

# A5: Relational Schema, validation and schema refinement

## 1. Relational Schema

The Relational Schema includes the relation schemas, attributes, domains, primary keys, foreign keys and other integrity rules: UNIQUE, DEFAULT, NOT NULL, CHECK. Relation schemas are specified in the compact notation:

Relation reference	Relation Compact Notation
R01	Administrator( <b>id</b> , username UK NN, password NN)
R02	Country( <b>id</b> , countryName UK NN)
R03	Member( <b>id</b> , address, age NN, email UK NN, name NN, password NN, paypalEmail, phone NN, postalCode NN, username UK NN, dateCreated NN DF Today, status DF 'normal' CK status IN MemberStatus, dateBanned CK status='banned', dateSuspended CK status='suspended', dateTerminated CK status='terminated', <b>idCountry</b> → Country NN, <b>idImage</b> → Image)
R04	RequestedTermination( <b>id</b> , dateRequested NN DF Today, <b>idMember</b> -> Member)
R05	Auction( <b>id</b> , author NN, description NN, duration NN CK duration>5, ISBN NN, title NN, dateCreated NN DF Today, status DF 'waitingApproval' CK status IN AuctionStatus, dateApproved CK status='approved', dateRemoved CK status='removed', <b>idPublisher</b> → Publisher, <b>idLanguage</b> -> Language NN, <b>idSeller</b> → Member NN)
R06	Category( <b>id</b> , categoryName UK NN)
R07	Publisher( <b>id</b> , publisherName UK NN)
R08	Language( <b>id</b> , language UK NN)
R09	CategoryAuction( <b>idCategory</b> -> Category, <b>idAuction</b> -> Auction)
R10	WishList( <b>idBuyer</b> -> Member, <b>idAuction</b> -> Auction)
R11	Bid( <b>idBuyer</b> -> Member, <b>idAuction</b> -> Auction, bidDate NN DF Today, bidValue CK bidValue>0.0)
R12	AuctionModification( <b>id</b> , dateRequested NN DF Today, newDescription, is_approved, dateApproved CK is_approved IS TRUE, <b>idApprovedAuction</b> -> Auction NN)
R13	Notification( <b>id</b> , dateSent NN DF Today, information, is_seen DF FALSE, dateSeen, <b>idMember</b> -> Member NN)
R14	NotificationAuction( <b>idAuction</b> -> Auction, <b>idNotification</b> -> Notification)
	Image( <b>id</b> , source UK NN, <b>idAuction</b> → Auction CK idAuctionModification IS NULL,

R15            **idAuctionModification**->AuctionModification CK idAuction IS NULL)

R16            Message(**id**, dateSent NN DF Today, text NN, **idSender** → Member NN, **idReceiver** → Member NN)

R17            Comment(**id**, datePosted NN DF Today, liked, text NN, isRemoved DF FALSE, **idParent** → Comment, **idUser** → Member NN)

where UK means UNIQUE KEY, NN means NOT NULL, DF means DEFAULT and CK means CHECK.

## 2. Domains

The specification of additional domains can also be made in a compact form, using the notation:

Domain Name	Domain Specification
Today	DATE DEFAULT CURRENT_DATE
MemberStatus	ENUM ('moderator', 'suspended', 'banned', 'normal','terminated')
AuctionStatus	ENUM ('approved', 'removed', 'waitingApproval')

## 3. Functional Dependencies and schema validation

To validate the Relational Schema obtained from the Conceptual Model, all functional dependencies are identified and the normalization of all relation schemas is accomplished. Should it be necessary, in case the scheme is not in the Boyce–Codd Normal Form (BCNF), the relational schema is refined using normalization.

Table R01 (Administrator)	
Keys: { id, username }	
Functional Dependencies	
FD0101	{id} → {username, password}
FD0102	{username} → {id, password}
Normal Form	BCNF

Table R02(Country)	
Keys: { id, countryName }	
Functional Dependencies	
FD0201	{ id } → {countryName}
FD0202	{ countryName } → {id}
Normal Form	BCNF

Table R03 (Member)	
Keys: {id, email, username}	
Functional Dependencies	
FD0301	{id} → {address, age, email, name, password, paypalEmail, phone, postalCode, username, dateCreated, status, dateBanned, dateSuspended, dateTerminated, idCountry, idImage}
FD0302	{email} → {id, address, age, name, password, paypalEmail, phone, postalCode, username, dateCreated, status, dateBanned, dateSuspended, dateTerminated, idCountry, idImage}
FD0303	{username} → {id, address, age, email, name, password, paypalEmail, phone, postalCode, dateCreated, status, dateBanned, dateSuspended, dateTerminated, idCountry, idImage}
<b>Normal Form</b>	BCNF

Table R04 (RequestedTermination)	
Keys: {id}	
Functional Dependencies	
FD0401	{ id } → {dateRequested,idMember}
<b>Normal Form</b>	BCNF

Table R05 (Auction)	
Keys: {id}	
Functional Dependencies	
FD0501	{id} → {author, description, duration, ISBN, title, dateCreated, status, dateApproved, dateRemoved, idLanguage, idPublisher, idSeller}
<b>Normal Form</b>	BCNF

Table R06 (Category)	
Keys: {id, categoryName}	
Functional Dependencies	
FD0601	{id} → {categoryName}
FD0601	{categoryName} → {id}
<b>Normal Form</b>	BCNF

--

Table R07 (Publisher)	
Keys: {id, publisherName}	
Functional Dependencies	
FD0701	{id} → {publisherName}
FD0701	{publisherName} → {id}
Normal Form	BCNF

Table R08 (Language)	
Keys: {id, language }	
Functional Dependencies	
FD0801	{id} → {language }
FD0801	{language } → {id}
Normal Form	BCNF

Table R09 (CategoryAuction)	
Keys: {idCategory , idAuction}	
Functional Dependencies	
(none)	
Normal Form	BCNF

Table R10 (WishList)	
Keys: {idBuyer , idAuction}	
Functional Dependencies	
(none)	
Normal Form	BCNF

Table R11 (Bid)	
Keys: {idBuyer , idAuction}	
Functional Dependencies	
FD1101	{idBuyer , idAuction} → {bidDate, bidValue}
Normal Form	BCNF

<b>Table R12</b> (AuctionModification)	
<b>Keys:</b> {id}	
<b>Functional Dependencies</b>	
D1201	{id} → {dateRequested, newDescription, is_approved, dateApproved, idApprovedAuction}
<b>Normal Form</b>	BCNF

<b>Table R13</b> (Notification)	
<b>Keys:</b> {id}	
<b>Functional Dependencies</b>	
FD1301	{id} → {dateSent, information, is_seen, dateSeen, idMember}
<b>Normal Form</b>	BCNF

<b>Table R14</b> (NotificationAuction)	
<b>Keys:</b> {idNotification, idAuction}	
<b>Functional Dependencies</b>	
(none)	
<b>Normal Form</b>	BCNF

<b>Table R15</b> (Image)	
<b>Keys:</b> {id, source}	
<b>Functional Dependencies</b>	
FD1501	{id} → {source, idAuction, idAuctionModification}
FD1502	{source} → {id, idAuction, idAuctionModification}
<b>Normal Form</b>	BCNF

<b>Table R16</b> (Message)	
<b>Keys:</b> {id}	
<b>Functional Dependencies</b>	
FD1601	{id} → {dateSent, text, idSender, idReceiver}
<b>Normal Form</b>	BCNF

Table R17 (Comment)	
Keys: { id }	
Functional Dependencies	
FD1701	{id} → {datePosted, liked, text, isRemoved, idParent, idUser}
Normal Form	BCNF

## 4. SQL Code

SQL code necessary to build (and rebuild) the database. This code should also be included in the group's github repository as an SQL script, and a link include here.

## Revision history

Changes made to the first submission:

1. Item 1
2. Item 2

---

GROUP17xx, xx/xx/2018

Group member 1 name, email Group member 2 name, email