# Bazar.com

Yazan Diab, 11924326

Ibrahim Ashour, 11923942

## Architecture:

1. Catalog servers: we have two catalog servers same operations one run port: 8000 and port: 8004, we change function update to send invalidate to front end after any update operation.

2. Order server: same previous operation do, just purchase which is send update to order server.

3. Frontend server: Firstly, we add a local file to cache requested books.
   in info function: check if requested book is exists in local cache, if it exists return it from cache, but if not exists request it from catalog server, each request on catalog server go to server1 and the second request go to server2 and so on, to do load balancing.

in search function: we do same idea of info function.

in purchase function: same idea of search and info function but the request go to order server in load balance.

And we also add invalidate function for delete a certain book from cache based on request came from catalog server

## Program Design:

We use lumen framework and 2 sqlite database in catalog server.

We build 5 Servers in lumen, two to Catalog server, each one has sqlite database to store info of books and make 3 functions for info, search , update.

And to make the two databases have same data (consistent), especially when update operations on one of them, so when update operation occur in one catalog server it must send request to frontend to clear this item if exists In its cache, and send new data to another catalog server.

## Connection

We set for each server a unique port, to front end: 8001

And for catalogs server: 8000,8004, and for order servers: 8002,8003.

Each request came to frontend and frontend send requests to catalog server or order server based on balanced two servers.

**By using lumen command to run each server:**

**php -S localhost:8001 -t public**, example for frontend

## Routes

| URI | Method | Description |
|---|---|---|
| CatalogServer/info/{id} | get | This return all info about book by book id |
| CatalogServer/search/{topic} | get | This return titles of topics |
| CatalogServer/update/{id}/{type}/{value} | put | To update cost/quantity |
| OrderServer/purchase/{id} | put | To purchase book |
| /info/{id} | get | Frontend – get info |
| /search/{topic} | get | Frontend – search |
| /purchase/{id} | put | Frontend – purchase |