

# A7: High-level architecture. Privileges. Web resources specification

---

## 1. Overview

Identify and overview the modules that will be part of the application.

### **M01: Authentication and Individual Profile**

Web resources associated with user authentication and individual profile management, includes the following system features: login/logout, registration, view and edit personal profile information.

---

### **M02: Products**

Web resources associated with available items, includes the following system features: items list, search and filter, view, edit and delete items.

---

### **M03: Reviews and Wish list**

Web resources associated with reviews and wish list, includes the following system features: add comment, answer a comment, list and delete comments; add and remove items from the wish list, list a customer's wish list.

---

### **M04: Purchase**

Web resources associated with purchases, includes the following system features: view purchase history, make a purchase, list a shopping cart, update loyalty points.

---

### **M05: User**

Web resources associates with user management, specifically: moderate comments

**Administration and Static pages** and profile pictures, ban users from using the comment section, Web resources with static content are associated with this module: configurator, about, contact, services, faq and 404.

## 2. Permissions (Define the permissions used by each module, necessary to access its data and features)

This section defines the permissions used in the modules to establish the conditions of access to resources.

<b>VISI</b>	Visitor	Group of users without privileges
<b>CUST</b>	Customer	Group of users that can update their profiles and have privileges to buy items
<b>MOD</b>	Moderator	Group of moderators that can delete comments and ban users from the comment section
<b>ADMIN</b>	Administrator	Group of administrators that can update the product database and manage moderators

## 3. Modules

This section documents each web resource of each module, indicating the URL, HTTP methods, the possible parameters, interfaces with the user (referring to the A3 artefact) — or JSON responses in the event of APIs.

Module M01: Authentication and Individual Profile

### Endpoints of Module Authentication and Individual Profile

These are the endpoints available in the Authentication and Individual Profile Module.

- R101: Login Form /login
- R102: Login Action /login
- R103: Logout Action /logout
- R104: Register Form /register
- R105: Register Action /register
- R106: View Profile /users/{id}
- R107: Edit Profile Form /users/{id}/edit
- R108: Edit Profile Action /users/{id}/
- R109: Password Reset Form /password/reset
- R110: Password Reset Action /password/reset

#### **R101: Login Form**

<b>URL</b>	/login
<b>Description</b>	Page with a form to login to a user account.
<b>Method</b>	GET
<b>UI</b>	UI12
<b>SUBMIT</b>	R102
<b>Permissions</b>	VISI

#### **R102: Login Action**

<b>URL</b>	/login
<b>Description</b>	This web resource logs the user into the system. Redirects to the user profile page on success and the login form on failure.
<b>Method</b>	POST

<b>Request Body</b>	+username:String	Username
	+password:String	Password
<b>Redirects</b>	R106	Success
	R101	Error
<b>Permissions</b>	VISI	

### R103: Logout Action

<b>URL</b>	/logout	
<b>Description</b>	This web resource logs the authenticated user out of the system. Redirects to the homepage on success.	
<b>Method</b>	POST	
<b>Request Body</b>	+username:String	Username
<b>Redirects</b>	R106	
<b>Permissions</b>	CUST, MOD, ADMIN	

### R104: Register Form

<b>URL</b>	/register	
<b>Description</b>	Page with a form to register a new user account.	
<b>Method</b>	GET	
<b>UI</b>	UI13	
<b>SUBMIT</b>	R105	
<b>Permissions</b>	VISI	

## R105: Register Action

<b>URL</b>	/register	
<b>Description</b>	This web resource inserts the user into the system. Redirects to the user profile page on success and the register form on failure.	
<b>Method</b>	POST	
<b>Request Body</b>	+username:String	Username
	+password:String	Password
	+confirmPassword:String	Password Confirmation
	+firstName:String	First Name
	+lastName:String	Last Name
	+email:String	Email
	+address:String	Adress
<b>Redirects</b>	R106	Success
	R104	Error
<b>Permissions</b>	VISI	

## R106: View Profile

<b>URL</b>	/users/{id}	
<b>Description</b>	Shows the user's individual profile page	
<b>Method</b>	GET	
<b>Parameters</b>	+id:Integer	user table primary key

<b>UI</b>	UI14
<b>Permissions</b>	CUST

#### R107: Edit Profile Form

<b>URL</b>	/users/{id}/edit	
<b>Description</b>	Shows the user's individual profile page with editable fields.	
<b>Method</b>	GET	
<b>Parameters</b>	+id:Integer	user table primary key
<b>UI</b>	UI14	
<b>SUBMIT</b>	R108	
<b>Permissions</b>	CUST	

#### R108: Edit Profile Action

<b>URL</b>	/register	
<b>Description</b>	Web resource that changes user profile info based on the input received. Redirects to the user profile on success and edit profile page on failure.	
<b>Method</b>	POST	
<b>Request Body</b>	+password:String	New Password
	+confirmPassword:String	New Password Confirmation
	+firstName:String	New First Name
	+lastName:String	New Last Name

<b>Redirects</b>	+email:String	New Email
	+address:String	New Address
	+picture:String	New profile picture
	R106	Success
	R107	Error
<b>Permissions</b>	CUST	

#### R109: Password recovery Form

<b>URL</b>	/password/reset	
<b>Description</b>	Page with a form to request a token to reset the password.	
<b>Method</b>	GET	
<b>UI</b>	UI12	
<b>SUBMIT</b>	R110	
<b>Permissions</b>	VISI	

#### R110: Password recovery Action

<b>URL</b>	/password/reset	
<b>Description</b>	Web resource that sends a reset password link to the specified email.	
<b>Method</b>	POST	
<b>Request Body</b>	+email:String	User email
	UI01	Success

**Redirects**

R109

Error

**Permissions**

VISI

## Module M02: Products

**Endpoints of Product pages**

- R201: Homepage /home
- R202: Search product API /api/products
- R203: Compare Products products/compare
- R204: List Products with filter /products/
- R205: Single Product /product/{id}
- R206: Edit Product Form /product/{id}/edit
- R207: Edit Product Action /product/{id}
- R208: Add Product Form /product/create
- R209: Add Product Action /product/
- R210: Delete Product Action /product/{id}

**R201: Homepage**

<b>URL</b>	/home
<b>Description</b>	Homepage
<b>Method</b>	GET
<b>AJAX Calls</b>	R202
<b>UI</b>	UI01
<b>Permissions</b>	VISI

**R202: Search product API**

<b>URL</b>	/api/products
------------	---------------



<b>Description</b>	This web resource searches the system for products and returns the results.	
<b>Method</b>	GET	
<b>Parameters</b>	+categoryID:Integer	Category
	+productBrand:String	Product Brand
	+minPrice:Integer	Price Lower Bound
	+maxPrice:String	Price Higher Bound
	+productAvailability:boolean	Product Availability
<b>Response Body</b>	JSON202	
<b>Permissions</b>	VISI	

### R203: Compare Selected Products

<b>URL</b>	/products/compare
<b>Description</b>	Product Comparator Page
<b>Method</b>	GET
<b>UI</b>	UI07
<b>Permissions</b>	VISI

### R204: Products with filter

<b>URL</b>	/products
<b>Description</b>	Page with a list of filtered products
<b>Method</b>	GET

<b>Parameters</b>	?categoryID:Integer	Category
	?productBrand:String	Product Brand
	?minPrice:Integer	Price Lower Bound
	?maxPrice:String	Price Higher Bound
	?productAvailability:boolean	Product Availability
<b>AJAX Calls</b>	R202	
<b>UI</b>	UI06	
<b>Permissions</b>	VISI	

#### R205: Single Product

<b>URL</b>	/product/{id}
<b>Description</b>	Product Page
<b>Method</b>	GET
<b>Parameters</b>	+productId:Integer    Product Identifier
<b>UI</b>	UI08
<b>Permissions</b>	VISI

#### R206: Edit Product Form

<b>URL</b>	/product/{id}/edit
<b>Description</b>	Product page with editable fields
<b>Method</b>	GET
<b>Parameters</b>	+id:Integer    Product Identifier
<b>UI</b>	UI17

**SUBMIT**

R207

**Permissions**

ADMIN

**R207: Edit Product Action****URL**

/product/{id}

**Description**

Web resource that changes a product's info based on the input received. Redirects to the product page on success and edit product page on failure.

**Method**

PUT

**Request  
Body**

?title:String

Product Title

?

description:String

Product's description

?price:String

Product's price

?discPrice:String

Product's discount price

?

attribute:String[]

Values of the attributes of the product

?picture:String[]

Paths to the product pictures

**Redirects**

R205

Success

R206

Error

**Permissions**

ADMIN

**R208: Add Product Form****URL**

/product/create

**Description**

Product page with empty editable fields

<b>Method</b>	GET
<b>UI</b>	UI17
<b>SUBMIT</b>	R209
<b>Permissions</b>	ADMIN

#### R209: Add Product Action

<b>URL</b>	/product/	
<b>Description</b>	Web resource that adds a product and it's info, based on the input received, to the database. Redirects to the product page on success and add product page on failure.	
<b>Method</b>	POST	
<b>Request Body</b>	+title:String	Product Title
	+description:String	Product's description
	+price:String	Product's price
	+discPrice:String	Product's discount price
	+attribute:String[]	Values of the attributes of the product
	+picture:String[]	Paths to the product pictures
<b>Redirects</b>	R205	Success
	R208	Error
<b>Permissions</b>	ADMIN	

#### R210: Delete Product Action

<b>URL</b>	/product/{id}	
<b>Description</b>	This web resource removes the work from the system. Redirects to the Homepage on success and the View Work page on failure.	
<b>Method</b>	DELETE	
<b>Parameters</b>	+id:Integer	Product Identifier
	200 OK	Product was successfully deleted.
<b>Returns</b>	400 Bad Request	Error. Error message is specified via a HTTP header.
	404 Not Found	Error. No item with the specified primary key exists.
<b>Permissions</b>	ADMIN	

## Module M03: Reviews and Wish list

### Endpoints of Reviews and Wish list

These are the endpoints available in the Reviews and Wish list Module.

- R301: Write a comment form /product/{id}
- R302: Write a comment action /product/{id}/comment
- R303: Flag comment /comment/{id}/flag
- R304: List comments /product/{id}/comments
- R305: Delete comment /comment/{id}
- R306: Add an item to the wish list /users/{id}/favorites
- R307: Remove an item from the wish list /users/{id}/favorites
- R308: List a customer's wish list /users/{id}/favorites

#### R301: Write a comment form

**URL**            /product/create

<b>Description</b>	Product page comment section form
<b>Method</b>	GET
<b>UI</b>	UI09
<b>Parameters</b>	+id:integer    Product to be commented identifier
<b>SUBMIT</b>	R302
<b>Permissions</b>	CUST

### R302: Write a comment action

<b>URL</b>	/product/{id}/comment	
<b>Description</b>	This web resource creates a comment based on the input received.	
<b>Method</b>	POST	
<b>Parameters</b>	+id: integer	Product id.
<b>Request Body</b>	+user_id: text	User that comments.
	+parent_id: integer	Parent comment or new comment if null.
<b>Redirects</b>	UI09	Success
	UI09	Error
<b>Permissions</b>	CUST	

### R303: Flag comment

<b>URL</b>	/comment/{id}/flag
<b>Description</b>	Flag comment.

<b>Method</b>	POST
<b>Parameters</b>	+id: integer    Comment id.
<b>Permissions</b>	CUST, MOD

#### R304: List comments

<b>URL</b>	/product/{id}/comments
<b>Description</b>	List comments of a product.
<b>Method</b>	GET
<b>Parameters</b>	+id: integer    Product id.
<b>Response body</b>	JSON304
<b>Permissions</b>	VISI, CUST, MOD, ADMIN

#### R305: Delete comment

<b>URL</b>	/comment/{id}
<b>Description</b>	Delete comment of a product.
<b>Method</b>	DELETE
<b>Parameters</b>	+id: integer    Comment id.
<b>Permissions</b>	MOD

#### R306: Add an item to the wish list

<b>URL</b>	/users/{id}/favorites
<b>Description</b>	Choose a product as a favorite.
<b>Method</b>	PUT

<b>Parameters</b>	+id: text	Customer id.
<b>Request Body</b>	+product_id: integer	Product selected as a favorite.
<b>Returns</b>	200 OK	Product was successfully added as a favorite.
	400 Bad Request	Error. Error message is specified via a HTTP header.
	404 Not Found	Error. No item with the specified primary key exists.
<b>Permissions</b>	CUST	

#### R307: Remove an item from the wish list

<b>URL</b>	/users/{id}/favorites	
<b>Description</b>	Remove an item from the wish list.	
<b>Method</b>	DELETE	
<b>Parameters</b>	+id: text	Customer id.
<b>Permissions</b>	CUST	

#### R308: List a customer's wish list

<b>URL</b>	/users/{id}/favorites	
<b>Description</b>	List a customer's wish list.	
<b>Method</b>	GET	
<b>Parameters</b>	+id: text	Customer identifier.
<b>Response body</b>	JSON202	



<b>UI</b>	UI16
<b>Permissions</b>	CUST

## Module M04: Purchase

### Endpoints of Purchase

These are the endpoints available in the Purchase Module.

- R401: View purchase history /users/{id}/purchasehistory
- R402: Make a purchase form /users/{id}/purchase
- R403: Make a purchase action /users/{id}/purchase/checkout
- R404: List a shopping cart /users/{id}/shoppingCart
- R405: Update loyalty points /users/{id}/loyaltypoints

#### R401: View purchase history

<b>URL</b>	/users/{id}/purchasehistory	
<b>Description</b>	Get all my purchase history.	
<b>Method</b>	GET	
<b>Parameters</b>	+id: integer	Username
<b>UI</b>	UI15	
<b>Permissions</b>	CUST	

#### R402: Make a purchase form

<b>URL</b>	/users/{id}/purchase	
<b>Description</b>	Page with a form to input required information to checkout a purchase.	
<b>Method</b>	GET	

<b>Parameters</b>	+id:integer	Identifier for the user making the purchase
<b>SUBMIT</b>	R403	
<b>UI</b>	UI11	
<b>Permissions</b>	CUST	

#### R403: Make a purchase action

<b>URL</b>	/users/{id}/purchase/checkout	
<b>Description</b>	Purchase the current shopping cart.	
<b>Method</b>	POST	
<b>Parameters</b>	+id: integer	Customer Username.
<b>Request Body</b>	+productId:integer[]	List of products purchased.
<b>Redirects</b>	UI15	Success
	R402	Error
<b>Permissions</b>	CUST	

#### R404: List a shopping cart

<b>URL</b>	/shoppingCart	
<b>Description</b>	List a shopping cart.	
<b>Method</b>	GET	
<b>Parameters</b>	+id: integer	Customer Username.
<b>UI</b>	UI10	

**Permissions** VISI, CUST

#### R405: Update loyalty points

<b>URL</b>	/users/{id}/loyaltypoints	
<b>Description</b>	Web resource used to update a customers loyalty points amount	
<b>Method</b>	POST	
<b>Parameters</b>	+id: integer	Customer id.
<b>Request Body</b>	+loyaltyPoints: integer	Number of loyaltyPoints to add.
<b>Returns</b>	200 OK	The loyalty points was successfully updated.
	400 Bad Request	Error. Error message is specified via a HTTP header.
	404 Not Found	Error. No item with the specified primary key exists.
<b>Permissions</b>	CUST	

### Module M05: User Administration and Static pages

#### Endpoints of User Administration and Static pages

These are the endpoints available in the User Administration and Static pages Module.

- R501: Configurator /configurator
- R502: About /about
- R503: Contact /contact
- R504: FAQ /faq

- R505: 404 /404
- R506: Product Overview /admin/products
- R507: Moderators Overview /admin/moderators
- R508: Flagged Comments View /moderator/comments
- R509: Banned Users /moderator/banned
- R510: Moderator Homepage /moderator
- R511: Administrator Homepage /administrator

### R501: Configurator

<b>URL</b>	/configurator
<b>Description</b>	Get configurator page.
<b>Method</b>	GET
<b>UI</b>	UI05
<b>Permissions</b>	VISI

### R502: About

<b>URL</b>	/about
<b>Description</b>	Get about page.
<b>Method</b>	GET
<b>UI</b>	UI02
<b>Permissions</b>	VISI

### R503: Contact

<b>URL</b>	/contact
<b>Description</b>	Get contact page.

<b>Method</b>	GET
<b>UI</b>	UI03
<b>Permissions</b>	VISI

#### R504: FAQ

<b>URL</b>	/faq
<b>Description</b>	Get faq page.
<b>Method</b>	GET
<b>UI</b>	UI04
<b>Permissions</b>	VISI

#### R505: 404

<b>URL</b>	/404
<b>Description</b>	Get 404 page.
<b>Method</b>	GET
<b>UI</b>	UI20
<b>Permissions</b>	PUB

#### R506: Product Overview

<b>URL</b>	/admin/products
<b>Description</b>	Web resource used to get all products for an administrator view.
<b>Method</b>	GET

<b>Response body</b>	JSON202
----------------------	---------

<b>Permissions</b>	ADMIN
--------------------	-------

#### R507: Moderators Overview

<b>URL</b>	/admin/moderators
------------	-------------------

<b>Description</b>	Web resource used to get all active moderators for an administrator view.
--------------------	---

<b>Method</b>	GET
---------------	-----

<b>Response body</b>	JSON507
----------------------	---------

<b>Permissions</b>	ADMIN
--------------------	-------

#### R508: Flagged Comments View

<b>URL</b>	/moderator/comments
------------	---------------------

<b>Description</b>	Flagged Comments View.
--------------------	------------------------

<b>Method</b>	GET
---------------	-----

<b>Response body</b>	JSON508
----------------------	---------

<b>Permissions</b>	MOD
--------------------	-----

#### R509: Banned Customers

<b>URL</b>	/moderator/banned
------------	-------------------

<b>Description</b>	Banned Customers.
--------------------	-------------------

<b>Method</b>	GET
---------------	-----

<b>Response body</b>	JSON509
<b>Permissions</b>	MOD

#### R510: Moderator Homepage

<b>URL</b>	/moderator
<b>Description</b>	Moderator Homepage.
<b>Method</b>	GET
<b>UI</b>	UI18
<b>AJAX Calls</b>	R508, R509
<b>Permissions</b>	MOD

#### R511: Administrator Homepage

<b>URL</b>	/administrator
<b>Description</b>	Administrator Homepage.
<b>Method</b>	GET
<b>UI</b>	UI19
<b>AJAX Calls</b>	R506, R507
<b>Permissions</b>	ADMIN

## 4. JSON/XML Types

Document the JSON or XML responses that will be used by the web resources.

The structure of the JSON formatted answers must be documented as illustrated below.

## JSON202: List of products

```
{
  "product": [
    {
      "sku":
      "title":
      "idCat":
      "price":
      "discountPrice":
      "rating":
      "stock":
    },
    {
      "sku":
      "title":
      "idCat":
      "price":
      "discountPrice":
      "rating":
      "stock":
    }
  ]
}
```

## JSON304: List comments

```
{
  "comments": [
    {
      "id":
      "user_id":
      "date":
      "description":
    },
  ],
}
```



```
{
  "id":
  "user_id":
  "date":
  "description":
}
]
"answers": [
{
  "parent_id":
  "child_id":
},
{
  "parent_id":
  "child_id":
}
]
}
```

## JSON507: Moderators Overview

```
{
  "moderator": [
    {
      "username":
      "email":
      "joinDate":
      "picture":
    },
    {
      "username":
      "email":
      "joinDate":
      "picture":
    }
  ]
}
```

```
}
```

## JSON508: Flagged comments

```
{
  "comments": [
    {
      "id":
      "user_id":
      "date":
      "description":
    },
    {
      "id":
      "user_id":
      "date":
      "description":
    }
  ]
}
```

## JSON509: Banned customers

```
{
  "banned": [
    {
      "username":
      "bannedDate":
      "moderator":
    },
    {
      "username":
      "bannedDate":
      "moderator":
    }
  ]
}
```

```
}  
]  
}
```

## Web resources descriptors Do not include on the final artefact

- URL - Resource identifier, following the RESTful resource naming conventions
- Description - Describe the resource, when it's used and why
- UI - Reference to the A3 user interface used by the resource
- SUBMIT - Reference to the actions/requests integrated with the resource
- Method - HTTP request Method
- Parameters - Information that is sent through the URL, by a query string or path
- Request Body - Data associated and transmitted with each request
- Returns - HTTP code returned from a request
- Response Body - Data sent from the server, in response to a given request
- Permissions - Required permissions to access the resource

## Revision history

---

GROUP1723, 11/04/2018

Ana Cláudia Fonseca Santos, up200700742@fe.up.pt

Eduardo de Mendonça Rodrigues Salgado Ramos,  
up201505779@fe.up.pt

Mariana Lopes da Silva, up201506197@fe.up.pt

Xavier Reis Fontes, up201503145@fe.up.pt