Coupon System project : part 2

Chen Alkabets / Yonatan Azgad

1. Overview

This part introduce a REST API to access the Coupon System. the API provides access to 3 users :

1. Customer
2. Company
3. Admin

The web application will run inside a Tomcat servlet container and a REST client side is implemented as Single Page Application using AngularJS.

- The project shipped as csee5.war file.
- Required *.jar :
    - cs.jar
    - mysql-connector-java-5.1.35-bin.jar
    - JTattoo-1.6.11.jar (used for old Swing UI)
    - All resteasy jars

2. System Components

- Tomcat : The web container, once the application war deployed a deployment descriptor web.xml loads :
  - WebApp servlet listener
  - CouponSystemLoader : Helper servlet that loads the WEB-INF\cs.properties and instantiate the Coupon System singleton.

- REST Filter : a ServletFilter that enforce the user to login before accessing resources. If “login” keyword exist in the client URL request - the filter is bypassed since the client try to login, if “login” keyword does not exist - the filter assumes the client is trying to get data from resources and it checks if HttpSession and Facade object available. in case HttpSession and Facade object are not found - the client is redirected to index.html (main page)

- WebAppListener : Used to log and track HttpSession events

- JAX-RS Resources : Customer / Company / Admin Resources that provides common operation APIs for CRUD operations and Coupon management.

- Single Page Application : HTTP web interface enables the clients to connect to the system and do operations.

3. Directory Structure

```
│   index.html
│
├───css
│       buttons.css
│       clientname.css
│       leftmenu.css
│       page.css
│       table.css
│       textfields.css
│
├───js
│       angular.min.js
│       main.js
│
├───META-INF
│       MANIFEST.MF
│
└───WEB-INF
    │   angular2.jpg
    │   couponsystem.pdf
    │   couponsystem_part2.pdf
    │   cs.properties
    │   csee.doc
    │   web.xml
    │
    ├───classes
    │   └───com
    │       └───jbp
    │           └───ee
    │               ├───main
    │               │       CouponSystemLoader.class
    │               │       CouponSystemLoader.java
    │               │       RestFilter.class
    │               │       RestFilter.java
    │               ├───rest
    │               │       AdminResource.class
    │               │       AdminResource.java
    │               │       Client.class
    │               │       Client.java
    │               │       CompanyResource.class
    │               │       CompanyResource.java
    │               │       CustomerResource.class
    │               │       CustomerResource.java
    │               │       ManageResource.class
    │               │       ManageResource.java
    │               │       RestApplication.class
    │               │       RestApplication.java
    │               └───utils
    │                       ClientDTO.class
    │                       ClientDTO.java
    │                       CSExceptionHandler.class
    │                       CSExceptionHandler.java
    │                       WebappListener.class
    │                       WebappListener.java
    │
    └───lib
            activation-1.1.jar
            aopalliance-1.0.jar
            apache-mime4j-0.6.jar
            async-http-servlet-3.0-3.0.9.Final.jar
            bcmail-jdk16-1.46.jar
            bcprov-jdk16-1.46.jar
            classmate-0.8.0.jar
```

commons-codec-1.6.jar
commons-io-2.1.jar
commons-logging-1.1.1.jar
cs.jar
FastInfoset-1.2.7.jar
guice-3.0.jar
hibernate-jpa-2.0-api-1.0.1.Final.jar
hibernate-validator-5.0.1.Final.jar
httpclient-4.2.6.jar
httpcore-4.2.5.jar
infinispan-core-5.1.6.FINAL.jar
istack-commons-runtime-2.16.jar
jackson-annotations-2.4.1.jar
jackson-core-2.4.1.jar
jackson-core-asl-1.9.12.jar
jackson-databind-2.4.1.jar
jackson-jaxrs-1.9.12.jar
jackson-jaxrs-base-2.4.1.jar
jackson-jaxrs-json-provider-2.4.1.jar
jackson-mapper-asl-1.9.12.jar
jackson-module-jaxb-annotations-2.4.1.jar
jackson-xc-1.9.12.jar
javax.el-3.0.0.jar
javax.inject-1.jar
javax.json-1.0.3.jar
javax.json-api-1.0.jar
jaxb-api-2.2.7.jar
jaxb-core-2.2.7.jar
jaxb-impl-2.2.7.jar
jaxrs-api-3.0.9.Final.jar
jboss-annotations-api_1.1_spec-1.0.1.Final.jar
jboss-el-api_3.0_spec-1.0.0.Final.jar
jboss-logging-3.1.0.GA.jar
jboss-marshalling-1.3.11.GA.jar
jboss-marshalling-river-1.3.11.GA.jar
jboss-transaction-api_1.1_spec-1.0.0.Final.jar
jcip-annotations-1.0.jar
jettison-1.3.2.jar
jgroups-3.0.11.Final.jar
jose-jwt-3.0.9.Final.jar
jsr250-api-1.0.jar
JTattoo-1.6.11.jar
mail-1.5.0-b01.jar
mysql-connector-java-5.1.35-bin.jar
oauth-20100527.jar
oauth-provider-20100527.jar
resteasy-atom-provider-3.0.9.Final.jar
resteasy-cache-core-3.0.9.Final.jar
resteasy-client-3.0.9.Final.jar
resteasy-crypto-3.0.9.Final.jar
resteasy-fastinfoset-provider-3.0.9.Final.jar
resteasy-guice-3.0.9.Final.jar
resteasy-hibernatevalidator-provider-3.0.9.Final.jar
resteasy-jackson-provider-3.0.9.Final.jar
resteasy-jackson2-provider-3.0.9.Final.jar
resteasy-jaxb-provider-3.0.9.Final.jar
resteasy-jaxrs-3.0.9.Final.jar
resteasy-jettison-provider-3.0.9.Final.jar
resteasy-jsapi-3.0.9.Final.jar
resteasy-json-p-provider-3.0.9.Final.jar
resteasy-links-3.0.9.Final.jar
resteasy-multipart-provider-3.0.9.Final.jar
resteasy-oauth-3.0.9.Final.jar
resteasy-servlet-initializer-3.0.9.Final.jar
resteasy-spring-3.0.9.Final.jar
resteasy-validator-provider-11-3.0.9.Final.jar

```
resteasy-yaml-provider-3.0.9.Final.jar
rhq-pluginAnnotations-3.0.4.jar
skeleton-key-core-3.0.9.Final.jar
snakeyaml-1.8.jar
spring-aop-3.0.6.RELEASE.jar
spring-asm-3.0.6.RELEASE.jar
spring-beans-3.0.6.RELEASE.jar
spring-context-3.0.6.RELEASE.jar
spring-context-support-3.0.6.RELEASE.jar
spring-core-3.0.6.RELEASE.jar
spring-expression-3.0.6.RELEASE.jar
spring-web-3.0.6.RELEASE.jar
spring-webmvc-3.0.6.RELEASE.jar
stax2-api-3.1.1.jar
validation-api-1.1.0.Final.jar
weld-se-2.1.0.Final.jar
woodstox-core-asl-4.1.1.jar
```

a. Packages and classes :

com.jbp.ee.main

- *Coupon system Loader and Rest Filter*

com.jbp.ee.rest

- *Client.java : parent class that encapsulate common utilities for all client-types (Customer/Company/Admin). the derived classes (JAX-RS resources) will be use its functionality.*
- *JAX-RS resources : API's for clients, documented at WEB-INF\csee.doc*

com.jbp.ee.utils :

- *ClientDTO.java : Transfer object of client (annotated as XmlRootElement). this will be use to send a "client object" upon successful login/logout*
- *CSExceptionHandler.java : catches all exceptions to be handled.*
- *WebAppListener : logging information about session events*

b. Additional folders and files :

/WEB-INF/cs.properties

- *Main configuration file of the system. values loaded upon starting the application.*

/WEB-INF/csee.doc

- *API docs and example for all resources*

/WEB-INF/angular2.jpg

- *Image represents AngularJS/HTML structure, logic and binding.*

/WEB-INF/web.xml

- *Deployment Descriptor : defines the project behavior and operation.*

/WEB-INF/couponsystem.pdf

- *Coupon System project documentation reference.*

/WEB-INF/couponsystem_part2.pdf

- *This document.*

/WEB-INF/lib

- *REST-EASY jars and Coupon system jars*

.

4. System properties

System properties is configurable via WEB-INF/cs.properties. a new attribute has been added to this web version, providing the ability to disable “cleaner thread” when running the app in a web-container (which manages its own threads)

```
# cs.properties file holds global application configuration.
# invalid parameters may lead to unexpected behavior
###########################################################################

# md5(1234) = 81dc9bdb52d04dc20036dbd8313ed055
# admin should login with password of '1234'
# md5 calculator : http://www.md5.cz/
ADMIN_HASH=81dc9bdb52d04dc20036dbd8313ed055

# available templates name appears below
LOOK_AND_FEEL_TEMPLATE=com.jtattoo.plaf.graphite.GraphiteLookAndFeel

# max TCP connections in pool
MAX_DB_CONNECTIONS=2

# global name for this singleton system
SYSTEM_NAME=KoopOnLine-2015

# server ip address/credentials and optional parameters
# allowMultiQueries=true need to be passed to enable few queries at once
# ie . 'SELECT .... ; SELECT ....'
DB_URL=jdbc:mysql://appsrv.kukinet.net/DBCoupons?allowMultiQueries=true
DB_USERNAME=admin
DB_PASSWORD=[--removed--]

# enable / disable the cleaner task
THREAD_ENABLED=false

# 1440M = 24H
THREAD_INTERVAL_MINUTES=1440

# enable / disable system logs
LOGGING_ENABLED=true
```

5. web.xml

global configuration for the project (some output omitted):

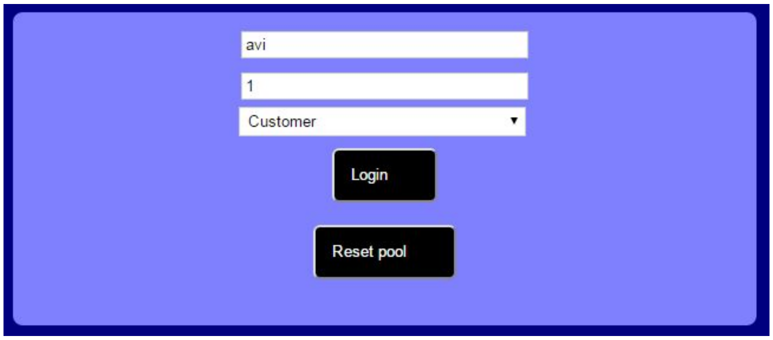
- *Defines the WebAppListener*
- *Map /rest to JAX-RS rest-easy servlet*
- *Defines that all resources (/rest/*) should be mapped to the Rest Filter*
- *Loads the CouponSystemLoader class automatically upon webapp deploying*

```
                <display-name>csee5</display-name>
                <welcome-file-list>
                        <welcome-file>index.html</welcome-file>
                </welcome-file-list>


                <context-param>
                        <param-name>resteasy.scan</param-name>
                        <param-value>true</param-value>
                </context-param>
                <context-param>
                        <param-name>resteasy.servlet.mapping.prefix</param-name>
                        <param-value>/rest</param-value>
                </context-param>
                <listener>
                        <listener-class>

org.jboss.resteasy.plugins.server.servlet.ResteasyBootstrap</listener-class>
                </listener>

                <servlet>
                        <servlet-name>CouponSystemLoader</servlet-name>
                        <servlet-class>com.jbp.ee.main.CouponSystemLoader</servlet-class>
                        <load-on-startup>1</load-on-startup>
                </servlet>
                <servlet-mapping>
                        <servlet-name>CouponSystemLoader</servlet-name>
                        <url-pattern>/load</url-pattern>
                </servlet-mapping>



                <servlet>
                        <description>JAX-RS Tools Generated - Do not modify</description>
                        <servlet-name>jaxrs</servlet-name>

<servlet-class>org.jboss.resteasy.plugins.server.servlet.HttpServletDispatcher</servlet-class>
                        <load-on-startup>2</load-on-startup>
                </servlet>
                <servlet-mapping>
                        <servlet-name>jaxrs</servlet-name>
                        <url-pattern>/rest/*</url-pattern>
                </servlet-mapping>


        </web-app>
```

6. Running system - example output

```
2015-11-20 13:50:56.883      DEBUG  CouponSystemLoader    init() invoked,
file=/WEB-INF/cs.properties
2015-11-20 13:50:56.884      DEBUG  CouponSystemLoader    properties from file loaded to a
hashmap
2015-11-20 13:50:56.884      INFO   CouponSystemLoader    trigerring CouponSystem singleton
2015-11-20 13:50:56.886      INFO   CouponSystem   Coupon system created.
2015-11-20 13:50:56.886      DEBUG  CouponSystem   hashmap assigned to local vars,
configuration applied.
2015-11-20 13:50:56.886      DEBUG  ConnPool       private c'tor invoked for this singleton.
2015-11-20 13:50:57.631      DEBUG  ConnPool       Connection object created. connId=1
2015-11-20 13:50:58.151      DEBUG  ConnPool       Connection object created. connId=2
2015-11-20 13:50:58.736      DEBUG  ConnPool       Connection object created. connId=3
2015-11-20 13:50:59.325      DEBUG  ConnPool       Connection object created. connId=4
2015-11-20 13:50:59.864      DEBUG  ConnPool       Connection object created. connId=5
2015-11-20 13:50:59.865      INFO   ConnPool       singleton c'tor finished. Total of 5 DB
Connections.
2015-11-20 13:50:59.872      INFO   DAOFactory     factory initialize connection pool
2015-11-20 13:50:59.878      INFO   DAOFactory     CustomerDBDAO instantiated.
2015-11-20 13:50:59.887      INFO   DAOFactory     CompanyDBDAO instantiated.
2015-11-20 13:50:59.893      INFO   DAOFactory     CouponDBDAO instantiated.
2015-11-20 13:50:59.893      INFO   CouponSystem   3 DAO objects created.
2015-11-20 13:50:59.894      INFO   CouponSystemLoader    init() done.
Nov 20, 2015 1:50:59 PM org.jboss.resteasy.plugins.server.servlet.ServletContainerDispatcher
INFO: Deploying javax.ws.rs.core.Application: class com.jbp.ee.rest.RestApplication
Nov 20, 2015 1:50:59 PM org.jboss.resteasy.plugins.server.servlet.ServletContainerDispatcher
INFO: Adding singleton resource com.jbp.ee.rest.ManageResource from Application class
com.jbp.ee.rest.RestApplication
Nov 20, 2015 1:50:59 PM org.jboss.resteasy.plugins.server.servlet.ServletContainerDispatcher
INFO: Adding singleton resource com.jbp.ee.rest.AdminResource from Application class
com.jbp.ee.rest.RestApplication
Nov 20, 2015 1:50:59 PM org.jboss.resteasy.plugins.server.servlet.ServletContainerDispatcher
INFO: Adding singleton resource com.jbp.ee.rest.CompanyResource from Application class
com.jbp.ee.rest.RestApplication
Nov 20, 2015 1:50:59 PM org.jboss.resteasy.plugins.server.servlet.ServletContainerDispatcher
INFO: Adding singleton provider com.jbp.ee.utils.CSExceptionHandler from Application class
com.jbp.ee.rest.RestApplication
Nov 20, 2015 1:50:59 PM org.jboss.resteasy.plugins.server.servlet.ServletContainerDispatcher
INFO: Adding singleton resource com.jbp.ee.rest.CustomerResource from Application class
com.jbp.ee.rest.RestApplication
INFO: Server startup in 6450 ms
2015-11-20 13:51:35.603      INFO   CompanyResource       trying customer login : Castro/1234
2015-11-20 13:51:35.603      INFO   CompanyResource       trying login : Castro/1234 from
Client Class
2015-11-20 13:51:35.603      DEBUG  CouponSystem   COMPANY login invoked with pass = 1234
2015-11-20 13:51:35.603      INFO   ConnPool       valid connection provided by pool. free
connections 5/5
2015-11-20 13:51:35.68       INFO   CompanyDBDAO   login ok company : [3]  -  Castro.
2015-11-20 13:51:35.68       INFO   ConnPool       connection returned to pool. free
connections 5/5
2015-11-20 13:51:35.692      INFO   CompanyResource
session:A7365624406FA2EB1CF44C6C705E9073 - new.
```

1. browse to http://{tomcat-ip}:8080/csee5/ a login page appears



avi

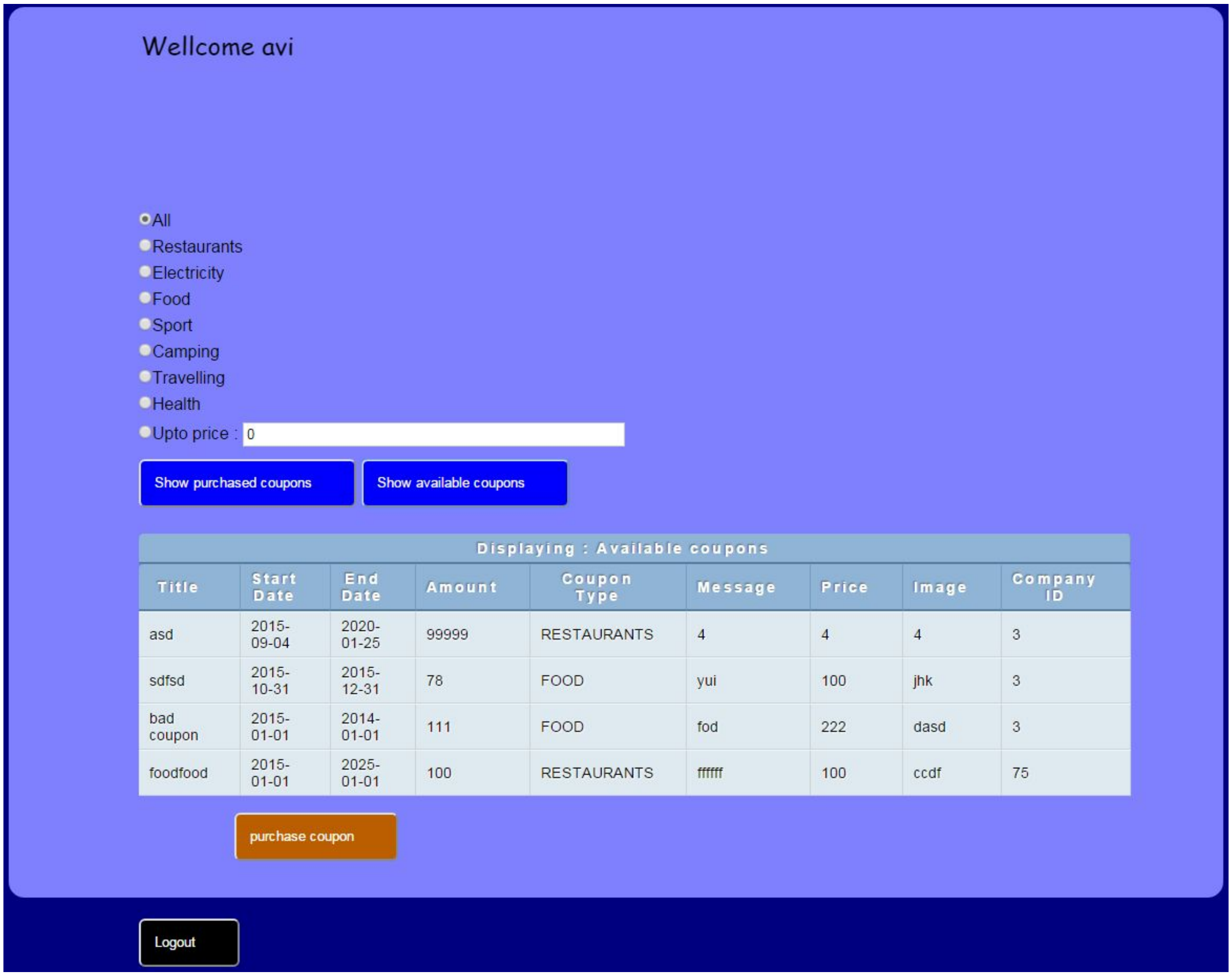
1

Customer ▾

- *Managed by angularJS login controller*
- *generates HTTP GET request to ../rest/customer/login/avi/1*
- *Same for company and admin*
- *Admin login is ../rest/admin/login/admin/1234 (admin appears twice for consistency with other API's)*
- *example of succesul login :*

```
2015-11-12 19:22:34.294 DEBUG   RestFilter     getReqURI=/csee5/rest/company/login/castro/1234
2015-11-12 19:22:34.296 DEBUG   RestFilter      request contains /login. bypassing filter...
2015-11-12 19:22:34.3   INFO    CompanyResource trying customer login : castro/1234
2015-11-12 19:22:34.302 INFO    CompanyResource trying login : castro/1234 from Client Class
2015-11-12 19:22:34.303 DEBUG   CouponSystem    COMPANY login invoked with pass = 1234
2015-11-12 19:22:34.305 INFO    ConnPool        valid connection provided by pool. free conns
2/2
2015-11-12 19:22:34.363 INFO    CompanyDBDAO    login ok company : [3]  -  castro.
2015-11-12 19:22:34.364 INFO    ConnPool        connection returned to pool. free connections
2/2
2015-11-12 19:22:34.369 DEBUG   WebappListener  session created=
B4866C1AD4BBD89B1B6180F3244138A5
2015-11-12 19:22:34.37  DEBUG   WebappListener  total client sessions = 1
2015-11-12 19:22:34.372 INFO    CompanyResource session:B4866C1AD4BBD89B1B6180F3244138A5 -
new.
2015-11-12 19:22:34.375 DEBUG   RestFilter      filter done.
```

Successful customer login



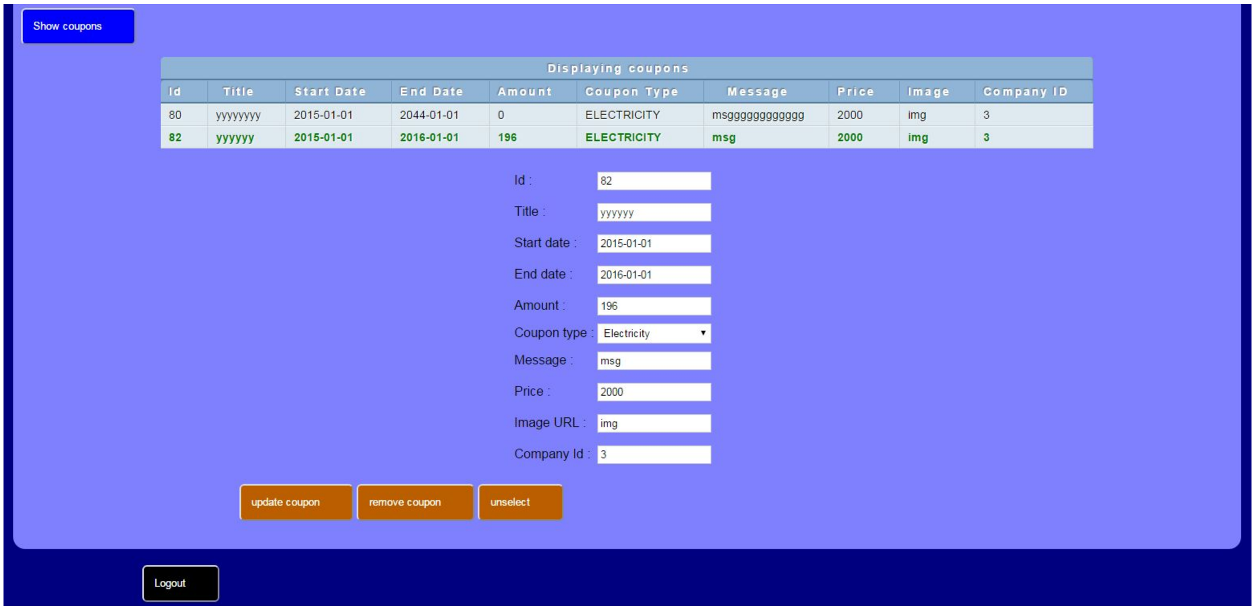
● All
○ Restaurants
○ Electricity
○ Food
○ Sport
○ Camping
○ Travelling
○ Health
○ Upto price : 0

Displaying : Available coupons

Title	Start Date	End Date	Amount	Coupon Type	Message	Price	Image	Company ID
asd	2015-09-04	2020-01-25	99999	RESTAURANTS	4	4	4	3
sdfsd	2015-10-31	2015-12-31	78	FOOD	yui	100	jhk	3
bad coupon	2015-01-01	2014-01-01	111	FOOD	fod	222	dasd	3
foodfood	2015-01-01	2025-01-01	100	RESTAURANTS	ffffff	100	ccdf	75

- *ClientDTO object returns by JAX-RS*
- *Radio buttons filters the table (All by default, can specify a type / max price)*
- *Table shows either Available coupons **OR** Purchased coupons*
- *on “Available coupon” mode - selecting a row will enable to purchase the coupon*
- *logout button exists after successful login*

Successful company login

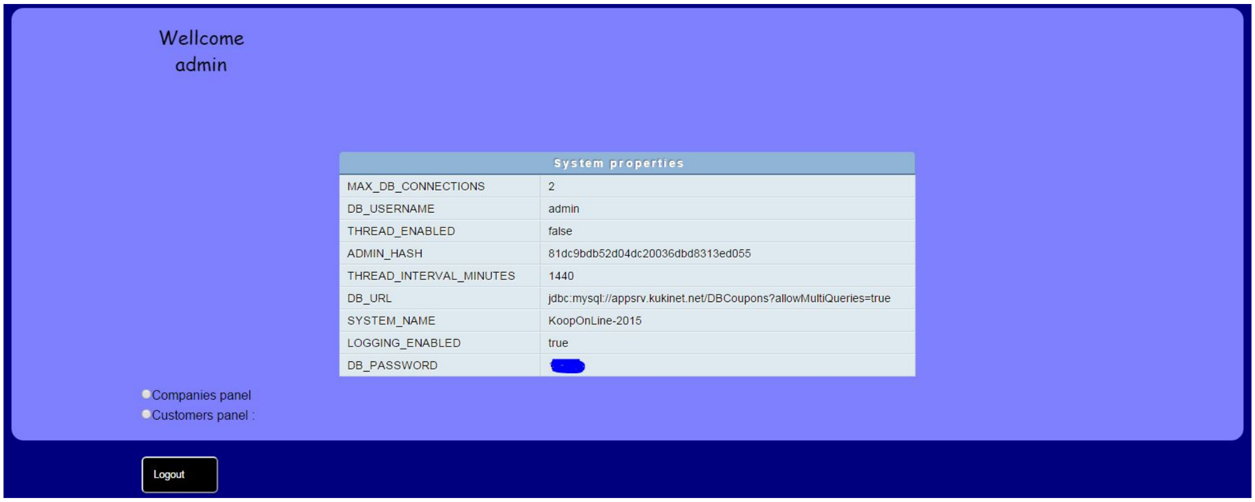


Displaying coupons									
Id	Title	Start Date	End Date	Amount	Coupon Type	Message	Price	Image	Company ID
80	yyyyyyyy	2015-01-01	2044-01-01	0	ELECTRICITY	msgggggggggggg	2000	img	3
82	**yyyyyy**	**2015-01-01**	**2016-01-01**	**196**	**ELECTRICITY**	**msg**	**2000**	**img**	**3**

Id :	82
Title :	yyyyyy
Start date :	2015-01-01
End date :	2016-01-01
Amount :	196
Coupon type :	Electricity
Message :	msg
Price :	2000
Image URL :	img
Company Id :	3

- *ClientDTO object returns by JAX-RS*
- *same filtering method like Customer resource*
- *When selecting a coupon from table the fields shows the coupon and "update" button is optional*
- *"unselect" will clear the fields and the UI is ready for a new coupon (with hints)*

Successful admin login


Wellcome
admin

System properties	
MAX_DB_CONNECTIONS	2
DB_USERNAME	admin
THREAD_ENABLED	false
ADMIN_HASH	81dc9bdb52d04dc20036dbd8313ed055
THREAD_INTERVAL_MINUTES	1440
DB_URL	jdbc:mysql://appsrv.kukinet.net/DBCoupons?allowMultiQueries=true
SYSTEM_NAME	KoopOnLine-2015
LOGGING_ENABLED	true
DB_PASSWORD	[redacted]

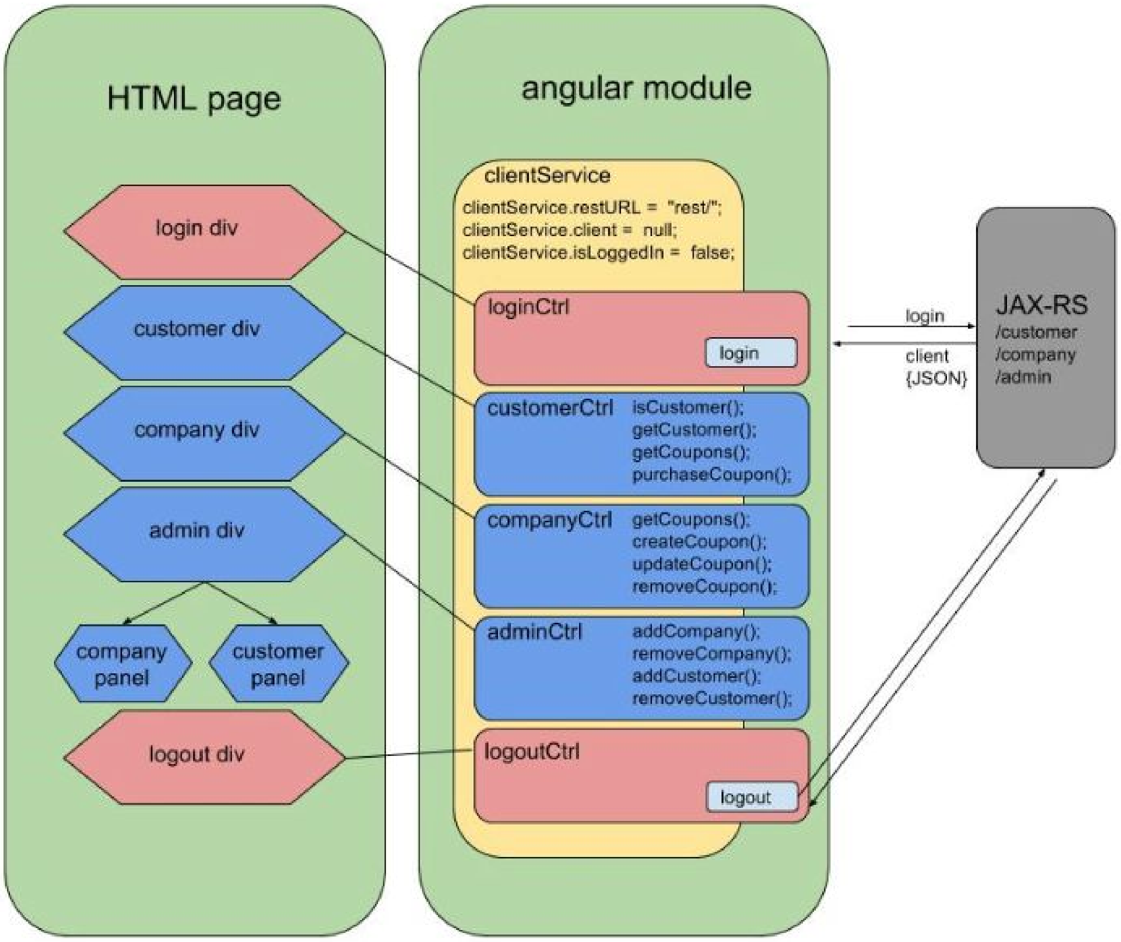
○Companies panel
○Customers panel :

- *ClientDTO object returns by JAX-RS*
- *2 management panels (customer/company) after pressing the radio button (not shown)*
  - *Customer panel : same UI with table/unselect/update/new/etc.*
  - *Companies panel : same UI with table/unselect/update/new/etc.*
- *system properties will appear when mouse is over the table. this happens by ManageResource.*
- *When selecting a coupon from table the fields shows the coupon and "update" button is optional*

7. AngularJS web interface

- AngularJS contains 1 file with 5 controllers :
    - *Login controller - login functionality for all*
    - *Logout controller - logout functionality for all*
    - *Customer, Company and Admin (3 controllers)*
    - *clientService - used to share information between controllers (the current logged in user)*

  Each controller represented as a <div> on the view (index.html) which use a logic of ng-show/ng-hide to show the relevant <div> based on the relevant context. a shared $service injected to all controllers for sharing the current “clientDTO” object received after successful login.



8. REST API's

```
==========================================================
/rest/customer api :
        /whoami                                     (GET/JSON) [returns JSON...]
        /login/{username}/{password}                (GET/JSON)
        /logout                                     (GET/JSON)
        /purchaseCoupon                             (POST/JSON)
        /getAllPurchasedCoupons                     (GET/JSON)
        /getAllPurchasedCoupons/type/{type}         (GET/JSON)
        /getAllPurchasedCoupons/price/{price}              (GET/JSON)
        /getAllAvailableCoupons                     (GET/JSON)
        /getAllAvailableCoupons/type/{type}         (GET/JSON)
        /getAllAvailableCoupons/price/{price}              (GET/JSON)
==========================================================
/rest/company api :
        /whoami                                     (GET/JSON)
        /login/{username}/{password}                (GET/JSON)
        /logout                                     (GET/JSON)
        /createCoupon                               (POST/JSON)
        /removeCoupon                               (DELETE/JSON)
        /updateCoupon                               (PUT/JSON)
        /getCoupon/{id}                                    (GET/JSON)
        /getAllCoupons                              (GET/JSON)
        /getAllCoupons/type/{type}                  (GET/JSON)
==========================================================
/rest/admin api :
        /whoami                                     (GET/JSON)
        /login/{username}/{password}                (GET/JSON)
        /logout                                     (GET/JSON)
        /createCompany                              (POST/JSON)
        /removeCompany                              (DELETE/JSON)
        /updateCompany                              (PUT/JSON)
        /getCompany/{id}                            (GET/JSON)
        /getAllCompanies                            (GET/JSON)
        /createCustomer                                    (POST/JSON)
        /removeCustomer                                    (DELETE/JSON)
        /updateCustomer                                    (PUT/JSON)
        /getCustomer/{id}                           (GET/JSON)
        /getAllCustomers                            (GET/JSON)
        /getConfig                                  (GET/JSON)

==========================================================
/rest/manage api :
        /resetPool                                  (GET/PLAIN)

* helper method to reset the connection pool. no need to login
this specific URL is allowed to bypass the filter.
```

EXAMPLES :

```
==========================================================
GET
http://localhost:8080/csee5/rest/company/login/Castro/1234
==========================================================
POST
http://localhost:8080/csee5/rest/company/createCoupon
body :
{
        "message": "msg2",
        "type": "ELECTRICITY",
        "endDate": "2050-01-01",
        "amount": 100,
        "startDate": "2015-01-01",
        "price": 200,
        "image": "img2",
        "title": "cpn2"
}
==========================================================
POST
http://localhost:8080/csee5/rest/admin/createCompany
body :
{
        "password": "1234",
        "email": "bestcomp@gmail.com",
        "compName": "Best-Comp"
}
==========================================================
```

9. Installation and notes

- The project compressed as a war file which can be imported to eclipse by using this procedure :
    - File -> Import -> WAR file -> Choose “csee5.war” -> Finish
    - The war file will be extracted as an eclipse project
    - Right click on the project -> Run as -> Run on server
    - Tomcat will deploy and auto-start the webapp
        - *NOTE : assuming a new workspace available with Tomcat 8 (Tested on Tomcat 8.0.28)*

- Username / Passwords are case insensitive
- CSS effects appears on the client name (tested on Chrome browser)
- Project csee5 available on svn repository http://svn.kukinet.net/svn/repos/public
- Tested on Tomcat 8.0.27 / windows 7.
- Tested on Tomcat 8.0.28 / windows 10.
- ManagerResource - a special class that bypass the filter (like login). This expose a method to refresh the connection pool (because of connectivity problems after a long period) - the method must be called by external REST client.
- AdminResource - added /getConfig to return an object of properties (displaying on the Admin page.
- /whoami method exposed for all clients, was used for debugging.
- Database credentials appears inside cs.properties file.
- MySQL Database hosted on appsrv.kukinet.net, John-Bryce blocks outbound access by 3306 port (MySQL port).
- Example of active users :

```
/rest/customer/login/avi/1
/rest/company/login/castro/1234
/rest/admin/login/admin/1234
```