

A9: Main accesses to the database and transactions

GROUP1714, 22/04/2018

- Daniel Ribeiro de Pinho - up201505302@fe.up.pt
- Francisco José Sousa Silva - up201502860@fe.up.pt
- Rui André Rebolo Fernandes Leixo - up201504818@fe.up.pt
- Vitor Emanuel Fernandes Magalhães - up201503447@fe.up.pt

1. Main Accesses

1.1. M01: Authentication and Individual Profile

SQL101 **Creates a new Member in the system**

Web Resource R107

```
INSERT INTO "member" VALUES
(null, $username, $password, $birthdate, $email, $country, null, null,
null, null, null, now(), true, false, false);
```

SQL102 **Updating a Member's Information**

Web Resource R110

```
UPDATE "member"
SET name = $name, password = $password, country = $country, city = $city,
address = $address,
taxpayernumber = $payerNumber, about = $des, profilepicture = $profilePic
WHERE "idmember" = $_SESSION['currentMember'];
```

SQL103 **Member's Notification Feed**

Web Resource R108

```
SELECT timestamp, community, recipient, context, comment, event
FROM "notification"
WHERE "notification".recipient = $selectedMember
LIMIT $selectedLimit OFFSET $selectedOffset
ORDER BY "notification".timestamp;
```

SQL104 User Profile Information

Web Resource R108

```
SELECT name, username, birthdate, email, country, city,
address, taxPayerNumber, about, profilePicture, registrationDate
FROM "Member"
WHERE "Member".idMember = $selectedMember;
```

1.2. M02: Member Management**SQL105 Search for Members**

Web Resource R201

```
SELECT name, username, profilepicture
FROM memeber
WHERE username = $selectedUsername
```

SQL106 Befriend user

Web Resource R202

```
SELECT friend.idf2
FROM friend
WHERE (friend.idf1 = $_SESSION['currentMember'] AND friend.accepted = true)
OR
(friend.idf2 = $_SESSION['currentMember'] AND friend.accepted = true)
LIMIT $selectedLimit OFFSET $selectedOffset
```

SQL107 Set Flag of Friendship

Web Resource R202

```
UPDATE Friend
SET accepted = true
WHERE ("idFriend1" = $_SESSION['currentCommunity'] AND "idFriend2" =
$notificationFriend) OR
("idFriend2" = $_SESSION['currentCommunity'] AND "idFriend1" =
$notificationFriend);
```

1.3. M03: Events

SQL108 Search for Events By Date

Web Resource R301

```
SELECT name, date, imagepath
FROM event
WHERE startdate >= $startDate AND enddate <= $endDate;
```

SQL109 Search for Events By Name

Web Resource R301

```
SELECT name, date, imagepath
FROM event
WHERE name = $selectedName
```

SQL110 View Event

Web Resource R302

```
SELECT e1.name, e1.description, e1.imagePath, e1.startday, e1.endday,
e1.starttime, e1.endtime,
e1.country, e1.city, e1.address, category
FROM "event" e1
INNER JOIN LATERAL(
  SELECT categ.name as category
  FROM "event_eventcategory" con, "eventcategory" categ
  WHERE categ.ideventcategory = con.ideventcategory
) e2 ON true
WHERE "event".idEvent = $selectedEvent;
```

SQL111 Create Event

Web Resource R304

```
SELECT e1.name, e1.description, e1.imagePath, e1.startday, e1.endday,
e1.starttime, e1.endtime,
e1.country, e1.city, e1.address, category
FROM "event" e1
INNER JOIN LATERAL(
  SELECT categ.name as category
  FROM "event_eventcategory" con, "eventcategory" categ
  WHERE categ.ideventcategory = con.ideventcategory
) e2 ON true
WHERE "event".idEvent = $selectedEvent;
```

SQL112 **Edit Event**

Web Resource R306

```
SELECT e1.name, e1.description, e1.imagePath, e1.startday, e1.endday,
e1.starttime, e1.endtime,
e1.country, e1.city, e1.address, category
FROM "event" e1
INNER JOIN LATERAL(
  SELECT categ.name as category
  FROM "event_eventcategory" con, "eventcategory" categ
  WHERE categ.ideventcategory = con.ideventcategory
) e2 ON true
WHERE "event".idEvent = $selectedEvent;
```

SQL113 **Get Top Events by number of attendants**

Web Resource R301

```
SELECT name, imagePath, startday, endday, description
FROM "event" e1
INNER JOIN LATERAL
(
  SELECT COUNT("idmember") as numberMembers
  FROM "event_member"
  WHERE "event_member".idevent= e1.idevent
) e2 ON true
LIMIT $selectedLimit OFFSET $selectedOffset
ORDER BY numberMembers DESC
```

SQL114 **Event names**

Web Resource R301

```
SELECT idevent, name
FROM "event"
LIMIT $selectedLimit OFFSET $selectedOffset
```

SQL115 **Event Participants**

Web Resource R302

```
SELECT "member".username, "member".profilePicture
FROM "member", "event", "event_member"
WHERE "event_member".idevent = $selectedEvent
AND "member".idmember = "event_member".idmember
LIMIT $selectedLimit OFFSET $selectedOffset
```

SQL116 Event comments from specified event

Web Resource R302

```
SELECT c1.text, c1.timestamp, userName FROM comment c1
INNER JOIN LATERAL(
  SELECT member.username AS userName
  FROM member
  WHERE c1.author = member.idmember
) c2 on true
WHERE c1.event = $selectedEvent
LIMIT $selectedLimit OFFSET $selectedOffset
```

SQL117 Add/Remove new Admin from an Event

Web Resource R302

```
UPDATE "event_member"
SET isAdmin = $permissions
WHERE "idmember" = $selectedMember
AND "idevent" = $selectedEvent
```

SQL118 Member Joining Event

Web Resource R302

```
INSERT INTO "event_member" VALUES
($selectedEvent, $_SESSION["currentMember"], false);
```

SQL119 Invite a Member to an Event

Web Resource R302

```
INSERT INTO "notification" VALUES
($idNotification, $timestamp, 'event', null, $_SESSION["currentMember"],
'You were invited to', null, $idEvent);
```

SQL120 Remove Member from Event

Web Resource R302

```
DELETE FROM "Event_Member"
WHERE "idMember" = $_SESSION['currentMember']
AND "idEvent" = $selectedEvent;
```

1.4. M04: Tickets

SQL121 Purchase Ticket

Web Resource R402

```
INSERT INTO "ticket" VALUES
($type, $_SESSION['currentMember'], $idInvoice);
```

SQL122 Member ticket purchases

Web Resource R403

```
SELECT ticketType, ticketDescription, ticketPrice, invoiceTaxPayerNr,
invoiceName, invoiceQuantity, invoiceAmount, invoiceAddress, invoiceDate
FROM ticket t1
INNER JOIN LATERAL(
    SELECT invoice.taxpayernumber AS invoiceTaxPayerNr,
    invoice.idinvoice AS idInvoice,
    invoice.name AS invoiceName,
    invoice.quantity AS invoiceQuantity,
    invoice.amount AS invoiceAmount,
    invoice.address AS invoiceAddress,
    invoice.date AS invoiceDate
    FROM invoice
    WHERE t1.idticket = invoice.idinvoice
) c2 ON true

INNER JOIN LATERAL(
    SELECT ticketType.type AS ticketType,
    ticketType.description AS ticketDescription,
    ticketType.price AS ticketPrice,
    ticketType.event AS eventTicket
    FROM tickettype
    WHERE t1.type = tickettype.idtickettype
) c3 ON true

WHERE t1.buyer = $_SESSION['currentMember']
LIMIT $selectedLimit OFFSET $selectedOffset
```

1.5. M05: Communities

SQL123 Search for Communities

Web Resource R501

```
SELECT c1.name, c1.imagepath, numberMembers AS FROM community c1
INNER JOIN LATERAL(
  SELECT COUNT("idmember") as numberMembers
  FROM "community_member"
  WHERE "community_member".idcommunity= c1.idcommunity
) c2 ON true
WHERE name = $selectedName
```

SQL124 View Community

Web Resource R502

```
SELECT c1.name, description, imagePath, creationDate, numberMembers,
category, admin
FROM "community" c1
INNER JOIN LATERAL(
  SELECT COUNT("idmember") as numberMembers
  FROM "community_member"
  WHERE "community_member".idcommunity= c1.idcommunity
) c2 ON true
INNER JOIN LATERAL(
  SELECT cat.name as category
  FROM "community_communitycategory" con, "communitycategory" cat
  WHERE cat.idcommunitycategory = con.idcommunitycategory
) c3 ON true
INNER JOIN LATERAL (
  SELECT member.name as admin
  FROM "member", "community_member" con
  WHERE (member.idmember = con.idmember AND isadmin = true)
) c4 ON TRUE
WHERE c1.idcommunity = $selectedCommunity;
```

SQL125 Create Community

Web Resource R504

```
INSERT INTO "community" VALUES
($name, $description, $timestamp, $imagePath, $publicLink, $isPublic);
```

SQL126 **Community's Activity Feed**

Web Resource R502

```
SELECT timestamp, recipient, context, comment, community
FROM "notification"
WHERE "notification".community = $selectedCommunity
LIMIT $selectedLimit OFFSET $selectedOffset
ORDER BY "notification".timestamp;
```

SQL127 **Get Top Communities**

Web Resource R501

```
SELECT name, description, imagePath, creationDate, numberMembers
FROM "community" c1
INNER JOIN LATERAL
(
SELECT COUNT("idmember") as numberMembers
FROM "community_member"
WHERE "community_member".idcommunity= c1.idcommunity
) e2 ON true
LIMIT $selectedLimit OFFSET $selectedOffset
ORDER BY numberMembers DESC
```

SQL128 **Community Participants**

Web Resource R502

```
SELECT "member".username, "member".profilePicture
FROM "member", "community", "community_member"
WHERE "community_member".idcommunity = $selectedCommunity
AND "member".idmember = "community_member".idmember
LIMIT $selectedLimit OFFSET $selectedOffset
```

SQL129 **Community names**

Web Resource R501

```
SELECT idcommunity, name
FROM "community"
LIMIT $selectedLimit OFFSET $selectedOffset
```


SQL130 Member Joining Community

Web Resource R501

```
INSERT INTO "community_member" VALUES
($selectedCommunity, $_SESSION["currentMember"], false);
```

SQL130 Invite a Member to a Community

Web Resource R508

```
INSERT INTO "notification" VALUES
($idNnotification, $timestamp, 'community', $idCommunity,
$_SESSION["currentMember"],
'You were invited to', null, null);
```

SQL131 Remove Member from Community

Web Resource R508

```
DELETE FROM "Community_Member"
WHERE "idMember" = $_SESSION['currentMember']
AND "idCommunity" = $selectedCommunity;
```

1.6. M06: Website Administration**SQL132 View Reports**

Web Resource R601

```
SELECT userName, c1.context, communityName, commentText, eventName FROM
report c1
INNER JOIN LATERAL(
  SELECT member.username as userName
  FROM member
  WHERE c1.member = member.idmember
) c2 ON true
INNER JOIN LATERAL(
  SELECT community.name as communityName
  FROM community
  WHERE c1.community = community.idcommunity
) c3 ON true
INNER JOIN LATERAL(
  SELECT comment.text as commentText
  FROM comment
```

```
WHERE c1.comment = comment.idcomment
) c4 ON true
INNER JOIN LATERAL(
  SELECT event.name as eventName
  FROM event
  WHERE c1.event = event.idevent
) c5 ON true
LIMIT $selectedLimit OFFSET $selectedOffset
```

SQL133 Process Report

Web Resource R602

```
UPDATE "report"
SET solved = true
WHERE idReport = $selectedReport;
```

SQL134 Ban Event

Web Resource R604

```
UPDATE Event
SET isactive = false
WHERE "idEvent" = $selectedCommunity;
```

SQL135 Ban Community

Web Resource R606

```
UPDATE Community
SET isactive = false
WHERE "idCommunity" = $selectedCommunity;
```

SQL136 Ban Member

Web Resource R608

```
UPDATE Member
SET isactive = false
WHERE "idMember" = $selectedMember;
```

SQL137 Remove comment

SQL137 Remove comment

Web Resource R608

```
DELETE FROM "comment"
WHERE "idcomment" = $selectedComment
AND "author" = $selectedMember;
```

2. Transactions

T01 Member purchasing a ticket

Justification When purchasing a ticket, the money has to be withdrawn from the account before giving the ticket to the member. If an error occurs, a ROLLBACK is issued so the Member doesn't lose any money.

Isolation level REPEATABLE READ

```
BEGIN TRANSACTION;
SET TRANSACTION ISOLATION LEVEL REPEATABLE READ

INSERT INTO "ticket" VALUES
($type, $_SESSION['currentMember'], $idInvoice);

COMMIT;
```

T02 Get Top Events by number of attendants

Justification The top events are ordered by number of attendants. The transaction can occur when a Member is browsing the top events and the event with the most number of attendants becomes second and vice-versa.

Isolation level SERIALIZABLE READ ONLY

```
BEGIN TRANSACTION;
SET TRANSACTION ISOLATION LEVEL SERIALIZABLE READ ONLY

SELECT name, imagePath, startday, endday, description
FROM "event" e1
INNER JOIN LATERAL
(
SELECT COUNT("idmember") as numberMembers
FROM "event_member"
WHERE "event_member".idevent= e1.idevent
) e2 ON true
```

```
LIMIT $selectedLimit OFFSET $selectedOffset
ORDER BY numberMembers DESC

COMMIT;
```

T03 **Invite a Member to an Event**

Justification	When inviting a Member, the transaction has to make sure that there was a notification generated for the invited Member
---------------	---

Isolation level	REPEATABLE READ
-----------------	-----------------

```
INSERT INTO "notification" VALUES
($idNnotification, $timestamp, 'event', null, $_SESSION["currentMember"],
'You were invited to', null, $idEvent)
```

T04 **Invite a Member to a Community**

Justification	When inviting a Member, the transaction has to make sure that there was a notification generated for the invited Member
---------------	---

Isolation level	REPEATABLE READ
-----------------	-----------------

```
INSERT INTO "notification" VALUES
($idNnotification, $timestamp, 'community', $idCommunity,
$_SESSION["currentMember"],
'You were invited to', null, null);
```