A9: Main accesses to the database and transactions

This A9 shows the main accesses to the database, including the transactions.

For each transaction, the isolation level is explicitly stated and read-only transactions are identified to improve global performance.

1. Main Accesses

Main accesses to the database.

1.1 M01: Registration

SQL101	Creates a new customer in the platform
Web Resource	R105

```sql
INSERT INTO "user" VALUES ($username, $passw , $email);
INSERT INTO customer VALUES ($username, $name , $address);
```

1.2 M02: Full text search on Products

SQL201	Searches a product on the database
Web Resource	R202 , R204

```sql
SELECT *
FROM product P, category C
WHERE P.search @ plainto_tsquery('english', $search_words) and C.id = P.category_idcat
ORDER BY ts_rank(P.search, plainto_tsquery('english', $search_words)) DESC
LIMIT $max_number_results OFFSET $page;
```

1.3 M03: Comments

SQL301	Gets comments of a given product
Web Resource	R304

```sql
SELECT *
FROM comment C, answer A, "user" U,
WHERE   C.product_idproduct = $sku
        AND C.user_username = U.username
        AND ( ( C.id = A.idparent ) OR ( C.id = A.idchild ) OR ( NOT EXISTS ( SELECT * FROM  answer A2 w
ORDER BY C."date" ASC;
```

1.4 M04: Moderation

SQL401	Gets comments for moderator to review
Web Resource	R508

```sql
SELECT *
FROM comment C, "user" U,
WHERE   C.deleted = false
        AND C.flagsno > $threshold,
        AND C.user_username = U.username;
```

2. Transactions

T01	Insert a new customer in the system
Isolation level	SERIALIZABLE
Justification	When a customer is created there is an insert at the User table and at the Customer table and any operation in between must be such that there aren't any Phantom Reads.

```sql
BEGIN TRANSACTION;
SET TRANSACTION ISOLATION SERIALIZABLE
-- Insert User
INSERT INTO "user" VALUES ($username, $password, $email);
-- Insert Customer
INSERT INTO customer VALUES ($username, $name, $address);
COMMIT;
```

T02	Get customer info
Isolation level	READ UNCOMMITTED READONLY
Justification	When a customer views it's profile, it isn't very worrysome if not all of their information is updated. As such, we don't mind dirty reads.

```sql
BEGIN TRANSACTION;
SET TRANSACTION ISOLATION READ UNCOMMITTED READ ONLY
-- Get customer info
SELECT *
FROM "user" U, customer C
WHERE   U.username = $username
        AND     U.username = C.username;
-- Get customer purchases
SELECT *
FROM purchase PUR, purchase_product PURPRO, product PRO
WHERE   PUR.customer_username = $username
        AND PUR.id = PURPRO.purchase_idpurchase
        AND PURPRO.product_idproduct = PRO.sku
ORDER BY PUR."date" DESC
-- Get customer favorites
SELECT *
FROM product P, favorite F
WHERE   F.customer_username = $username
        AND     P.sku = F.product_idproduct;
COMMIT;
```

T03	Get products for main page
Isolation level	READ COMMITTED READONLY
Justification	When a customer is in the homepage we want to display discounted and recommended products and allow the user to search products but we don't want dirty reads to happen because products might be being updated.

```sql
BEGIN TRANSACTION;
SET TRANSACTION ISOLATION READ COMMITTED READ ONLY
-- Get discounted products
SELECT * , (P.price - P.discountprice)/P.price AS rank
FROM product P
WHERE   P.discountprice IS NOT NULL
ORDER BY rank DESC
LIMIT $limit;
-- Get products with higher rating
SELECT *
FROM product P
ORDER BY rating DESC
LIMIT $limit;
-- Get best selling products
SELECT *, SUM(PP.quantity) as sumQ
FROM product P, purchase_product PP
WHERE P.sku = PP.product_idproduct
GROUP BY P.sku
ORDER BY sumQ DESC
LIMIT $limit;
COMMIT;
```

Revision history

GROUP1723, 25/04/2018

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