

Dynamic Web Programming Homework 2

E-Commerce Website with PHP and MySQL

Kamil Onur Algan | 1842200001 | k4milonur@gmail.com

The Project website is a so-called Techno Shop, includes technological products in it. System was developed by using Php as backend and MySql as database.

In short explanation, this website has products in a relative category.

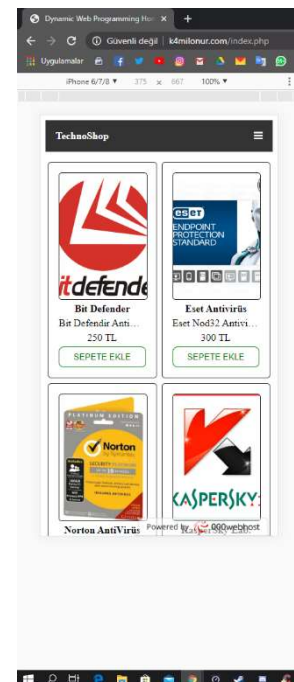
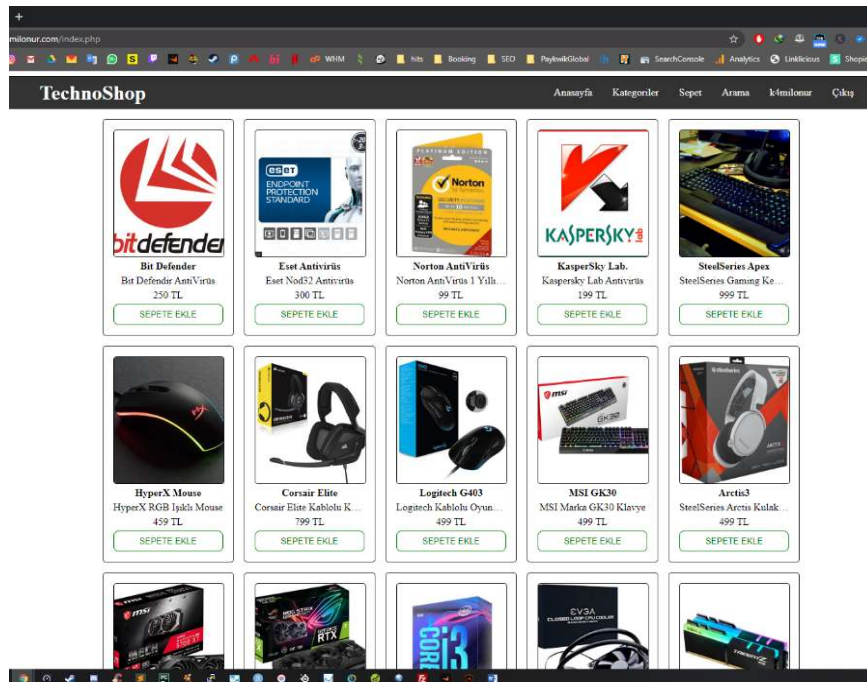
Users can search products on the website or list the products by category. They can add products to cart, remove an item or remove all items from the cart. However, they have to login before they complete transaction (Since this is a homework project and there is no payment, completing transaction leads nowhere). So, users can register and login.

There are two types of users in the system in terms of Normal and Admin. Normal users can update their profile information, such as personal information or contact information. If a user is defined as Admin, there is a link appear on the profile page which leads to administration page. Administration page has three main parts to manage. Admins can manage products, categories and users.

In the administration panel, admins can add or delete products, add or delete categories. They can also remove a user or change a user to Admin or to Normal.

The project website is also full responsive and it is able to use in any devices.

Desktop and Mobile Views:



There are several pages on the website:

Db.php: Includes the database information and provides the database connection. This page is included in all pages.

Index.php: The landing page. Includes a navigation bar and all the products listing.

Header.php: Includes the navigation bar. If user logged in, it shows a link with username and this link leads to profile page.

Register.php: Where users can register. Requires a Name, Username and Password. In order to keep it simple there is no password check.

Login.php: Where registered users can log in. Checks the input username and password from database. If it is correct, starts a session for user.

Logout.php: Ends the session.

Kategori.php: Includes a list of categories. Each category link lists the related products.

Sepet.php: The cart page. When a product is added to cart, it will be shown in this page. User can remove items from cart or complete transaction here.

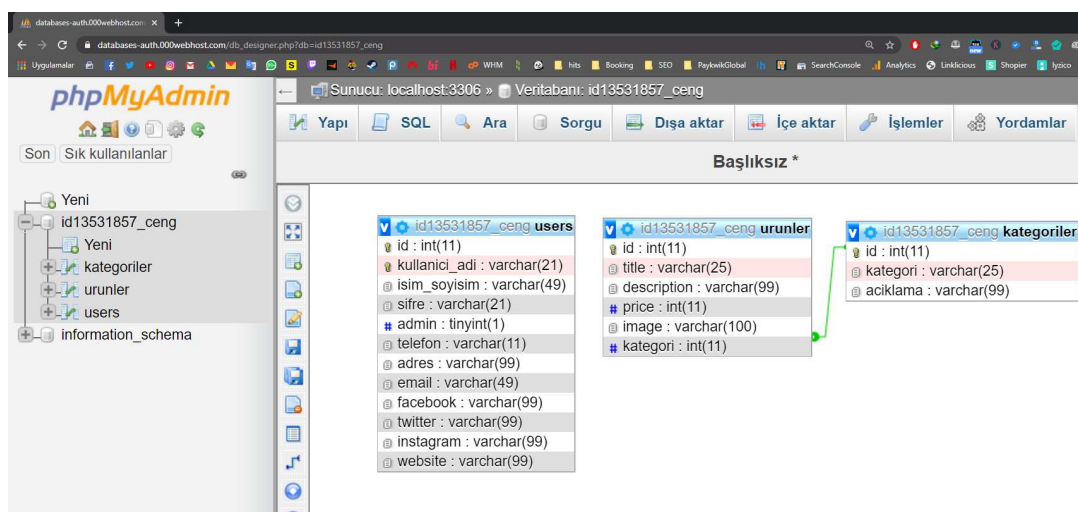
Search.php: Users can search any term here. According to the search term, this page lists all products which include the search term in title or description.

Profil.php: This page includes user information. This is accessible only if user is logged in. Otherwise, it leads to login page. Additionally, if the user is Admin, this page includes a link to admin panel.

Update.php: User information can be updated here. This page is accessible from user profile page.

Admin.php: Accessible only if the user is admin. Otherwise, it returns to the home page. Admins can add or remove product or category from this page. Also can remove or change user mod of other users.

Database Table: It is accessible in project files as database.sql file. Database has three separated tables in terms of users, products and categories. Product table has a foreign key comes from Category table.



How it Works?

- All pages include a session start function and database including page. This keeps the system connected to database and also manage the user session.
- Home page includes a loop for listing all products from database. All products have Add to Cart button which starts a session called '*sepet*', which is used for product listing in *sepet.php* to list the products on the cart. This session keeps the products on cart in array and this array listed in cart page.
- Login and Register pages only have a form for user input. Login page checks if user input has a match on database, register pages add user input into database. If user is logged in, a session called 'username' is set, so other pages will be loaded accordingly.
- Profile page list the information from database according to the username set in the session. Update page is the same but it allows the user to update the data. Profile page has Admin Panel link if the user is admin.
- Category page lists the category names from database. Using GET Method, category id is taken and another loop runs on page which creates a list of products that have same category id.
- On Cart Page, if user wants to remove an item from the cart, selected item gets removed from array. If the user wants to remove all, it unset the '*sepet*' session. As mentioned before, because this is a homework project (there is no payment, there is no real shopping), complete transaction button only shows an alert as thanks for shopping, then unset the session.
- Admin page has several functions. Firstly, if the user is not an admin, page does not load. There are several submit inputs on the page. Adding category or removing user/category/product are simply adding a new row to the database.
- In order to add new product, another function is used for managing file uploads. The uploaded file is firstly moved to images/ folder because all product listing templates include an image source as images/filename. If file input is empty, a default image is set for the new product.
- In order to provide user mod changing, submit action firstly check user mod from database. If it is admin, it updates user to normal. If it is normal user, it updates user to admin. On the user list, it also does not list the user who is logged in. So this user cannot change his/her own mod. (I changed my mod to normal user many times by a miss click).

Any Deficient?

Register page does not double check for the password. Since there is a password type input, user cannot see the password when typing, it is easy to forget or miss typing the password. This is more likely a user experience problem than a bug.

When admin adds new product, it does not check the type or the size of file. It also does not check if there is a file with same name.

On the product table, type of image column is a variation of character and has maximum length of 100 character. Uploading a file with a name that is too long for database may cause a problem.

Renaming the uploaded file with product id number can solve these two bugs.

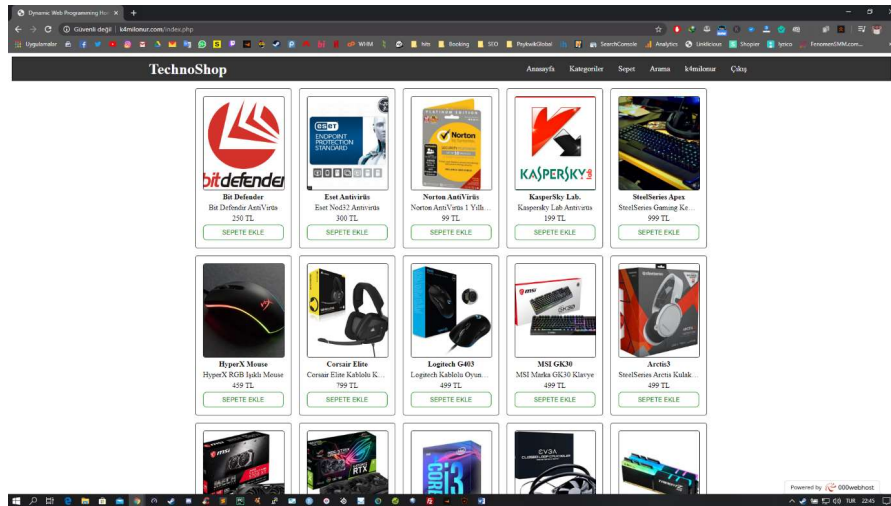
Removing a category also removes all the products that is included into that category. When I was looking for a solution, I found that InnoDB does not allow to set the product category to default. Solution may be changing the type of database.

The project is uploaded to <http://k4milonur.com>

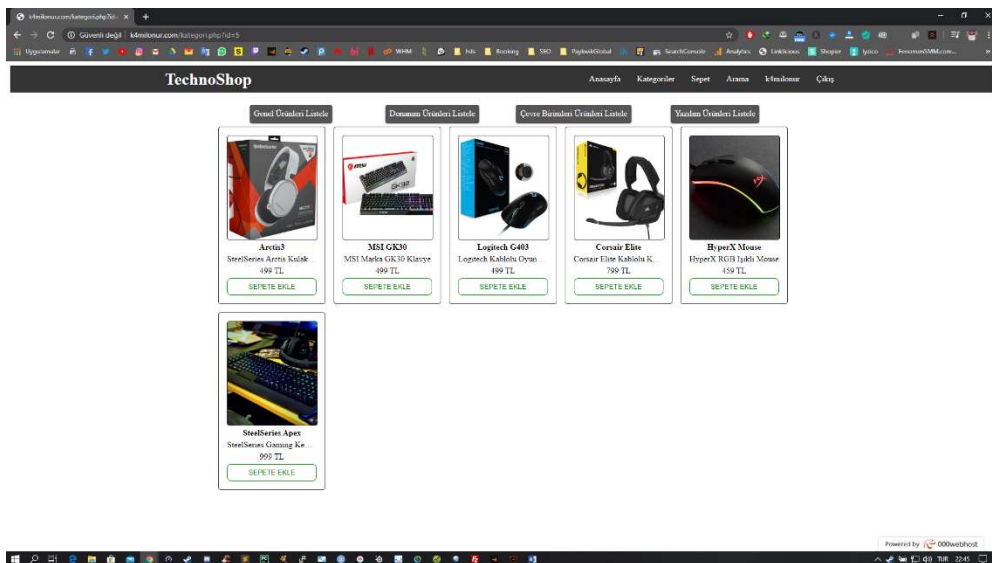
As Admin Login:

Username: gurhan

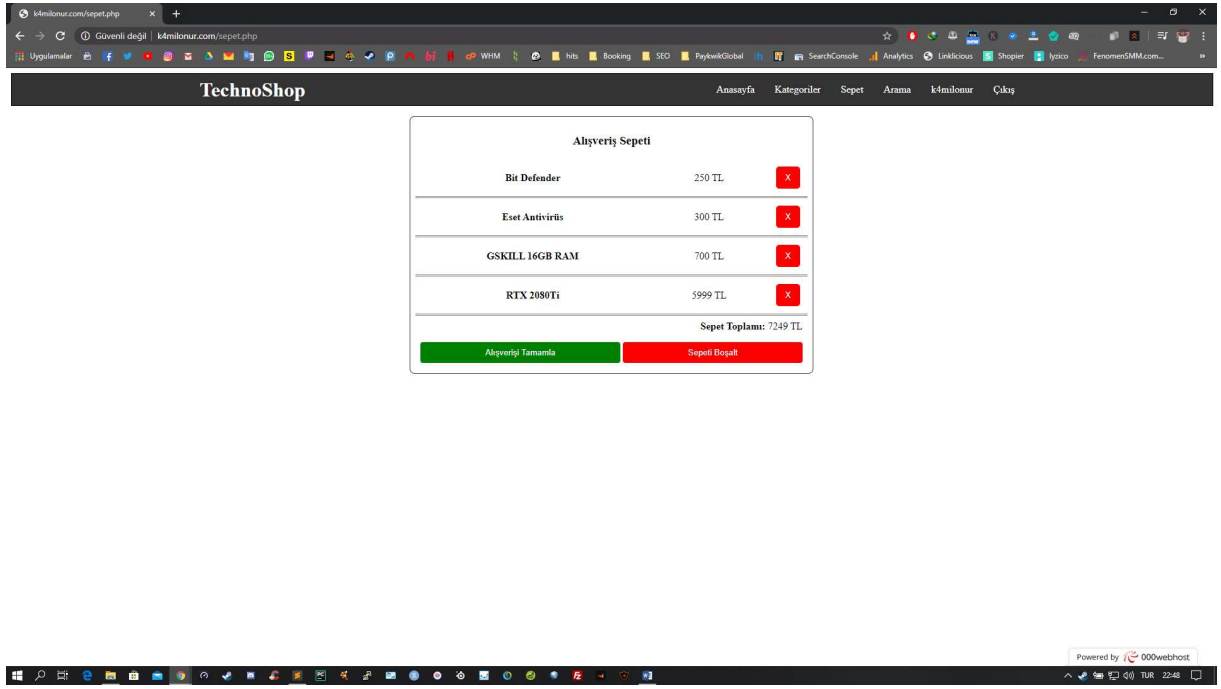
Password: 123456



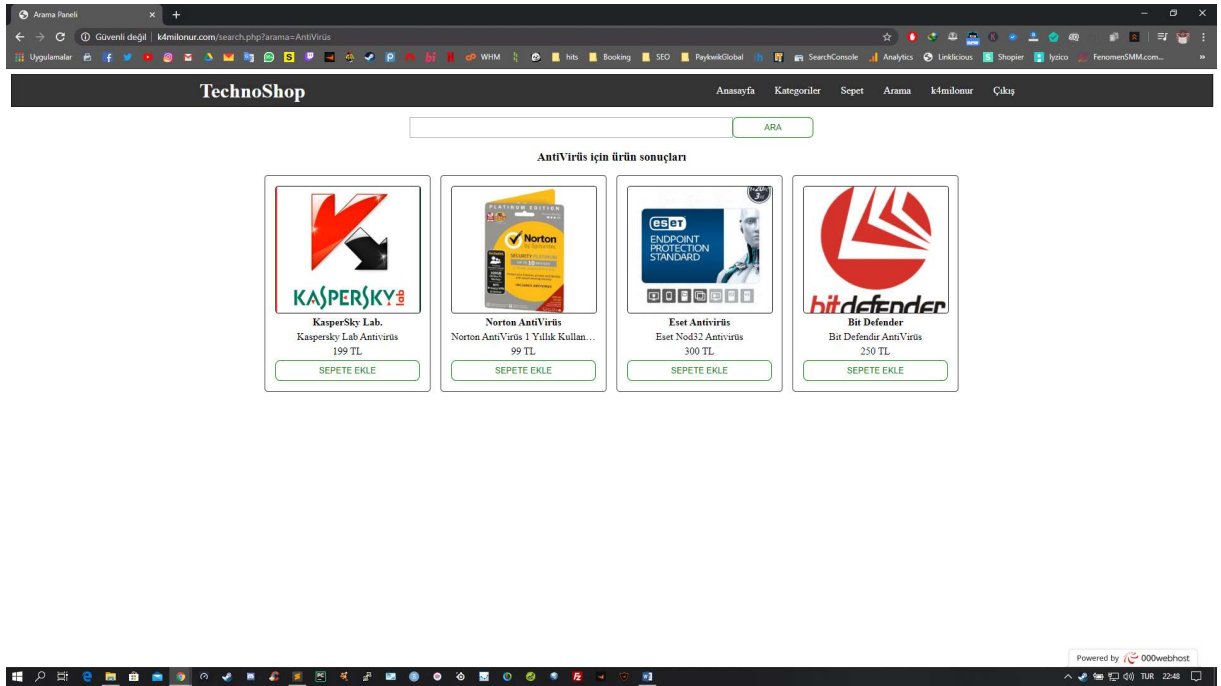
Screen Shot 1 Home Page



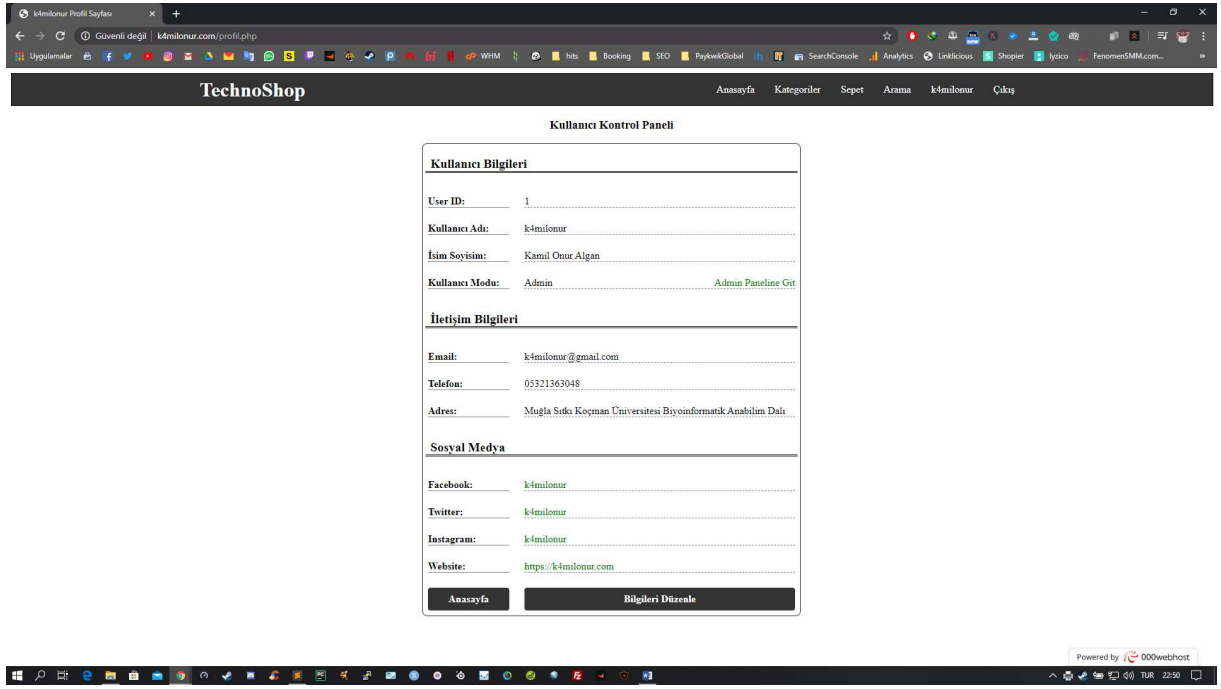
Screen Shot 2 Category Based Listing



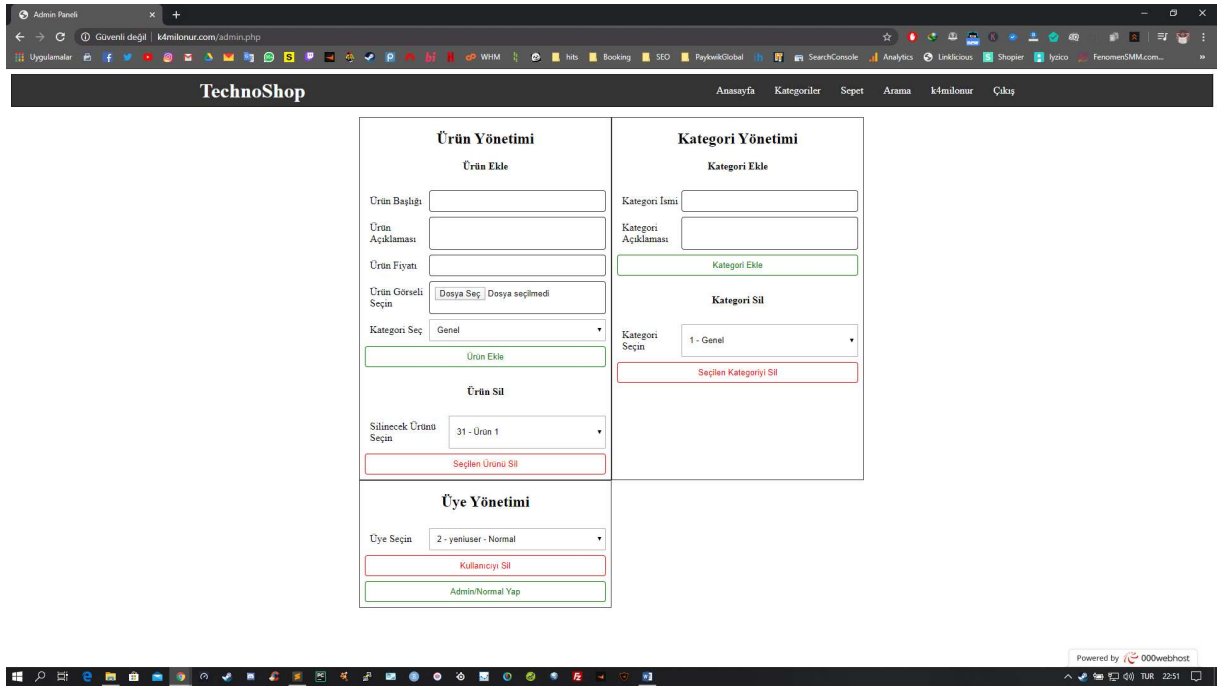
Screen Shot 3 Cart Page



Screen Shot 4 Searching Example



Screen Shot 5 Profile Page



Screen Shot 6 Admin Page