UNIVERSIDAD METROPOLITANA CASTRO CARAZO

SEDE PUNTARENAS

BACHILLERATO EN INGENIERIA INFORMATICA

ANÁLISIS DEL PROCESO DE ADMINISTRACIÓN Y CONTROL DEL PERSONAL QUE UTILIZA LA UNIDAD DEPORTIVA DEL GIMNASIO DEL COMITÉ DE DEPORTE Y RECREACION, OROTINA, ALAJUELA

TESIS PRESENTADA PARA OPTAR POR EL GRADO DE BACHILLERATO EN INGENIERIA INFORMATICA

KEVIN CASTILLO MORA

CARNÉ: 604280878

PUNTARENAS, COSTA RICA

18 DE JUNIO DE 2013

**TRIBUNAL EXAMINADOR**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

Lic. Roberto Gustavo Escobar Agüero

Presidente del tribunal

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

Lic. Xxx xxxx xxxx

Asesor Académico

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

Melbert Dalorzo

Asesor Académico

**DECLARACIÓN JURADA**

Yo, Kevin Castillo Mora, estudiante de la Carrera de Ingeniería Informática, portador de la cédula de identidad número 6 0428 0878, en este acto debidamente percibido y entendido de las penas y consecuencias con que se castiga en el Código Penal de nuestro país el delito de perjurio ante quienes constituyen el Tribunal Examinador de mi Proyecto de Graduación, juro solemnemente que este trabajo de investigación es una obra original y que he respetado todo lo pre-aceptado por las leyes penales, así como los derechos de autor.

No omito en señalar que quedo advertido que la Universidad Metropolitana Castro Carazo sede Puntarenas se reserva el derecho de protocolizar este documento ante un notario público.

En fe de lo anterior, firmo en la ciudad de Puntarenas, a los \_\_\_\_\_\_ días del mes de \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ del dos mil trece.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Xxx xxxx xxxx

xxx xxxx xxxx

**DEDICATORIA**

Mi familia ha sido el pilar más importante en esta etapa de mi vida y a ellos exclusivamente se los dedico y les doy las gracias infinitas por siempre apoyarme, confiar en mí por darme el respaldo y todas las comodidades siempre y sobre todo por darme todo su amor. Sin importar en donde estén o si alguna vez llegan a leer estas dedicatorias, les doy las gracias por formar parte de mí vida, por todo lo que me han brindado y por todas sus bendiciones.

A ti DIOS por principalmente darme la salud y sabiduría necesaria para permitirme cumplir este sueño, sé que siempre me acompañas y estarás junto a mí.

Tengo a el mejor padre y a la mejor madre del mundo, gracias a los 2 por todos sus esfuerzos bendiciones y amor que me han dado siempre los tendré en mi corazón y presente que sin ustedes dos no sería lo que hoy soy, los amo demasiado.

**AGRADECIMIENTOS**

En primer lugar a DIOS, Quien me guía y fortalece en cada momento de mi vida y con sus bendiciones ha estado delante de mí a cada momento.

A mis padres y hermanos, gracias a su apoyo, consejos, confianza y amor que me han servido de apoyo para finalizar este trabajo.

A mis profesores, no solo de la carrera sino de toda la vida. Mil gracias porque han sido parte de mi formación como profesional.

**TABLA DE CONTENIDOS**

CAPÍTULO I

INTRODUCTORIO

**CAPÍTULO I**

1. **INTRODUCTORIO**
   1. **ASPECTOS SITUACIONALES DE LA INSTITUCIÓN**

* **Historia del Sector**

En la época precolombina el territorio que actualmente corresponde al cantón de Orotina estuvo habitado por indígenas de dos etnias principales: los [huetares](https://es.wikipedia.org/wiki/Huetares" \o "Huetares) y los [orotiñas](https://es.wikipedia.org/wiki/Oroti%C3%B1a" \o "Orotiña). Los huetares, cuyo rey a la llegada de los españoles era [Garabito](https://es.wikipedia.org/wiki/Garabito_(rey)" \o "Garabito (rey)), se asentaron en el llamado Valle de Coyoche, llamado más tarde Valle de la Cruz por los españoles, que en la actualidad corresponde a los llanos de [San Mateo](https://es.wikipedia.org/wiki/Cant%C3%B3n_de_San_Mateo" \o "Cantón de San Mateo), Orotina y [Esparza](https://es.wikipedia.org/wiki/Esparza_(Costa_Rica)" \o "Esparza (Costa Rica)). Los orotiñas, gobernados por el rey [Gurutina](https://es.wikipedia.org/wiki/Gurutina" \o "Gurutina) de etnia [chorotega](https://es.wikipedia.org/wiki/Chorotega" \o "Chorotega), habitaron en la región de [Orotiña](https://es.wikipedia.org/wiki/Oroti%C3%B1a" \o "Orotiña), que comprendía, además de parte de Orotina, parte del cantón central de [Puntarenas](https://es.wikipedia.org/wiki/Cant%C3%B3n_de_Puntarenas" \o "Cantón de Puntarenas).

* **Ubicación geográfica**

Alajuela, Orotina frente a tribunales de justicia.

* **Misión de la institución**

Promover el deporte para así disminuir la delincuencia y la pobreza en el cantón.

* **Visión de la institución**

Tener deportistas de alto rendimiento que representen tanto al cantón como al país.

* **Valores institucionales**

Solidaridad, ayudar, promover.

* **Estructura organizacional**
* **Consejo Directivo**
  1. **ANTECEDENTES DEL PROBLEMA**

El principal problema era la falta de organización y tener el control en todo momento era muy difícil tener que buscar y separar cada día a los clientes que ya les tocaba pagar y que estaban atrasados.

* 1. **JUSTIFICACIÓN DEL PROBLEMA**

Con la implementación del software para solucionar dicho problema es mucho más fácil poder saber en cualquier momento datos de un cliente especifico al igual que su estado.

* 1. **DELIMITACIÓN DEL PROBLEMA**
* **El presente trabajo:**

Va a satisfacer todas las necesidades básicas en la administración de un gimnasio desde poder llevar el control de todos los clientes los cuales estén al día o atrasados también, así como llevar un registro de todos los pagos realizados por cada uno de ellos.

* **El objetivo general de la investigación:**

Poder darle al gimnasio del comité de deportes de Orotina una mejor y más eficiente administración tanto para el comité como para los clientes y su administración.

* + 1. **Formulación del problema como pregunta**

¿Qué tan importante es tener un buen control en la administración de las mensualidades de un gimnasio?

Esta fue la pregunta mas importante que pude realizarme a la hora de empezar este proyecto, llegando a la conclusión reuniéndome e investigando que, si es sumamente importante tener un buen control de este tema, la razón principal es para llevar un control de las finanzas, que en una empresa de cualquier tipo es en donde hay que tener mucho cuidado para saber cómo están marchando las cosas y si es o no factible un producto.

* 1. **OBJETIVOS**

El mejoramiento de la administración del gimnasio tanto financiera como de control de clientes.

* + 1. **Objetivo general**
* **Objetivo General Propuesta:**
* **Objetivos específicos**
  1. **DEFINICIÓN DE LAS VARIABLES**
* **Definición conceptual:**
* **Definición operativa:**
  1. **DEFINICIÓN DEL MAPA CONCEPTUAL**
     1. **Mapa Conceptual**

CAPÍTULO II

MARCO TEÓRICO

**CAPíTULO ii**

1. **MARCO TEÓRICO**

En el siguiente capítulo se desarrollaron conceptos básicos relevantes en el análisis, diseño e implementación de un Sistema informático para el control de las mensualidades de un gimnasio

* 1. **INFORMÁTICA**

La Informática es la ciencia aplicada que abarca el estudio y aplicación del tratamiento automático de la información, utilizando sistemas computacionales, generalmente implementados como dispositivos electrónicos. También está definida como el procesamiento automático de la información.

**http://es.wikipedia.org/wiki/Inform%C3%A1tica**

1. **HISTORIA**

La informática no es un invento en sí misma, sino, un largo proceso a través de la Historia, cuyos inicios se pueden remontar al año 3500 a. C. con el invento de ábaco en Babilonia, considerándolo como la primera máquina con la que se podían hacer cálculos.

**http://es.ticnologia.wikia.com/wiki/BREVE\_HISTORIA\_DE\_LA\_INFORM%C3%81TICA**

1. **INTERACCIÓN PERSONA-COMPUTADOR**

En términos generales, es la disciplina que estudia el [intercambio de información](https://es.wikipedia.org/wiki/Comunicaci%C3%B3n) mediante [software](https://es.wikipedia.org/wiki/Interfaz_de_usuario) entre las [personas](https://es.wikipedia.org/wiki/Usuario_(inform%C3%A1tica)) y las [computadoras](https://es.wikipedia.org/wiki/Computadora). Esta disciplina se encarga del diseño, evaluación e [implementación](https://es.wikipedia.org/wiki/Implementaci%C3%B3n) de los aparatos tecnológicos interactivos, estudiando el mayor número de casos que les pueda llegar a afectar. El objetivo es que el intercambio sea más eficiente: minimizar errores, incrementar la satisfacción, disminuir la frustración y, en definitiva, hacer más productivas las tareas que rodean a las personas y los computadores.

**https://es.wikipedia.org/wiki/Interacci%C3%B3n\_persona-computadora**

1. **RAÍCES HISTÓRICAS**

El inicio y el primer dispositivo por el cual podemos empezar a llamarlo parte de la historia informática es el el Ábaco, que servía para representar números en el sistema decimal y contar, permitiendo la realización de operaciones aritméticas sencillas. Después de ahí surgieron muchos más inventos primitivos y manuales hasta que en los años 50’s nace la primera generación de computadoras digitales, llamado ENAC, era una enorme máquina que ocupaba todo el sótano de la universidad, pesaba varias toneladas y conseguía hacer 5000 sumas por segundo. La siguieron EDVAC (1949), primer ordenador programable y UNIVACI (1951), primer ordenador que se comercializó. A partir de ahí la historia tuvo 3un cambio y fue pasando de generación en generación hasta llegar a la quinta generación situada en los años 80’s donde se creó el primer computador personal durante este periodo se perfecciona la tecnología para conseguir realizar varias tareas simultáneamente, se introducen varios procesadores en la Unidad Central de Proceso.

**http://es.ticnologia.wikia.com/wiki/BREVE\_HISTORIA\_DE\_LA\_INFORM%C3%81TICA**

1. **PRINCIPALES COMPONENTES**

**El procesador**

También llamado Unidad Central de Proceso (UCP) (en inglés, Central Process Unit, CPU), es el lugar donde se interpretan y ejecutan las instrucciones de los programas.

**La memoria**

Es uno de los principales componentes de la computadora. Su función consiste en almacenar las instrucciones y los datos durante la ejecución de los programas. Según su comportamiento, se pueden distinguir dos tipos de memorias: la memoria RAM y la memoria ROM.

**La memoria RAM (Random Access Memory)**

Es el lugar donde se almacenan las instrucciones de los programas que se están ejecutando y los datos que éstos manejan. La capacidad de la memoria RAM de la computadora se indica en megabytes.

**La memoria ROM (Read Only Memory)**

Es una memoria permanente, es decir, no se borra al apagar la computadora. En ella están almacenadas las instrucciones básicas para el funcionamiento de la máquina y para las operaciones de entrada/salida (en inglés, Basic Input Output System, BIOS). Entre estas instrucciones se encuentra la rutina de arranque,  
que se encarga de indicar al hardware los pasos para comprobar el estado de los componentes de la computadora y cargar el sistema operativo para empezar a trabajar.

**Tarjeta madre.**

La tarjeta madre es la tableta en donde se montan todos los componentes y sus interconexiones a manera de pistas (cobre adherido a la tarjeta). Básicamente es el dispositivo que aglutina a todos los demás, tales como [microprocesador](http://mx.geocities.com/pcmuseo/mecatronica/procesador.htm), [buses](http://mx.geocities.com/pcmuseo/mecatronica/buses.htm), [ranuras](http://mx.geocities.com/pcmuseo/mecatronica/ranuras.htm), [zócalos](http://mx.geocities.com/pcmuseo/mecatronica/ranuras.htm" \l "Zocalos), [memoria](http://mx.geocities.com/pcmuseo/mecatronica/memoria.htm), [puertos](http://mx.geocities.com/pcmuseo/mecatronica/puertos.htm), [conectores](http://mx.geocities.com/pcmuseo/mecatronica/conectores.htm), [video](http://mx.geocities.com/pcmuseo/mecatronica/video.htm), etc.

[**http://biuinformatica1.blogspot.com/2009/08/principales-componentes-de-**](http://biuinformatica1.blogspot.com/2009/08/principales-componentes-de-)**la.html**

1. **USUARIO**

Un usuario es aquel que usa algo o que usa ordinariamente algo. ​ Por ejemplo un usuario de una [biblioteca](https://es.wikipedia.org/wiki/Biblioteca) es un lector interesado en leer algún volumen de su archivo. Sin embargo, en informática existe varios tipos de usuario.

**Usuario registrado**

Refiere aquellas personas que para utilizar un servicio online tienen un identificador y una clave que le permite el acceso a este. Ejemplo, los usuarios de redes sociales quienes tienen que llenar un registro para obtener una cuenta de ingreso para poder acceder a la red.

**Usuario anónimo**

Es el usuario que utiliza a diario un servicio producto sin que nadie lo reconozca, un ejemplo de ello son los usuarios online en el campo de la informática, quienes logran navegar por el Internet sin registrarse, lo cual le hace imposible a la empresa quien desarrolla ciertas páginas web, reconocer de quien se trata el usuario.

**Usuario tester**

Describe a los tipos de usuarios que tienen como fin validar la funcionalidad de un web o programa, teniendo como objetivo principal velar por el correcto funcionamiento del mismo.

**Usuario administrador**

Se utiliza este nombre para designar a la persona que maneja un producto informativo, sea un software o un hardware, quien tiene la total autorización de realizar cualquier ajuste o cambio a lo que está desarrollando.

**Usuario beta tester**

Refiere al tipo de usuario que se mantiene entre el usuario administrador y el tester, quien tiene la función principal de dar a conocer posibles errores que presente el desarrollador o programador, así mismo es quien busca los medios de que el programa creado logre llegar al usuario final sin presentar ningún error.

**Usuarios inexpertos**

Son las personas que utilizan tecnologías y maquinarias, pero no tienen mucha experiencia en cómo manejarse con estas.

**Usuarios habilitados**

Son usuarios que pueden ser buenos en un campo pero carecen de alguna habilidad que le evitan ser experto.

**Usuario principiante**

Refiere a las personas que antes han tenido el contacto con un tecnología o maquinaria, pero no conocen como desenvolverse en ella, aprendiendo poco a poco a cómo hacerlo.

**Usuarios expertos**

Con esta denominación se identifica aquellas personas que tienen un alto conocimiento en el manejo y funcionamiento de la tecnología y de ciertas maquinarias, por lo cual pueden manejar la misma a su antojo.

**Usuario de sistema**

Son las personas que se registran en un determinado sistema con el fin de utilizar un servicio que este ofrece. Estos usuarios llegan a presentar diversos niveles o perfiles, donde acorde al nivel que presente podrán tener determinados privilegios que ofrece el sistema.

[**http://www.mastiposde.com/usuarios.html**](http://www.mastiposde.com/usuarios.html)

1. **COMPUTADOR**

Es una [máquina electrónica](https://es.wikipedia.org/wiki/Aparato_electr%C3%B3nico) que recibe y procesa [datos](https://es.wikipedia.org/wiki/Dato), para convertirlos en información conveniente y útil, que posteriormente se envía a las unidades de salida, para que pueda ser analizada y utilizada por humanos. Un [ordenador](https://es.wikipedia.org/wiki/Ordenador_digital) está formado físicamente por numerosos [circuitos integrados](https://es.wikipedia.org/wiki/Circuito_integrado) y muchos componentes de apoyo, extensión y accesorios, que en conjunto pueden ejecutar tareas diversas con suma rapidez y bajo el control de un [programa](https://es.wikipedia.org/wiki/Programa_inform%C3%A1tico) (software).

Dos partes esenciales la constituyen, el [hardware](https://es.wikipedia.org/wiki/Hardware" \o "Hardware) (hard = duro) que es su estructura física (circuitos electrónicos, cables, gabinete, teclado, etc), y el [software](https://es.wikipedia.org/wiki/Software" \o "Software) que es su parte intangible (programas, datos, información, señales digitales para uso interno, etc).

**https://es.wikipedia.org/wiki/Computadora**

1. **ORIGEN DEL PROCESO INTERACTIVO**

La historia digital es un campo que cambia rápidamente. Nuevos métodos y formatos se están desarrollando actualmente. Esto significa que la "historia digital" es un término difícil de definir. Sin embargo, es posible identificar las características generales. La Historia digital representa una democratización de la historia en la que cualquier persona con acceso a Internet puede hacer oír su voz, incluidos los grupos marginados que fueron excluidos a menudo en el "grandes narrativas" de la nación y el imperio. A diferencia de los formatos de los medios de comunicación anteriores, los textos de historia digital tienden a ser no lineales e interactivos, fomentando la participación y compromiso del usuario. La Historia Digital se estudia desde varias perspectivas disciplinarias y en relación con una serie de actividades y temas relacionados entre sí. El campo incluye la discusión de: archivos, bibliotecas, imágenes y enciclopedias; museos y exposiciones virtuales; Lectura identidad digital y la biografía; juegos digitales y los mundos virtuales; ​ las comunidades online y redes sociales; ​ Web 2.0; Boggs y e-investigación y la infraestructura cibernética

**https://es.wikipedia.org/wiki/Historia\_digital**

1. **PRINCIPIOS DE DISEÑO**

El diseño y la construcción del software está soportada por varios principios fundamentales. Estos principios favorecen que los objetivos de calidad del software se alcancen con mayor facilidad. En este tema se estudiarán los principios y técnicas que permiten construir arquitecturas software correctas. Primeramente, se introducirá la fase de diseño y el proceso de diseño, para posteriormente centrarse en los principios y conceptos fundamentales del diseño del software, haciendo un especial hincapié en todos aquéllos que permitan alcanzar un diseño modular eficaz, basado en módulos altamente cohesionados, con bajo acoplamiento y construidos sobre la base de la ocultación de la información.

**http://ocw.usal.es/ensenanzas-tecnicas/ingenieria-del-software/contenidos/Tema5-Principiosdeldisenodelsoftware-1pp.pdf7**

1. **METODOLOGÍAS DE DISEÑO**

Cada metodología de desarrollo de software tiene más o menos su propio enfoque para el [desarrollo de software](https://es.wikipedia.org/wiki/Desarrollo_de_software). Estos son los enfoques más generales, que se desarrollan en varias metodologías específicas. Estos enfoques son los siguientes:

[**Modelo en cascada**](https://es.wikipedia.org/wiki/Modelo_en_cascada)**:** Framework lineal.

[**Prototipado**](https://es.wikipedia.org/wiki/Desarrollo_con_prototipaci%C3%B3n): Framework iterativo.

[**Incremental**](https://es.wikipedia.org/wiki/Desarrollo_iterativo_y_creciente): Combinación de framework lineal e iterativo.

[**Espiral**](https://es.wikipedia.org/wiki/Desarrollo_en_espiral): Combinación de framework lineal e iterativo.

**RAD**: Rapid Application Development, framework iterativo.

**https://es.wikipedia.org/wiki/Metodolog%C3%ADa\_de\_desarrollo\_de\_software#Modelo\_en\_cascada**

1. **¿QUÉ ES UN PROYECTO?**

Es un sistema de cursos de acción simultánea y/o secuencial que incluye personas, equipamientos de hardware, software y comunicaciones, enfocados en obtener uno o más resultados deseables sobre un sistema de información. El inicio de un proyecto informático generalmente está dado en la solicitud de requerimientos de los usuarios, y siendo que los diferentes sistemas de Información abordan los diferentes tipos de problemas organizacionales; podemos clasificar a los Sistemas de Información según sean las aplicaciones que necesite cada usuario en: Sistemas de Transacciones, Sistemas de Soporte para la toma de decisiones, y Sistemas Expertos.

[**http://www.monografias.com/trabajos-pdf2/proyectoinformatico/proyecto-informatico.pdf**](http://www.monografias.com/trabajos-pdf2/proyecto-informatico/proyecto-informatico.pdf)

1. **¿QUÉ ES TECNOLOGÍA?**

La tecnología es el conjunto de conocimientos con las que el hombre desarrolla un mejor entorno, más saludable, agradable y sobre todo cómodo para la optimización de la vida. La tecnología combina la técnica de mejoramiento de un [espacio](http://conceptodefinicion.de/espacio/) con las distintas revoluciones que se han suscitado en los últimos siglos, específicamente hablando de la [revolución industrial](http://www.historialuniversal.com/2010/09/revolucion-industrial.html), en ella, se marcó un antes y un después en la tecnología, el [trabajo](http://conceptodefinicion.de/trabajo/) a mano paso a ser un trabajo en serie producido por una máquina a base de vapor con la que se movía algún tipo de herramienta y correa de [transporte](http://conceptodefinicion.de/transporte/) en él, desarrollando así una materia prima en mayor cantidad, mejor presupuesto y de excelente calidad.

**http://conceptodefinicion.de/tecnologia/**

1. **¿QUE ES INGENIERIA?**

La ingeniería es el conjunto de conocimientos científicos y tecnológicos para la innovación, invención, desarrollo y mejora de técnicas y herramientas para satisfacer las necesidades de las empresas y la sociedad.

El ingeniero utiliza las [matemáticas](https://es.wikipedia.org/wiki/Matem%C3%A1ticas), la [física](https://es.wikipedia.org/wiki/F%C3%ADsica), la [química](https://es.wikipedia.org/wiki/Qu%C3%ADmica), la [programación](https://es.wikipedia.org/wiki/Programaci%C3%B3n) y otras ciencias tanto para el desarrollo de tecnologías, como para el manejo eficiente y productivo de recursos y fuerzas de la naturaleza en beneficio de la sociedad. La ingeniería es una actividad que transforma el conocimiento en algo práctico.

La ingeniería aplica los conocimientos y métodos científicos a la invención o perfeccionamiento de tecnologías de manera pragmática y ágil, adecuándose a las limitaciones de tiempo, recursos, requisitos legales, requisitos de seguridad, ecológicos, etc.

**https://es.wikipedia.org/wiki/Ingenier%C3%ADa**

1. **LA INGENIERÍA INFORMÁTICA**

La ingeniería informática o [ingeniería en computación](https://es.wikipedia.org/wiki/Ingenier%C3%ADa_en_computaci%C3%B3n) es la rama de la [ingeniería](https://es.wikipedia.org/wiki/Ingenier%C3%ADa) que aplica los fundamentos de la [ciencia de la computación](https://es.wikipedia.org/wiki/Ciencias_de_la_computaci%C3%B3n), la [ingeniería electrónica](https://es.wikipedia.org/wiki/Ingenier%C3%ADa_electr%C3%B3nica) y la [ingeniería de software](https://es.wikipedia.org/wiki/Ingenier%C3%ADa_de_software), para el desarrollo de soluciones integrales de cómputo y comunicaciones, capaces de procesar [información](https://es.wikipedia.org/wiki/Informaci%C3%B3n) de manera automática.

**https://es.wikipedia.org/wiki/Ingenier%C3%ADa\_inform%C3%A1tica**

1. **¿QUE ES UN SISTEMA DE INFORMACIÓN?**

Un [sistema](http://www.monografias.com/trabajos11/teosis/teosis.shtml) de información es un conjunto de elementos que interactúan entre sí con el fin de apoyar las actividades de [una empresa](http://www.monografias.com/trabajos11/empre/empre.shtml) o negocio.

El equipo computacional: el [hardware](http://www.monografias.com/Computacion/Hardware/) necesario para que el sistema de información pueda operar.

El recurso humano que interactúa con el Sistema de Información, el cual está formado por las personas que utilizan el sistema.

Un sistema de información realiza cuatro actividades básicas: entrada, [almacenamiento](http://www.monografias.com/trabajos12/dispalm/dispalm.shtml), procesamiento y salida de información.

**http://www.monografias.com/trabajos7/sisinf/sisinf.shtml4**

1. **LENGUAJE DE PROGRAMACIÓN**

Un lenguaje de programación es un [lenguaje formal](https://es.wikipedia.org/wiki/Lenguaje_formal) diseñado para realizar [procesos](https://es.wikipedia.org/wiki/Proceso_(inform%C3%A1tica)) que pueden ser llevados a cabo por máquinas como las [computadoras](https://es.wikipedia.org/wiki/Computadora).

Pueden usarse para crear [programas](https://es.wikipedia.org/wiki/Software) que controlen el comportamiento físico y lógico de una máquina, para expresar [algoritmos](https://es.wikipedia.org/wiki/Algoritmo) con precisión, o como modo de comunicación humana.

Está formado por un conjunto de símbolos y reglas [sintácticas](https://es.wikipedia.org/wiki/Sintaxis) y [semánticas](https://es.wikipedia.org/wiki/Sem%C3%A1ntica) que definen su estructura y el significado de sus elementos y expresiones. Al proceso por el cual se escribe, [se prueba](https://es.wikipedia.org/wiki/Beta_tester), [se depura](https://es.wikipedia.org/wiki/Depurador), [se compila](https://es.wikipedia.org/wiki/Compilador) (de ser necesario) y se mantiene el [código fuente](https://es.wikipedia.org/wiki/C%C3%B3digo_fuente) de un [programa informático](https://es.wikipedia.org/wiki/Programa_inform%C3%A1tico) se le llama [programación](https://es.wikipedia.org/wiki/Programaci%C3%B3n).

**https://es.wikipedia.org/wiki/Lenguaje\_de\_programación**

1. **QUE ES UN NAVEGADOR WEB**

Un navegador web (en inglés, web browser) es un software, aplicación o programa que permite el acceso a la Web, interpretando la información de distintos tipos de archivos y sitios web para que estos puedan ser visualizados.

La funcionalidad básica de un navegador web es permitir la visualización de documentos de texto, posiblemente con recursos multimedia incrustados. Además, permite visitar páginas web y hacer actividades en ella, es decir, enlazar un sitio con otro, imprimir, enviar y recibir correo, entre otras funcionalidades más.

[**https://es.wikipedia.org/wiki/Navegador\_web**](https://es.wikipedia.org/wiki/Navegador_web)

1. **QUE ES EL INTERNET**

El nombre Internet procede de las palabras en [inglés](http://conceptodefinicion.de/ingles/) Interconnected Networks, que significa “redes interconectadas”. Internet es la unión de todas las redes y computadoras distribuidas por todo el mundo, por lo que se podría definir como una red global en la que se conjuntan todas las redes que utilizan protocolos TCP/IP y que son compatibles entre sí.

[**http://conceptodefinicion.de/internet/**](http://conceptodefinicion.de/internet/)

1. **APLICACIONES WEB**

En la [ingeniería de software](https://es.wikipedia.org/wiki/Ingenier%C3%ADa_de_software) se denomina aplicación web a aquellas [herramientas](https://es.wikipedia.org/wiki/Aplicaci%C3%B3n_inform%C3%A1tica) que los usuarios pueden utilizar accediendo a un [servidor web](https://es.wikipedia.org/wiki/Servidor_web) a través de [internet](https://es.wikipedia.org/wiki/Internet) o de una [intranet](https://es.wikipedia.org/wiki/Intranet) mediante un [navegador](https://es.wikipedia.org/wiki/Navegador_web). En otras palabras, es un programa que se codifica en un lenguaje interpretable por los navegadores web en la que se confía la ejecución al navegador.

Las aplicaciones web son populares debido a lo práctico del [navegador web](https://es.wikipedia.org/wiki/Navegador_web) como [cliente ligero](https://es.wikipedia.org/wiki/Cliente_ligero), a la independencia del sistema operativo, así como a la facilidad para actualizar y mantener aplicaciones web sin distribuir e instalar software a miles de usuarios potenciales.

https://es.wikipedia.org/wiki/Aplicaci%C3%B3n\_web

1. **Que es front end y back end**

Front-end y back-end son términos que se refieren a la [separación de intereses](https://es.wikipedia.org/wiki/Separaci%C3%B3n_de_intereses" \o "Separación de intereses) entre una [capa de presentación](https://es.wikipedia.org/wiki/Capa_de_presentaci%C3%B3n" \o "Capa de presentación) y una [capa de acceso a datos](https://es.wikipedia.org/wiki/Capa_de_acceso_a_datos" \o "Capa de acceso a datos), respectivamente. Pueden traducirse al español el primero como interfaz, frontal final o frontal y el segundo como motor, dorsal final[1](https://es.wikipedia.org/wiki/Front-end_y_back-end" \l "cite_note-1)​o zaga, aunque es común dejar estos términos en inglés.

Front end

Trabaja del lado Cliente, en el navegador, en el lado de lo que se ve. Principalmente se ocupa de los componentes externos del sitio web o de la aplicación web. Es el encargado con todo lo que respecta a lo visual de una página o aplicación web.

Back end

El desarrollador back-end trabaja del lado Servidor, detrás del escenario, permitiendo con su trabajo que el usuario disfrute de su experiencia. Sin él, el desarrollo llevado a cabo por su anterior compañero no se sostendría. Es el encargado de trabajar con todo lo relacionado con las bases de datos autenticación y muchas mas cosas.

**https://www.campusmvp.es/recursos/post/Desarrollador-web-Front-end-back-end-y-full-stack-Quien-es-quien.aspx**

1. **¿Qué ES HTML?**

HTML es el lenguaje que se emplea para el desarrollo de páginas de internet. Está compuesto por una seríe de etiquetas que el navegador interpreta y da forma en la pantalla. HTML dispone de etiquetas para imágenes, hipervínculos que nos permiten dirigirnos a otras páginas, saltos de línea, listas, tablas, etc. Podríamos decir que HTML sirve para crear páginas web, darles estructura y contenido.

**https://www.aprenderaprogramar.com/index.php?option=com\_content&view=article&id=435:ique-es-y-para-que-sirve-html-el-lenguaje-mas-importante-para-crear-paginas-webs-html-tags-cu00704b&catid=69&Itemid=192**

1. **Que es pug**

Pug (anteriormente conocido como [Jade](https://github.com/pugjs/pug/issues/2184)) es un motor de plantilla de [Node.js](https://nodejs.org/es/). con el que seremos capaces de escribir código HTML de una sintaxis mucho más sencilla, clara y directa, tanto a la hora de escribir como de leer y modificar.

**https://www.silocreativo.com/introduccion-primeros-pasos-pug/**

1. **¿QUE ES CSS?**

CSS es un lenguaje de hojas de estilos creado para controlar el aspecto o presentación de los documentos electrónicos definidos con HTML y XHTML. CSS es la mejor forma de separar los contenidos y su presentación y es imprescindible para crear páginas web complejas.

Separar la definición de los contenidos y la definición de su aspecto presenta numerosas ventajas, ya que obliga a crear documentos HTML/XHTML bien definidos y con significado completo (también llamados "documentos semánticos"). Además, mejora la accesibilidad del documento, reduce la complejidad de su mantenimiento y permite visualizar el mismo documento en infinidad de dispositivos diferentes.

**http://librosweb.es/libro/css/capitulo\_1.html**

1. **¿QUE ES JavaScript?**

JavaScript (abreviado comúnmente JS) es un [lenguaje de programación](https://es.wikipedia.org/wiki/Lenguaje_de_programaci%C3%B3n) [interpretado](https://es.wikipedia.org/wiki/Int%C3%A9rprete_(inform%C3%A1tica)), dialecto del estándar [ECMAScript](https://es.wikipedia.org/wiki/ECMAScript). Se define como [orientado a objetos](https://es.wikipedia.org/wiki/Programaci%C3%B3n_orientada_a_objetos),[3](https://es.wikipedia.org/wiki/JavaScript#cite_note-3)​ [basado en prototipos](https://es.wikipedia.org/wiki/Programaci%C3%B3n_basada_en_prototipos), [imperativo](https://es.wikipedia.org/wiki/Programaci%C3%B3n_imperativa), débilmente tipado y dinámico.. JavaScript se empezó a utilizar principalmente del lado del cliente permitiendo crear efectos atractivos y dinámicos en las páginas web. Los navegadores modernos interpretan el código JavaScript integrado en las páginas web. Sin embargo javascript se ha vuelto tan popular que incluso ahora ya existe o se puede usar javascript del lado del servidor (Node.js) esto quiere decir que se podría crear una aplicación completa que tanto el front end como el backend esten escritos en javascript, al igual que tambien se pueden crear aplicaciones de escritorio (Electron) móvil android y ios (react-native) por todo esto es que javascript es unos de los lenguajes mas populares actualmente

**https://es.wikipedia.org/wiki/JavaScript**

1. **QUE ES ECMAScript**

Para muchos la palabra ECMA no resulta tan conocida. Se trata de un acrónimo de “European Computer Manufacturers Association (ECMA)”, una organización internacional basada en membresías de estándares para la comunicación y la información. Actualmente, para expresar su alcance internacional, el nombre ya no se considera un acrónimo y no se escribe completamente en mayúsculas.

En el año de 1997 se crea un comité (TC39) en la ECMA para estandarizar JavaScript. A partir de entonces, los estándares de JavaScript se rigen como ECMAScript. No solo JavaScript se basa el lenguaje ECMAScript, existen otros como JScript y ActionScript 3 que también lo hacen. Haciendo una analogía, diremos que ECMAScript es el lenguaje y JavaScript, JScript y ActionScript 3 son dialectos de este lenguaje, siendo JavaScript su dialecto más conocido y utilizado.

Debido a esto no existen versiones propias de JavaScript, sino de su estándar contenedor ECMAScript, que desde el 2015 se encuentra en su versión número 6, esta trae cambios en la sintaxis del lenguaje importantes.

1. **QUE ES BABEL**

Babel.js es un transcompilador que nos permite convertir nuestro código de JavaScript ES6(ECMAScript6) en código de ES5(ECMAScript5). Esta característica se está convirtiendo en algo crítico para mucha gente ya que las nuevas características de ES6 hacen deseable trabajar con el lenguaje lo antes posible. Lamentablemente hoy en día en la mayor parte de los navegadores y distintas plataformas de JavaScript tienen un soporte parcial de ES6. Debido a eso nace Babel para poder trabajar con lo más nuevo del estándar de JavaScript y estar seguros de que va a servir en cualquier navegador.

**https://www.arquitecturajava.com/introduccion-babel-js-javascript-es6/**

1. **Que es Eslint**

ESLint es una herramienta de linting para Javascript, un [linter](https://es.wikipedia.org/wiki/Lint" \t "_blank) es un programa que se encarga de revisar el código escrito y es capaz de señalar errores y posibles bugs que podemos corregir para mejorar nuestros programas.

JavaScript es un lenguaje dinámico y [débilmente tipado](https://es.wikipedia.org/wiki/Tipado_fuerte" \t "_blank) especialmente propenso a errores por parte del programador. Debido a la falta de un proceso de compilación lo que normalmente hacemos para encontrar errores es ejecutarlo, pero aún haciendo esto podemos pasar por alto algunos problemas como variables no declaradas o con el nombre mal escrito.

El propósito de ESLint es encontrar estos errores de manera automática sin tener que ejecutar el código y poder corregirlos manualmente, o incluso algunos errores se pueden corregir de manera automática.

**http://www.enrique7mc.com/2016/09/mejora-la-calidad-de-tu-codigo-con-eslint/**

1. **¿Qué es una SPA (Single page aplication)?**

Un single-page application (SPA), o aplicación de página única es una aplicación web o es un sitio web que cabe en una sola página con el propósito de dar una experiencia más fluida a los usuarios como una aplicación de escritorio. En un SPA todos los códigos de [HTML](https://es.wikipedia.org/wiki/HTML" \o "HTML), [JavaScript](https://es.wikipedia.org/wiki/JavaScript" \o "JavaScript), y [CSS](https://es.wikipedia.org/wiki/Hojas_de_estilo_en_cascada" \o "Hojas de estilo en cascada) se carga de una vez​ o los recursos necesarios se cargan dinámicamente como lo requiera la página y se van agregando, normalmente como respuesta de las acciones del usuario. La página no tiene que cargar otra vez en ningún punto del proceso tampoco se transfiere a otra página. La interacción con las aplicaciones de página única puede involucrar comunicaciones dinámicas con el servidor web que está detrás.

**https://es.wikipedia.org/wiki/Single-page\_application**

1. **¿Qué son frameworks?**

En los sistemas informáticos, un framework es a menudo una estructura en capas que indica qué tipo de programas pueden o deben ser construidos y cómo se interrelacionan. Algunos marcos de trabajo de sistemas informáticos también incluyen programas reales, especifican interfaces de programación u ofrecen herramientas de programación para usar los marcos. Un framework puede servir para un conjunto de funciones dentro de un sistema y cómo se interrelacionan; las capas de un sistema operativo; las capas de un subsistema de aplicación; cómo debería normalizarse la comunicación en algún nivel de una red; etcétera. [Un marco de trabajo es generalmente más completo](http://searchdatacenter.techtarget.com/es/definicion/ITSM-gestion-de-servicios-de-TI) que un protocolo y más prescriptivo que una estructura.

**http://searchdatacenter.techtarget.com/es/definicion/Framework**

1. **QUE ES VUE.JS**

Vue (pronunciado /vjuː/ en inglés, como view) es un framework progresivo para construir interfaces de usuario. A diferencia de otros frameworks monolíticos, Vue está diseñado desde el inicio para ser adoptado incrementalmente. La biblioteca principal se enfoca solo en la capa de la vista, y es muy simple de utilizar e integrar con otros proyectos o bibliotecas existentes. Por otro lado, Vue también es perfectamente capaz de soportar aplicaciones sofisticadas de una sola página (en inglés single-page-application o SPA) cuando se utiliza en combinación con [herramientas modernas](https://es-vuejs.github.io/vuejs.org/v2/guide/single-file-components.html) y [librerías compatibles](https://github.com/vuejs/awesome-vue" \l "libraries--plugins" \t "_blank).

**https://es-vuejs.github.io/vuejs.org/v2/guide/index.html**

1. **QUE ES BOOTSTRAP**

Bootstrap es un [framework](https://es.wikipedia.org/wiki/Framework" \o "Framework) o conjunto de herramientas de [Código abierto](https://es.wikipedia.org/wiki/C%C3%B3digo_abierto" \o "Código abierto) para diseño de sitios y aplicaciones web. Contiene plantillas de diseño con tipografía, formularios, botones, cuadros, menús de navegación y otros elementos de diseño basado en [HTML](https://es.wikipedia.org/wiki/HTML" \o "HTML) y [CSS](https://es.wikipedia.org/wiki/Hojas_de_estilo_en_cascada" \o "Hojas de estilo en cascada), así como, extensiones de [JavaScript](https://es.wikipedia.org/wiki/JavaScript" \o "JavaScript) opcionales adicionales.

Es el proyecto más popular en [GitHub](https://es.wikipedia.org/wiki/GitHub" \o "GitHub)[1](https://es.wikipedia.org/wiki/Bootstrap_(framework)#cite_note-1)​ y es usado por la [NASA](https://es.wikipedia.org/wiki/NASA" \o "NASA) y la [MSNBC](https://es.wikipedia.org/wiki/MSNBC" \o "MSNBC) entre otras organizaciones.

**https://es.wikipedia.org/wiki/Bootstrap\_(framework)**

1. **QUE ES WEBPACK**

Webpack es un empaquetador de módulos, es decir, te permite generar un archivo único con todos aquellos módulos que necesita tu aplicación para funcionar. Para hacerte una idea, te permite meter todos tus archivos javascript en un único archivo, llamémoslo bundle.js.

Como definición inicial basta, pero Webpack va mas allá y se ha convertido en una herramienta de build muy versátil. Entre otras cosas, destaca que:

Puedes generar solo aquellos fragmentos de JS que realmente necesita cada página (haciendo más rápida su carga).

Tiene varios loaders para importar y empaquetar también otros recursos (CSS, templates, …) así como otros lenguajes (ES6 con Babel, TypeScript, SaSS, etc).

Sus plugins te permiten hacer otras tareas importantes como por ejemplo minificar y ofuscar el código.

1. **QUE ES GIT**

Git, es un software de control de versiones diseñado por Linus Torvalds. La pregunta es ¿qué es control de versiones? Pues bien, se define como control de versiones a la gestión de los diversos cambios que se realizan sobre los elementos de algún producto o una configuración del mismo es decir a la gestión de los diversos cambios que se realizan sobre los elementos de algún producto o una configuración, y para los que aún no les queda claro del todo, control de versiones es lo que se hace al momento de estar desarrollando un software o una página web. Exactamente es eso que haces cuando subes y actualizas tu código en la nube, o le añades alguna parte o simplemente le editas cosas que no funcionan como deberían o al menos no como tú esperarías. Las principales funcionalidades e importancias de git son que puedes estar trabajando en arreglar o implementar una nueva funcionalidad en tu proyecto si necesidad de tocar tu código fuente en resumen es hacer una copia de tu código fuente y trabajar en ella y cuando termines y compruebes que ya todo esa funcionando bien puedes agregar los cambios de la copia en el código fuente original llamado master también es muy útil para que varias personas estén trabajando en un mismo proyecto si algún miembro del equipo termina una labor pueda subir los cambios y todos las demás personas del equipo las pueden ver en su código.

**https://codigofacilito.com/articulos/que-es-git**

1. **Que es Node**

Node.js es un entorno JavaScript de lado de servidor que utiliza un modelo asíncrono y dirigido por eventos.

Es una Máquina Virtual tremendamente rápida y de gran calidad escrita por gente como Lars Bak, uno de los mejores ingenieros del mundo especializados en VMs(Virtual Machines). No olvidemos que V8 es actualizado constantemente y es uno de los intérpretes más rápidos que puedan existir en la actualidad para cualquier lenguaje dinámico. Además las capacidades de Node.js para I/O (Entrada/Salida) son realmente ligeras y potentes, dando al desarrollador la posibilidad de utilizar a tope la I/O del sistema.

Uno de los puntos fuertes de Node.js es su capacidad de mantener muchas conexiones abiertas y esperando. En Apache por ejemplo el parámetro MaxClients por defecto es 256. Este valor puede ser aumentado para servir contenido estático, sin embargo si se sirven aplicaciones web dinámicas en PHP u otro lenguaje es probable que al poner un valor alto el servidor se quede bloqueado ante muchas conexiones esto dependerá del trabajo que la aplicación web de lado del servidor y de su capacidad hardware.

**https://devcode.la/blog/que-es-nodejs/**

1. **Que es NPM (Node Package Manager)**

JavaScript ha pasado de ser un lenguaje utilizado para brindar interactividad a las páginas web, a ser utilizado ahora también en el lado de los servidores gracias a Node.js.En Node, el código se organiza por módulos o paquetes, así que al momento de trabajar con él va a ser necesario agregar más módulos, es aquí donde entra npm.

Node Package Manager o simplemente npm es un gestor de paquetes, el cual hará más fáciles nuestras vidas al momento de trabajar con Node, ya que gracias a él podremos tener cualquier librería disponible con solo una línea de código, npm nos ayudará a administrar nuestros módulos, distribuir paquetes y agregar dependencias de una manera sencilla.

Cuando instalamos nuevos paquetes lo que hace npm es instalarlo de manera local en nuestro proyecto dentro de la carpeta node\_modules, pero nosotros podemos decirle que lo instale de manera global de ser necesario.

**https://devcode.la/blog/que-es-npm/**

1. **Que es el package.json**

Es uno de los ficheros elementales dentro de nuestras aplicaciones realizadas en Node.js el package.json. Comprender su utilidad y aprender a utilizarlo es algo bastante rápido, y nos va a ahorrar mucho tiempo durante el desarrollo con Node. Este fichero puede ser propiamente un documento en formato JSON, o un objeto literal de JavaScript, y su objetivo es documentar los paquetes Node, como pueden ser las aplicaciones que desarrollemos. En el se va a guardar datos importantes sobre el proyecto por ejemplo: nombre del creador, nombre del proyecto, su repositorio de alojamiento, la versión actual, una breve descripción, script de ejecuacion y asi como todos sus paquetes que instales mediante npm se van a guardar automáticamente en el package.json esto es lo mas importante desde este documento podemos ver cuales fueron todas los modulos, frameworks o herramientas que se utilizaron para crear el programa.

**https://nodeymongo.wordpress.com/2014/06/12/el-fichero-package-json/**

1. **Que es una base de datos**

Una base de datos o banco de datos es un conjunto de datos pertenecientes a un mismo contexto y almacenados sistemáticamente para su posterior uso. En este sentido; una biblioteca puede considerarse una base de datos compuesta en su mayoría por documentos y textos impresos en papel e indexados para su consulta. Actualmente, y debido al desarrollo tecnológico de campos como la [informática](https://es.wikipedia.org/wiki/Inform%C3%A1tica) y la [electrónica](https://es.wikipedia.org/wiki/Electr%C3%B3nica), la mayoría de las bases de datos están en formato digital, siendo este un componente electrónico, por tanto se ha desarrollado y se ofrece un amplio rango de soluciones al problema del almacenamiento de datos.

Existen [programas](https://es.wikipedia.org/wiki/Programa_inform%C3%A1tico) denominados [sistemas gestores de bases de datos](https://es.wikipedia.org/wiki/Sistema_de_gesti%C3%B3n_de_bases_de_datos), abreviado SGBD (del inglés database management system o DBMS), que permiten almacenar y posteriormente acceder a los datos de forma rápida y estructurada. Las propiedades de estos DBMS, así como su utilización y administración, se estudian dentro del ámbito de la informática.

**https://es.wikipedia.org/wiki/Base\_de\_datos**

1. **Que es Posgresql**
2. **Que es un servidor web**

Un servidor web o servidor HTTP es un [programa informático](https://es.wikipedia.org/wiki/Programa_(computaci%C3%B3n)) que procesa una [aplicación del lado del servidor](https://es.wikipedia.org/wiki/Servidor_web#Aplicaci.C3.B3n_del_lado_del_Servidor), realizando conexiones bidireccionales o unidireccionales y [síncronas o asíncronas](https://es.wikipedia.org/wiki/Sincron%C3%ADa) con el cliente y generando o cediendo una respuesta en cualquier lenguaje o [Aplicación del lado del cliente](https://es.wikipedia.org/wiki/Servidor_web#Aplicaci.C3.B3n_del_lado_del_cliente). El código recibido por el cliente es renderizado por un [navegador web](https://es.wikipedia.org/wiki/Navegador_web). Para la transmisión de todos estos datos suele utilizarse algún protocolo. Generalmente se usa el protocolo [HTTP](https://es.wikipedia.org/wiki/HTTP) para estas comunicaciones, perteneciente a la capa de aplicación del [modelo OSI](https://es.wikipedia.org/wiki/Modelo_OSI). El término también se emplea para referirse al ordenador.

**https://es.wikipedia.org/wiki/Servidor\_web**

1. **QUE ES UNA API REST**

REST cambió por completo la ingeniería de software a partir del 2000. Este nuevo enfoque de desarrollo de proyectos y servicios web fue definido por [Roy Fielding](http://roy.gbiv.com/), el padre de la especificación HTTP y uno los referentes internacionales en todo lo relacionado con la Arquitectura de Redes, en su disertación [‘](http://www.ics.uci.edu/~fielding/pubs/dissertation/top.htm)[Architectural Styles and the Design of Network-based Software Architectures’](http://www.ics.uci.edu/~fielding/pubs/dissertation/top.htm). En el campo de las APIs, REST (Representational State Transfer- [Transferencia de Estado Representacional](https://es.wikipedia.org/wiki/Representational_State_Transfer" \t "_blank)) es, a día de hoy, el alfa y omega del desarrollo de servicios de aplicaciones.

En la actualidad no existe proyecto o aplicación que no disponga de una API REST para la creación de servicios profesionales a partir de ese software. Twitter, YouTube, los sistemas de identificación con Facebook… hay cientos de empresas que generan negocio gracias a REST y las APIs REST. Sin ellas, todo el crecimiento en horizontal sería prácticamente imposible. Esto es así porque REST es el estándar más lógico, eficiente y habitual en la creación de APIs para servicios de Internet.

Buscando una definición sencilla, REST es cualquier interfaz entre sistemas que use HTTP para obtener datos o generar operaciones sobre esos datos en todos los formatos posibles, como XML y JSON.

**https://bbvaopen4u.com/es/actualidad/api-rest-que-es-y-cuales-son-sus-ventajas-en-el-desarrollo-de-proyectos**

1. **QUE ES FETCH**

La API Fetch proporciona un canal para obtener recursos a través de la red. Basándose en un sistema de Peticiones (Request) y Respuestas (Response) permite a un documento o aplicación mantener un diálogo con el servidor de forma segura y semántica. En resumen, sirve para obtener, editar, eliminar o agregar datos por medio de una API a una base de datos.

[**https://developer.mozilla.org/es/docs/Web/API/Fetch\_API/Utilizando\_Fetch**](https://developer.mozilla.org/es/docs/Web/API/Fetch_API/Utilizando_Fetch)

1. **Que es express**
2. **Que es sequelize**
3. **Que son los JWT(Json Web Tokens)**
4. **Que son los Test unitarios y de integración.**

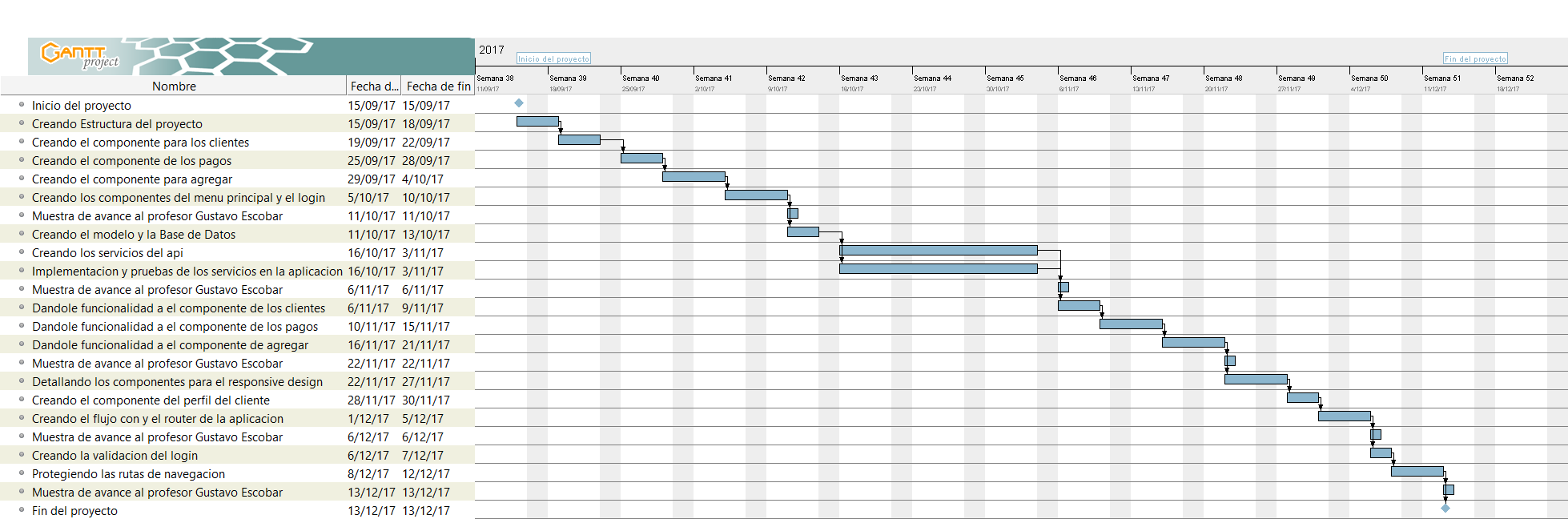
### 

### CAPÍTULO III

### MARCO METODOLÓGICO

### CAPITULO III

### CALENDARIO DE TRABAJO:



**Imagen 1: Calendario de Trabajo**

### TIPOS DE INVESTIGACIÓN

Existen tres tipos de investigación de los cuales surgen los diversos tipos de investigaciones que se realizan y son: Histórica, Descriptiva, Experimental.

La investigación histórica trata de la experiencia pasada, describe lo que era y representa una búsqueda crítica de la verdad que sustenta los acontecimientos pasados. El investigador depende de fuentes primarias y secundarias las cuales proveen la información y a las cuáles el investigador deberá examinar cuidadosamente con el fin de determinar su confiabilidad por medio de una crítica interna y externa. En el primer caso verifica la autenticidad de un documento o vestigio y en el segundo, determina el significado y la validez de los datos que contiene el documento que se considera auténtico.

La investigación descriptiva, según se mencionó, trabaja sobre realidades de hecho y su característica fundamental es la de presentar una interpretación correcta. Esta puede incluir los siguientes tipos de estudios: Encuestas, Casos, Exploratorios, Causales, De Desarrollo, Predictivos, De Conjuntos, De Correlación.

La investigación experimental consiste en la manipulación de una (o más) variable experimental no comprobada, en condiciones rigurosamente controladas, con el fin de describir de qué modo o por qué causa se produce una situación o acontecimiento particular. El experimento provocado por el investigador, le permite introducir determinadas variables de estudio manipuladas por él, para controlar el aumento o disminución de esas variables y su efecto en las conductas observadas.

### TIPO DE INVESTIGACION A UTILIZAR

La investigación aplicada: Se caracteriza por su interés en la aplicación, utilización y consecuencias prácticas de los conocimientos. La investigación aplicada busca el conocer para hacer, para actuar, para construir, para modificar.

**RECOLECCIÓN DE DATOS:**

Reuniones de Grupo

Son reuniones de grupos de personas que dialogan y discuten entre ellas sobre un tema previamente establecido, con la finalidad de encontrar una solución a un determinado problema o de proporcionar información sobre éste, hacer sugerencia de algún producto o servicio, encontrar causas de determinadas situaciones, cambiar características de productos, etc.

Con el debate que se genera en estas sesionesengrupo, se espera que ese calor de la discusión que se genera apoye el surgimiento de ideas, opiniones, actitudes que no detectamos en una entrevista en profundidad.

**ESTUDIOS DE DESARROLLO: (Métodos de análisis de información que les permite llegar a conclusiones que les permita trabajar mejor y definir las tareas a completar con base en la información recolectada)**

**EVALUACIÓN: (Resultados)**

### MÉTODO DE DESARROLLO DE SISTEMAS: (Prototipo, cascada, en caso de ser un proyecto de redes deben describir si es un proyecto con topologías tipo estrella, bus, etc)

### METODOLOGÍA PROGRAMACIÓN

La programación orientada a objetos es un paradigma de programación que utiliza la abstracción para crear modelos basados ​​en el mundo real. Utiliza diversas técnicas de paradigmas previamente establecidas, incluyendo la modularidad, polimorfismo y encapsulamiento. Hoy en día, muchos lenguajes de programación (como Java, JavaScript, C#, C++, Python, PHP, Ruby y Objective-C) soportan programación orientada a objetos (POO).

La programación orientada a objetos puede considerarse como el diseño de software a través de un conjunto de objetos que cooperan, a diferencia de un punto de vista tradicional en el que un programa puede considerarse como un conjunto de funciones, o simplemente como una lista de instrucciones para la computadora. En la programación orientada a objetos, cada objeto es capaz de recibir mensajes, procesar datos y enviar mensajes a otros objetos. Cada objeto puede verse como una pequeña máquina independiente con un papel o responsabilidad definida.

### METODOLOGÍA DE LA CREACIÓN DE LA BD

Una base de datos relacional es una colección de elementos de datos organizados en un conjunto de tablas formalmente descritas desde la que se puede acceder a los datos o volver a montarlos de muchas maneras diferentes sin tener que reorganizar las tablas de la base. La base de datos relacional fue inventada por E.F. Codd en IBM en 1970.

La interfaz estándar de programa de usuario y aplicación a una [base de datos](http://searchdatacenter.techtarget.com/es/definicion/Base-de-datos) relacional es el [lenguaje de consultas estructuradas (SQL)](http://searchdatacenter.techtarget.com/es/definicion/SQL-o-lenguaje-de-consultas-estructuradas). Los comandos de SQL se utilizan tanto para consultas interactivas para obtener información de una base de datos relacional y para la recopilación de datos para los informes.

Además de ser relativamente fáciles de crear y acceder, una base de datos relacional tiene la importante ventaja de ser fácil de extender. Después de la creación original de una base de datos, una nueva categoría de datos se puede añadir sin necesidad de que todas las aplicaciones existentes sean modificadas.

### TIPO DE BASE DE DATOS

MySQL es un sistema de gestión de base de datos relacional (RDBMS) de código abierto, basado en lenguaje de consulta estructurado (SQL).

[MySQL se ejecuta en prácticamente todas las plataformas, incluyendo Linux, UNIX y Windows](http://searchdatacenter.techtarget.com/es/tutoriales/Normalizacion-de-bases-de-datos-en-MySQL-Cuatro-pasos-faciles-y-rapidos). A pesar de que se puede utilizar en una amplia gama de aplicaciones, MySQL se asocia más con las aplicaciones basadas en la web y la publicación en línea y es un componente importante de una pila empresarial de código abierto llamado LAMP. LAMP es una plataforma de desarrollo web que utiliza Linux como sistema operativo, Apache como servidor web, MySQL como sistema de gestión de base de datos relacional y PHP como lenguaje de programación orientado a objetos (a veces, Perl o Python se utiliza en lugar de PHP).

MySQL, que fue concebido originalmente por la compañía sueca MySQL AB, fue adquirida por Oracle en 2008.

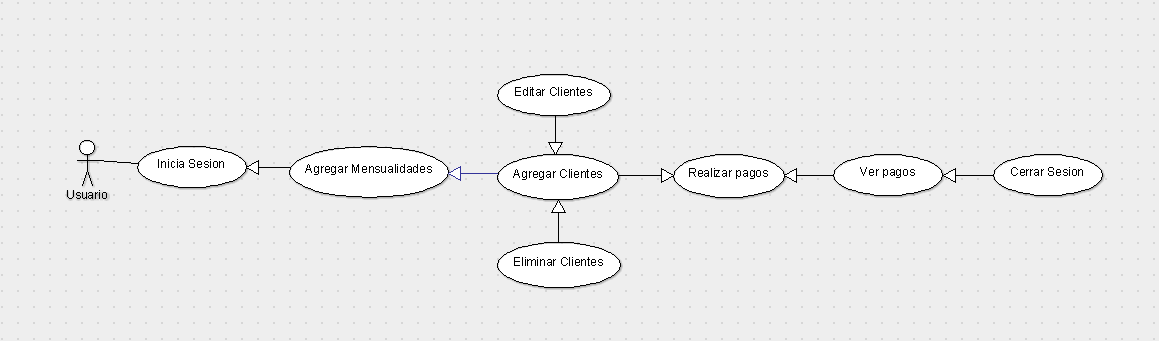
### EURÍSTICA, REGLAS Y MÉTODOS DE PROGRAMACIÓN A UTILIZAR

### TIPO DE APLICACIÓN (CLIENTE-SERVIDOR, WEB, ETC)

En la [ingeniería de software](https://es.wikipedia.org/wiki/Ingenier%C3%ADa_de_software) se denomina aplicación web a aquellas [herramientas](https://es.wikipedia.org/wiki/Aplicaci%C3%B3n_inform%C3%A1tica) que los usuarios pueden utilizar accediendo a un [servidor web](https://es.wikipedia.org/wiki/Servidor_web) a través de [internet](https://es.wikipedia.org/wiki/Internet) o de una [intranet](https://es.wikipedia.org/wiki/Intranet) mediante un [navegador](https://es.wikipedia.org/wiki/Navegador_web). En otras palabras, es un programa que se codifica en un lenguaje interpretable por los navegadores web en la que se confía la ejecución al navegador.

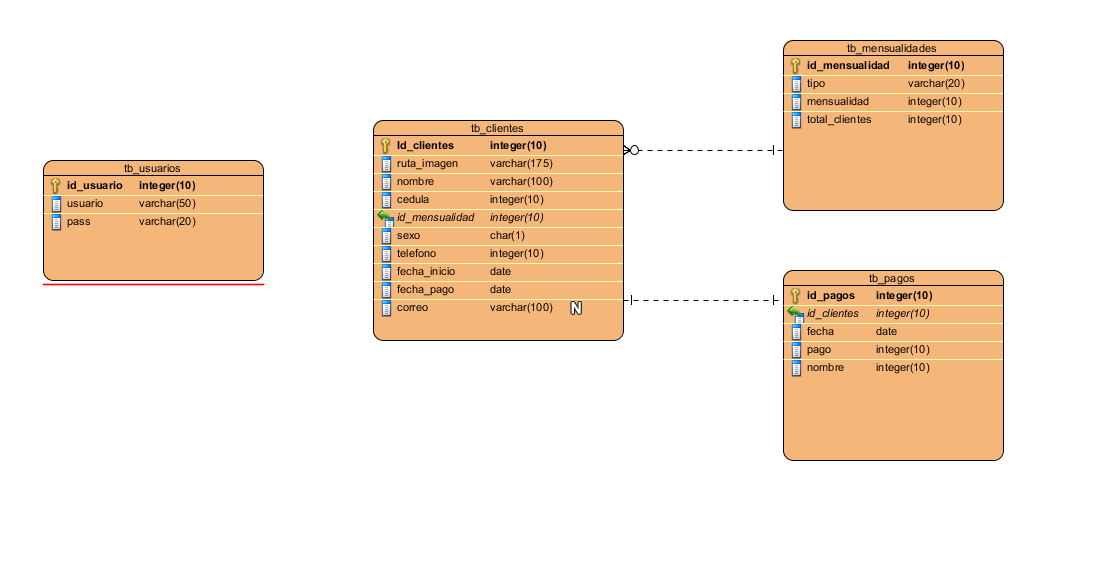
Las aplicaciones web son populares debido a lo práctico del [navegador web](https://es.wikipedia.org/wiki/Navegador_web) como [cliente ligero](https://es.wikipedia.org/wiki/Cliente_ligero), a la independencia del sistema operativo, así como a la facilidad para actualizar y mantener aplicaciones web sin distribuir e instalar software a miles de usuarios potenciales.

### DIAGRAMA DE FLUJO DE PROCESO

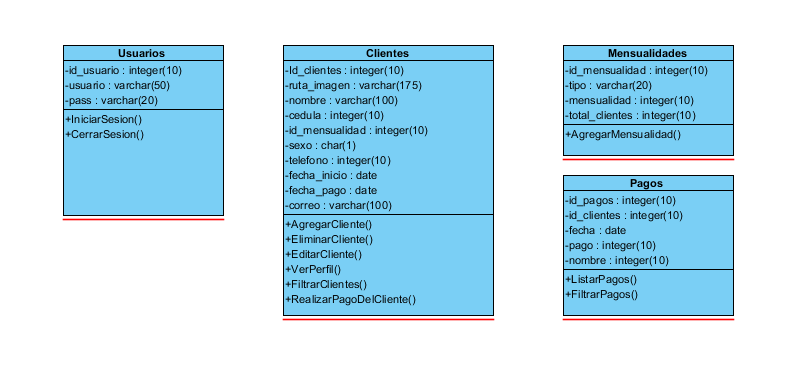


**Imagen 2: Diagrama de flujo de proceso.**

### DIAGRAMA ENTIDAD – RELACION

**Imagen 3: Diagrama entidad relación.**

### DIAGRAMA DE CLASES DE PROGRAMACION

**Imagen 3: Diagrama de clases de programación.**

### DIAGRAMA DE LA BD

### DIAGRAMAS UML

Capítulo IV

Análisis E Interpretación de los Resultados

**CAPÍTULO IV**

**FACTIBILIDADES DEL ANÁLISIS ACTUAL (CUADROS COMPARATIVOS, GRÁFICOS, ESTADÍSTICAS): Es necesario que creen cuadro de análisis donde muestren los beneficios o avances que tiene el modelo de negocio al implementar el proyecto en su trabajo. Estadísticas de tiempo de atención al cliente, por ejemplo, efectividad, etc.**

**FACTIBILIDAD TÉCNICA: Demostración mediante un análisis que técnicamente es un proyecto factible (me refiero a que es realizable y alcanzable con los conocimientos adquiridos, o sea que se puede hacer y quedar bien).**

**FACTIBILIDAD OPERATIVA: Quiero que demuestren que es un beneficio operacional la implementación del modelo de negocio, osea, quiero que expliquen de qué forma evita tiempos muertos, evita desórdenes de facturas, etc…**

**FACTIBILIDAD ECONÓMICA: Es sumamente rentable ya que lo único necesario es tener internet y un dispositivo compatible con cualquier navegador web, por ejemplo: una computadora una tableta o un celular.**

**FACTIBILIDAD LEGAL: Todas las tecnologías utilizadas en el proyecto son open source (Código Abierto) por lo cual no hay o va a ver ningún inconveniente legal.**

**REQUERIMIENTOS DEL HARDWARE: Ya que el proyecto es web y esta transpirado a código legible y ejecutable con cualquier navegador los requerimientos del hardware son mínimos necesita únicamente tener antena wi-fi para conectarse a una red y un navegador web.**

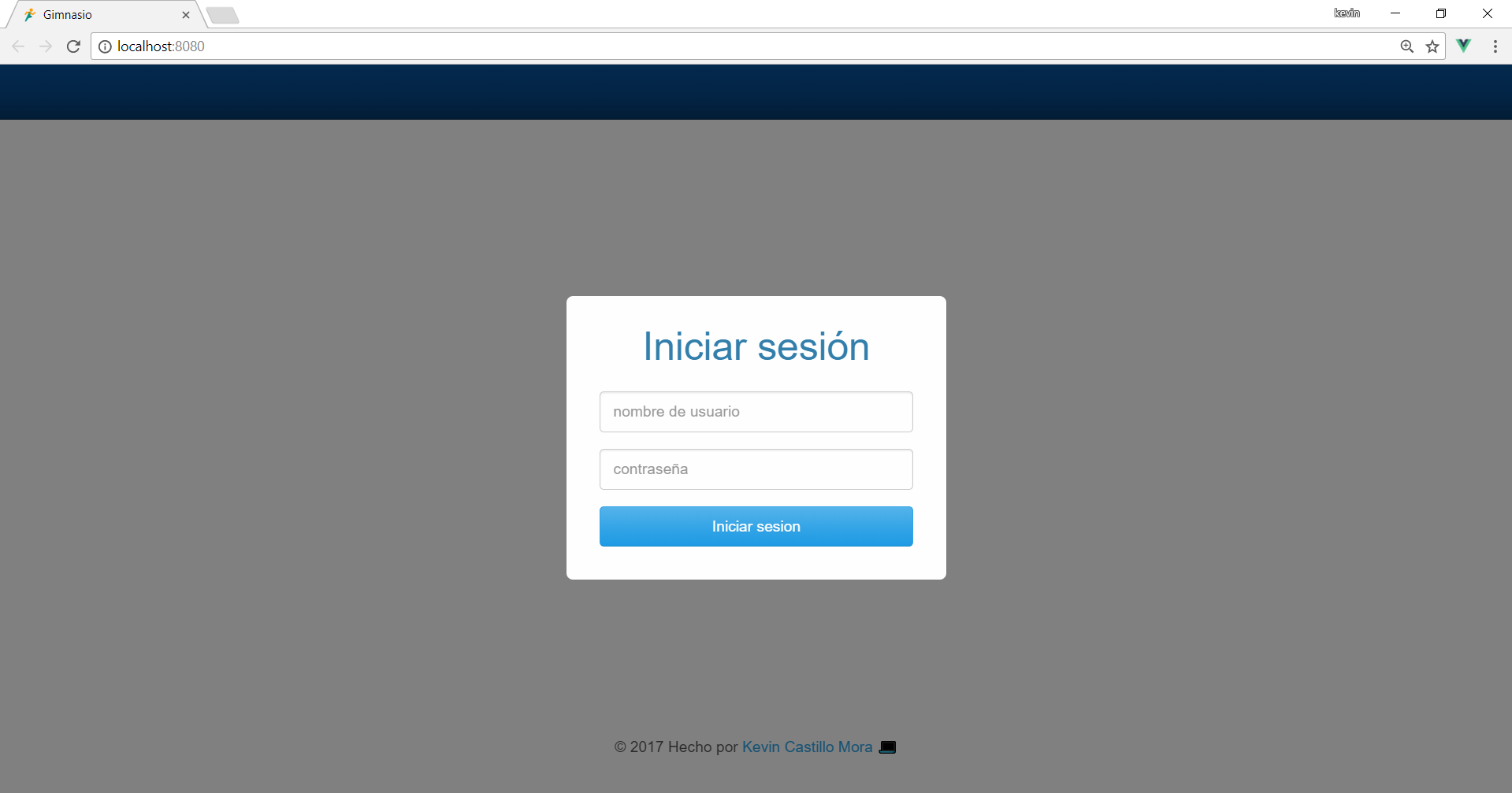
**REQUERIMIENTOS DEL SOFTWARE: La única licencia necesaria es la de un navegador web y todas son gratis.**

**ANALISIS CRÍTICO DE LOS RESULTADOS: Análisis o resumen de todas las factibilidades anteriores y conclusiones de las verdades encontradas.**

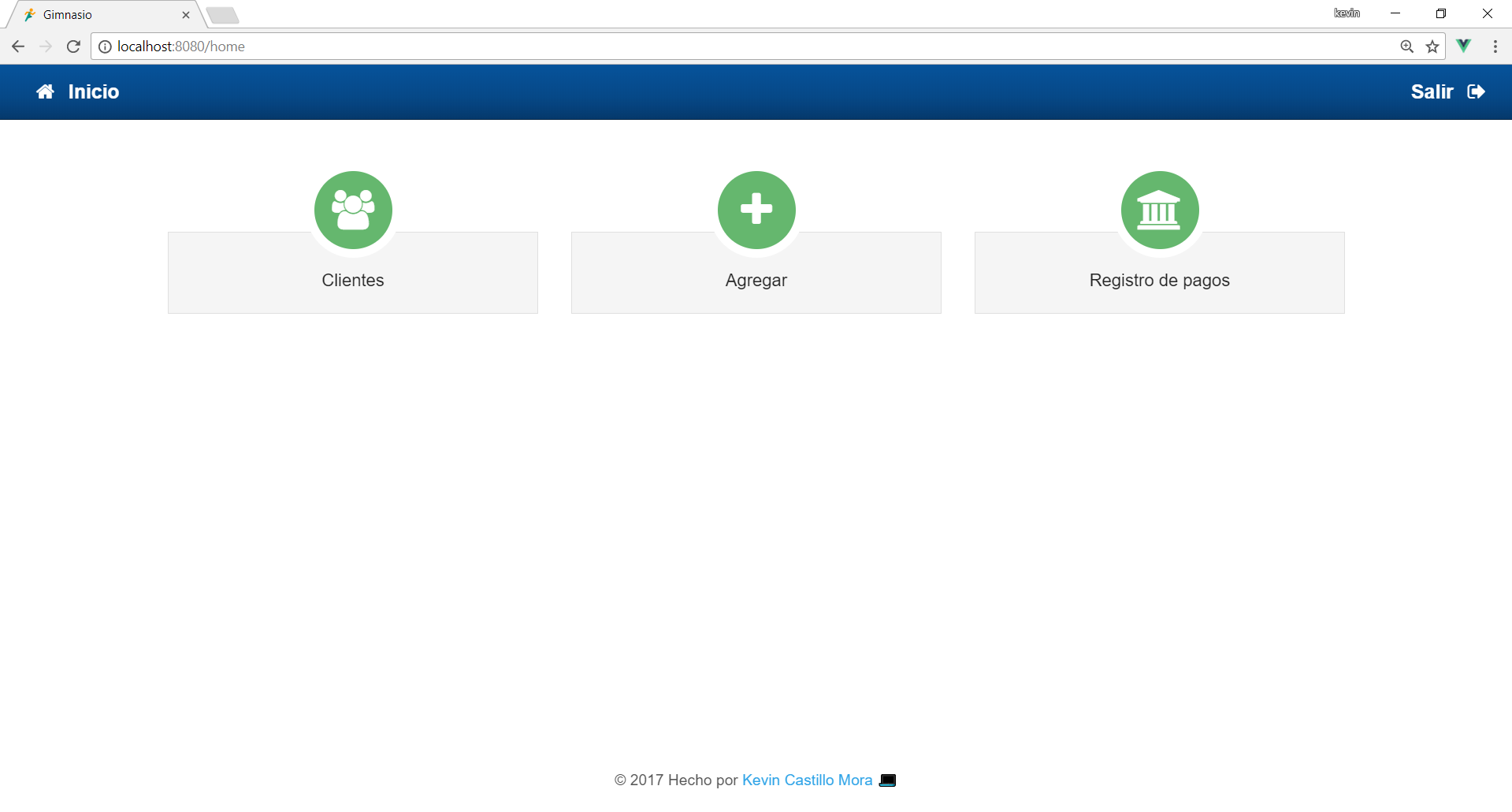
**PERFIL DE USUARIOS: Solo existe un tipo de usuario y está encargado de poder agregar clientes, realizar pagos, actualizar el perfil de los clientes, eliminar, agregar mensualidades y agregar otros usuarios.**

**CARACTERTICAS DEL SISTEMA: Módulos\*\*\*\***

**La primera pantalla es la de inicio de sesión.**

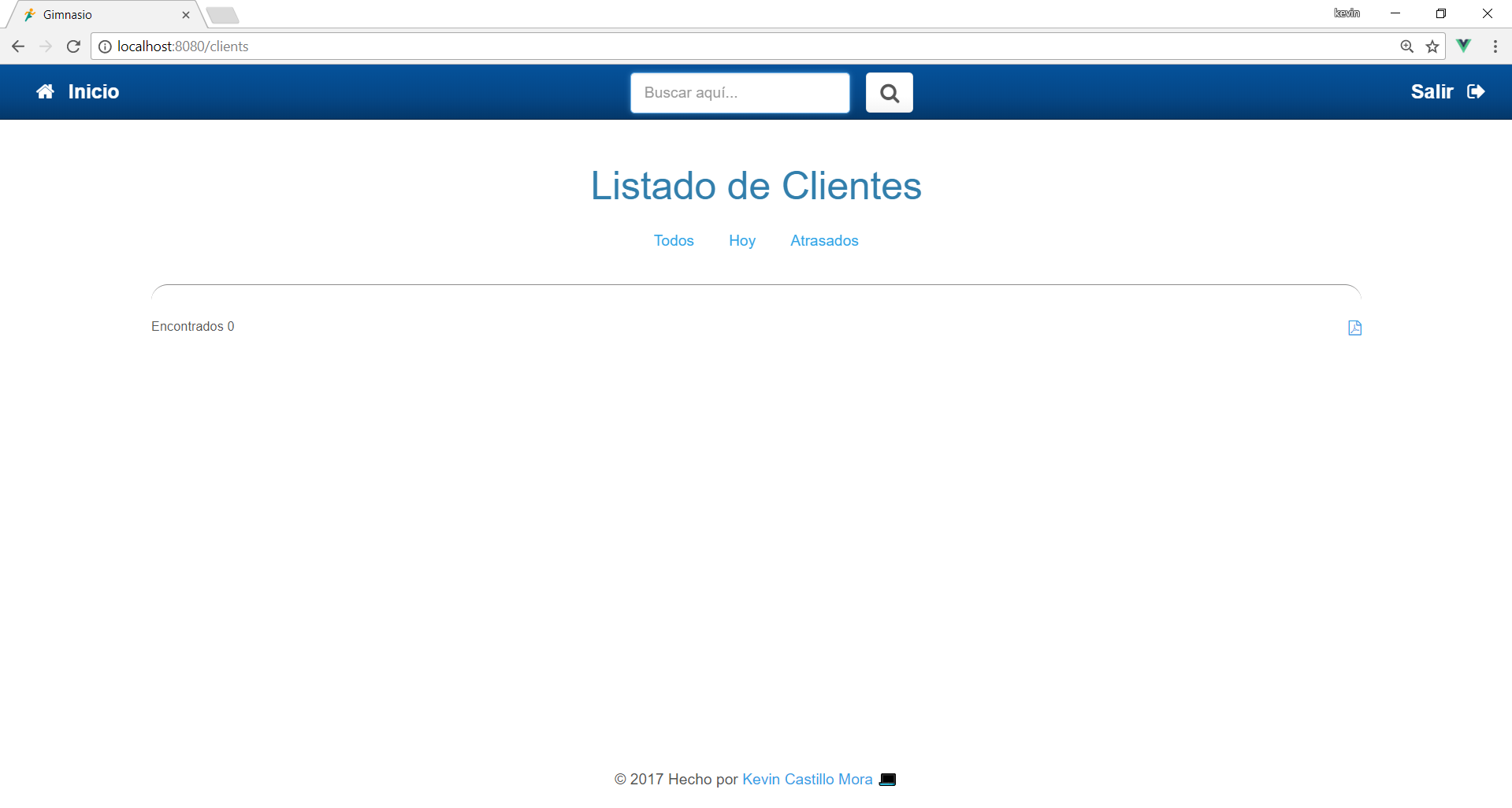
**Imagen 4: Pantalla de inicio de sesión.**

**Después de haber iniciado sesión la segunda pantalla que podemos ver es la del menú de opciones con 3 categorías diferentes clientes, agregar y registro de pagos.**



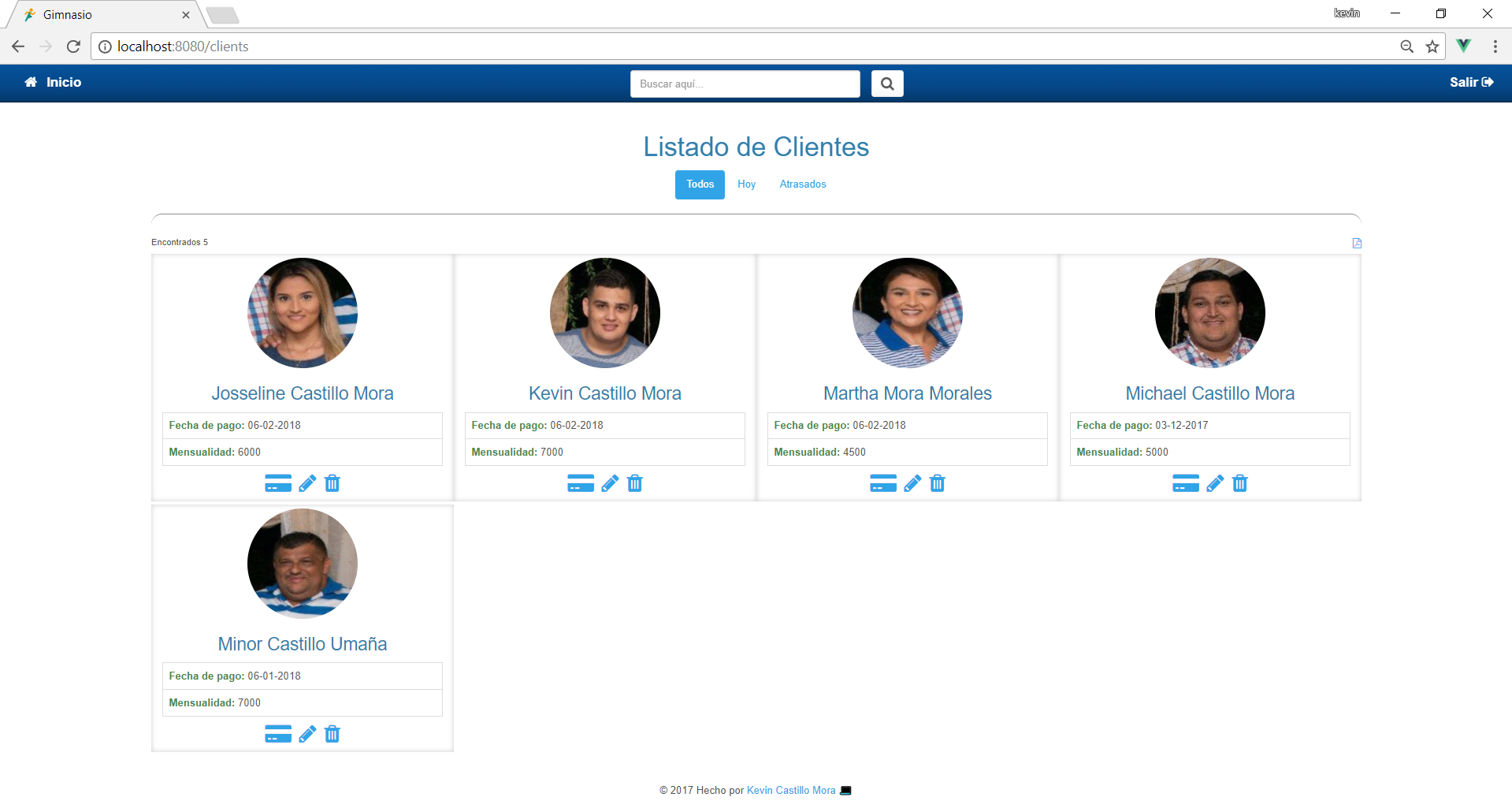
**Imagen 5: Pantalla de menu principal**

**Al darle click a la pestaña de clientes lo primero que vamos a ver son 3 pestañas: Todos, Hoy, Atrasados y una barra de búsqueda arriba.**



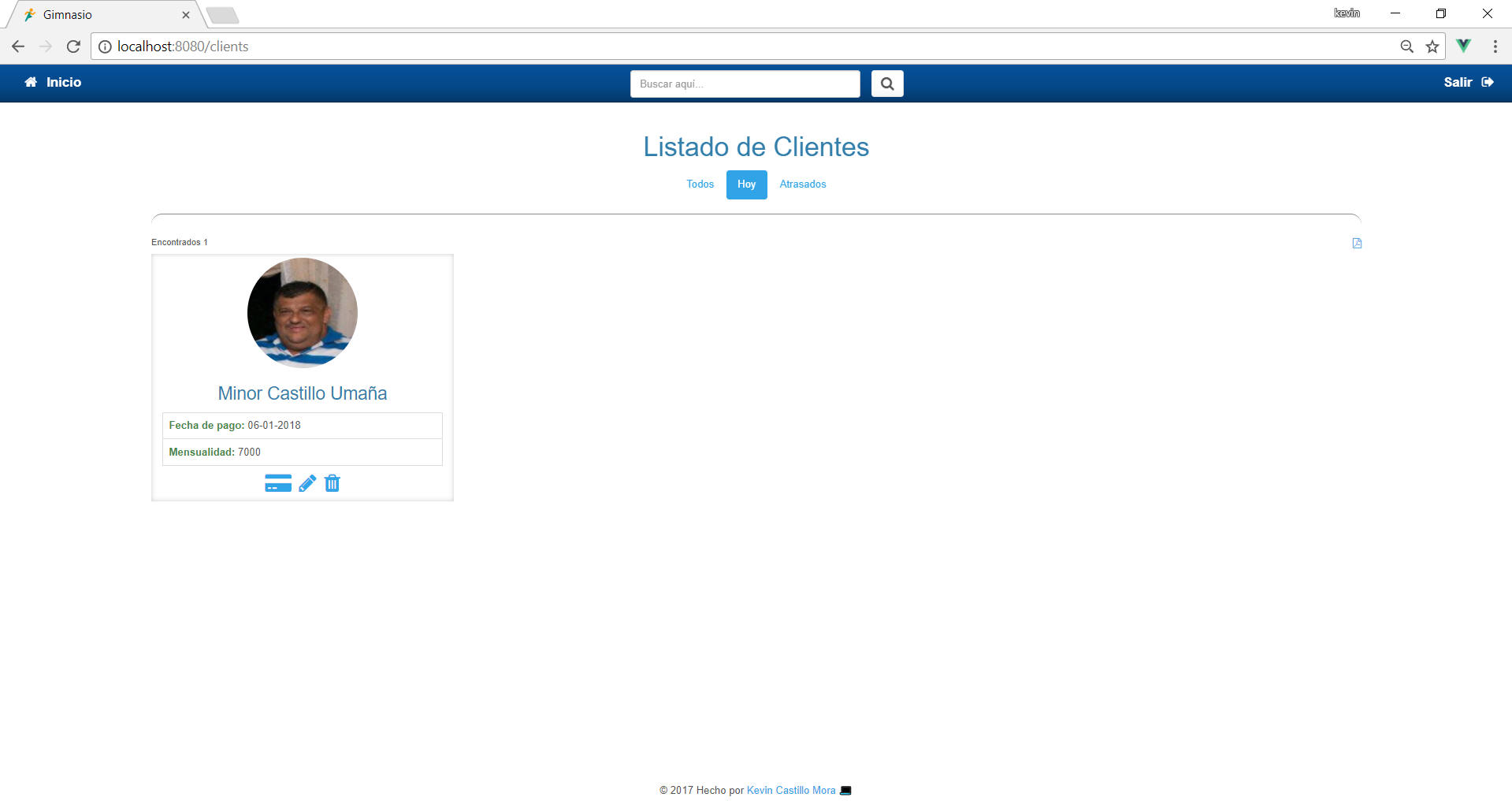
**Imagen 6: Sección de listado de cliente**

**La primera pestaña nos mostrara una pequeña tarjeta con los principales datos de todos los clientes: Foto, Fecha de pago, Mensualidad.**



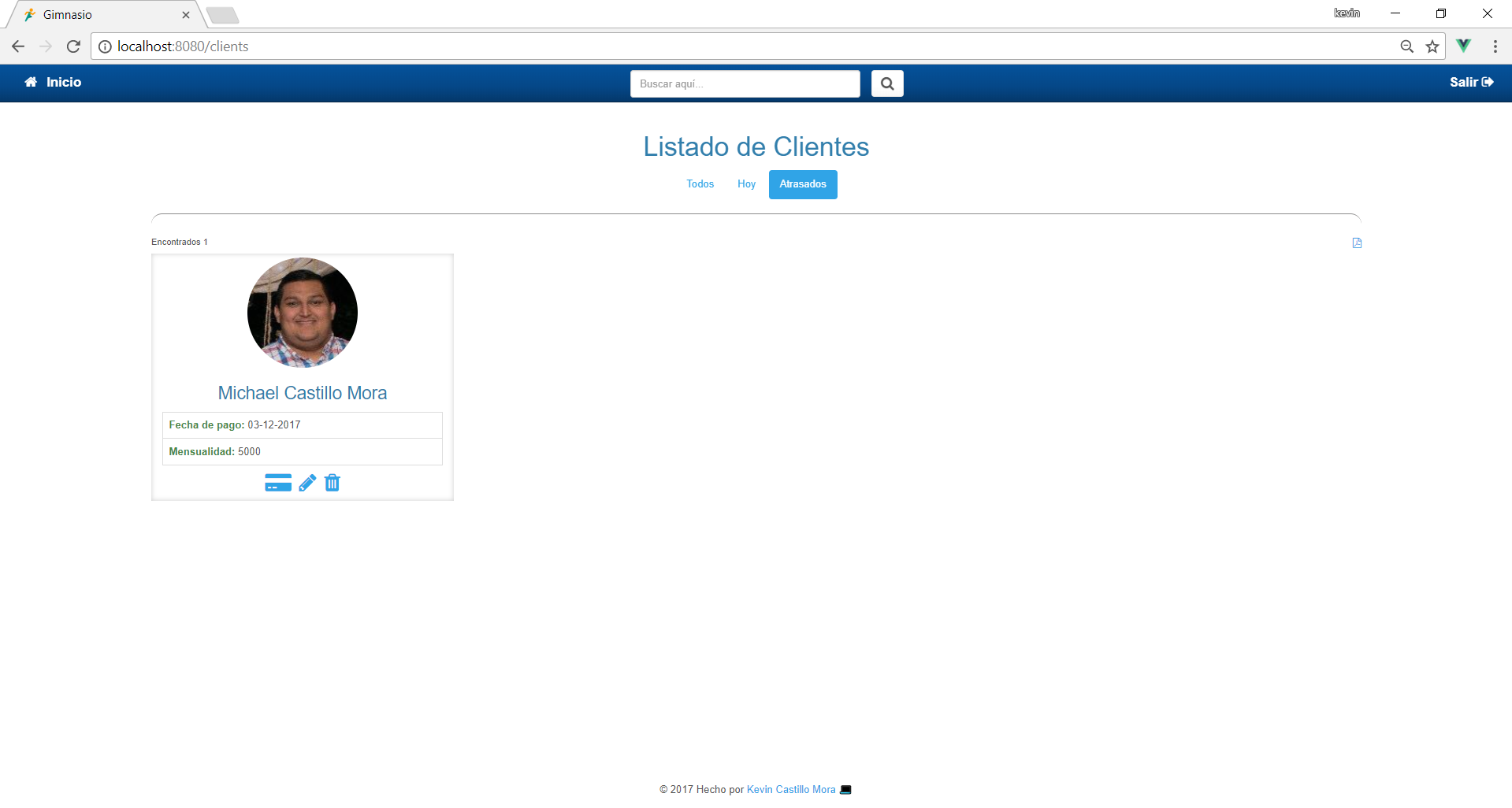
**Imagen 7: Sección de listado de clientes (Todos)**

**La segunda pestaña nos mostrara la misma vista pero solo con los clientes que su fecha de pago sea igual a la del día actual.**



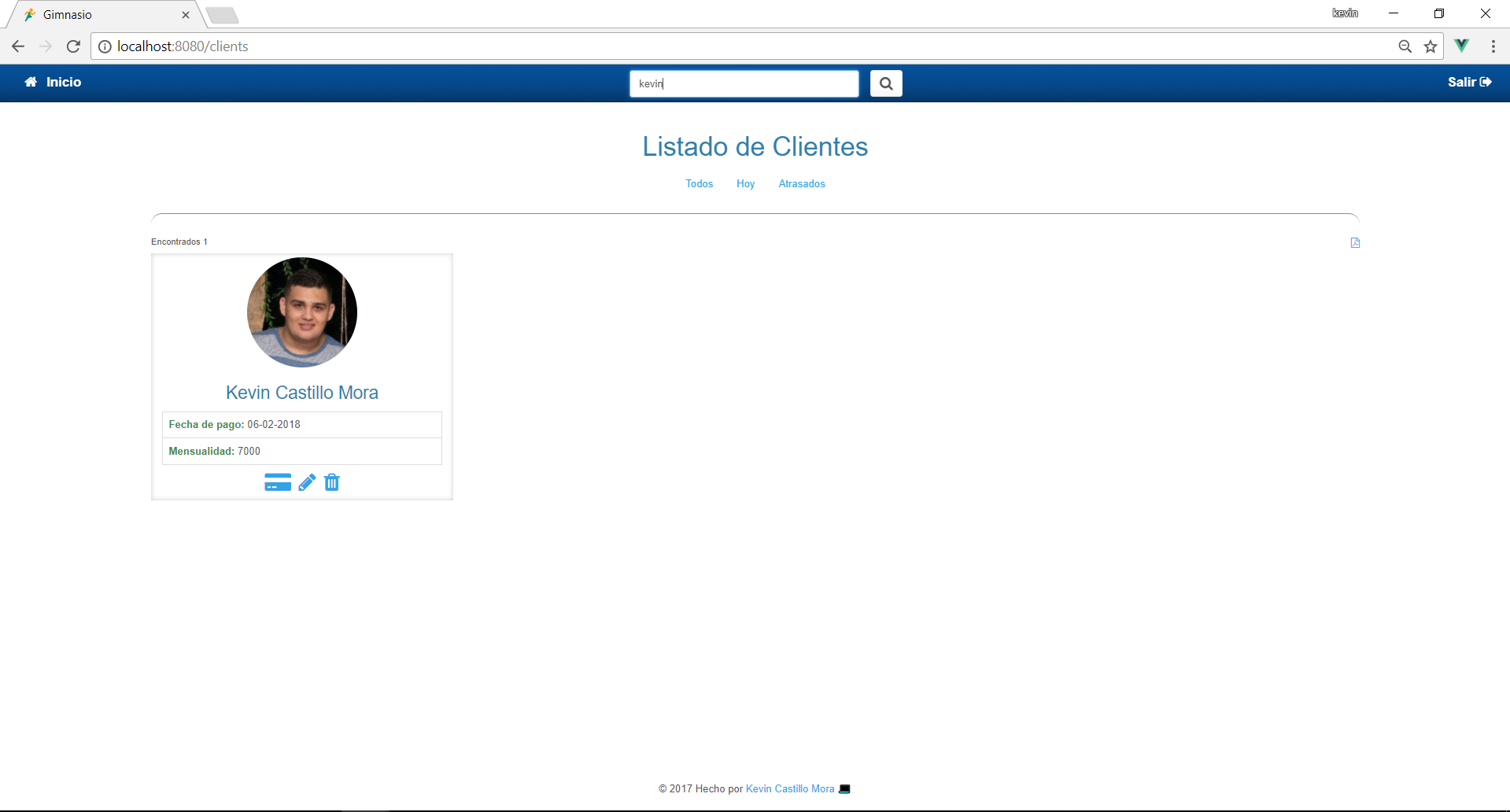
**Imagen 8: Sección de listado de clientes (Hoy)**

**La tercera pestaña al igual que la primera y segunda nos muestra la misma vista pero solo con los clientes que su fecha de pago es anterior a la actual.**



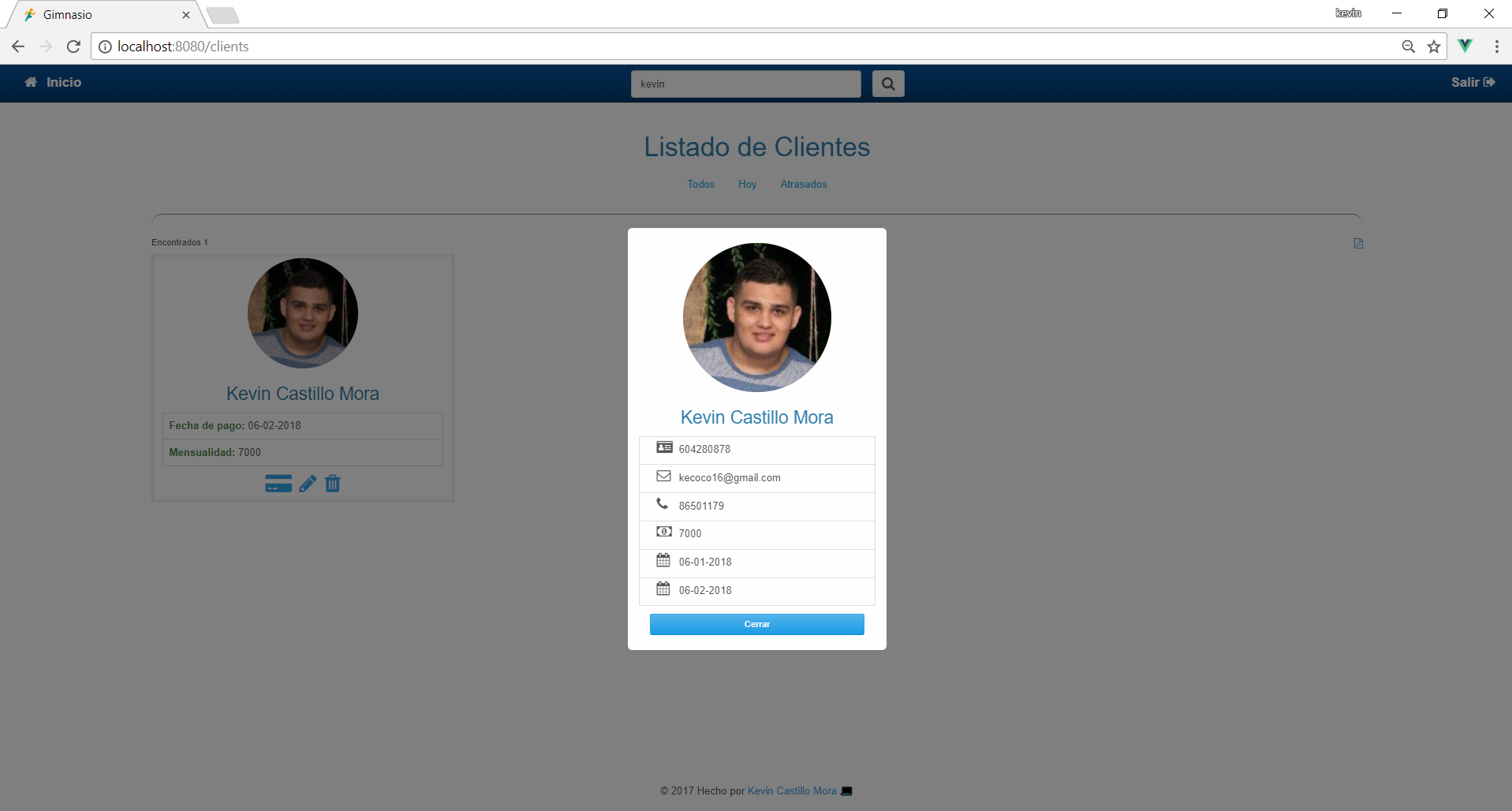
**Imagen 9: Sección de listado de clientes (Atrasados)**

**Esas son las 3 pestañas de clientes sin embargo también se puede hacer una búsqueda del cliente por medio del nombre.**



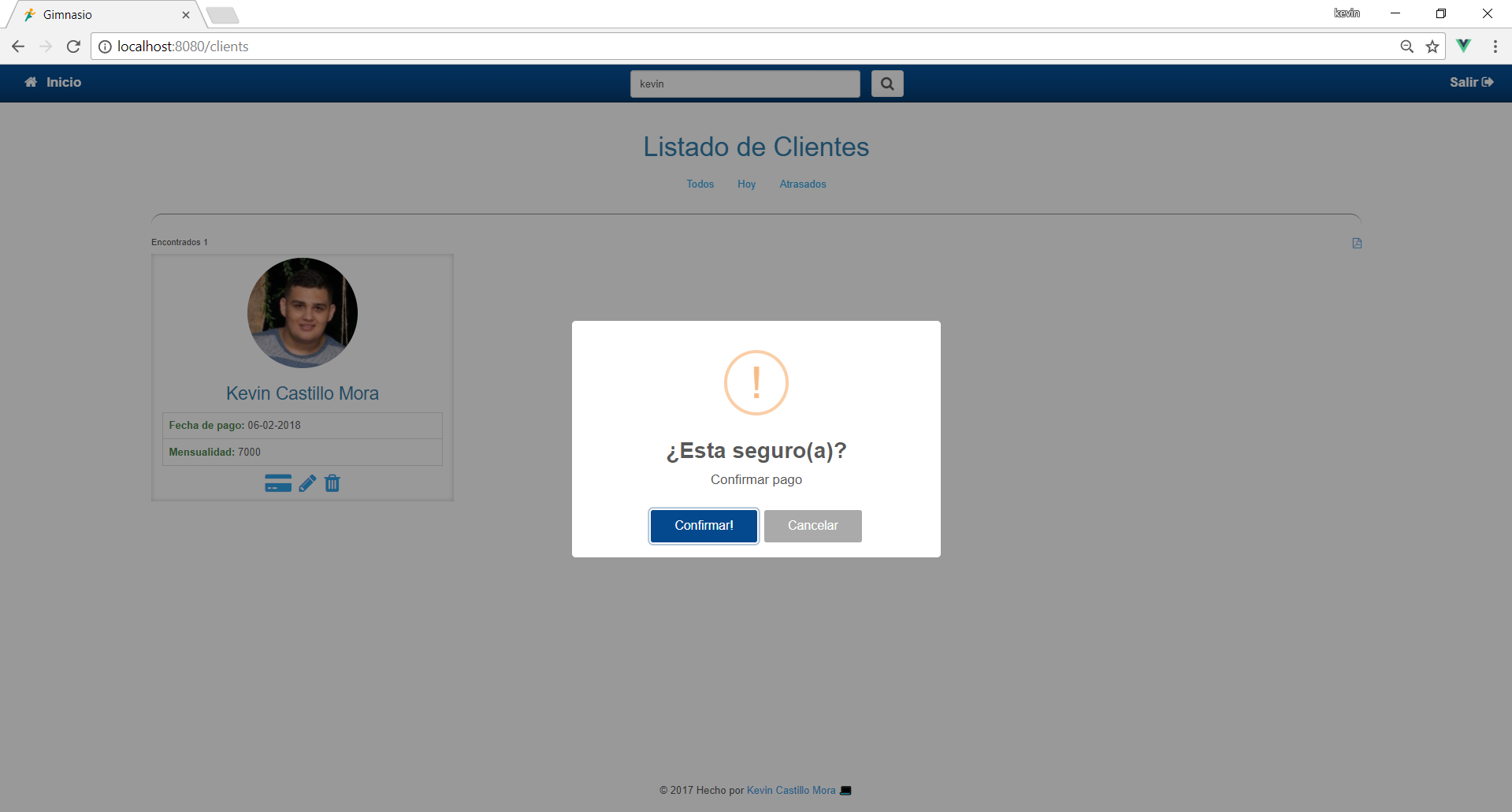
**Imagen 9: Sección de listado de clientes (buscar por nombre)**

**Cada tarjeta de los clientes tiene varias funcionalidades la primera es al darle click a la foto nos va a mostrar el perfil completo de la persona.**



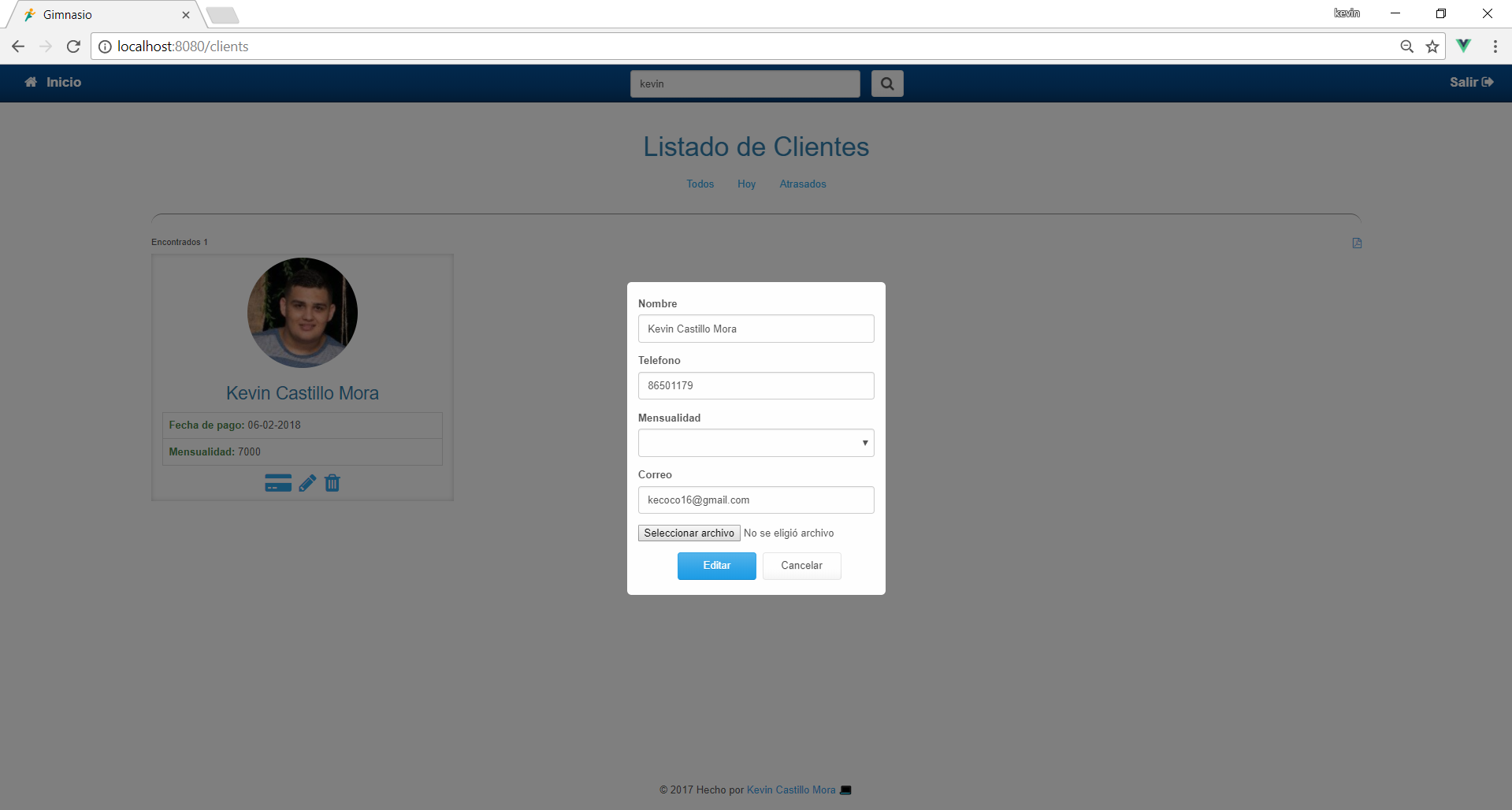
**Imagen 10: Sección de perfil del cliente.**

**Otra funcionalidad es el botón de la tarjeta lo que hace es indicar que el cliente va a realizar el pago de su mensualidad y al darle click nos muestra una opción para confirmar.**



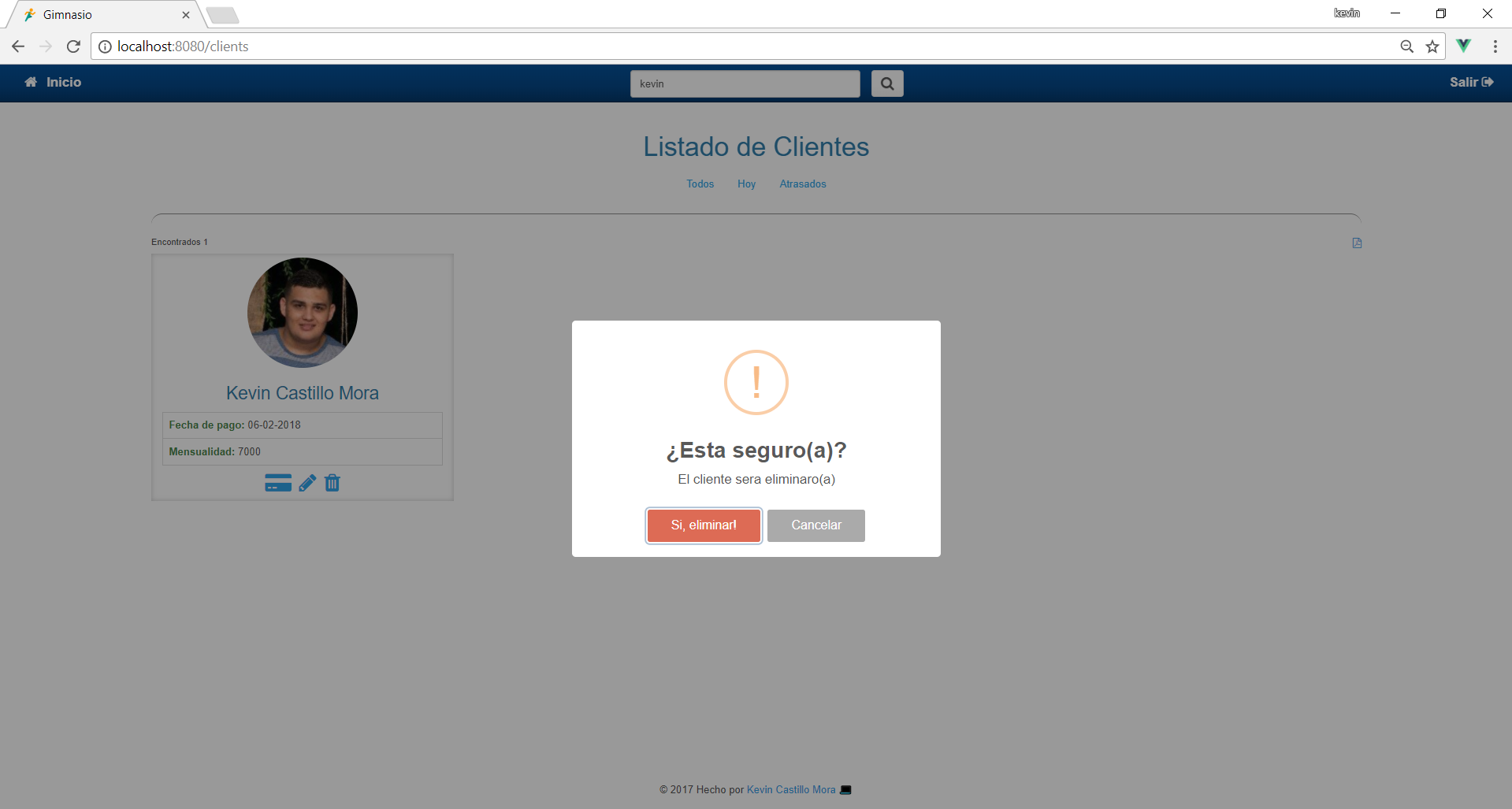
**Imagen 11: Sección de pago de mensualidad.**

**También tenemos la posibilidad de editar información de los clientes por medio del icono del lápiz esto nos mostrara los datos y nos habilita la acción de poder editarlos.**



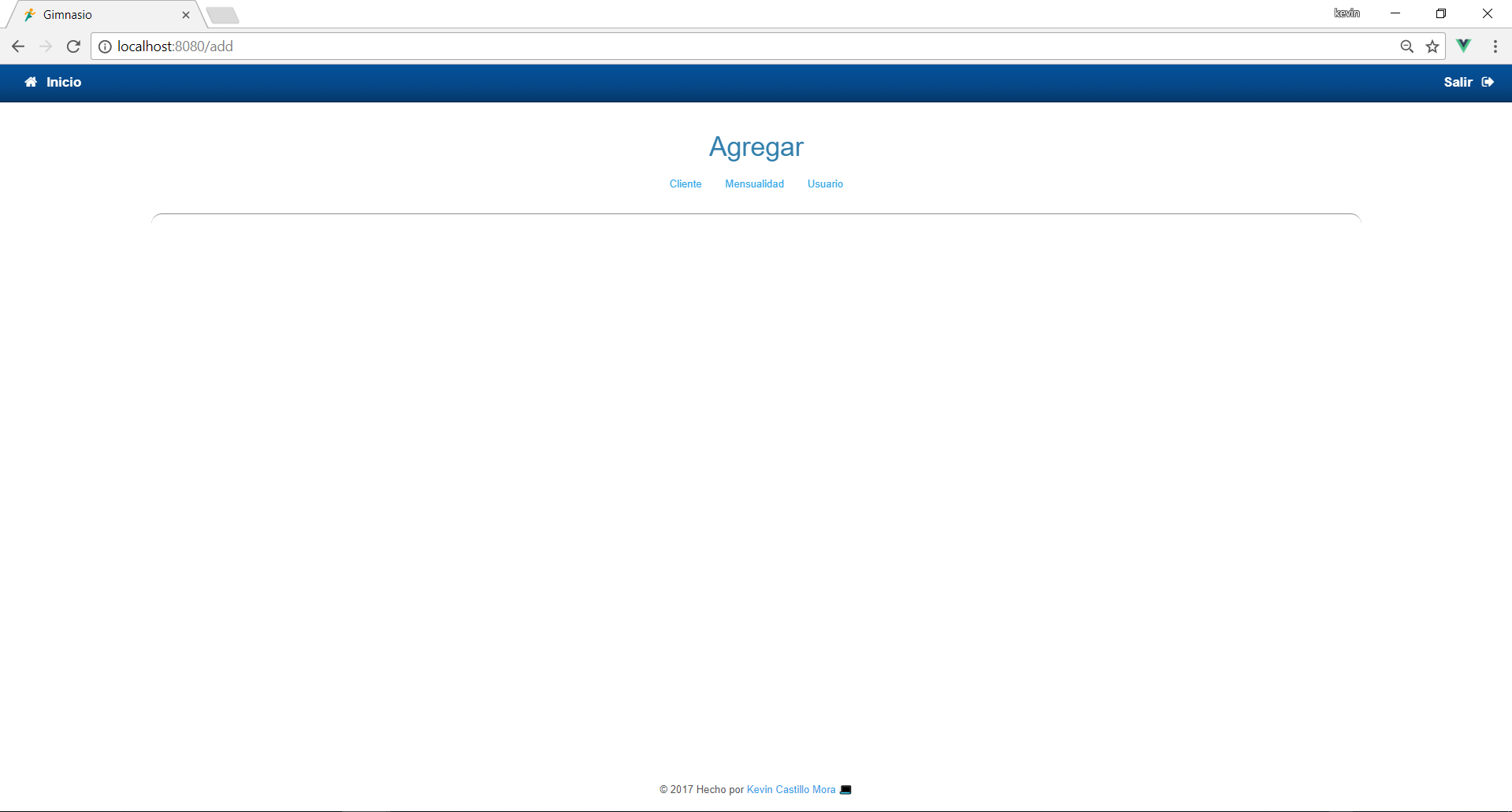
**Imagen 12: Sección de editar un cliente.**

**La última funcionalidad de las tarjetas es la de eliminar a un cliente por medio del icono de la basura también nos mostrara una opción para confirmar.**



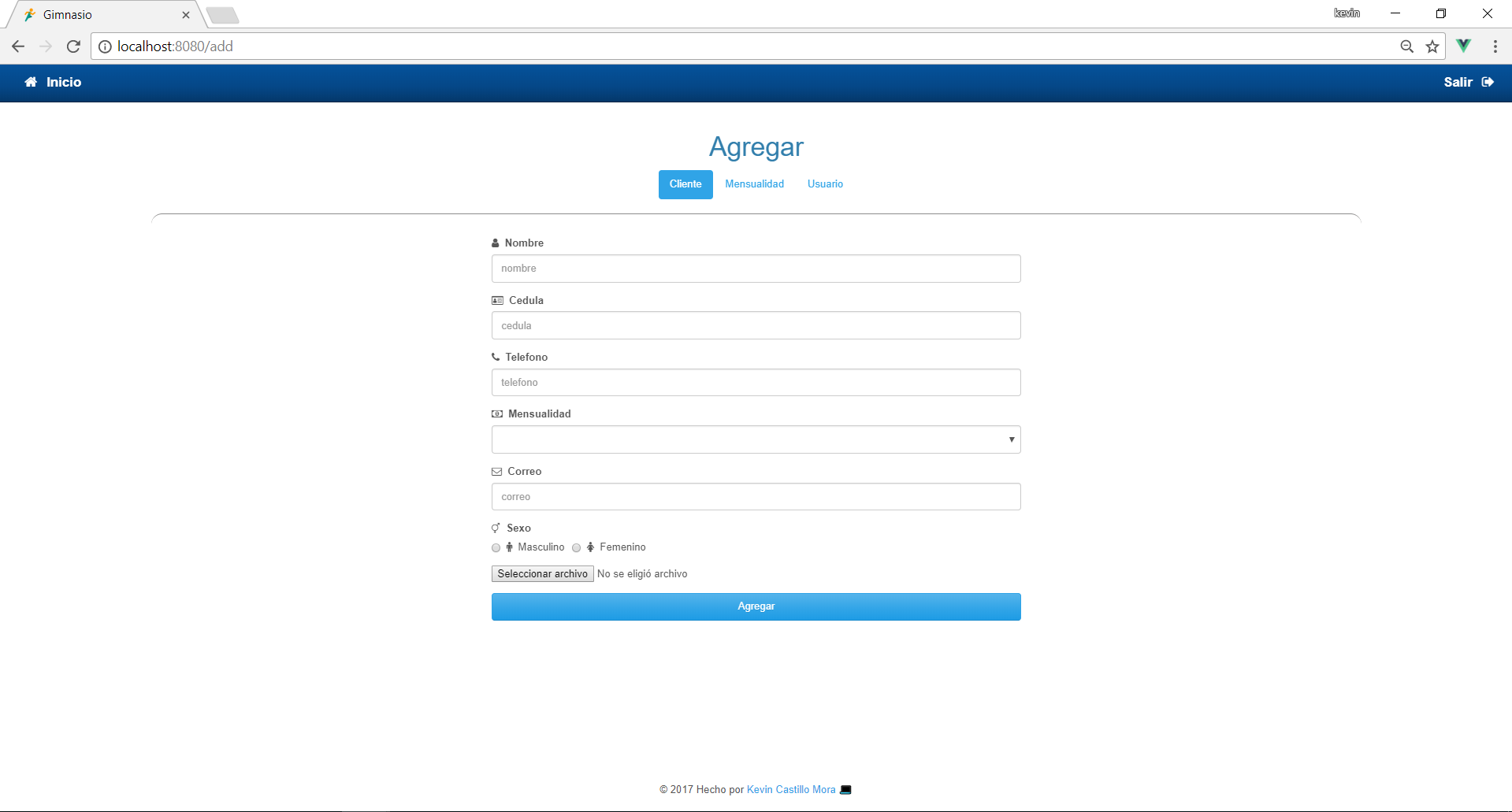
**Imagen 13: Sección de eliminar un cliente**

**Estas son todas las funcionalidades de la pestaña de clientes luego nos devolvemos al menú usando el icono y la palabra de inicio ubicada siempre en la esquina superior izquierda y seleccionamos la pestaña de agregar esta va a ser la vista principal que contiene 3 pestañas: Cliente, Mensualidad, Usuario.**



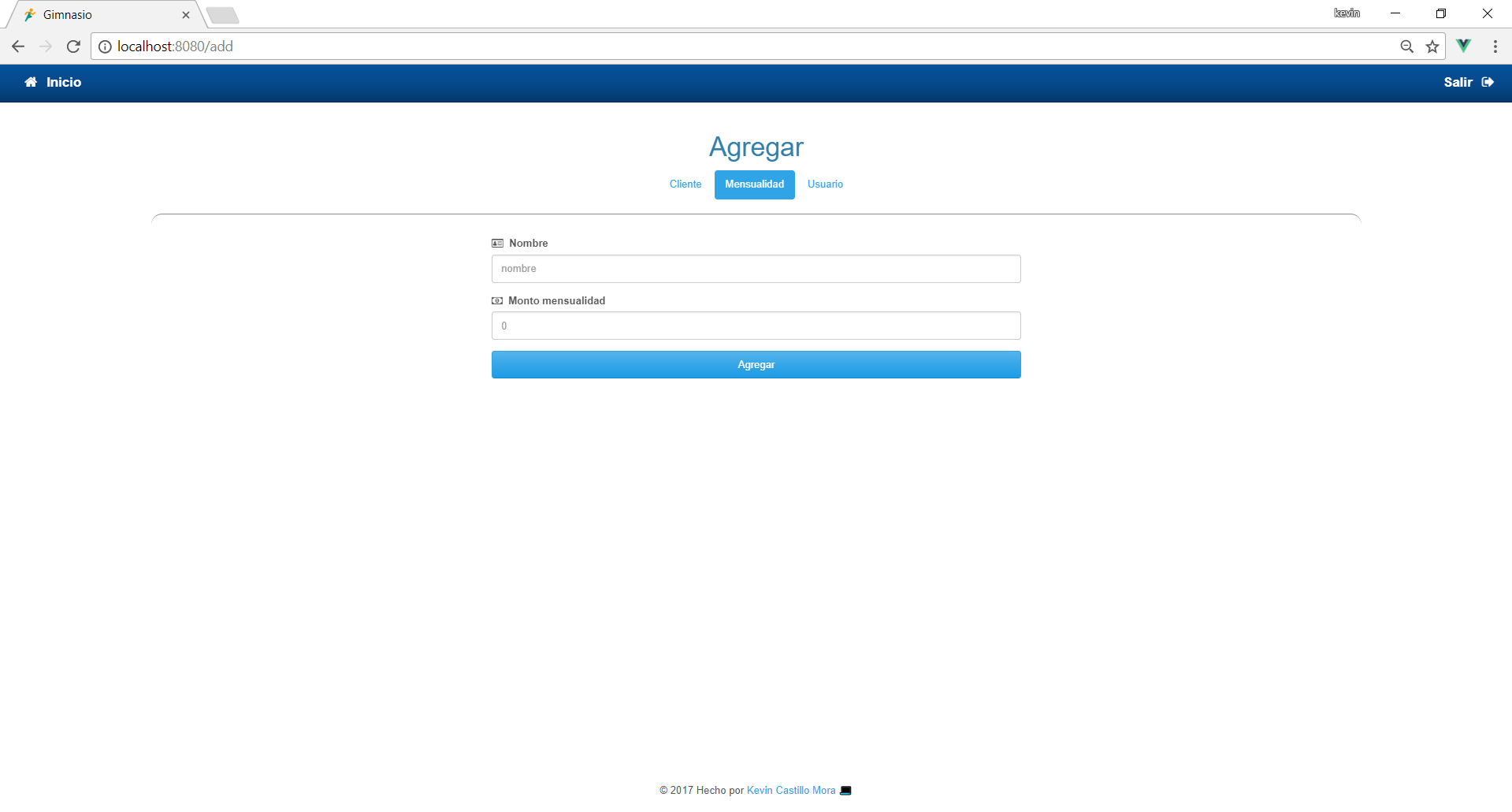
**Imagen 14: Sección de agregar**

**Al darle click a la pestaña de clientes nos aparecerá el formulario para agregar a un cliente nuevo.**



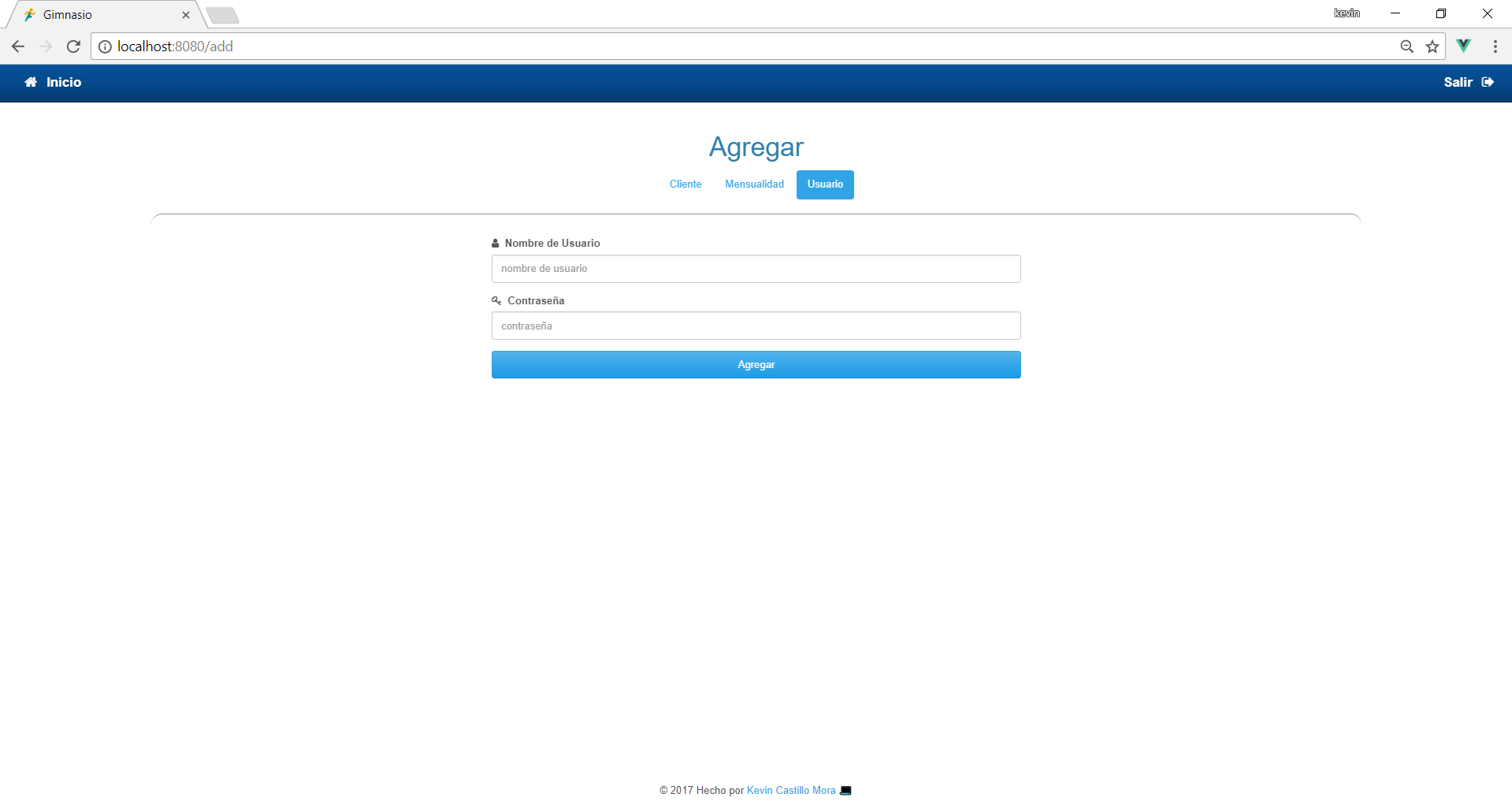
**Imagen 15: Sección de agregar un cliente.**

**La segunda pestaña de mensualidad nos abre el formulario para agregar una nueva mensualidad.**



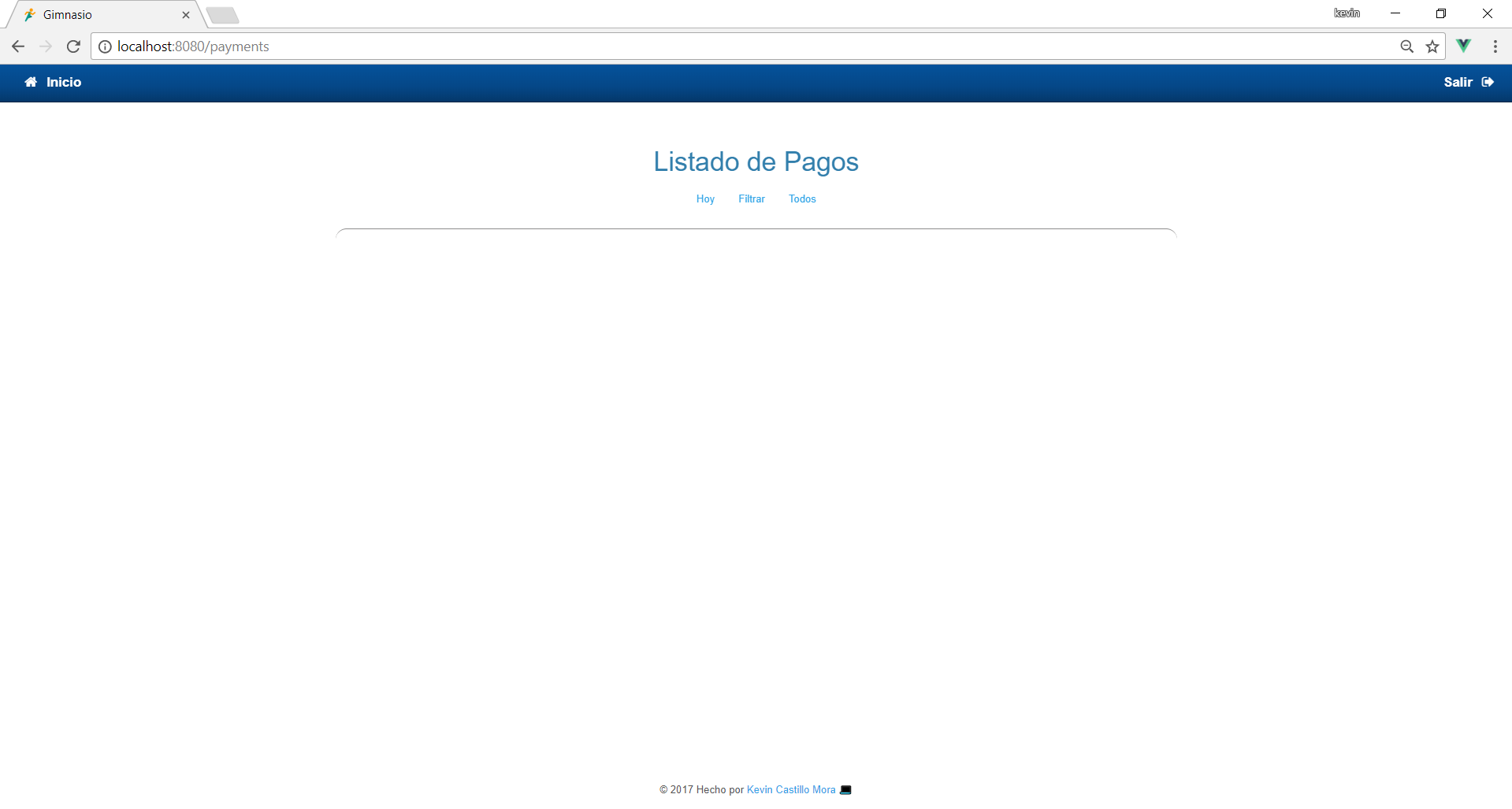
**Imagen 16: Sección de agregar una mensualidad.**

**Y la tercera y última pestaña de esta sección es la de agregar a un usuario nuevo que también nos muestra el formulario para hacerlo.**



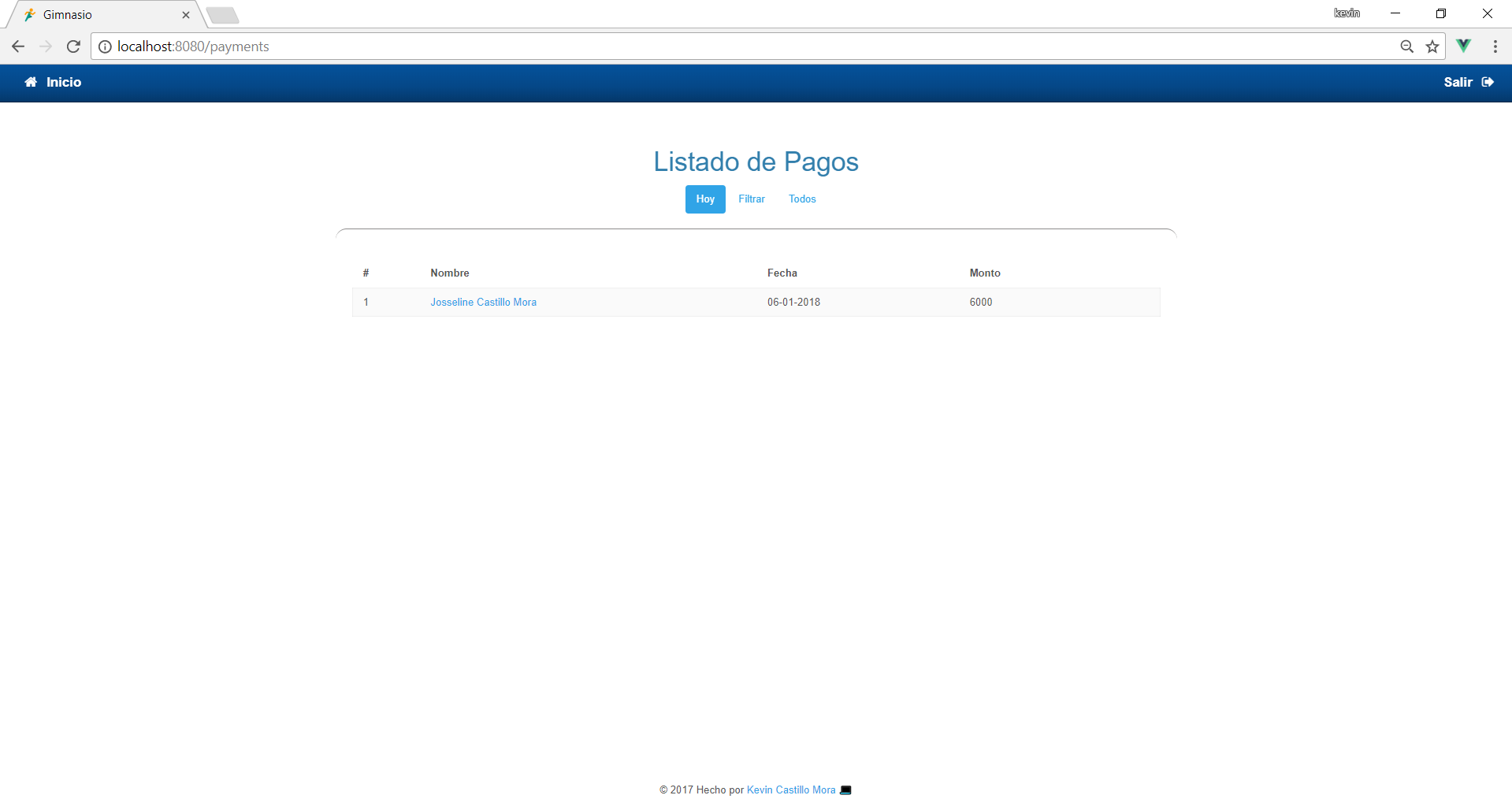
**Imagen 17: Sección de agregar un usuario.**

**Luego de esto nos podemos devolver de nuevo al menú principal con el icono de inicio y podemos seleccionar la ultima pestaña de registro de pagos que a primera vista se van a ver 3 diferentes pestañas: Hoy, Filtrar, Todos.**



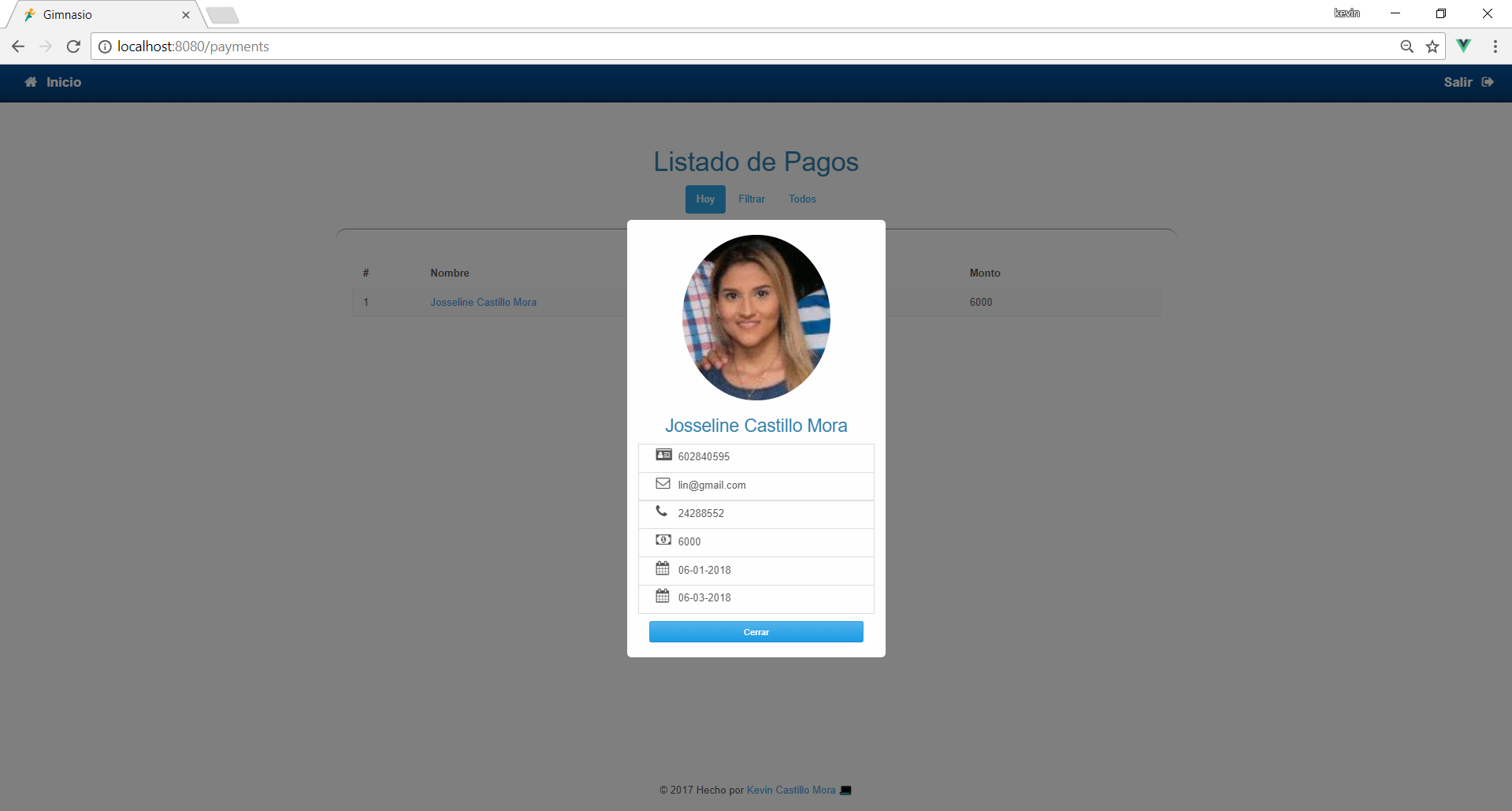
**Imagen 18: Sección de listado de pagos.**

**En la pestaña de hoy solo vamos a observar la información de los pagos que hubo en el día.**



**Imagen 19: Sección de listado de pagos de hoy.**

**También podemos observar que al darle click al nombre de la persona nos muestra un cuadro con toda su información de contacto.**



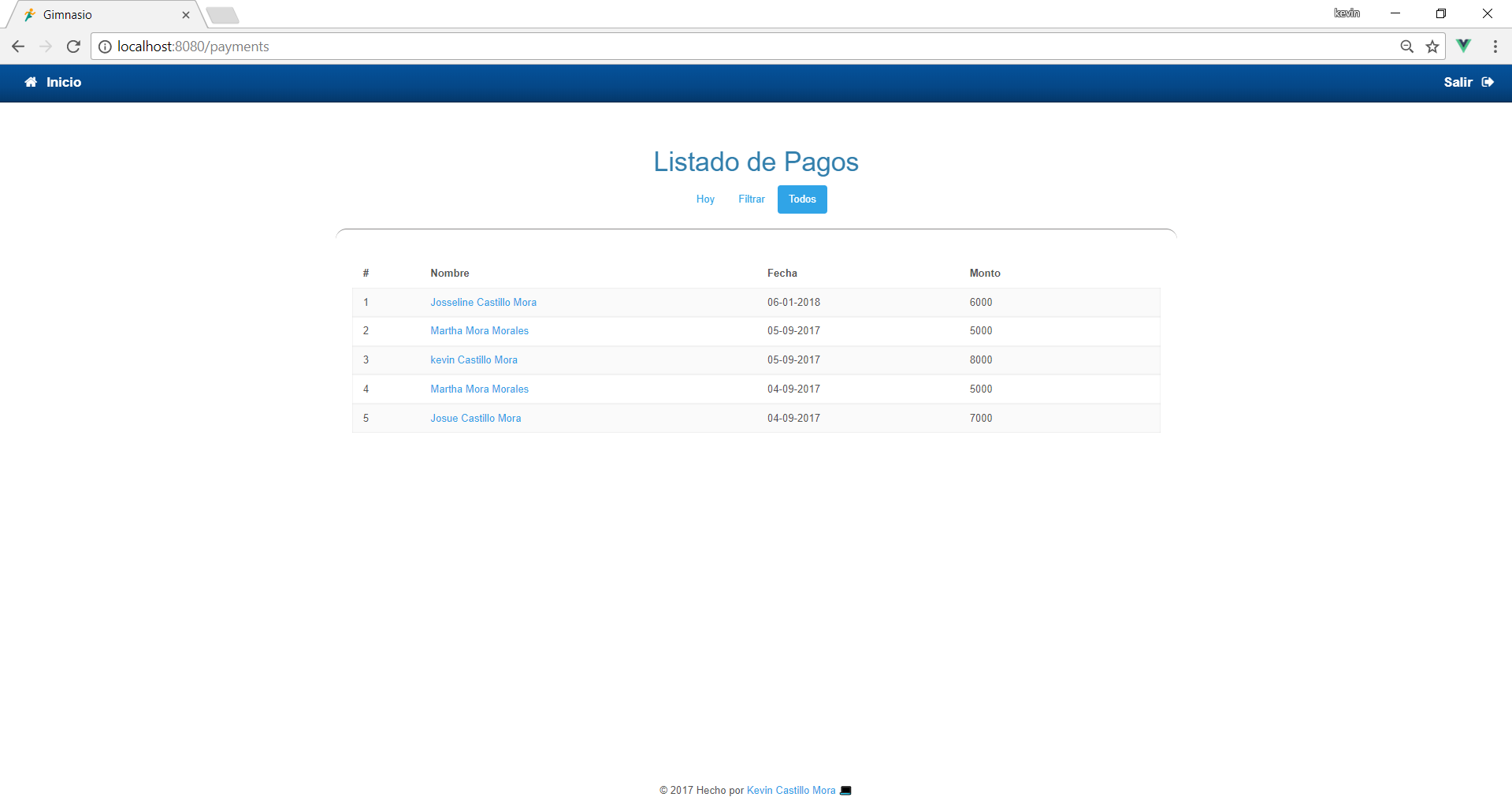
**Imagen 20: Perfil del cliente desde sección de pagos.**

**En la pestaña de filtrar podemos hacer una búsqueda selectiva entre fechas seleccionadas al darle click nos aparecerá el filtro donde debemos de seleccionar las fechas en las cuales queremos buscar los pagos.**



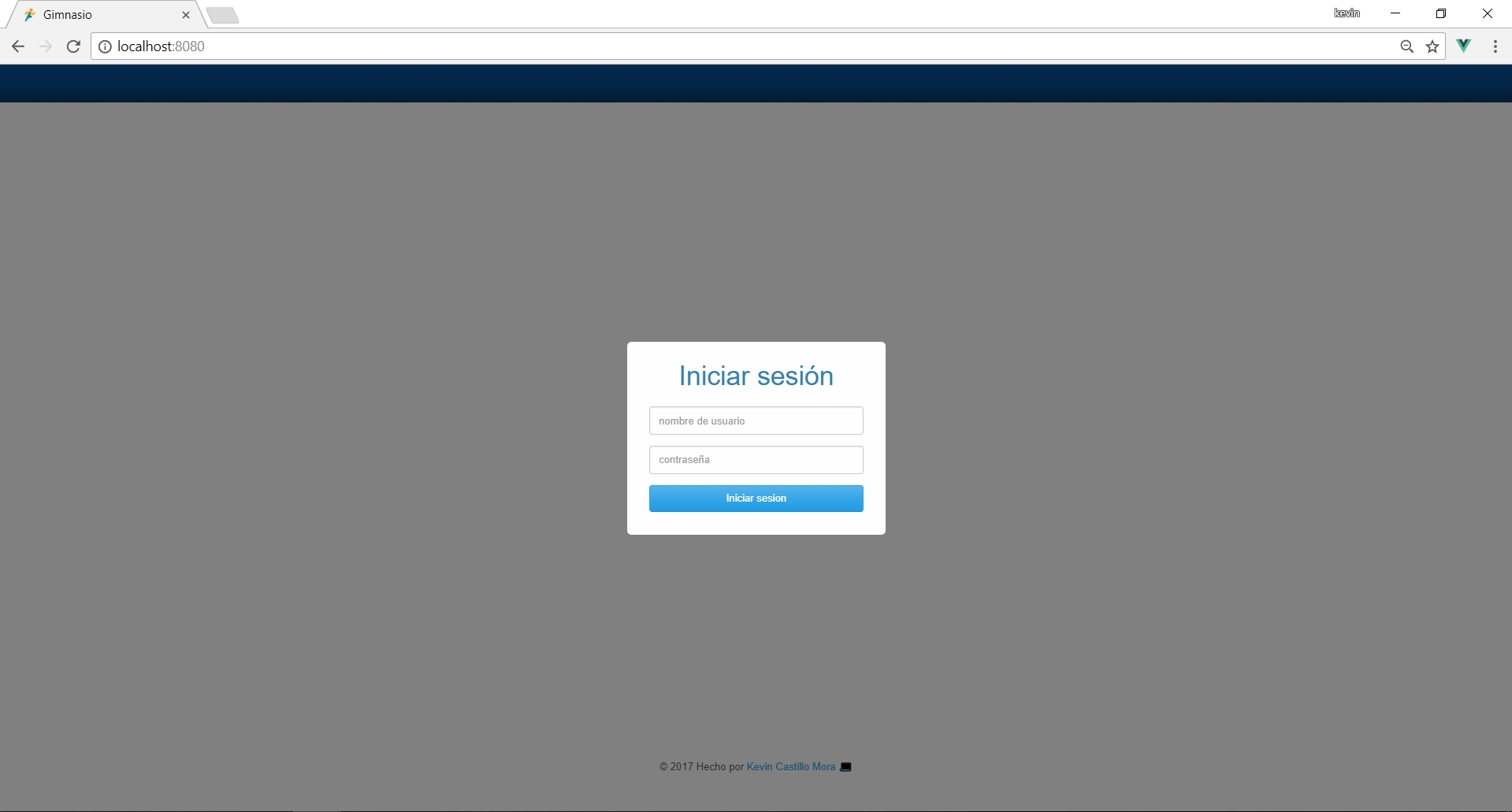
**Imagen 21: Listado de pagos por búsqueda de fechas.**

**En la pestaña de todos nos aparecerá todos los pagos realizados ordenados por fecha.**



**Imagen 22: Listado del historial de todos los pagos.**

**La última funcionalidad es para cerrar sesión debemos de darle click al icono en la esquina superior derecha y esto nos devolverá de nuevo a la página de inicio de sesión.**

 **Imagen 23: Cerrar sesión.**

**MANTENIMIENTOS: Uno por Uno\*\*\***

**REPORTES: Uno por Uno\*\*\***

**INSTALACIÓN Y PUESTA EN MARCHA: Manual de Instalación\*\*\*\***

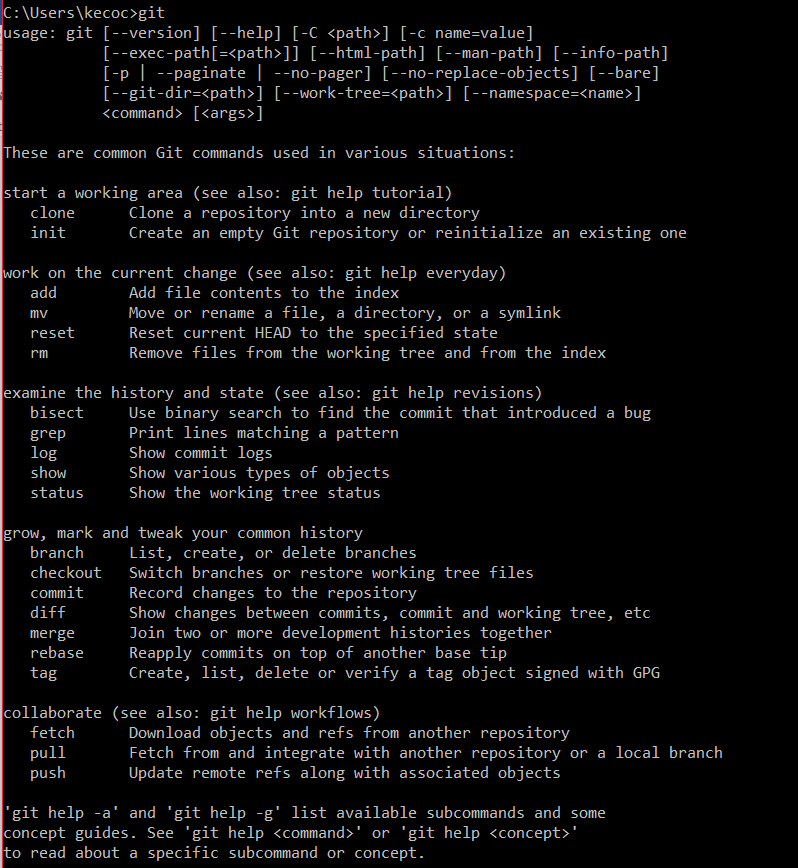
**El proyecto está alojado en este repositorio:**

[**https://github.com/kecoco16/Gimnasio-Vue**](https://github.com/kecoco16/Gimnasio-Vue)

**Para inicializarlo lo primero que debes hacer es tener Git instalado en tu pc:**

[**https://git-scm.com/book/en/v2/Getting-Started-Installing-Git**](https://git-scm.com/book/en/v2/Getting-Started-Installing-Git)

**Luego debes de abrir una terminal para comprobar que instalaste Git puedes ejecutar el comando git y te deberían de salir todas las opciones que provee git.**

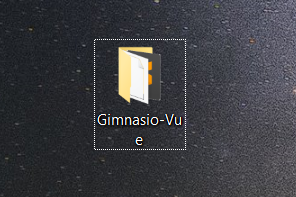


**Imagen 24: Comandos de git.**

**Una vez comprobado tener git instalado nos deberemos de dirigir a la ruta donde queramos posicionar el proyecto por ejemplo el escritorio y ejecutar el comando git clone (ruta repositorio)**

 **Imagen 25: Clonación del repositorio.**

**Esto deberá crear una carpeta en la ruta especificada igual a esta:**



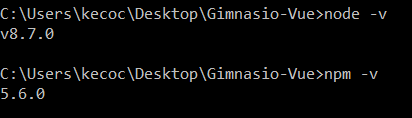
**Imagen 26: Carpeta del proyecto.**

**El siguiente paso es instalar node y npm**

[**https://nodejs.org/es/download/**](https://nodejs.org/es/download/)

**Para comprobar haber instalados bien podemos utilizar el comando:**

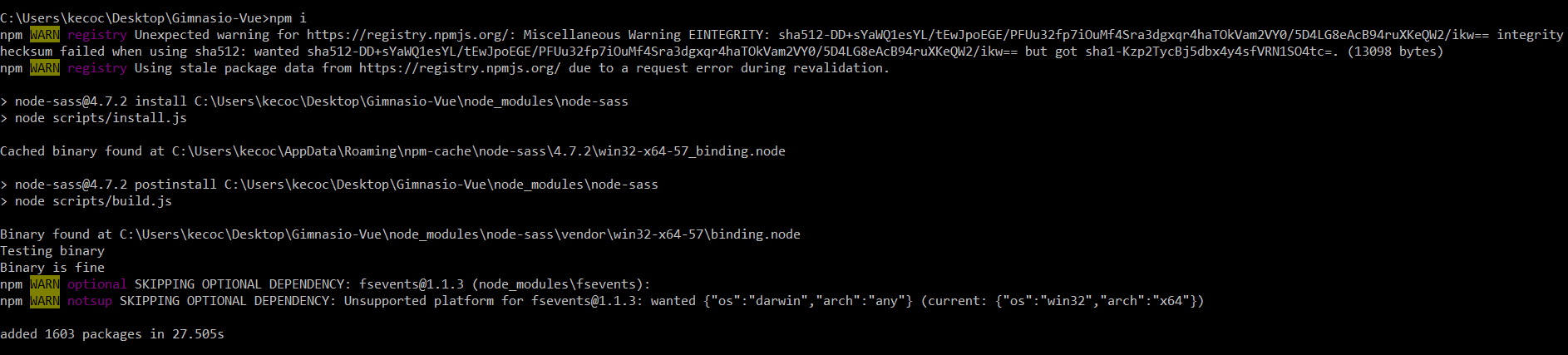
**(nombre del programa) -v**



**Imagen 27: Versiones de node y npm.**

**Luego de esto debes dirigirse dentro de la carpeta del proyecto y ejecutar el comando:**

**npm i**

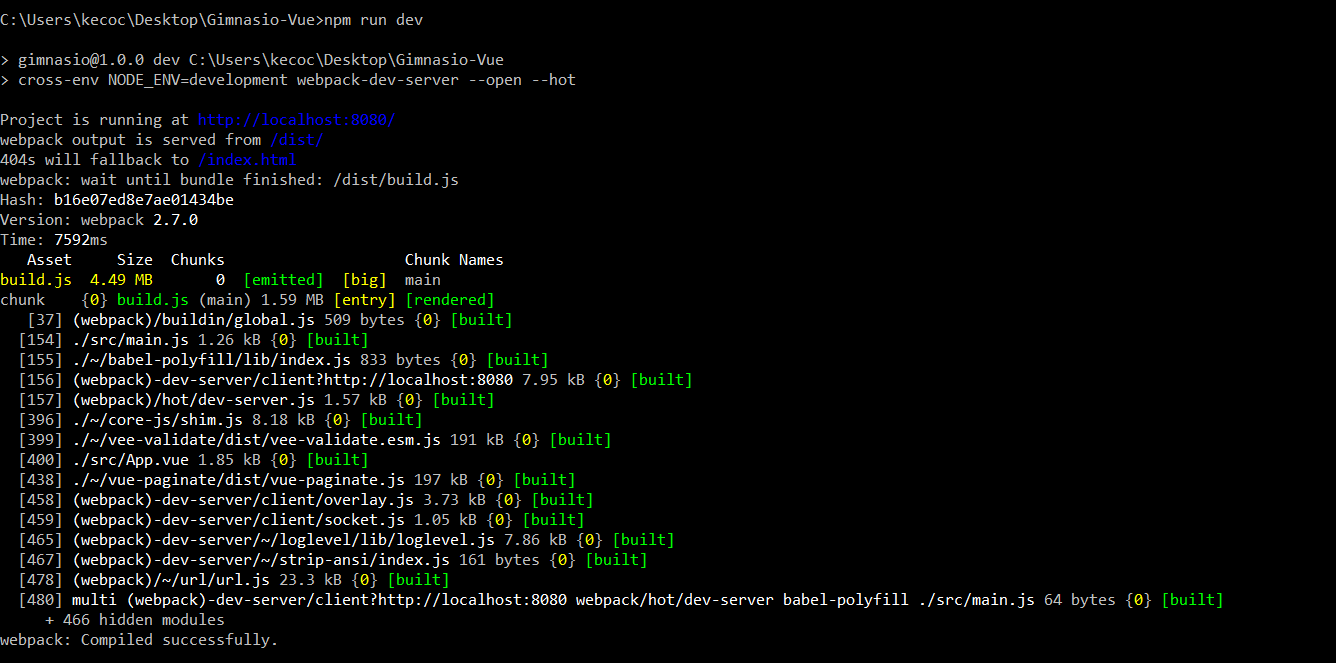


**Imagen 28: Instalando las dependencias del proyecto.**

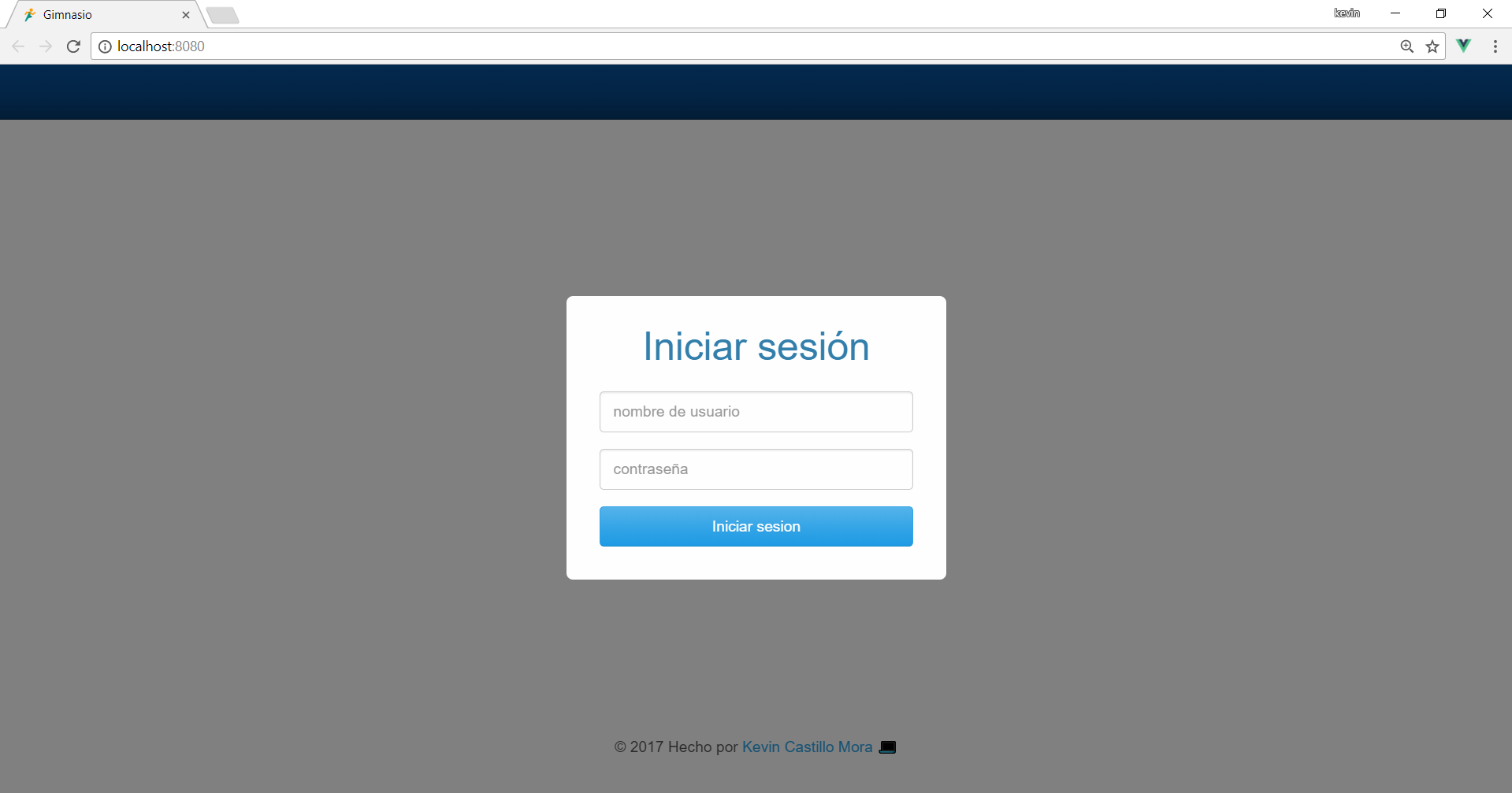
**El último paso es ejecutar el comando:**

**npm run dev**

**Y esto nos levantara un servidor en el localhost y nos abrirá el proyecto en el navegador.**



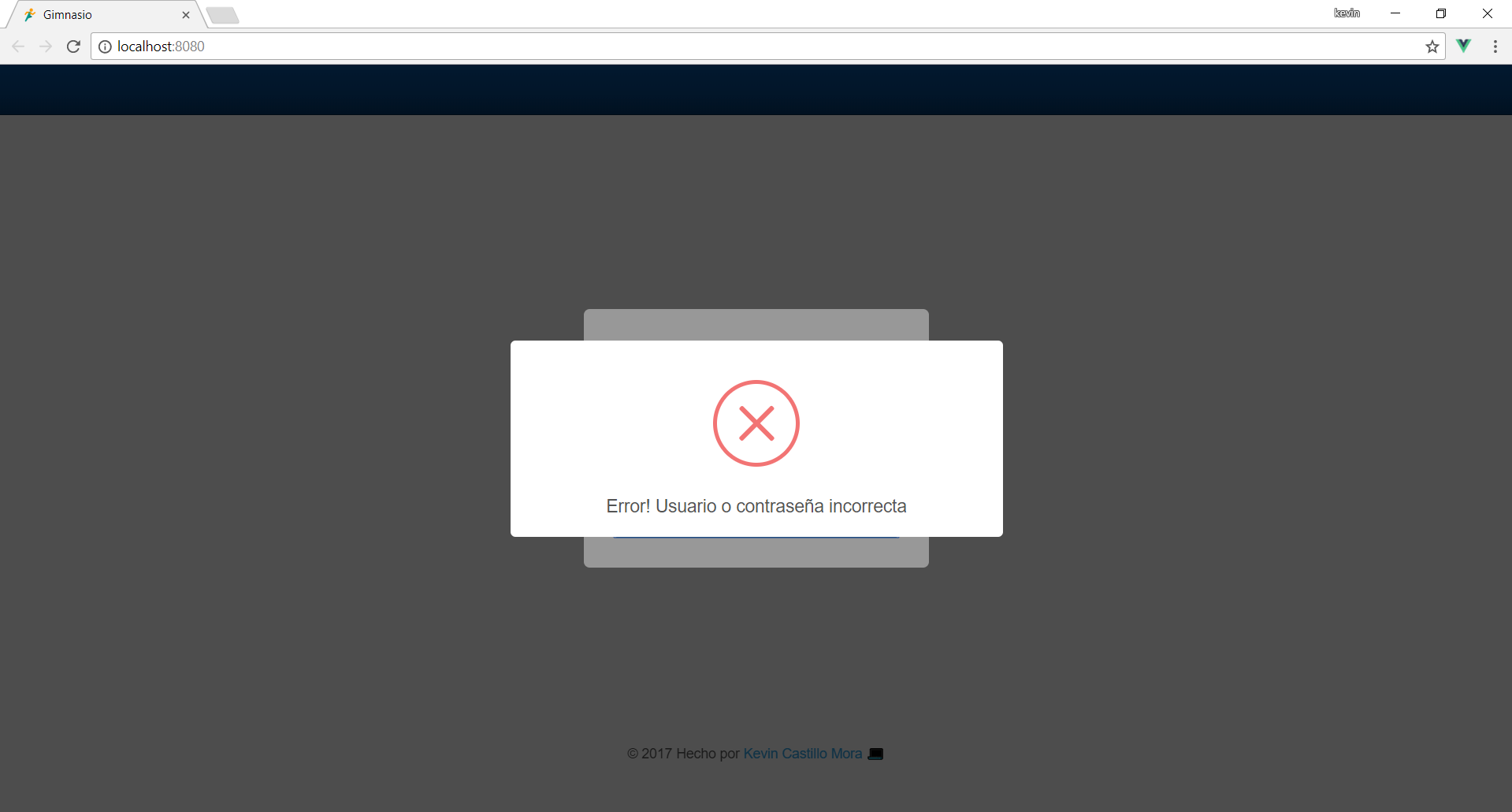
**Imagen 29: Corriendo el proyecto.**



**Imagen 30: Pantalla de inicio del proyecto.**

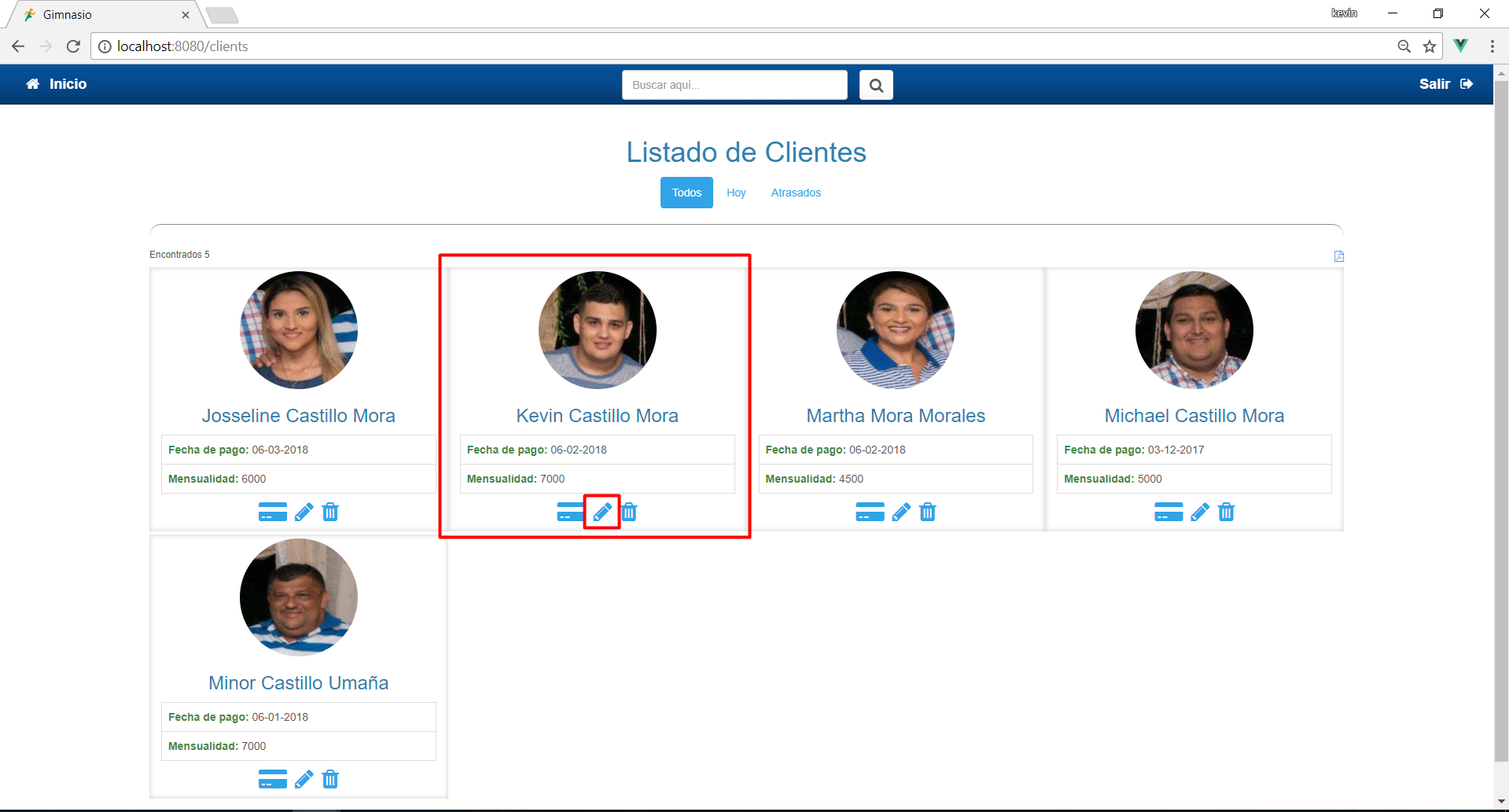
**PRUEBAS OPERATIVAS:**

**Se probo satisfactoriamente que la única forma de pasar la pantalla del inicio de sesión es con un usuario y una contraseña correcta.**

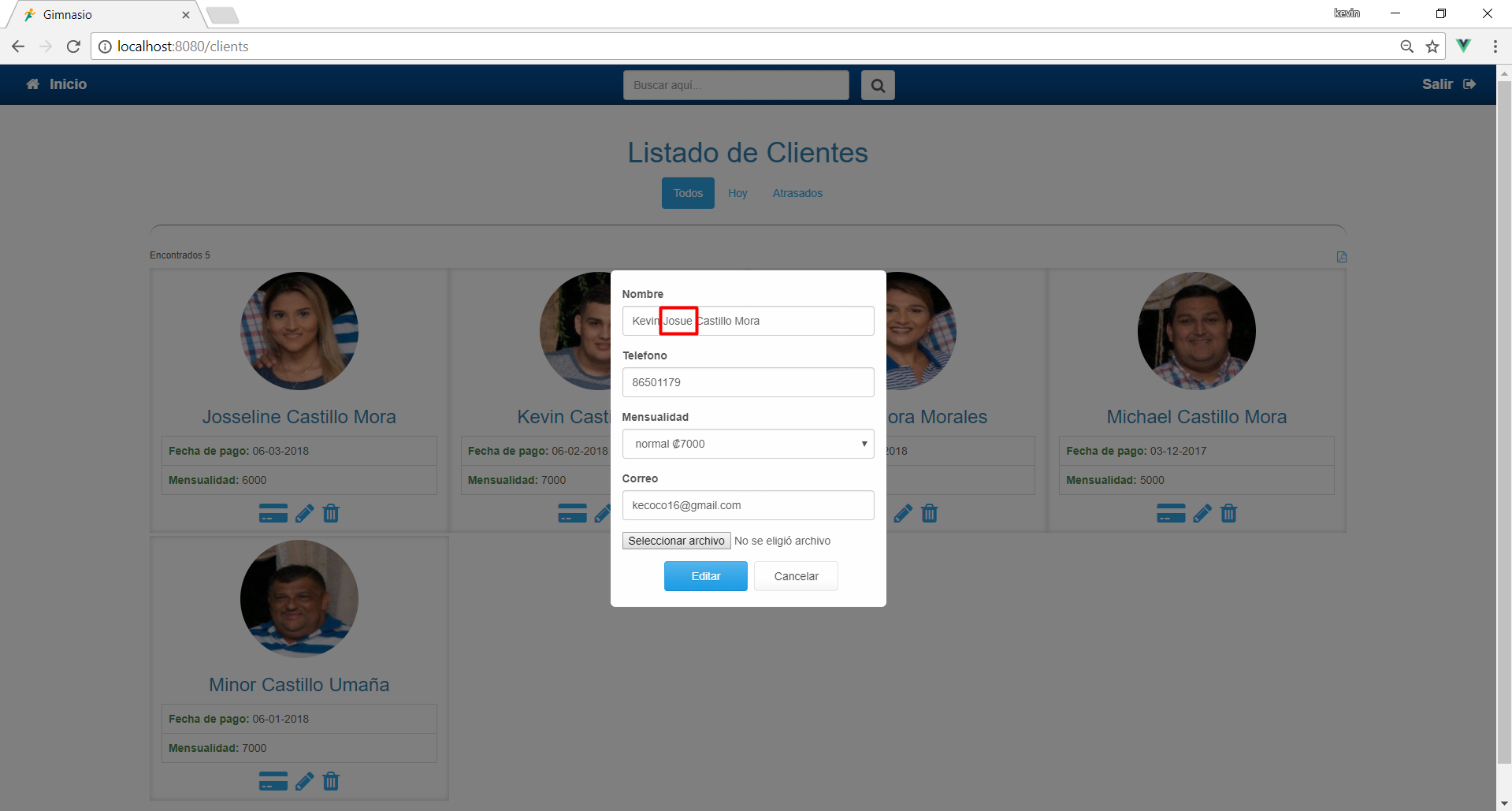


**Imagen 31: Mensaje de error al iniciar sesión.**

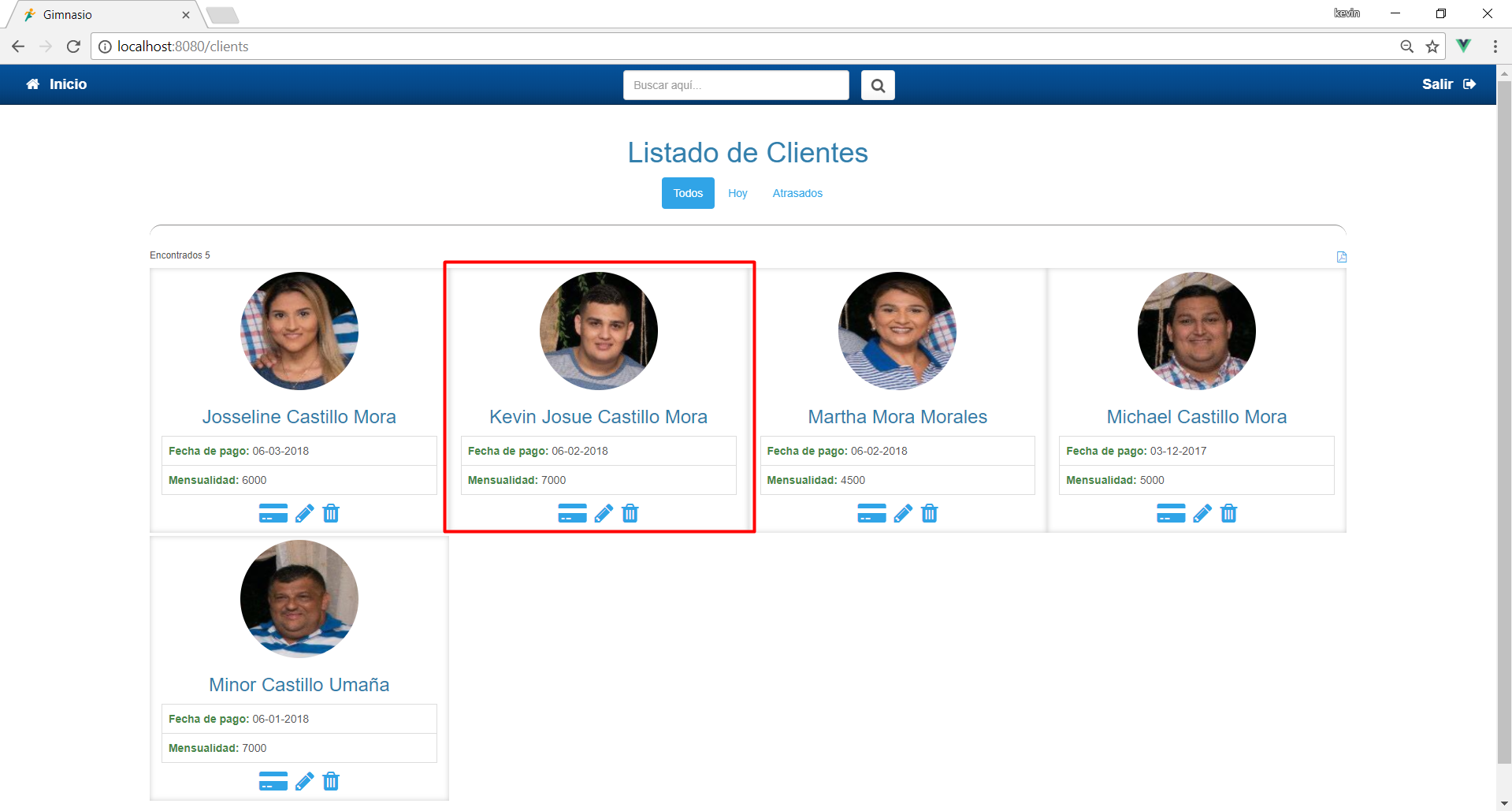
**Se probó que se puede editar un cliente satisfactoriamente y que lo nuevos valores se actualizan.**



**Imagen 32: Mostrando el botón para editar un cliente.**

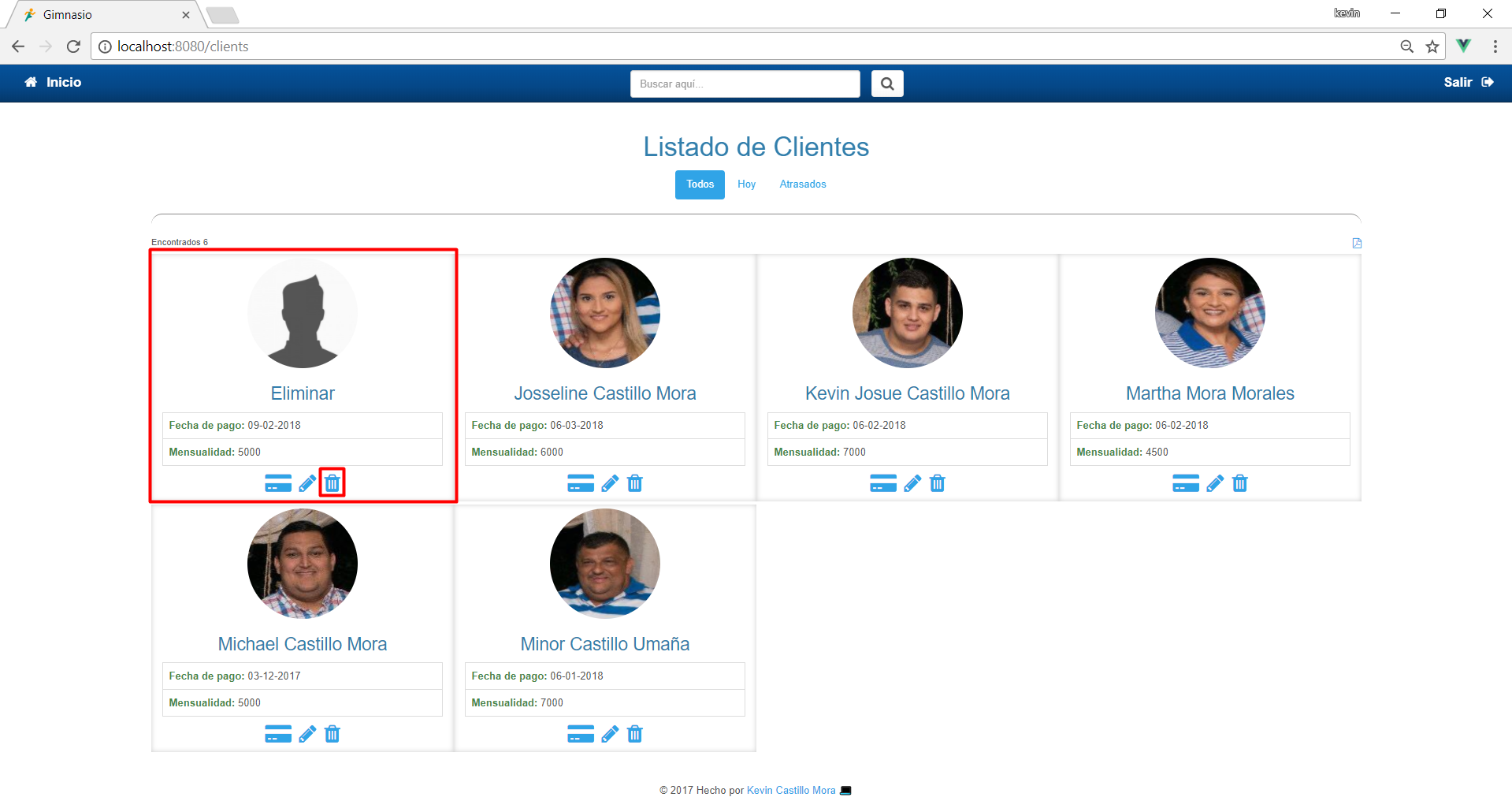


**Imagen 33: Modal con las opciones para editar del cliente.**



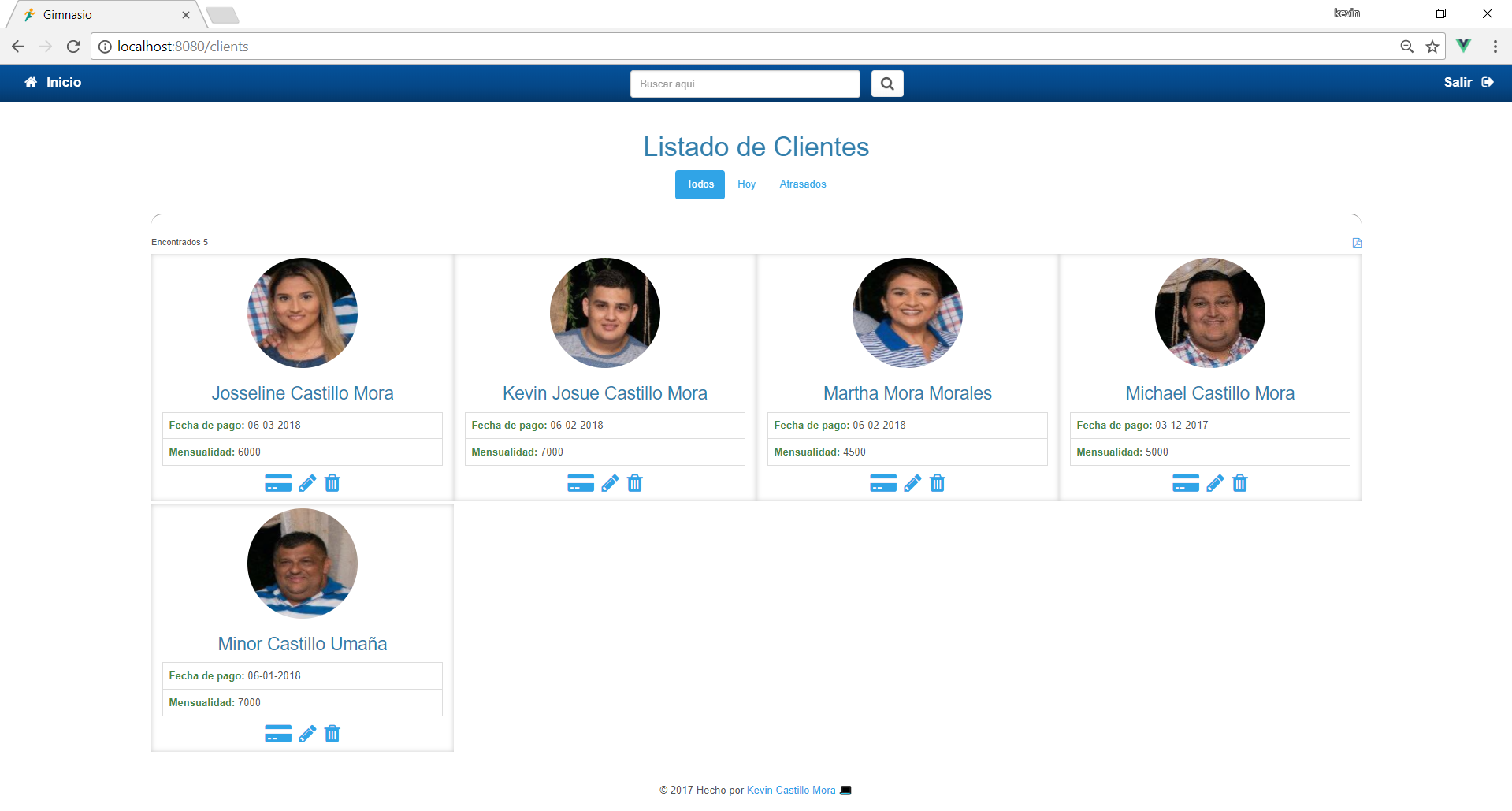
**Imagen 34: Comprobando que los cambios se hallan realizado.**

**Se probo que se puede eliminar un cliente satisfactoriamente.**



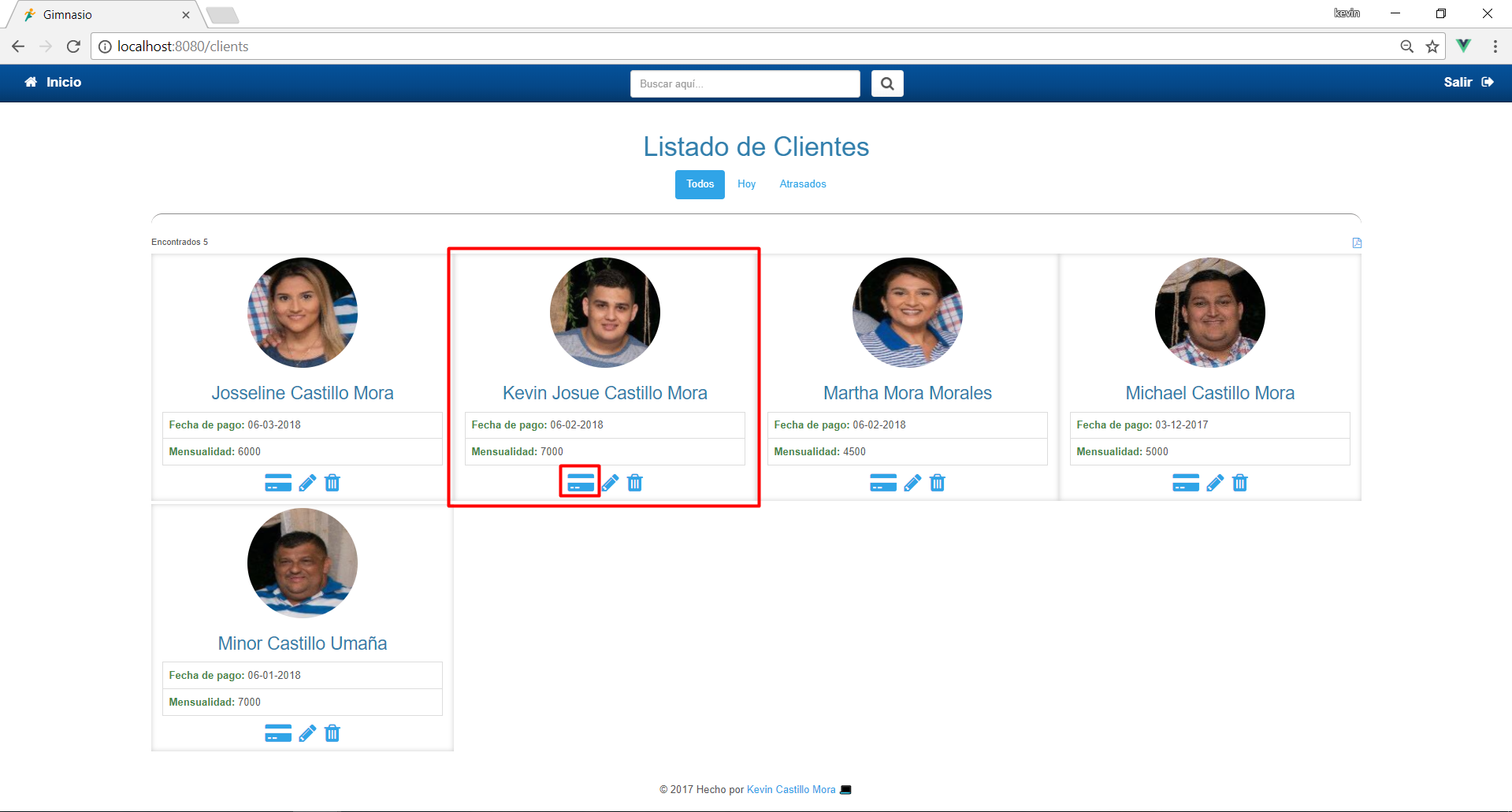
**Imagen 35: Mostrando el botón para eliminar un cliente.**

 **Imagen 36: Modal con mensaje de confirmación para eliminar al cliente.**

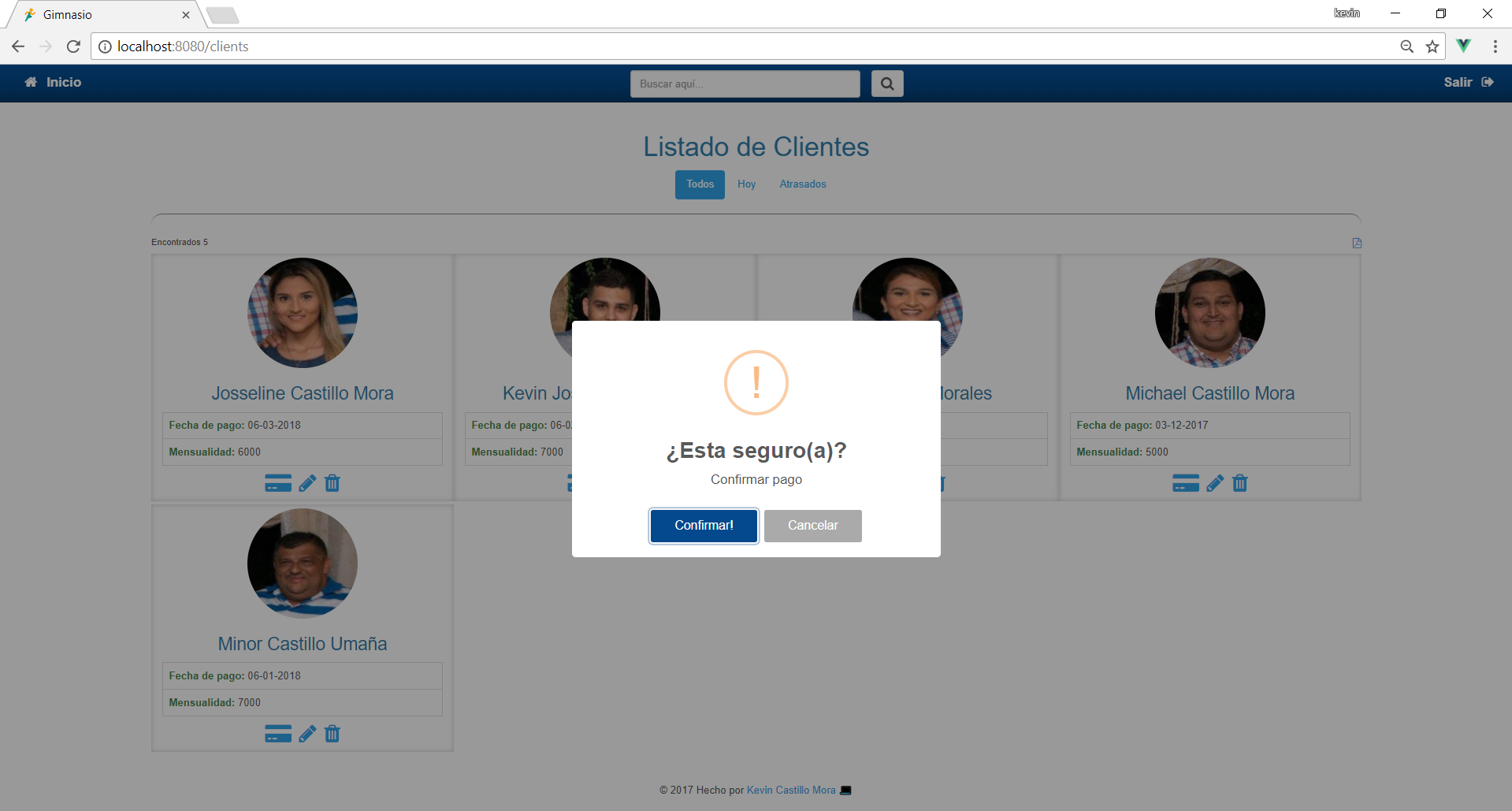


**Imagen 37: Comprobando que el cliente se haya eliminado.**

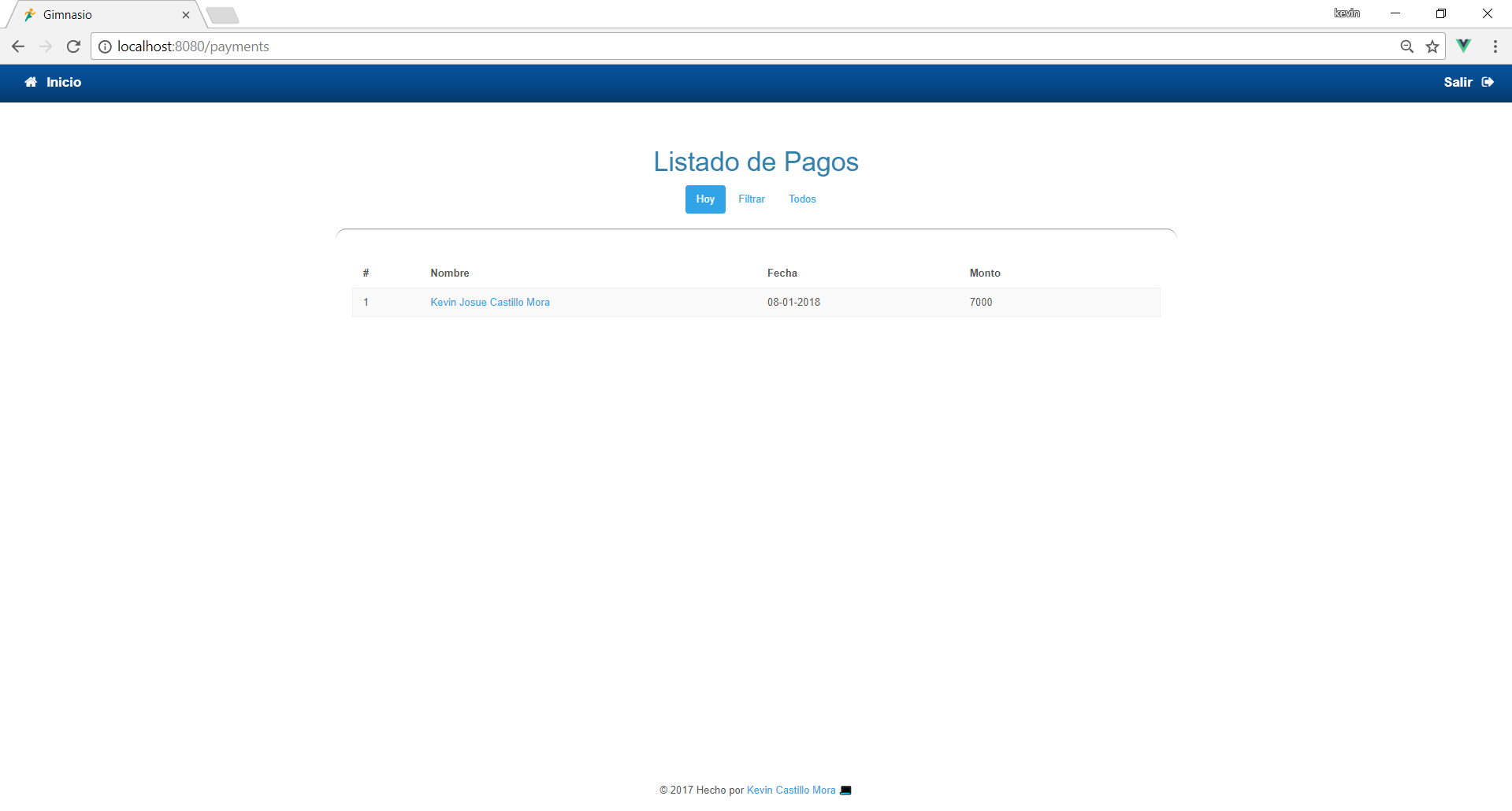
**Se probo que al darle al realizar el pago de un cliente la fecha de pago se actualiza y aparece en la etiqueta de pagos realizados hoy.**



**Imagen 38: Mostrando el botón para realizar el pago de la mensualidad del cliente.**

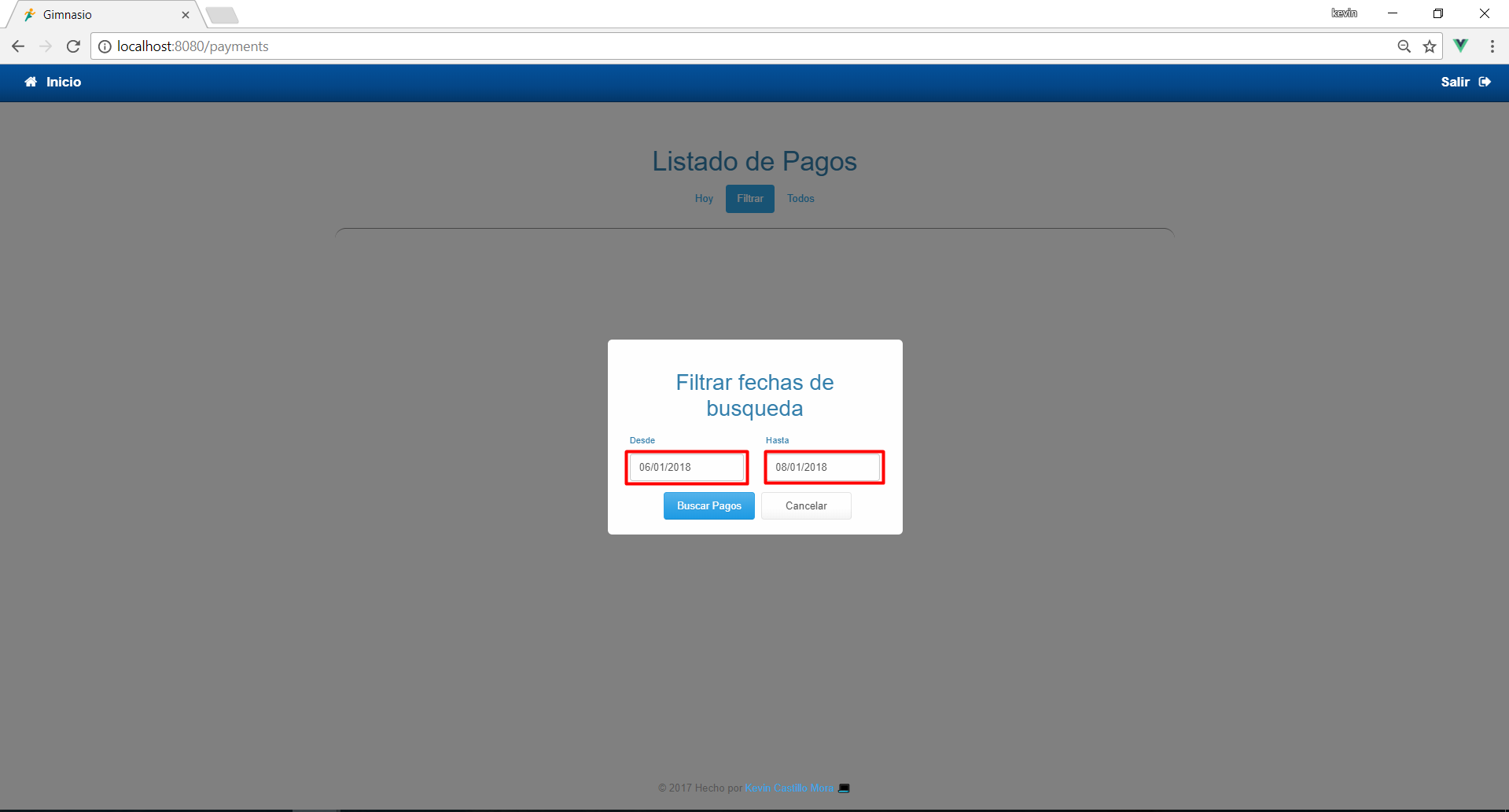


**Imagen 39: Modal con el mensaje de confirmación de realizar el pago.**

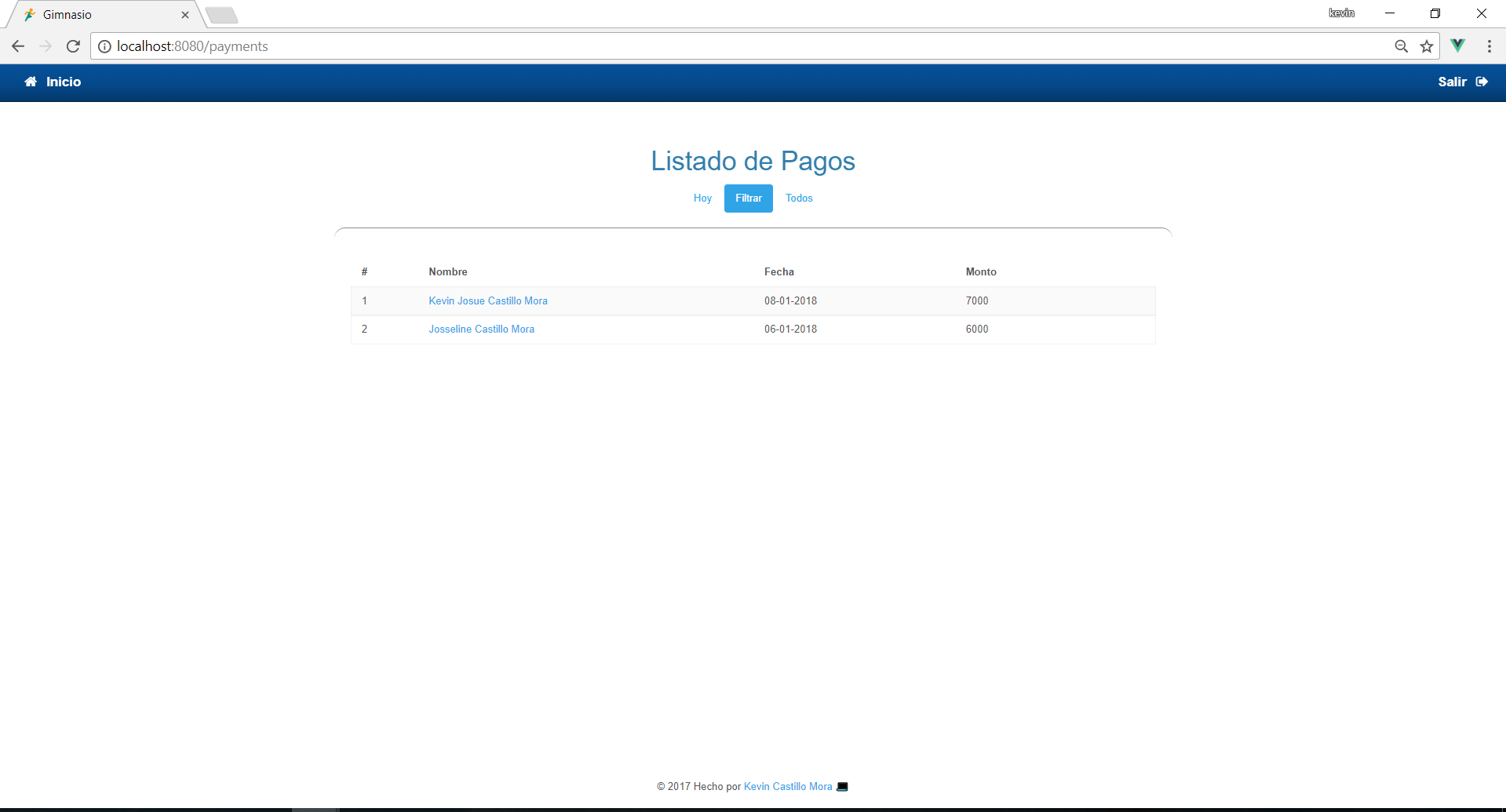


**Imagen 40: Comprobando que el pago se realizó con éxito.**

**Se probo que el filtro del registro de pagos funciona correctamente en este momento solo existen 2 pagos entre esas fechas.**

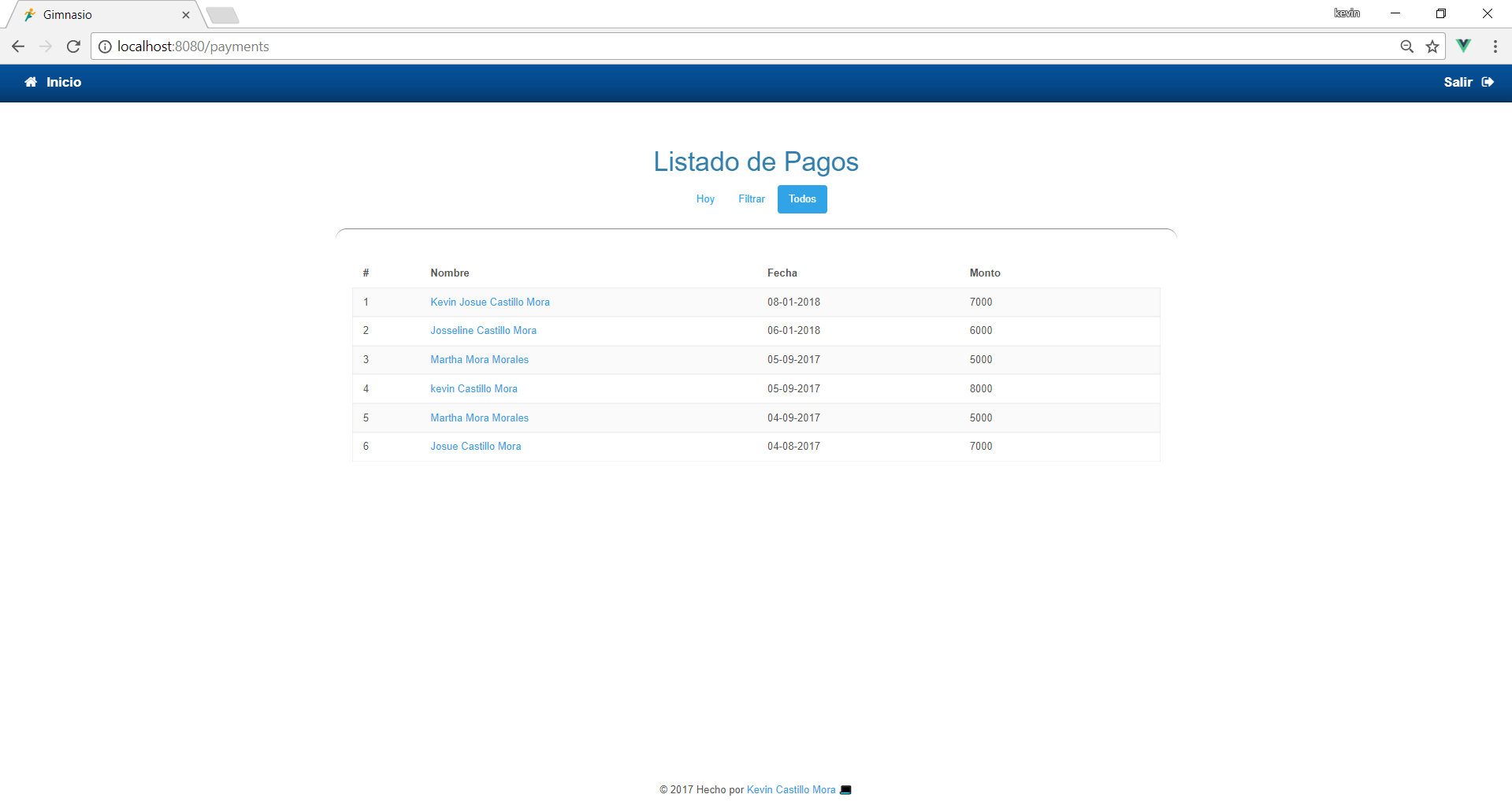


**Imagen 41: Modal para hacer el filtro de la búsqueda por fechas.**



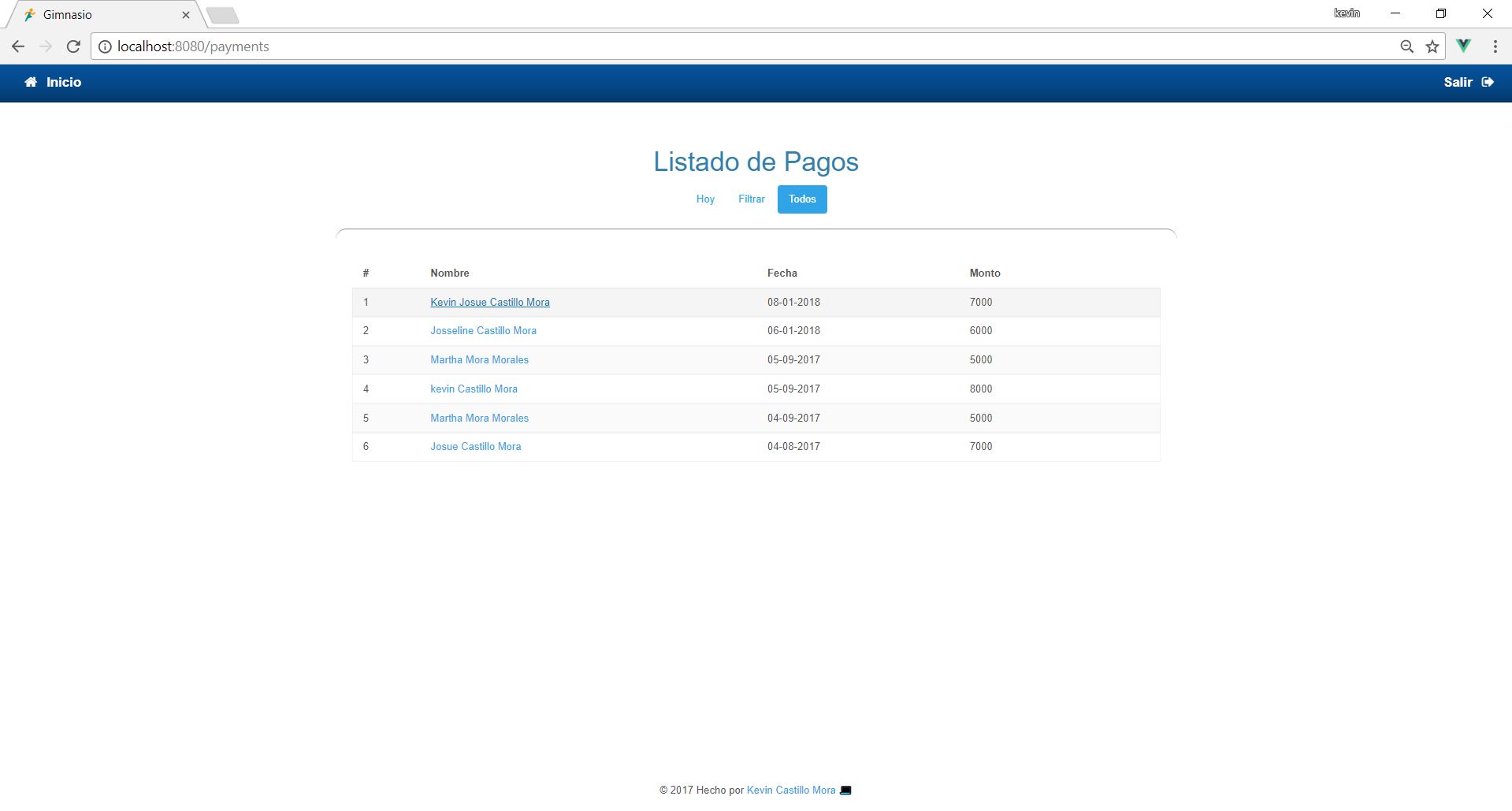
**Imagen 42: Comprobando que la búsqueda por fechas funciona correctamente.**

**Se puede apreciar en la etiqueta de Todos que es correcto.**

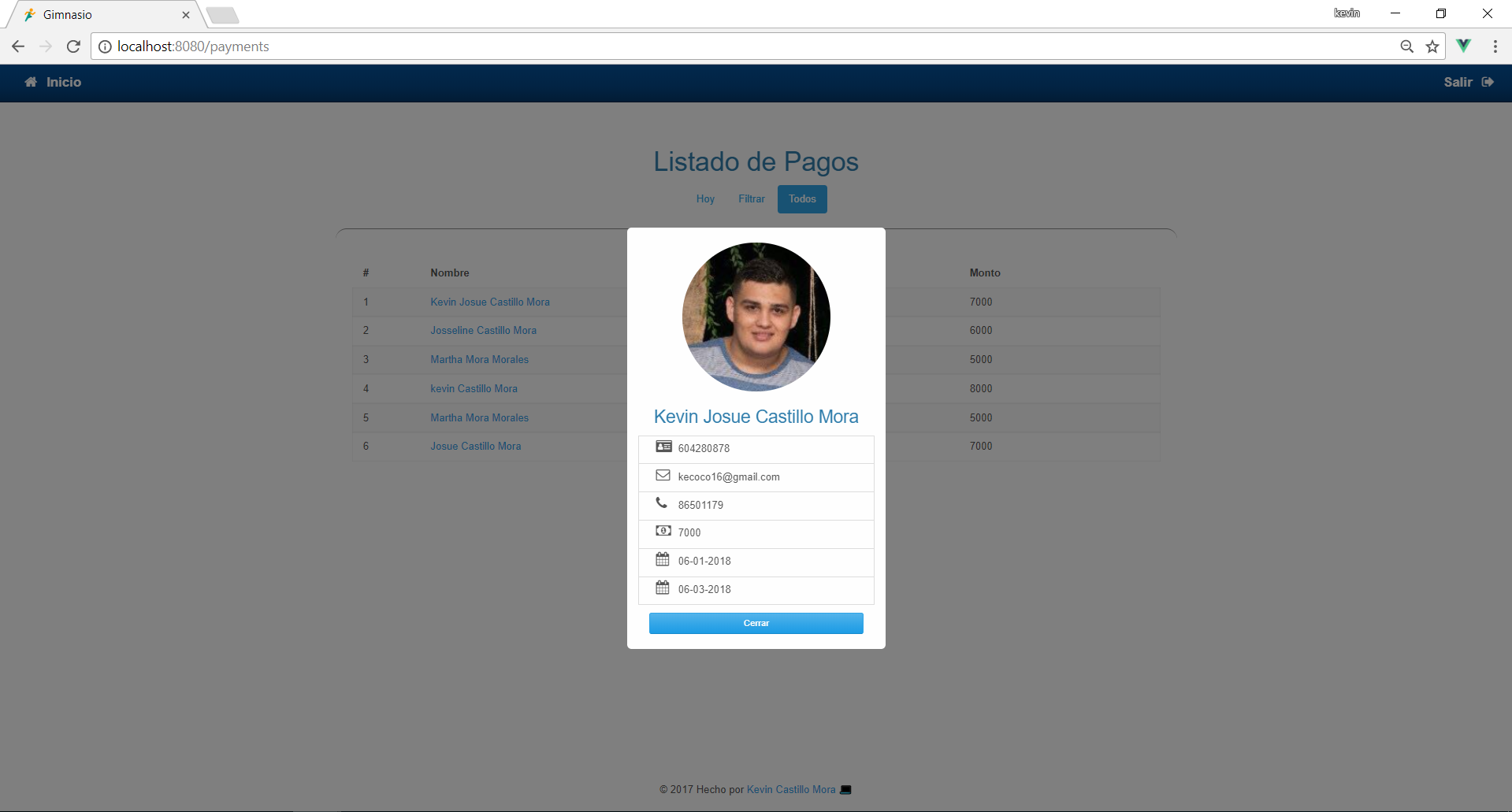


**Imagen 43: Mostrando la pestaña de todos los pagos.**

**Se probo que al darle click al nombre aparece el perfil del cliente seleccionado**

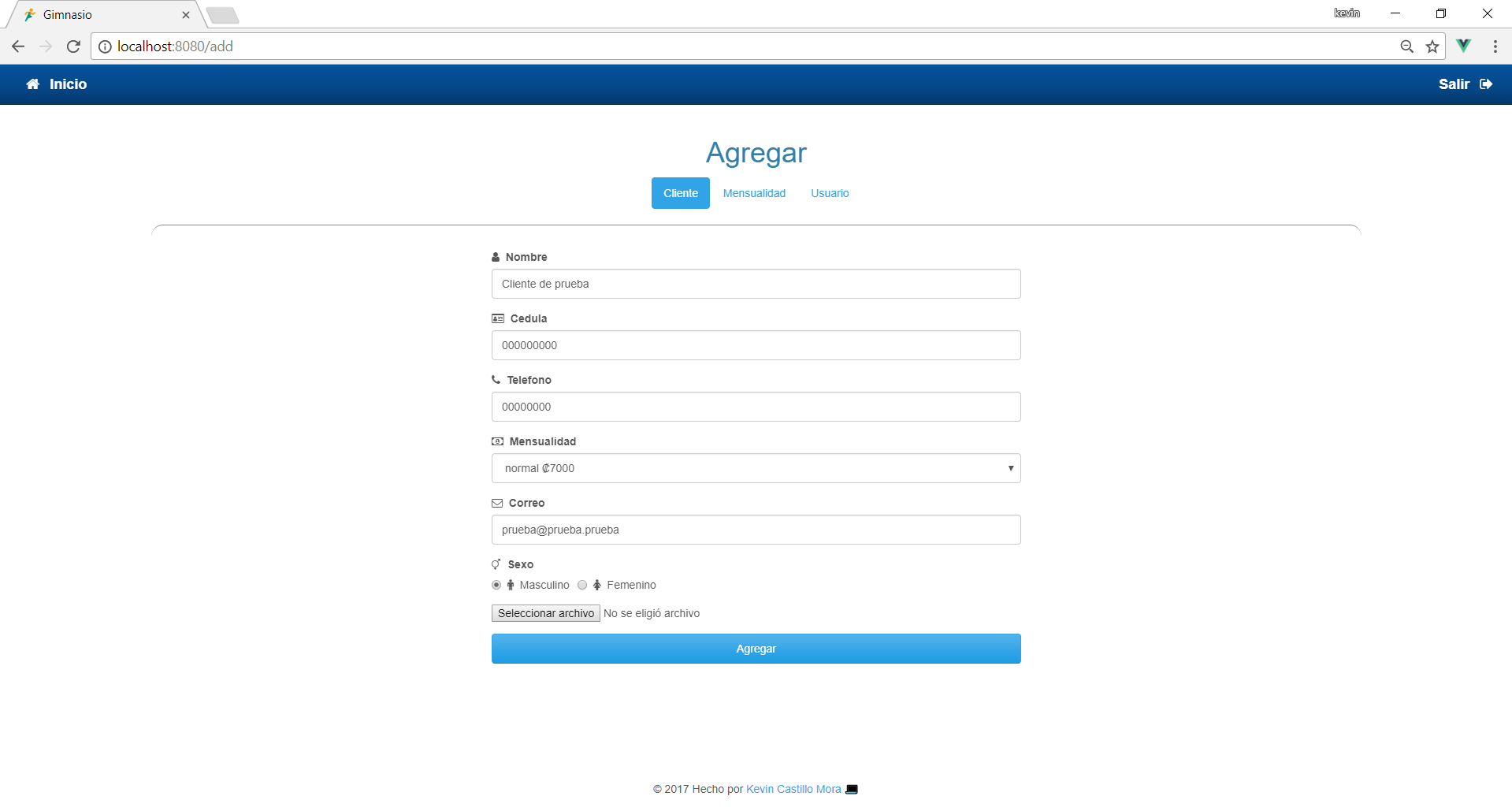


**Imagen 44: Mostrando que los nombres son clickeables para ver su información de cliente.**

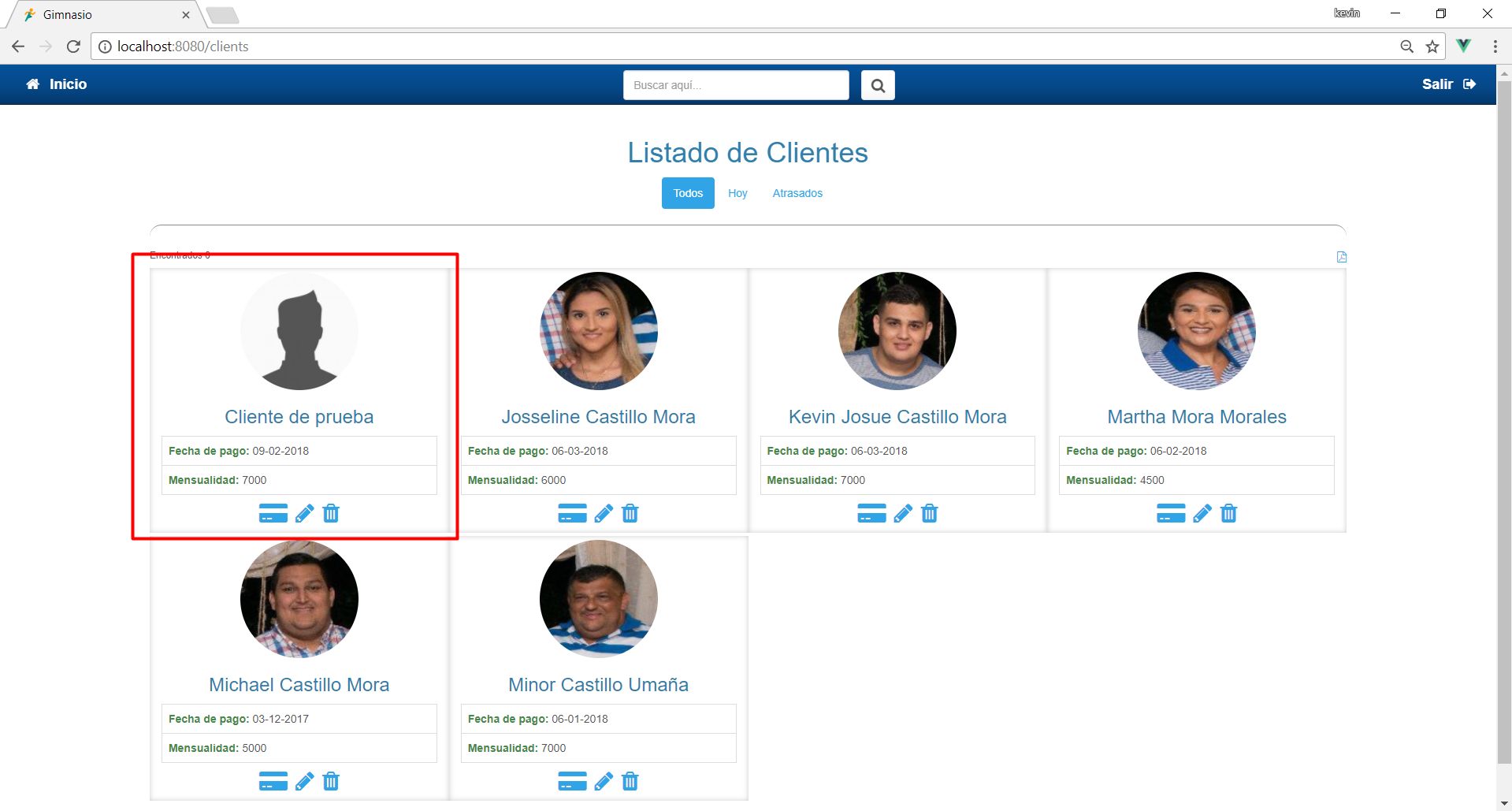


**Imagen 45: Mostrando el modal con la información del cliente.**

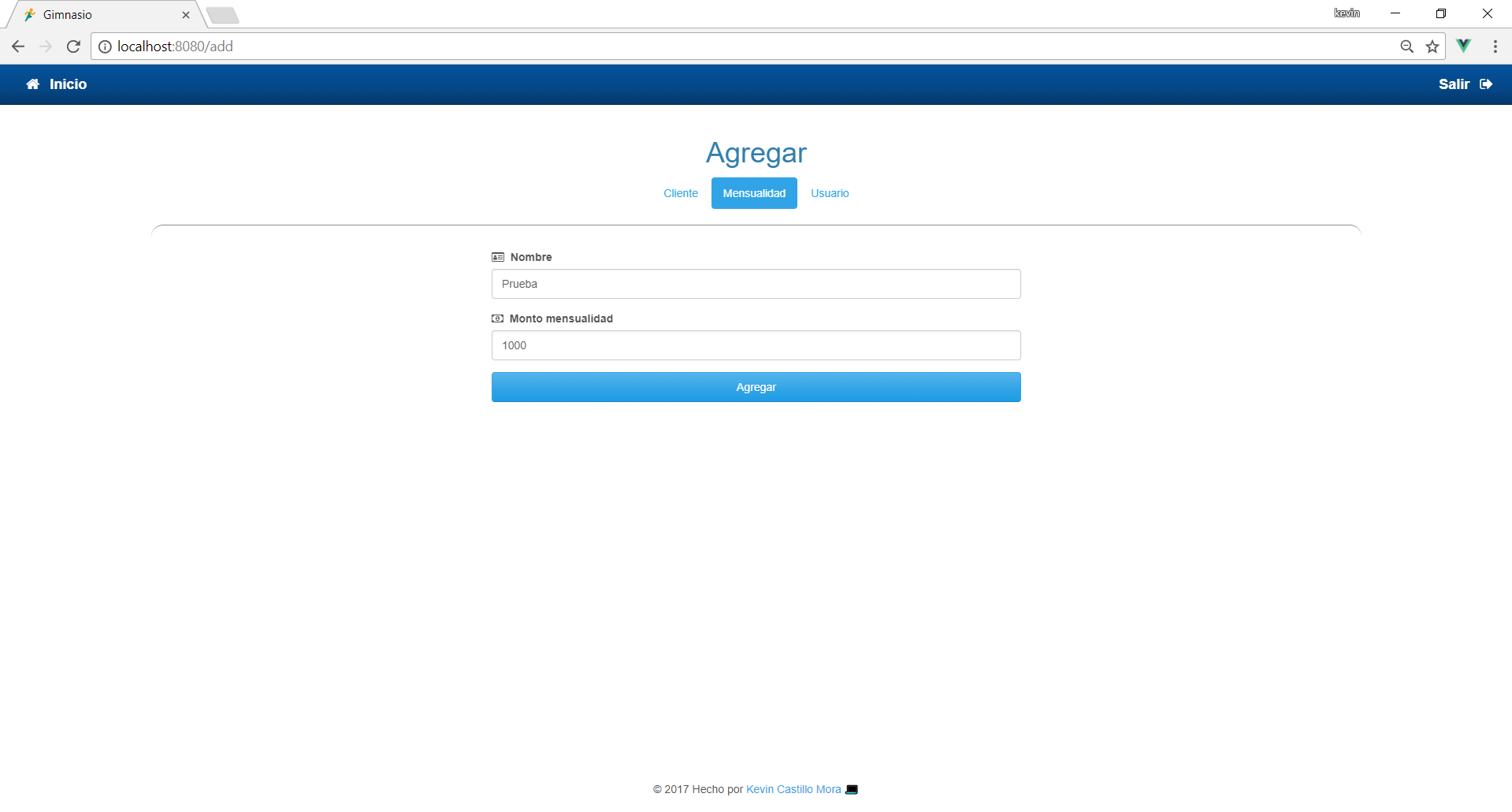
**Se probo que se puede añadir clientes satisfactoriamente**



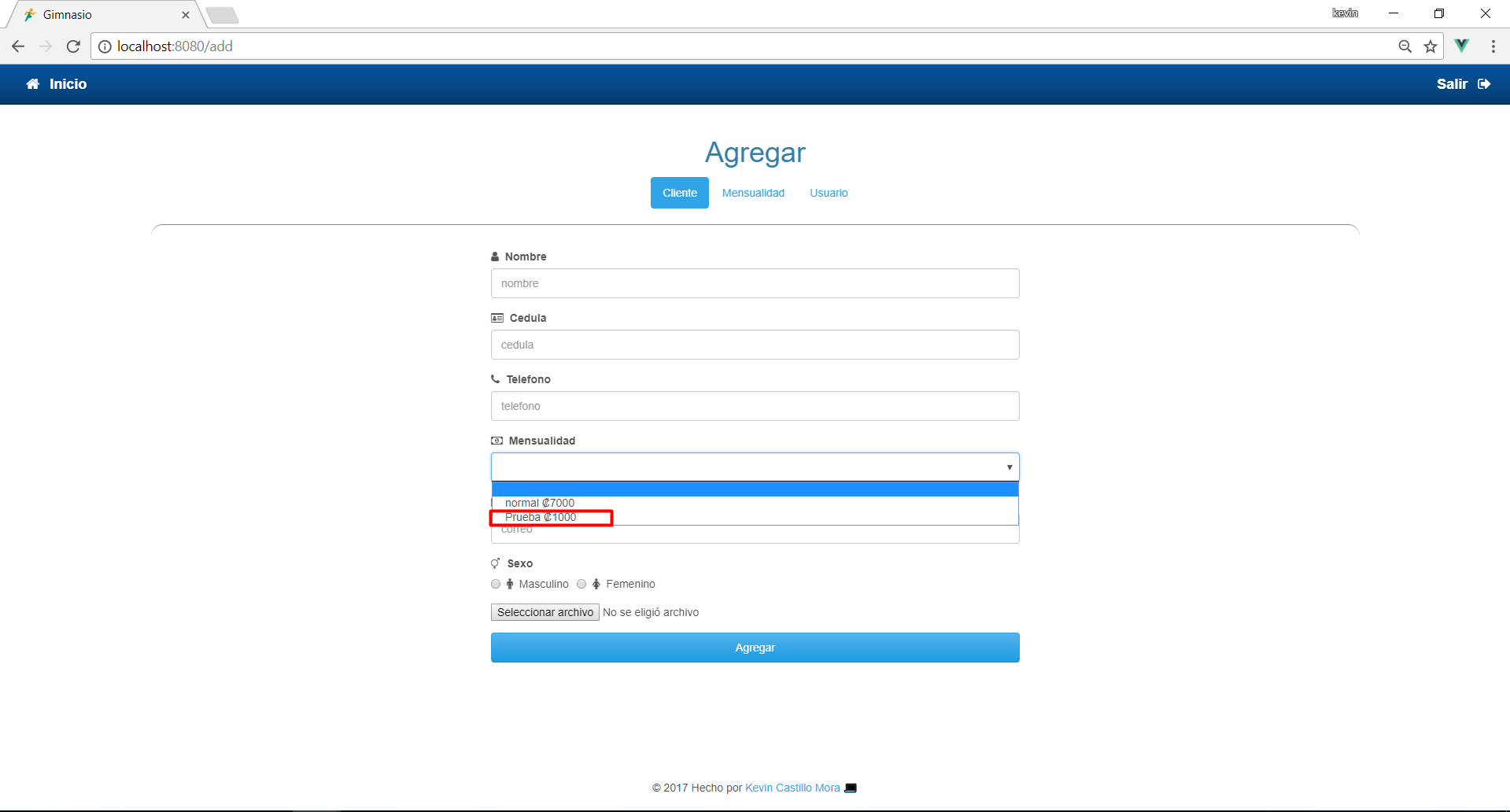
**Imagen 45: Formulario para añadir clientes.**



**Imagen 45: Comprobando que se agregó al cliente correctamente.**

**Se probo que se pueden agregar mensualidades satisfactoriamente.**

**Imagen 46: Formulario para agregar mensualidades.**



**Imagen 47: Comprobando que la mensualidad se agrego correctamente.**

**ANÁLISIS DE PRUEBAS: Resultados, trabas, eventualidades encontradas en el momento de la instalación y puesta en marcha que son importante tener en cuenta y anotar.**

**CARTA DE ACEPTACIÓN: Escaneo del documento de aceptación de conclusión del proyecto por parte del patrocinador.**

Capítulo V

CONCLUSIONES Y RECOMENDACIONES

**CAPITULO V**

**5. CONCLUSIONES DE LA PRÁCTICA EMPRESARIAL.**

Terminado el sistema de información para…. Y una vez configurado y probado, se determinan las siguientes conclusiones y recomendaciones:

**5.1** **CONCLUSIONES:**

1. Actualmente el departamento de metrología lleva un sistema de control de los equipos, solicitudes, órdenes de trabajo en forma manual donde en un cuaderno se apuntan las salidas y las entradas. También existe una serie de hojas electrónicas preestablecidas para el control, sin embargo las mismas no son utilizadas, tal vez lo anterior se debe a desconocimiento o falta de capacitación en Excel.

1. Las tareas que lleva a cabo este departamento son muchas, pero no se cuenta con una base de datos que almacene todo esta información, para la generación de reportes de dichas tareas.
2. Se determina que los equipos de cómputo con que cuenta el departamento de metrología, cuentan con la capacidad para administra un sistema de información gerencial, lo cual es muy importante porque permitió la factibilidad de que la práctica se realizara con un bajo costo.
3. Se procedió a Capacitar al personal encargado de utilizar el nuevo sistema de información con el objetivo de tener un uso adecuado del mismo.
   1. **RECOMENDACIONES DEL PROYECTO.**
4. Se recomienda que la Compañía Alimentos ProSalud S.A, elabore un catálogo con códigos de los diferentes equipos que se utilizan, lo cual seria de mucha importancia a la hora de que la compañía se quiera certificar bajo una nueva norma iso.
5. Es recomendable que la Compañía Alimentos ProSalud S.A. establezca cual va a ser la plataforma que ha seguir para la creación de futuros programas, con la finalidad de unificar criterios y de esta forma todos los programassean compatibles ypuedan integrar al software general de la empresa.
6. Es recomendable que cuando se necesite la creación de un programa y éste se asigne a un estudiante para ser elaborado en su práctica empresaria, se le proporcione al estudiante por escrito, los requerimientos y necesidades existentes, además debe indicársele sobre que leguaje que se va a programar, la base de datos a utilizar.

Capítulo VI

BIBLIOGRAFÍA Y ANEXOS

**CAPITULO VI**

* 1. **Bibliografía**

SERRA HESSELBACH [**ANALISIS DE REDES Y SISTEMAS DE COMUNICACIONES**](http://www.quedelibros.com/libro/22198/Analisis-De-Redes-Y-Sistemas-De-Comunicaciones.html)

SENN JAMES **ANÁLISIS Y DISEÑO DE SISTEMAS DE INFORMACIÓN.** 1990

. TANENBAUM ANDREWS S **REDES DE COMPUTADORAS, TERCERA EDICIÓ.**

WesleyAddison **SISTEMAS DE BASES DE DATOS, 2ª,** 2000

Nixdorf Heinz **HISTORIA DE LA COMPUTADORA**, 2000

### ASPECTO SITUACIONAL DE LA COMPAÑIA

### http://es.wikipedia.org/wiki/Instituto\_Costarricense\_de\_Electricidad

### http://www.grupoice.com/wps/portal/asi\_somos#.Ui9o9dI98hM

### 

### MISION Y VISION

### http://www.grupoice.com/wps/portal/asi\_somos#.Ui9o9dI98hM

### 

### RESEÑA HISTORICA

### http://www.grupoice.com/wps/portal/gice/acerca\_ice/acerca\_ice\_asi\_somos/acerca\_ice\_asi\_somos\_historia/!ut/p/c5/04\_SB8K8xLLM9MSSzPy8xBz9CP0os\_gQL0N\_D2cLEwN\_Vy8XA08zY09TUzNTi1BnI6B8JC55gwBjE5J0GwRYugLlXX0tg8xcDQwMiNJtgAM4GhDQ7eeRn5uqX5AbGhpR7qgIAPkP--s!/dl3/d3/L2dBISEvZ0FBIS9nQSEh/

### 

### META

### http://www.grupoice.com

### 

### INFORMATICA

### <http://es.wikipedia.org/wiki/Inform%C3%A1tica>

ANEXOS

**Cartas (Carta de aceptación de la persona o institución en la que se realiza el proyecto, indicando que el mismo concluyó de forma exitosa y aprobando el mismo)**

**Anexos Varios (Documentos, fotos, todo lo que consideren necesario agregar)**

Puntarenas, 21/04/14

**Sres.** Universidad Metropolitana Castro Carazo

Sede Puntarenas

Estimados señores, por medio de la presente me permito saludarlos y al mismo tiempo hago constar de forma escrita que el estudiante Kevin Josue Castillo Moraportador de la cédula 604280878 se ha presentado continuamente a nuestra institución con el fin de desarrollar su proyecto de graduación que lleva como título PROGRAMA PARA LA ADMINISTRACIÓN DEL GIMNASIO, el mismo ha sido instalado y probado de forma correcta, cumpliendo con los objetivos planteados.

Sin más que agregar por el momento se despide,

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Harold Mora y Cédula**

**Presidente**