

Name: Abdullahi Shidane
Student number: 180104174

Basic information about development

Website hyperlink: <https://180104174.cs2410-web01pvm.aston.ac.uk/coursework/signIn.php>

Github coursework link: <https://github.com/abdullahi180/projects>

~~As the project is not yet completed, the following information is provided for reference only.~~

User Username: thor **Password:** User123

Public user requirements

Page name	links	requirement
List lost items for guest user.	lostItemsGuestPage.php	View basic item information (e.g. category, color, date) based on the three categories.
The register page	register.php	Register to become a registered user

Registered user requirements

Page name	links	requirement
Log in /logout	signIn.php page to log in/ and regUserHomePage.php from home page to log out.	Log in/out the system
Create page	createUser.php	Add a found item by inputting the item details
List of lost items registered user can view	lostItemsListRegUser.php	View basic item information (e.g. category, color, date) based on the three categories and click one item to see more details (e.g. photo, place, description etc.)
Requesting page	requestItemToAdmin.php	Request an item and give the reason of the request

Admin requirements

Page name	links	requirement
Login/log out	Admin/admin.php page to log in/ and admin/adminHome.php from admin home page to log out.	Log in/out the system
Admin can view and reject	Admin/requestAdmin.php	View all the requests

Name: Abdullahi Shidane
Student number: 180104174

Admin can reject/accept/ remove and delete requests	Admin/requestAdmin.php	Approve/refuse the requests
---	------------------------	-----------------------------

Security features

Page name	links	requirement
Only logged in user/ admin can have more access with different levels of access.	signIn.php admin/admin.php	Authentication/Authorization
Registrering page	register.php	Form validation/password validation
Images page	Admin/Images.php lostItemsListRegUser.php	Restrict file upload to only images
Closing brackets/html tags Good html practice Clean code	Throughout code	Handle injections(SQL/HTML)

Stretchers

Page name	links	requirement
The three different lists sorted by headings	lostItemsGuestPage.php lostItemsListRegUser.php Admin/lostItemsListAdmin.php	Allow the item tables to be sorted based on some headings
Registered users	registredUsers.php	Administrator can see all registered users on the website
Click the edit button on the table.	Admin/lostItemsListAdmin.php	Allow the administer account to edit an item's details
Approve/deny email sent	Admin/reject.php Admin/emailRejected.php Admin/emailAccepted.php Admin/acceptemail.php Admin/emailSent.php	When a request is approved or refused, an email should be sent to the users
Lost item page	Admin/lostItemsListAdmin.php	Admin can delete a record
Registred users page	admin/registredUsers.php	Admin can edit a user's email/ username

Name: Abdullahi Shidane
Student number: 180104174

Brief description of key technology of the system

To create my website, I used html and php. To add database to my website and to allow it to store data I used (phpMyAdmin) in localhost to start of with then I uploaded it on the php server I was given. I created the tables manually from scratch using html and php, and then I connected them to phpMyAdmin which stored all the data for the users, items and requests for my website, the images are stored in my coursework folder. For the login I created it using html5 and then I used php for the backend and validation to make sure the user was registered before logging in. I also made sure the registered user's detail was updated to the database.

When the website is visited the users are presented with a home page which for me is the sign in page, other pages such as about us and contact pages are available. The guest user can view the list of lost items with limited functionality by clicking the lost item button. The other pages accessible to the guest user will give the user details about the website. The about page tells the guest user what FiLo systems is about and the benefits they could gain by becoming a registered user.

Furthermore, there is also a home page which has much more functionalities available when user is logged in. When a guest user becomes a registered user, they will be able to view basic item information (e.g. category, color, date), however they are also able to see more details (e.g. photo, place, description etc.). Registered users can make item requests which are sent directly to the admin. The registered users also can create lost items and add them to the database. Admin users have ultimate control, where they can see the list in detail with all categories and they are able to create/add lost items to the database, they can also edit and delete items from the table. The requests the users make can be viewed by the admin, they can either be approved or denied. However, when a request is approved or denied it is not deleted from the table. There are two options; to remove or to delete. To remove it means the item has been approved and is ready to be removed from the database, to delete means the request has been denied but the item remains in system for someone else to request it. When an item is accepted or rejected an email is sent to the user. The admin is also able to see the names and emails for all registered users.

List your database schema

To create the database, I used PHP and HTML. I created the tables from scratch then Imported it to phpMyAdmin database system.

The user table contains primary key ID, name, email, password. This table holds all user records. When a user registered the details are saved onto this table. When a user is logging the data, he enters to log in is checked against this database.

The items table contains id as primary key, ID used to assign an id to each item. It also contains color, location, type, date, description, image. This table holds all the lists of all the lost items on the website that are added by the user as well as the admin. The requests itemid (foreign key) connects with items id(primary key).

I also created an Admin table to store the admins details, it has a id, username and password, when admin log ins they would use the details stored in this table. This table makes sure that only admins are logged in to this side of the website.

The requests table stored all the requests a user makes, it has a primary key id, item id, username, email and reason. When a user enters the details for the item it is sent to the admin, the data is also stored in the request table.