla 41 y Tabla 42 se detalla la especificación de los casos del actor Usuario Restaurante para el alta de restaurantes.

Tabla 41. Caso de uso 0015 - Alta plato

Caso de Uso	C.U. 0015	
Nombre	Alta plato	
Descripción	El actor da de alta un nuevo plato a uno de sus restaurantes.	
Actores	Usuario Restaurante	
Precondición	El actor está en el detalle de un restaurante.	
Postcondición	El actor ha dado de alta un nuevo plato al restaurante.	
Secuencia normal	Paso	Acción
	1.	El actor selecciona la opción de alta de un nuevo plato.

	2.	El sistema redirige al actor al formulario de alta de un nuevo plato para el restaurante.
	3.	El actor cumplimenta el formulario de alta del plato, con la información del plato, información sobre los ingredientes y sobre el contenido de alérgenos.
	4.	El sistema valida la información introducida por el actor, persiste los datos, muestra un mensaje del resultado de la operación y redirige al actor al detalle del restaurante.
Secuencia alternativa	-	
Secuencia de error	Paso	Acción
	4.	El sistema detecta que algún campo del formulario no es correcto o es vacío y muestra un mensaje al actor, para que subsane el error. Se vuelve al Paso 3 de Secuencia normal.
Importancia	Alta	
Urgencia	Inmediata	
Observaciones	-	

(Elaboración propia)

Tabla 42. Caso de uso 0016 - Alta restaurantes

Caso de Uso	C.U. 0016	
Nombre	Alta restaurantes	
Descripción	El actor da de alta en el sistema un nuevo restaurante.	
Actores	Usuario Restaurante	
Precondición	El actor ha iniciado sesión en el sistema.	
Postcondición	Se ha dado de alta en el sistema un nuevo restaurante asociado al actor.	
Secuencia normal	Paso	Acción
	1.	El actor selecciona la opción de alta de un nuevo restaurante.
	2.	El sistema redirige al actor al formulario de alta de un nuevo restaurante.
	3.	El actor cumplimenta el formulario de alta de nuevo restaurante con la información sobre el restaurante y su ubicación.
	4.	El sistema valida la información introducida por el actor, persiste los datos, muestra un mensaje al actor del estado de la operación.
	5.	Bucle: El sistema redirige al usuario al Alta de platos. C.U 0015
Secuencia alternativa	-	
Secuencia de error	Paso	Acción

	4.	El sistema detecta que algún campo del formulario no es correcto o es vacío y muestra un mensaje al actor, para que subsane el error. Se vuelve al Paso 3 de Secuencia normal.
Importancia	Alta	
Urgencia	Inmediata	
Observaciones	-	

(Elaboración propia)

➤ **Casos de uso Gestionar de restaurante**

En la Figura 33 se observa el diagrama de los casos de uso para el proceso de gestión de restaurantes en el sistema por parte del actor Usuario Restaurante.

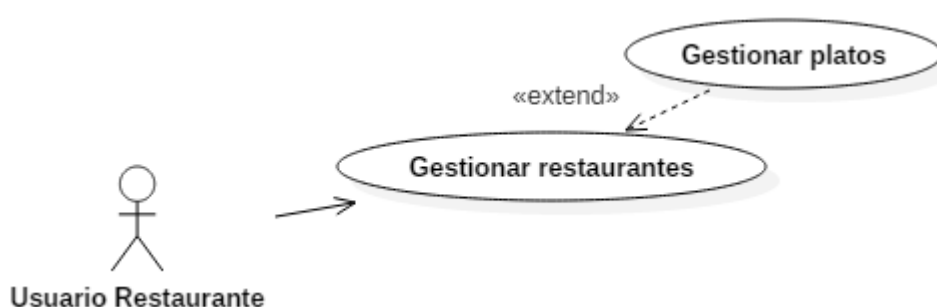


Figura 33. Caso de uso Gestión de restaurante

En las siguientes Tabla 43 y Tabla 44 se detalla la especificación de los casos del actor Usuario Restaurante para la gestión de restaurantes.

Tabla 43. Caso de uso 0017 - Gestionar platos

Caso de Uso	C.U. 0017	
Nombre	Gestionar platos	
Descripción	El actor consulta/modifica la información de los platos que aparecen en sus restaurantes en el sistema. El actor puede dar de baja un plato en el sistema.	
Actores	Usuario Restaurante	
Precondición	El actor ha iniciado sesión en el sistema. El actor se encuentra en el detalle de un restaurante C.U. 0018 .	
Postcondición	El actor ha consultado la información del plato de su restaurante o ha dado de baja el plato del restaurante.	
Secuencia normal	Paso	Acción
	1.	El actor accede al listado de platos del restaurante.
	2.	El sistema presenta la lista de platos del restaurante.
	3.	El actor accede y consulta la información de un plato.

Secuencia alternativa 1	Paso	Acción
	4.	El actor modifica algunos datos del plato.
	5.	El sistema valida la información introducida por el actor, persiste los datos, muestra un mensaje al actor del estado de la operación y redirige al actor al listado de platos.
Secuencia alternativa 2	Paso	Acción
	4.	El actor selecciona la opción de dar de baja el plato en el sistema.
	5.	El sistema muestra al actor un mensaje de confirmación de la operación de baja en el sistema.
	6.	El actor confirma la operación de baja en el sistema.
Secuencia de error a la Secuencia alternativa 1	Paso	Acción
	5.	El sistema detecta que algún campo del formulario no es correcto o es vacío y muestra un mensaje al actor, para que subsane el error. Se vuelve al Paso 4 de Secuencia alternativa 1.
Importancia	Media	
Urgencia	No	
Observaciones	-	

(Elaboración propia)

Tabla 44. Caso de uso 0018 - Gestionar restaurantes

Caso de Uso	C.U. 0018	
Nombre	Gestionar restaurantes	
Descripción	El actor consulta/modifica la información de un restaurantes en el sistema. El actor puede dar de baja el restaurante en el sistema.	
Actores	Usuario Restaurante	
Precondición	El actor ha iniciado sesión en el sistema.	
Postcondición	El actor ha consultado la información del restaurante o ha dado de baja el restaurante en el sistema.	
Secuencia normal	Paso	Acción
	1.	El actor accede a la opción del listado de sus restaurantes.
	2.	El sistema presenta la información de los restaurantes del actor.
Secuencia alternativa 1	Paso	Acción
	4.	El actor modifica algunos datos del restaurante.
	5.	El sistema valida la información introducida por el actor, persiste los datos y muestra un mensaje al actor del estado de la operación.

Secuencia alternativa 2	Paso	Acción
	4.	El actor selecciona la opción de dar de baja el restaurante en el sistema.
	5.	El sistema muestra al actor un mensaje de confirmación de la operación de baja en el sistema.
	6.	El actor confirma la operación de baja en el sistema.
	7.	El sistema da de baja el restaurante, muestra al actor un mensaje de operación realizada correctamente y redirige al actor al listado de restaurantes.
Secuencia alternativa 3	Paso	Acción
	4.	Bucle: El actor Gestiona platos del restaurante. C.U. 017.
Secuencia de error Secuencia alternativa 1	Paso	Acción
	5.	El sistema detecta que algún campo del formulario no es correcto o es vacío y muestra un mensaje al actor, para que subsane el error. Se vuelve al Paso 4 de Secuencia alternativa 1.
Importancia	Media	
Urgencia	No	
Observaciones	-	

(Elaboración propia)

4.3.3. Modelo entidad-relación

Denominado E/R por sus siglas, es una herramienta utilizada para el modelado de datos de un sistema mediante la representación de las entidades más relevantes, sus propiedades e interrelaciones. [23]

En la Figura 34 se ha diseñado el diagrama E/R para los tipos de usuario o roles de usuario que gestiona el aplicativo.

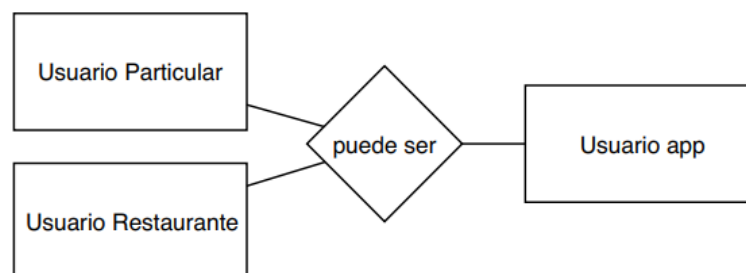


Figura 34. E/R Tipos de usuario

En la Figura 35 se define el modelo para el tipo de usuario restaurante. Estos usuarios podrán gestionar sus restaurantes y la carta de cada uno.

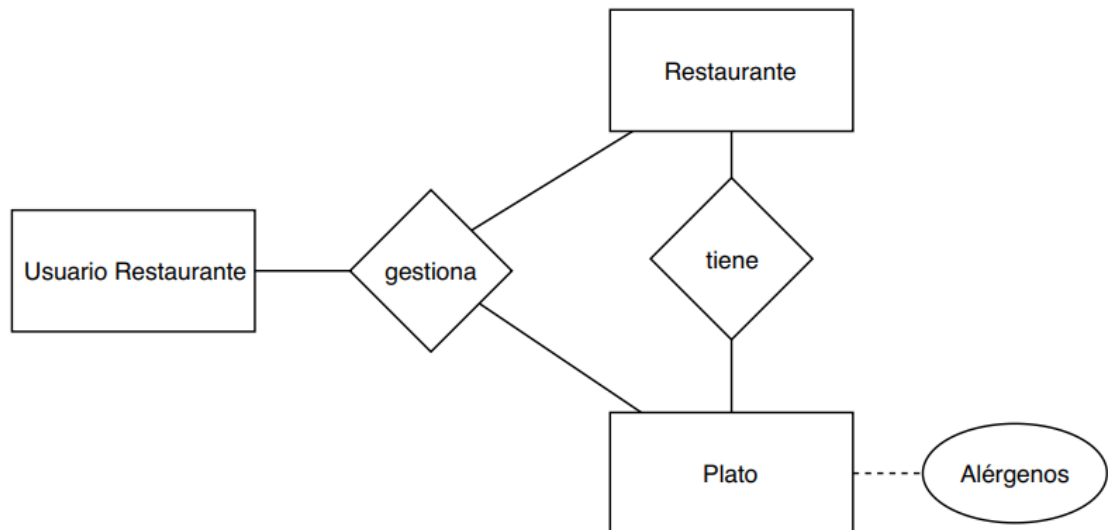


Figura 35. E/R Usuario Restaurante

Por último, en la Figura 36 se define el modelo relacional asociado a los usuarios particulares de la aplicación. Estos usuarios pueden consultar información de los restaurantes y su carta, pueden valorar tanto los restaurantes como cada plato y, además, pueden realizar pedidos a domicilio o reserva de mesa en restaurantes.

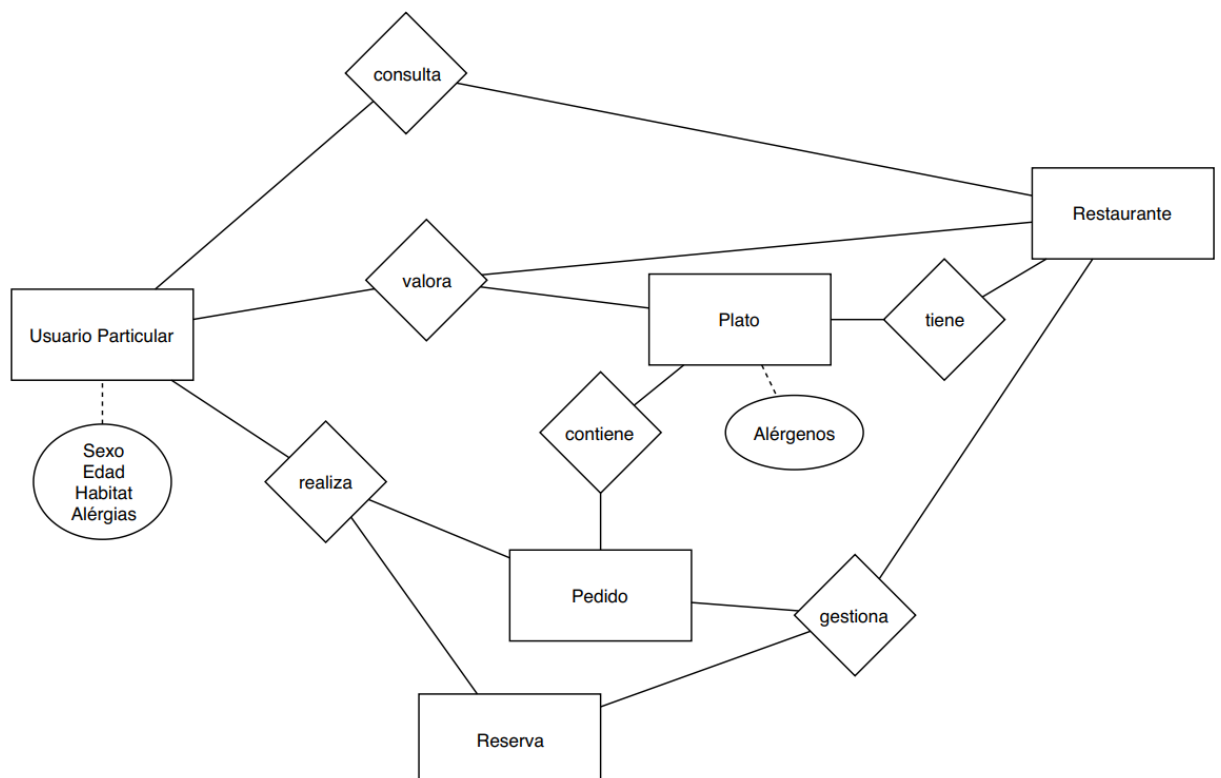


Figura 36. E/R Usuario Particular

4.3.4. Diseño físico de datos

En el Anexo III. Modelo relacional de base de datos, se encuentra el modelo relacional de la base de datos utilizado en el sistema donde se detallan las tablas, campos y relaciones entre tablas. A continuación, se describe cada una de las tablas del sistema.

La mayoría de las tablas tiene en común una serie de campos de auditoría comunes. Las únicas tablas que no tiene esta información son las tablas maestras, ya que contienen datos constantes y las tablas de relaciones. Los campos de auditoría se desglosan en la Tabla 45.

Tabla 45. Descripción Campos de auditoría

Nombre	Campos de auditoría	
Descripción	Son los campos comunes a la mayoría de las tablas para mantener un control de alta y baja de los registros.	
Campo	Tipo	Descripción
fecha_alta	DATETIME	Fecha de alta del registro en el sistema.
fecha_baja	DATETIME	Fecha de baja del registro en el sistema.

(Elaboración propia)

Las tablas maestras corresponden con las siguientes Tabla 46, Tabla 47, Tabla 48, Tabla 49, Tabla 50 y Tabla 51:

Tabla 46. Descripción tabla Alérgenos

Nombre	Alérgenos	
Descripción	Contiene la información de los alérgenos utilizados en el sistema.	
Campo	Tipo	Descripción
id	UUID	Primary key.
nombre	VARCHAR(100)	Nombre del alérgeno.
descripcion	VARCHAR(500)	Descripción del alérgeno.

(Elaboración propia)

Tabla 47. Descripción tabla Edades

Nombre	Edades	
Descripción	Contiene la información de los rangos de edades utilizados por el sistema.	
Campo	Tipo	Descripción
id	UUID	Primary key.
nombre	VARCHAR(100)	Nombre del intervalo de edades.
edad_minima	INT(11)	Edad mínima del intervalo.
edad_maxima	INT(11)	Edad máxima del intervalo.

(Elaboración propia)

Tabla 48. Descripción tabla Grupos ingredientes

Nombre	Grupos ingredientes	
Descripción	Contiene la información de los grupos de ingredientes principales utilizados por el sistema.	
Campo	Tipo	Descripción
id	UUID	Primary key.
nombre	VARCHAR(100)	Nombre del grupo de ingrediente.
descripcion	VARCHAR(200)	Descripción del grupo de ingrediente.

(Elaboración propia)

Tabla 49. Descripción tabla Grupos usuarios

Nombre	Grupos usuarios	
Descripción	Contiene la información de los grupos de los roles a los que puede pertenecer un usuario el sistema.	
Campo	Tipo	Descripción
id	UUID	Primary key.
nombre	VARCHAR(100)	Nombre del grupo de usuario.
descripcion	VARCHAR(200)	Descripción del grupo de usuario.

(Elaboración propia)

Tabla 50. Descripción tabla Habitat

Nombre	Habitat	
Descripción	Contiene la información de los rango de habitat utilizados por el sistema.	
Campo	Tipo	Descripción
id	UUID	Primary key.
nombre	VARCHAR(100)	Nombre del intervalo de población.
poblacion_minima	INT(11)	Población mínima del intervalo.
poblacion_maxima	INT(11)	Población máxima del intervalo.

(Elaboración propia)

Tabla 51. Descripción tabla Sexo

Nombre	Sexo	
Descripción	Contiene la información de sexos utilizados por el sistema.	
Campo	Tipo	Descripción
id	UUID	Primary key.
nombre	VARCHAR(45)	Nombre del sexo.

(Elaboración propia)

El resto de las tablas se describen a continuación, Tabla 52, Tabla 53, Tabla 54, Tabla 55, Tabla 56, Tabla 57, Tabla 58, Tabla 59, Tabla 60, Tabla 61, Tabla 62, Tabla 63, Tabla 64, Tabla 65, Tabla 66, Tabla 67, Tabla 68, Tabla 69 y Tabla 70:

Tabla 52. Descripción tabla Alergenos_x_plato

Nombre	Alergenos_x_plato	
Descripción	Contiene la información relacional del contenido de ingredientes alérgenos por plato.	
Campo	Tipo	Descripción
id	UUID	Primary key.
fk_id_alergeno	INT(11)	Foreign key hacia la tabla alergenosen.
fk_id_plato	INT(11)	Foreign key hacia la tabla platos_x_restaurantes.

(Elaboración propia)

Tabla 53. Descripción tabla Alergias_x_usuario

Nombre	Alergias_x_usuario	
Descripción	Contiene la información relacional de las alergias por usuario.	
Campo	Tipo	Descripción
id	UUID	Primary key.
fk_id_alergeno	INT(11)	Foreign key hacia la tabla alergenosen.
fk_id_usuario	INT(11)	Foreign key hacia la tabla usuarios.

(Elaboración propia)

Tabla 54. Descripción tabla Clasificaciones_base_usuarios

Nombre	Clasificaciones_base_usuarios	
Descripción	Contiene la información relacional de la clasificación de los usuarios en base al sexo, edad y habitat.	
Campo	Tipo	Descripción
id	UUID	Primary key.
fk_id_sexo	INT(11)	Foreign key hacia la tabla sexos.
fk_id_edad	INT(11)	Foreign key hacia la tabla edades.
fk_id_habitat	INT(11)	Foreign key hacia la tabla habitat.

(Elaboración propia)

Tabla 55. Descripción tabla Factores_base_ingrediente_x_edad

Nombre	Factores_base_ingrediente_x_edad	
Descripción	Contiene la información muestral por rango de ingrediente y edad.	
Campo	Tipo	Descripción
id	UUID	Primary key.
fk_id_ingrediente	INT(11)	Foreign key hacia la tabla ingredientes.
fk_id_edad	INT(11)	Foreign key hacia la tabla edades.
porcentaje_base	INT(11)	Valor probabilístico de la muestra.
numero_muestra	INT(11)	Tamaño actual muestral.

(Elaboración propia)

Tabla 56. Descripción tabla Factores_base_ingrediente_x_habitat

Nombre	Factores_base_ingrediente_x_habitat	
Descripción	Contiene la información muestral por rango de ingrediente y habitat.	
Campo	Tipo	Descripción
id	UUID	Primary key.
fk_id_ingrediente	INT(11)	Foreign key hacia la tabla ingredientes.
fk_id_habitat	INT(11)	Foreign key hacia la tabla habitat.
porcentaje_base	INT(11)	Valor probabilístico de la muestra.
numero_muestra	INT(11)	Tamaño actual muestral.

(Elaboración propia)

Tabla 57. Descripción tabla Factores_base_ingrediente_xsexo

Nombre	Factores_base_ingrediente_xsexo	
Descripción	Contiene la información muestral por rango de ingrediente y sexo.	
Campo	Tipo	Descripción
id	UUID	Primary key.
fk_id_ingrediente	INT(11)	Foreign key hacia la tabla ingredientes.
fk_id_sexo	INT(11)	Foreign key hacia la tabla sexo.
porcentaje_base	INT(11)	Valor probabilístico de la muestra.
numero_muestra	INT(11)	Tamaño actual muestral.

(Elaboración propia)

Tabla 58. Descripción tabla Ingredientes

Nombre	Ingredientes	
Descripción	Contiene la información de los ingredientes que maneja el sistema.	
Campo	Tipo	Descripción
id	UUID	Primary key.
nombre	VARCHAR(100)	Nombre del ingrediente.
descripcion	VARCHAR(200)	Descripción del ingrediente.
fk_id_grupo_ingrediente	INT(11)	Foreign key hacia la tabla grupo_ingrediente.
factor_consumo	DECIMAL(4,2)	Factor de consumo del ingrediente dentro del grupo.

(Elaboración propia)

Tabla 59. Descripción tabla Ingredientes_x_platos

Nombre	Ingredientes_x_platos	
Descripción	Contiene la información relacional de los ingredientes por plato.	
Campo	Tipo	Descripción
id	UUID	Primary key.
fk_id_plato	INT(11)	Foreign key hacia la tabla platos_x_restaurantes.

fk_id_ingredient	INT(11)	Foreign key hacia la tabla ingredientes.
------------------	---------	--

(Elaboración propia)

Tabla 60. Descripción tabla Platos_x_restaurantes

Nombre	Platos_x_restaurantes	
Descripción	Contiene la información de los platos que contiene un restaurante.	
Campo	Tipo	Descripción
id	UUID	Primary key.
nombre	VARCHAR(100)	Nombre del plato.
descripcion	VARCHAR(200)	Descripción del plato.
fk_id_restaurante	INT(11)	Foreign key hacia la tabla restaurantes.
motivo_baja	DECIMAL(4,2)	El motivo de la baja del plato.

(Elaboración propia)

Tabla 61. Descripción tabla Puntuaciones_base_usuarios

Nombre	Puntuaciones_base_usuarios	
Descripción	Contiene la información la ponderación de cada usuario con los ingredientes y en función de sus características.	
Campo	Tipo	Descripción
id	UUID	Primary key.
fk_id_clasificacion_base	INT(11)	Foreign key hacia la tabla clasificaciones_base_usuarios.
fk_id_ingredient	INT(11)	Foreign key hacia la tabla ingredientes.
puntuacion	DECIMAL(7,5)	Puntuación de la ponderación del usuario.

(Elaboración propia)

Tabla 62. Descripción tabla Puntuaciones_usuarios_x_ingredientes

Nombre	Puntuaciones_usuarios_x_ingredientes	
Descripción	Contiene la información las puntuaciones de los usuarios a los ingredientes.	
Campo	Tipo	Descripción
id	UUID	Primary key.
fk_id_usuario	INT(11)	Foreign key hacia la tabla usuarios.
fk_id_ingredient	INT(11)	Foreign key hacia la tabla ingredientes.
puntuacion	INT(11)	Puntuación del usuario al ingrediente.

(Elaboración propia)

Tabla 63. Descripción tabla Puntuaciones_usuarios_x_plato

Nombre	Puntuaciones_usuarios_x_plato	
Descripción	Contiene la información las puntuaciones de los usuarios a los platos.	
Campo	Tipo	Descripción
id	UUID	Primary key.

fk_id_usuario	INT(11)	Foreign key hacia la tabla usuarios.
fk_id_plato	INT(11)	Foreign key hacia la tabla plato_x_restaurantes.
puntuacion	INT(11)	Puntuación del usuario al plato.

(Elaboración propia)

Tabla 64. Descripción tabla Puntuaciones_usuarios_x_restaurante

Nombre	Puntuaciones_usuarios_x_restaurante	
Descripción	Contiene la información las puntuaciones de los usuarios a los restaurantes.	
Campo	Tipo	Descripción
id	UUID	Primary key.
fk_id_usuario	INT(11)	Foreign key hacia la tabla usuarios.
fk_id_restaurante	INT(11)	Foreign key hacia la tabla restaurantes.
puntuacion	INT(11)	Puntuación del usuario al restaurante .

(Elaboración propia)

Tabla 65. Descripción tabla Restaurantes

Nombre	Restaurantes	
Descripción	Contiene la información de los restaurantes que gestiona el sistema.	
Campo	Tipo	Descripción
id	UUID	Primary key.
nombre	VARCHAR(100)	Nombre del restaurante.
descripcion	VARCHAR(200)	Descripción del restaurante.
fk_id_ubicacion	INT(11)	Foreign key hacia la tabla ubicaciones.
fk_id_usuario_alta	INT(11)	Foreign key hacia la tabla usuarios.
motivo_baja	DECIMAL(4,2)	El motivo de la baja del restaurante.

(Elaboración propia)

Tabla 66. Descripción tabla Sugerencias_x_usuario

Nombre	Sugerencias_x_usuario	
Descripción	Contiene la información las sugerencias, opiniones y aportaciones de los usuarios a la aplicación.	
Campo	Tipo	Descripción
id	UUID	Primary key.
fk_id_usuario	INT(11)	Foreign key hacia la tabla usuarios.
asunto	VARCHAR(100)	Asunto o motivo de la sugerencia.
comentario	VARCHAR(500)	Descripción o contenido de la sugerencia.

(Elaboración propia)

Tabla 67. Descripción tabla Ubicaciones

Nombre	Ubicaciones
--------	-------------

Descripción	Contiene la información de las ubicaciones de los restaurantes.	
Campo	Tipo	Descripción
id	UUID	Primary key.
latitud	DECIMAL(10,8)	Latitud de la ubicación.
longitud	DECIMAL(11,8)	Longitud de la ubicación.

(Elaboración propia)

Tabla 68. Descripción tabla Usuarios

Nombre	Usuarios	
Descripción	Contiene la información de los usuarios de la aplicación.	
Campo	Tipo	Descripción
id	UUID	Primary key.
nombre	VARCHAR(100)	Nombre del usuario.
nick	VARCHAR(50)	Nick o alias del usuario.
contraseña	CHAR(128)	Contraseña cifrada del usuario.
fk_id_rol	INT(11)	Foreign key hacia la tabla grupos_usuarios.
edad	INT(11)	Edad del usuario.
fk_id_edad	INT(11)	Foreign key hacia la tabla edades.
fk_sexo	INT(11)	Foreign key hacia la tabla sexos.
fk_habitat	INT(11)	Foreign key hacia la tabla habitat.
domicilio	VARCHAR(300)	Domicilio del usuario.

(Elaboración propia)

Tabla 69. Descripción tabla Valoraciones_usuarios_x_platos

Nombre	Valoraciones_usuarios_x_platos	
Descripción	Contiene la información de las valoraciones de los usuarios a los platos.	
Campo	Tipo	Descripción
id	UUID	Primary key.
fk_id_usuario	INT(11)	Foreign key hacia la tabla usuarios.
fk_id_plato	INT(11)	Foreign key hacia la tabla platos_x_restaurantes.
presentacion	INT(1)	Valoración de 1 a 5 de la presentación del plato.
atractivo	INT(1)	Valoración de 1 a 5 del atractivo del plato.
tipicidad	INT(1)	Valoración de 1 a 5 de la tipicidad del plato.
materia_prima	INT(1)	Valoración de 1 a 5 de la materia prima del plato.
texturas_coccion	INT(1)	Valoración de 1 a 5 de las texturas y al punto de cocción del plato.
aromas	INT(1)	Valoración de 1 a 5 de los aromas del plato.

sabor	INT(1)	Valoración de 1 a 5 del sabor del plato.
-------	--------	--

(Elaboración propia)

Tabla 70. Descripción tabla Valoraciones_usuarios_x_restaurantes

Nombre	Valoraciones_usuarios_x_restaurantes	
Descripción	Contiene la información de las valoraciones de los usuarios a los restaurantes.	
Campo	Tipo	Descripción
id	UUID	Primary key.
fk_id_usuario	INT(11)	Foreign key hacia la tabla usuarios.
fk_id_restaurante	INT(11)	Foreign key hacia la tabla restaurantes.
comida	INT(1)	Valoración de 1 a 5 a la comida.
servicio	INT(1)	Valoración de 1 a 5 al servicio.
limpieza	INT(1)	Valoración de 1 a 5 a la limpieza.
ambiente	INT(1)	Valoración de 1 a 5 al ambiente.
precio	INT(1)	Valoración de 1 a 5 al precio.

(Elaboración propia)

4.3.5. Diseño de clases

En el Anexo IV. Diagrama de clases se incluye el diagrama de clases completo del sistema. A continuación, se definen las características de diseño llevadas a cabo en su realización.

Para la normalización se han tenido las siguientes características:

- Las **Clases** empiezan con letra **mayúscula**.
- Los **Atributos** de las clases empiezan con letra **minúscula**.
- Las clases y los atributos **solo** contienen **letras**.

Para la simbología del diagrama se han tenido las siguientes consideraciones:

- Los atributos y métodos **privados** empiezan con -
- Los atributos y métodos **públicos** empiezan con +
- Los atributos **únicos** llevan la etiqueta **{unique}**

La correspondencia de datos entre el lenguaje de BBDD y las Clases es la siguiente Tabla 71:

Tabla 71. Correspondencia de Tipos BBDD - Lógica de Negocio

Tipo dato BBDD	Tipo dato Lógica de Negocio
----------------	-----------------------------

INT	Integer
VARCHAR	String
DATETIME	Date
DOUBLE	Double
DECIMAL	Decimal

(Elaboración propia)

4.4. Desarrollo

En este apartado se presenta el resultado del desarrollo e implementación realizado para la aplicación *TuPaladar*. Para ello se incluye y se describen cada una de las pantallas de las que se componen la aplicación, y una evaluación del sistema en base a una batería de pruebas, integrales aplicadas a cada requerimientos funcional, y de usabilidad realizada por varios usuarios finales para la detección de problemas y errores derivados en esta faceta.

4.4.1. Aplicación

En este apartado se describen las funcionalidades más relevantes de la aplicación *TuPaladar* en base al estado final del desarrollo.

En la aplicación se distinguen dos tipos de acceso, usuario particular o usuario restaurante. Los usuarios particulares son las personas finales a las que van orientados los servicios de comidas que se ofrece. Los usuarios restaurantes son particulares, dueños de restaurantes, que quieren dar a conocer sus servicios mediante *TuPaladar*.

La estructura de la aplicación para los usuarios particulares está estructurada en 3 pestañas. La primera corresponde con la pantalla inicial, donde se visualiza las novedades en la aplicación, la segunda corresponde con el listado de restaurantes y la última es un acceso rápido a las valoraciones realizadas por el usuario. Además, la aplicación se complementa con un menú lateral con todas las opciones disponibles, como se puede observar en la Figura 42.

La primera funcionalidad que destacar es el acceso a la aplicación. Para ello, es necesario estar previamente registrado. En las Figura 40 y Figura 41 podemos ver las pantallas diseñadas para el login de los usuarios y para su registro. De este registro, es muy importante tres bloques de información. El primero tiene que ver con el sexo, edad y domicilio del usuario, el segundo corresponde con sus gustos culinarios en los 8 ingredientes propuestos, y el tercero sus alergias.

Este triángulo de información, y una vez que el usuario se ha registrado en el sistema, va a permitir retroalimentar la base de conocimiento de la que se sustenta la lógica de negocio asociada al algoritmo de predicción, y además va a permitir catalogar al usuario en cuanto a la restricción de platos con contenidos alérgenos.

La información mostrada de la carta de los restaurantes es la funcionalidad con mayor importancia en este trabajo. En la Figura 45 se muestra un ejemplo de la carta de un restaurante, donde la información está dividida en tres bloques:

1. El primero corresponde con la predicción resultante de aplicar el algoritmo de *Naïve Bayes* a las características del usuario. Esta predicción sugiere el plato de la carta del restaurante que mayor probabilidad tiene de ser ideal para el usuario y que además no contiene ningún tipo de alérgeno, asociado a alguna de sus alergias.
2. El segundo bloque corresponde con los platos de la carta del restaurante que contienen algún tipo de alérgeno peligroso para el usuario. Esta información es muy importante para orientar a los usuarios y preavisarles de que la ingesta de ese plato puede ser perjudicial para él.
3. El tercer bloque contiene todos los platos de la carta del restaurante.

Por último, se puede destacar la funcionalidad que permite el feed-back de los usuarios, tanto al propio sistema, como a los restaurantes. Para ello se han diseñado una serie de pantallas para permitir a los usuarios el envío de valoraciones.

Para valorar un restaurante, los usuarios acceden al detalle y seleccionan la opción de: valorar restaurante o valorar platos. En las Figura 47 y Figura 48 se puede observar como son las pantallas destinadas a las valoraciones de restaurantes y platos.

Las valoraciones realizadas podrán servir, en posteriores versiones de la aplicación, como opiniones, sugerencias u orientaciones entre usuarios. Además, también se pueden utilizar para fomentar un mejor servicio a los restaurante, reportándoles entre otros, que platos tiene mayor aceptación por sus clientes, carencias que tiene sus restaurante y en general unas conclusiones que pueden ayudar a mejorar los servicios prestados por ellos.

También se permite a los usuarios la valoración de la propia aplicación. En la Figura 53 se muestra el formulario, a cumplimentar por los usuarios que así lo deseen, que permite enviar sus opiniones, errores o sugerencias de su uso con la aplicación. Esto puede permitir una evaluación del sistema en base a experiencias reales para mejorar los servicios de esta.

En el Anexo V. Aplicación se pueden consultar todas las pantallas de las que esta provista la aplicación *TuPaladar*.

4.4.2. Evaluación del sistema

La evaluación de la aplicación se va a realizar en base a dos pilares. El primero consistente en una batería de pruebas de integración para el análisis funcional de la aplicación. El segundo consiste en un test y pruebas de usabilidad por usuarios externos para la evaluación y satisfacción con la aplicación.

Pruebas de funcionalidad

Las pruebas realizadas sobre la aplicación están aplicadas a cada caso de uso diseñado, y siguen el patrón de la plantilla de la Tabla 72.

Tabla 72. Plantilla prueba funcional

Prueba	
Caso de Uso asociado	
Nombre	
Descripción	
Resultado esperado	
Estado	
Observaciones	

(Elaboración propia)

Las columnas: caso de uso asociado, nombre y descripción, corresponden con la información relativa al caso de uso a evaluar. En la columna estado esperado, se describen las acciones esperadas a realizar por el sistema asociadas al paso del caso de uso. En cuanto al Estado, los posibles valores son: **Satisfactorio**, si la prueba ha desarrollado con éxito; **Con errores**, si la prueba ha fallado; **No aplica**, si la prueba no se puede llevar a cabo.

En el Anexo VI. Pruebas de funcionalidad se incluye la batería de pruebas llevadas a cabo por cada requisito funcional del aplicativo.

Pruebas de usabilidad

La usabilidad es una disciplina que estudia la interacción de los usuarios con las aplicaciones y sitios web. Por ello, en el ámbito software, la usabilidad en una aplicación se refiere a la

facilidad y la comodidad de uso de un sistema por los usuarios, y de si su uso es intuitivo para realizar las tareas para las que está programado. Los atributos principales para considerar que un sistema es usable son los siguientes: [24]

- **Efectividad.** Se trata de evaluar la precisión y la completitud de objetivos por los usuarios al utilizar la aplicación. Para ello, se evalúa la calidad de la solución y la tasa de errores cometidos por los usuarios.
- **Eficiencia.** Es la relación entre efectividad y el esfuerzo llevados a cabo por los usuarios para alcanzar el objetivo. Se evalúa el tiempo de finalización de tareas y el tiempo de aprendizaje.
- **Satisfacción.** Es el grado de satisfacción de los usuarios al utilizar la aplicación. Este atributo es subjetivo y muestra la experiencia de calidad y satisfacción del sistemas en los usuarios.

Para evaluar la usabilidad en la aplicación *TuPaladar*, se va a realizar una prueba con 3 usuarios reales. Esta prueba se realiza mediante una plantilla, donde se solicita a los usuarios la ejecución de 3 tareas en la aplicación, para la evaluación de la aplicación en base a los criterios expuestos anteriormente. La plantilla de la prueba de usabilidad es la siguiente:

Objetivo de la prueba

Se pretende evaluar la usabilidad del sistema *TuPaladar* bajo los criterios de efectividad, eficiencia y satisfacción con usuarios reales. Para ello, se define una prueba experimental que consiste en definir y completar una serie de tareas por cada usuario, para realizar un análisis cualitativo de los resultados obtenidos.

Tareas

En la aplicación se destacan 3 principales a evaluar:

- Registro y login.
- Búsqueda de restaurante y platos, y reserva de mesa.
- Valoración de un restaurante y un plato del restaurante.

Usuarios

Como información de los usuarios se tomará la siguiente información:

- Sexo
- Edad

- Ciudad habitual

Plantilla de la prueba

Tarea 1. Registro, login y logout

Esta tarea consiste en la realización del registro en la aplicación *TuPaladar* y el posterior acceso al sistema con el usuario creado. Los datos introducidos en el formulario de registro pueden ser personales o ficticios. Posteriormente tiene que cerrar sesión en la aplicación.

Medidas a evaluar

Eficiencia:

- Tiempo total en la realización de la tarea.
- Tiempo total en el registro.
- Tiempo total en login.
- Tiempo total en logout.

Eficacia:

- Ha conseguido introducir los datos personales.
- Ha conseguido introducir una contraseña segura.
- Ha conseguido registrarse.
- Ha conseguido acceder a la aplicación con el usuarios creado.
- Ha conseguido cerrar sesión en la aplicación.
- Número de errores:
 - Número y descripción de error.
 - ...

Observaciones:

- ...

Tarea 2. Búsqueda de restaurantes y platos, y reserva de mesa

Esta tarea consiste en buscar el restaurante *La Mafia se sienta a la Mesa* de Mérida, acceder a su carta para la visualización de platos y posteriormente simular una reserva de mesa en el restaurante.

Medidas a evaluar**Eficiencia:**

- Tiempo total en la realización de la tarea.
- Tiempo total en localizar el restaurante.
- Tiempo total en localizar la carta del restaurante.
- Tiempo total en visualizar y entender la carta.
- Tiempo en localizar la reserva de mesa.
- Tiempo total en reservar mesa.

Eficacia:

- Ha conseguido localizar el restaurante.
- Ha conseguido localizar la carta del restaurante.
- Ha conseguido localizar la reserva de mesa.
- Ha conseguido reservar mesa.
- Número de errores:
 - Número y descripción de error.
 - ...

Observaciones:

- ...

Tarea 3. Valoración de un restaurante y un plato del restaurante

Esta tarea consiste en buscar el restaurante *La Mafia se sienta a la Mesa* de Mérida, valorar la satisfacción con el restaurante, bajo la suposición de haber comido en él, y, además, valorar el plato *Sticks crujientes de focaccia*.

Medidas a evaluar**Eficiencia:**

- Tiempo total en la realización de la tarea.
- Tiempo total en localizar el restaurante.
- Tiempo total en localizar valorar restaurante.

- Tiempo total en valorar el restaurante.
- Tiempo total en localizar valorar plato.
- Tiempo total en localizar el plato.
- Tiempo total en valorar el plato.

Eficacia:

- Ha conseguido localizar el restaurante.
- Ha conseguido valorar el restaurante.
- Ha conseguido localizar valorar platos.
- Ha conseguido localizar el plato.
- Ha conseguido valorar el plato.
- Número de errores:
 - Número y descripción de error.
 - ...

Observaciones:

- ...

Satisfacción:

1. En general estoy satisfecho con la facilidad de realizar las tareas en la aplicación, donde 1 es poco satisfecho y 5 es muy satisfecho:

1 2 3 4 5

2. En general estoy satisfecho con el tiempo tardado en realizar las tareas en la aplicación, donde 1 es poco satisfecho y 5 es muy satisfecho:

1 2 3 4 5

3. En general estoy satisfecho con los mensajes de aviso realizar las tareas en la aplicación, donde 1 es poco satisfecho y 5 es muy satisfecho:

1 2 3 4 5

4. Satisfacción general con la aplicación:

1

2

3

4

5

En el Anexo VII. Pruebas de usabilidad se incluyen las pruebas de usabilidad llevadas a cabo por usuarios externos a la aplicación.

5. Conclusiones y trabajo futuro

5.1. Conclusiones

Una vez realizado este trabajo, se destacan una serie de conclusiones al mismo que han ido surgiendo en el transcurso del desarrollo.

La idea inicial para el desarrollo de este trabajo ha surgido en la localización de la carencia de sugerencia de comida personalizada para las personas. A raíz de esa idea se plantea la opción del desarrollo de una aplicación móvil que ofrezca información de restaurantes y platos a sus usuarios y además realice un proceso de sugerencia de comida personalizada para él.

Lo primero a realizar ha sido un estudio sobre las diferentes arquitecturas para el desarrollo de aplicaciones móviles. Para ello, se ha realizado una investigación comparativa entre aplicaciones nativas y aplicaciones híbridas, con sus ventajas y desventajas, para poder decidir, en función de las necesidades requeridas, que tipo utilizar para el desarrollo de aplicaciones móviles.

Una vez realizado el estudio se selecciona la opción de desarrollar *TuPaladar* como una aplicación híbrida, y, por tanto, basada en lenguaje web. Esta decisión ha tenido como consecuencia directa el estudio y aprendizaje de Ionic como framework de desarrollo de la aplicación.

El segundo estudio realizado corresponde con la investigación de *Machine Learning*, conceptos, algoritmos y diferentes variantes posibles para poder utilizar alguna de ellas en la predicción personalizada de comidas. Valorado el problema, por sus características de clasificación, se aplica el aprendizaje supervisado y en concreto la opción del algoritmo de *Naive Bayes* debido a su especialización en este tipo de problemas. Mediante la investigación del algoritmo, se realiza la adaptación y la aplicación práctica del mismo, bajo la abstracción del problema de predicción de alimentos en base a preferencias de personas. Además, para la realización de esta aplicación práctica, es necesario realizar una investigación de estudios estadísticos que puedan servir como sustento a la base de conocimiento del algoritmo, base muestral para futuras predicciones.

La tercera vertiente de estudio corresponde con un análisis del mercado de aplicaciones móviles enfocadas a ofrecer algún tipo de servicios de comidas. En este estudio se valoran y describen varias aplicaciones reales y se realiza una comparación funcional entre ellas.

De este estudio, se ratifica que la sugerencia de comidas no está implementada en ninguna de las opciones evaluadas y además se destaca que no ofrecen una información completa en cuanto a contenidos alérgenos.

En base a los 3 estudios realizados, se realiza el desarrollo de la aplicación TuPaladar, con la utilización de la metodología RUP, caracterizada por sus en proyectos de gran envergadura debido a los procesos y requerimientos que se definen, se define el ciclo de vida de la aplicación.

En la fase de análisis se definen y describen los requisitos del aplicativo y las decisiones de negocio que se van a desarrollar. En la fase de Diseño se moldea la aplicación y se definen todos los patrones de diseño. Posteriormente se realiza la implementación de la aplicación con Ionic y se realiza una batería de pruebas funcionales para validar su comportamiento. Por último, se evalúa la usabilidad de la aplicación mediante un test de usabilidad con usuarios finales.

5.2. Líneas de trabajo futuro

Como líneas de trabajo futuro para continuar y complementar este trabajo se pueden destacar varias opciones.

Este trabajo está basado en el uso del Clasificador de Bayes, como método predictivo y probabilístico dentro del ámbito del Machine Learning y en concreto del aprendizaje automático supervisado, para la predicción de gustos alimenticios para personas. Una posible vertiente futura puede ser la elección de un método diferente a Bayes para la predicción de gustos alimenticios. Este nuevo estudio puede estar enfocado en la comparación de ambos algoritmos para valorar por un lado el rendimiento de cada uno, o, por otro lado, evaluar cuál de los algoritmos predice las mejores hipótesis.

Otra posible vertiente puede corresponder con la ampliación de las etiquetas a manejar por el algoritmo de Bayes, con vistas a realizar una evaluación sobre el comportamiento del algoritmo en un espectro mayor de valores. Aunque el algoritmo de Bayes se caracteriza por su sencillez y rapidez, se puede analizar su efectividad y rendimiento para el aumento de información a procesar dentro de la base de conocimiento y la repercusión que pueda tener en la aplicación.

Además, se puede evaluar el algoritmo de Bayes, ampliando el conjunto de hipótesis en la clasificación de los ingredientes, o incluso, la toma de un conjunto de hipótesis diferentes a ingredientes.

En cuanto al desarrollo de la aplicación, se puede contemplar la opción de un diferente entorno tecnológico para el desarrollo de la aplicación móvil. Implementar la aplicación mediante arquitectura nativa en lugar de arquitectura híbrida, y posteriormente se puede realizar un estudio sobre el rendimiento de ambas arquitecturas.

Por último, se podría dotar a la aplicación de mayor funcionalidad y continuar con un desarrollo más maduro de la misma.

6. Bibliografía

- [1] S. Matson, Market Guide for Mobile Application Testing Services, Gartner, Jun 2017.
- [2] «Informe Mobile en España y en el Mundo 2017,» *ditrendia*, 2017.
- [3] B. Lantz, Machine Learning with R Second Edition, Packt Publishing, Julio 2015.
- [4] C. García Cambronero y I. Gómez Moreno, Algoritmos de Aprendizaje: KNN & KMEANS, Inteligencia en Redes de Telecomunicación: Universidad Carlos III de Madrid.
- [5] T. Mitchell, Machine Learning, McGraw-Hill, 1997.
- [6] M. A. Gutiérrez Naranjo, F. J. Martín Mateos y J. L. Ruiz Reina, Aprendizaje inductivo: árboles y reglas de decisión, Universidad de Sevilla: Dpto. Ciencias de la Computación e Inteligencia Artificial, 2012-2013.
- [7] «Clasificador Naive Bayes,» [En línea]. Available: <http://naivebayes.blogspot.com/>. [Último acceso: 10 08 2019].
- [8] I. Software, El desarrollo de aplicaciones móviles nativas, Web o híbridas, IBM Corporation, Abr 2012.
- [9] C. Griffith, Mobile App Development with Ionic, Cross-Platform Apps with Ionic, Angular & Cordova, O'Reilly Media, Sep 2017.
- [10] «Ionic Framework,» [En línea]. Available: <https://ionicframework.com/docs/intro>. [Último acceso: 10 Julio 2019].
- [11] «JUST EAT,» [En línea]. Available: <https://www.just-eat.es/info/acerca-de-just-eat>. [Último acceso: 12 08 2019].
- [12] «Glovo,» [En línea]. Available: <https://glovoapp.com/es/legal/terms>. [Último acceso: 12 Agosto 2019].
- [13] P. Kruchten, The Rational Unified Process An Introduction, Second Edition, Addison Wesley, 2000.

- [14] Rational Software, Rational Unified Process, Best Practices for Software, Rational Software Corporation, 1998 Rev. Ene 2001.
- [15] G. Booch, J. Rumbaugh y I. Jacobson, Unified Modeling Language User Guide, Second Edition, Addison Wesley Professional, May 2005.
- [16] UNIR, «Tema 2 - El proceso de desarrollo de software,» de *Máster en Ingeniería del Software y Sistemas Informáticos - Metodologías, Desarrollo y Calidad de la Ingeniería de Software*, 2018.
- [17] C. Péraire, M. Edwards, A. Fernandes, E. Mancin y K. Carroll, The IBM Rational Unified Process for System z, Redbooks, Jul 2007.
- [18] I. Sommerville, Ingeniería del software Séptima Edición, Pearson, 2005.
- [19] A. E. d. C. S. A. y. N. «Reglamento Información alimentaria facilitada al consumidor,» Ministerio de Sanidad, Consumo y Bienestar Social, 13 Julio 2017. [En línea]. Available:
http://www.aecosan.msssi.gob.es/AECOSAN/web/seguridad_alimentaria/subdetalle/futura_legislacion.htm. [Último acceso: 15 Agosto 2019].
- [20] M. d. M. A. y. M. R. y. M. Instituto Cerdá, «Monográfico Carne - Estudio de mercado,» Observatorio del Consumo y la Distribución Alimentaria, Septiembre 2008. [En línea]. Available: https://www.mapa.gob.es/es/alimentacion/temas/consumo-y-comercializacion-y-distribucion-alimentaria/monografico_carne_08_tcm30-89283.pdf. [Último acceso: 5 Agosto 2019].
- [21] M. d. M. A. y. M. R. y. Marino, «Monográfico Productos de Acuicultura - Estudio de mercado,» Observatorio del Consumo y la Distribución Alimentaria, Diciembre 2009. [En línea]. Available: https://www.mapa.gob.es/es/alimentacion/temas/consumo-y-comercializacion-y-distribucion-alimentaria/consumo_prod_acuicultura_tcm30-89475.pdf. [Último acceso: 5 Agosto 2019].
- [22] G. Génova Fúster y J. Llorens Morillo, «Naturaleza de las relaciones entre actores y casos de uso. Novática: Revista de la Asociación de Técnicos de Informática,» *Novática*, vol. 30, nº 169, pp. 56-61, May-Jun 2004.
- [23] R. Barker, Case*Method Entity Relationship Modelling, Addison-Wesley, 1990.

- [24] J. G. Enriquez y S. I. Casas, Usabilidad en aplicaciones móviles, <http://dx.doi.org/10.22305/ict-unpa.v5i2.71>: N° 0681/13-R-UNPA, 2013.

Anexos

Anexo I. Artículo

TuPaladar, aplicación para la sugerencia de menús basada en predicción de gustos

Raúl Sánchez Romero

Área de Ingeniería y Tecnología
Universidad Internacional de la Rioja
Email: sanchezrraul@gmail.com

Resumen- El auge de Internet y de las aplicaciones en el que nos encontramos actualmente en nuestra sociedad nos proporciona un sinnúmero de aplicaciones disponibles para utilizar en nuestros teléfonos móviles. TuPaladar es una aplicación móvil, que integra mecanismos de Machine Learning mediante el algoritmo de Clasificación de Bayes, para la predicción de gustos alimenticios personalizados y sugerencia de platos ideales a los usuarios. Además, incorpora información detallada de contenidos alérgenos en toda la carta de los restaurantes.

Para el desarrollo de la aplicación se ha utilizado la metodología de trabajo RUP y el modelado y diseño de sus componentes a través de UML. El aplicativo tiene una arquitectura basada en lenguaje web, Angular, mediante el uso del Framework Ionic, que incorpora las ventajas tecnológicas del desarrollo de aplicaciones móviles híbridas permitiendo, entre otras características, generar aplicaciones portables para cualquier arquitectura móvil.

Palabras Clave- Machine Learning, Predicción alimenticia, Clasificador de Bayes, Ionic, Ingeniería del Software.

I. INTRODUCCIÓN

En el siguiente trabajo se desarrolla una aplicación híbrida móvil, TuPaladar, enfocada en ofrecer servicios de comida, añadiendo una funcionalidad innovadora en cuanto al sector para la sugerencia personalizada de comidas.

El objetivo principal es desarrollar una aplicación móvil, denominada “Tu paladar”, que proporcione información detallada de restaurantes y contenidos alérgenos a los usuarios, que esté dotada de funcionalidades como localizar restaurantes por ubicación, consultar la carta y precios, reservar mesas y consultar valoración y opiniones de otros clientes, y cuya finalidad innovadora es la predicción de comida o plato “ideal” para cada usuario / restaurante mediante la utilización de técnicas de inteligencia artificial.

Además, se destacan una serie de objetivos específicos:

- Proporcionar información detallada de la carta y platos de los restaurantes, así como del contenido de ingredientes alérgenos.
- Proporcionar sugerencias de platos ideales para los comensales por restaurantes mediante un sistema de aprendizaje personalizado y automatizado por usuario mediante la combinación, por un lado, de sus gustos,

alergias y opiniones y, por otro lado, la información de los platos, de los restaurantes y de las opiniones de otros usuarios.

- Determinar que parámetros más característicos se pueden tener en cuanto, para los usuarios de la aplicación, para basarlos en la búsqueda de predicción de sugerencia de platos.
- Identificar que algoritmo de Inteligencia Artificial puede ofrecer una solución factible para la predicción de sugerencia de platos.
- Desarrollar una única aplicación híbrida móvil para las dos principales arquitecturas de móviles, Android e iOS con el objetivo de reducir costes derivados del desarrollo de dos aplicaciones.

II. CONTEXTO

Hoy en día, la estrategia comercial para muchas organizaciones pasa por la implementación de app corporativas para dispositivos móviles, como teléfonos, relojes inteligentes, coches, etc. Ya sea para ofrecer soluciones en esos dispositivos, promovidos por el marketing aumentando las opciones para darse a conocer o para permitir una gestión más accesible y fácil para sus clientes.

Además, según afirma Gartner [1] desde el estudio Market Guide for Mobile Application Testing Services, “El 70% de las interacciones que se llevarán a cabo desde 2022 serán desde un dispositivo móvil”. En el caso de España, “El móvil es el dispositivo más utilizado para acceder a Internet y usado ya por el 94,6%” [2].

En las aplicaciones móviles que ofrecen servicios de comida es muy importante reflejar contenido de alérgenos, ya que para aquellas personas que tienen algún tipo de intolerancia puede ser un problema esta desinformación.

Además, sería interesante que estos servicios ofrecieran sugerencias de comida a los comensales en base a sus preferencias.

En este ámbito, podemos encontrar numerosas aplicaciones para móviles que tiene por objetivo facilitarnos información de restaurante, bares y establecimientos de comidas para ofrecernos servicios, como la reserva de mesa en un restaurante

o pedir comida a domicilio. A continuación, en la Fig. 1 se puede observar una comparación de entre las más comunes.

Aplicación	Globo	Just Eat	El Tenedor	Deliveroo	Yelp
Búsqueda	✓	✓	✓	✓	✓
Carta	✓	✓	✓	✓	✗
Información	✗	✓	✓	✓	✓
Reservas	✗	✗	✓	✗	✗
Opiniones	✗	✓	✓	✗	✓
Incentivos	✓	✓	✓	✗	✗
Alérgenos	✗	✗	✗	✗	✗
Avanzada	✗	✓	✗	✗	✗
Sugerencias	✗	✗	✗	✗	✗
Coste	✗	✓	✓	✓	✗

Fig. 1. Comparativa app servicios de comida

III. MACHINE LEARNING

Es una disciplina dentro de la Ingeniería de la Computación basada en la creación de sistemas software capaces de llevar a cabo un proceso de aprendizaje de manera autónoma. El sistema aprende mediante el uso de un algoritmo que procesa y cataloga conjunto de datos con el objetivo de predecir situaciones futuras. Tiene 3 conceptos principales: [3][4]

- Base de conocimiento. Es la información que contiene el sistema y utilizada en el proceso de predicción del algoritmo. Se distinguen, conjunto de entrenamiento para identificar los parámetros de clasificación y conjunto de pruebas para el cálculo de la estimación de error.
- Clasificador. Es la relación de los datos de la base de conocimiento que se utilizan para la predicción.
- Aprendizaje. Es el procedimiento que construye un clasificador partiendo de la base de conocimiento. Además, se complementa con mecanismos de evaluación y retroalimentación de nuevo conocimiento.

TIPOS DE ALGORITMOS MACHINE LEARNING

Se pueden distinguir 3 tipos de categorías: [3]

- Aprendizaje supervisado. Se caracterizan por tener claramente definido el procedimiento de que y como aprender. Trabajan con datos etiquetados, donde dada una entrada, encuentran la función más adecuada para la salida. Se basan en conjunto de entrenamiento y aprendizaje, para la ponderación predictiva de la etiqueta de salida. Suelen utilizar para problemas de Clasificación.
- Aprendizaje no supervisado. Estos algoritmos no se basan en un modelo predictivo sino en un modelo descriptivo, donde al no haber una única característica, no hay un objetivo concreto para aprender, es decir, no tienen una base de conocimiento previamente etiquetada. Suelen utilizarse en problemas donde no hay un objetivo definido, basándose en la exploración y la agrupación de los datos.
- Aprendizaje por refuerzo. Estos algoritmos no tienen una tarea de aprendizaje específica, sino que se centran en aprender el cómo aprender de la manera más eficiente, es decir, estos algoritmos se basan en mejorar la respuesta del

modelo usando un proceso de retroalimentación. El sistema aprende en base ensayo-error.

ALGORITMO DE NAIVE BAYES

Se trata de una técnica de clasificación y predicción supervisada caracterizada por su simplicidad y rapidez. Es considerado uno de los más importantes clasificadores, debido a su perspectiva intuitiva y útil para la comprensión de algoritmos que no manipulan explícitamente las probabilidades. Basa su conocimiento en un modelo para la predicción de probabilidades de posibles resultados. Tiene las siguientes características: [5]

- Cada muestra puede aumentar o disminuir de manera incremental la probabilidad de estimación de la hipótesis resultado, aportando flexibilidad de aprendizaje.
- El conocimiento o experiencia se combina con los datos de la observación para el cálculo de la hipótesis resultante mediante predicciones probabilísticas de la hipótesis.
- Las muestras observadas se pueden clasificar combinando las probabilidades de las hipótesis.
- Está basado en el teorema de Bayes.

TEOREMA DE BAYES

Proporciona un método directo para el cálculo de las probabilidades de una hipótesis basada en su probabilidad previa, las probabilidades de la observancia de varios datos dada la hipótesis y los datos observados en sí.

Supongamos una hipótesis h que pertenece al conjunto de hipótesis H y un conjunto de entrenamiento D . El teorema de Bayes calcula la hipótesis h más probable, dado D . [5] La ecuación del Teorema de Bayes es la siguiente Ec. 1 :

$$P(h|D) = \frac{P(D|h)P(h)}{P(D)} \quad (1)$$

Donde:

- $P(h)$ es la probabilidad previa de h antes de haber observado D . En el caso de no tener el conocimiento previo de $P(h)$, se posibilita que simplemente se asigne una probabilidad equitativa entre cada hipótesis candidata.
- $P(D)$ es la probabilidad previa de que se observen los datos de entrenamiento, es decir, la probabilidad de que D no tenga conocimientos sobre la h .
- $P(D|h)$ es la probabilidad de observar los datos de D dado que la h exista.
- $P(h|D)$ es la probabilidad que tiene la h dado D . Denominada probabilidad posterior h , porque muestra la confianza de la h después de haber observado D . Cabe destacar que $P(h|D)$ muestra la influencia de los D , de manera contraria, $P(h)$ es independiente de D .

En Ec. 1 se observa que, $P(h|D)$ aumenta cuanto mayor sean las $P(D|h)$ y $P(h)$, y disminuye cuanto mayor sea $P(D)$. Cuanto más probable es que no se observe h en D , menos evidencia hay de h en D . [5]

El escenario más común para aplicar el teorema, es dado un conjunto de hipótesis candidatas H , estar interesado en encontrar la hipótesis más probable h que pertenece a H dados los datos observados D . Si hubiera diferentes hipótesis, el resultado sería la hipótesis máxima a posteriori o hipótesis máxima probable MAP . [5]

$$h_{MAP} = \operatorname{argmax}_{h \in H} P(h|D)$$

$$\begin{aligned}
 &= \operatorname{argmax}_{h \in H} \frac{P(D|h)P(h)}{P(D)} \\
 &= \operatorname{argmax}_{h \in H} P(D|h)P(h)
 \end{aligned} \quad (2)$$

Se puede observar de Ec. 2 que el divisor $P(D)$ ha desaparecido, ya que es una constante que no tiene dependencia con la hipótesis. [5]

Por último, la ecuación se puede simplificar aún más, Ec. 3, considerando asumir que cada h de H es igualmente probable a priori. Es decir, dado h_i y h_j que pertenecen al conjunto de H , suponemos que $P(h_i) = P(h_j)$, ha esta hipótesis también se la denomina, hipótesis de máxima probabilidad h_{ML} : [5]

$$h_{ML} = \operatorname{argmax}_{h \in H} P(D|h) \quad (3)$$

IV. TECNOLOGÍAS DESARROLLO APLICACIONES MÓVILES

Para el desarrollo e implementación de app móviles existen dos soluciones principales.

Por un lado, están las aplicaciones nativas. Para crear una aplicación nativa es necesario implementar la aplicación mediante código fuente específico para esa plataforma. Una vez implementado, el código es compilado y generado un ejecutable que contendrá la aplicación empaquetada y lista para su instalación. Al conjunto de herramientas que incluyen el acceso a recursos y archivos suministrado para realizar el desarrollo se denomina SDK (Software Development Kit) [6]. En la Fig 2. Se muestran las principales tecnologías.



	Apple iOS	Android	Blackberry OS	Windows Phone
Lenguajes	Objective-C, C, C++	Java (algunos C, C++)	Java	C#, VB.NET, etc.
Herramientas	Xcode	Android SDK	BB Java Eclipse Plug-in	Visual Studio, Windows Phone
Formato	.app	.apk	.cod	.xap
Tiendas	Apple App Store	Google Play	Blackberry App World	Windows Phone Marketplace

Fig. 2. Características aplicaciones nativas

Por otro lado, están las aplicaciones híbridas y en concreto Ionic. Es un framework diseñado para el desarrollo de aplicaciones móviles mediante el lenguaje de tecnologías estandarizadas web. Las aplicaciones híbridas se caracterizan por ser multiplataforma. En la Fig.3 se puede observar una comparativa entre ambas opciones de Desarrollo.









Característica	Aplicación Nativa	Aplicación Híbrida
Tecnología necesaria	Objective-C, iOS SDK Java, Android SDK C#, Windows SDK	HTML, CSS, JS + framework
Distribución		
Tiempos desarrollo		
Costes	€ € €	€
Rendimiento		
Funcionalidad disponible		

Fig. 3. Comparativa aplicaciones nativas vs híbridas

V. METODOLOGÍA Y REQUERIMIENTOS

Se va a utilizar la metodología RUP para el ciclo de desarrollo de software con las siguientes características, Fig. 4: [7]

- Dirigido a casos de uso.
- Centrado en la arquitectura.
- Iterativo e incremental

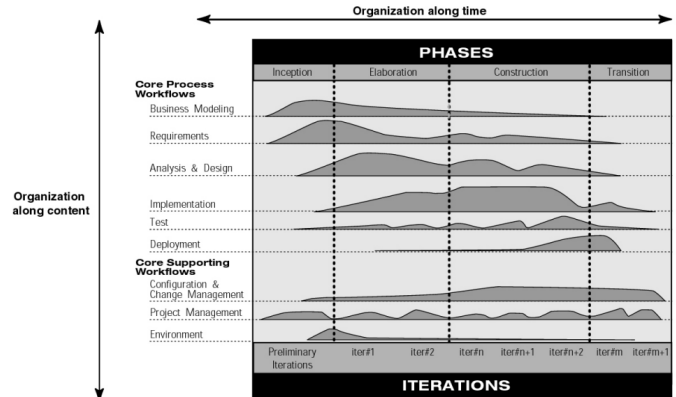


Fig. 4. Representación de iteraciones y fases metodología RUP

Los requisitos o requerimientos de un sistema son la descripción de los servicios proporcionados por el sistema y sus restricciones operativas. [8]

REQUISITOS FUNCIONALES

Los requisitos funcionales del sistema son las descripciones detalladas del comportamiento que debe tener el software y como debe manejar la información. [8]

1. El sistema debe contemplar dos roles de usuarios: Usuario particular consumidor de la app y Usuario restaurante.
2. El sistema debe permitir el registro de usuarios.
3. El sistema debe validar los datos introducidos en formularios.
4. El sistema debe permitir el acceso a usuarios previamente registrados.
5. El sistema debe permitir la modificación de la información personal a los usuarios.
6. El sistema debe permitir la consulta de restaurantes.
7. El sistema debe sugerir a los usuarios platos de los restaurantes.
8. El sistema debe permitir a los usuarios pedir comida a domicilio.
9. El sistema debe permitir reservar mesa en un restaurante a los usuarios.
10. El sistema debe permitir a los usuarios valorar restaurantes.
11. El sistema debe permitir a los usuarios, que han iniciado sesión, cerrar la sesión.
12. El sistema debe permitir registrar nuevos restaurantes.
13. El sistema debe permitir modificar información de restaurantes.
14. El sistema debe permitir el envío de sugerencias.
15. El sistema debe permitir la actualizar la base de conocimiento del algoritmo predictivo.

REQUISITOS NO FUNCIONALES

Estos requisitos son aquellos requerimientos que no se refieren a las funciones específicas proporcionadas por el sistema, sino que representan características generales y restricciones de la aplicación. [8]

1. Usabilidad
 - a. El sistema es responsive.

- b. El sistema se desarrolla siguiendo pautas de usabilidad como fácil uso y acceso, diseño estético y fácil aprendizaje.
- c. El sistema implementa un procedimiento de envío de opiniones.
- 2. Rendimiento
 - a. El sistema debe responder correctamente a todas las demandas de procesamiento.
 - b. El sistema debe ejecutar el proceso de sugerencia de platos en tiempo adecuado.
- 3. Portabilidad
 - a. La aplicación es portable a diferentes entornos: iOS, Android, Windows Phone y PWA.
- 4. Privacidad
 - a. El sistema implementa Políticas y Condiciones de Privacidad.
- 5. Seguridad
 - a. El sistema implementa Políticas de Seguridad y Condiciones de uso.
 - b. El sistema identifica y autentica usuarios mediante usuario/contraseña.
 - c. El sistema implementa mecanismos de validación.
 - d. El sistema implementa un mecanismo para la encriptación de contraseñas por algoritmo de autenticación digest SHA-256.
 - e. El sistema implementará mecanismos de auditorías.
 - f. El sistema realiza un envío y recepción de datos mediante el cifrado de la información punto a punto.
 - g. El sistema implementa un sistema de backup de Base de Datos para salvaguardar la información de manera periódica.
 - h. El sistema implementa medidas para preservar los R.N.F de Privacidad y Disponibilidad de la información.
- 6. Disponibilidad
 - a. El sistema debe permitir la disponibilidad de los datos en todo momento.

VI. APLICACIÓN DEL ALGORITMO DE BAYES

En este apartado se van a aplicar los conceptos del Algoritmo de Bayes al problema de la sugerencia de comida.

FORMULACIÓN DEL PROBLEMA

El problema para solventar en este trabajo se formula con las siguientes cuestiones:

¿Qué platos de la carta de un restaurante es el más indicado para sugerir a una persona? ¿Cuál sería el plato ideal para sugerir a un comensal en un restaurante concreto?

BASE DE CONOCIMIENTO INICIAL

Los datos que vamos a etiquetar de la base de conocimiento para las personas son:

- **Sexo:** Hombre o mujer.
- **Edad:** Entre 20 y 35 años, entre 36 y 45 años, entre 46 y 55 años, entre 56 y 65 años, o más de 65 años.
- **Habitat:** Menos de 10.000 habitantes, entre 10.001 y 50.000 habitantes, entre 50.001 y 100.000 habitantes, o más de 100.000 habitantes.

Para los platos de la carta de los restaurantes, vamos a etiquetar la siguiente información para los **ingredientes principales**. Divididos en dos grupos. Carnes: Vacuno, Porcino, Pollo, Ovino/Caprino, Conejo, Pavo y otras aves. Pescados: Blanco y Azul.

Además, es necesario los ingredientes **alérgenos**: Gluten, crustáceos, pescado, cacahuetes, soja, lácteos, frutos de cáscara, apio, mostaza, sésamo, sulfitos, moluscos y altramuces.

FUENTE DE DATOS

La base de conocimiento se va a contemplar con los datos de los estudios de carne y pescado en España [9] [10]. En ellos se muestra la frecuencia de consumo de carne y pescado, respectivamente, en función de características de sexo, rango de edad y rango de habitat en el lugar de residencia.

DATOS DE ENTRADA AL ALGORITMO

Información del usuario:

- Sexo del usuario.
- Edad del usuario.
- Información de su domicilio.
- Valoración o ponderación de sus gustos sobre los diferentes tipos de carnes y pescados.
- Información de alergias.

Información del restaurante:

- Platos de la carta.
- De cada plato su contenido en ingredientes principales.
- De cada plato el contenido alérgeno.

HIPÓTESIS O CLASES DEL ALGORITMO

Se van a utilizar los ingredientes principales:

- Carne de vacuno.
- Carne de porcino.
- Carne de pollo.
- Carne de ovino/caprino.
- Carne de conejo.
- Carne de pavo y otras aves.
- Pescado blanco.
- Pescado azul.

APLICACIÓN DEL ALGORITMO

La predicción del algoritmo, por tanto, tiene que ser uno de los 8 ingredientes principales, el que tenga mayor probabilidad. Por tanto, partimos de la Ec. 2. Además, se parte de la situación base, de que cada clase, a priori, tiene la misma probabilidad de ser elegida, es decir, todas las clases son equiprobables a ser elegidas. Por tanto, la ecuación a integrar en el problema es Ec. 3. Para ello vamos a motivar este razonamiento con un ejemplo: Dada una persona mujer de 39 años, residente en Madrid, se quiere predecir la sugerencia de plato, carne de vacuno o carne de pollo. Se tiene en la base de conocimiento la siguiente información, en base a las etiquetas de sexo, edad y hábitat del rango de la persona:

- Valor carne vacuno / Mujer : 0,78 ; Valor carne vacuno / 36-45 años : 0,89 ; Valor carne vacuno / Más 100.000 hab. : 0,82

- Valor carne pollo / Mujer : 0,90 ; Valor carne pollo / 36-45 años : 0,80 ; Valor carne pollo / Más 100.000 hab. : 0,83

Aplicamos la ecuación de la Figura 4 para ambas clases:

$$P(\text{Carne vacuno}) = P(\text{Mujer}|\text{Carne vacuno}) * P(36 - 45 \text{ años}|\text{Carne vacuno}) * P(100.000\text{hab}|\text{Carne vacuno})$$

$$P(\text{Carne vacuno}) = 0,78 * 0,89 * 0,82 = 0,569$$

$$P(\text{Carne pollo}) = P(\text{Mujer}|\text{Carne pollo}) * P(36 - 45 \text{ años}|\text{Carne pollo}) * P(100.000\text{hab}|\text{Carne pollo})$$

$$P(\text{Carne pollo}) = 0,90 * 0,80 * 0,83 = 0,597$$

$$h_{ML} = \text{Carne pollo}$$

El siguiente paso será discriminar de la lista de platos aquellos que no tengan como ingrediente principal el sugerido por el algoritmo.

Y, por último, a esa lista de platos, hay que discriminar aquellos que contengan algún contenido alérgenos para la persona. Pueden darse como resultado final los siguientes casos:

- Si la lista resultante es vacía, no se sugiere ningún plato al usuario.
- Si la lista resultante contiene un plato, este plato se sugiere al usuario.
- Si la lista resultante contiene más de un plato, se sugiere uno de ellos al usuario.

ACTUALIZACIÓN BASE DE CONOCIMIENTO DEL ALGORITMO

Para la actualización de la base de conocimiento se va a utilizar la ecuación Ec. 4 (Es de elaboración propia).

$$NVM_k = \frac{(VAM_k * TM_k) + NVI}{NTM_k = TM_k + 1} \quad (4)$$

Donde:

- k: es la Clase cuyos valores se van a actualizar en la base de conocimiento.
- NVM: es el Nuevo Valor Muestral que contendrá la base de conocimiento.
- VAM: es el Valor Actual Muestral que contiene la base de conocimiento.
- TM: corresponde con el Tamaño Muestral actual que contiene la base de conocimiento.
- NVI: es el Nuevo Valor Individual que corresponde con la valoración del usuario registrado en la aplicación.
- NTM: es el Nuevo Tamaño Muestral que contendrá la base de conocimiento.

Cabe destacar que las ecuaciones mostradas en Figura 23, se deben de procesar por cada valoración del usuario en cada clase.

VII. DISEÑO

La aplicación se ha diseñado mediante el uso de diferentes métodos. En las Fig. 5, Fig. 6 y Fig. 7 se puede observar el diagrama de casos de uso, modelo relacional de base de datos y el diagrama de clases, respectivamente.



Fig. 5. Diagrama de Casos de uso

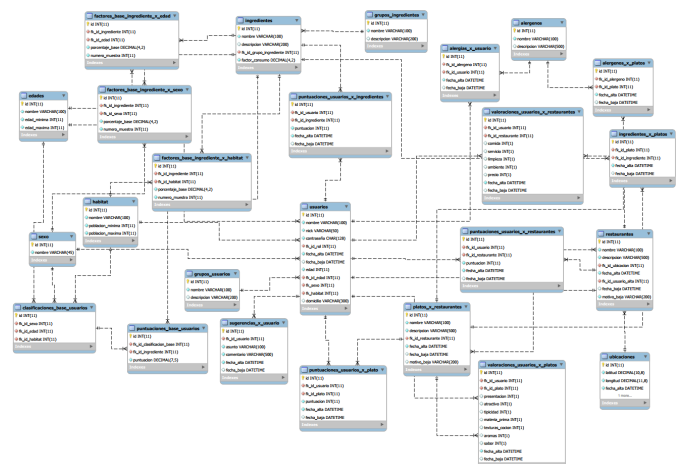


Fig. 6. Modelo relacional de base de datos

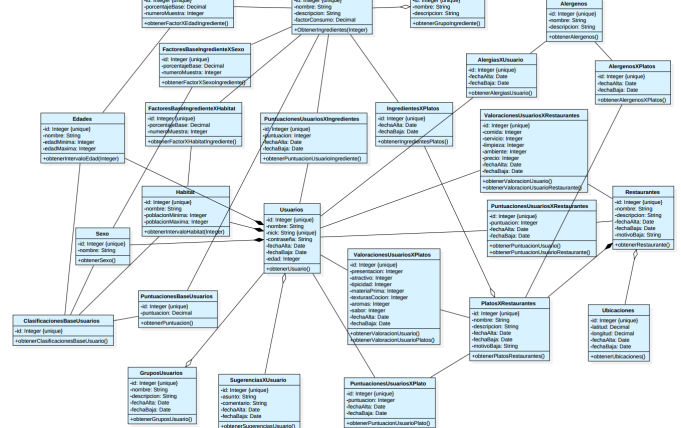


Fig. 7. Diagrama de clases

VIII. APLICACIÓN

En cuanto a la aplicación, se pueden destacar 3 funcionalidades relevantes.

La primera corresponde con los usuarios de la aplicación y con el acceso a la misma. Los usuarios particulares que accedan a la aplicación se encontraran con una estructura de 3 pestañas y un

menú lateral para poder acceder y realizar todas las opciones disponibles, Fig. 8.

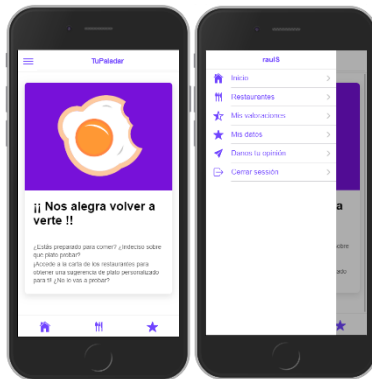


Fig. 8. Página principal y menú

Otra funcionalidad para destacar en la aplicación es el detalle de los restaurante. En la Fig. 9, se puede ver la información de un restaurante y las opciones disponibles en él. Además, destacar la carta del restaurante, donde por cada usuario, aparece la sugerencia de plato ideal para él en el restaurante y los platos potencialmente peligrosos de ingerir en base a sus alergias.

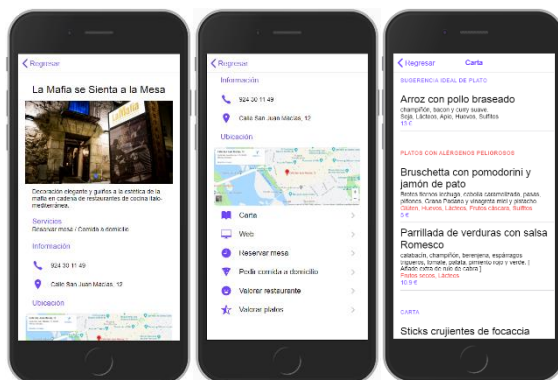


Fig. 9. Detalle de restaurante y carta

Por último, cada usuario podrá valorar su experiencia y opinión de los restaurante y platos. Para ello, en la aplicación se han habilitado unos formularios para permitir enviar opiniones a los usuarios, Fig. 10.

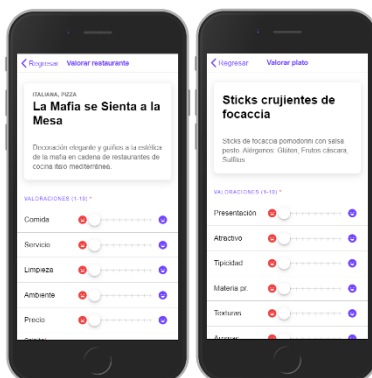


Fig. 10. Valoración de restaurante y plato

IX. CONCLUSIONES

A partir de la investigación y del trabajo desarrollado se obtienen las siguientes conclusiones.

Debido a la carencia se sugerencias de comidas y de información de alérgenos en las aplicaciones de servicios de comida, se realiza una investigación de aplicaciones en el mercado actual y se plantea la necesidad de desarrollar una aplicación para solventar este problema. Para ello se ha realizado un estudio entre aplicaciones híbridas y nativas, para utilizar las híbridas como arquitectura de la aplicación.

Se realiza un estudio de Machine Learning para aplicar el algoritmo de Bayes como método predictivo de sugerencia de alimentos.

En base a estos estudios se desarrolla la aplicación TuPaladar mediante la metodología de trabajo RUP aplicada a todo el ciclo de vida de la aplicación.

X. LÍNEAS DE TRABAJO FUTURAS

Como continuación a este trabajo, se pueden valorar varias ramas de actuación. La primera vertiente puede ser la elección de otro método de predicción para comparar su rendimiento y su efectividad con el clasificador de Naïve Bayes.

Otra línea de trabajo puede ser la ampliación en el espectro etiquetas a tener en cuenta en la predicción y evaluar su resultado. Además, y continuando con el proceso predictivo, se puede ampliar el conjunto de hipótesis, es decir, la inclusión de más ingredientes.

También se puede abordar la implementación de la aplicación con arquitectura nativa, o la ampliación de funcionalidad de la misma concluyendo con un desarrollo más completo.

REFERENCIAS

- [1] S. Matson, Market Guide for Mobile Application Testing Services, Gartner, Jun 2017.
- [2] «Informe Mobile en España y en el Mundo 2017,» ditrendia, 2017.
- [3] B. Lantz, Machine Learning with R Second Edition, Packt Publishing, Julio 2015.
- [4] C. García Cambrónero y I. Gómez Moreno, Algoritmos de Aprendizaje: KNN & KMEANS, Inteligencia en Redes de Telecomunicación: Universidad Carlos III de Madrid.
- [5] T. Mitchell, Machine Learning, McGraw-Hill, 1997.
- [6] C. Griffith, Mobile App Development with Ionic, Cross-Platform Apps with Ionic, Angular & Cordova, O'Reilly Media, Sep 2017.
- [7] Rational Software, Rational Unified Process, Best Practices for Software, Rational Software Corporation, 1998 Rev. Ene 2001.
- [8] I. Sommerville, Ingeniería del software Séptima Edición, Pearson, 2005.
- [9] M. d. M. A. y. M. R. y. M. Instituto Cerdá, «Monográfico Carne - Estudio de mercado,» Observatorio del Consumo y la Distribución Alimentaria, Septiembre 2008. [En línea]. Available: https://www.mapa.gob.es/es/alimentacion/temas/consumo-y-comercializacion-y-distribucion-alimentaria/monografico_carne_08_tcm30-89283.pdf. [Último acceso: 5 Agosto 2019].
- [10] M. d. M. A. y. M. R. y. Marino, «Monográfico Productos de Acuicultura - Estudio de mercado,» Observatorio del Consumo y la Distribución Alimentaria, Diciembre 2009. [En línea]. Available: https://www.mapa.gob.es/es/alimentacion/temas/consumo-y-comercializacion-y-distribucion-alimentaria/consumo_prod_acuicultura_tcm30-89475.pdf. [Último acceso: 5 Agosto 2019].

Anexo II. Diagrama de casos de usos

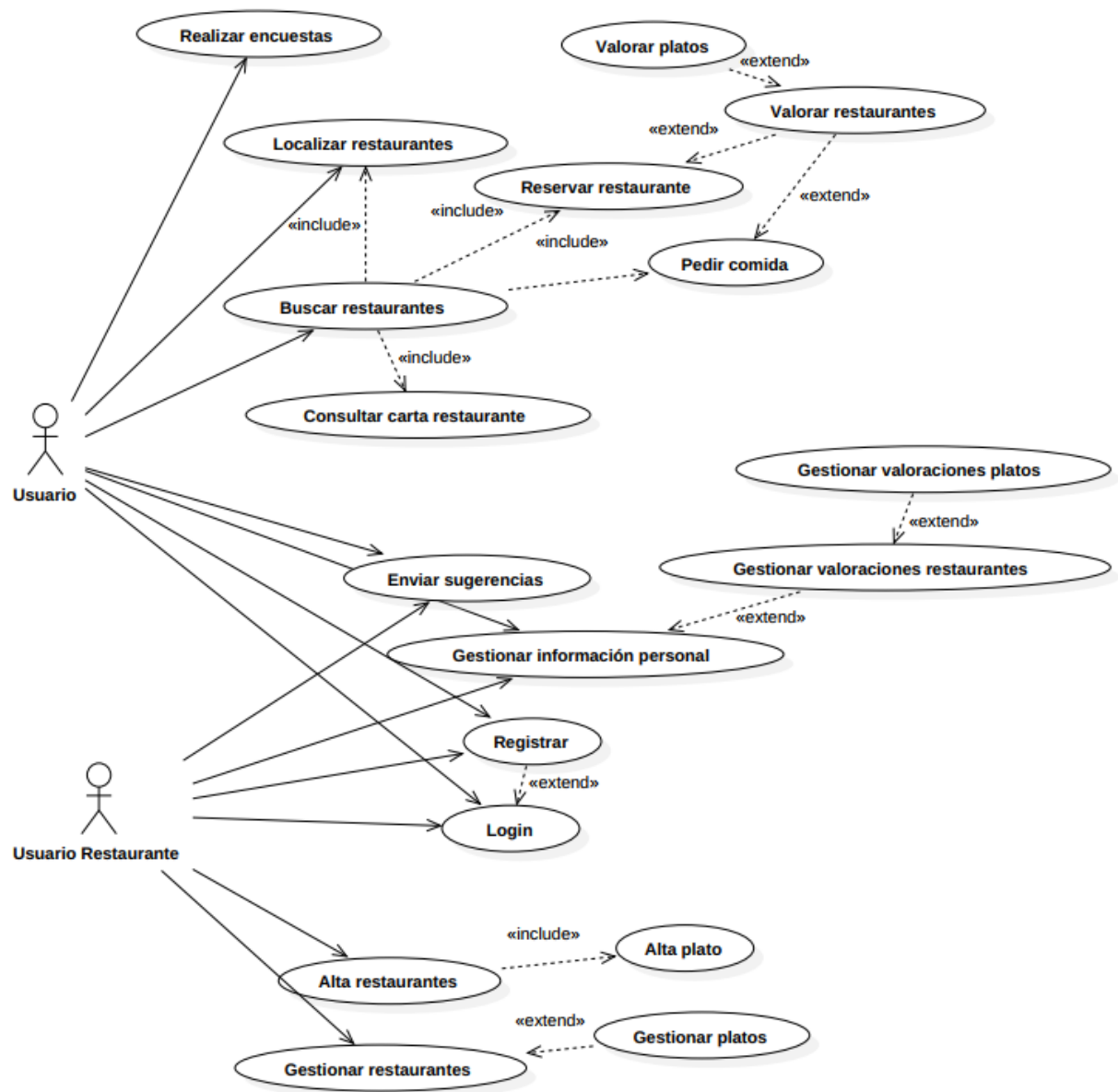


Figura 37. Diagrama de casos de uso

Anexo III. Modelo relacional de base de datos

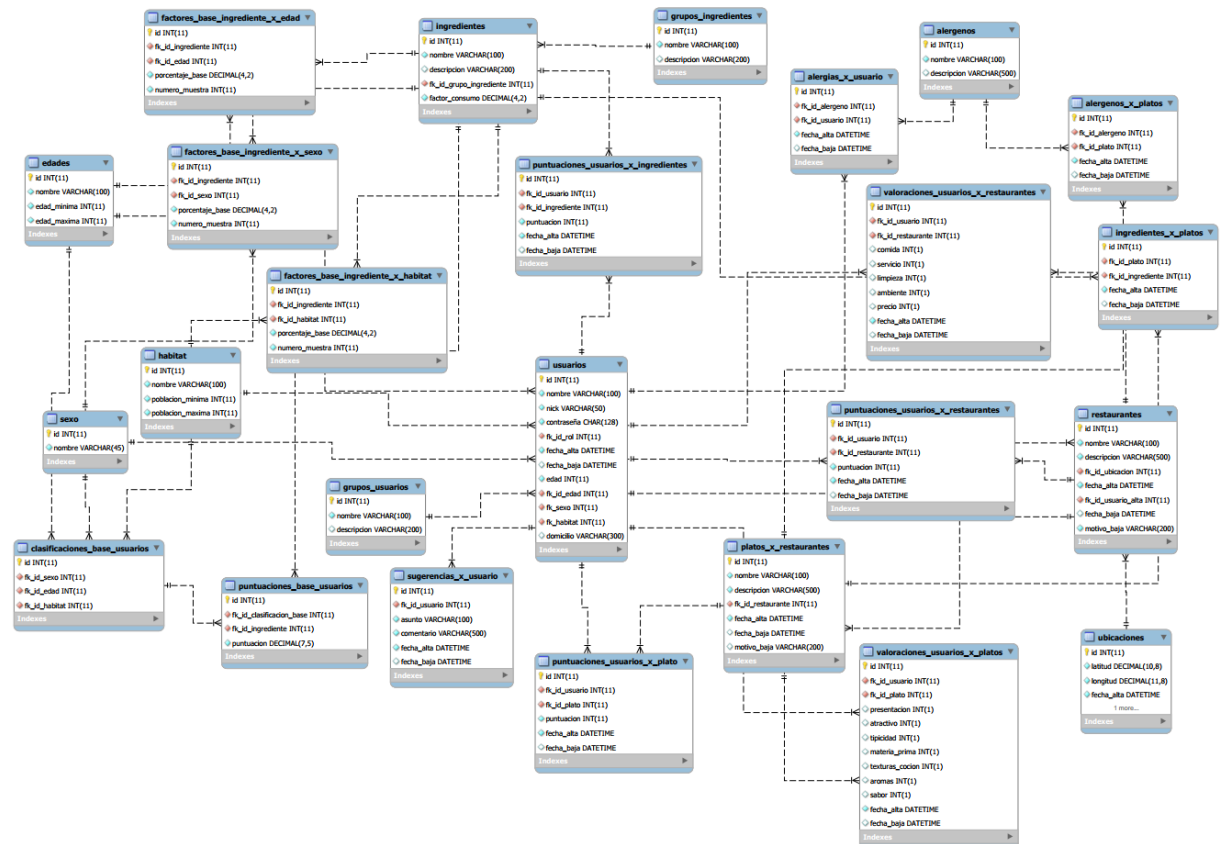


Figura 38. Modelo relacional de base de datos

Anexo IV. Diagrama de clases

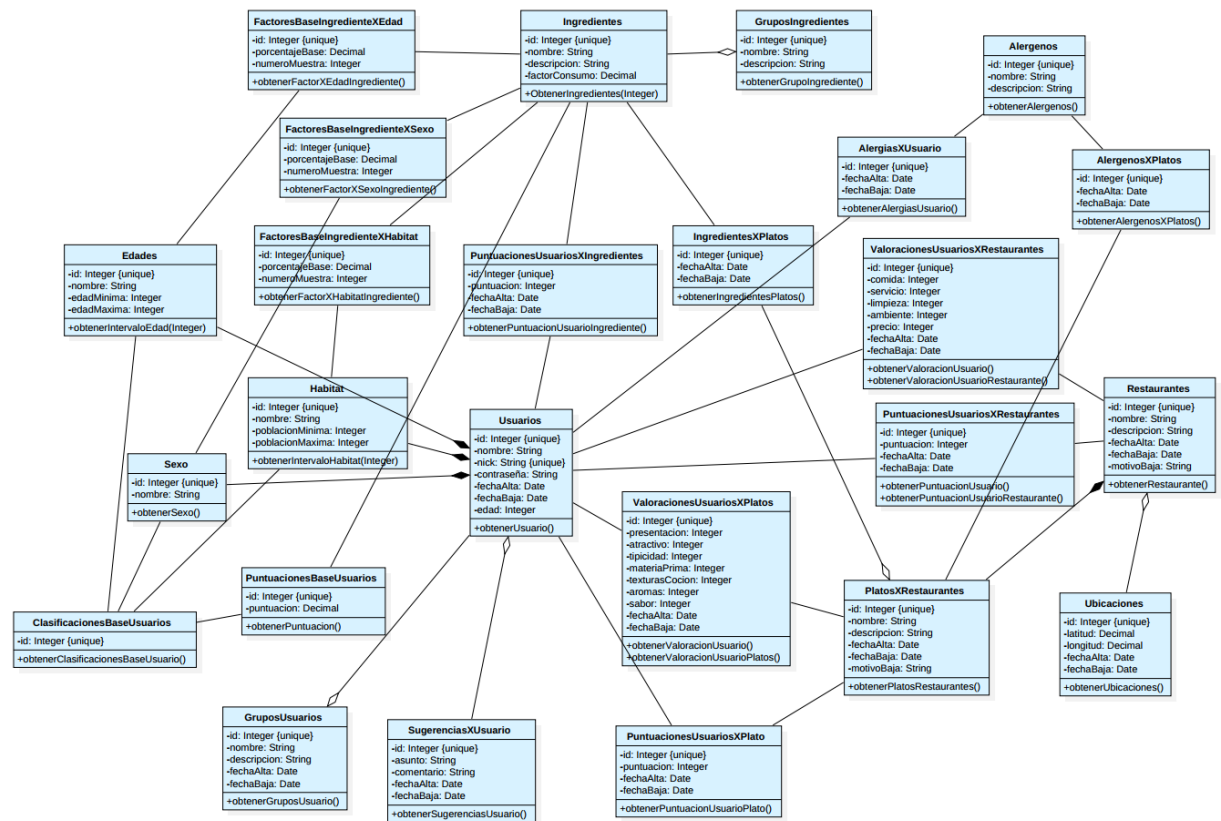


Figura 39. Diagrama de clases

Anexo V. Aplicación

Login

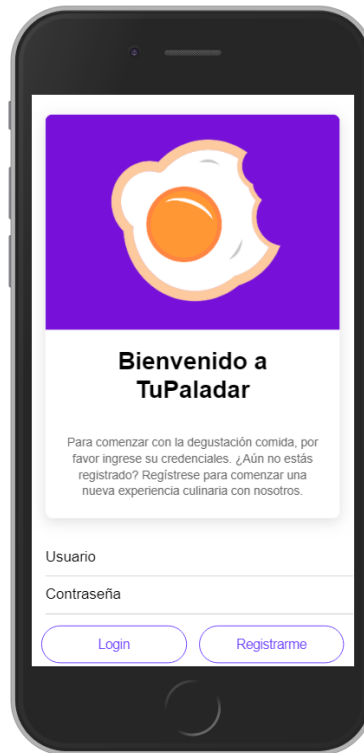


Figura 40. Pantalla login

Registro y modificación de datos usuario particular

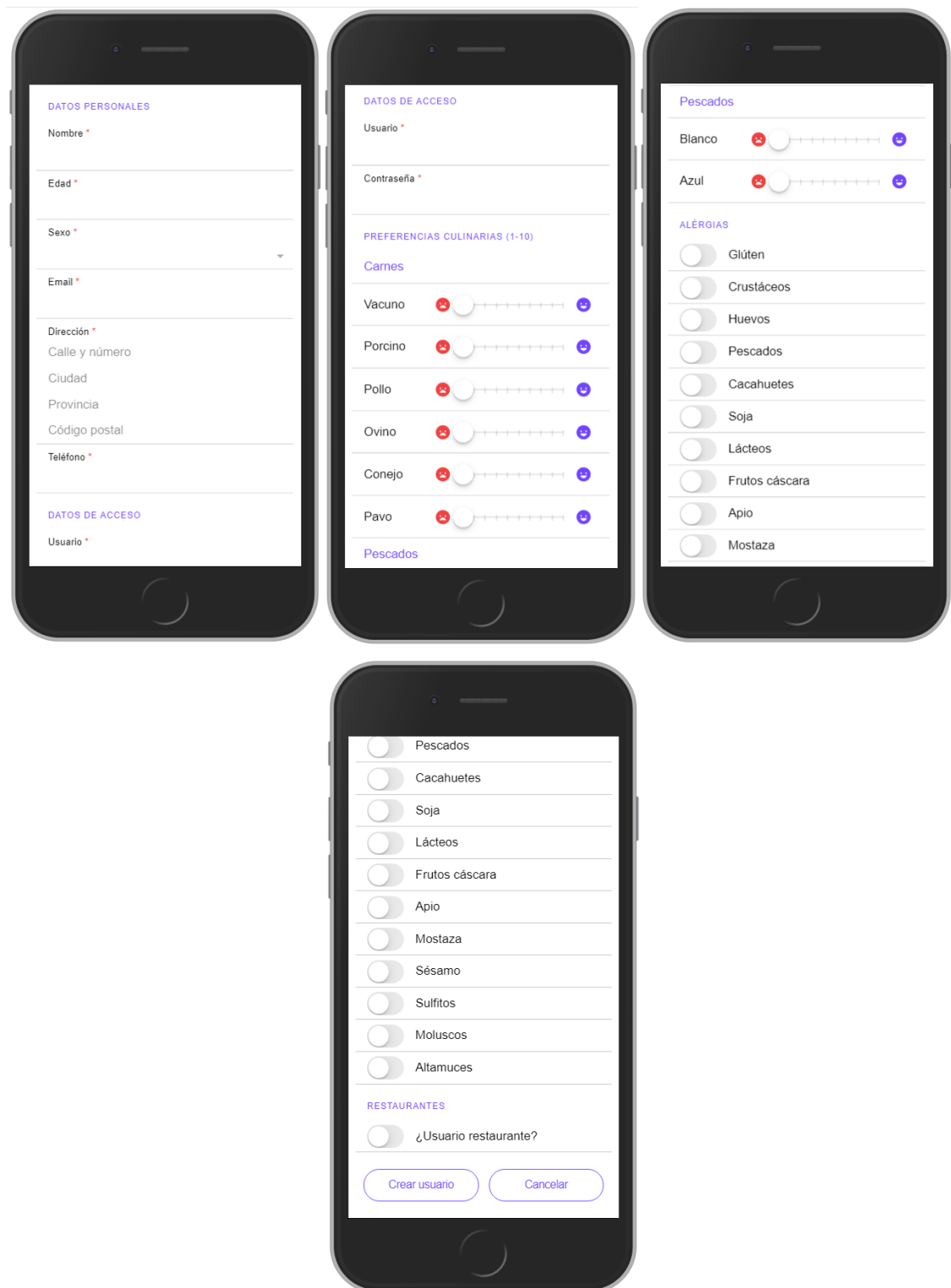


Figura 41. Pantalla registro y modificación de datos usuario

Página principal y menú

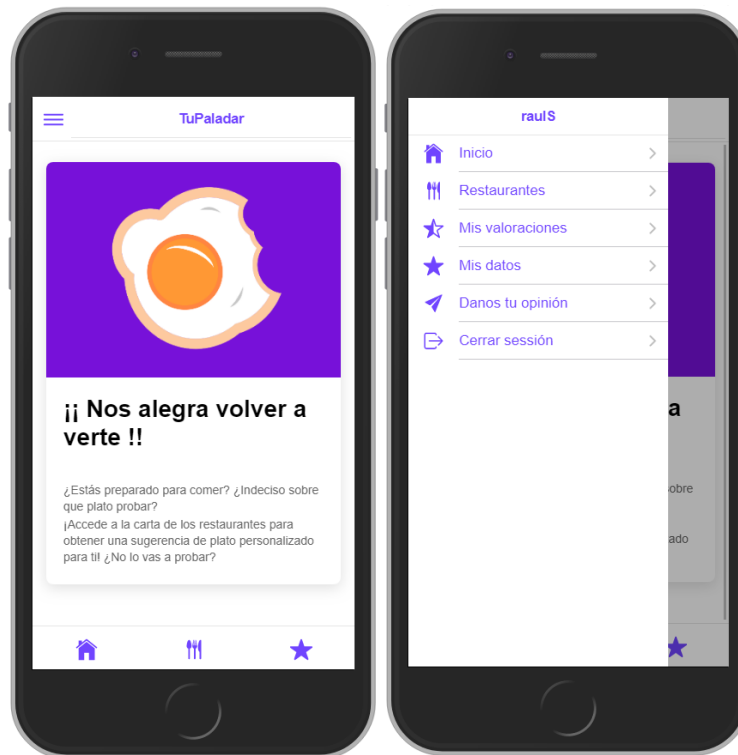


Figura 42. Pantalla principal y menú

Listado restaurantes

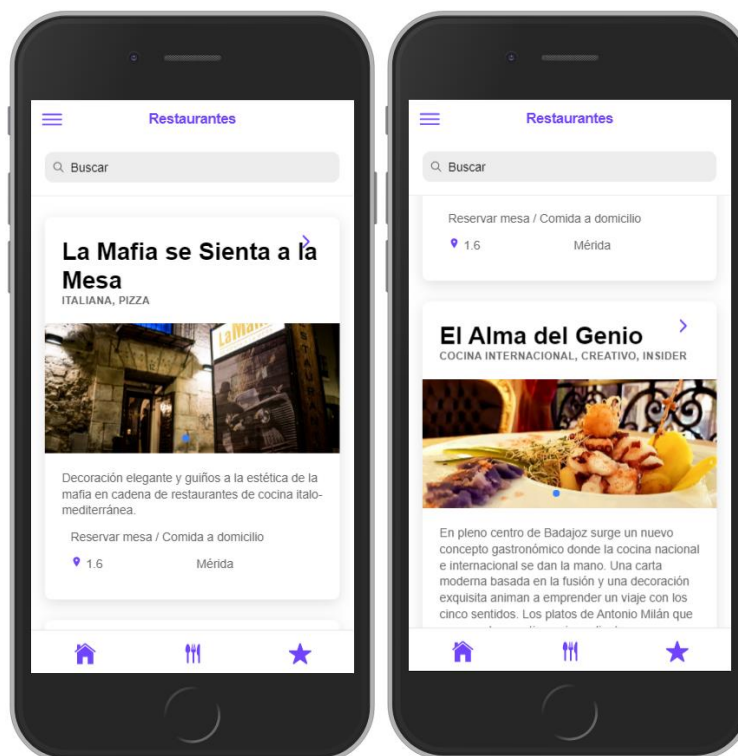


Figura 43. Pantalla listado de restaurantes

Detalle restaurante

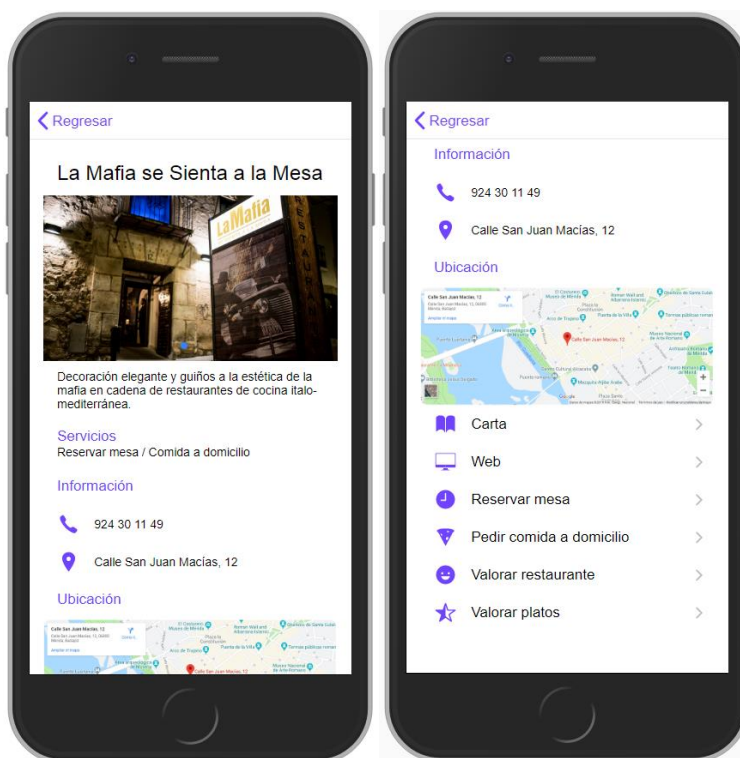


Figura 44. Pantalla detalle restaurante

Listado de platos



Figura 45. Pantalla listado de platos

Reserva de mesa

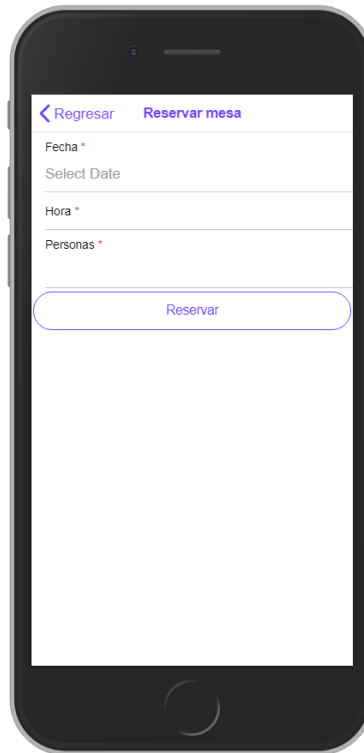


Figura 46. Pantalla reservar mesa

Valorar restaurante

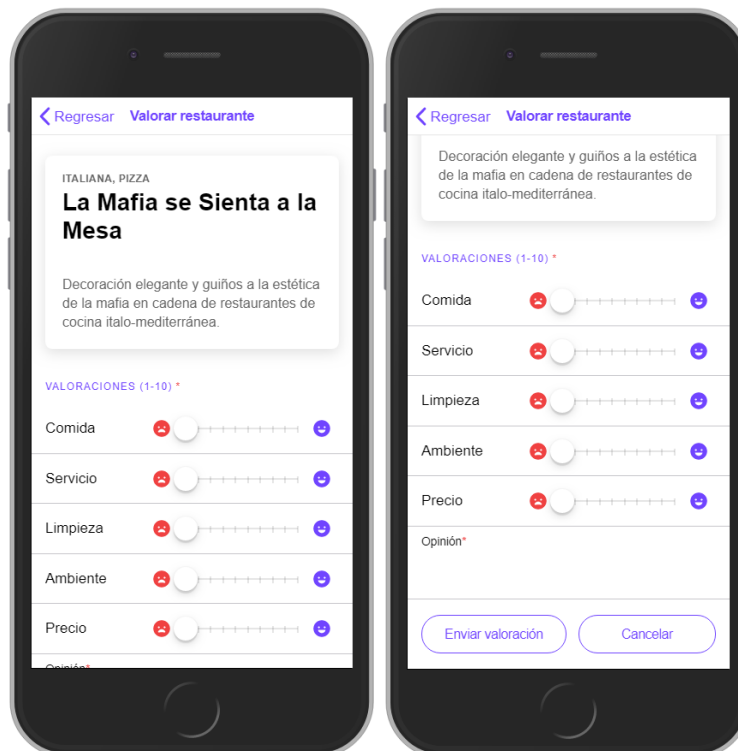


Figura 47. Pantalla valorar restaurante

Valorar plato

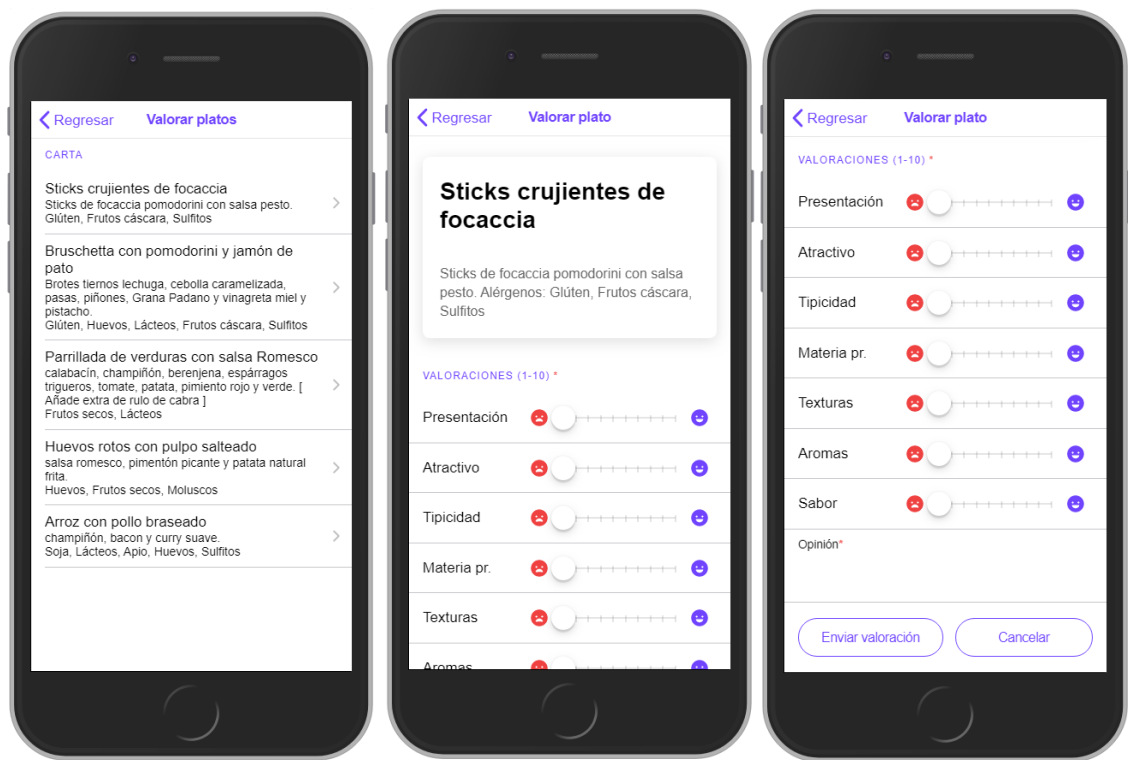


Figura 48. Pantalla valorar plato

Mis valoraciones

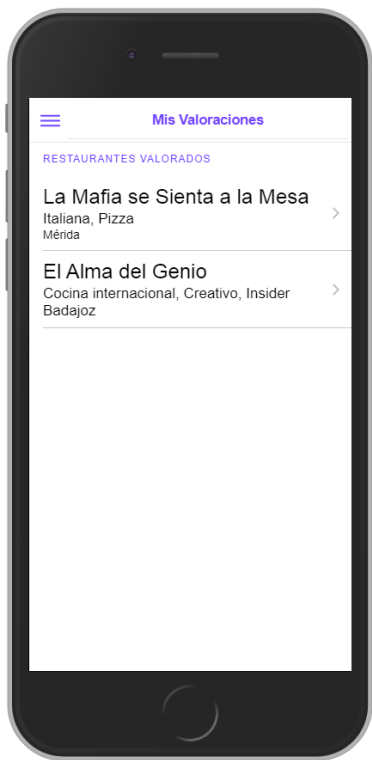


Figura 49. Pantalla mis valoraciones

Inicio usuario restaurante y mis restaurantes



Figura 50. Pantalla inicio usuario restaurante y mis restaurantes

Alta / detalle mis restaurante

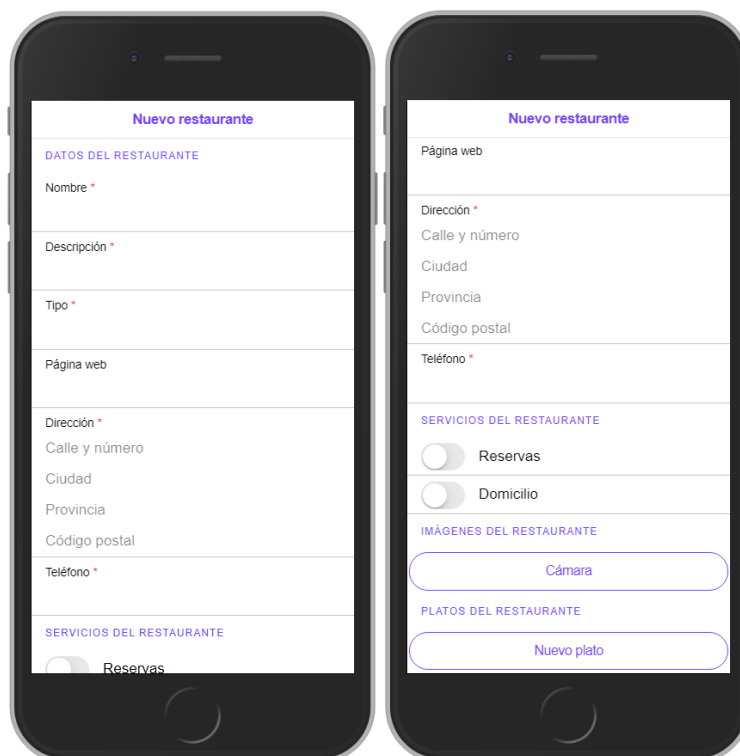


Figura 51. Pantalla alta / detalle restaurante

Alta / detalle del plato

The image shows two smartphones side-by-side, both displaying the 'Nuevo plato' (New dish) form. The left phone shows the 'DATOS DEL PLATO' section with fields for 'Nombre *', 'Descripción *', and 'Precio € *'. The right phone shows the 'CONTENIDO ALÉRGICO' section with a list of allergens and toggle switches. The allergens listed are: Huevos, Pescados, Cacahuets, Soja, Lácteos, Frutos cáscara, Apio, Mostaza, Sésamo, Sulfitos, Moluscos, and Altamuces. At the bottom of the right phone, there are two buttons: 'Guardar plato' and 'Cancelar'.

Figura 52. Pantalla alta / detalle del plato

Sugerencias

The image shows a smartphone displaying the 'Danos tu opinión' (Give us your opinion) form. The form has a title field 'Título del comentario *', a description field 'Descripción *', and a rating scale 'Valora tu experiencia (1-10) *'. The rating scale is a horizontal line with a slider and a plus/minus button. At the bottom, there are two buttons: 'Enviar opinión' and 'Cancelar'.

Figura 53. Pantalla danos tu opinión

Anexo VI. Pruebas de funcionalidad

Tabla 73. Prueba funcional 001 - Login

Prueba	P.F.C.U. 001	
Caso de Uso asociado	C.U. 001	
Nombre	Login	
Descripción	El actor realiza el proceso de identificación para acceder al sistema.	
Resultado esperado	Paso	Acción
	1.	El sistema abre la ventana de Login.
	2.	El sistema abre la ventana de Registro, si el actor selecciona registrarse.
	3.	El sistema valida los datos del formulario. Si son vacíos muestra un mensaje de error. El sistema comprueba que el usuario está registrado en el sistema. Si no está registrado muestra mensaje de error.
	4.	El sistema muestra la pantalla de inicio al usuario. Pestaña home en el caso de un usuario particular, Mis restaurantes en caso de un usuario restaurante.
Estado	Satisfactorio	
Observaciones	-	

(Elaboración propia)

Tabla 74. Prueba funcional 002 - Registrar

Prueba	P.F.C.U. 002
Caso de Uso asociado	C.U. 002
Nombre	Registrar
Descripción	El actor realiza el proceso de registro en el sistema, para poder acceder con posterioridad al mismo.

Resultado esperado	Paso	Acción
	1.	El sistema abre la ventana de Login.
	2.	El sistema abre la ventana de Registro, si el actor selecciona registrarse.
	4.	El sistema valida los datos del formulario. Si son vacíos los campos obligatorios o algún campo es incorrecto muestra un mensaje de error. El sistema comprueba que el usuario no esté registrado en el sistema, validando el email y el nick. Si está registrado muestra mensaje de error.
	5.	El sistema guarda la información y abre la pantalla de inicio al usuario. Pestaña home en el caso de un usuario particular, Mis restaurantes en caso de un usuario restaurante.
Estado	Satisfactorio	
Observaciones	-	

(Elaboración propia)

Tabla 75. Prueba funcional 003 - Gestionar información personal

Prueba	P.F.C.U. 003	
Caso de Uso asociado	C.U. 003	
Nombre	Gestionar información personal	
Descripción	El actor consulta/modifica la su información contenida en el sistema. Además, el actor puede darse de baja en el sistema.	
Resultado esperado	Paso	Acción
	1.	El sistema abre la ventana de Mis datos.
	2.	El sistema busca la información del usuario y carga el formulario de Mis datos.
	5.	El sistema valida los datos del formulario. Si son vacíos los campos obligatorios o algún campo es incorrecto muestra un mensaje de error.

		El sistema guarda la información del usuario y muestra un mensaje de operación correcta.
Estado	Satisfactorio	
Observaciones	-	

(Elaboración propia)

Tabla 76. Prueba funcional 004 - Gestionar valoraciones platos

Prueba	P.F.C.U. 004	
Caso de Uso asociado	C.U. 004	
Nombre	Gestionar valoraciones platos	
Descripción	El actor consulta/modifica la información de valoración de un plato contenida en el sistema. Además, el actor puede darse de baja esa valoración del sistema.	
Resultado esperado	Paso	Acción
	1.	El sistema abre la ventana de Valorar platos.
	2.	El sistema busca la información de las valoraciones de platos del usuario y carga el formulario.
	3.	El sistema abre la ventana Valorar plato.
	4.	El sistema busca la información de la valoración del plato y carga el formulario.
	6.	El sistema valida los datos del formulario. Si son vacíos los campos obligatorios o algún campo es incorrecto muestra un mensaje de error. El sistema guarda la información de la valoración y muestra un mensaje de operación correcta.
Estado	Satisfactorio	
Observaciones	-	

(Elaboración propia)

Tabla 77. Prueba funcional 005 - Gestionar valoraciones restaurantes

Prueba	P.F.C.U. 005	
Caso de Uso asociado	C.U. 005	
Nombre	Gestionar valoraciones restaurantes	
Descripción	El actor consulta/modifica la información de valoración de un restaurante contenida en el sistema. Además, el actor puede darse de baja esa valoración del sistema.	
Resultado esperado	Paso	Acción
	1.	El sistema abre la ventana de Mis valoraciones.
	2.	El sistema busca la información de las valoraciones de los restaurantes del usuario y carga el formulario.
	3.	El sistema abre la ventana Valorar restaurante.
	4.	El sistema busca la información de la valoración del restaurante y carga el formulario.
	6.	El sistema valida los datos del formulario. Si son vacíos los campos obligatorios o algún campo es incorrecto muestra un mensaje de error. El sistema guarda la información de la valoración y muestra un mensaje de operación correcta.
Estado	Satisfactorio	
Observaciones	-	

(Elaboración propia)

Tabla 78. Prueba funcional 006 - Enviar sugerencias

Prueba	P.F.C.U. 006
Caso de Uso asociado	C.U. 006
Nombre	Enviar sugerencias

Descripción	El actor envía sugerencias o feedback con la aplicación, problemas o incidencias que haya tenido al sistema.	
Resultado esperado	Paso	Acción
	1.	El sistema abre la ventana de Danos tu opinión.
	4.	El sistema valida los datos del formulario. Si son vacíos los campos obligatorios o algún campo es incorrecto muestra un mensaje de error. El sistema guarda la información de la opinión y muestra un mensaje de operación correcta.
Estado	Satisfactorio	
Observaciones	-	

(Elaboración propia)

Tabla 79. Prueba funcional 007 - Realizar encuesta

Prueba	P.F.C.U. 007
Caso de Uso asociado	C.U. 007
Nombre	Realizar encuestas
Descripción	El actor realiza el envío de una encuesta en el sistema con el objetivo de la mejora de la base de conocimiento del algoritmo predictivo.
Resultado esperado	-
Estado	No aplica
Observaciones	Esta funcionalidad no se ha desarrollado en esta versión de TuPaladar.

(Elaboración propia)

Tabla 80. Prueba funcional 008 - Valorar plato

Prueba	P.F.C.U. 008
Caso de Uso asociado	C.U. 008
Nombre	Valorar platos

Descripción	El actor puntúa su valoración de un plato.	
Resultado esperado	Paso	Acción
	1.	El sistema abre la ventana de Valora plato.
	4.	El sistema valida los datos del formulario. Si son vacíos los campos obligatorios o algún campo es incorrecto muestra un mensaje de error. El sistema guarda la información de la valoración y muestra un mensaje de operación correcta.
Estado	Satisfactorio	
Observaciones	-	

(Elaboración propia)

Tabla 81. Prueba funcional 009 - Valorar restaurantes

Prueba	P.F.C.U. 009	
Caso de Uso asociado	C.U. 009	
Nombre	Valorar restaurantes	
Descripción	El actor puntúa y valora un restaurante.	
Resultado esperado	Paso	Acción
	1.	El sistema abre la ventana de Valora restaurante.
	4.	El sistema valida los datos del formulario. Si son vacíos los campos obligatorios o algún campo es incorrecto muestra un mensaje de error. El sistema guarda la información de la valoración y muestra un mensaje de operación correcta.
	6.	El sistema abre la ventana de Valorar plato.
Estado	Satisfactorio	
Observaciones	-	

(Elaboración propia)

Tabla 82. Prueba funcional 010 - Reservar restaurante

Prueba	P.F.C.U. 010	
Caso de Uso asociado	C.U. 010	
Nombre	Reservar restaurante	
Descripción	El actor reserva mesa en un restaurante.	
Resultado esperado	Paso	Acción
	1.	El sistema abre la ventana de Reservar mesa.
	4.	El sistema valida los datos del formulario. Si son vacíos los campos obligatorios o algún campo es incorrecto muestra un mensaje de error. El sistema guarda la información de la reserva y muestra un mensaje de con el resultado de la reserva. Si no hay disponibilidad se solicita que seleccione otro hora o día.
Estado	Satisfactorio	
Observaciones	-	

(Elaboración propia)

Tabla 83. Prueba funcional 011 - Pedir comida

Prueba	P.F.C.U. 011
Caso de Uso asociado	C.U. 011
Nombre	Pedir comida
Descripción	El actor pide comida a domicilio de un restaurante.
Resultado esperado	-
Estado	No aplica
Observaciones	Esta funcionalidad no se ha desarrollado en esta versión de TuPaladar.

(Elaboración propia)

Tabla 84. Prueba funcional 012 - Consultar carta restaurante

Prueba	P.F.C.U. 012	
Caso de Uso asociado	C.U. 012	
Nombre	Consultar carta restaurante	
Descripción	Para cada restaurante el actor puede consultar la información y carta de platos del restaurante.	
Resultado esperado	Paso	Acción
	1.	El sistema abre la ventana de Platos restaurante.
	2.	El sistema busca la información de la cara del restaurante y la carga en el formulario. El sistema aplica el proceso predictivo de sugerencia de platos y carga el plato sugerido en el formulario. El sistema procesa los platos de la carta con contenidos alérgenos perjudicial para el usuario y carga la información en el formulario.
Estado	Satisfactorio	
Observaciones	-	

(Elaboración propia)

Tabla 85. Prueba funcional 013 - Localizar restaurantes

Prueba	P.F.C.U. 013
Caso de Uso asociado	C.U. 013
Nombre	Localizar restaurantes
Descripción	El actor puede localizar restaurantes en un mapa mediante geolocalización.
Resultado esperado	-
Estado	No aplica

Observaciones	Esta funcionalidad no se ha desarrollado en esta versión de TuPaladar.
----------------------	--

(Elaboración propia)

Tabla 86. Prueba funcional 014 - Buscar restaurantes

Prueba	P.F.C.U. 014	
Caso de Uso asociado	C.U. 014	
Nombre	Buscar restaurantes	
Descripción	El actor puede buscar restaurante en el sistema.	
Resultado esperado	Paso	Acción
	1.	El sistema abre la ventana de Restaurantes.
	3.	El sistema busca en los títulos o en la localidad de la lista de restaurantes y recarga el resultado obtenido en el formulario.
Estado	Satisfactorio	
Observaciones	-	

(Elaboración propia)

Tabla 87. Prueba funcional 015 - Alta plato

Prueba	P.F.C.U. 015	
Caso de Uso asociado	C.U. 015	
Nombre	Alta plato	
Descripción	El actor da de alta un nuevo plato a uno de sus restaurantes.	
Resultado esperado	Paso	Acción
	1.	El sistema abre la ventana de Nuevo plato.
	4.	El sistema valida los datos del formulario. Si son vacíos los campos obligatorios o algún campo es incorrecto muestra un mensaje de error. El sistema guarda la información del plato y muestra un mensaje de operación correcta y abre la ventana Detalle restaurante.

Estado	Satisfactorio
Observaciones	-

(Elaboración propia)

Tabla 88. Prueba funcional 016 - Alta restaurantes

Prueba	P.F.C.U. 016	
Caso de Uso asociado	C.U. 016	
Nombre	Alta restaurantes	
Descripción	El actor da de alta un nuevo plato a uno de sus restaurantes.	
Resultado esperado	Paso	Acción
	1.	El sistema abre la ventana de Nuevo restaurante.
	4.	El sistema valida los datos del formulario. Si son vacíos los campos obligatorios o algún campo es incorrecto muestra un mensaje de error. El sistema guarda la información del restaurante y muestra un mensaje de operación correcta.
Estado	Satisfactorio	
Observaciones	-	

(Elaboración propia)

Tabla 89. Prueba funcional 017 - Gestionar platos

Prueba	P.F.C.U. 017	
Caso de Uso asociado	C.U. 017	
Nombre	Gestionar platos	
Descripción	El actor consulta/modifica la información de los platos que aparecen en sus restaurantes en el sistema. El actor puede dar de baja un plato en el sistema.	
	Paso	Acción

Resultado esperado	1.	El sistema abre la ventana de Platos restaurante.
	2.	El sistema busca la información de los platos del restaurante y carga el formulario.
	3.	El sistema abre la ventana Detalle plato.
	4.	El sistema busca la información del plato y carga el formulario. El sistema muestra un mensaje de confirmación de baja del plato al usuario.
	5.	El sistema valida los datos del formulario. Si son vacíos los campos obligatorios o algún campo es incorrecto muestra un mensaje de error. El sistema guarda la información del plato y muestra un mensaje de operación correcta. El sistema abre la ventana Platos restaurante.
	7.	El sistema da de baja el plato y muestra un mensaje de operación correcta. El sistema abre la ventana Platos restaurante.
Estado	Satisfactorio	
Observaciones	-	

(Elaboración propia)

Tabla 90. Prueba funcional 018 - Gestionar restaurantes

Prueba	P.F.C.U. 018	
Caso de Uso asociado	C.U. 018	
Nombre	Gestionar restaurantes	
Descripción	El actor consulta/modifica la información de un restaurantes en el sistema. El actor puede dar de baja el restaurante en el sistema.	
Resultado esperado	Paso	Acción
	1.	El sistema abre la ventana de Mis restaurantes.
	2.	El sistema busca la información de los restaurante del usuario y carga el formulario.

	3.	El sistema abre la ventana Detalle restaurante.
	4.	El sistema busca la información del restaurante y carga el formulario. El sistema muestra un mensaje de confirmación de baja del restaurante al usuario.
	5.	El sistema valida los datos del formulario. Si son vacíos los campos obligatorios o algún campo es incorrecto muestra un mensaje de error. El sistema guarda la información del restaurante y muestra un mensaje de operación correcta. El sistema abre la ventana Mis restaurantes.
	7.	El sistema da de baja el restaurante y muestra un mensaje de operación correcta. El sistema abre la ventana Mis restaurantes.
Estado	Satisfactorio	
Observaciones	-	

(Elaboración propia)

Anexo VII. Pruebas de usabilidad

Usuario 1

Datos Usuario: Hombre, 54 años y vive en Mérida.

Tarea 1. Registro, login y logout

Esta tarea consiste en la realización del registro en la aplicación *TuPaladar* y el posterior acceso al sistema con el usuario creado. Los datos introducidos en el formulario de registro pueden ser personales o ficticios. Posteriormente tiene que cerrar sesión en la aplicación.

Medidas a evaluar

Eficiencia:

- Tiempo total en la realización de la tarea. **168 seg.**
- Tiempo total en el registro. **152 seg.**
- Tiempo total en login. **10 seg.**
- Tiempo total en logout. **6 seg.**

Eficacia:

- Ha conseguido introducir los datos personales. **Si.**
- Ha conseguido introducir una contraseña segura. **Si.**
- Ha conseguido registrarse. **Si.**
- Ha conseguido acceder a la aplicación con el usuarios creado. **Si.**
- Ha conseguido cerrar sesión en la aplicación. **Si.**
- Número de errores:
 - Contraseña segura: 3 errores de validación en la contraseña.
 - Email: 1 error en la validación del email.

Observaciones:

Donde más se ha tardado en el registro es en la cumplimentación del domicilio.

Tarea 2. Búsqueda de restaurantes y platos, y reserva de mesa

Esta tarea consiste en buscar el restaurante *La Mafía se sienta a la Mesa* de Mérida, acceder a su carta para la visualización de platos y posteriormente simular una reserva de mesa en el restaurante.

Medidas a evaluar

Eficiencia:

- Tiempo total en la realización de la tarea. **57 seg.**
- Tiempo total en localizar el restaurante. **20 seg.**
- Tiempo total en localizar la carta del restaurante. **3 seg.**
- Tiempo total en visualizar y entender la carta. **22 seg.**

- Tiempo en localizar la reserva de mesa. **2 seg.**
- Tiempo total en reservar mesa. **10 seg.**

Eficacia:

- Ha conseguido localizar el restaurante. **Si.**
- Ha conseguido localizar la carta del restaurante. **Si.**
- Ha conseguido localizar la reserva de mesa. **Si.**
- Ha conseguido reservar mesa. **Si.**
- Número de errores: **0**

Observaciones:

Sin ningún tipo de problema con la familiarización con el detalle del restaurante.

Tarea 3. Valoración de un restaurante y un plato del restaurante

Esta tarea consiste en buscar el restaurante *La Mafia se sienta a la Mesa* de Mérida, valorar la satisfacción con el restaurante, bajo la suposición de haber comido en él, y, además, valorar el plato *Sticks crujientes de focaccia*.

Medidas a evaluar**Eficiencia:**

- Tiempo total en realizar la tarea. **46 seg.**
- Tiempo total en localizar el restaurante. **5 seg.**
- Tiempo total en localizar valorar restaurante. **2 seg.**
- Tiempo total en valorar el restaurante. **12 seg.**
- Tiempo total en localizar valorar plato. **2 seg.**
- Tiempo total en localizar el plato. **10 seg.**
- Tiempo total en valorar el plato. **15 seg.**

Eficacia:

- Ha conseguido localizar el restaurante. **Si.**
- Ha conseguido valorar el restaurante. **Si.**
- Ha conseguido localizar valorar platos. **Si.**
- Ha conseguido localizar el plato. **Si.**
- Ha conseguido valorar el plato. **Si.**
- Número de errores:
 - **Valorar restaurante: 1. No había cumplimentado el campo de observaciones.**

Observaciones:

Sin observaciones relevantes.

Satisfacción:

16. En general estoy satisfecho con la facilidad de realizar las tareas en la aplicación, donde 1 es poco satisfecho y 5 es muy satisfecho:

1 2 3 4 5

17. En general estoy satisfecho con el tiempo tardado en realizar las tareas en la aplicación, donde 1 es poco satisfecho y 5 es muy satisfecho:

1 2 3 4 5

18. En general estoy satisfecho con los mensajes de aviso realizar las tareas en la aplicación, donde 1 es poco satisfecho y 5 es muy satisfecho:

1 2 3 4 5

19. Satisfacción general con la aplicación:

1 2 3 4 5

Usuario 2

Datos Usuario: Mujer, 48 años y vive en Mérida.

Tarea 1. Registro, login y logout

Esta tarea consiste en la realización del registro en la aplicación *TuPaladar* y el posterior acceso al sistema con el usuario creado. Los datos introducidos en el formulario de registro pueden ser personales o ficticios. Posteriormente tiene que cerrar sesión en la aplicación.

Medidas a evaluar

Eficiencia:

- Tiempo total en la realización de la tarea. **180 seg.**
- Tiempo total en el registro. **170 seg.**
- Tiempo total en login. **7 seg.**
- Tiempo total en logout. **3 seg.**

Eficacia:

- Ha conseguido introducir los datos personales. **Si.**
- Ha conseguido introducir una contraseña segura. **Si.**
- Ha conseguido registrarse. **Si.**
- Ha conseguido acceder a la aplicación con el usuarios creado. **Si.**
- Ha conseguido cerrar sesión en la aplicación. **Si.**
- Número de errores:
 - **Contraseña segura: 1 errores de validación en la contraseña.**

Observaciones:

En las valoraciones de los ingredientes se ha realizado una pausa.

Tarea 2. Búsqueda de restaurantes y platos, y reserva de mesa

Esta tarea consiste en buscar el restaurante *La Mafia se sienta a la Mesa* de Mérida, acceder a su carta para la visualización de platos y posteriormente simular una reserva de mesa en el restaurante.

Medidas a evaluar

Eficiencia:

- Tiempo total en la realización de la tarea. **62 seg.**
- Tiempo total en localizar el restaurante. **15 seg.**
- Tiempo total en localizar la carta del restaurante. **4 seg.**
- Tiempo total en visualizar y entender la carta. **25 seg.**
- Tiempo en localizar la reserva de mesa. **3 seg.**
- Tiempo total en reservar mesa. **15 seg.**

Eficacia:

- Ha conseguido localizar el restaurante. **Si.**
- Ha conseguido localizar la carta del restaurante. **Si.**
- Ha conseguido localizar la reserva de mesa. **Si.**
- Ha conseguido reservar mesa. **Si.**
- Número de errores: **0**

Observaciones:

Sin observaciones.

Tarea 3. Valoración de un restaurante y un plato del restaurante

Esta tarea consiste en buscar el restaurante *La Mafia se sienta a la Mesa* de Mérida, valorar la satisfacción con el restaurante, bajo la suposición de haber comido en él, y, además, valorar el plato *Sticks crujientes de focaccia*.

Medidas a evaluar

Eficiencia:

- Tiempo total en realizar la tarea. **49 seg.**
- Tiempo total en localizar el restaurante. **4 seg.**
- Tiempo total en localizar valorar restaurante. **4 seg.**
- Tiempo total en valorar el restaurante. **10 seg.**
- Tiempo total en localizar valorar plato. **4 seg.**
- Tiempo total en localizar el plato. **7 seg.**
- Tiempo total en valorar el plato. **20 seg.**

Eficacia:

- Ha conseguido localizar el restaurante. **Si.**
- Ha conseguido valorar el restaurante. **Si.**
- Ha conseguido localizar valorar platos. **Si.**

- Ha conseguido localizar el plato. **Si.**
- Ha conseguido valorar el plato. **Si.**
- Número de errores: **0**

Observaciones:

Sin observaciones relevantes.

Satisfacción:

20. En general estoy satisfecho con la facilidad de realizar las tareas en la aplicación, donde 1 es poco satisfecho y 5 es muy satisfecho:

1 2 3 4 **5**

21. En general estoy satisfecho con el tiempo tardado en realizar las tareas en la aplicación, donde 1 es poco satisfecho y 5 es muy satisfecho:

1 2 3 4 **5**

22. En general estoy satisfecho con los mensajes de aviso realizar las tareas en la aplicación, donde 1 es poco satisfecho y 5 es muy satisfecho:

1 2 3 4 **5**

23. Satisfacción general con la aplicación:

1 2 3 4 **5**

Usuario 3

Datos Usuario: Mujer, 29 años y vive en Mérida.

Tarea 1. Registro, login y logout

Esta tarea consiste en la realización del registro en la aplicación *TuPaladar* y el posterior acceso al sistema con el usuario creado. Los datos introducidos en el formulario de registro pueden ser personales o ficticios. Posteriormente tiene que cerrar sesión en la aplicación.

Medidas a evaluar**Eficiencia:**

- Tiempo total en la realización de la tarea. **134 seg.**
- Tiempo total en el registro. **120 seg.**
- Tiempo total en login. **10 seg.**
- Tiempo total en logout. **4 seg.**

Eficacia:

- Ha conseguido introducir los datos personales. **Si.**
- Ha conseguido introducir una contraseña segura. **Si.**
- Ha conseguido registrarse. **Si.**

- Ha conseguido acceder a la aplicación con el usuarios creado. **Si.**
- Ha conseguido cerrar sesión en la aplicación. **Si.**
- Número de errores: **0**

Observaciones:

Se destaca la fluidez al realizar el registro.

Tarea 2. Búsqueda de restaurantes y platos, y reserva de mesa

Esta tarea consiste en buscar el restaurante *La Mafia se sienta a la Mesa* de Mérida, acceder a su carta para la visualización de platos y posteriormente simular una reserva de mesa en el restaurante.

Medidas a evaluar**Eficiencia:**

- Tiempo total en la realización de la tarea. **36 seg.**
- Tiempo total en localizar el restaurante. **10 seg.**
- Tiempo total en localizar la carta del restaurante. **2 seg.**
- Tiempo total en visualizar y entender la carta. **10 seg.**
- Tiempo en localizar la reserva de mesa. **3 seg.**
- Tiempo total en reservar mesa. **11 seg.**

Eficacia:

- Ha conseguido localizar el restaurante. **Si.**
- Ha conseguido localizar la carta del restaurante. **Si.**
- Ha conseguido localizar la reserva de mesa. **Si.**
- Ha conseguido reservar mesa. **Si.**
- Número de errores: **0**

Observaciones:

Sin observaciones.

Tarea 3. Valoración de un restaurante y un plato del restaurante

Esta tarea consiste en buscar el restaurante *La Mafia se sienta a la Mesa* de Mérida, valorar la satisfacción con el restaurante, bajo la suposición de haber comido en él, y, además, valorar el plato *Sticks crujientes de focaccia*.

Medidas a evaluar**Eficiencia:**

- Tiempo total en realizar la tarea. **35 seg.**
- Tiempo total en localizar el restaurante. **3 seg.**
- Tiempo total en localizar valorar restaurante. **2 seg.**
- Tiempo total en valorar el restaurante. **8 seg.**
- Tiempo total en localizar valorar plato. **2 seg.**

- Tiempo total en localizar el plato. **10 seg.**
- Tiempo total en valorar el plato. **15 seg.**

Eficacia:

- Ha conseguido localizar el restaurante. **Si.**
- Ha conseguido valorar el restaurante. **Si.**
- Ha conseguido localizar valorar platos. **Si.**
- Ha conseguido localizar el plato. **Si.**
- Ha conseguido valorar el plato. **Si.**
- Número de errores:

Observaciones:

Sin observaciones relevantes.

Satisfacción:

24. En general estoy satisfecho con la facilidad de realizar las tareas en la aplicación, donde 1 es poco satisfecho y 5 es muy satisfecho:

1 2 3 4 **5**

25. En general estoy satisfecho con el tiempo tardado en realizar las tareas en la aplicación, donde 1 es poco satisfecho y 5 es muy satisfecho:

1 2 3 4 **5**

26. En general estoy satisfecho con los mensajes de aviso realizar las tareas en la aplicación, donde 1 es poco satisfecho y 5 es muy satisfecho:

1 2 **3** 4 5

27. Satisfacción general con la aplicación:

1 2 3 **4** 5

