# 

# Universidad de Valparaíso INC 404

# Facultad de Ingeniería Metodología de Diseño

# Ingeniería Civil Informática Proyecto

# 

Grupo:

* Héctor Cancino
* Jorge Quiroz
* Garance Récalde

Fecha: 30 de Abril de 2019

Profesor: Eduardo Godoy

Índice

[Instrucciones para comenzar 1](#_gjdgxs)

[1. Introducción 2](#_30j0zll)

[2. Problemática 2](#_1fob9te)

[2.1. Definición del problema 2](#_3znysh7)

[2.2. Solución Propuesta 3](#_2et92p0)

[2.3. Propósitos 3](#_tyjcwt)

[2.4. Objetivos](#_3dy6vkm) (Principal y Específicos) 3

[3. Definición de Requerimientos 3](#_1t3h5sf)

[4. Plan de trabajo 3](#_4d34og8)

[5. Componentes técnicos de la solución 4](#_2s8eyo1)

[6. Conclusiones 4](#_17dp8vu)

# Introducción

El comercio electrónico consiste en la venta y compra de productos mediante medios electrónicos como las redes sociales o páginas web, usando formas electrónicas de pago, como las tarjetas de crédito o débito.

El e-commerce ha ido en rápido aumento y su masificación ha ido de la mano de la masificación del acceso a internet y del acceso a medios de pago electrónico, junto a ello el desarrollo de tecnologías web seguras crean un marco de desarrollo que permite la creación de completas plataformas que hoy en día compiten en una proporción no menor con las ventas presenciales.

El proyecto se inserta en el marco del comercio electrónico, en donde existen diversos componentes -claramente definidos y modularizados- entre los que podemos encontrar un catálogo de productos, con portada, funciones de búsqueda avanzada y por categoría y una plataforma de pagos segura. Adicionalmente, en este informe se explora la mejora en la experiencia de usuario al implementar un *carrito de compras*, que funciona a modo de *buffer* donde el usuario selecciona artículos para posteriormente obtener un total que es procesado en la plataforma de pagos.

# Problemática

# Definición del problema

En un entorno de comercio online (e-commerce) los usuarios con frecuencia adquieren más de un producto de una tienda, lo que puede llegar a ser un proceso engorroso debido a todos los pasos a seguir para poder procesar una transacción, los que se tendrían que repetir para cada producto que se desea comprar. Esto es precisamente lo que ocurre en la tienda de teléfonos celulares ‘Cell-Shop’ que posee un sistema básico de ventas.

Por otro lado, muchas veces un cliente encuentra un producto pero no lo compra, muchas veces con el objetivo de concretar la compra posteriormente, momento en el que debe volver a buscar el producto, lo que puede resultar engorroso si el catálogo es muy extenso o, en este caso, no permite una búsqueda refinada.

# Solución Propuesta

Se propone el desarrollo de un sistema tipo ‘carrito de compras’ al cual el usuario puede agregar productos para posteriormente comprarlos todos, realizando de este modo solamente una transacción por el monto total.

El carrito presentará opciones para agregar o eliminar productos además de modificar las cantidades de los mismos.

Adicionalmente, este sistema permitirá al usuario guardar el carrito con sus productos y cantidades para finalizar la compra en un momento posterior, dicho almacenamiento se realiza en la caché del navegador, sin implementar cuentas de usuario, esto hace que el proceso sea rápido y el carrito siempre se encuentre disponible, ahorrando al usuario volver a tener que buscar el producto además de presentar el beneficio que se realiza solamente una transacción.

# Propósitos

Se construirá un sistema de carrito de compras en modalidad de aplicación web, utilizando herramientas y tecnologías open-source incluyendo PHP, el *framework* CodeIgniter, MySQL como gestor de base de datos y *bootstrap*, el que se incorporará al catálogo ya existente de la tienda.

# Objetivos (Principal y Específicos)

El objetivo principal del proyecto es desarrollar un sistema de carrito de compras que funcione como aplicación web, para lo cual una serie de procesos específicos se deben llevar a cabo: Asignación de roles y tiempos entre los integrantes del equipo de trabajo, instalación, configuración y enlace de todas las tecnologías web que forman parte del proyecto, entender, esquematizar y definir los requerimientos utilizando recursos como las historias de usuario y diagramas de casos de uso. Todo esto con la finalidad de elaborar un plan de trabajo, el cual permitirá el desarrollo de la aplicación web de manera eficiente y ordenada, además de generar documentación que permita el adecuado uso y mantención del sistema por parte de sus actores.

# Definición de Requerimientos

Dado que el proyecto está inserto en el marco del desarrollo solamente de la funcionalidad del carrito de compras, no se contempla el desarrollo del catálogo de productos, ni de la plataforma de pagos. El desarrollo de este carrito asume que ya existe un catálogo y plataforma de pagos.

Solamente se desarrollará un catálogo acotado y sencillo para poder agregar y verificar la funcionalidad de agregar productos.

### Requerimientos funcionales

RQSW001: Yo como usuario de la aplicación web quiero poder agregar artículos que deseo comprar a mi carrito.

RQSW002: Yo como usuario quiero ver los artículos que seleccioné en mi carrito.

RQSW003: Yo como usuario quiero poder modificar la cantidad de artículos que son en mi carrito.

RQSW004: Yo como usuario quiero poder ver el precio total de los productos que son en mi carrito, que se calcule de manera automática.

RQSW005: Yo como usuario quiero seleccionar artículos que pretendo borrar.

RQSW006: Yo como usuario quiero borrar artículos que no necesito.

RQSW007: Yo como usuario quiero poder realizar compra solo con mi rut, es decir sin tener que crear una cuenta.

RQSW008: Yo como usuario quiero poder ver mi carrito posteriormente, es decir que mis posibles compras sean almacenadas en caché, en caso de alguna situación inesperada.

RQSW009: Yo como usuario quiero poder validar mi carrito, una vez que estoy satisfecho con mi elección.

**Requerimientos no funcionales**

RQSW010: Yo como usuario, quiero que la aplicación tenga una interfaz simple y fácil de usar.

RQSW011: Yo como usuario, quiero que la aplicación sea disponible 24 / 7.

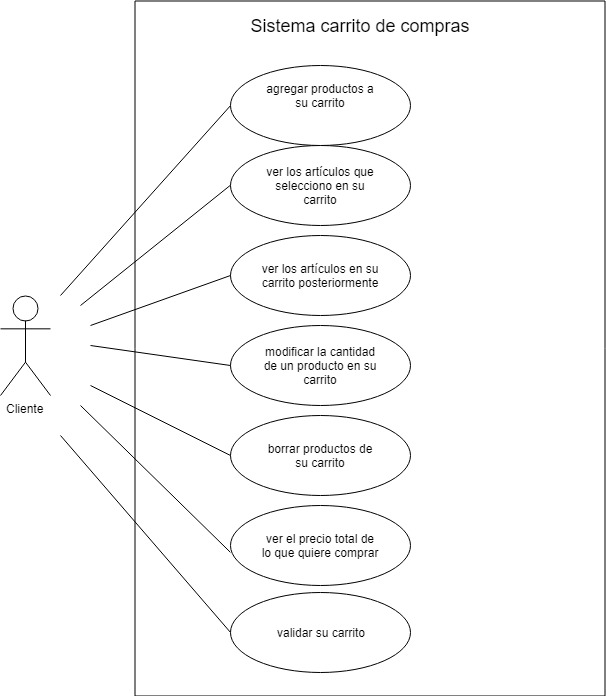
RQSW012: Yo como usuario, quiero que el tiempo de respuesta de la aplicación sea menos de 2 segundos.

# Análisis de Requerimientos

### Actores del sistema

* **Cliente** : Usuario final del sistema que utiliza el carro para realizar compras on-line.
* **Administrador** : En un sistema más completo habría un administrador que sería parte del catálogo y quizás en la plataforma de pagos, pero el carrito en sí debería ser manejado en un 100% por el usuario, el administrador no debería poder agregar o quitar artículos del carrito de un cliente.

### Diagrama de Casos de Uso



# Plan de trabajo

* 1. *Equipo de Trabajo*

|  |  |  |
| --- | --- | --- |
| Integrante | Horas semanales (aproximadas) | Tareas |
| Héctor Cancino | 3 - 5 | Diseño, arquitectura |
| Jorge Quiroz | 3 - 5 | Programación |
| Garance Récalde | 3 - 5 | Documentación, revisión, pruebas de funcionalidad |

* 1. *Programación*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Semana** | **Horas** | **Hito** | **Caso de uso** | **Detalle** |
| **29 Abril** | 3 | 1 | N/A | Planificación y redacción de informe |
| **6 Mayo** | 6 | 1 | N/A | Diseño de aplicación, arquitectura, aprendizaje de herramientas. |
| **13 Mayo** | 6 | 1 | N/A | Redacción de informe, Entrega  Arquitectura |
| **20 Mayo** | 6 | 2 | Ver artículos | Configuración, instalación y aprendizaje |
| **27 Mayo** | 6 | 2 | Obtener precio total | Programación y diseño |
| **3 Junio** | 6 | 2 | Validar carrito | Programación y diseño |
| **10 Junio** | 3 | 2 | Validar carrito | Entrega 2, redacción de informe, avance del código funcional. |
| **17 Junio** | 6 | 3 | Modificar carrito  Eliminar producto | Programación |
| **24 Junio** | 6 | 3 | Guardar carrito | Programación |
| **1 Julio** | 3 | 3 | Guardar carrito | Entrega final  Código y redacción de informe. |

# Componentes técnicos de la solución

*CodeIgniter*, se basa en el patrón de diseño MVC (Model-View-Controller), el cual es implementado de un modo muy organizado y visible, se refiere principalmente al manejo de la interfaz de usuario. El controlador invoca a la vista, modularizando el desarrollo. MVC logra separar la parte visual de la parte lógica.

Como ventajas encontramos la posibilidad de que múltiples desarrolladores puedan trabajar en el código simultáneamente, que un modelo puede tener múltiples vistas y además el ordenamiento lógico que se produce en el controlador y las vistas. En contra podemos comentar que la curva de aprendizaje puede ser más pronunciada por la división funcional del código, que introduce más capas de abstracción, que hacen que el código se vuelva más complejo, además de la necesidad del equipo de desarrollo de manejar bien todas las tecnologías web que requiere CodeIgniter, incluyendo PHP, CSS y bases de datos.

# Conclusiones

A modo de conclusión, para abordar este proyecto se requiere un mayor manejo en todas las herramientas a utilizar, en caso de encontrar nuevas mejoras. Finalmente con este proyecto se pretende desarrollar un buen prototipo de carro de compras óptimo para los usuarios.