

Université de Cergy-Pontoise

## **Rapport du projet**

UE Projet Base de données/ Réseau

*Licence d'informatique troisième année*

Sur le sujet

## **Gestion de cinéma**

Rédigé par :

CHLOUCHI Meryem, ROUAS Lila, MALEK Feriel

Encadré par :

Mr. Marc LEMAIRE



## Table des matières

<b>1. Présentation du contexte</b>	4
<b>2. Cahier de charges</b>	4
<b>3. Planning</b>	6
<b>4. Protocole réseau</b>	7
<b>5. Architecture du projet</b>	7
<b>6. Dictionnaire de données</b>	8
<b>7. Schéma E/A</b>	10
<b>8. Schéma relationnel</b>	11
<b>9. Schémas applicatifs</b>	12
<i>Cas 1 : Ticket valide</i>	12
<i>Cas 2 : Ticket invalide</i>	13
<i>Cas 3 : Carte valide</i>	14
<i>Cas 4 : Carte invalide</i>	15
<b>10. Jeu de données</b>	16
Table Film	16
Table Client	16
Table Client_passager	16
Table Client_abonne	16
Table Seance	17
Table salle	17
Table carte_abonnement	17
Table ticket	17
Table employé	18
Table sites	18
Table genre	18
<b>11. Requêtes SQL significatives</b>	19

## Table de figures

Figure 1 Planning du projet .....	6
Figure 2 Architecture du projet.....	7
Figure 3 Schéma E/A.....	10
Figure 4 Schéma applicatif (ticket valide) .....	12
Figure 5 Schéma applicatif (ticket invalide) .....	13
Figure 6 Schéma applicatif (carte valide) .....	14
Figure 7 Schéma applicatif (carte invalide) .....	15

## ***1. Présentation du contexte***

L'informatique est l'ensemble des connaissances qui vise à systématiser un processus, à utiliser et traiter des données afin d'atteindre les objectifs de la manière la plus efficace possible. C'est dans ce cadre que s'inscrit notre projet Bases de Données et Réseau qui consiste en la conception et la réalisation d'un système de gestion d'un cinéma (gestion des données, gestion des tickets et gestion des cartes d'abonnement des clients).

L'objectif de notre projet est de mettre en évidence nos connaissances en Bases de Données et Réseau et générer un système qui permet de faciliter la coordination entre les différentes entités du cinéma et la génération des tickets et des cartes.

## ***2. Cahier de charges***

### ***Besoins fonctionnels :***

#### **Le système doit permettre la gestion des tickets et des cartes :**

- Le système doit permettre aux clients d'avoir un ticket correspondant à leurs cas et au film choisi.
- Le système doit permettre aux clients de valider leurs tickets pour accéder à la salle du film choisi.
- Le système doit permettre la gestion des cartes d'abonnement des clients.
- Le système doit permettre aux clients d'avoir une carte d'abonnement qui leur permet d'accéder à certains films pendant la période choisie (an, trimestre, ...).
- Le système doit permettre aux clients de valider leurs cartes pour accéder à la salle du film choisi.

#### **Le système doit permettre la gestion du cinéma :**

- Le système doit permettre aux employés de consulter la liste des films, ajouter un film, modifier ses informations et le supprimer.
- Le système doit permettre aux employés de consulter la liste des salles de cinéma, ajouter une salle et modifier ses informations.
- Le système doit permettre aux employés de gérer le calendrier et d'attribuer une salle pour un film à un moment donné.
- Le système doit permettre aux employés de gérer les cartes d'abonnements et les tickets.

- Le processus d'authentification de type login / password est représenté par les employés qui doivent se connecter pour accéder au site et modifier les informations de la base de données.

*Besoins et contraintes techniques :*

- Un serveur de bases de données utilisant le SGBD Postgresql.
- Un serveur web avec une connexion en PHP vers le serveur de bases de données.
- Un client réseau écrit en Java.
- Un serveur réseau écrit en C avec une connexion vers le serveur de base de données utilisant ODBC.
- Un client web graphique (navigateur).

### 3. Planning

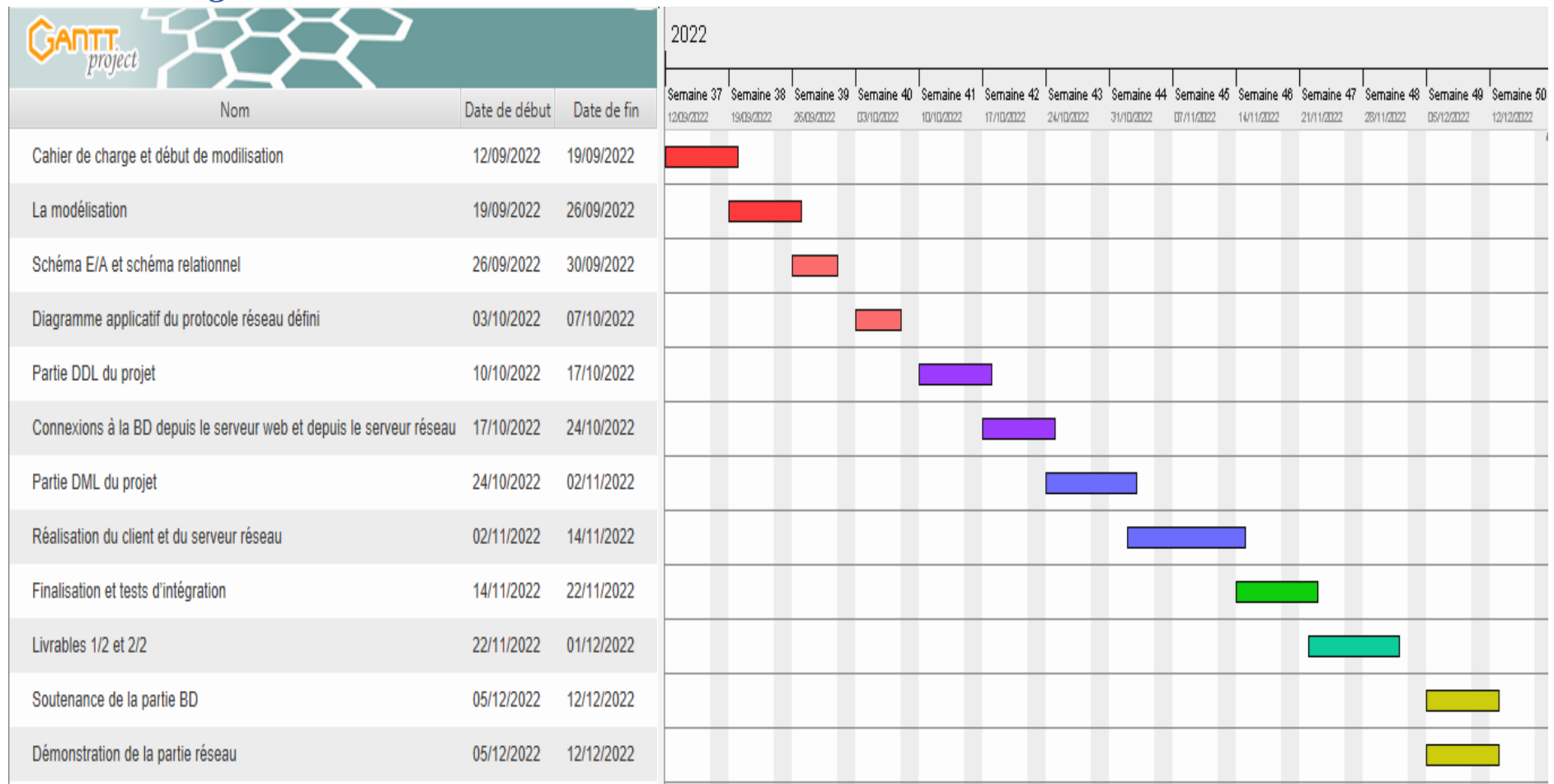


Figure 1 Planning du projet

## 4. Protocole réseau

Pour permettre aux clients de valider leurs cartes ou leurs tickets pour accéder à la salle du film, on a besoin de communiquer les informations nécessaires du ticket du client réseau au serveur réseau et du serveur réseau au serveur base de données et ensuite récupérer les réponses de ce dernier pour pouvoir soit donner l'accès ou pas avec des réponses dans les deux cas. Pour garantir cet échange de données, on utilisera le protocole TCP qui est un protocole de transport fiable, orienté connexion, qui fournit un flux d'octets fiable 4 assurant l'arrivée des données sans altérations et dans l'ordre, avec transmission en cas de perte, et élimination des données dupliquées.

## 5. Architecture du projet

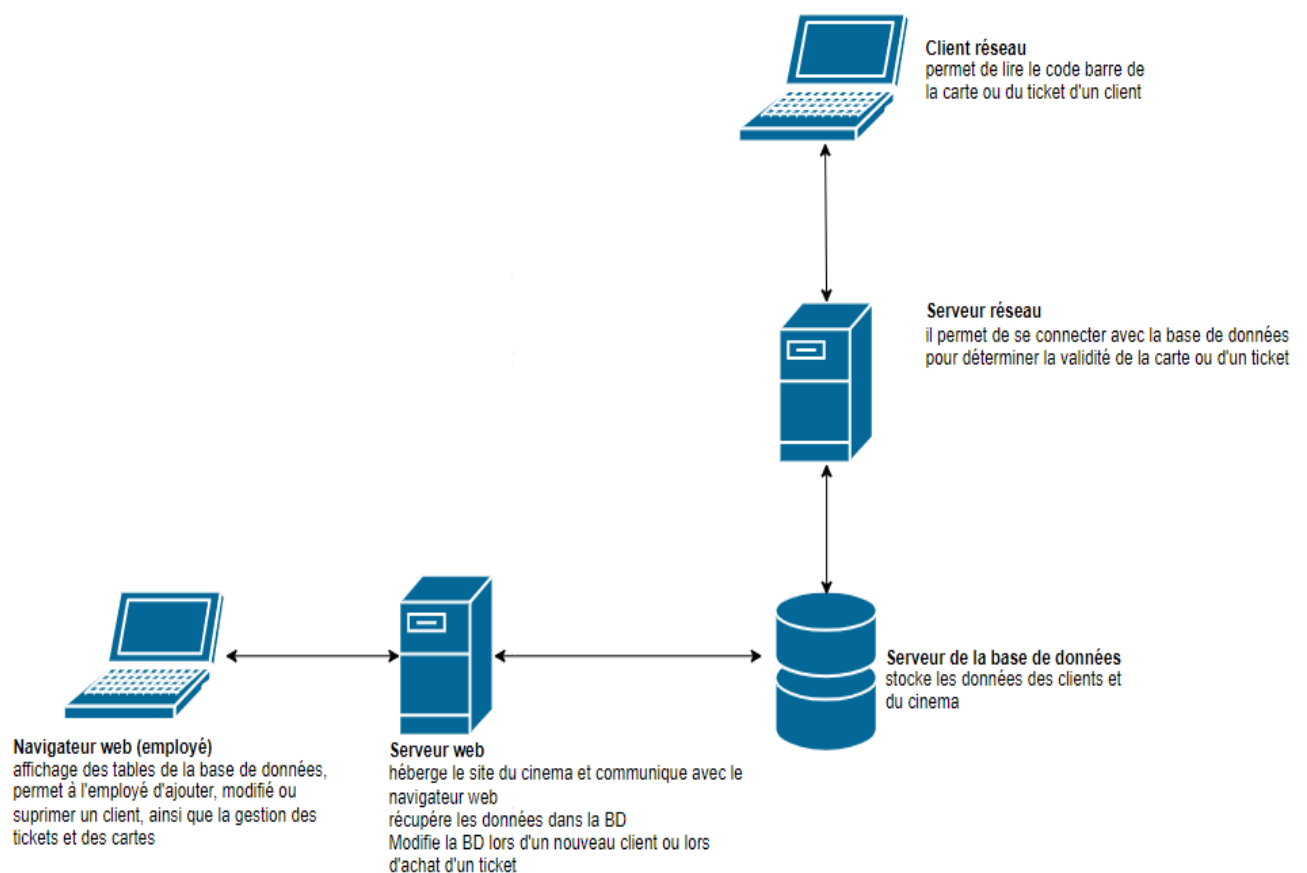


Figure 2 Architecture du projet

## 6. Dictionnaire de données

Entité	Attribut	Type	Domaine	NULL ou NN	Informations
Film	id_film	Char (6)	f00001<x<f99999	NN	PK
	nom_film	Varchar (20)	Alphabétique	NN	
	date_film	DATE	1800<x<2022	NN	
	lang_film	Char (20)	Alphabétique	NN	
	auteur_film	Varchar (20)		NN	
Genre	id_genre	Char (6)	g00000<x<g99999	NN	PK
	nom_genre	Varchar (20)	Alphabétique	NN	
Client	id_client	Char (8)	c0000001<x<c9999999	NN	
	datenai_client	DATE	1930<x<2019	NN	
	fonction_client	Varchar (20)	Alphabétique	NN	
Client_passager	id_client_pass	Char (8)		NN	
Client_abonne	id_client_ab	Char (8)		NN	
	nom_client	Varchar (20)	Alphabétique	NN	
	adress_client	Varchar (100)	alpha-numerique	NN	
	tel_client	Char (10)	Numérique	NN	
	mail_client	Varchar (30)	alpha-numerique	NN	
Seance	id_seance	Char (8)	se0000001<x<se999999	NN	PK
	horaire_seance	Date Time		NN	
Salle	id_salle	Char (6)	s00001<x<s99999	NN	PK
	nom_salle	Varchar (20)	Alphabétique	NN	
	num_salle	Varchar (10)	alpha-numerique	NN	
	capacite_salle	Integer		NN	
Carte_abonnement	id_carte	Char (10)	ca000001<x<ca999999	NN	PK
	nom_carte	Varchar (20)	alpha-numerique	NN	



	date_delivrance	DATE	Current Date	NN	
	date_debut_ab	DATE		NULL	
	date_fin_ab	DATE		NULL	
	type_ab	Varchar (20)		NULL	
	nb_seances	Integer		NULL	
	prix_ab	Float		NULL	
Ticket	id_ticket	Char (10)	t0000001<x<t9999999	NN	PK
	nom_ticket	Varchar (20)	alpha-numerique	NN	
	date_ticket	DATE		NN	
	numplace_ticket	Integer		NN	
	tarif_ticket				
	prix_ticket				
Employé	id_emp	Char (4)	e000<x<e999	NN	PK
	nom_emp	Varchar (20)	Alphabétique	NN	
	salaire_emp	Float		NN	
	statut_emp	Varchar (20)	Alphabétique	NN	
	tel_emp	Char (10)	Numérique		
	mail_emp	Varchar (30)			
	mot_de_passe				
Sites	id_site	Char (4)	si00<x<si99	NN	PK
	nom_site	Varchar (20)	Alpha-numérique	NN	
	adresse	Varchar (40)	Alpha-numérique	NN	
	codePostal_site	Integer		NN	
	ville_site	Varchar (20)	Alphabétique		
	pays_site	Varchar (20)	Alphabétique		

## 7. Schéma E/A

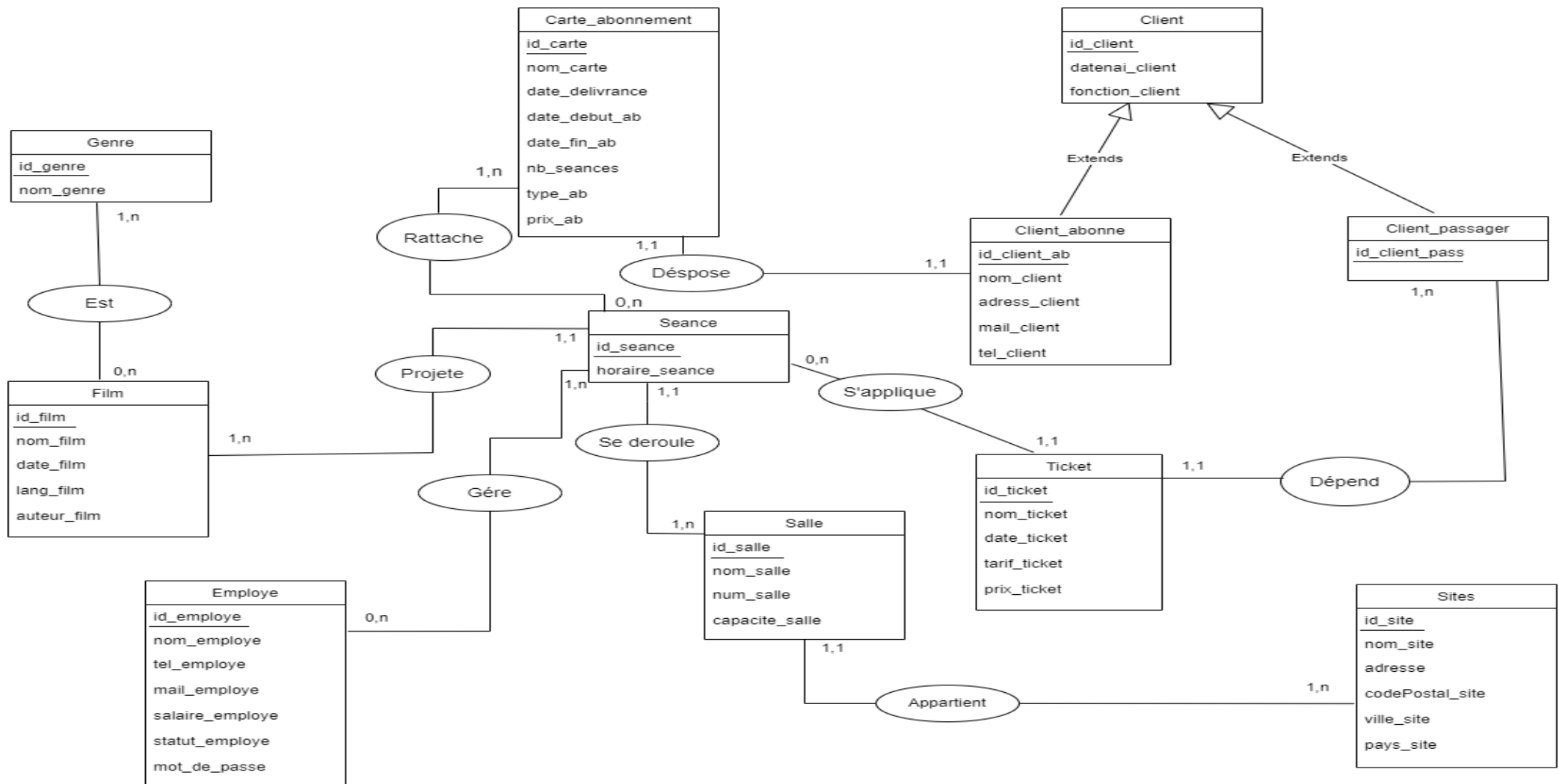


Figure 3 Schéma E/A

## 8. Schéma relationnel

- **Employe** (id\_emp, nom\_emp, salaire\_emp, statut\_emp, tel\_emp, mail\_emp, mot\_de\_passe)
- **Genre** (id\_genre, nom\_genre)
- **Film** (id\_film, nom\_film, date\_film, lang\_film, auteur\_film, #id\_genre)
- **Sites** (id\_site, nom\_site, adresse, codePostal\_site, ville\_site, pays\_site)
- **Salle** (id\_salle, nom\_salle, num\_salle, capacite\_salle, #id\_site)
- **Client** (id\_client, datenai\_client, fonction\_client)
- **Client\_passager** (#id\_client\_pass)
- **Client\_abonne** (#id\_client\_ab, nom\_client, adress\_client, tel\_client, mail\_client)
- **Ticket** (id\_ticket, nom\_ticket, date\_ticket, tarif\_ticket, prix\_ticket, #id\_client\_pass, #id\_seance)
- **Carte\_abonnement** (id\_carte, nom\_carte, date\_delivrance, date\_debut\_ab, date\_fin\_ab, type\_ab, nb\_seances, prix\_ab, #id\_client\_ab, #id\_seance)
- **Seance** (id\_sceance, horaire\_seance, #id\_salle, #id\_film, #id\_emp)

## 9. Schémas applicatifs

Nous avons choisi de faire 4 diagrammes pour illustrer notre protocol applicatif où on représente les différentes étapes de communication entre notre client et serveur réseau ainsi que les différentes données échangées

### Cas 1 : Ticket valide

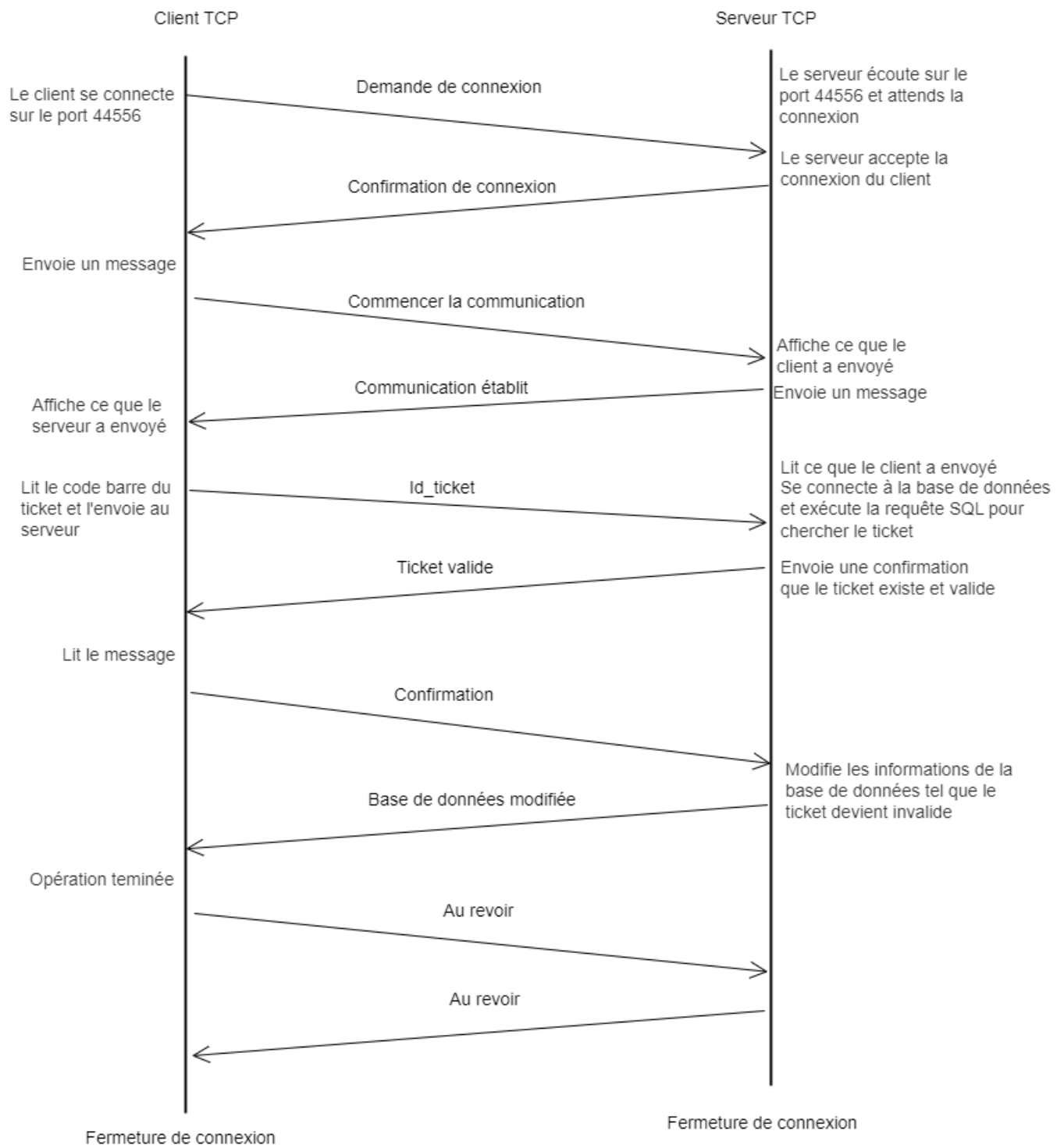


Figure 4 Schéma applicatif (ticket valide)

## Cas 2 : Ticket invalide

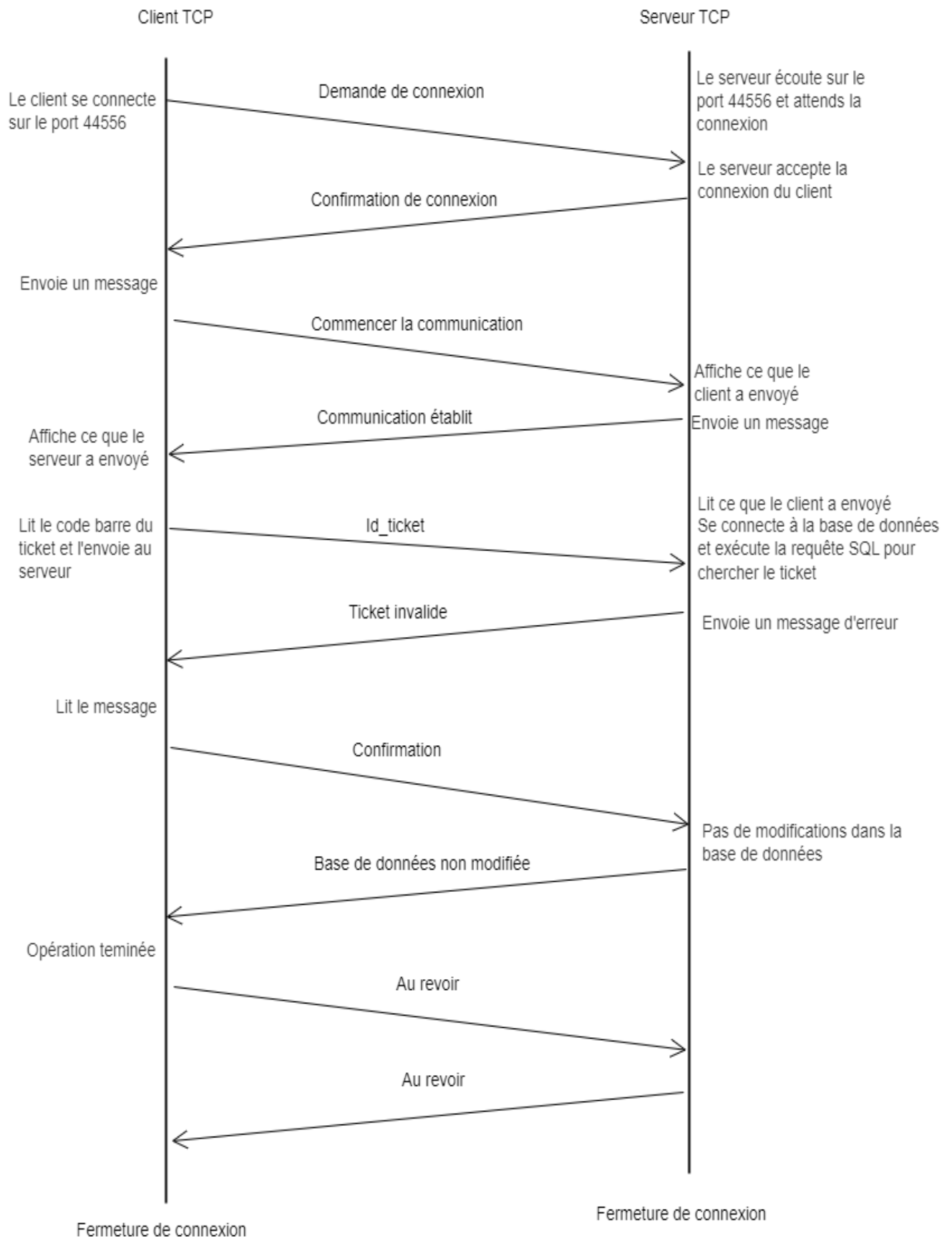


Figure 5 Schéma applicatif (ticket invalide)

### Cas 3 : Carte valide

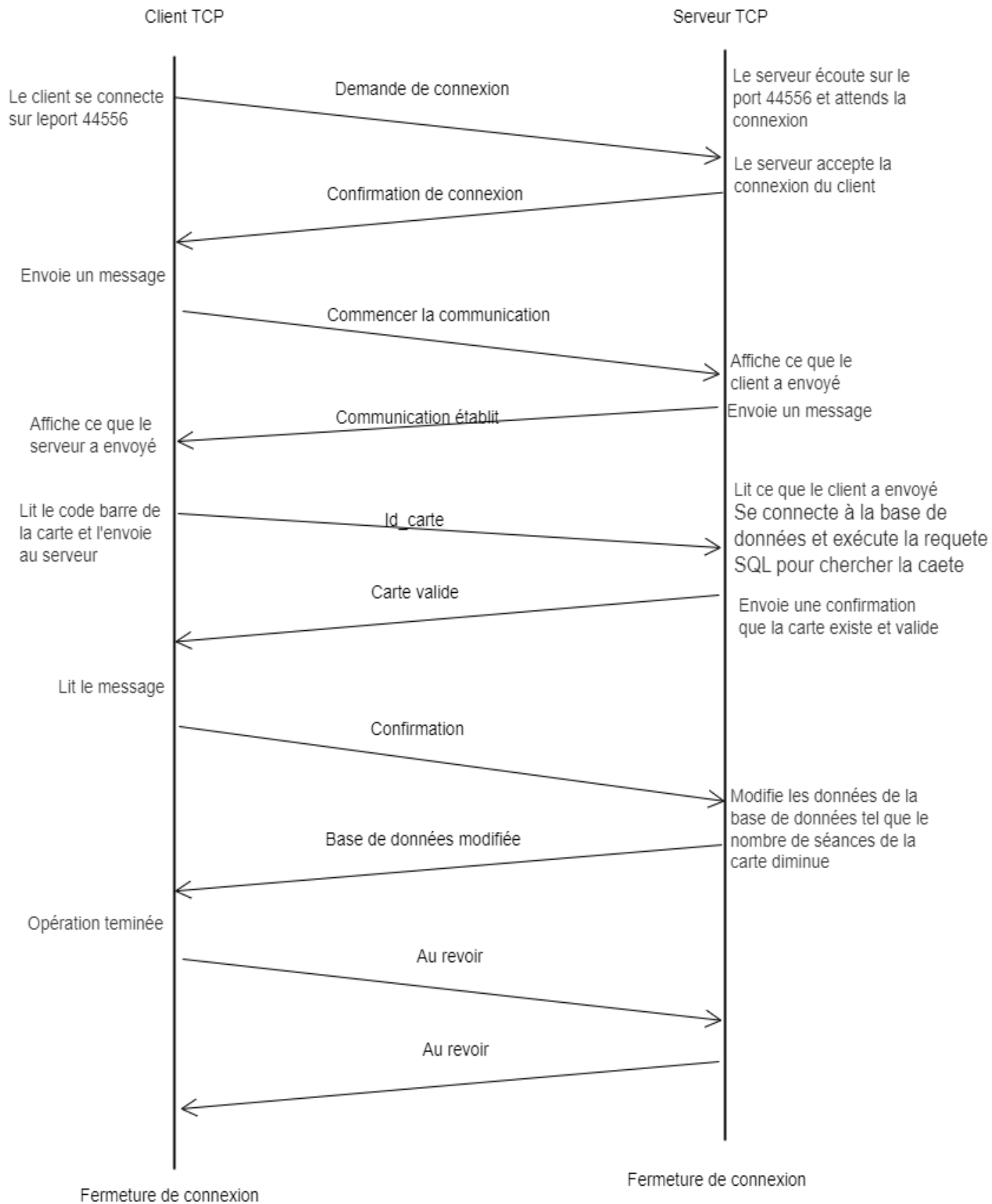


Figure 6 Schéma applicatif (carte valide)

#### Cas 4 : Carte invalide

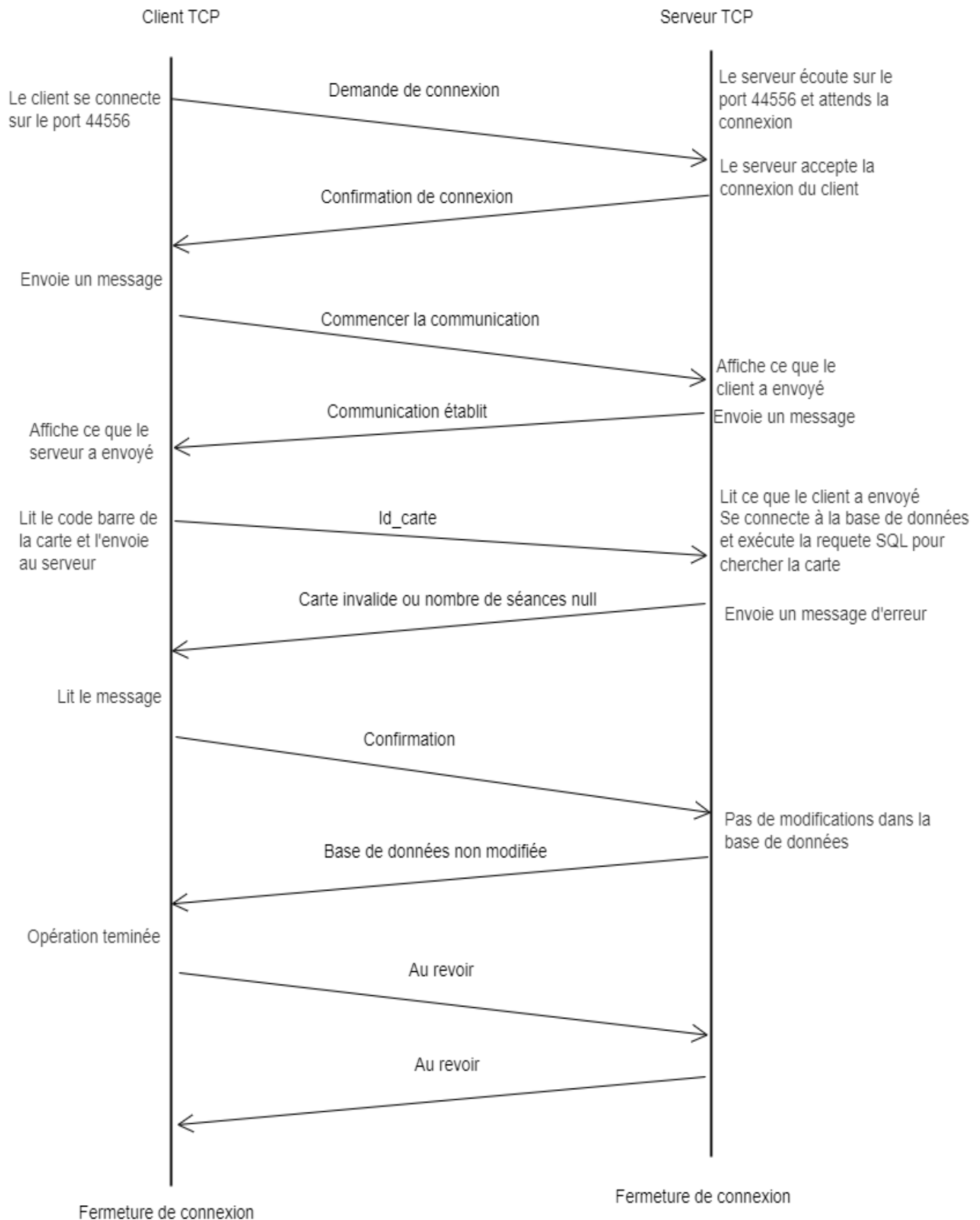


Figure 7 Schéma applicatif (carte invalide)

## 10. *Jeu de données*

**Table Film**

id_film	nom_film	date_film	lang_film	auteur_film	#id_genre
f00001	film1	20/10/2009	français	auteur1	g00001
f00002	film2	03/02/2020	chinois	auteur2	g00002
f00003	film3	01/01/1990	allemand	auteur3	g00003
f00004	film4	09/12/2001	arabe	auteur4	g00004

**Table Client**

id_client	datenai_client	fonction_client
c0000001	12/06/1965	Retraité
c0000002	04/03/1990	Jeune
c0000003	02/07/2010	Enfant
c0000004	06/10/2000	Étudiant

**Table Client\_passager**

id_client_passager
c0000001
c0000002
c0000003
c0000004

**Table Client\_abonne**

id_client_ab	nom_client	adresse_client	tel_client	mail_client
c0000001	thomas	23 avenue charle de gaulle 75000 ,Paris	0665656533	thomas@gmail.co m
c0000002	gabriel	20 rue franklin 91000,Evry	0767890987	gabriel@yahoo.fr
c0000003	feriel	21 rue saint martin 95000,Cergy	0689065432	feriel@outlook.fr
c0000004	eva	03 avenue de paris 94300,Vincennes	0654321909	eva@gmail.com



Table Seance

id_seance	horraire_seance	#id_salle	#id_film	#id_emp
se000001	2022-05-11 17:10:00+00	s00001	f00001	e00
se000002	2023-11-09 10:00:00+00	s00002	f00002	e002
se000003	2022-12-12 21:30:00+00	s00003	f00003	e003
se000004	2022-11-17 14:45:00+00	s00004	f00004	e004

Table salle

id_salle	nom_salle	num_salle	capacite_salle	id_site
s00001	Gaumont	G213	200	si01
s00002	Pathé	P9500	150	si02
s00003	CGR	CRG2390	80	si03
s00004	Mégarama	M75130	50	si04

Table carte\_abonnement

id_carte	nom_carte	date_delivrance	date_debut_ab	date_fin_ab	type_ab	nb_seances	prix_ab
ca000001	carte1	01/03/2020	01/03/2020	01/04/2021	annuel	240	250
ca000002	carte2	29/10/2019	01/11/2019	01/11/2020	annuel	240	250
ca000003	carte3	04/04/2019	04/04/2019	04/05/2019	mensuel	20	80
ca000004	carte4	15/07/2022	15/08/2022	31/08/2022	semaine	5	25

Table ticket

id_ticket	nom_ticket	date_ticket	numplace_ticket	tarif_ticket	prix_ticket
t0000001	ticket1	12/12/2022	45	Étudiant	6 euros
t0000002	ticket2	02/03/2022	20	Standard	10 euros
t0000003	ticket3	17/07/2022	110	Jeune (-26ans)	8 euros
t0000004	ticket4	29/09/2022	2	Enfant	4 euros

Table employé

id_emp	nom_employé	salaire_employé	mail-employé	statut_employé	tel_employé	mot_de_passe
e001	chabane	1500,00	chabane@gmail.com	Agent de sécurité	075643321	\$2y\$10\$yw0V.s5...
e002	zakaria	2000,00	zakaria@yahoo.fr	Gérant	0645876520	\$2y\$10\$A/eh....
e003	fadma	740,00	fadma@yahoo.fr	Vendeuse	0765432102	\$2y\$10\$/5hb....
e004	malha	1000,00	malha@gmail.com	Employée polyvalente	0654300001	\$2y\$10\$hZVkh....

Table sites

id_site	nom_site	adresse	codePostal_site	ville_site	pays_site
si01	cinema1	23 rue de paris	7500	Paris	france
si02	cinema2	2 rue du Lendemain	95800	Cergy le haut	
si03	cinema3	2 rue gabriel	91000	Evry	france
si04	cinema4	13 boulevard saint denis	77000	Marne la vallée	

Table genre

id_genre	nom_genre
g00001	Comedie
g00002	Horreur
g00003	Romantique
g00004	Action

## 11. Requetes SQL significatives

- Extraire les clients qui ont assistés à la séance du 2022-05-11 à 17h10 :

```
SELECT cl.*
FROM client_abonne cl, carte_abonnement c, seance s
WHERE cl.id_client_ab = c.id_client_ab
AND c.id_seance = s.id_seance
AND s.horaire_seance = '2022-05-11 17:10:00';
```

```
rouas_projet_cinema=> SELECT cl.*
rouas_projet_cinema-> FROM client_abonne cl, carte_abonnement c, seance s
rouas_projet_cinema-> WHERE cl.id_client_ab = c.id_client_ab
rouas_projet_cinema-> AND c.id_seance = s.id_seance
rouas_projet_cinema-> AND s.horaire_seance = '17:10:00';
id_client_ab | nom_client | adress_client | mail_client | tel_client
-----+-----+-----+-----+-----
c0000001    | thomas    | 23 avenue charle de gaulle 75000 ,Paris | thomas@gmail.com | 0665656533
(1 ligne)
```

- Extraire les films qui se déroulent dans la salle CGR la séance du 17/11/2022 à 14h45 :

```
SELECT f.nom_film
FROM film f
INNER JOIN seance s
ON f.id_film = s.id_film
INNER JOIN salle sa
ON sa.id_salle = s.id_salle
WHERE s.horaire_seance = '2022-11-17 14:45:00'
AND sa.nom_salle LIKE 'CGR';
```

```
rouas_projet_cinema=> SELECT f.nom_film
rouas_projet_cinema-> FROM film f
rouas_projet_cinema-> INNER JOIN seance s
rouas_projet_cinema-> ON f.id_film = s.id_film
rouas_projet_cinema-> INNER JOIN salle sa
rouas_projet_cinema-> ON sa.id_salle = s.id_salle
rouas_projet_cinema-> WHERE s.horaire_seance = '2022-12-12 21:30:00'
rouas_projet_cinema-> AND sa.nom_salle LIKE 'CGR';
nom_film
-----
film3
(1 ligne)
```

- Extraire le nombre de clients abonnées qui ont assisté à la séance du 09/11/2023 à 10h :

```
SELECT COUNT(ca.id_client_ab) AS Clients_abonnees
FROM client_abonne ca, seance s, carte_abonnement c
WHERE ca.id_client_ab = c.id_client_ab
AND s.id_seance=c.id_seance
AND s.horaire_seance='2023-11-09 10:00:00';
```

```
rouas_projet_cinema=> SELECT COUNT(ca.id_client_ab) AS Clients_abonnees
rouas_projet_cinema-> FROM client_abonne ca, seance s, carte_abonnement c
rouas_projet_cinema-> WHERE ca.id_client_ab = c.id_client_ab
rouas_projet_cinema-> AND s.id_seance=c.id_seance
rouas_projet_cinema-> AND s.horaire_seance='2023-11-09 10:00:00';
clients_abonnees
-----
1
(1 ligne)
```

- Extraire le nombre de clients passagers qui ont assisté à la séance du 17/11/2022 à 14h45 :

```
SELECT COUNT(cp.id_client_pass) AS Clients_passager
FROM client_passager cp, seance s, ticket t
WHERE cp.id_client_pass = t.id_client_pass
AND s.id_seance=t.id_seance
AND s.horaire_seance ='2022-11-17 14:45:00';
```

```
rouas_projet_cinema=> SELECT COUNT(cp.id_client_pass) AS Clients_passager
rouas_projet_cinema-> FROM client_passager cp, seance s, ticket t
rouas_projet_cinema-> WHERE cp.id_client_pass = t.id_client_pass
rouas_projet_cinema-> AND s.id_seance=t.id_seance
rouas_projet_cinema-> AND s.horaire_seance ='2022-11-17 14:45:00';
clients_passager
-----
1
(1 ligne)
```

- Extraire toutes les informations de l'employé qui gagne le moins :

```
SELECT *
FROM employe WHERE salaire_emp = (
    SELECT MIN(salaire_emp)
    FROM employe);
```

```
rouas_projet_cinema=> SELECT *
rouas_projet_cinema-> FROM employe WHERE salaire_emp = (
rouas_projet_cinema(> SELECT MIN(salaire_emp)
rouas_projet_cinema(> FROM employe);
rouas_projet_cinema(>
id_emp | nom_emp | salaire_emp | statut_emp | tel_emp | mail_emp
-----+-----+-----+-----+-----+-----
e003   | fadma   | 740         | vendeuse   | 0765432102 | fadma@yahoo.fr
(1 ligne)
```

- Extraire les films projetés dans cinema1 :

```
SELECT f.nom_film
FROM film f
INNER JOIN seance s
ON s.id_film=f.id_film
INNER JOIN salle sa
ON sa.id_salle=s.id_salle
INNER JOIN sites st
ON st.id_site = sa.id_site
WHERE st.nom_site LIKE 'cinema1';
```

```
rouas_projet_cinema=> SELECT f.nom_film
rouas_projet_cinema-> FROM film f
rouas_projet_cinema-> INNER JOIN seance s
rouas_projet_cinema-> ON s.id_film=f.id_film
rouas_projet_cinema-> INNER JOIN salle sa
rouas_projet_cinema-> ON sa.id_salle=s.id_salle
rouas_projet_cinema-> INNER JOIN sites st
rouas_projet_cinema-> ON st.id_site = sa.id_site
rouas_projet_cinema-> WHERE st.nom_site LIKE 'cinema1';
nom_film
-----
film1
film2
(2 lignes)
```

- Extraire les employés responsables du nettoyage de la salle Gaumont :

```
SELECT e.nom_emp
FROM employe e
INNER JOIN seance s ON e.id_emp=s.id_emp
INNER JOIN salle sa ON sa.id_salle=s.id_salle
WHERE sa.nom_salle LIKE 'Gaumont';
```

```
rouas_projet_cinema=> SELECT e.nom_emp
rouas_projet_cinema-> FROM employe e
rouas_projet_cinema-> INNER JOIN seance s ON e.id_emp=s.id_emp
rouas_projet_cinema-> INNER JOIN salle sa ON sa.id_salle=s.id_salle
rouas_projet_cinema-> WHERE sa.nom_salle LIKE 'Gaumont';
nom_emp
-----
chabane
(1 ligne)
```

- Extraire les films projetés dans le site cinema1 mais pas dans le site cinema2 :

```
SELECT DISTINCT f.nom_film
FROM film f
INNER JOIN seance se ON f.id_film=se.id_film
INNER JOIN salle sa ON sa.id_salle=se.id_salle
WHERE sa.id_site IN (SELECT id_site
                     FROM sites
                     WHERE nom_site LIKE 'cinema1')
AND sa.id_site NOT IN (SELECT id_site
                      FROM sites
                      WHERE nom_site LIKE 'cinema2');
```

```

rouas_projet_cinema=> SELECT DISTINCT f.nom_film
rouas_projet_cinema-> FROM film f
rouas_projet_cinema-> INNER JOIN seance se ON f.id_film=se.id_film
rouas_projet_cinema-> INNER JOIN salle sa ON sa.id_salle=sa.id_salle
rouas_projet_cinema-> WHERE sa.id_site IN (SELECT id_site
rouas_projet_cinema(>                                FROM sites
rouas_projet_cinema(>                                WHERE nom_site LIKE 'cinema1')
rouas_projet_cinema-> AND sa.id_site NOT IN (SELECT id_site
rouas_projet_cinema(>                                FROM sites
rouas_projet_cinema(>                                WHERE nom_site LIKE 'cinema2');
rouas_projet_cinema(>
rouas_projet_cinema(>                                nom_film
rouas_projet_cinema(>
rouas_projet_cinema(>                                -----
rouas_projet_cinema(>                                film3
rouas_projet_cinema(>                                film2
rouas_projet_cinema(>                                film4
rouas_projet_cinema(>                                film1
rouas_projet_cinema(>                                (4 lignes)

```

- Extraire pour chaque statut les employés responsables de la gestion de la séance du 17/11/2022 à 14H45 :

```

SELECT statut_emp, nom_emp
FROM employe e, seance s
WHERE e.id_emp=s.id_emp
AND s.horaire_seance='2022-11-17 14:45:00'
GROUP BY statut_emp, nom_emp;

```

```

rouas_projet_cinema=> SELECT statut_emp, nom_emp
rouas_projet_cinema-> FROM employe e, seance s
rouas_projet_cinema-> WHERE e.id_emp=s.id_emp
rouas_projet_cinema-> AND s.horaire_seance='2022-11-17 14:45:00'
rouas_projet_cinema-> GROUP BY statut_emp, nom_emp;
rouas_projet_cinema->
rouas_projet_cinema->      statut_emp      | nom_emp
rouas_projet_cinema-> -----+-----
rouas_projet_cinema-> employ0e polyvalente | malha
rouas_projet_cinema-> (1 ligne)

```

- Extraire le nombre de clients passager par jour :

```
SELECT (COUNT(id_ticket))
FROM ticket t, seance s
WHERE s.id_seance=t.id_seance
GROUP BY s.horaire_seance;
```

```
rouas_projet_cinema=> SELECT (COUNT(id_ticket))
rouas_projet_cinema-> FROM ticket t, seance s
rouas_projet_cinema-> WHERE s.id_seance=t.id_seance
rouas_projet_cinema-> GROUP BY s.horaire_seance;
count
-----
      2
      1
      2
(3 lignes)
```

- Extraire le nombre d'abonnements ajoutés l'an dernier :

```
SELECT COUNT(id_carte)
FROM carte_abonnement
WHERE date_delivrance BETWEEN '2020-01-01' AND '2020-12-31';
```

```
rouas_projet_cinema=> SELECT COUNT(id_carte)
rouas_projet_cinema-> FROM carte_abonnement
rouas_projet_cinema-> WHERE date_delivrance BETWEEN '2020-01-01' AND '2020-12-31';
count
-----
      1
(1 ligne)
```

- Modifier le nombre de séances du client abonné après qu'il assiste à une séance:

```
UPDATE carte_abonnement
SET nb_seance = nb_seance -1
WHERE id_client_ab='c00000002';
```

```
rouas_projet_cinema=> UPDATE carte_abonnement
rouas_projet_cinema-> SET nb_seance = nb_seance -1
rouas_projet_cinema-> WHERE id_client_ab='c00000002';
UPDATE 0
rouas_projet_cinema=>
```



- Un salaire d'un employé supérieur ou égale à 1000 euros, à condition que ce salaire soit supérieur à 1 :

```
SELECT salaire_emp, count(id_emp)
FROM employe WHERE salaire_emp >= 1000.00
GROUP BY salaire_emp
HAVING(count(id_emp) >= 1);
```

```
rouas_projet_cinema=> SELECT salaire_emp, count(id_emp)
rouas_projet_cinema-> FROM employe WHERE salaire_emp >= 1000.00
rouas_projet_cinema-> GROUP BY salaire_emp
rouas_projet_cinema-> HAVING(count(id_emp) >= 1);
salaire_emp | count
-----+-----
1000 | 1
1500 | 1
2000 | 1
(3 lignes)
```

- Ajouter un nouveau ticket, un client passager et un client abonné :

```
INSERT INTO client(id_client, datenai_client, fonction_client)
VALUES('c0000005', '2002-09-12', 'etudiant');
```

```
rouas_projet_cinema=> INSERT INTO client(id_client, datenai_client, fonction_client)
rouas_projet_cinema-> VALUES('c0000005', '2002-09-12', 'etudiant');
INSERT 0 1
rouas_projet_cinema=>
```

```
INSERT INTO client_passager (id_client_pass)
VALUES ('c0000005');
```

```
rouas_projet_cinema=> INSERT INTO client_passager (id_client_pass)
rouas_projet_cinema-> VALUES ('c0000005');
INSERT 0 1
rouas_projet_cinema=>
```

```
INSERT INTO Ticket (id_ticket, nom_ticket, date_ticket, tarif_ticket,
prix_ticket, numplace_ticket, id_client_pass, id_seance)
VALUES('t0000005', 'ticket5', 'NOW()', 'etudiant', 6, 37, 'c0000001', 'se0000
03');
```

```
rouas_projet_cinema-> VALUES('t0000005', 'ticket5', 'NOW()', 'etudiant', 6, 37, 'c0000001', 'se000003');
INSERT 0 1
rouas_projet_cinema=>
```