ClothApp

ClothApp offers a search engine for clothes from multiple Israeli fashion stores. For example, a shopper would like to find a red dress, size M and up to 200 NIS.

Using ClothApp, the shopper can easily find the item he wants and save a lot of time!

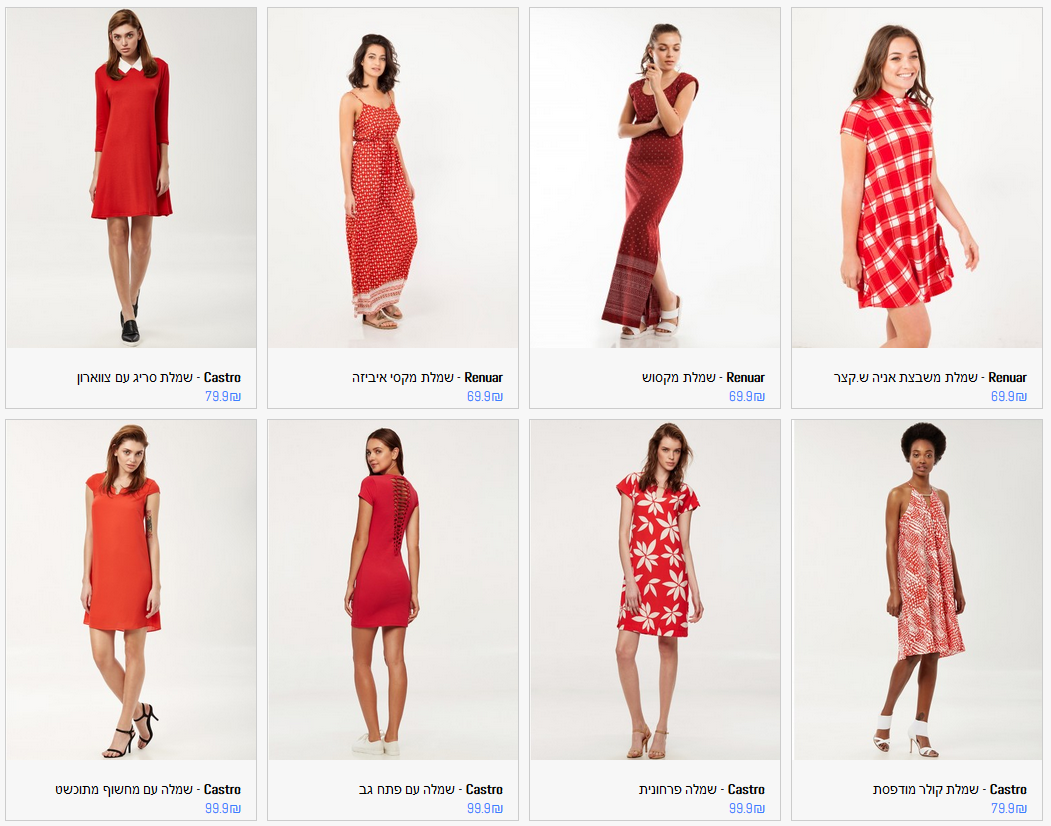
**User Interface**

Clothapp has an easy to use UI which allows the shopper to find the exact item he would like to purchase. The UI is responsive and therefore ClothApp fits every resolution and device, including mobile phones. By default, all clothing items would be shown. The infinite scrolling on the page offers an efficient way to browse all the results.

The user can filter the results by gender, item type, color, size, price and brand.



After filtering, only the suitable and available items will be shown. By clicking on an item the user can get further information about it and purchase it.



Demo: <https://drive.google.com/file/d/0B8vUTXh_QgDkVU1PWEFGeEZkNWM/view>

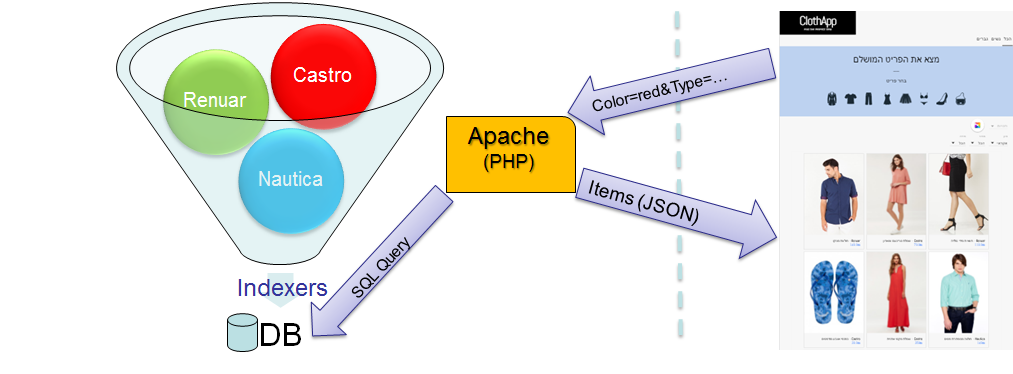
The user can contact us by sending an email to [contact.clothapp@gmail.com](mailto:contact.clothapp@gmail.com).

**High-Level Architecture**

The UI is a [single page application](https://en.wikipedia.org/wiki/Single-page_application) built from scratch using HTML, CSS and Javascript with AngularJS.

There is a complete separation between the UI and the server. The server will never print html/js code. It only provides the data in a JSON format, according to the client’s request.

The communication between the client and server is done by HTTP AJAX requests.



On the server side we use a dedicated web server located in Germany (CentOS 6.5), with Apache, PHP and MySQL database.

Our tables were defined with proper indexes to enable us to scale without any major performance hit, and to allow fast and responsive user experience.

We created a single “interface” to define clothing items, and only that interface knows the database structure. Hence, whoever adds new store indexer, should know nothing about the database structure.

Our indexers run automatically once a day (with a [cron](https://en.wikipedia.org/wiki/Cron)) to make sure the clothes are up to date. It is a few hours process but we have a hitless switch, and from the user point of view there is no downtime at all, even when the indexing process is in place. (see updateDB.php for implementation details)

**Files Details**

**Web application files:**

* **Index.html** - ClothApp main page.
* **filters.js** - includes all the functions that are responsible for filtering and retrieving relevant data from the server.
* **ng-infinite-scroll.js** - infinite scrolling for AngularJs.
* **style.css** -style sheet.
* **images/ (folder)** - contains icons for buttons.

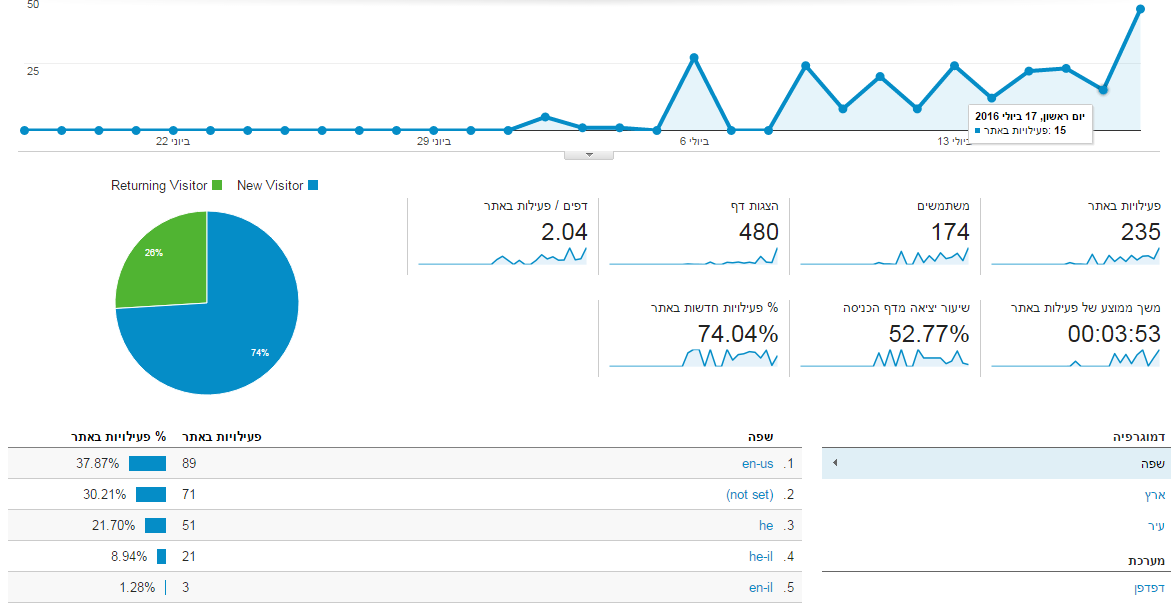
**Server side files:**

* **updateDB.php** - Runs the scripts which update the database. Calls each cloth shop indexer (currently Castro, Renuar, and Nautica), and adds all the items to temporary tables. Once all the indexers are done, it switches the old tables (used by the web application) with the temporary tables. The switch is done very fast by renaming the tables.
* **ClothItem.php** - A class that represents a cloth item. Each indexer should create such object for each cloth.
* **db.php -** contains database connection details
* **Items.php** - Used by the UI to request cloth items as json according to filters passed as uri parameters.
* **unifyCastroCategories.php** - some post processing for castro indexing process, such as unifying “t-shirts” and “shirts” to the same category.
* **unifyDB.php** - Maintenance script for the database. It organises all the different colors/sizes given by the indexers to predefined colors/sizes.
* **The following website indexers:**
  + **castroIndexer.php**
  + **nauticaIndexer.php**
  + **renuarIndexer.php**

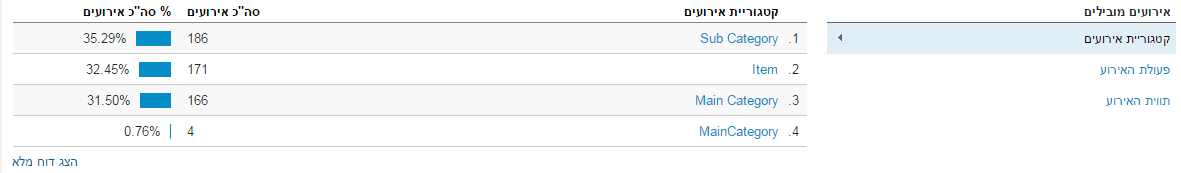
**Visitors Review (Pilot)**

We are using Google Analytics to track all incoming traffic to the site.

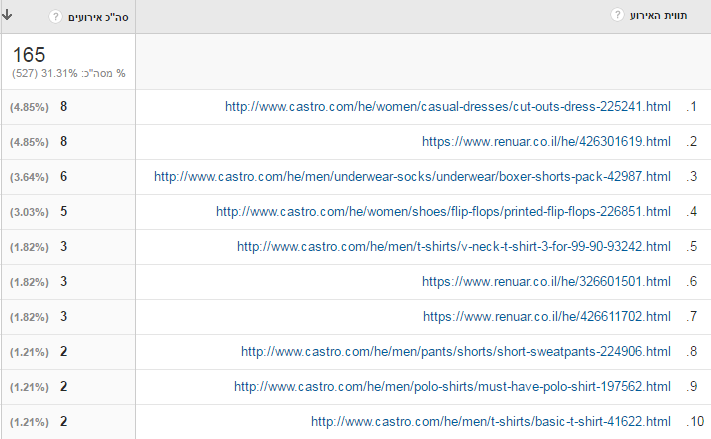
1. It allows us to view the number of visitors of each day.



1. It tracks specific buttons or links we are interested in, for example: which clothes are the most popular. We are monitoring the following tracked click events:
   1. Filtering by main category (All, Men, Women)
   2. Filtering by sub category (Shirts, dresses, skirts, etc.)
   3. Clicking on a specific cloth item



(Event categories)



(Most popular clothes)