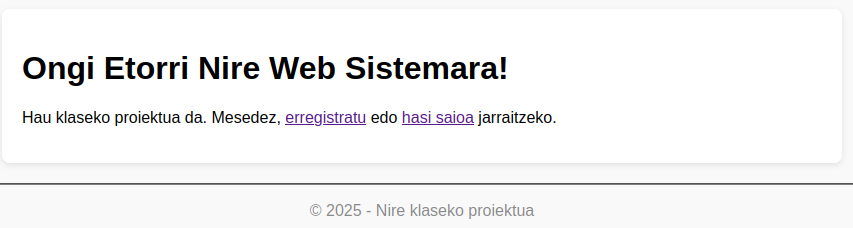
**Gure web-orriaren**

**dokumentazioa**



Egileak: Lier Laiseka, Asier Fernández, Hodei Castro eta Andoni Ortiz de Zarate.

## Aurkibidea

[Aurkibidea 1](#_Toc1291453827)

[Azalpena 2](#_Toc1209463046)

[Kodearen egitura eta Azalpena: 3](#_Toc296465891)

[config.php 4](#_Toc351031202)

[HTML txantiloiak 4](#_Toc871367989)

[.htaccess 5](#_Toc1705763271)

[main.js 5](#_Toc1490889736)

[Erreferentziak 5](#_Toc1241787185)

[Funtzionalitateak 6](#_Toc1423750501)

[PHP 6](#_Toc1548517823)

[JavaScript 7](#_Toc1538128067)

[Funtzionalitateen lorpena 7](#_Toc594320292)

[Bazkideen Kudeaketa (Erabiltzaileak) 8](#_Toc547637674)

[Elementuen Kudeaketa (Liburuak) 8](#_Toc1408444339)

[Saio hasi edo erregistratu 9](#_Toc1942785650)

[Profilaren datuak aldatu 10](#_Toc1807201230)

[Elementuak ikusi eta gehitu 11](#_Toc1044132142)

[Karritora elementuak gehitu eta ikusi 13](#_Toc1602422604)

[Elementuak aldatu eta ezabatu 14](#_Toc683441452)

## Azalpena

Sistema honen helburu nagusia erabiltzaileek liburutegiaren katalogoa kontsultatzea, liburu pertsonalak gehitzea/aldtzea/ezabatzea eta erabiltzaileak gorde nahi dituen liburuen zerrenda bat kudeatzea da.

## Kodearen egitura eta Azalpena:

Gure proiektuaren kodea ez da fitxategi bakar batean (index.php) nahasten; bi bloketan banatu egin dugu.

* **"Irudia" (Bistak):** Erabiltzaileak ikusten duena dira. Fitxategi hauek (adibidez, register.php , login.php , items.php ) HTML egitura nagusia dute. Era berean WebSistemak dokumentuaren azaltzen den bezala, <form> etiketak erabili ditugu datuak biltzeko, action atributuarekin logikari deitzeko eta method="POST" erabiliz datuak bidaltzeko.
* **"Logika" (Kontrolatzaileak):** Erabiltzaileak ez ditu ikusten. Fitxategi hauek (adibidez, procesar\_register.php , procesar\_login.php ) PHP purua dira. Formularioetatik bidalitako datuak $\_POST superglobalaren bidez jasotzen dituzte . Fitxategi hauen ardura da datu horiek balioztatzea eta datu-basearekin komunikatzea.

Funtzionamendua, teoriako dokumentuan azaldutakoa bezalakoa da. Bezeroak eskaera bat egiten dio zerbitzariari (Apache) , honek PHP exekutatzen du , PHPk Datu Basearekin hitz egiten du , eta, azkenik, HTML hutsa itzultzen dio erabiltzaileari.

### config.php

Fitxategi honek proiektu osorako balio duten bi elementu kritiko kudeatzen ditu:

* **Datu Base Konexioa:**Hemen sortzen da new mysqli(...) bidezko konexio bakarra ($conn).
* **Saio Hasiera:** Fitxategi honen hasieran session\_start() deia egiten da, . Horrela, saioa ($\_SESSION) sistema osoan erabilgarri dagoela ziurtatzen dugu.

### HTML txantiloiak

* Webgunearen egitura bisual komuna (menuak, CSS estiloak, JavaScript fitxategien karga) header.php fitxategian zentralizatuta dago. Orriaren amaiera footer.php fitxategian dago.
* "Irudia" erakusten duten fitxategi guztiek include('header.php'); eta include('footer.php'); deiak erabiltzen dituzte, kodea errepikatu behar ez izateko.

### .htaccess

Proiektuaren egiturak URL lagungarriak(/login, /items) erabiltzen ditu, **Apache zerbitzariaren** gaitasun bat erabiliz. app/.htaccess fitxategiak "itzultzaile" lana egiten du.

### main.js

main.js fitxategiak gure proiektuaren **bezero-aldeko (JavaScript) logika guztia zentralizatzen du**. Horrela kodea ez errepikatuz, header.php fitxategian behin bakarrik kargatzen baita eta horrela orrialde guztietan eskuragarri dago.

Fitxategiaren ardura nagusiak hauek dira:

* **Formularioen Balioztatzea:** Erabiltzaileak formulario bat bidali baino *lehen* datuak zuzenak direla egiaztatzen du.
* **Funtzio Nagusiak:** Honako funtzioak ditu: validateRegisterForm(), validateLoginForm(), validateModifyForm() eta validateItemForm(). Funtzio hauek formularioen onsubmit gertaerarekin lotuta daude.
* **Balioztatze Espezifikoak:** Erabiltzaileak sartutako datuak egiaztatzen ditu (NAN formatua, telefono luzera, email formatua, etab.
* **Erabiltzailearen Interakzioa:** Daturen bat okerra bada, script-ak formularioaren bidalketa **gelditzen du** (return false; itzuliz ) eta erabiltzaileari berehalako errore-mezuak erakusten dizkio (.innerHTML erabiliz), JuananPereira teoria dokumentua azaltzen den DOM manipulazioa erabiliz.

### Erreferentziak

#### Funtzionalitateak

Proiektuaren bi modulu nagusiek (Erabiltzaileak eta Liburuak) **CRUD (Create, Read, Update, Delete)** eredu estandarra jarraitzen dute. Eredu hau JuananPereiraHTML5.pdf dokumentuaren orrietan azaltzen da.

* **Create (Sortu):** Erabiltzaileak (procesar\_register.php ) eta elementuak (procesar\_add\_item.php ) sortzea INSERT bidez.
  + JuananPereiraHTML5.pdf "Datuak txertatzen".
* **Read (Irakurri):** Zerrendak (items.php ) eta elementu zehatzak (show\_item.php ) erakustea SELECT bidez.
  + JuananPereiraHTML5.pdf "Dokumentuak bilatzen" .
* **Update (Eguneratu):** Erabiltzaileen (procesar\_modify\_user.php ) eta elementuen (procesar\_modify\_item.php ) datuak UPDATE bidez aldatzea.
* **Delete (Ezabatu):** Elementuak (procesar\_delete\_item.php ) DELETE bidez ezabatzea.

#### PHP

* **Datu Base Konexioa (MySQLi):** config.php fitxategiak konexioa sortzen du.
  + WebSistemak.pdf dokumentuak zehazki metodo hau erakusten du: \$conexion = new mysqli("NombreHost", "Usuario", "Contraseña", "nombreBD"); .
* **Sententzia Prestatuak (Prepared Statements):** Gure procesar\_... fitxategi guztiek (procesar\_login.php , procesar\_add\_item.php , etab.) $conn->prepare(...), $stmt->bind\_param(...) eta $stmt->execute() erabiltzen dute.
  + Hau ez da emandako PDFetan zehazten (bertan $conexion->query() sinpleagoa agertzen da), baina **PHP-ren dokumentazio ofizialean (PHP.net)** oinarritutako segurtasun-praktika estandarra da SQL Injekzioa saihesteko.
* **Saio Kudeaketa (Sessions):** Erabiltzaileen identifikazioa kudeatzeko erabilia.
  + WebSistemak.pdf dokumentuak azaltzen du nola erabili session\_start() eta nola sortu eta egiaztatu $\_SESSION aldagaiak , gure config.php eta procesar\_login.php fitxategiek egiten duten bezala.
* **Fitxategien Igoera:** procesar\_add\_item.php eta procesar\_modify\_item.php fitxategiek argazkiak kudeatzen dituzte.
  + WebSistemak.pdf-k HTML zatia (<input type="file">) azaltzen du . Fitxategiak prozesatzeko PHP logika ($\_FILES superglobala, move\_uploaded\_file()) **PHP-ren dokumentazio ofizialean (PHP.net).**

#### JavaScript

* **DOM Manipulazioa eta Gertaerak:** main.js fitxategiak formularioen balioztatzea kudeatzen du.
  + **Erreferentzia Teorikoa:** JuananPereiraHTML5.pdf dokumentuak azaltzen du nola erabili document.getElementById() eta innerHTML errore-mezuak erakusteko. WebSistemak.pdf dokumentuak onsubmit gertaera erabiltzea azaltzen du.
* **NAN Balioztatze Funtzioa:** validarNIF() (PHPn) eta validarNIF\_JS() (JSn) funtzioek algoritmo espezifiko bat jarraitzen dute.
  + **NAN/NIF letrak kalkulatzeko algoritmo estandarrean (Mod 23)** oinarritzen da.

### Funtzionalitateen lorpena

#### **Bazkideen Kudeaketa (Erabiltzaileak)**

1. **Erabiltzaileen Erregistroa eta Balioztatze Konplexua:**
   1. **Helburua:** NAN, telefono, data eta email formatu espezifikoak balioztatzea.
   2. **Nola Lortu Da:** Bi balioztatze-geruza ezarri dira:
      1. **Bezeroaren Aldea (JavaScript):** main.js fitxategian, validateRegisterForm() eta validarNIF\_JS() funtzioek (onsubmit gertaerari lotuta ) erabiltzaileari berehalako feedbacka ematen diote. Honek WebSistemak eta JuananPereiraHTML5 dokumentuetakoko HTML5 balioztatze kontzeptuak aplikatzen ditu.
      2. **Zerbitzariaren Aldea (PHP):** Segurtasuna bermatzeko, procesar\_register.php fitxategiak funciones.php fitxategiko (balioztatuErabiltzaileDatuak, validarNIF) funtzioei deitzen die , datuak datu-basean sartu aurretik berriro egiaztatzeko.
2. **Erabiltzaileen Identifikazioa:**
   1. **Helburua:** Erabiltzailea izen eta pasahitz bidez identifikatzea.
   2. **Nola Lortu Da:** login.php formularioak datuak procesar\_login.php fitxategira bidaltzen ditu. Fitxategi honek SELECT kontsulta bat egiten du emaila aurkitzeko, eta password\_verify() funtzioa erabiltzen du datu-baseko *hash*-arekin bat datorren egiaztatzeko. Arrakasta izanez gero, Saioak ($\_SESSION) erabiltzen dira erabiltzailea sisteman identifikatuta mantentzeko ($\_SESSION['user\_id'] = ...) .
3. **Erabiltzailearen Datuen Aldaketa:**
   1. **Helburua:** Saioa hasi duen erabiltzaileak bere datuak aldatu ahal izatea, datuak berriro konprobatuz.
   2. **Nola Lortu Da:** modify\_user.php orriak saioan ($\_SESSION) gordetako IDa erabiltzen du datu-baseari SELECT kontsulta bat egiteko eta formularioa erabiltzailearen datuekin betetzeko . Datuak bidaltzean, procesar\_modify\_user.php fitxategiak erregistroan erabilitako balioztatze-funtzio berberak (balioztatuErabiltzaileDatuak) berrerabiltzen ditu, datuak datu-basean UPDATE komandoarekin eguneratu aurretik.

#### **Elementuen Kudeaketa (Liburuak)**

1. **Elementuak Gehitzeko Aukera (5 eremu):**
   1. **Nola :** add\_item.php formularioak 5 eremuak (izena, deskribapena, prezioa, stocka, kategoria) eta argazkia (portada\_fitxategia) jasotzen ditu. Datuak procesar\_add\_item.php fitxategira bidaltzen dira. Fitxategi honek INSERT SQL(CREATE) komando bat exekutatzen du .
2. **Elementuen Zerrenda (2 eremu):**
   1. **Nola:** items.php fitxategiak READ eragiketa bat egiten du (SELECT id, izena, kategoria... FROM elementuak) . Ondoren, while begizta bat erabiltzen du zerrenda sortzeko,eta argazkiaren ondoan izena eta kategoria erakusten ditu.
3. **Elementuen Datuak Aldatzeko Aukera (5 eremuak ikusi):**
   1. **Nola:** items.php zerrendako "Aldatu" estekak elementuaren IDa bidaltzen du URL bidez .

modify\_item.php orriak ID hori ($\_GET['item']) jasotzen du, SELECT kontsulta bat egiten du elementu horren 5 eremuak lortzeko, eta formularioa datu horiekin betetzen du . Aldaketak gordetzean, procesar\_modify\_item.php fitxategiak UPDATE eragiketa exekutatzen du .

1. **Elementuak Ezabatzeko Aukera (Konfirmazioarekin):**
   1. **Nola :** items.php zerrendako "Ezabatu" estekak IDa confirm\_delete\_item.php orrira bidaltzen du . Orri honek elementuaren izena erakusten du eta ekintza konfirmatzeko eskatzen dio erabiltzaileari . Bertan, id="item\_delete\_submit" botoia dago. Botoi horrek procesar\_delete\_item.php fitxategira deitzen du, zeinak DELETE eragiketa exekutatzen duen .

## Datu-basea Inportatu (Behin Bakarrik)

Sistema lehen aldiz abiaraztean (edo ./mysql karpeta ezabatzen duzun bakoitzean), datu-basea hutsik egongo da. Beharrezko egitura eta datuak kargatzeko:

1. Ireki **phpMyAdmin** zure nabigatzailean: <http://localhost:8890>
2. Hasi saioa (Zerbitzaria: db, Erabiltzailea: admin, Pasahitza: test).
3. Hautatu database datu-basea ezkerreko menuan.
4. Joan **"Inportatu" (Import)** fitxara.
5. Sakatu "Aukeratu fitxategia" (Choose File) eta hautatu proiektu honetako **database.sql** fitxategia.
6. Sakatu **"Jarraitu" (Go)** botoia.

Saio hasi edo erregistratu

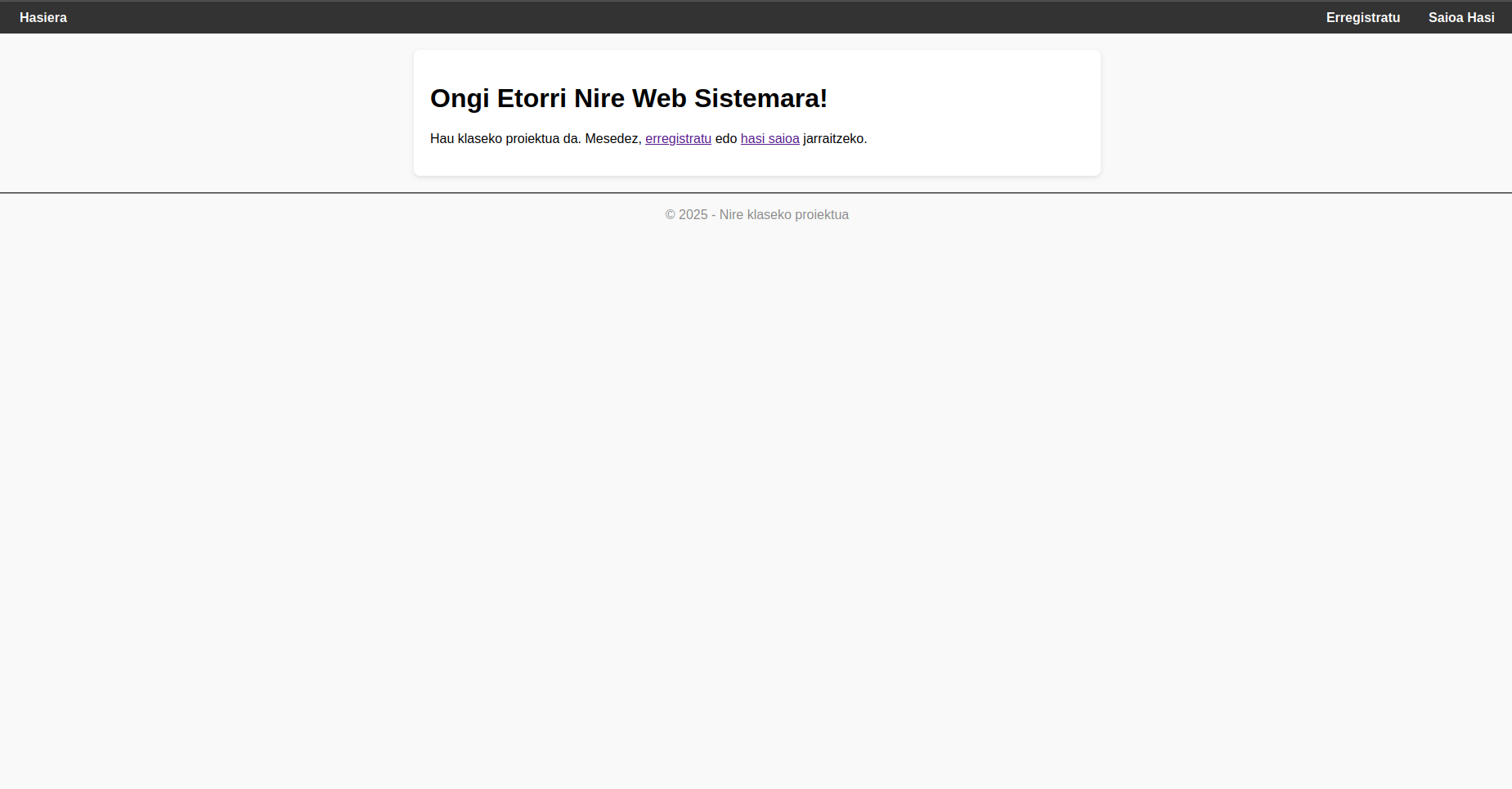
**1. argazkian** ikusten den moduan, gure web-orrira ongi etorria emantez zaigu, erregistratzeko edo saioa hasteko posibilitateak izanik.

Erregistratzen bagara, **2. argazkiko** pantaila agertuko zaigu, non gure datuak sartu behar ditugu, agertzen diren formatuak jarraituz.

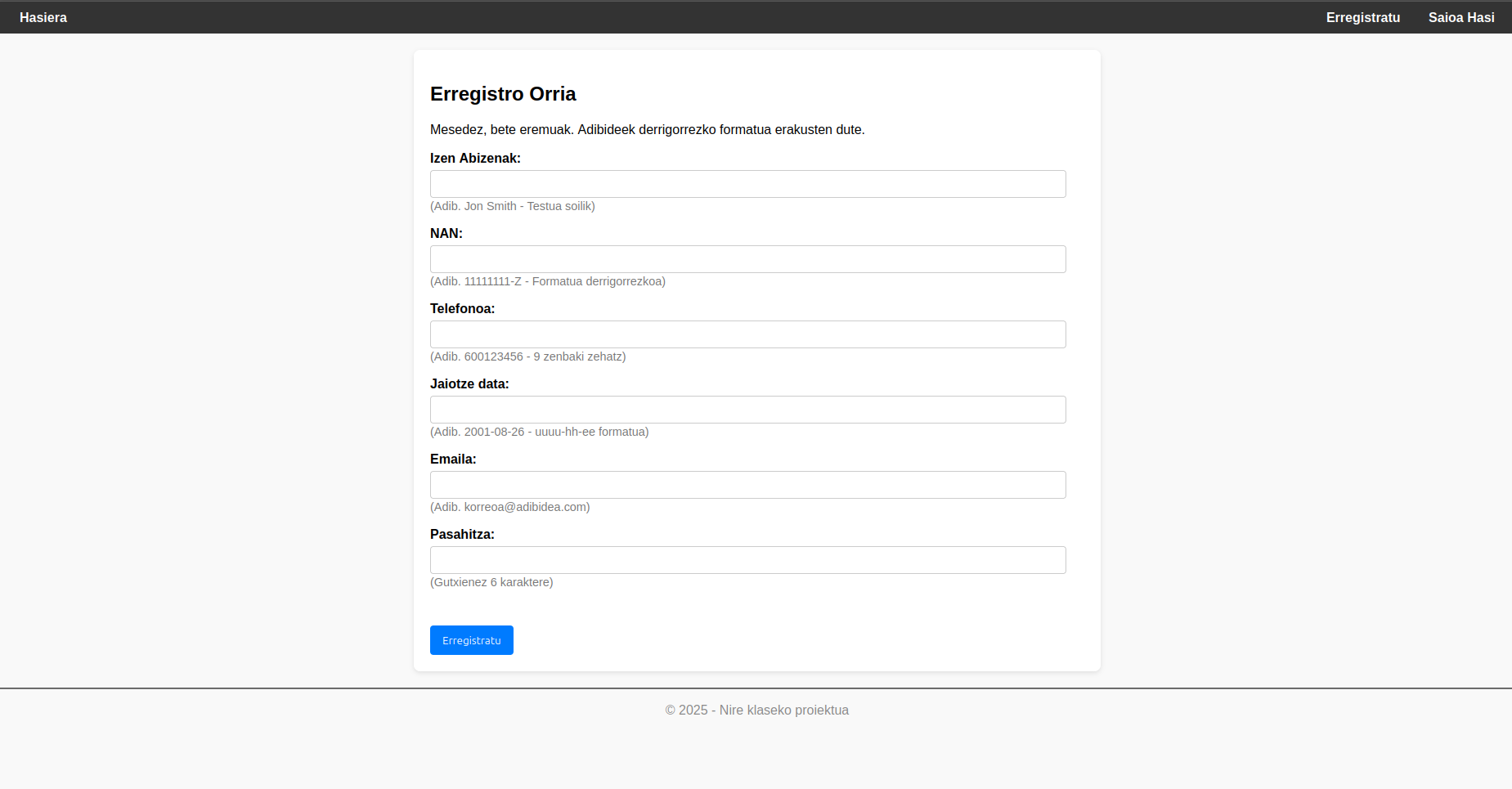
Behin erregistratutak, web-orrira sartzen gara.

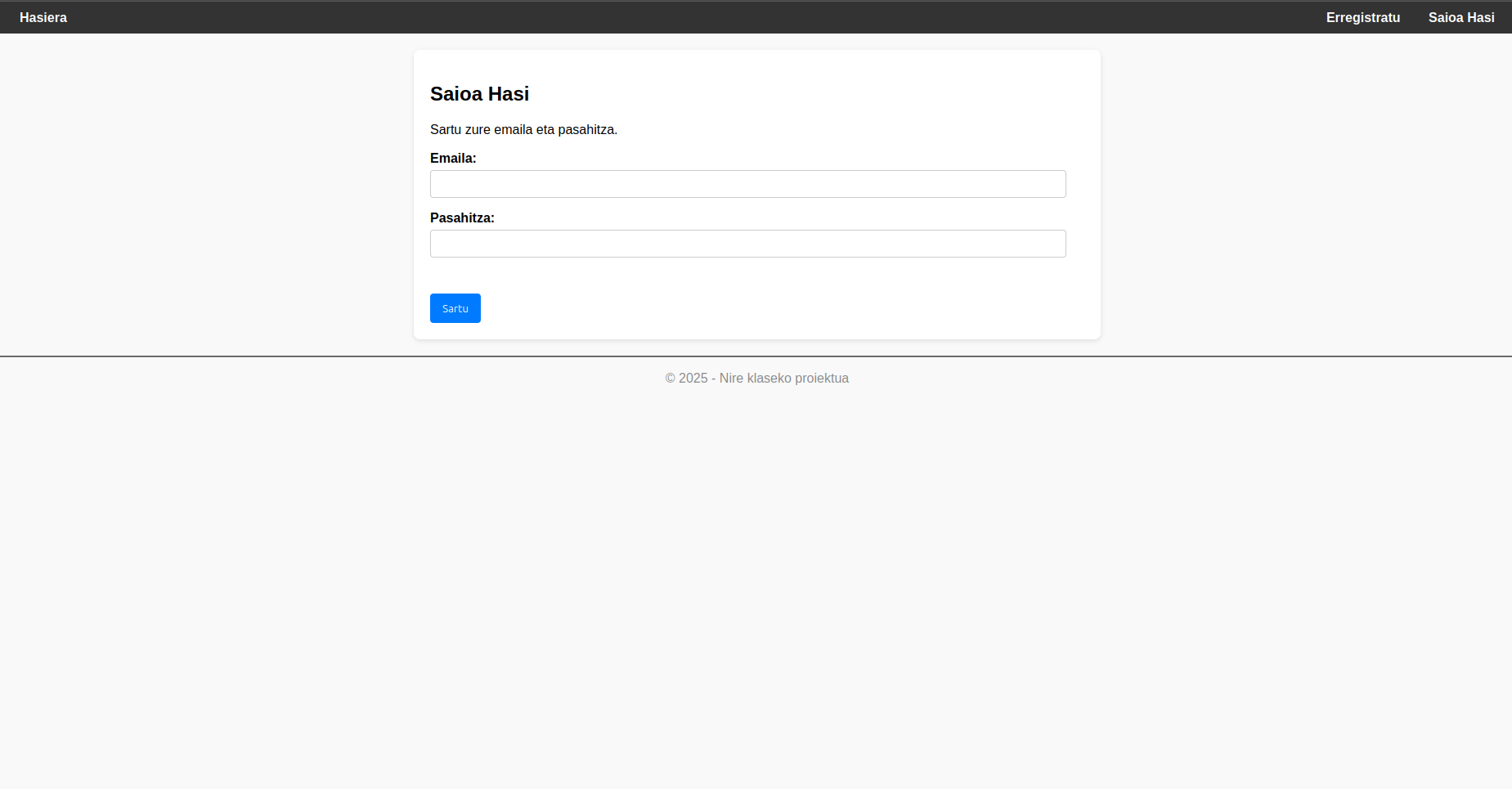
Kontua sortuta badaukagu, saioa hasi dezakegu (**3. argazkian** bezala). Behin saioa hasita, web-orrian sartuko gara ere bai.

**1)**



**2)**

**3)**

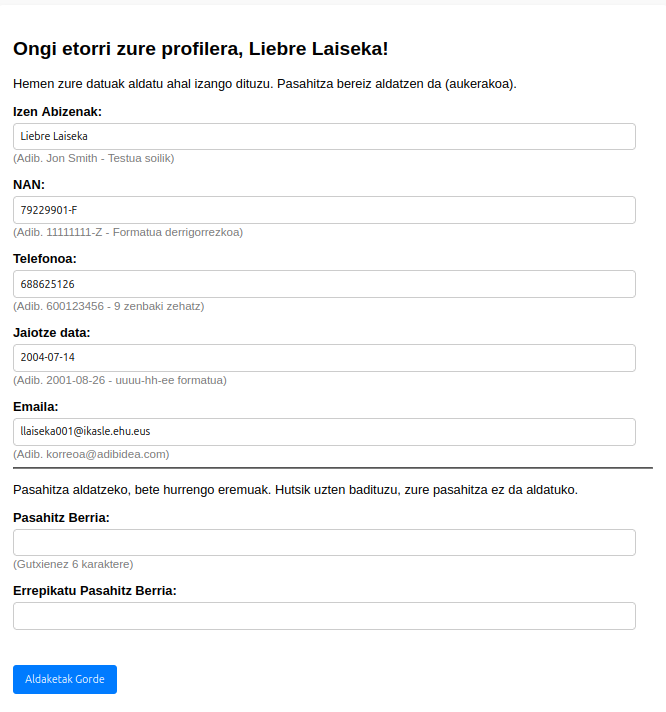


## Profilaren datuak aldatu

Saioa hasi eta gero, **agertzen zaizun lehen orria 4. irudian/argazkian dagoena da**. Bertan, **zure** **profil-datu guztiak** (Izena, Abizena, NANa, Telefonoa, Jaiotze-data eta Helbide elektronikoa/Emaila) **alda ditzakezu** (inprimaki bakoitzaren azpian **agertzen diren jarraibideei** **jarraituz**).

Profil-datuak aldatzeko atalaren jarraian, **pasahitza aldatzeko atala daukagu**. Horretarako, **pasahitz berria** sartu behar duzu lehen azpi-atalean eta, ondoren, pasahitz bera sartu behar duzu bigarrenean. **Pasahitzak berdinak ez badira**, “Pasahitzak ez datoz bat” mezua agertuko da.

**4)**



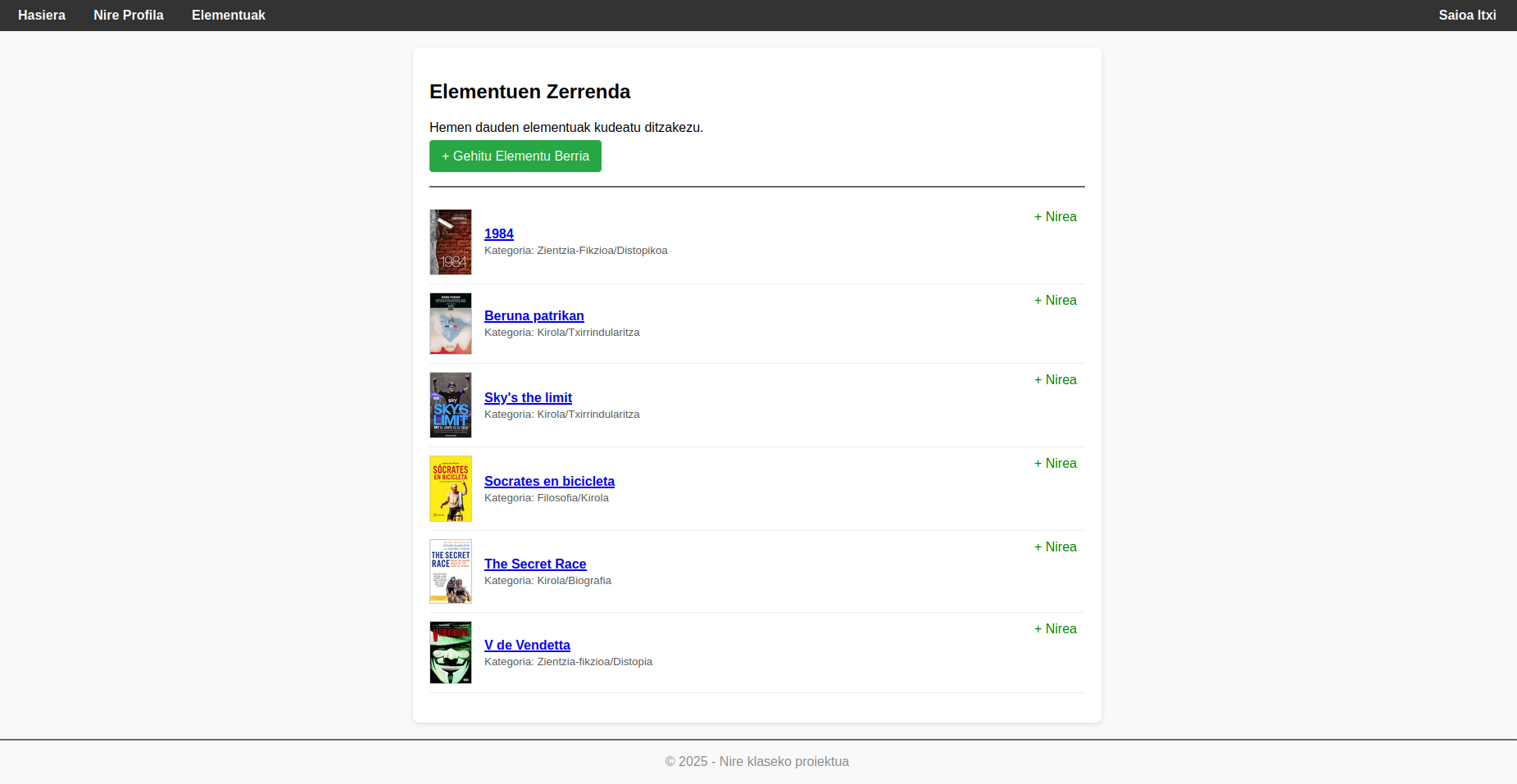
## Elementuak ikusi eta gehitu

Elementuen sekzioan, eskuragarri dauden liburuak ikusiko ditugu (**5. Argazkia**), han liburuaren izenburua eta zein kategoriakoa den ikusiko da. Era berean, liburuaren informazio ikusi nahi badugu izenburuan klik egin eta informazio gehiago eskainiko digun lehio joango gara**(6. Argazkia)**.

Liburu berri bat sartu nahi badugu, Insertando imagen... botoia pultsatuz gero, (7**. Argazkian)** ikus dezakeguna aterako zaigu. Han, liburuaren izenburua, deskribapena, prezioa, stocka, kategoria eta portadaren argazkia atzituko ditugu.  pultsatuz gero, liburua gehituko da.

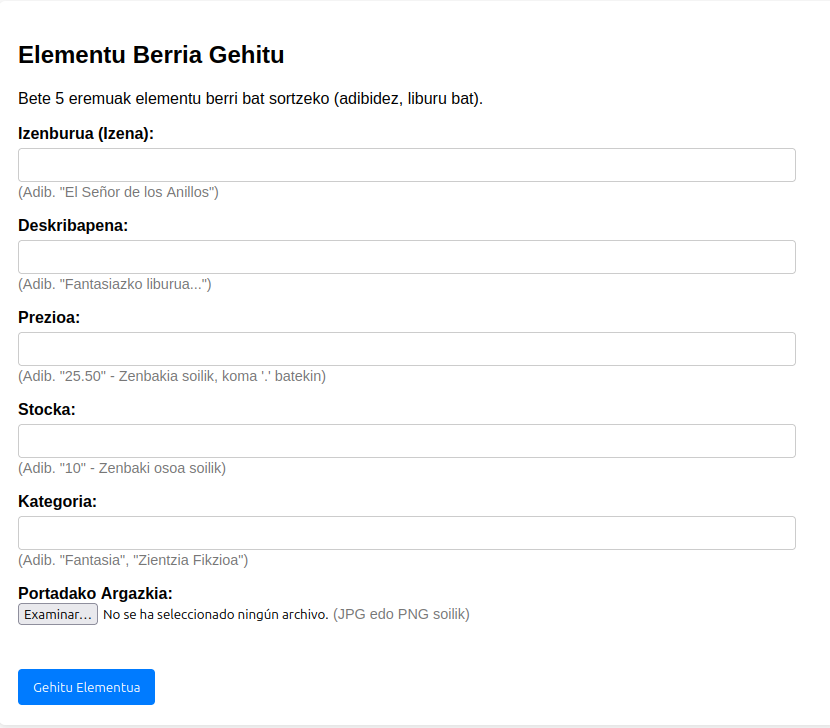
**Hau egitea posible izateko *~/docker-lamp/app* helbidean, *upload* direktoriak baimenak izan behar ditu, horretarako hurrenoga idatziko dugu: *$ sudo chmod -R 777 upload.***

**5)**

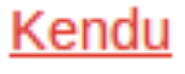


**6)** 

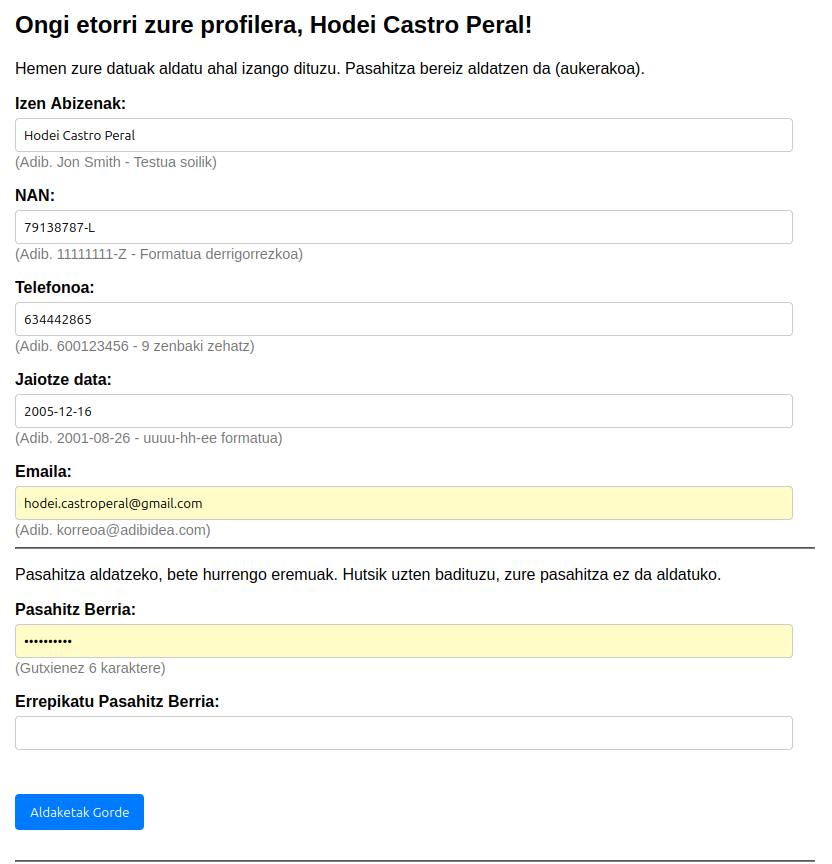
**7)**

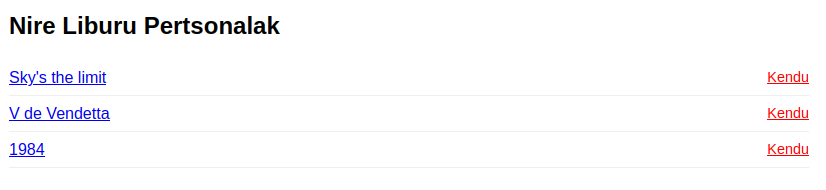


Karritora elementuak gehitu eta ikusi

Elementuetan, nahi dugun liburuan,  ikutuko dugu (5**. argazkian**), horrela liburua gure karritora gehituko da. Ondoren, karritoa gure profilearen beheko aldean egongo da (**8. argazkia**). Han ikus dezakegu  botoia, hori pultsatu ondoren, liburua gure karritotik kenduko da.

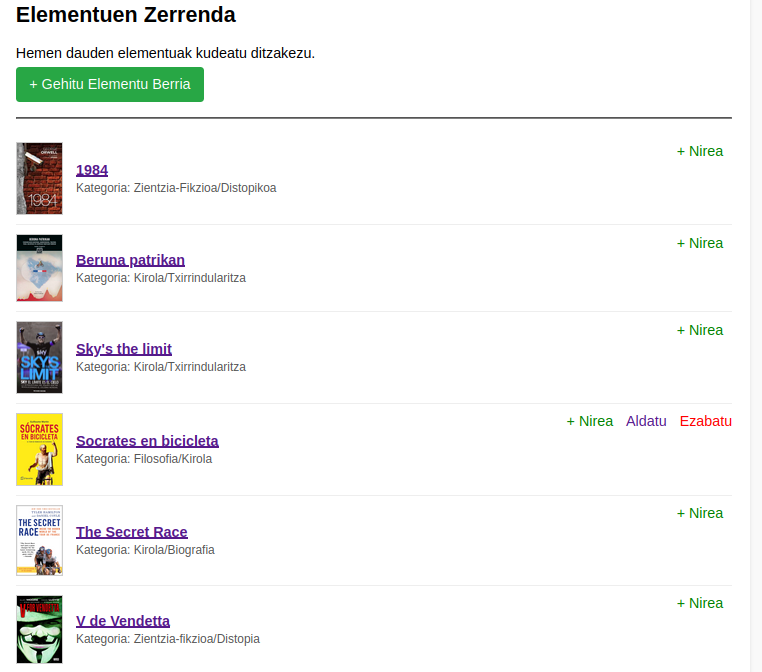
**8)**



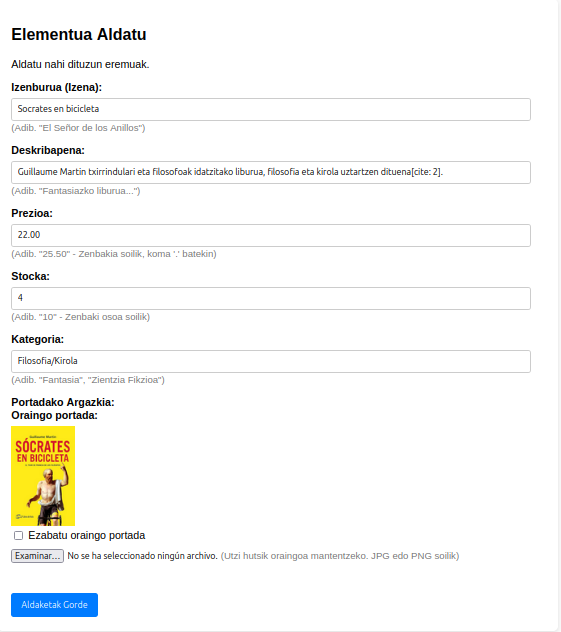


## Elementuak aldatu eta ezabatu

Elementuak ezabatzeko edo aldatzeko, Webgunera zuk igotako liburu bat izan behar da. Hots, ezin dituzu beste liburuak eraldatu(9 Argazkia). Aldatzeko momentuak, aukera dago 5 eremuak eraldatzeko eta nahiko bagenu portada ezabatzeko(10 Argazkia). Azkenik, ezabatzeko unean konfirmazio pantaila batera eramango gaitu webguneak ezabaketa baieztatzeko(11 Argazkia).

9)

10)



11)

