PC-Drome

ARES 3.0

Fecha de entrega

9 de marzo de 2021

Versión 1.0



Autores

Bryan Felipe Santillan Gómez

Diana Natalia Chaparro Sanabria

Diana Stefania García Yomayuza

Edwin Fabian Vesga Escobar

Fabian Andrés Rojas Molina

Jerson Giovany Osorio Cely

Julián David Tarazona Torres

# HISTORIAL DE CAMBIOS.

|  |  |  |  |
| --- | --- | --- | --- |
| Numero de cambio | Fecha del cambio | Descripción del cambio | Persona que realizó el cambio |
| 1 | 28/02/2021 | Se agregaron las características del usuario | Diana Chaparro |
| 2 | 28/02/2021 | Se agregan la información relacionada con los lenguajes y las herramientas para el correcto desarrollo del proyecto. | Edwin Vesga |
| 3 | 01/03/2021 | Se agregó la propuesta del proyecto con el modelo canvas | Stefania García |
| 4 | 01/03/2021 | Se agregaron las historias de usuario y los mockups. | Julián Tarazona |
| 5 | 02/03/2021 | Se agregaron criterios y modelo de ciclo de vida | Fabian Rojas |
| 6 | 03/03/2021 | Se agregaron los requisitos no funcionales | Jerson Osorio |

Tabla Historial de cambios

# TABLA DE CONTENIDOS

[1. HISTORIAL DE CAMBIOS. 2](#_Toc65759226)

[2. TABLA DE CONTENIDOS 3](#_Toc65759227)

[3. LISTA DE FIGURAS 4](#_Toc65759228)

[4. LISTA DE TABLAS 5](#_Toc65759229)

[5. INTRODUCCIÓN 6](#_Toc65759230)

[6. PROPUESTA DEL PROYECTO 7](#_Toc65759231)

[7. MODELO DE CICLO DE VIDA 8](#_Toc65759232)

[8. EQUIPO DE TRABAJO Y DISTRIBUCIÓN DE ROLES 8](#_Toc65759233)

[9. LENGUAJES Y HERRAMIENTAS 8](#_Toc65759234)

[9.1 Herramientas de equipo. 9](#_Toc65759235)

[9.2 Front-End. 9](#_Toc65759236)

[9.3 Back-End. 10](#_Toc65759237)

[10. FUNCIONES DEL PRODUCTO 10](#_Toc65759238)

[11. CARACTERÍSTICAS DEL USUARIO 13](#_Toc65759239)

[12. REQUISITOS NO FUNCIONALES 14](#_Toc65759240)

[13. PLAN DE TRABAJO DEL PROYECTO 14](#_Toc65759241)

[14. MÉTODOS Y HERRAMIENTAS DE ESTIMACIÓN 15](#_Toc65759242)

[15. CONCLUSIONES 15](#_Toc65759243)

[16. ANEXOS 15](#_Toc65759244)

[17. REFERENCIAS 16](#_Toc65759245)

# LISTA DE FIGURAS

[Figura 1 Tabla que describe el modelo canvas del proyecto. 7](#_Toc65759258)

[Figura 2 asignación de iteraciones de las historias de usuario. 11](#_Toc65759259)

[Figura 3 pantalla de inicio de PCDROME 13](#_Toc65759260)

# LISTA DE TABLAS

[Tabla 1 Historial de cambios 2](#_Toc65759269)

[Tabla 2 puntos de estimación según el riesgo de desarrollo 11](#_Toc65759270)

[Tabla 3 Historia de usuario del cliente 12](#_Toc65759271)

[Tabla 4 Descripción de las características del usuario cliente 13](#_Toc65759272)

[Tabla 5 Descripción de las características del usuario administrador 14](#_Toc65759273)

# INTRODUCCIÓN

En este documento, se encuentran descritos todos los aspectos para esta primera mirada a el proyecto del curso de *Fundamentos de ingeniería de software.*

# PROPUESTA DEL PROYECTO

PC-DROME ofrece a las personas la alternativa de conseguir las partes de un computador y armarlo de acuerdo con el precio que esté dispuesto a pagar y las características que esté buscando, teniendo en cuenta que no se debe conocer mucho del tema para tener el equipo que desee. PC-DROME ofrece al usuario ensamble y mantenimiento, donde si el cliente no conoce mucho del tema se le ofrecerá un asesoramiento para armar un buen computador teniendo en cuenta sus especificaciones; Por último, la plataforma permite ingresar las características del dispositivo para mostrar al usuario las mejoras que se pueden realizar y sus costos. Como producto de software se realizará una página web que contenga todo lo propuesto anteriormente.

Para representar la información clave del proyecto se utilizó como referencia el modelo canvas presentado en la Figura 1 de este documento.

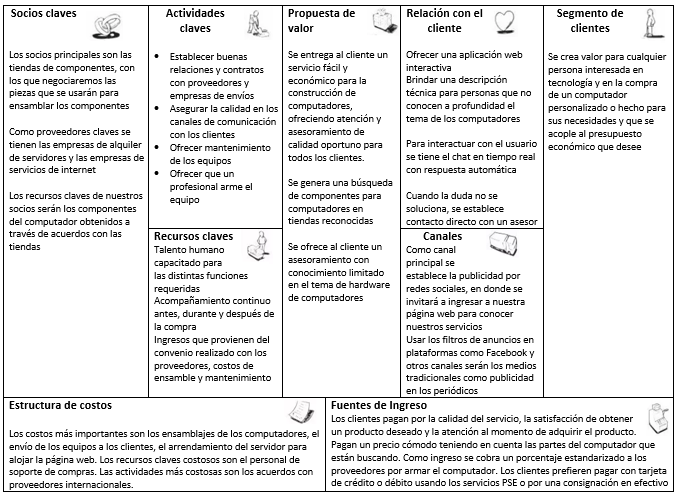


Figura 1 Tabla que describe el modelo canvas del proyecto.

# MODELO DE CICLO DE VIDA

**Criterios:**

Queremos un modelo de ciclo de vida:

1. El cual describa sus etapas o fases de forma sencilla y detallada.
2. Con el que estemos más familiarizados y tengamos experiencia en los procesos que pide.
3. Que sea sencillo de entender, de trabajar y que a la vez se efectivo para un proyecto.
4. Un modelo que sea reconocido por muchas personas en el área de la ingeniería de software.
5. Un modelo que en los procesos iniciales se pueda realizar la especificación de los requisitos y el diseño del sistema.
6. Un modelo que permita especificar qué actividades hay que hacer, en qué orden y cuales dependen de otras para poder iniciarlas.

Teniendo en cuenta los criterios para el ciclo de vida, el que más se acoplaría a ellos es el *Modelo de ciclo de vida en Cascada,* ya que cumple con la mayoría de ellos.

Con respecto al criterio 1, el modelo en cascada lo cumple perfectamente, ya que este modelo define 5 fases principales que tienen actividades específicas para cada una, estas fases son: Análisis de requisitos, Diseño del sistema, Implementación, Verificación o pruebas del sistema y finalmente el Mantenimiento (Ionos, 2019).Por otro lado, varios de los integrantes del equipo estamos familiarizados con el orden en el que se emplea el modelo de ciclo de vida, y con la mayoría de los procesos que este emplea, como el análisis de requisitos, diseño del sistema, la implementación y pruebas del sistema, por lo que este modelo cumple también el segundo criterio.

# EQUIPO DE TRABAJO Y DISTRIBUCIÓN DE ROLES

# LENGUAJES Y HERRAMIENTAS

En este apartado, definiremos los lenguajes de programación y las herramientas más importantes para el correcto desarrollo de este proyecto de software. Estos, serán separados en 3 secciones: *Herramientas de equipo,* *Front-End y Back-End*. Primero, veamos que son las *Herramientas de equipo.* Estas son aquellas herramientas que nos permitirán sincronizar nuestro trabajo, documentar el proyecto y publicar cambios o actividades. En *Front-End* encontraremos las herramientas que facilitaran la maquetación visual de nuestro software y también organizar estos para tener una idea de cómo será la experiencia de usuario. También este apartado habla de los lenguajes de codificación de estilos. Los lenguajes para desarrollar la interfaz gráfica y hacer el llamado a él Back-End En cuanto al *Back-End,* nos referimos a todo lo que tenga que ver con el control de los datos de cara a el servidor.

## 9.1 Herramientas de equipo.

1. GIT.

Dentro de los proyectos de software, el sistema GIT es una de las herramientas que se vuelve fundamental a la hora de sincronizar el trabajo en grupo. Este sistema es un controlador y revisor de versiones para documentos y archivos involucrados en un proyecto. Muchos IDE’s incluyen dentro de sus funcionalidades, sistemas gráficos de control de versiones con GIT. Además, GIT es el sistema base para sincronizar proyectos en plataformas de repositorios como GITHUB O GITLAB. Esta herramienta será fundamental para sincronizar las versiones de nuestro software.

1. GITHUB.

Github una plataforma para la carga y gestión de proyectos de software basándose en el sistema GIT. Permite guardar de manera remota un proyecto, agregar participantes a el proyecto, gestionar roles, gestionar tareas y licenciar de manera genérica nuestros proyectos de software. Github será el espacio donde encontraremos el código fuente de PC-Drome, además de una presentación de todos los integrantes de este equipo dentro de la Wiki.

1. MIRO.

Miro, es una plataforma online para la creación de modelos visuales que permitan la sincronización entre los integrantes del equipo. Permite crear varios espacios visuales que van desde planificadores hasta gráficas y mapas de ideas. Es una plataforma maravillosa para plantear ideas de grupo, organizar diagramas y organizar las tareas del equipo con la posibilidad de ver cómodamente la información.

## 9.2 Front-End.

1. Adobe XD.

Adobe XD, es un software de diseño y maquetación de interfaces, además de permitir un prototipado interactivo de estas para analizar y mejorar la experiencia de usuario. Es un software de la empresa Adobe Systems Inc. y posee una versión gratuita y otra de pago. Ambas, pueden usarse para proyectos de cualquier tipo, solo que la de pago ofrece una mayor gama de herramientas. Un plus de esta herramienta es que el diseño al ser exportado para visualización muestra algunos componentes con su código CSS para ser usado posteriormente en el proyecto.

1. HTML5, CSS y JavaScript.

Esta es la triada fundamental del desarrollo web. Primeramente, tenemos el Lenguaje de marcado de hipertexto, el cual nos permite crear los documentos que mostraran la información en nuestra página web. CSS, que es el lenguaje de codificación de estilos para nuestra página web, permitiendo modificar aspectos como el color, el tamaño y la ubicación de los componentes de nuestra página. Por último, tenemos a JavaScript, un lenguaje de programación que nos permite controlar la interacción con los componentes de la página, así como hacer llamadas a él Back-End de nuestra aplicación web.

1. Angular.

Angular, es un framework basado en TypeScript (Un superconjunto de JavaScript) que permite al igual que JavaScript, la sincronización entre interfaz y servidor. Además, está enfocado a las PWA, es decir, *Progressive Web Application.* Este tipo de aplicaciones web ofrecen alto rendimiento, además de algunos aspectos sin conexión que las hace tener características similares a las aplicaciones nativas. Todo esto, compatible con todos los navegadores modernos.

## 9.3 Back-End.

1. Firebase y base de datos no relacional.

Firebase es una plataforma en la nube para el desarrollo de aplicaciones web y móvil. Es un servicio de la empresa Google y brinda grandes ventajas. Entre las más relevantes encontramos hosting, almacenamiento en la nube, autenticación de usuarios y su real time database, la cual es una base de datos no relacional que almacena la información con archivos JSON. Esta base de datos es muy buena ya que nos permite atender múltiples transacciones sin bloqueos. Además, permite almacenar los errores para hacer una revisión y reparación oportuna de estos. Todo, desde el mismo dashboard.

1. SQL y base de datos relacional (Opción B).

SQL es un lenguaje estructurado de consultas para la gestión de informacion dentro de una base de datos relacional. Es el modelo de bases de datos más usado en la actualidad, por su rigurosidad y seguridad, pero si el diseño relacional está mal planteado, puede generar bloqueos a la hora de buscar información. Como plus, todos dentro de nuestro grupo han usado este lenguaje y creado bases de datos relacionales.

# FUNCIONES DEL PRODUCTO

Las funciones del producto son las características que le otorgaran valor a PCDROME y para ello hemos encapsulado las funciones que debe cumplir nuestra página web en historias de usuarios y las hemos analizado para determinar que prioridad tienen en el negocio, que tan complicado será implementar esa historia y en qué momento del desarrollo de la aplicación web será implementada.

Para la priorización en el negocio se utiliza la técnica MoSCoW para determinar el valor que la historia de usuario aporta al software y el efecto que este tendrá dentro del producto final (Menzinsky, 2018).

Para calcular el riesgo en desarrollo utilizaremos una escala de 1 a 5, siendo 1 la historia de usuario con el riesgo de desarrollo más bajo y el 5 siendo la historia con el riesgo de desarrollo más alto.

Para los puntos estimados tendremos en cuenta riesgo en desarrollo que se le coloque a cada historia de usuario. Los valores se presentan en la siguiente tabla:

|  |  |  |
| --- | --- | --- |
|  | Puntos de riesgo en desarrollo | Puntos estimados |
| Muy difícil | 5 | 25 |
| Difícil | 4 | 20 |
| Moderado | 3 | 15 |
| Fácil | 2 | 10 |
| Muy fácil | 1 | 5 |

Tabla 2 puntos de estimación según el riesgo de desarrollo

Para realizar la asignación de iteraciones se ordenaron las historias de usuario de la siguiente manera:

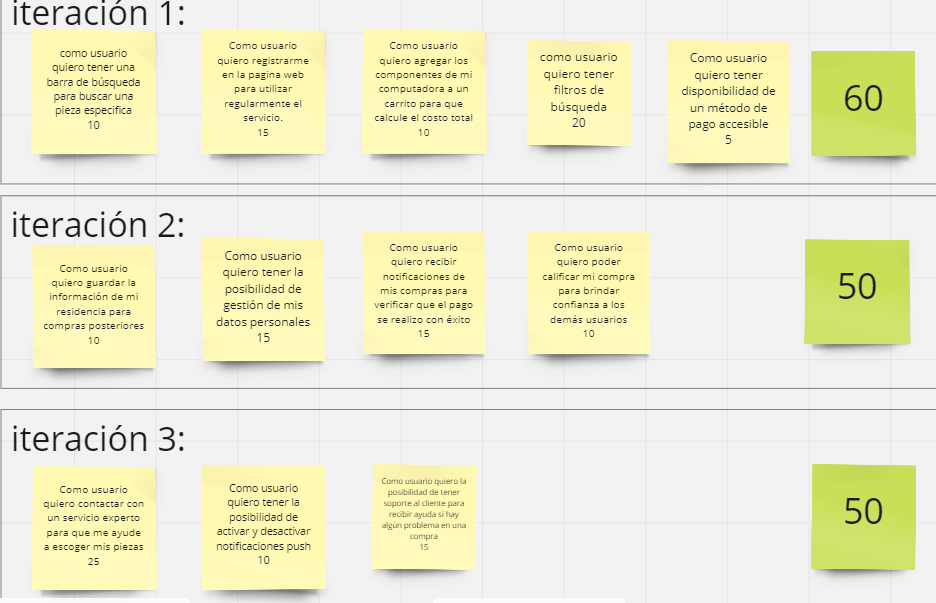


Figura 2 asignación de iteraciones de las historias de usuario.

|  |  |
| --- | --- |
| **Historia de usuario** | |
| **Número: 1** | **Usuario: Cliente** |
| **Nombre de la historia: Registrarse** | |
| **Prioridad en negocio: Es necesario** | **Riesgo en desarrollo: 3** |
| **Puntos/Horas estimados:15** | **Iteración asignada:1** |
| **Programador responsable:** | |
| **Descripción: Como cliente quiero registrarme en la página web para utilizar regularmente el servicio.** | |
| **Criterios de aceptación:**   * **El nombre de usuario debe ser único.** * **La contraseña debe tener más de 10 caracteres.** * **El teléfono de contacto debe estar compuesto solo por números.** * **El correo electrónico debe tener un dominio valido.** * **Los campos de correo electrónico, nombre de usuario, nombre, apellido y contraseña deben ser llenados obligatoriamente.** * **La información ingresada en contraseña y confirmar contraseña deben coincidir.** | |
| **Validación: El cliente puede registrarse en la página web utilizando su correo electrónico. Ingresando su nombre y apellido, nombre de usuario, un teléfono de contacto y una contraseña.** | |

Tabla 3 Historia de usuario del cliente

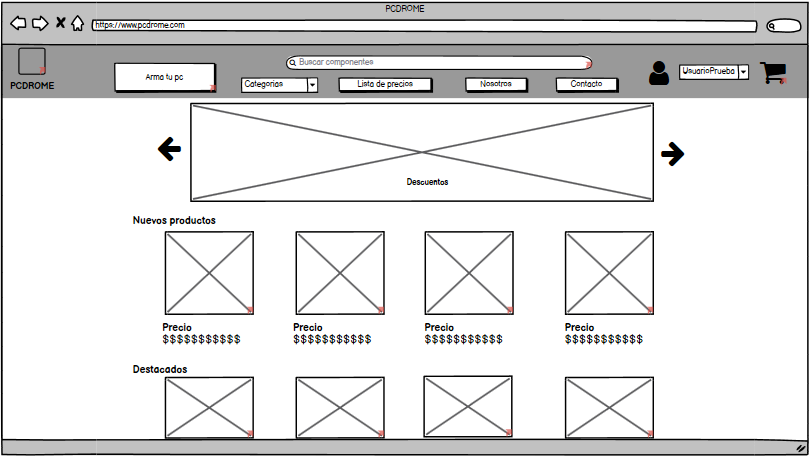


Figura 3 pantalla de inicio de PCDROME

# CARACTERÍSTICAS DEL USUARIO

Cliente

|  |  |
| --- | --- |
| Características del usuario | Un cliente es aquel que está interesado en comprar una o varias piezas en nuestra tienda. |
| Nivel de seguridad o de privilegios | Tiene acceso a crear una cuenta, a iniciar sesión cada vez que lo desee, a gestionar los datos de su cuenta, a añadir productos al carrito, a gestionar sus métodos de pago y a comprar los productos que seleccione. |
| Roles | Visitante  Comprador |
| Nivel de estudios o experiencia técnica | No se requiere ningún nivel de estudio. |
| Frecuencia de uso | Cada vez que desee comprar una pieza para su computador. |

Tabla 4 Descripción de las características del usuario cliente

Administrador

|  |  |
| --- | --- |
| Características del usuario | Un administrador es aquel que actualiza la página cada vez que se requiera agregar un producto nuevo, eliminar algún producto, agregar una promoción o modificar los filtros. |
| Nivel de seguridad o de privilegios | Tiene acceso a una cuenta especial en la que puede realizar los ajustes necesarios. |
| Roles | Administrador: Puede agregar un producto nuevo, eliminar algún producto, agregar una promoción o modificar los filtros.  Moderador: Puede eliminar comentarios inapropiados cuando se califique una compra. |
| Nivel de estudios o experiencia técnica | Se requiere conocimiento en cómo funciona la página para los administradores. |
| Frecuencia de uso | Semanalmente o cada vez que se requiera gestionar la página web. |

Tabla 5 Descripción de las características del usuario administrador

# REQUISITOS NO FUNCIONALES

* El sistema debe estar en español.
* El sistema debe adaptarse al dispositivo, sea móvil, computador o Tablet.
* El sistema debe tener control de acceso para los usuarios. Para poder comprar, un usuario debe tener una cuenta, el sistema debe ser capaz de controlar el acceso a la cuenta mediante la solicitud de contraseña para el inicio de sesión. Si un usuario ha olvidado su contraseña, el sistema hace uso de la verificación por correo.
* Para el inicio de sesión o creación de cuenta, el sistema debe esconder la contraseña, y si el usuario lo desea puede mostrar los caracteres de la contraseña.
* El sistema debe mostrar recomendaciones cuando se está escribiendo en la barra de búsqueda.
* El sistema no debe demorar más de 10 segundos para mostrar los resultados de búsqueda.
* El sistema debe permitir distintos medios de pago, tarjeta de débito, tarjeta de crédito, etc.

# PLAN DE TRABAJO DEL PROYECTO

**Propósito**

Que los integrantes del proyecto sepan qué es lo que deben hacer durante la ejecución del proyecto.

Recuerde que un plan incluye: quién, cómo, cuándo y con qué herramientas se ejecutará el plan, junto con las actividades del plan.

* Muestre el User Story Map, si la imagen no se muestra con calidad, incluirla en un anexo.
* Utilice una herramienta para el control de su proyecto (e.g. Trello, Airtable, Miro, etc). Debe tener todas las historias de usuario, las iteraciones donde se ejecutará dicha historia y el estado actual de la historia, quien o quienes son responsables de la historia y la fecha de vencimiento para tener la historia terminada.
* Puede mostrar una imagen con algún ejemplo en el documento e incluir el enlace a la herramienta para su profesor.

# MÉTODOS Y HERRAMIENTAS DE ESTIMACIÓN

**Propósito**

Fundamentar adecuadamente todas las estimaciones realizadas en el presente documento

**Contenido**

* Para la estimación realizada en el presente documento, explicar los métodos y herramientas utilizadas para elaborar dicha estimación.
* Puede incluir el proceso de estimación en un anexo.
* Explicar en el documento los resultados de la estimación.

# CONCLUSIONES

# ANEXOS

* <https://drive.google.com/file/d/1-r2IFXnQUW1VQxB3-QE8654JGdinsJDk/view?usp=sharing>
* <https://drive.google.com/file/d/14wixY30gMQxj9eALAjLzLtYyeR4_G-6G/view?usp=sharing>

# REFERENCIAS

Ionos. (11 de Marzo de 2019). *Ionos*. Obtenido de https://www.ionos.es/digitalguide/paginas-web/desarrollo-web/el-modelo-en-cascada/#:~:text=El%20modelo%20en%20cascada%20de,%2C%20implementaci%C3%B3n%2C%20verificaci%C3%B3n%20y%20mantenimiento.

Menzinsky, A., López, G., & Palacio, J. (2018). Historias de usuario.

Scrummanager.

https://scrummanager.net/files/historias\_usuario\_scrum\_manager.pdf