PopulateDatabase.XML’s Structure

# User Accounts:

There will be 9 user accounts in the basic populate:

* 1 administrator (ADMINISTRATOR)
* 4 authors (AUTHOR)
* 4 reviewers (REVIEWER)

We will begin with : id = “userAccount1” and so on, until we reach userAccount9. The usernames will be:

* admin (UserAccount1)
* author1, author2, author3, author4 (From uA 2 to 5)
* reviewer1, reviewer2, reviewer3, reviewer4 (From uA 6 to 9)

The passwords in natural language and in MD5 will be:

* For administrators : admin -> 21232f297a57a5a743894a0e4a801fc3
* For authors : author1 , author2, author3, author4
* For reviewers : reviewer1, reviewer2, reviewer3, reviewer4

The basic structure will be:

<bean id=*"userAccount1"* class=*"security.UserAccount"*>

<property name=*"username"* value=*"admin"* />

<property name=*"password"* value=*"21232f297a57a5a743894a0e4a801fc3"* />

<property name=*"authorities"*>

<list>

<bean class=*"security.Authority"*>

<property name=*"authority"* value=*"ADMINISTRATOR"* />

</bean>

</list>

</property>

</bean>

# Actors

The actors have some common attributes and some unique attributes depending of the type of actor. The basic structure is:

<bean id=*"admin"* class=*"domain.Administrator"*>

<property name=*"name"* value=*"Admin Name"* />

<property name=*"middleName"* value=*"Admin Middle Name"* />

<property name=*"surname"* value=*"Admin Surname"* />

<property name=*"photo"* value=*"https://photoadmin.jpg"* />

<property name=*"email"* value=*"admin@us.es"* />

<property name=*"phoneNumber"* value=*"666666666"* />

<property name=*"address"* value=*"Admin Address"* />

<property name=*"userAccount"* ref=*"userAccount1"* />

</bean>

# Configuration Parameters

This object will contain a list with the credit card makes and a list with references to the existing topics.

We can see the structure here:

<!-- CONFIGURATION PARAMETERS -->

<bean id=*"configurationParameters"*

class=*"domain.ConfigurationParameters"*>

<property name=*"sysName"* value=*"Acme Conference"* />

<property name=*"banner"*

value=*"https://i.ibb.co/GVpZCtM/acme-conference.png"* />

<property name=*"message"*

value=*"Welcome to Acme Conference! Your scientific event manager"* />

<property name=*"messageEs"*

value=*"¡Bienvenidos a Acme Conference! Su gestor de eventos científicos"* />

<property name=*"countryCode"* value=*"+34"* />

<property name=*"defaultCountry"* value=*"Spain"* />

<property name=*"creditCardMakes"*>

<list>

<value>VISA</value>

<value>MASTER</value>

<value>DINNERS</value>

<value>AMEX</value>

</list>

</property>

<property name=*"topics"*>

<list>

<ref bean=*"inquiry"*/>

<ref bean=*"rebuttal"*/>

<ref bean=*"registration"*/>

<ref bean=*"topic"*/>

<ref bean=*"other"*/>

</list>

</property>

</bean>

<!-- CONFIGURATION PARAMETERS FINISHED -->

# Messages

There will be one message sent by each actor to 2 other actors, that will be the next 2 actors. And there will be 1 broadcast message with no sender and every actor as a recipient.

Th estructure is:

<bean id=*"message1"* class=*"domain.Message"*>

<property name=*"moment"* value=*"2019/08/05 22:00"* />

<property name=*"subject"* value=*"Message1's subject"* />

<property name=*"body"* value=*"Message1's body"* />

<property name=*"topic"* ref=*"other"* />

<property name=*"sender"* ref=*"admin"* />

<property name=*"recipients"*>

<list>

<ref bean=*"author1"* />

<ref bean=*"author2"* />

</list>

</property>

</bean>

There are 9 messages, 1 for each author.

# Credit Cards

Only authors have Credit Cards. So each one of them will have one.

The estructure is:

<!-- CREDIT CARDS -->

<bean id=*"creditCard1"* class=*"domain.CreditCard"*>

<property name=*"holderName"* value=*"author1"* />

<property name=*"brandName"* value=*"Acme Conference's brand"* />

<property name=*"number"* value=*"4716299708340661"* />

<property name=*"expirationMonth"* value=*"03"* />

<property name=*"expirationYear"* value=*"20"* />

<property name=*"cvv"* value=*"458"* />

<property name=*"make"* value=*"VISA"* />

<property name=*"owner"* ref=*"author1"* />

</bean>

# Conference

There will be 2 conferences. The first will have 3 submissions, and the second will have just 1. The estructure is:

<!-- CONFERENCES -->

<bean id=*"conference1"* class=*"domain.Conference"*>

<property name=*"title"* value=*"Conference1's title"* />

<property name=*"acronym"* value=*"Conference1's acronym"* />

<property name=*"venue"* value=*"Conference1's venue"* />

<property name=*"submissionDeadline"* value=*"2019/10/01 21:00"* />

<property name=*"notificationDeadline"*

value=*"2019/10/02 21:00"* />

<property name=*"cameraReadyDeadline"* value=*"2019/10/03 21:00"* />

<property name=*"startDate"* value=*"2019/10/10 12:00"* />

<property name=*"endDate"* value=*"2019/10/20 21:00"* />

<property name=*"summary"* value=*"Conference1's summary"* />

<property name=*"fee"* value=*"200."* />

<property name=*"administrator"* ref=*"admin"* />

<property name=*"registrations"*>

<list>

</list>

</property>

<property name=*"submissions"*>

<list>

</list>

</property>

<property name=*"activities"*>

<list>

</list>

</property>

</bean>

# Submissions

Submissions have papers and camera ready papers. Papers will exist apart from the submissions, and when an author makes a submission, he will choose between his uploaded papers. This means that author will be able to create papers.

There will be 4 submissions, one for each author. 1º Conference will have 3 and the 2º will have 1.

The estructure is:

<!-- SUBMISSIONS -->

<bean id=*"submission1"* class=*"domain.Conference"*>

<property name=*"ticker"* value=*"AAA-RR19"* />

<property name=*"moment"* value=*"2019/09/10 19:00"* />

<property name=*"status"* value=*"UNDER-REVIEW"* />

<property name=*"author"* ref=*"author1"* />

<property name=*"paper"* ref=*"paper1"* />

</bean>

# Papers

There will be 4 papers, one for each submission. The authors will be the author with the number of the paper and the next author, so each paper will have 2 authors. The estructure is:

<bean id=*"paper1"* class=*"domain.Paper"*>

<property name=*"title"* value=*"Paper1's title"* />

<property name=*"authors"*>

<list>

<ref bean=*"author1"* />

<ref bean=*"author2"* />

</list>

</property>

<property name=*"summary"* value=*"Paper1's summary"* />

<property name=*"document"*

value=*"http://google.com/paperdoc1.html"* />

<property name=*"cameraReady"* value=*"false"* />

</bean>

# Reports

There will be 8 reports, 2 for each submission, so 2 done by each reviewer. We will have 1 on border-line and 1 accepted. The structure is:

<bean id=*"report1"* class=*"domain.Report"*>

<property name=*"originalityScore"* value=*"7.5"* />

<property name=*"qualityScore"* value=*"7.5"* />

<property name=*"readabilityScore"* value=*"7.5"* />

<property name=*"decision"* value=*"BORDER-LINE"* />

<property name=*"reviewer"* ref=*"reviewer1"* />

<property name=*"submission"* ref=*"submission1"* />

<property name=*"comments"*>

<list>

</list>

</property>

</bean>

# Registrations

There will be 6 registrations: 4 for first conference and 2 for the second.