МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА»

Кафедра інженерії програмного забезпечення

**КУРСОВА РОБОТА**

(ПОЯСНЮВАЛЬНА ЗАПИСКА)

з дисципліни: «Web-технології. Ч.2»

на тему:

**«ІНТЕРНЕТ-МАГАЗИН ОДЯГУ»**

студента ІІ курсу групи ІПЗ-20-1

спеціальності 121 «Інженерія програмного забезпечення»

Кравчук Володимир Олександрович

(прізвище, ім’я та по-батькові)

Керівник ктн, доц. ІПЗ Морозов А.В.,

Дата захисту: "30" червня 2023 р.

Національна шкала \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Кількість балів: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Оцінка: ECTS \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Члени комісії \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ М.С. Граф

(підпис) (прізвище та ініціали)

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_. О.Г. Чижмотря

(підпис) (прізвище та ініціали)

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ О.В. Кузьменко

(підпис) (прізвище та ініціали)

Житомир – 2023

ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА»

Факультет інформаційно-комп’ютерних технологій

Кафедра інженерії програмного забезпечення

Освітній рівень: бакалавр

Спеціальність 121 «Інженерія програмного забезпечення»

«ЗАТВЕРДЖУЮ»

Зав. кафедри

А.В.Морозов

“\_\_\_\_”\_\_\_\_\_\_\_\_\_\_\_\_\_2021 р.

ЗАВДАННЯ

НА КУРСОВУ РОБОТУ СТУДЕНТУ

Кравчук Володимир Олександрович

1. Тема роботи: Розробка програми «Інтернет-магазин одягу»,

керівник роботи: ктн, доц Морозов А.В.

1. Строк подання студентом: “30” червня 2023р.
2. Вхідні дані до роботи: Розробити сайт " Інтернет-магазин одягу ".
3. Зміст розрахунково-пояснювальної записки(перелік питань. Які підлягають розробці)
   * + 1. Постановка завдання
       2. Аналіз аналогічних розробок
       3. Алгоритми роботи програми

4. Опис роботи програми

5. Програмне дослідження

1. Перелік графічного матеріалу(з точним зазначенням обов’язкових креслень)

1.Презентація до КП

2.Посилання на репозиторій:

1. Консультанти розділів проекту (роботи)

|  |  |  |  |
| --- | --- | --- | --- |
| Розділ | Прізвище, ініціали та посади консультанта | Підпис, дата | |
| завдання  видав | завдання прийняв |
| 1 | ст. викладач кафедри ІПЗ Чижмотря О.Г. |  |  |
| 2 | ст. викладач кафедри ІПЗ Чижмотря О.Г. |  |  |
| 3 | ст. викладач кафедри ІПЗ Чижмотря О.Г. |  |  |
|  |  |  |  |

1. Дата видачі завдання “ 22 ” вересня 2021 р.

|  |  |  |  |
| --- | --- | --- | --- |
| № з/п | Назва етапів курсового проекту | Строк виконання етапів проекту | Примітки |
| 1 | Постановка задачі | 10.11.2021 |  |
| 2 | Пошук, огляд та аналіз аналогічних розробок | 11.05.2023 |  |
| 3 | Формулювання технічного завдання | 12.05.2023 |  |
| 4 | Опрацювання літературних джерел | 13.05.2023 |  |
| 5 | Проектування структури | 15.05.2023 |  |
| 6 | Написання програмного коду | 20.05.2023 |  |
| 7 | Відлагодження | 12.06.2023 |  |
| 8 | Написання пояснювальної записки | 25.06.2023 |  |
| 9 | Захист | 30.06.2023 |  |

**КАЛЕНДАРНИЙ ПЛАН**

**Студент** \_\_\_\_\_\_\_\_\_\_\_\_\_\_ Кравчук В. О.

(підпис) (прізвище та ініціали)

**Керівник проекту** \_\_\_\_\_\_\_\_\_\_\_\_\_\_ Чижмотря О.Г.

(підпис) (прізвище та ініціали)

РЕФЕРАТ

Завданням на курсову роботу було створення сайту інтернет-магазину одягу із mvc-системою, можливість авторизації, регістрації, адміністрування.

Пояснювальна записка до курсового проекту (роботи) на тему «Інтернет-магазин одягу» складається з вступу, трьох розділів, висновків, списку використаної літератури та додатків.

Текстова частина викладена на 24 сторінках друкованого тексту.

Пояснювальна записка має 10 сторінок додатків. Список використаних джерел містить 11 найменувань і займає 1 сторінку. В роботі наведено 30 рисунків. Загальний обсяг роботи – 34 сторінки.

Ключові слова: САЙТ, МАГАЗИН, ОДЯГ.

ЗМІСТ

ВСТУП……………………………………………………………….………………7

РОЗДІЛ 1. АНАЛІЗ ПРОБЛЕМАТИКИ, МЕТОДІВ ТА ЗАСОБІВ ВИРІШЕННЯ ЗАДАЧІ……………………………………………………………….…….………..7

* 1. Аналіз задачі, засобів та методів її вирішення………………………7
  2. Аналіз існуючого програмного забезпечення за тематикою курсового проекту………………………………………………………......................7
  3. Технічне завдання на курсову роботу………………….…………….9

Висновки до першого розділу…………………………………….…………10

РОЗДІЛ 2. ПРОЕКТУВАННЯ ТА РОЗРОБКА ВЕБ ДОДАТКУ...........................11

2.1 Загальний алгоритм роботи веб-додатку……………….….…………...11

2.2 Розробка функціональних алгоритмів роботи програми..…………….11

2.3 Розробка програмного забезпечення…………………….……………...11

Висновки до розділу………………………………………….……………...12

РОЗДІЛ 3. ОПИС РОБОТИ З ВЕБ-ДОДАТКОМ ТА ЙОГО ТЕСТУВАННЯ.....13

3.1 Опис роботи з додатком……………………………………...….………13

3.2 Тестування роботи веб-додатку……………….……………….….…….20

Висновок до розділу……………………………………………….…….…...20

ВИСНОВКИ……………………………………………………………………........21

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ…………………………………….…….22

ДОДАТКИ……………………………………………………………………….......23

ПЕРЕЛІК УМОВНИХ СКОРОЧЕНЬ:

ОС – Операційна система

ВБ – Веб-додаток

ТЗ – Технічне завдання

ПКМ – права клавіша миши

Рис – рисунок

БД - база данних

І-п.с - інформаційно-пошукові системи

Б-ка – бібліотека

**РОЗДІЛ 1. АНАЛІЗ ПРОБЛЕМАТИКИ, МЕТОДІВ ТА ЗАСОБІВ ВИРІШЕННЯ ЗАДАЧІ**

**Аналіз задачі, засобів та методів її вирішення**

Задача полягає у тому, що потрібно:

1) Обрати потрібну інформацію

2) Визначити потреби звичайного користувача

3) Створити простий та зрозумілий для користувача інтерфейс сайту.

Було вирішено розробити веб-додаток за допомогої технологій HTML/CSS, PHP, SQL, Bootstrap, JavaScript.

**Аналіз існуючого програмного забезпечення за тематикою курсової роботи**.

Для аналізу існуючого програмного забезпечення за тематикою курсового проекту, було вибрано три конкуруючі сайти:

OLX - платформа онлайн-оголошень, яка об'єднує людей для покупки, продажу або обміну товарами та послугами. Станом на 2018 рік на майданчику зареєстровано 1,5 млн продавців, розміщено понад 11 млн оголошень і кожну хвилину додається близько 100 нових.



**Рис.1.1.** Olx

#### Shafa – це сторінка, на якій зручним чином представлені ваші речі з можливістю фільтрувати за розмірами, станом і ціною. А також є можливість сортувати речі й переглядати ваші оновлення.

#### SHAFA ᐈ Одежда, Обувь, Аксессуары, выгодные цены в Киеве в Украине

**Рис.1.2.** Shafa

#### Prom - український маркетплейс, проєкт IT-компанії EVO. На його платформі підприємці самостійно створюють інтернет-магазини і / або розміщують свої товари в загальному каталозі. Для покупців на Промі зібрано більше 100 мільйонів товарів.



**Рис.1.3.** Prom

**Найменування програмного засобу**

Повне найменування веб-додатку: “Інтернет-магазин одягу” (надалі "веб-додаток"). Коротка назва веб додатку - “Tailor ”

**Призначення розробки та область застосування**

Веб-сайт “Tailor ” призначений для розміщення одягу, їх купівлі, та продажу.

Веб-сайт дозволить користувачам загружати товари на сайт та ділитись ними із іншими, переглядати інші пропозиції.

**Найменування розробника та замовника.**

Розробник даного продукту – студент групи ІПЗ-20-1 Кравчук Володимир Олександрович (надалі "розробник").

Замовник програмного продукту – кафедра інженерія програмного забезпечення Житомирського державного технологічного університету в межах виконання курсової з дисципліни «Інтернет-програмування» Чижмотря Олена Генадіївна, Морозов Андрій Васильович

**Документ на підставі якого ведеться розробка**

Робота ведеться на підставі навчального плану за напрямом 121 “Інженерія програмного забезпечення “

**Вимоги до сайту:**

- мінімум 5 модулів (наприклад, "[Новини](https://learn.ztu.edu.ua/mod/forum/view.php?id=12739)", "Користувачі сайту", "Фотогалерея", "Сторінки", "Каталог продукції", "Форум", "Чат", "Приватні повідомлення", "Файли" і т.п); модулі вибрати самостійно;

- при правильному введенні логіну та паролю, повинен відкриватися доступ до системи адміністрування, яка забезпечить додавання, редагування та видалення даних;

- передбачити можливість виходу з режиму редагування і повернення у режим користувача;

- реалізувати асинхронний обмін даними за допомогою JSON;

-обов'язково використати: HTML5, CSS3, JavaScript, паттерн [MVC](http://ruseller.com/project.php?id=11), PHP7 та об'єктно-орієнтований підхід, бази даних MySQL);

**Сайт повинен виконувати наступні функції:**

1. Авторизація/Реєстрація, адміністрування.

2. Можливість додавання блоку для портфоліо

3. Можливість редагування та видалення

4. Можливість виходу з аккаунту

Потрібно притримуватись максимальної простоти інтерфейсу.

У разі виконання некоректних дій передбачено підказки та приклад максимальне роз’яснення причини, через яку певна дія не вдалась чи вдалась некоретно.

**Вимоги до надійності:**

Передбачити ситуації при виконанні некоректних дій з боку користувача.

**Вимоги до інтерфейсу**: Інтерфейс cайту повинен містити всі можливості сайту та бути простим й зрозумілим.

**Спеціальні вимоги:** Немає.

**Висновки до першого розділу**

У ході виконання першого розділу було визначено дозволені можливості користувача та зону роботи, можливість доступу до інформаціях з різними рівнями безпеки. Було визначено необхідні засоби та інструменти для реалізації веб-додатку, а також інтерфейсу користувача, яку необхідно створити.

**РОЗДІЛ 2. ПРОЕКТУВАННЯ ТА РОЗРОБКА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ**

**2.1 Проектування загального алгоритму роботи програми**

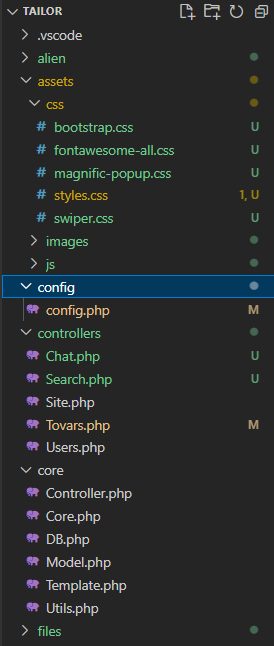
На даному етапі необхідно створити загальний алгоритм роботи програмного додатку. На початку роботи сайту користувача зустрічає головна сторінка, на якій розміщена основна інформація про сайт, є можливість входу та регістрації, також меню за яким можна перейти на інші сторінки сайту.

**2.2 Розробка функціональних алгоритмів роботи програми**

Користувач має можливість створення нових паролів, редагування. Для створення таких завдань створювались різні алгоритми.

**2.3 Розробка веб-додатку**

Після завершення проектування можна перейти до розробки веб-додатку. Етап проектування спростив розробку додатку та дав поштовх для розробки сайту. Було створено рішення та проект, який складається з форм та БД.



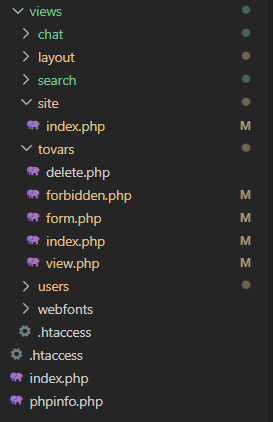


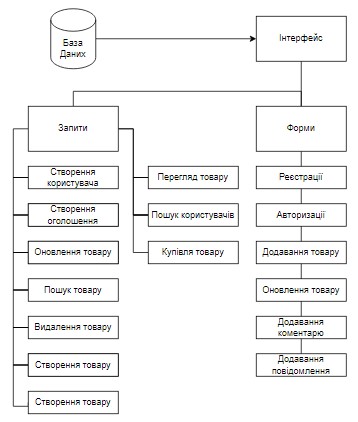
Рис 2.3 Вигляд структури сайту

**Висновки до другого розділу**

У ході виконання другого розділу було спроектовано та розроблено у вигляді блок-схеми загальний алгоритм, розроблено веб-додаток на основі блок-схеми.

**РОЗДІЛ 3. ОПИС РОБОТИ З ВЕБ-ДОДАТКОМ ТА ЙОГО ТЕСТУВАННЯ**

**3.1. Опис роботи з програмним додатком**

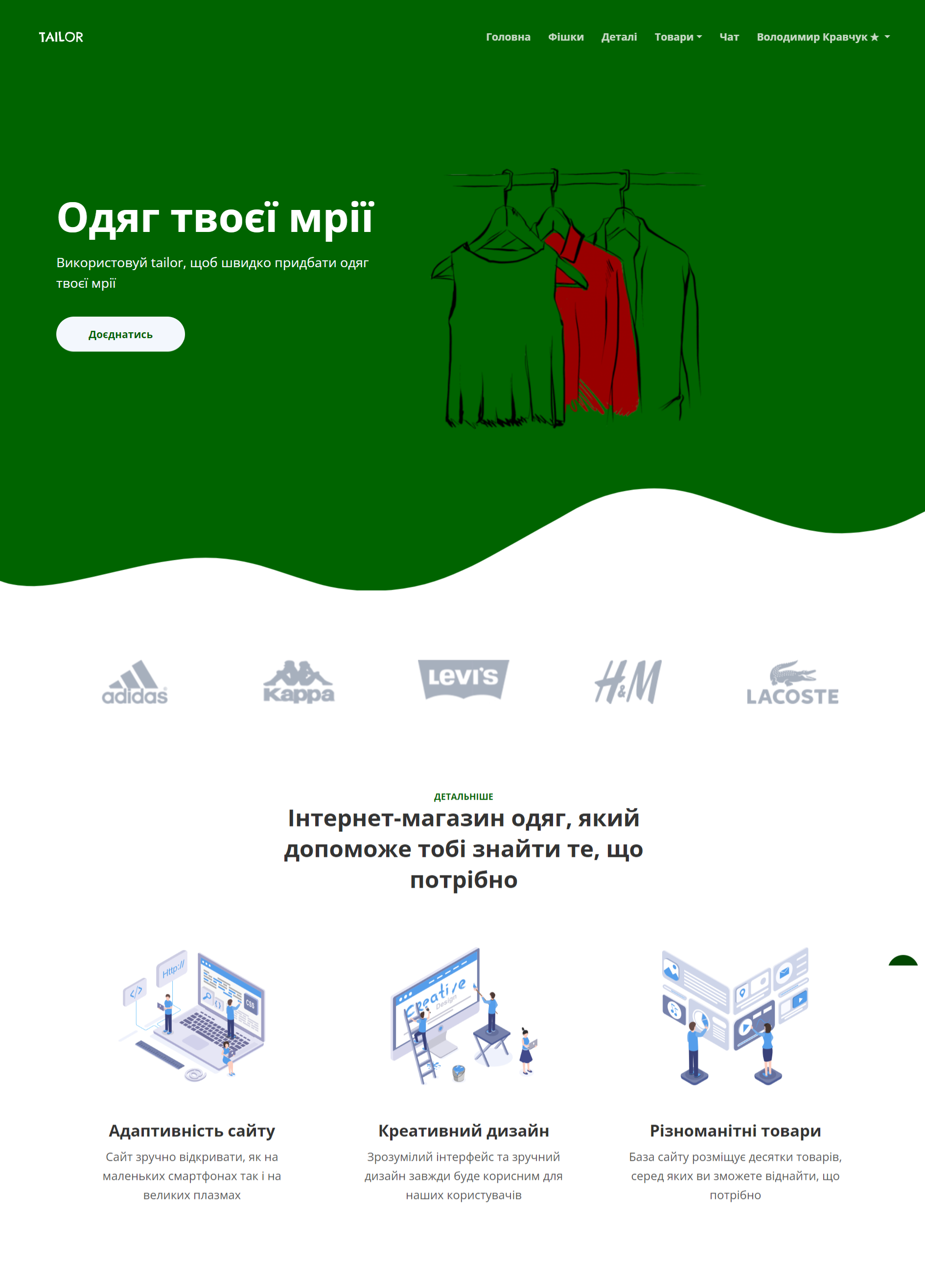


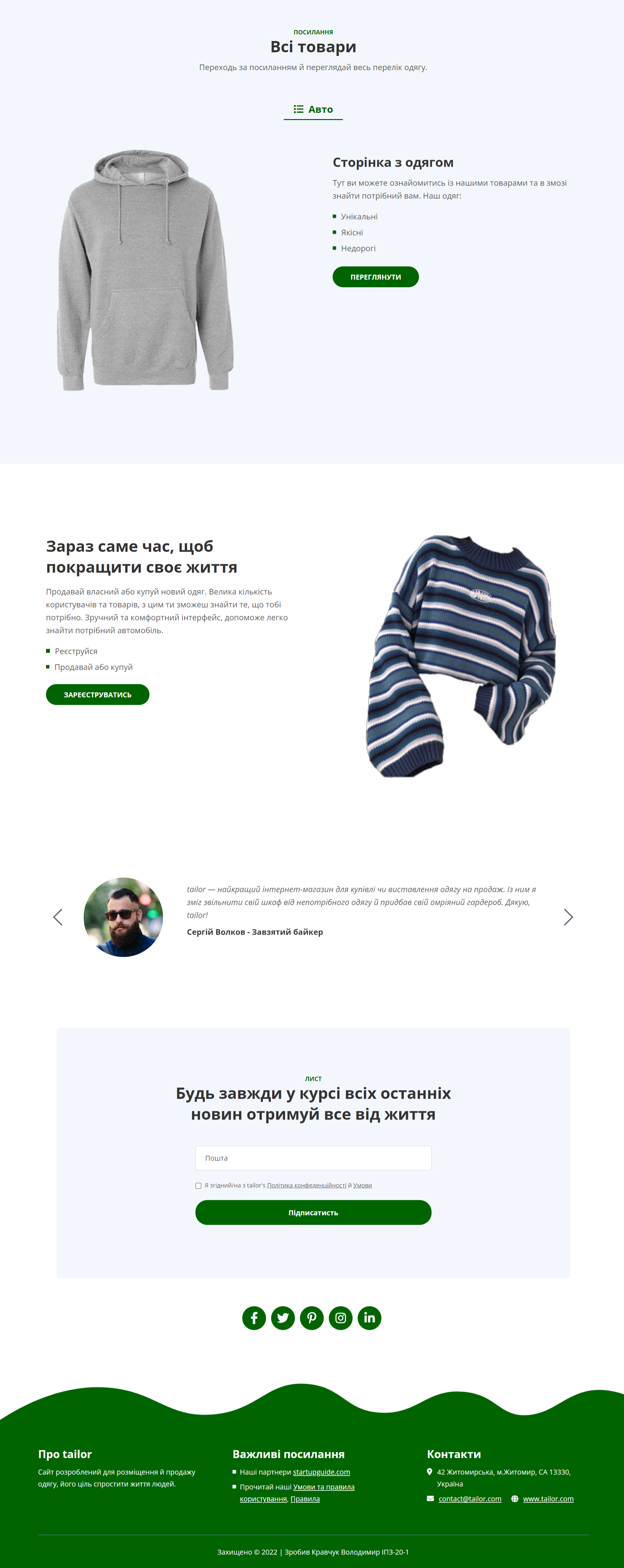
**Рис.3.1** Структура інтерфейсу

На рисунку 13 було зображено загальну схему інтерфейсу сайту інтернет-магазину. Далі будуть зображені форми та сторінки, що показують фінальний результат виконання проектування інтерфейсу веб-сайту. Кожна з форм чи сторінок супроводжується розробкою функцій. Більшість випадків побудови форми редагування та додавання були об’єднані з метою зменшення коду.

**Форми та сторінки інтерфейсів:**

Відкриваємо наш сайт за адресою: **http://tailor**[**/**.](http://trocar/) Відкривається головна сторінка сайту, на ній зоображені основні блоки сайту: хеадер, карусель із брендами, переваги сайту, про нас, відгуки, форма для пошти і тд…

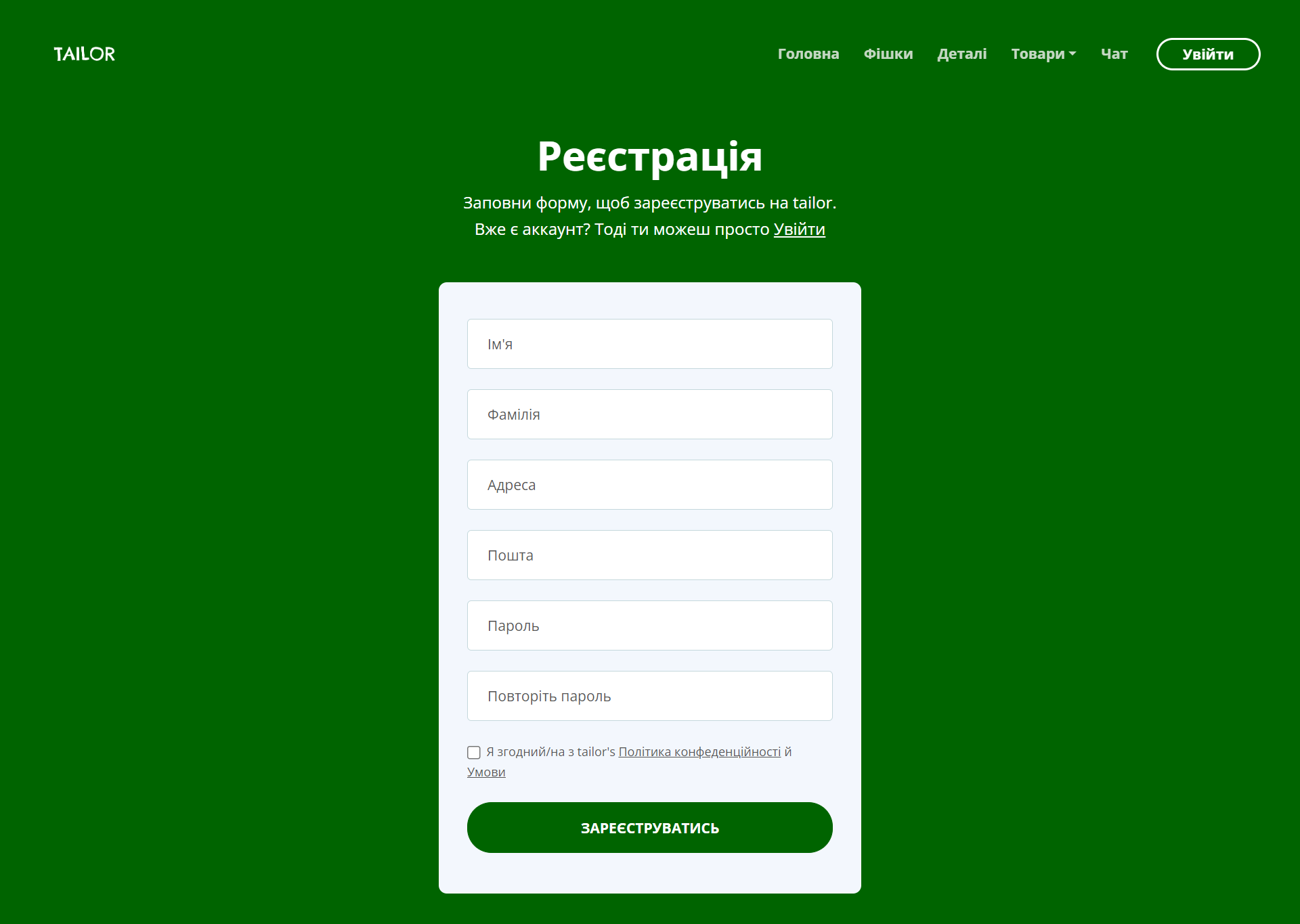




**Рис.3.2** Головна сторінка сайту

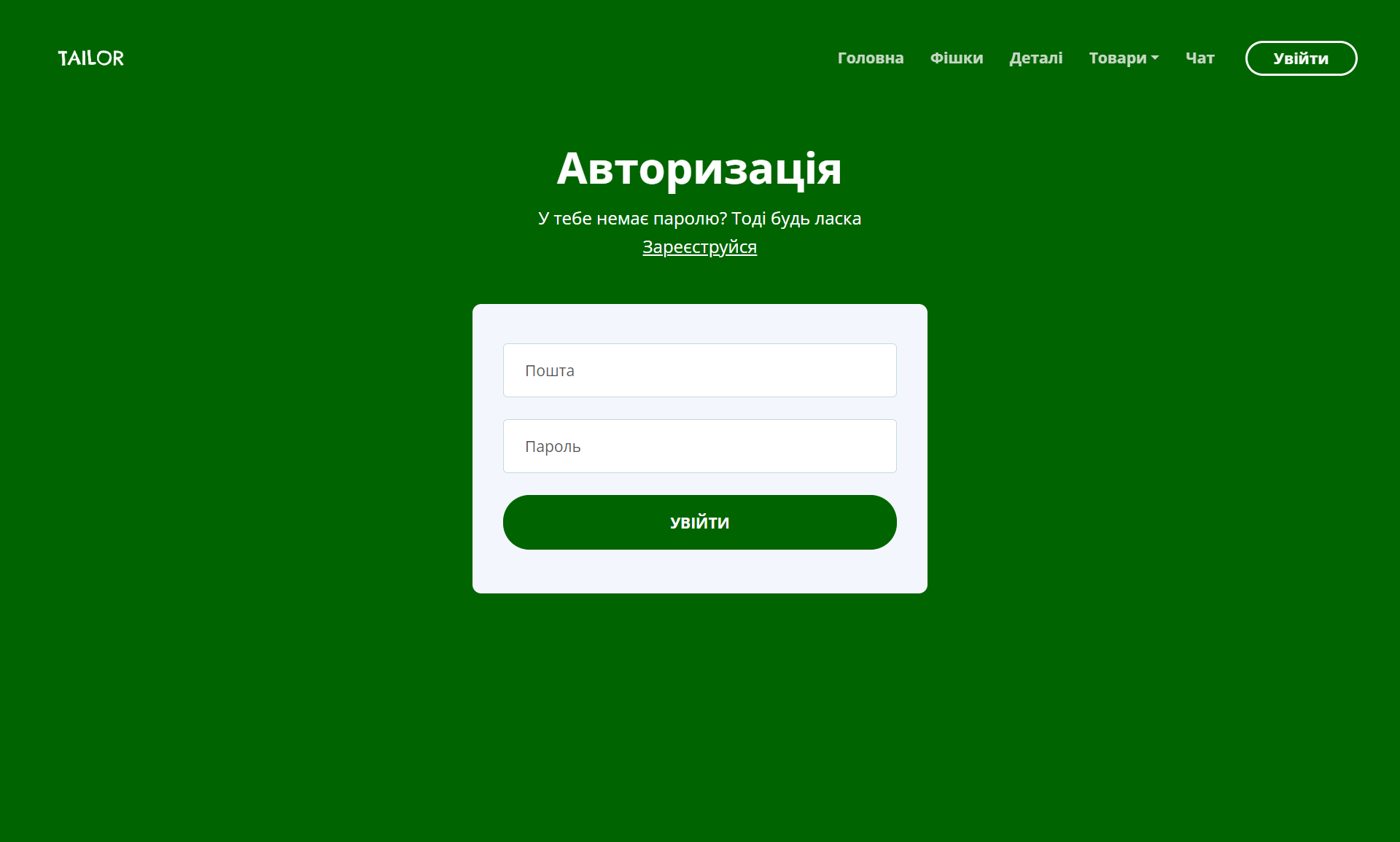
Реєструємось на сайті. Форма реєстрації приймає ім’я, прізвище, пошту, номер телефону, пароль та повторний пароль. Перераховані поля є

обов’язковими. Перед відправкою форми на сервер відбувається перевірка того, чи співпадають паролі та чи існує користувач із такою самою поштою, в негативному випадку, користувача повідомляють про це і попросять змінити дані.

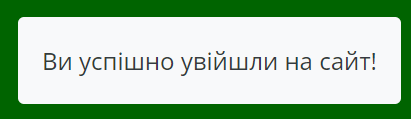


**Рис.3.3** Форма для реєстрації користувача в систему

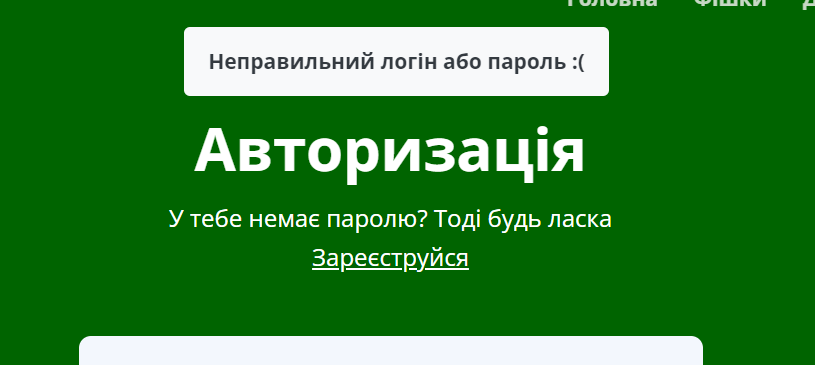
Заходимо на сайт. Дана форма приймає логін та пароль. При відправці запиту на під’єднання користувача до веб-додатку, відбувається перевірка чи співпадають пароль та логін користувача. Якщо ж веденого користувача не виявлено чи навпаки увійти вдалось, буде висвітлено сповіщення про невдалу чи вдалу авторизацію.

****

**Рис.3.4** Сторінка авторизації

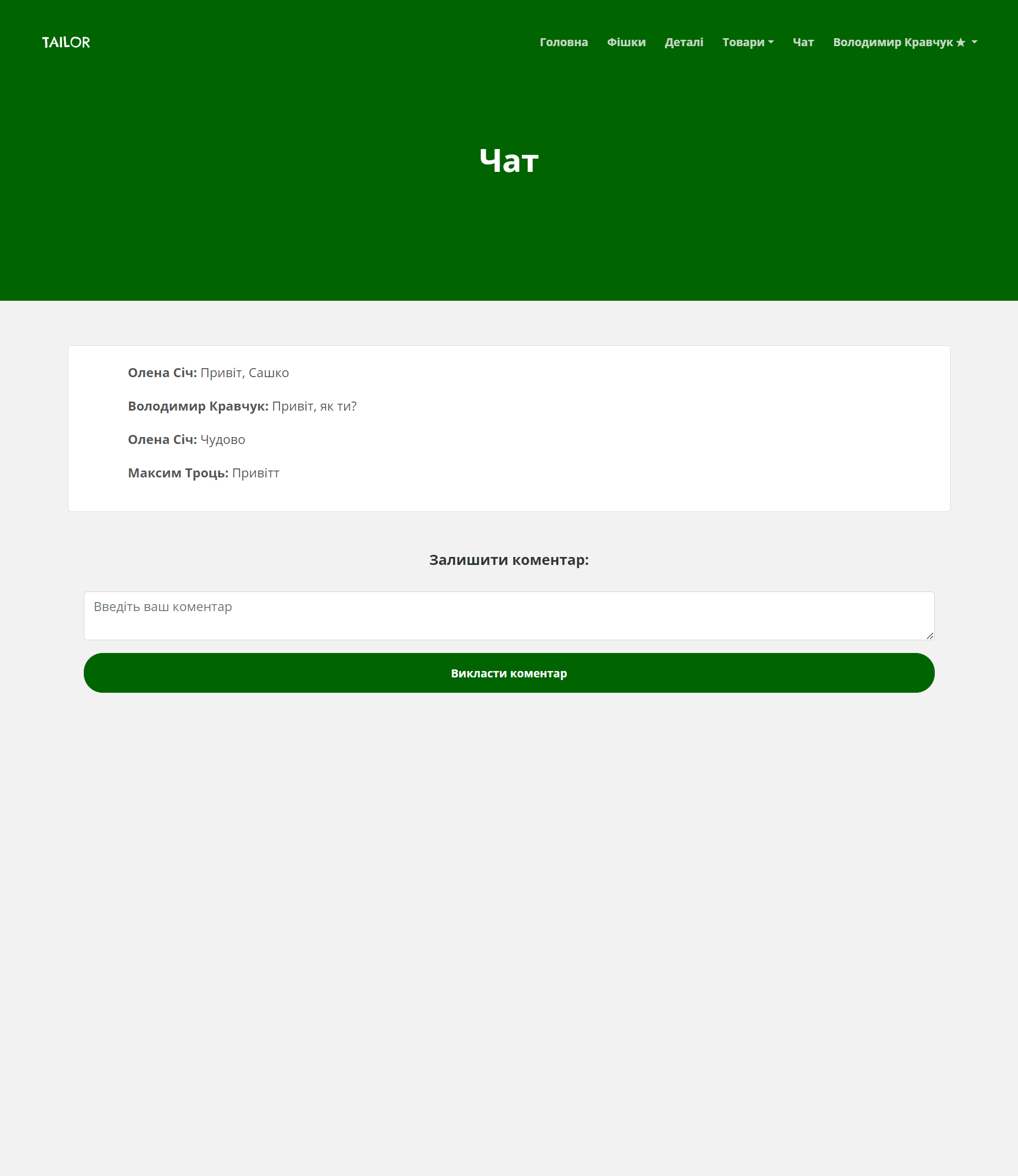


**Рис.3.5** Сповіщення про вдалу авторизацію



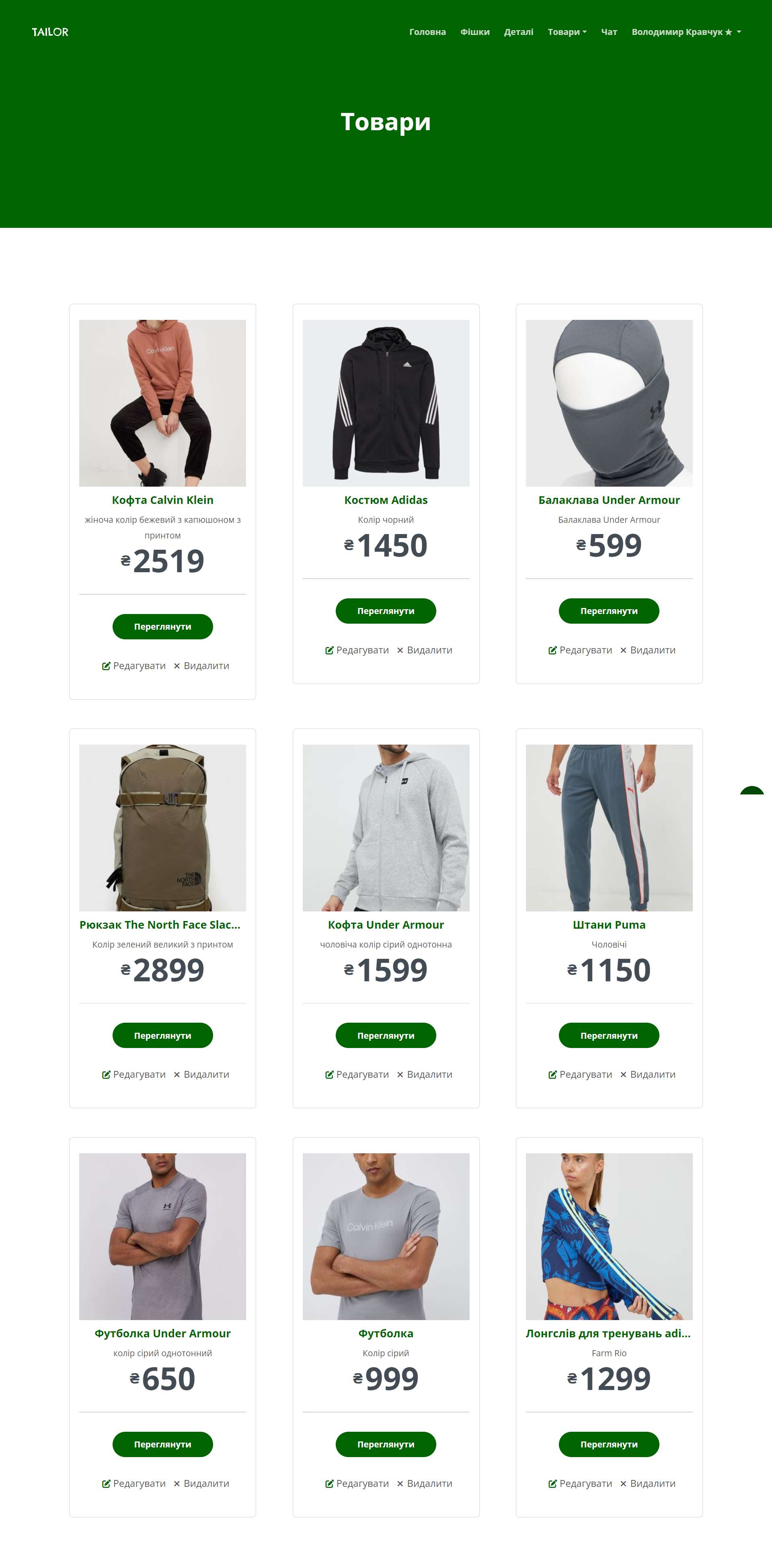
**Рис.3.6** Сповіщення про невдалу авторизацію

Переглядаємо сторінку Чат, тут реалізована можливість залишити повідомлення в чат користувачів, для цього створена окрема таблиця в базі даних **chat**, форма приймає лише повідомлення, можливість залишити його мають лише авторизовані користувачі:



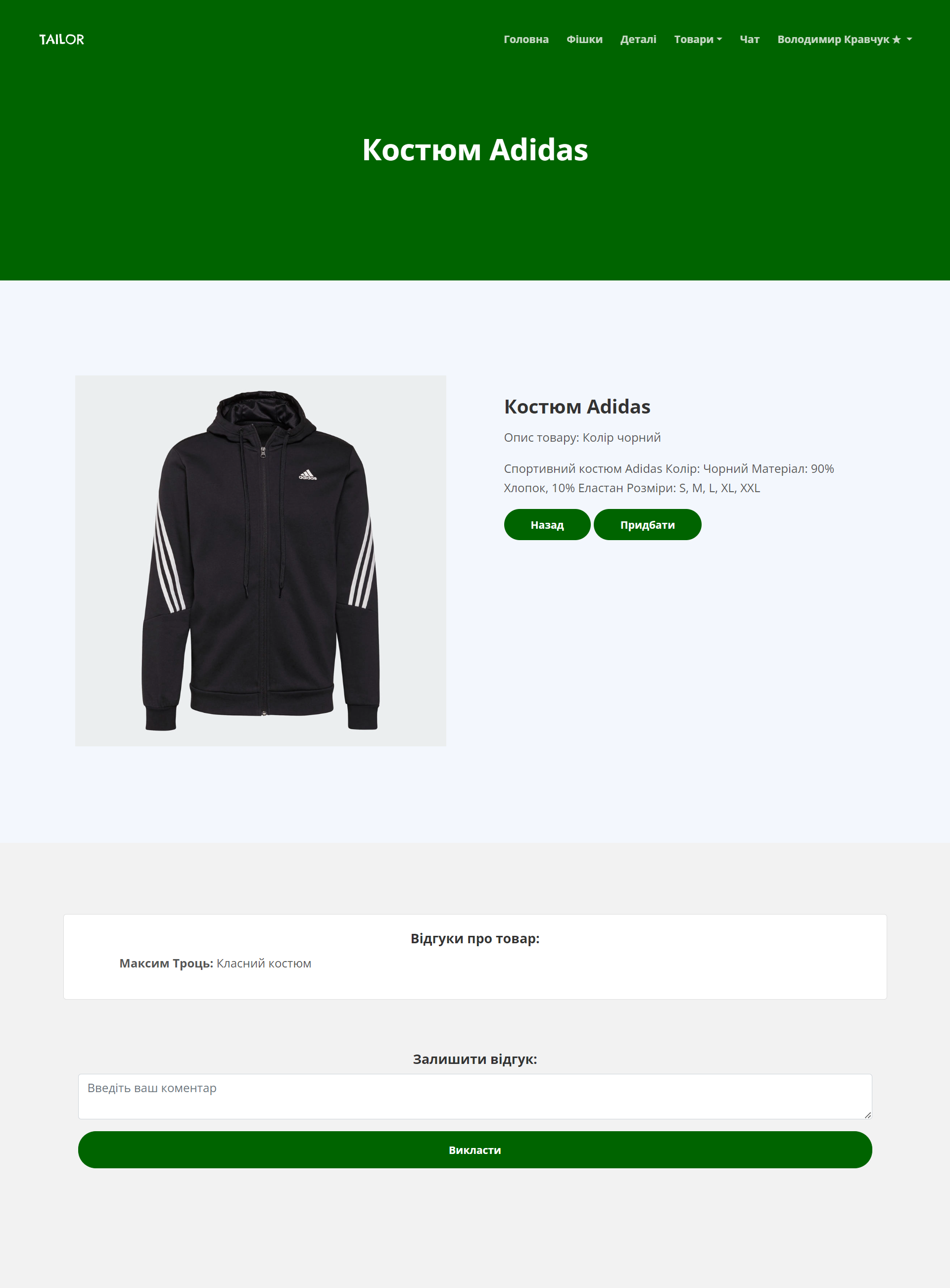
**Рис.3.7** Сторінка “чат користувачів

Переглядаємо сторінку Товару, на ній висвітлюється список із оголошень, і оскільки ми зайшли через роль адміністратора, в нас є функція відредагувати чи видалити повідомлення:



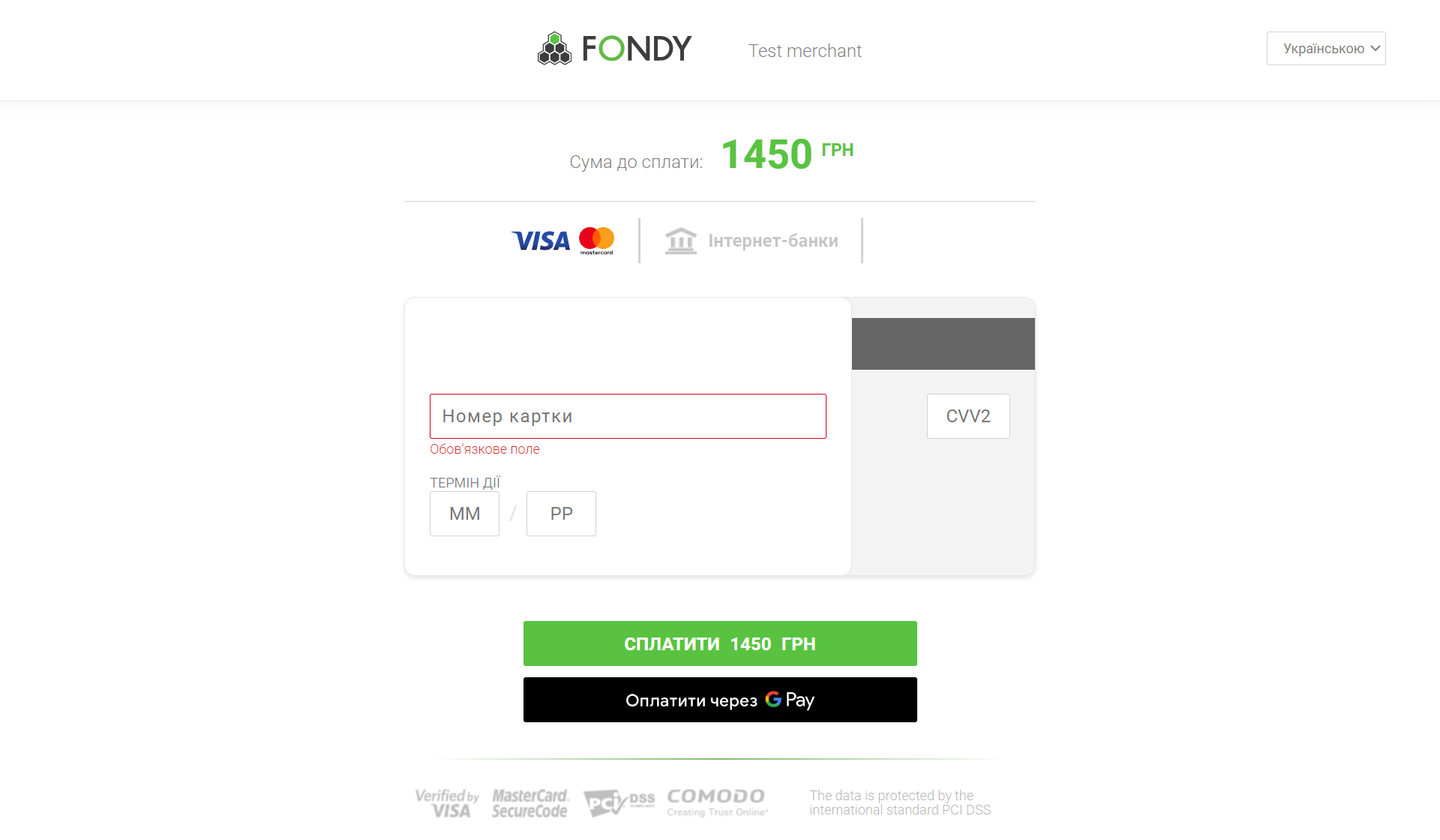
**Рис.3.8** Сторінка “товарів”

Переглянемо одного із товару детальніше, із бази данних підтягується вся інформація про цей товар, можна залишити коментар до нього:



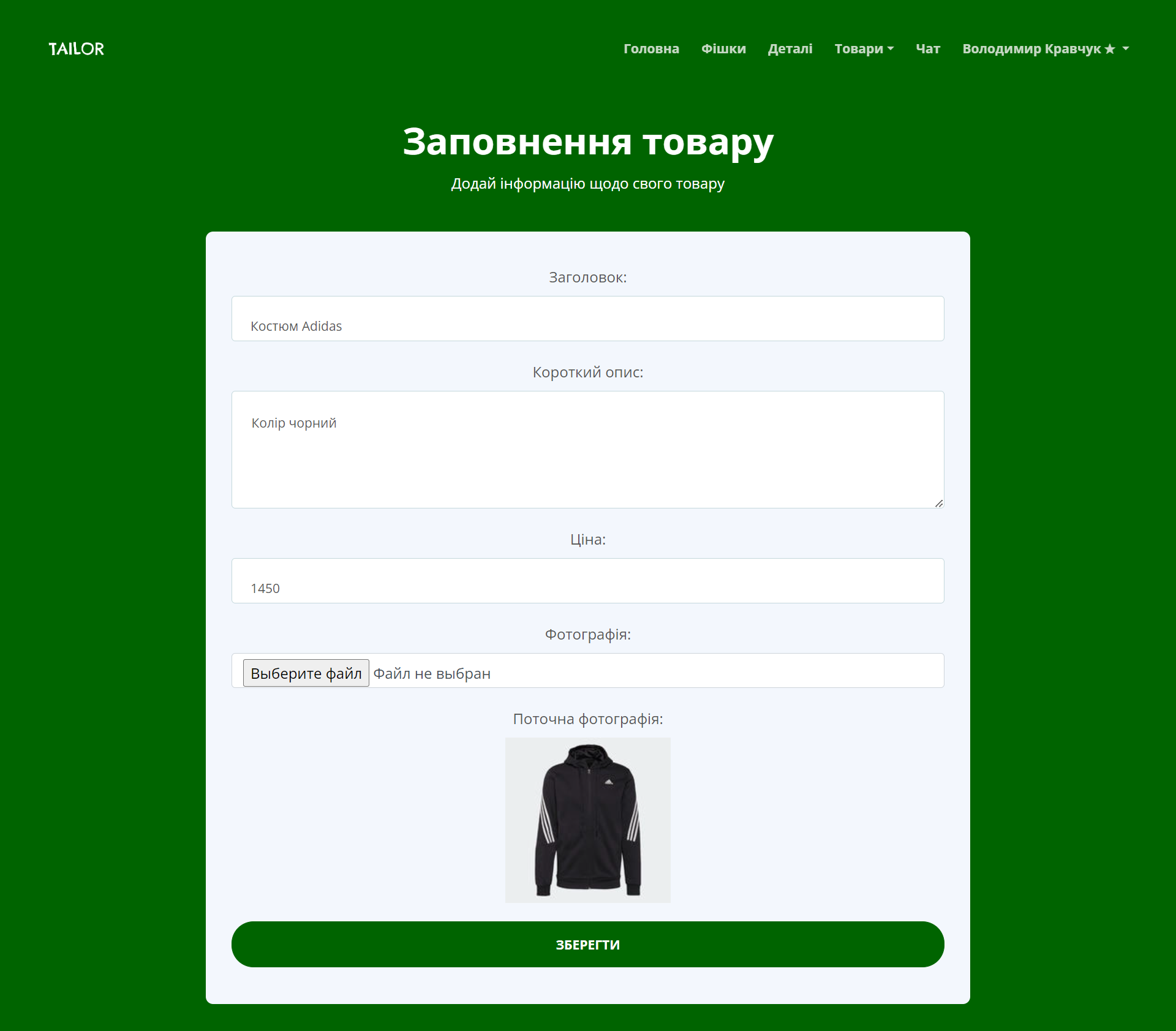
**Рис.3.9** Сторінка товару

Нажавши на кнопку придбати, покупця переводить на сторінкою із оплатою

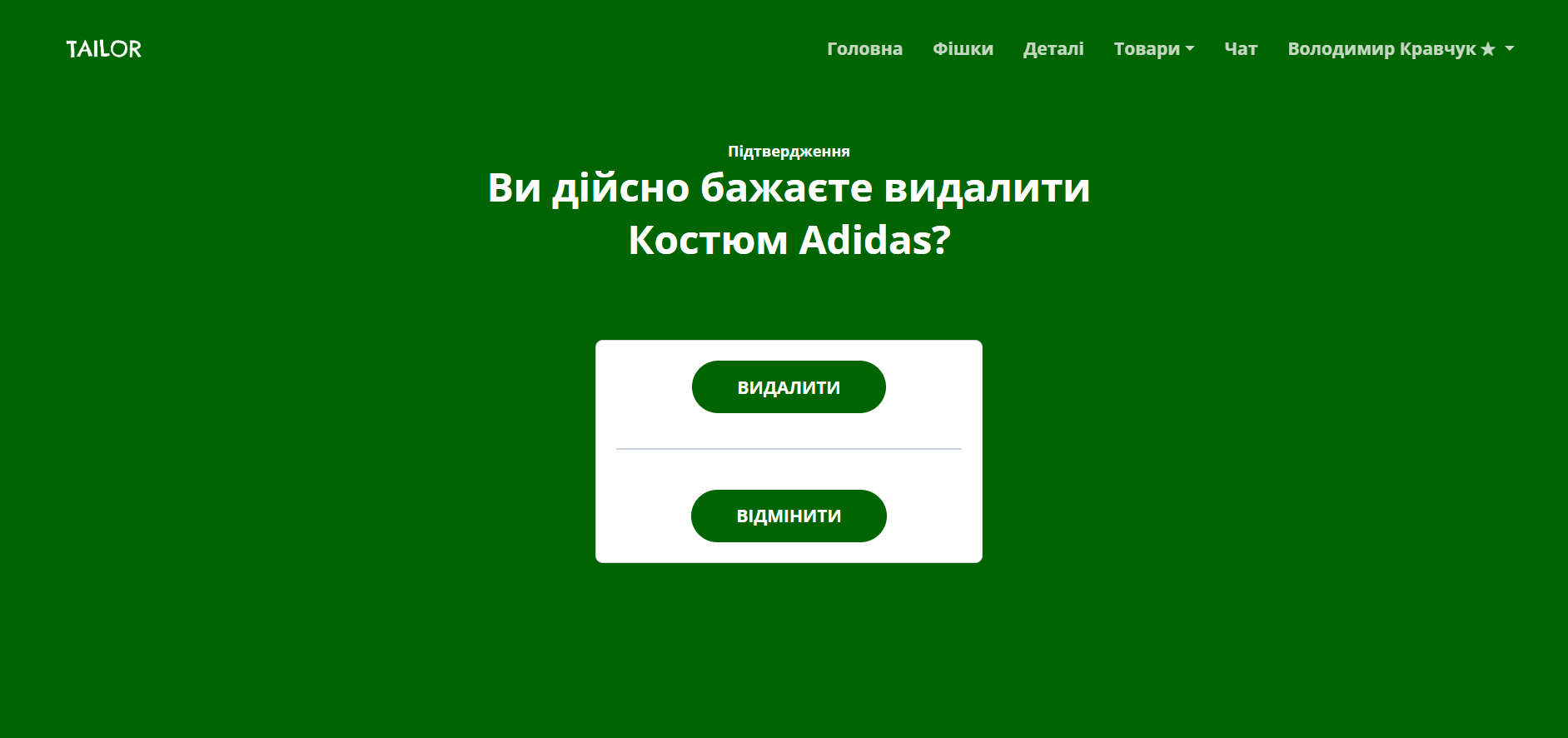


**Рис.3.10** Купівля товару

Ті товару, які були створені активним аккаунтом або коли користувач це адміністратор, можна редагувати або видаляти дописи. Ось приклад редагування або створення допису. Форма приймає заголовок, назву, текст, і фотографію:



**Рис.3.11** Сторінка “товару”. Приклад редагування товару



**Рис.3.14** Сторінка “товару”. Приклад видалення

**3.2. Тестування роботи програмного забезпечення**

Було прийняте рішення проведення ручного тестування. Та після підтвердження відсутності помилок, було прийнято рішення, що помилок немає або вони не впливають на роботу. Однак якщо користувач введе не відповідні значення його про це попередить программа

**Висновок до третього розділу**

В цьому розділі була описана документація до розробленого веб-додатку, яке описує роботу здодатком для звичайного користувача. Програма має загалом інтуїтивно зрозумілий інтерфейс.

В результаті тестування були виявлені та виправлені помилки, після виправляння помилок, тестування було проведено повторно. На даному етапі робота програми є стійкою до збоїв.

Загалом у процесі тестування було засвідчено, що програма виконує свої функції рівно так, як від неї очікувалось

**ВИСНОВКИ**

В результаті виконання цієї роботи вивчено основи прикладної галузі, також отримав досвід у написанні веб-додатку ”Інтернет-магазин одягу”. На перший погляд тема є дуже простою та що в сайті немає нічного складного, але в процесі реалізації постало завдання, зробити все в найкращому вигляді, автоматично та з передбаченням можливих помилок. Було проведено тестування у ручному режимі, після виявлення помилок, їх було виправлено та пере-протестовано додаток, що дає можливість визнати, що додаток є стабільним. Таким чином була розроблена програма з використанням мови PHP,JS та засобів SQL, HTML CSS, використаний фреймфорк Bootstrap та phpMyAdmin.

**Список використаних джерел**

1. А.Зандстра, Ф.Матт. PHP Objects, Patterns, and Practice, 2013. Режим доступу: <https://www.apress.com/la/book/9781430260325>
2. Выразительный JavaScript [електронний ресурс] - 2017. Режим доступу: https://legacy.gitbook.com/book/karmazzin/eloquentjavascript\_ru/details
3. Все операции с БД в графическом приложении [Електронний ресурс]

Режим доступу до ресурсу: <https://metanit.com/sharp/adonet/3.5.php>

1. Дмитрий Кирсанов. “Веб-дизайн” [електронний ресурс] - 2006. Режим доступу: http://web-diz.com.ua/skachat/veb-dizayn--dmitriy-kirsanov-skachatbesplatno/
2. Документація бібліотеки JQuery [Електронний ресурс] // JQuery incorp. –

2016. – Режим доступ: [https://jquery-docs.ru](https://jquery-docs.ru/)

1. Основы наследования [електронний ресурс] - 2015. Режим доступу: [https://professorweb.ru/my/ level7/7\_1.php](https://professorweb.ru/my/csharp/charp_theory/level7/7_1.php)
2. Плюси і мінуси обєктно-орієнтованого програмування [електронний ресурс] - 2015. Режим доступу: [http://damp.biz/plyusi-i-minusi-obyektnooriyentovanogo-programuvannya/](http://damp.biz/plyusi-i-minusi-obyektno-oriyentovanogo-programuvannya/)
3. Создание SQL запросов [електронний ресурс] - 2016. Режим доступу:

<https://www.politerm.com/zuludoc/spatial_query_sql.htm#_sql_overview>

1. Тим Кедлек. “Адаптивный дизайн. Делаем сайты для любых устройств”

[електронний ресурс] - 2013. Режим доступу: <https://www.ozon.ru/context/detail/id/21098804/>

1. SQL [електронний ресурс] - 2016. Режим доступу: <https://uk.wikibooks.org/wiki/SQL>
2. MS Access и C# [Електронний ресурс]– 2018. – Режим доступу до ресурсу: [https://vscode.ru/prog-lessons/ms-access-i-c-sharp-rabotaem-s- bd.html](https://vscode.ru/prog-lessons/ms-access-i-c-sharp-rabotaem-s-%20bd.html)

**ДОДАТКИ**

Додаток А

Вихідний код проекту

**config.php**

<?php

$Config =

    [

        'Database' =>

        [

            'Server' => 'localhost',

            'Username' => 'root',

            'Password' => '',

            'Database' => 'tailor',

        ],

        'TovarsCount' => 25

    ];

**Tovars.php**

<?php

namespace controllers;

use core\Controller;

/\*\*

 \* Контроллер для модуля Tovars

 \*/

class Tovars extends Controller

{

    protected $user;

    protected $TovarsModel;

    protected $userModel;

    public function \_\_construct()

    {

        $this->userModel = new \models\Users();

        $this->TovarsModel = new \models\Tovars ();

        $this->user = $this->userModel->GetCurrentUser();

    }

    /\*\*

     \* Відображення початкової сторінки модуля

     \*/

    public function actionIndex()

    {

        global $Config;

        $title = 'Новини';

        $lastTovars = $this->TovarsModel->GetLastTovars($Config['TovarsCount']);

        return $this->render('index', ['lastTovars'=>$lastTovars], [

            'MainTitle' => $title,

            'PageTitle' => $title,

        ]);

    }

    /\*\*

     \* Перегляд товару

     \*/

    public function actionView()

    {

        $id = $\_GET['id'];

        $Tovars = $this->TovarsModel->GetTovarsById($id);

        $title = $Tovars['title'];

        return $this->render('view', ['model' => $Tovars],

        [

            'MainTitle' => $title,

            'PageTitle' => $title,

        ]);

    }

    /\*\*

     \* Додавання новини

     \*/

    public function actionAdd()

    {

        if (empty($this->user))

        {

            $titleForbidden = 'Доступ заборонено';

            return $this->render('forbidden', null,

                [

                    'MainTitle' => $titleForbidden,

                    'PageTitle' => $titleForbidden,

                ]);

        }

        $title = 'Додавання товару;

        if($this->isPost())

        {

            $result = $this->TovarsModel->AddTovars($\_POST);

            if ($result['error'] === false) {

                $allowed\_types = ['image/png', 'image/jpeg'];

                if (is\_file($\_FILES['file']['tmp\_name']) && in\_array($\_FILES['file']['type'], $allowed\_types)) {

                    switch ($\_FILES['file']['tmp\_name']){

                        case 'image/png':

                            $extension = 'png';

                            break;

                        default :

                            $extension = 'jpg';

                    }

                    $name = $result['id'].'\_'.uniqid().'.'.$extension;

                    move\_uploaded\_file($\_FILES['file']['tmp\_name'], 'files/Tovars/'.$name);

                    $this->TovarsModel->ChangePhoto($result['id'], $name);

                }

                return $this->renderMessage('ok', 'Новину успішно додано', null,

                    [

                        'PageTitle' => $title,

                        'MainTitle' => $title

                    ]);

            } else

            {

                $message = implode('<br/>', $result['messages']);

                    return $this->render('form', ['model' => $\_POST], [

                        'PageTitle' => $title,

                        'MainTitle' => $title,

                        'MessageText' => $message,

                        'MessageClass' => 'danger'

                ]);

            }

        } else {

            return $this->render('form', ['model' => $\_POST],

                [

                    'MainTitle' => $title,

                    'PageTitle' => $title

                ]);

            }

    }

    /\*\*

     \* Редагування

     \*/

    public function actionEdit()

    {

        $id = $\_GET['id'];

        $Tovars = $this->TovarsModel->GetTovarsById($id);

        $titleForbidden = 'Доступ заборонено';

        if (empty($this->user) || $Tovars['user\_id'] != $this->userModel->GetCurrentUser()['id'] && $this->userModel->GetCurrentUser()['access'] != 1)

        {

            return $this->render('forbidden', null,

                [

                    'MainTitle' => $titleForbidden,

                    'PageTitle' => $titleForbidden,

                ]);

        }

        $title = 'Редагування;

        if($this->isPost())

            {

                $result = $this->TovarsModel->UpdateTovars($\_POST, $id);

                if ($result === true) {

                    $allowed\_types = ['image/png', 'image/jpeg'];

                if (is\_file($\_FILES['file']['tmp\_name']) && in\_array($\_FILES['file']['type'], $allowed\_types)) {

                    switch ($\_FILES['file']['tmp\_name']){

                        case 'image/png':

                            $extension = 'png';

                            break;

                        default :

                            $extension = 'jpg';

                    }

                    $name = $id.'\_'.uniqid().'.'.$extension;

                    move\_uploaded\_file($\_FILES['file']['tmp\_name'], 'files/Tovars/'.$name);

                    $this->TovarsModel->ChangePhoto($id, $name);

                }

                return $this->renderMessage('ok', 'Новину редаговано', null,

                    [

                        'MainTitle' => $title,

                        'PageTitle' => $title,

                    ]);

                } else {

                    $message = implode('<br/>', $result);

                    return $this->render('form', ['model' => $Tovars],

                        [

                            'MainTitle' => $title,

                            'PageTitle' => $title,

                            'MessageText' => $message,

                            'MessageClass' => 'ok',

                        ]);

                }

            } else {

                return $this->render('form',  ['model' => $Tovars],

                    [

                        'MainTitle' => $title,

                        'PageTitle' => $title,

                    ]);

                }

    }

    /\*\*

     \* Видалення

     \*/

    public function actionDelete()

    {

        $title = 'Видаленн;

        $id = $\_GET['id'];

        if(isset($\_GET['confirm']) && $\_GET['confirm'] == 'yes')

        {

            if($this->TovarsModel->DeleteTovars($id))

            {

                header('Location: /Tovars/');

            } else {

                return $this->renderMessage('error', 'Помилка видалення, null,

                [

                    'MainTitle' => $title,

                    'PageTitle' => $title,

                ]);

            }

        } else {

            $Tovars = $this->TovarsModel->GetTovarsById($id);

            return $this->render('delete', ['model' => $Tovars],

            [

                'MainTitle' => $title,

                'PageTitle' => $title,

            ]);

        }

    }

    /\*\*

     \* Відображення списку

     \*/

    public function actionList()

    {

        echo "actionList";

    }

}

**Site.php**

<?php

namespace controllers;

use core\Controller;

class Site extends Controller

{

    public function actionIndex()

    {

        return $this->render('index', null, [

            'MainTitle' => 'Головна',

            null

        ]);

    }

}

**Controller.php**

<?php

namespace core;

/\*\*

 \* Базовий клас для всіх контролерів

 \*/

class Controller

{

    public function isPost()

    {

        if ($\_SERVER['REQUEST\_METHOD'] == 'POST')

            return true;

        else

            return false;

    }

    public function isGet()

    {

        if ($\_SERVER['REQUEST\_METHOD'] == 'GET')

            return true;

        else

            return false;

    }

    public function postFilter($fields)

    {

        return Utils::ArrayFilter($\_POST, $fields);

    }

    public function render($viewName, $localParams = null, $globalParams = null)

    {

        $tpl = new Template();

        if (is\_array($localParams))

            $tpl->setParams($localParams);

        if (!is\_array($globalParams))

            $globalParams = [];

        $moduleName = strtolower((new \ReflectionClass($this))->getShortName());

        $globalParams['PageContent'] = $tpl->render("views/{$moduleName}/{$viewName}.php");

        return $globalParams;

    }

    public function renderMessage($type, $message, $localParams = null, $globalParams = null)

    {

        $tpl = new Template();

        if (is\_array($localParams))

            $tpl->setParams($localParams);

        $tpl->setParam('MessageText', $message);

        switch ($type)

        {

            case 'ok' :

                $tpl->SetParam('MessageClass', 'success');

                break;

            case 'error' :

                $tpl->SetParam('MessageClass', 'danger');

                break;

            case 'warning' :

                $tpl->SetParam('MessageClass', 'warning');

                break;

        }

        if (!is\_array($globalParams))

            $globalParams = [];

        $globalParams['PageContent'] = $tpl->render("views/layout/message.php");

        return $globalParams;

    }

}

**Core.php**

<?php

namespace core;

/\*\*

 \* Головний клас ядра системи

 \* (синглтон)

 \*/

class Core

{

    private static $instance;

    private static $mainTemplate;

    private static $db;

    private function \_\_construct()

    {

        global $Config;

        spl\_autoload\_register('\core\Core::\_\_autoload');

        self::$db = new \core\DB($Config['Database']['Server'],

            $Config['Database']['Username'],

            $Config['Database']['Password'],

            $Config['Database']['Database']);

    }

    /\*\*

     \* Повертає екземляр ядра системи

     \* @return Core

     \*/

    public static function getInstance()

    {

        if (empty(self::$instance)){

            self::$instance = new Core();

            return self::getInstance();

        }

        else

            return self::$instance;

    }

    /\*\*

     \* Отримати об'єкт-зєднання з базою даних

     \*/

    public function getDB()

    {

        return self::$db;

    }

    /\*\*

     \* Ініціалізація системи

     \*/

    public function init()

    {

        session\_start();

        self::$mainTemplate = new Template();

    }

    /\*\*

     \* Виконує основний процес роботи CMS-системи

     \*/

    public function run()

    {

        $path = $\_GET['path'];

        $pathParts = explode('/', $path);

        $className = ucfirst($pathParts[0]);

        if (empty($className))

            $fullClassName = 'controllers\\Site';

        else

            $fullClassName = 'controllers\\'.$className;

        $methodName = ucfirst($pathParts[1]);

        if (empty($methodName))

            $fullMethodName = 'actionIndex';

        else

            $fullMethodName = 'action'.$methodName;

        if (class\_exists($fullClassName)){

            $controller = new $fullClassName();

            if (method\_exists($controller, $fullMethodName)){

                $method = new \ReflectionMethod($fullClassName, $fullMethodName);

                $paramsArray = [];

                foreach ($method->getParameters() as $parameter)

                {

                    array\_push($paramsArray, isset($\_GET[$parameter->name]) ? $\_GET[$parameter->name] : null);

                }

                $result = $method->invokeArgs($controller, $paramsArray);

                if (is\_array($result))

                {

                    self::$mainTemplate->setParams($result);

                }

            }

            else

                throw new \Exception('404 Not Found');

        } else

            throw new \Exception('404 Not Found');

        //echo "Class: {$fullClassName}, method: {$fullMethodName}";

    }

    /\*\*

     \* Завершення роботи системи та виведення данних

     \*/

    public function done(){

        self::$mainTemplate->display('views/layout/index.php');

    }

    /\*\*

     \* Автозавантажувач класів

     \* @param $fullClassName string Назва класу

     \*/

    public static function \_\_autoload($fullClassName)

    {

        $fileName = $fullClassName.'.php';

        if (is\_file($fileName))

            include($fileName);

    }

}