Код в файлах php максимально закомментирован.

В файле .htaccess тоже есть методы защиты.

XSS

Обработка Get для категории.

$cat\_id = intval($\_GET['cat\_id']); // intval - переводит строку в число.  
if ($cat\_id) { // Если перевод получился  
 // проверяем сколько АКТИВНЫХ товаров в данной категории  
 $findSection = mysqli\_query($connect, "SELECT COUNT(section\_product.section\_id\_section) FROM section   
 INNER JOIN section\_product ON section\_product.section\_id\_section=section.id\_section   
 INNER JOIN product on section\_product.product\_id\_product=product.id\_product  
WHERE product.product\_activity=1 AND section.id\_section='$cat\_id';");  
 $findSection = mysqli\_fetch\_all($findSection);  
 if ($findSection[0][0] == 0) {  
 // Если в категории нет активных товаров, то отправляем ответ 404 и показываем стр. 404.  
 header('HTTP/1.1 404 Not Found');  
 header('Status: 404 Not Found');  
 require 'Error.html';  
 exit;  
 }  
} else {  
 // Если перевод строки в число не получился, то тоже ответ 404  
 header('HTTP/1.1 404 Not Found');  
 header('Status: 404 Not Found');  
 require 'Error.html';  
 exit;  
}

Для пагинации на странице категории.

// Проверка для пагинации  
if (isset($\_GET['page'])) { // Если у Get есть данный параметр  
 $page = abs(intval($\_GET['page'])); // Переводим в положительное число  
 if (!$page) { // Если перевод не произошел, то 404  
 header('HTTP/1.1 404 Not Found');  
 header('Status: 404 Not Found');  
 require 'Error.html';  
 exit;  
 }  
} else $page = 1; // Если в ссылке нет Get с таким параметром присвоить 1  
// Считаем с какой записи начинать выборку на данной странице  
$art = ($page \* $kol) - $kol;  
  
$total = $findSection[0][0]; // всего записей  
$str\_pag = ceil($total / $kol); // сколько будет страниц, с округлением в большую сторону  
if ($page > $str\_pag) { // Если пользователь ввёл число больше, чем есть у категории, то 404.  
 header('HTTP/1.1 404 Not Found');  
 header('Status: 404 Not Found');  
 require 'Error.html';  
 exit;  
}

Аналогичная защита и на странице товара.

$id = intval($\_GET['id']); // intval - переводит строку в число.  
if ($id) { // Если это число  
 // запрос для проверки, если ли активный товар с данным ID, данные будут использоваться и дальше, поэтому так много полей  
 $products = mysqli\_query($connect, "SELECT product.name, product.price,product.price\_without\_sale,product.price\_with\_promocode,product.description, images.alt, images.link\_image FROM product   
 INNER JOIN images on images.id\_image=product.id\_main\_image where product.id\_product = '$id' and product\_activity=1;");  
 $products = mysqli\_fetch\_all($products);// преобразуем в массив  
 if (!$products) { // Если ноль записей, то на 404  
 header('HTTP/1.1 404 Not Found');  
 header('Status: 404 Not Found');  
 require 'Error.html';  
 exit;  
 }  
} else {  
 //Если не получилось преобразовать в число, то 404  
 header('HTTP/1.1 404 Not Found');  
 header('Status: 404 Not Found');  
 require 'Error.html';  
 exit;  
}

Дополнительное задание XSS.  
  
создал функцию для удаление лишних символов, так же функцию для проверки длины, чтобы соответствовала атрибутам в таблице БД. Все значения полей вызывают эти функции.

// Функция для того, чтобы узнать подходит ли строка под заданую длину  
function check\_length($value, $min, $max)  
{  
 $result = (mb\_strlen($value) < $min || mb\_strlen($value) > $max);  
 return !$result;  
}  
  
// Функция для очистики строк от всякой гадости  
function clean($value)  
{  
 $value = trim($value); // Обрезаем пробелы по краям  
 $value = stripslashes($value); // для удаления экранированных символов  
 $value = strip\_tags($value);//для удаления HTML и PHP тегов  
 $value = htmlspecialchars($value); // преобразует специальные символы в HTML-сущности ('&' преобразуется в '&amp;' и т.д.)  
 return $value;  
}

Но для начала нужно проверить, отправляли ли вообще форму

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

Проверяем с помощью регулярных выражений значение полей, к примеру Имя

// В имени должна быть первая буква заглавной, а другие строчные и только русские или недопустимая длина, то зануляем флаг  
if (!(preg\_match("/^[А-ЯЁ]{1}[а-яё]\*[а-яё]{1}$/u", $input['name']) && check\_length($input['name'], 2, 45))) {  
 $output['name'] = false;  
}

Если все поля прошли валидацию, то создаём подключение к БД

if ($output['name'] && $output['email'] && $output['topic'] && $output['question'] && $output['check']) {

Защита от SQL инъекций.

// Подготовленный запрос для вставки в таблицу  
$sql = "INSERT INTO `feedback\_form` VALUES (?, ?, ?, ?, ?, ?, ?)";  
// Разбираем запрос  
$stmt = mysqli\_prepare($connect, $sql);  
$n = Null; // переменная доя автоинкримета в таблице  
// Задаём параметры для запроса, указывая тип параметров  
mysqli\_stmt\_bind\_param($stmt, "sssssss", $n, $input['name'], $input['email'], $input['sex'], $input['birthday'], $input['topic'], $input['question']);  
// Выполняем запрос с переданными параметрами и закрываем  
mysqli\_stmt\_execute($stmt);  
mysqli\_stmt\_close($stmt);