# Toy Shop Application (version 1)

## Introduction

This application is a demo application. It is an E-shop site for toys. The user can browse different type of toys and can add some of them to his shop cart then he can submit the order. The user also can see a statistic about the annual selling trend and see contact us information with Google map.

The site is created by HTML5, JavaScript, Angular Js and Nod.js and is deployed in a public cloud.

The application uses dummy data and doesn’t save or get the data from persistent store Like Nosql database.

## Functional Description

There are 3 scenarios that can be done by the customer (ordering toys scenario, trend Chart, contact us). Here are the description of each scenario

|  |  |  |  |
| --- | --- | --- | --- |
| Scenario Name | Main target | steps | Alternatives |
| Order toys scenario | Customer needs to buy some toys | 1. Customer accesses the application via site link. 2. Customer sees products with category “baby” 3. Customer presses “Add to cart” link for any toy. 4. Customer accesses “shopping cart” tab. 5. Customer reviews the order and press submit button 6. System sends order to backend 7. Backend prints the order in log 8. System shows customer message “Order has been submitted” | If customer needs to buy product with category “child”. He will do the same scenario but with a following different.  Instead of point 2 in “Order toys” scenario customer will presses link “child toy” then system will displays the products with type “child” then customer continue the same steps in the main scenario |
| Trend Chart scenario | Customer needs to see annual selling trend. It is a chart that displays the amount of selling per month. The x axe represents Time and The Y axe represents the selling by dollars | 1. Customer presses the tab “statistics”. 2. System requests the data from backend 3. Backend replies by the data for past 12 months. 4. System displays the chart |  |
| Contact us Scenario | Customer needs to see contact information | 1. Customer presses the tab “contact us” 2. Customer sees static page contains google map and some information. |  |

### High level Design

The system contains two main parts frontend part and backend part. Frontend part handles user interactions and uses backend to get the data and submit the order.

Here are sequence diagram for ordering toy scenario



### Low level Design

Frontend of the application is built by HTML5, javascript and AngluarJs and backend is built by Nodejs. The Two parts are deployed as one node.js project.

Here is the package structure of the project

|  |  |
| --- | --- |
| Artifact | Description |
| Public folder | Contains all frontend HTML pages and javascripts |
| node\_modules folder | Contains node js dependencies ( mainly for express module) |
| Server.js | This file contains main code of backend |

Public folder contains all frontend artifacts. here is what are inside.

|  |  |
| --- | --- |
| Artifact | Description |
| css | Contains main.css file which controls all the UI look and feel |
| Images | Contains images uses in the application |
| Scripts | Contains three files config.js : contains configuration information  Routing.js: contains routing map between pages  Main.js: contains main angluarJs logic , it contains controllers and services |
| Home.htm | Main page template |
| Products.htm | It displays the products |
| Shopcart.htm | It displays the shop cart of customers |
| Contactus.htm | It displays the contact information |
| Statistics.htm | It displays the Trend charts |

### Frontend

Most of Angluar js features ( MVC, routing, dependency injection and more ) are used in the application. The main js file that contains Angluar.js code is main.js . It contains controller for each page and contains three services. Here are controllers’ descriptions

|  |  |
| --- | --- |
| Controller | description |
| mainController | contains the global variables and functions of the application |
| productsController | contains variable and functions of products page |
| shopcartController | contains variable and functions of shopcart page |
| contactusController | contains variable and functions of contactus page |
| statisticsController | contains variable and functions of statistics page |

Here are services table

|  |  |
| --- | --- |
| userService | This service contains methods that handle users. This version assumes predefine user called "ayman" . there are no real user management like login |
| cartService | This service contains methods to get cart , delete it and method to add item to cart |
| statisticsService | this service contains method to get statistic from backend |

In this version, there is no module to handle login and permissions for the users. Also only one user is defined.

### Backend

Backend code exists in file called server.js. It just prints the submitted order and return back a predefine products and statistics list stored in memory to front end. The backend provides 3 Restful services to the Front end.

Here are Restful services table

|  |  |  |
| --- | --- | --- |
| Service URL | Description | HTTP method |
| /product | get products list | GET |
| /statistics | get statistics of the selling | GET |
| /cart | submit the shop cart | PUT |

### Installing the Application

To install the application in a local machine, follow these steps.

1. Install Node.js in your machine
2. Open public/scripts/config.js and change the serverURL parameter to <http://localhost:8080> . The port should be “8080” otherwise you need to change the port in server.js file.
3. Access the application folder and run the server by this command “node server.js”

## Cloud publishing

This version is published in cloud9 (public cloud site). You can access it via this link

<https://toyshop-aymanelshayeb.c9users.io/home.htm>

You can find the code in this Github link [git@github.com:aymanElshayeb/projects.git](mailto:git@github.com:aymanElshayeb/projects.git) under Nodejs/toyshop folder .

The code site is <https://github.com/aymanElshayeb/projects> under NodeJs/toyshop