

**Project Εργαστηρίου
Βάσεων Δεδομένων
2021 – 2022**

ΟΝΟΜΑ: *Ιωάννης Καλδίρης*
ΑΜ: *1080428*
e-mail: *up1080428@upnet.gr*

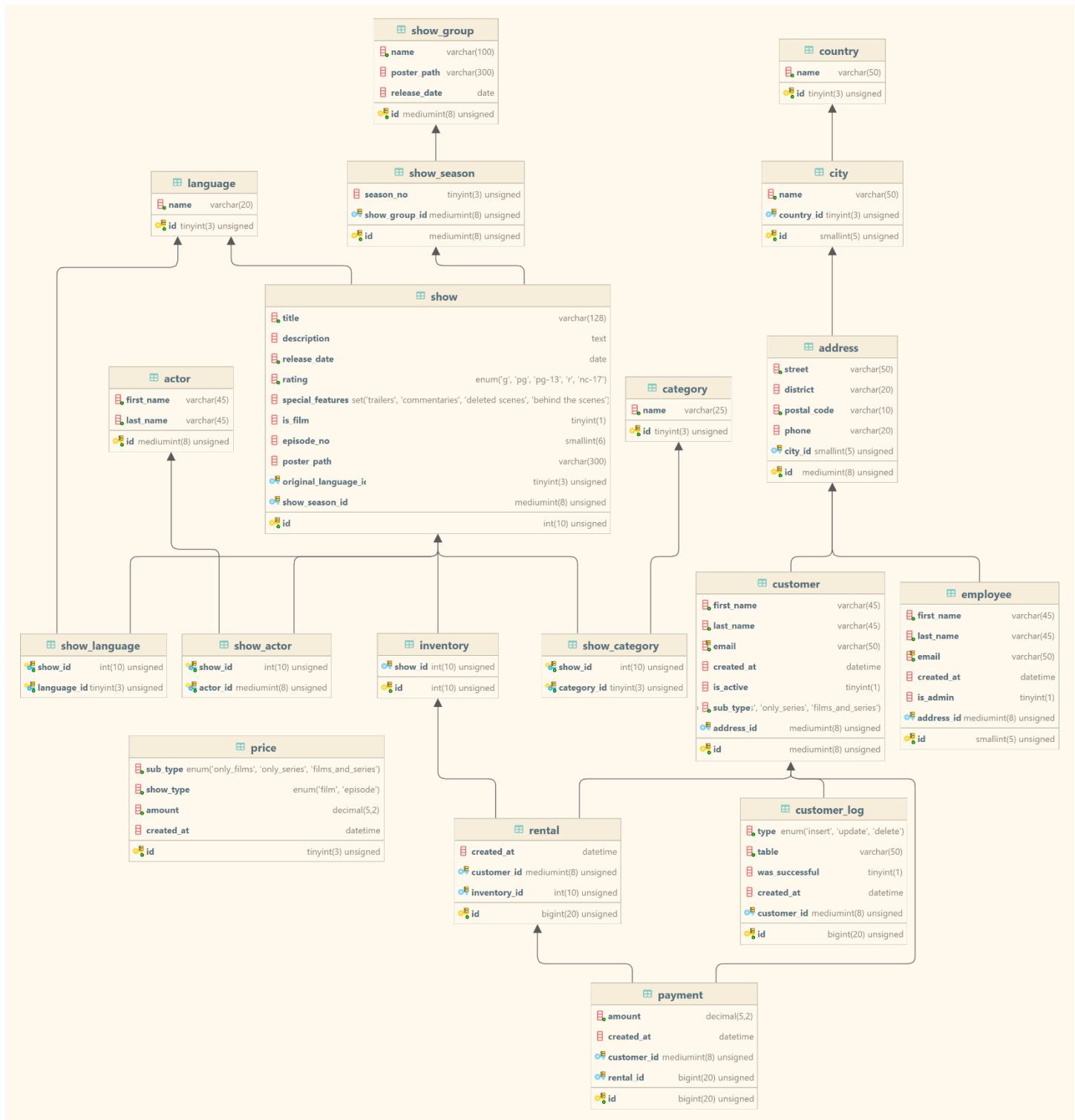
Περιεχόμενα

Μέρος Α: Σχεδιασμός ΒΔ και SQL.....	3
Κεφάλαιο 1: Σχεδίαση αναθεωρημένης ΒΔ.....	3
Α) Σχεσιακό Διάγραμμα (Υπάρχει αρχείο <i>er.svg</i> για μεγαλύτερη ευκρίνεια του διαγράμματος)	3
Β) Περιγραφή σχεδιασμού.....	4
Κεφάλαιο 2: Σχολιασμός νέων πινάκων	5
– <i>employee</i>	5
– <i>show</i>	5
<i>show_group</i>	5
<i>show_season</i>	5
– <i>show_language</i>	5
– <i>price</i>	5
– <i>customer_log</i>	5
Κεφάλαιο 3: Stored Procedures.....	6
1. Εμφάνιση συγκεκριμένου αριθμού ταινιών ή σειρών με τις περισσότερες ενοικιάσεις μέσα σε ένα χρονικό διάστημα.	6
2. Αριθμός ενοικιάσεων πελάτη για μία ημερομηνία.	7
3. Έσοδα από πληρωμές ανά μήνα.	8
4. Procedures Ηθοποιών	9
5. Λοιπά Stored Procedures.....	11
Κεφάλαιο 4: Triggers	12
1. Καταγραφή ενεργειών στον πίνακα <i>customer_log</i>	12
2. Έκπτωση σε ενοικίαση του χρήστη.	13
3. Απαγόρευση αλλαγής στοιχείων πελάτη.	14
4. Λοιπά Triggers	15
Υποσημείωση	15
Μέρος Β: GUIs	16
Κεφάλαιο 1: Διεπαφές.....	16
– Πελάτης (<i>customer</i>).....	17
– Υπάλληλος (<i>employee</i>).....	23
– Διαχειριστής (<i>administrator</i>)	25

Μέρος Α: Σχεδιασμός ΒΔ και SQL

Κεφάλαιο 1: Σχεδίαση αναθεωρημένης ΒΔ

A) Σχεσιακό Διάγραμμα (Υπάρχει αρχείο er.svg για μεγαλύτερη ευκρίνεια του διαγράμματος)



B) Περιγραφή σχεδιασμού

- Μετονομασία μεταβλητών και αλλαγή τύπων σε μερικούς από τους αρχικούς πίνακες.
- Προσθήκη νέου είδους χρήστη για τους υπαλλήλους με πληροφορία για το εάν είναι διαχειριστής ή απλός υπάλληλος.
- Αντικατάσταση του πίνακα film με τον πίνακα **show** ο οποίος αντιπροσωπεύει εκτός από ταινίες και επεισόδια σειρών.
- Δημιουργία πινάκων που αντιπροσωπεύουν ομάδες **show**. Αυτό μπορεί να είναι μία σειρά ή ένα σεντ ταινιών όπως π.χ. sequel, prequel, Marvel Cinematic Universe κτλ. Αντίστοιχα αποθηκεύουμε και υπό-ομάδες για κάθε ένα από αυτά, π.χ. σεζόν σειρών, πολλά sequel μίας ταινίας μαζί κτλ.
- Δημιουργία σχέσης **many-to-many** μεταξύ content και language γιατί οι εκπομπές μπορεί να προσφέρονται σε πολλές γλώσσες.
- Προσθήκη πίνακα **price** ο οποίος κρατά το κόστος για κάθε ενοικίαση ανάλογα με τη συνδρομή του χρήστη. Ο πίνακας αυτός λειτουργεί και σαν ιστορικό τιμών αφού για κάθε αλλαγή τιμής δημιουργείται νέα γραμμή στο πίνακα.
- Προσθήκη πίνακα **customer_log** που καταγράφει τις ενέργειες που εκτελεί κάποιος πελάτης πάνω σε κάποιο πίνακα κατά την ενοικίαση ταινιών. Μπορεί εύκολα να δημιουργηθεί πίνακας **employee_log** εάν θέλουμε να αποθηκεύσουμε ιστορικό ενεργειών για τους υπαλλήλους.

Κεφάλαιο 2: Σχολιασμός νέων πινάκων

– *employee*

Για κάθε **υπάλληλο** αποθηκεύονται οι ίδιες πληροφορίες με έναν πελάτη καθώς και εάν είναι διαχειριστής ή όχι.

– *show*

Αυτός ο πίνακας αντιπροσωπεύει είτε **ταινίες** είτε **επεισόδια** σειρών. Αποθηκεύονται ο τίτλος, η περιγραφή, η ημερομηνία κυκλοφορίας, το μήκος, η καταλληλότητα, εάν είναι κάποιο ιδιαίτερο είδος εκπομπής, εάν είναι ταινία, ο αριθμός επεισοδίου, link για την φωτογραφία της αφίσας εάν υπάρχει, η γλώσσα στην οποία κυκλοφόρησε αρχικά η εκπομπή και πληροφορίες για ομάδες που μπορεί να εμπλέκεται (στους πίνακες **show_season** και **show_group**).

show_group

Αποθηκεύει ομάδες εκπομπών στις οποίες μπορεί να ανήκουν επεισόδια ή ταινίες. Αντιπροσωπεύει δηλαδή πράγματα όπως σειρές, σετ ταινιών όπως (prequel, sequel) ή όπως το Marvel Cinematic Universe, τα οποία μπορούν να προσφερθούν στο χρήστη σαν ιδέες. Τα δεδομένα που αποθηκεύονται είναι ο τίτλος της ομάδας, η ημερομηνία δημιουργίας της εάν υπάρχει και link για τη φωτογραφία της αφίσας εάν υπάρχει.

show_season

Αποθηκεύει υπό-ομάδες του πίνακα **show_group**. Αυτό μπορεί να είναι σεζόν σειρών ή ομαδοποιημένα τα sequel μίας ταινίας, ή το Phase 1 του Marvel Cinematic Universe κτλ. Τα δεδομένα που αποθηκεύονται είναι ο αριθμός της σεζόν.

– *show_language*

Πίνακας για σχέση many-to-many μεταξύ των πινάκων **show** και **language** που αντιπροσωπεύει όλες τις γλώσσες που προσφέρεται μία εκπομπή.

– *price*

Διατηρεί πληροφορίες για τις τιμές των ενοικιάσεων ανάλογα με τον τύπο εγγραφής του χρήστη. Αποθηκεύει τον τύπο εγγραφής καθώς και το κόστος είτε για ταινίες είτε για επεισόδια ανάλογα με αυτόν, το κόστος της ενοικίασης, καθώς και ημερομηνία που δημιουργήθηκε αυτή η τιμή. Με αυτό το τρόπο μπορούμε να έχουμε και ένα ιστορικό για τις τιμές στην πλατφόρμα.

– *customer_log*

Για κάθε ενέργεια που εμπλέκεται στη διαδικασία ενοικίασης μίας εκπομπής αποθηκεύεται ο τύπος ενέργειας (INSERT, UPDATE, DELETE), ο πίνακας τον οποίο αφορά η ενέργεια, εάν ήταν επιτυχής, η ημερομηνία της και ο πελάτης που την έκανε.

Κεφάλαιο 3: Stored Procedures

1. Εμφάνιση συγκεκριμένου αριθμού ταινιών ή σειρών με τις περισσότερες ενοικιάσεις μέσα σε ένα χρονικό διάστημα.

Κώδικας:

```
DELIMITER $$
DROP PROCEDURE IF EXISTS shows_with_most_rentals$$
CREATE PROCEDURE shows_with_most_rentals(
  IN type ENUM('m', 's'),
  IN quantity INT UNSIGNED,
  IN start_date DATE,
  IN end_date DATE
)
BEGIN
  IF type = 'm' THEN
    SELECT s.id, s.title, COUNT(s.id) AS amount FROM rental r
    LEFT JOIN inventory i ON r.inventory_id = i.id
    LEFT JOIN `show` s ON i.show_id = s.id
    WHERE s.is_film = TRUE AND r.created_at BETWEEN start_date AND end_date
    GROUP BY s.id
    ORDER BY amount DESC
    LIMIT quantity;
  ELSEIF type = 's' THEN
    SELECT sg.id, sg.name, COUNT(sg.id) AS amount FROM rental r
    LEFT JOIN inventory i ON r.inventory_id = i.id
    LEFT JOIN `show` s ON i.show_id = s.id
    LEFT JOIN show_season ss ON s.show_season_id = ss.id
    LEFT JOIN show_group sg ON ss.show_group_id = sg.id
    WHERE s.is_film = FALSE AND r.created_at BETWEEN start_date AND end_date
    GROUP BY sg.id
    ORDER BY amount DESC
    LIMIT quantity;
  END IF;
END;$$
DELIMITER ;
```

Εκτέλεση:

```
CALL shows_with_most_rentals('m', 10, '2021-04-01', '2022-01-01');
```

	id	title	amount
1	4	Schindler"s List	3
2	6	Your Name.	3
3	16	The Empire Strikes Back	3
4	3	The Shawshank Redemption	2
5	8	The Green Mile	2
6	10	Pulp Fiction	2
7	15	A Silent Voice: The Movie	2
8	18	Demon Slayer -Kimetsu no Yaiba- The Movie: Mugen Train	2
9	20	Avengers: Endgame	2
10	1	Black Panther: Wakanda Forever	1

```
CALL shows_with_most_rentals('s', 5, '2019-10-16', '2020-03-04');
```

	id	name	amount
1	8	Game of Thrones	196
2	6	Breaking Bad	51
3	9	Fullmetal Alchemist: Brotherhood	47
4	4	Peaky Blinders	36
5	7	The Lord of the Rings: The Rings of Power	17

2. Αριθμός ενοικιάσεων πελάτη για μία ημερομηνία.

Κώδικας:

```
DELIMITER $$  
DROP PROCEDURE IF EXISTS customer_rentals$$  
CREATE PROCEDURE customer_rentals(  
    IN in_email VARCHAR(50),  
    IN in_date DATE,  
    OUT quantity SMALLINT UNSIGNED  
)  
BEGIN  
    SELECT COUNT(r.id) INTO quantity FROM rental r  
    LEFT JOIN customer c on r.customer_id = c.id  
    WHERE c.email = in_email AND DATE(r.created_at) = in_date;  
END;$$  
DELIMITER ;
```

Εκτέλεση:

```
CALL customer_rentals('customer@mail.com', '2020-11-27', @amount);  
SELECT @amount;
```

	@amount
1	3

3. Έσοδα από πληρωμές ανά μήνα.

Κώδικας:

```
DELIMITER $$
DROP PROCEDURE IF EXISTS sales_per_month$$
CREATE PROCEDURE sales_per_month()
BEGIN
    SELECT YEAR(r.created_at) AS yr, MONTH(r.created_at) AS mn,
    SUM(p.amount) AS total FROM payment p
    LEFT JOIN rental r ON r.id = p.rental_id
    LEFT JOIN inventory i ON r.inventory_id = i.id
    LEFT JOIN `show` s ON i.show_id = s.id
    WHERE s.is_film = TRUE
    GROUP BY MONTH(r.created_at)
    ORDER BY r.created_at DESC;

    SELECT YEAR(r.created_at) AS yr, MONTH(r.created_at) AS mn,
    SUM(p.amount) AS total FROM payment p
    LEFT JOIN rental r ON r.id = p.rental_id
    LEFT JOIN inventory i ON r.inventory_id = i.id
    LEFT JOIN `show` s ON i.show_id = s.id
    WHERE s.is_film = FALSE
    GROUP BY MONTH(r.created_at)
    ORDER BY r.created_at DESC;
END;$$
DELIMITER ;
```

Εκτέλεση:

```
CALL sales_per_month()
```

	yr	mn	f_sales
1	2022	11	7.35
2	2022	10	4.20
3	2021	5	5.40
4	2020	12	6.00
5	2020	8	3.90
6	2019	9	4.20
7	2019	7	6.90
8	2019	6	3.60
9	2019	4	4.80
10	2019	1	4.80
11	2018	3	3.00
12	2018	2	4.50

	yr	mn	s_sales
1	2022	8	40.50
2	2022	7	44.40
3	2022	5	39.80
4	2022	4	39.90
5	2020	11	32.70
6	2020	6	38.30
7	2020	2	38.60
8	2020	1	41.50
9	2019	12	31.90
10	2019	9	43.10
11	2018	10	43.10
12	2018	3	43.80

4. Procedures Ηθοποιών

Δημιουργήθηκε **index** στη στήλη **last_name** του πίνακα **actor** αφού θα γίνουν πολλές αναζητήσεις με βάση αυτό.

```
CREATE INDEX actor_last_name_index ON actor(last_name);
```

α) Επώνυμα ηθοποιών μεταξύ δύο παραμέτρων.

Κώδικας:

```
DELIMITER $$
DROP PROCEDURE IF EXISTS actors_in_range$$
CREATE PROCEDURE actors_in_range(
    IN start_name VARCHAR(50),
    IN end_name VARCHAR(50),
    OUT quantity MEDIUMINT UNSIGNED
)
BEGIN
    SELECT first_name, last_name FROM actor
    WHERE last_name BETWEEN start_name AND end_name;

    SELECT COUNT(last_name) INTO quantity FROM actor
    WHERE last_name BETWEEN start_name AND end_name;
END;$$
DELIMITER ;
```

Εκτέλεση:

```
CALL actors_in_range('Aco', 'Adk', @quantity);
SELECT @quantity;
```

	first_name	last_name
6	Cooper	Acosta
7	Jamiya	Acosta
8	Junior	Acosta
9	Keagan	Acosta
10	Christina	Acosta
11	Howard	Acosta
12	Dustin	Adams
13	Keyon	Adams
14	Alexandria	Adams
15	Jordyn	Adams
16	Monique	Adams
17	Abram	Adams
18	Esther	Adams

	@quantity
1	18

b) Ηθθοποιόί με το ίδιο επώνυμο.

Κώδικας:

```
DELIMITER $$
DROP PROCEDURE IF EXISTS actors_with_common_last_name$$
CREATE PROCEDURE actors_with_common_last_name(
  IN in_name VARCHAR(50),
  OUT quantity SMALLINT UNSIGNED
)
BEGIN
  SELECT first_name, last_name FROM actor
  WHERE last_name = in_name;

  SELECT COUNT(last_name) INTO quantity FROM actor
  WHERE last_name = in_name;
END;$$
DELIMITER ;
```

Εκτέλεση:

```
CALL actors_with_common_last_name('Acosta', @quantity);
SELECT @quantity;
```

	first_name	last_name
1	Pranav	Acosta
2	Aydan	Acosta
3	Jazmyn	Acosta
4	Raiden	Acosta
5	Krystal	Acosta
6	Cooper	Acosta
7	Jamiya	Acosta
8	Junior	Acosta
9	Keagan	Acosta
10	Christina	Acosta
11	Howard	Acosta

	@quantity
1	11

5. Λοιπά Stored Procedures

- Υπολογισμός τιμής ενοικίασης για κάποιον χρήστη.

Κώδικας:

```
DELIMITER $$
DROP PROCEDURE IF EXISTS calculate_rental_price$$
CREATE PROCEDURE calculate_rental_price(
  IN customer_id MEDIUMINT UNSIGNED,
  IN in_show_type ENUM('FILM', 'EPISODE'),
  OUT payment_amount DECIMAL(5, 2)
)
BEGIN
  SELECT amount INTO payment_amount FROM price
  WHERE sub_type = (SELECT sub_type FROM customer WHERE id = customer_id)
  AND show_type = in_show_type
  ORDER BY created_at DESC LIMIT 1;
END;$$
DELIMITER ;
```

Εκτέλεση:

Αυτό το procedure καλείται αυτόματα από το trigger **create_payment_for_rental** το οποίο καλείται μετά από εισαγωγή ενοικίασης στον πίνακα rental. Διαβάζει το είδος εγγραφής που έχει ο χρήστης και επιλέγει τη σωστή τιμή από τον πίνακα **price**.

```
CALL calculate_rental_price(1, 'FILM', @amount);
SELECT FORMAT(@amount, 2);
```

	FORMAT(@amount, 2)
1	0.30

```
CALL calculate_rental_price(1, 'EPISODE', @amount);
SELECT FORMAT(@amount, 2);
```

	FORMAT(@amount, 2)
1	0.10

Κεφάλαιο 4: Triggers

1. Καταγραφή ενεργειών στον πίνακα customer_log.

Οι πίνακες οι οποίοι εμπλέκονται στη διαδικασία ενοικίασης και θέασης κάποιου είδους content δεν έχουν αλλάξει οπότε οι πίνακες που θα δημιουργούν logs θα είναι οι **rental** και **payment**.

– rental

Κώδικας:

```
DELIMITER $$
DROP TRIGGER IF EXISTS rental_log$$
CREATE TRIGGER rental_log AFTER INSERT ON rental
FOR EACH ROW
BEGIN
    INSERT INTO customer_log (type, `table`, customer_id, created_at) VALUES
    ('INSERT', 'rental', NEW.customer_id, NEW.created_at);
END;$$
DELIMITER ;
```

– payment







Κώδικας:

```
DELIMITER $$
DROP TRIGGER IF EXISTS payment_log$$
CREATE TRIGGER payment_log AFTER INSERT ON payment
FOR EACH ROW
BEGIN
    INSERT INTO customer_log (type, `table`, customer_id, created_at) VALUES
    ('INSERT', 'payment', NEW.customer_id, NEW.created_at);
END;$$
DELIMITER ;
```

Εκτέλεση:

Υπενθύμιση πως με εισαγωγή ενοικίασης στον πίνακα **rental** γίνεται και αυτόματα εισαγωγή στον πίνακα **payment** με τη σωστή χρέωση η οποία υπολογίζεται και αυτή αυτόματα.

Αρκεί να δείτε πως στην εισαγωγή των δεδομένων δεν έχουμε κάποια εγγραφή στον πίνακα **customer_log** αλλά όταν γίνει η εισαγωγή των δεδομένων με ένα SELECT στον πίνακα **customer_log** βλέπουμε πως έχουν εισαχθεί γραμμές.

	 id	 type	 table	 was_successful	 created_at	 customer_id
1	1	INSERT	rental	1	2022-04-23 00:00:00	29
2	2	INSERT	payment	1	2022-04-23 00:00:00	29
3	3	INSERT	rental	1	2022-07-04 00:00:00	29
4	4	INSERT	payment	1	2022-07-04 00:00:00	29
5	5	INSERT	rental	1	2020-07-06 00:00:00	29
6	6	INSERT	payment	1	2020-07-06 00:00:00	29
7	7	INSERT	rental	1	2019-12-31 00:00:00	29
8	8	INSERT	payment	1	2019-12-31 00:00:00	29

2. Έκπτωση σε ενοικίαση του χρήστη.

Κώδικας:

```
DELIMITER $$
DROP TRIGGER IF EXISTS discount_payment$$
CREATE TRIGGER discount_payment BEFORE INSERT ON payment
FOR EACH ROW
BEGIN
    DECLARE l_email VARCHAR(50);
    DECLARE l_day_rentals SMALLINT UNSIGNED;











    SELECT email INTO l_email FROM customer WHERE id = NEW.customer_id;
    CALL customer_rentals(l_email, NOW(), l_day_rentals);

    IF l_day_rentals > 0 AND MOD(l_day_rentals, 3) = 0 THEN
        SET NEW.amount = NEW.amount / 2;
    END IF;
END;$$
DELIMITER ;
```

Εκτέλεση:

Υπενθύμιση πως με εισαγωγή ενοικίασης στον πίνακα **rental** γίνεται και αυτόματα εισαγωγή στον πίνακα **payment**.

```
INSERT INTO rental (customer_id, inventory_id)
VALUES
(1, 133),
(1, 36),
(1, 55),
(1, 133),
(1, 36),
(1, 55),
(1, 133),
(1, 36),
(1, 55);
```

	 id  1	 amount 	 created_at 	 customer_id 	 rental_id 
1	4967	0.15	2022-11-17 20:15:58	1	4967
2	4966	0.30	2022-11-17 20:15:58	1	4966
3	4965	0.30	2022-11-17 20:15:58	1	4965
4	4964	0.15	2022-11-17 20:13:53	1	4964
5	4963	0.30	2022-11-17 20:13:53	1	4963
6	4962	0.30	2022-11-17 20:13:53	1	4962
7	4961	0.15	2022-11-17 20:13:31	1	4961
8	4960	0.30	2022-11-17 20:13:31	1	4960
9	4959	0.30	2022-11-17 20:13:31	1	4959

3. Απαγόρευση αλλαγής στοιχείων πελάτη.

Κώδικας:

```
DELIMITER $$
DROP TRIGGER IF EXISTS valid_customer_update$$
CREATE TRIGGER valid_customer_update BEFORE UPDATE ON customer
FOR EACH ROW
BEGIN
    IF NEW.id != OLD.id OR NEW.email != OLD.email OR NEW.created_at != OLD.created_at THEN
        SIGNAL SQLSTATE '45000' SET MESSAGE_TEXT = 'Unauthorized data change.';
    END IF;
END;$$
DELIMITER ;
```

Εκτέλεση:

```
UPDATE customer SET email = 'new_email@mail.com' WHERE id = 1;
```

[45000][1644] (conn=35) Unauthorized data change.

4. Λοιπά Triggers

- Δημιουργία **payment** μετά από εισαγωγή ενοικίασης στον πίνακα **rental**.

Αυτό το trigger καλεί το stored procedure **pay_rental** το οποίο βρίσκει το είδος εγγραφής του χρήστη καθώς και το είδος του content που θέλει να ενοικιάσει και το αντίστοιχο κόστος της ενοικίασης.

Κώδικας:

```
DELIMITER $$
DROP TRIGGER IF EXISTS create_payment_for_rental$$
CREATE TRIGGER create_payment_for_rental AFTER INSERT ON rental
FOR EACH ROW
BEGIN
    DECLARE l_show_type ENUM('FILM', 'EPISODE');
    DECLARE payment_amount DECIMAL(5, 2);

    IF (
        SELECT s.is_film FROM rental r
        LEFT JOIN inventory i ON r.inventory_id = i.id
        LEFT JOIN `show` s ON i.show_id = s.id
        WHERE r.id = NEW.id
    ) THEN
        SET l_show_type = 'FILM';
    ELSE
        SET l_show_type = 'EPISODE';
    END IF;

    CALL calculate_rental_price(NEW.customer_id, l_show_type, payment_amount);

    INSERT INTO payment (amount, customer_id, rental_id, created_at) VALUES
    (payment_amount, NEW.customer_id, NEW.id, NEW.created_at);
END;$$
DELIMITER ;
```

Εκτέλεση:

```
INSERT INTO rental (customer_id, inventory_id)
VALUES
    (1, 133),
    (1, 36),
    (1, 55);
```

	id	amount	created_at	customer_id	rental_id
1	1	0.30	2022-11-10 16:52:42	3	1
2	2	0.30	2022-11-10 16:52:42	3	2
3	3	0.15	2022-11-10 16:52:42	3	3

Υποσημείωση

Για τη δημιουργία δεδομένων για τις σειρές και τις ταινίες δημιουργήθηκε ένα script σε **python** το οποίο συνδέεται με το API του [The Movie Database](#) μιας πλατφόρμας που έχει πληροφορίες για διάφορες σειρές και ταινίες και μετασχηματίζει τα δεδομένα και τα αποθηκεύει σε αρχεία τα οποία μπορεί ο χρήστης να εισάγει στην βάση.

Documentation για το API μπορεί να βρεθεί [εδώ](#).

Εάν επιθυμήσετε να τρέξετε το script θα χρειαστείτε το library **requests** και συνιστάται να διαβάσετε τις οδηγίες στο αρχείο **main.py** για τη σωστή χρήση του εργαλείου.

Μέρος B: GUIs

Κεφάλαιο 1: Διεπαφές

Η εφαρμογή δημιουργήθηκε σαν Full-Stack με τη χρήση διάφορων τεχνολογιών. Πιο συγκεκριμένα:

- **Για το Front-End**

Για τη κατασκευή του **Front-End** χρησιμοποιήθηκε το JavaScript framework [React](#) το οποίο βασίζεται στην χρήση **components** που είναι μικρά submodules στην εκάστοτε ιστοσελίδα τα οποία μπορεί κανείς να τα κάνει develop ανεξάρτητα από τα υπόλοιπα δημιουργώντας έτσι ένα γρήγορο περιβάλλον development καθώς και επεκτάσιμο αφού τα αντικείμενα είναι ανεξάρτητα. Για παράδειγμα θα δείτε παρακάτω πως οι ταινίες και σειρές που εμφανίζονται στον χρήστη είναι απλά ένα component το οποίο το εμφανίζουμε πολλές φορές. Με αυτό το τρόπο ότι αλλαγές θέλουμε να κάνουμε στο μέλλον αρκεί να γίνουν μία φορά. Η εφαρμογές που είναι φτιαγμένες με JavaScript frameworks συνήθως καλούνται και **Single-Page Applications** και με καλό λόγο. Η συνηθισμένη διαδικασία είναι πως όταν θέλουμε να πάμε σε μία νέα σελίδα, στέλνουμε request στο backend και αυτό μας απαντά με μία νέα σελίδα html η κάτι αντίστοιχο. Τα JS Frameworks αντίθετα φορτώνουν μία σελίδα στην αρχή και στη συνέχεια request στο server κάνουν μόνο τα components που χρειάζονται νέα δεδομένα. Αυτό έχει το προνόμιο πως η σελίδα ποτέ δεν εκφορτώνεται οπότε ο χρήστης δεν χρειάζεται ποτέ να περιμένει να φορτώσει μία νέα σελίδα.

Την εφαρμογή την τρέχουμε πάνω σε [Vite](#) ένα webpack server ο οποίος τρέχει live και εμφανίζει στο website ότι αλλαγές κάνουμε στα αρχεία αμέσως.

Το framework **React** είναι barebones οπότε χρειάζεται μερικά libraries για να λειτουργήσει στο επίπεδο που θέλουμε εμείς. Πιο συγκεκριμένα:

- ο [Redux](#) – Βιβλιοθήκη που κάνει manage το **global state** της εφαρμογής. Θα τη χρησιμοποιήσουμε για να αποθηκεύσουμε τον χρήστη που είναι logged in έτσι ώστε να έχουμε πρόσβαση σε όλη την εφαρμογή.
- ο [React-Router](#) – Βιβλιοθήκη που ελέγχει το **routing** της εφαρμογής. Ελέγχει πιο component θα εμφανίζεται όταν πάμε σε ένα συγκεκριμένο URL, καθώς και τη μεταφορά από το ένα URL στο άλλο. Είναι σημαντικό γιατί όπως αναφέραμε παραπάνω δεν θέλουμε να κάνουμε reload τη σελίδα καθώς όλη η εφαρμογή τρέχει με JavaScript στο runtime.
- ο [Chakra UI](#) – Βιβλιοθήκη που προσφέρει pre-built components που βοηθούν σε γρηγορότερο development και ομοιοποίηση της εφαρμογής με τη χρήση κοινού style μεταξύ των component. Αυτό σημαίνει πως δεν χρειάζεται ποτέ να γράψουμε CSS εφόσον είμαστε ικανοποιημένοι με το styl της βιβλιοθήκης. Μερικά components που προσφέρει είναι layouts, κουμπιά, inputs κτλ. Για πλήρη λίστα μπορείτε να δείτε το documentation.

- **Για το Back-End**

Για τη κατασκευή του back-end χρησιμοποιήθηκαν προφανώς MySQL για την βάση καθώς και [NodeJS](#) για τα requests μεταξύ του Front-End και της βάσης. Ο κώδικας περιέχει πολλά routers μέσα στο οποία απλώς στέλνουμε request στη βάση και επιστρέφουμε το αποτέλεσμα στο Front-End για να το κάνει manage εκείνο. Χρησιμοποιήθηκαν τα παρακάτω libraries:

- ο [express](#) – Framework που χρησιμοποιούμε για να λάβουμε τα requests από το Front-End. Έχει routers στα οποία κάνουμε assign URL και είδος request (GET, POST, PUT, κτλ.) και μέσα σε αυτά μπορούμε να εκτελέσουμε ό,τι χρειαζόμαστε.
- ο [mysql](#) – Βιβλιοθήκη με την οποία μπορούμε να συνδεθούμε με μία βάση MySQL και να εκτελέσουμε queries.
- ο [cors](#) – Βιβλιοθήκη στην οποία βάζουμε restrictions για το ποια URL μπορεί ο server να δέχεται request. Απαραίτητο αφού το Front-End και το Back-End τρέχουν σε διαφορετικούς server.
- ο [nodemon](#) – Live server ο οποίος κάνει reload όταν αλλάζουμε κάτι σε ένα αρχείο.

Αρχικά στον χρήστη εμφανίζεται μία σελίδα για να κάνει login. Να σημειωθεί πως όλες οι υπόλοιπες σελίδες είναι κλειδωμένες οπότε και να γράψει το URL θα γίνει αυτόματα redirect στη σελίδα για login.

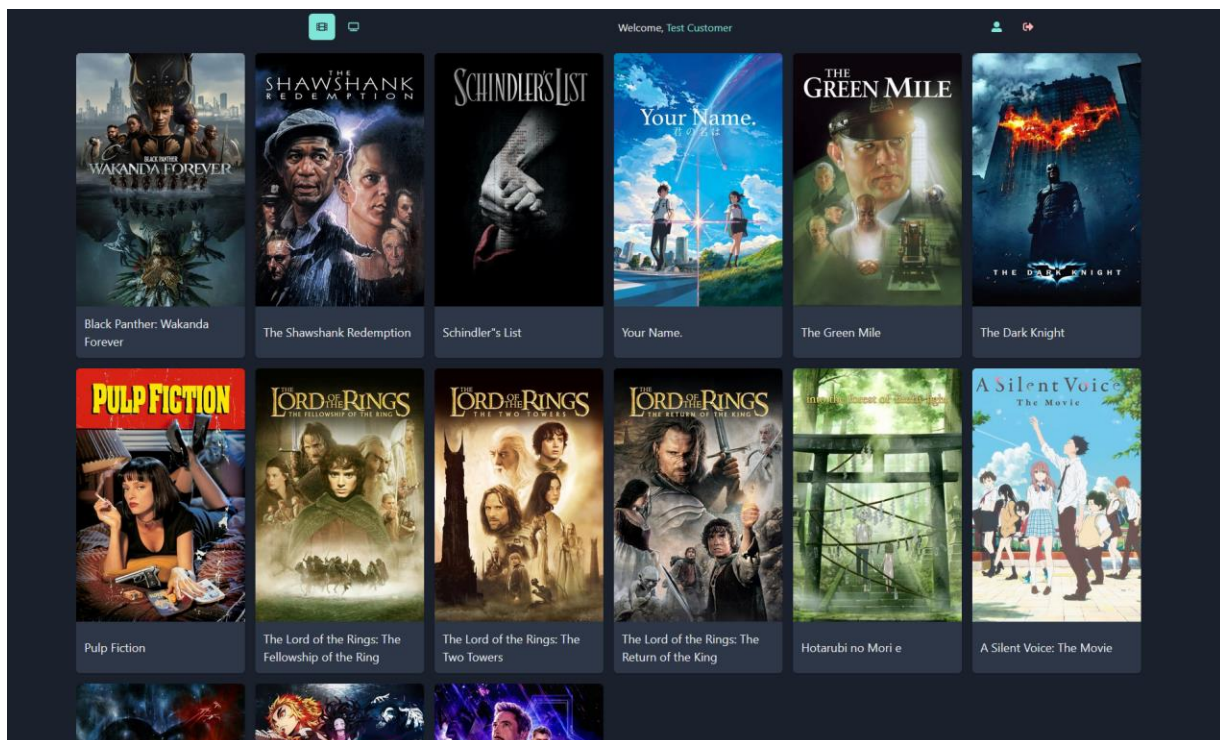


Ανάλογα με το email που εισάγει θα γίνει αυτόματα redirect στις σελίδες ενός **customer**, **employee** ή **admin**.

– **Πελάτης (customer)**

Στο πάνω μέρος εμφανίζεται navigation bar, χαιρετισμός στον χρήστη καθώς και κουμπί για logout.

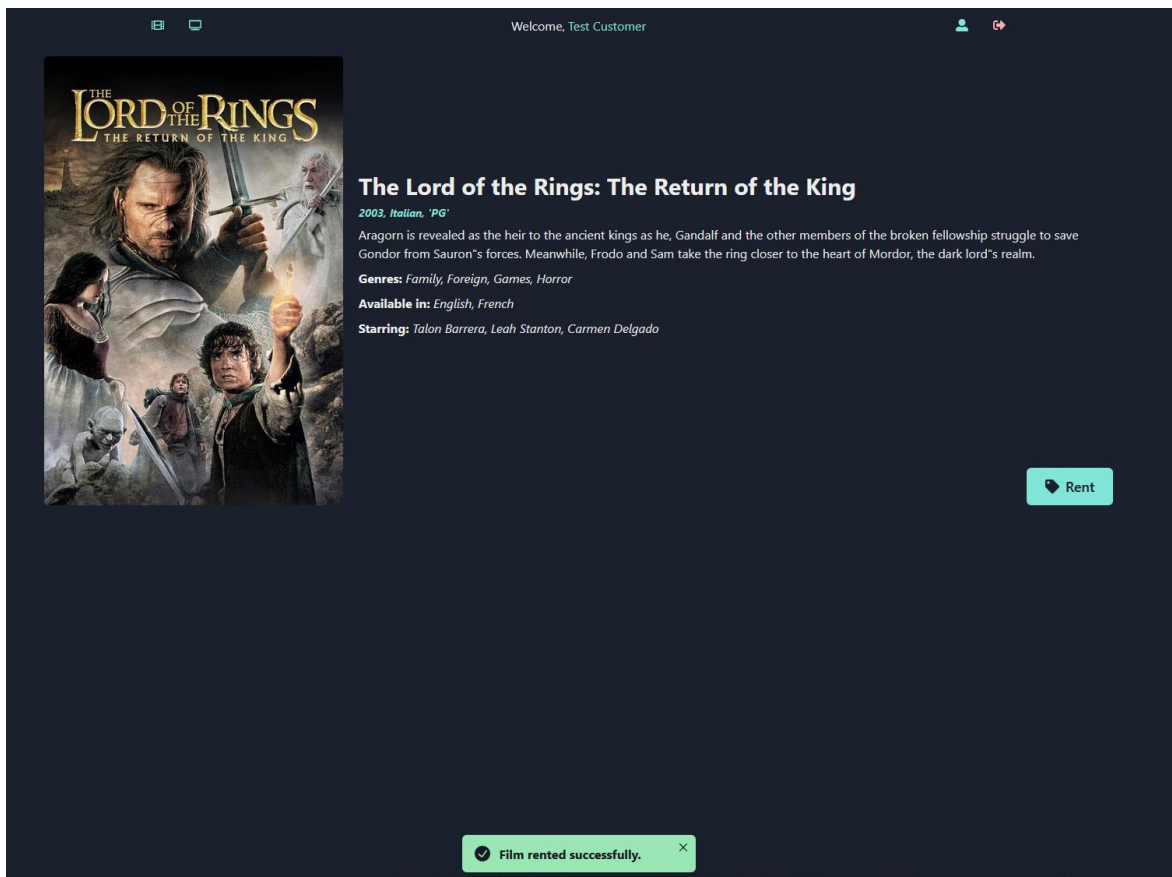
- Σελίδα **films**



Εμφανίζονται ταινίες σε κάρτες με τη χρήση του `poster_path` που πήραμε με τη χρήση του script που αναφέραμε στο Μέρος Α. Ο χρήστης μπορεί να πατήσει πάνω σε οποιαδήποτε από αυτές για να δει πληροφορίες και να κάνει μία ενοικίαση.



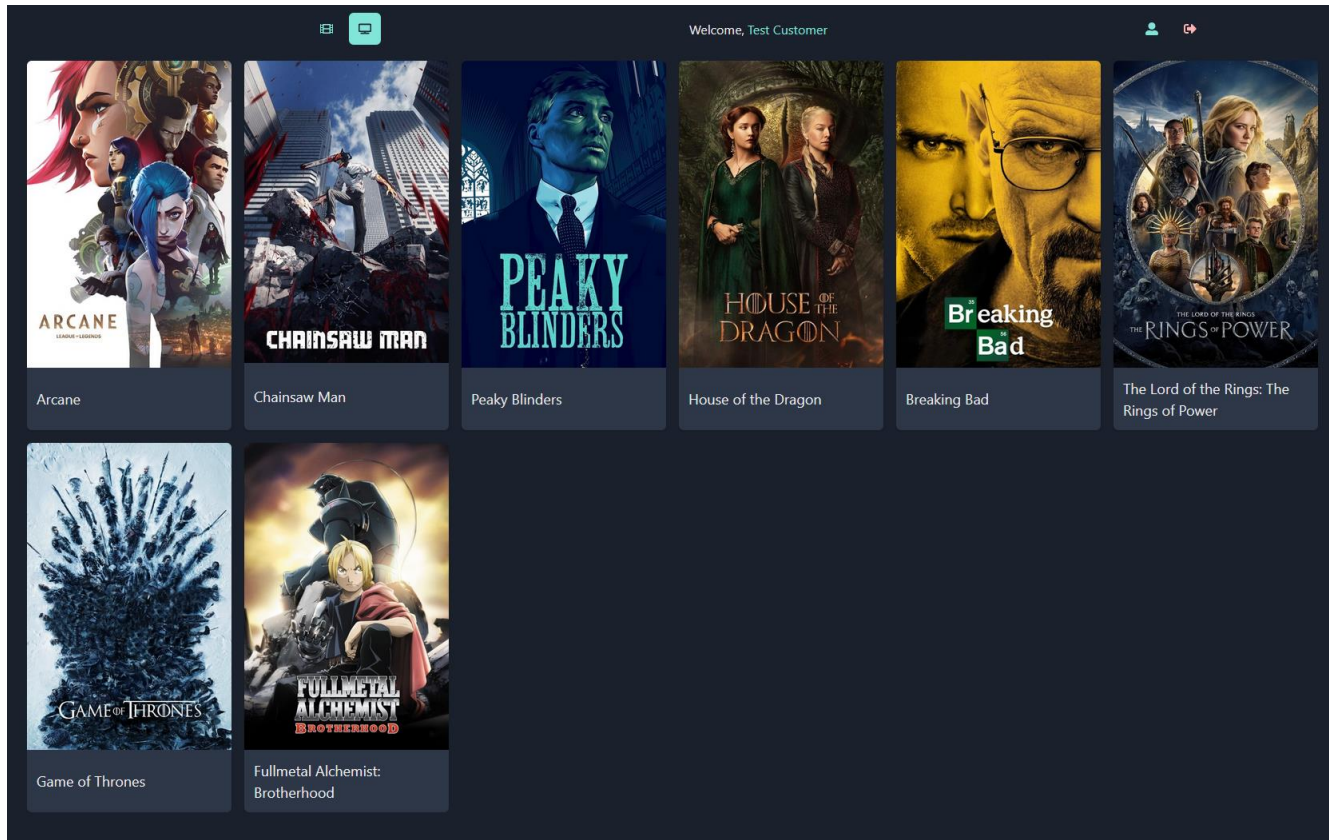
Οι πληροφορίες που εμφανίζονται υπάρχουν στην βάση και απλώς φορτώνονται για την αντίστοιχη ταινία όταν ο χρήστης κάνει navigate εκεί. Ο χρήστης αρκεί να πατήσει το κουμπί **Rent** για να δημιουργήσει ενοικίαση οπότε και εμφανίζεται αντίστοιχο feedback στο κάτω μέρος της σελίδας.



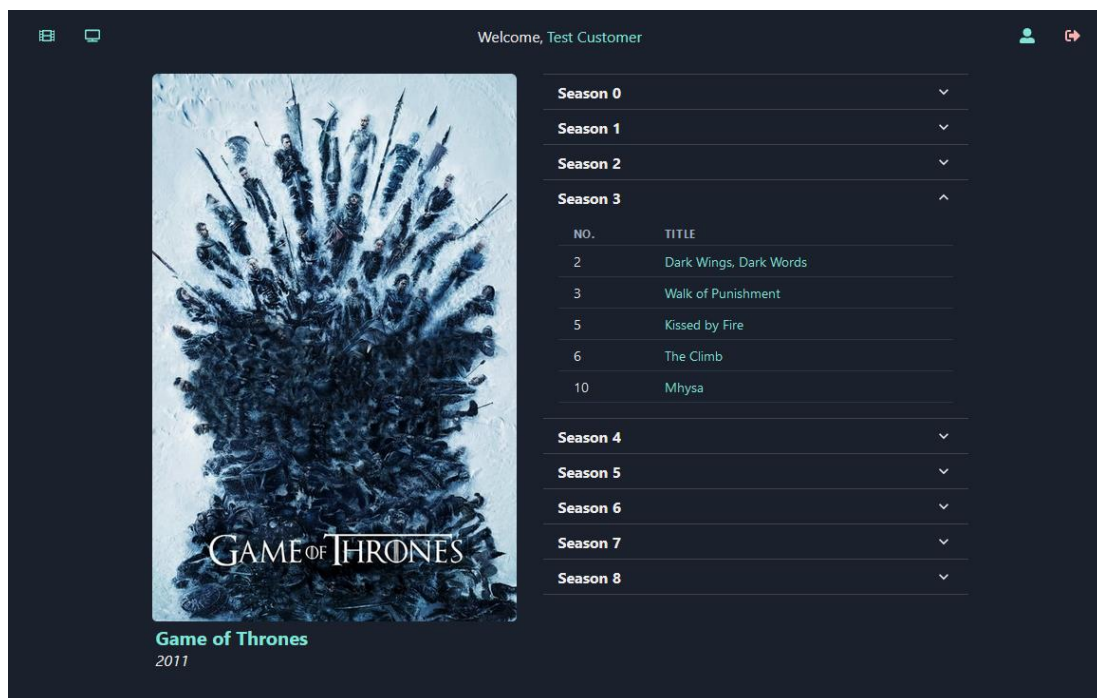
Παρακάτω θα δούμε την ενοικίαση στη σελίδα **profile**.

- Σελίδα **series**

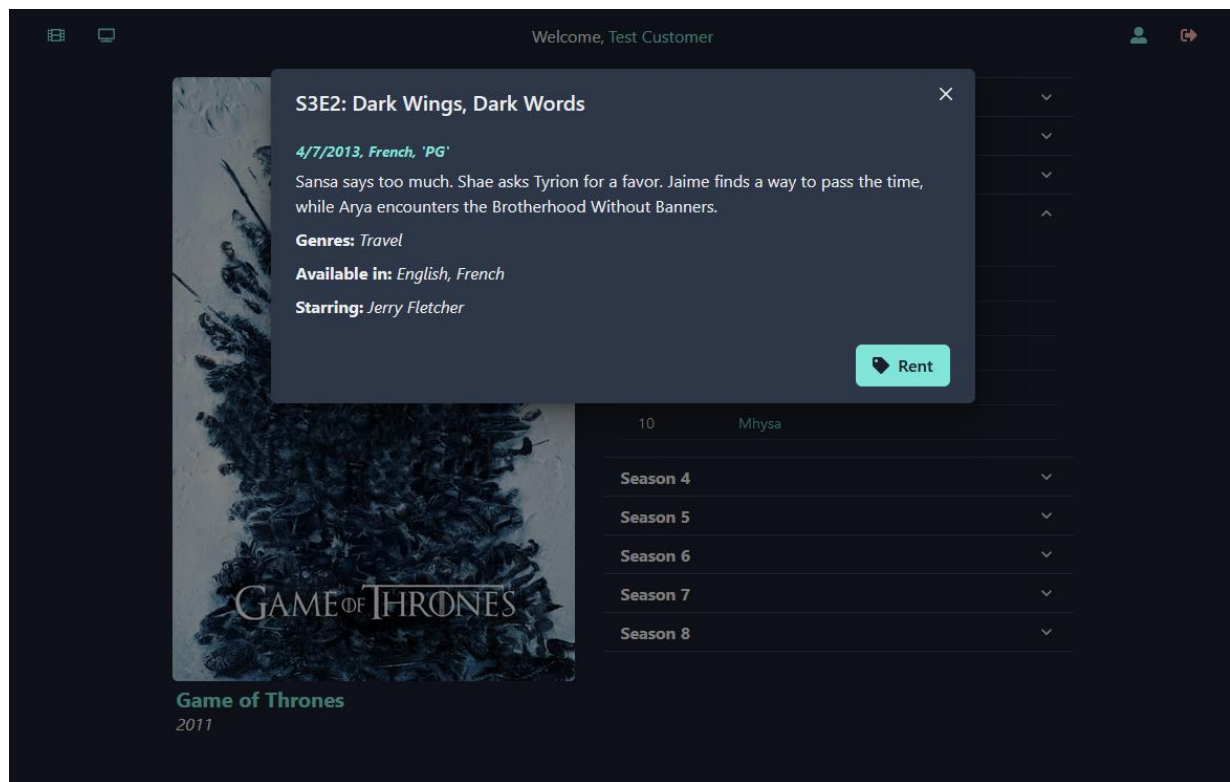
Ομοίως με τη σελίδα **films** εμφανίζεται λίστα με τις διαθέσιμες σειρές από τις οποίες ο χρήστης μπορεί να επιλέξει για να δει πληροφορίες για της σεζόν και τα επεισόδια.



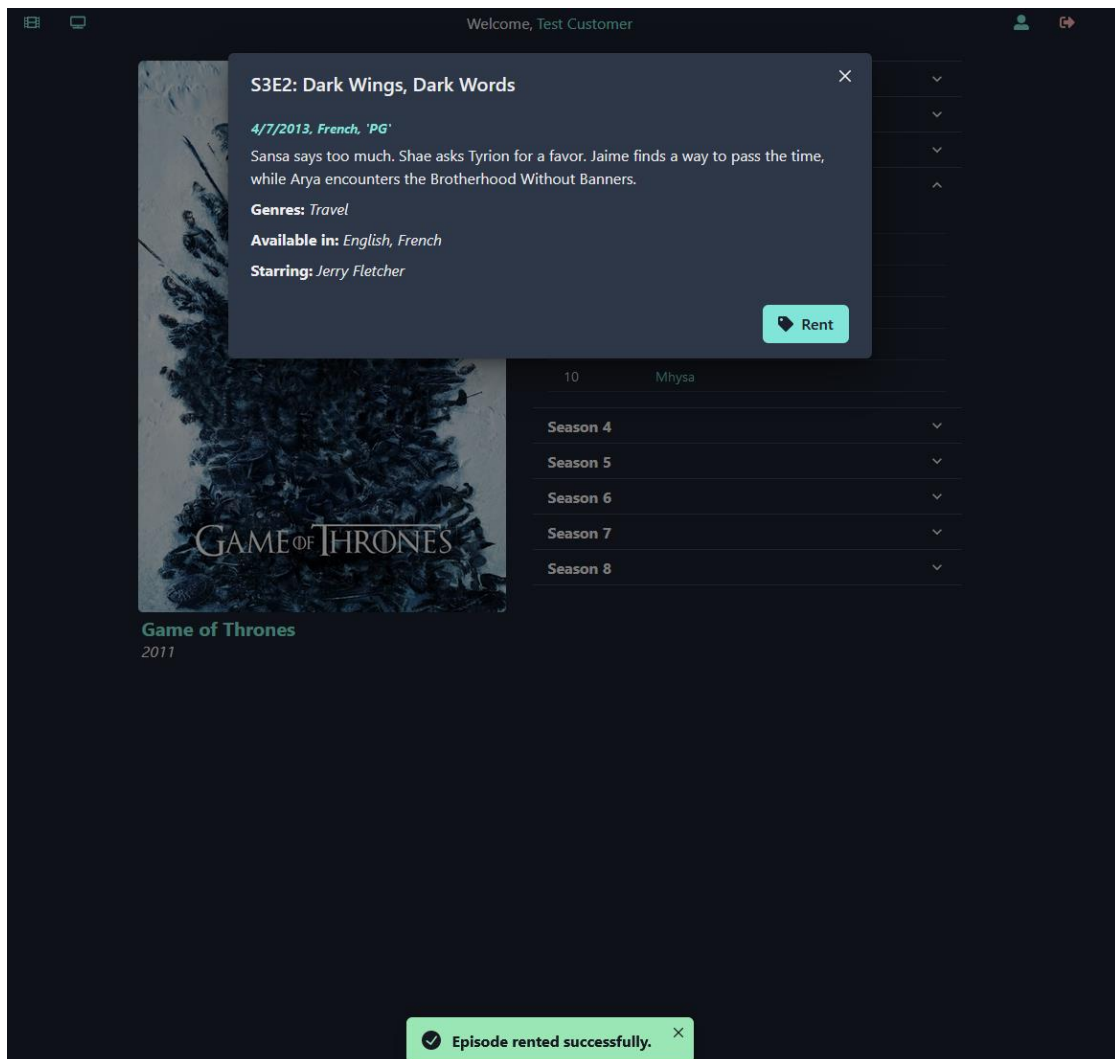
Πατώντας σε μία από αυτές εμφανίζονται οι παρακάτω πληροφορίες.



Εδώ μπορείτε να δείτε πως δεν εμφανίζονται όλα τα επεισόδια που υπάρχουν στην βάση αλλά μόνο αυτά που έχουν εισαχθεί στον πίνακα **inventory**. Πατώντας πάνω σε ένα επεισόδιο εμφανίζονται τα παρακάτω:



Πατώντας το κουμπί **Rent** ο χρήστης μπορεί και πάλι να δημιουργήσει ενοικίαση με αντίστοιχο feedback.



- Σελίδα **profile**

Στη σελίδα αυτή εμφανίζονται τα στοιχεία του χρήστη καθώς και ο πίνακας με τις προηγούμενες ενοικιάσεις του.

Welcome, Test Customer

First Name

Test

Country

Indonesia

Last Name

Customer

City

Brussel

Email

customer@mail.com

Address

P.O. Box 646, 9978 Egestas. Av., Baden W?rttemberg, D878 9TW, 1-881-868-6813

Subscription Type

Only Films

Only Series

Films & Series

ID: 1 Creation Date: 11/17/2022

Update

Order History

ORDER ID	ORDER TYPE	TITLE	ORDER DATE	PRICE
4979	EPISODE	Game of Thrones: S3 E2	11/21/2022	€ 0.15
4978	FILM	The Lord of the Rings: The Return of the King	11/21/2022	€ 0.30
4977	EPISODE	Breaking Bad: S2 E3	11/21/2022	€ 0.08
4976	FILM	Demon Slayer -Kimetsu no Yaiba- The Movie: Mugen Train	11/21/2022	€ 0.30
4975	EPISODE	House of the Dragon: S1 E7	11/21/2022	€ 0.15
4974	EPISODE	Arcane: S1 E7	11/20/2022	€ 0.10
4973	FILM	The Lord of the Rings: The Return of the King	11/20/2022	€ 0.15
4972	EPISODE	Fullmetal Alchemist: Brotherhood: S1 E63	11/20/2022	€ 0.10
4971	FILM	Your Name.	11/20/2022	€ 0.30
4970	FILM	Black Panther: Wakanda Forever	11/20/2022	€ 0.15
4969	EPISODE	Breaking Bad: S5 E16	11/20/2022	€ 0.10
4968	EPISODE	Arcane: S1 E1	11/20/2022	€ 0.10
4965	FILM	Pulp Fiction	11/17/2022	€ 0.30
4967	FILM	The Lord of the Rings: The Two Towers	11/17/2022	€ 0.15

Εδώ ο χρήστης μπορεί να αλλάξει τα στοιχεία του (εκτός από το e-mail) και οι αλλαγές γίνονται άμεσα στη βάση. Επίσης με αλλαγή στον τύπο εγγραφής απαγορεύεται η είσοδος στη σελίδα με τον άλλο τύπο εγγραφής στο χρήστη. Με την αλλαγή δεδομένων εμφανίζεται και αντίστοιχο feedback στο χρήστη.

Welcome, Updated Name

First Name

Updated

Country

Indonesia

Last Name

Name

City

Brussel

Email

customer@mail.com

Address

P.O. Box 646, 9978 Egestas. Av., Baden W?rttemberg, D878 9TW, 1-881-868-6813

Subscription Type

Only Films

Only Series

Films & Series

ID: 1 Creation Date: 11/17/2022

Update

Order History

ORDER ID	ORDER TYPE	TITLE	ORDER DATE	PRICE
4979	EPISODE	Game of Thrones: S3 E2	11/21/2022	€ 0.15
4978	FILM	The Lord of the Rings: The Return of the King	11/21/2022	€ 0.30
4977	EPISODE	Breaking Bad: S2 E3	11/21/2022	€ 0.08
4976	FILM	Demon Slayer -Kimetsu no Yaiba- The Movie: Mugen Train	11/21/2022	€ 0.30
4975	EPISODE	House of the Dragon: S1 E7	11/21/2022	€ 0.15
4974	EPISODE	Arcane: S1 E7	11/20/2022	€ 0.10
4973	FILM	The Lord of the Rings: The Return of the King	11/20/2022	€ 0.15
4972	EPISODE	Fullmetal Alchemist: Brotherhood: S1 E63	11/20/2022	€ 0.10
4971	FILM	Your Name.	11/20/2022	€ 0.30
4970	FILM	Black Panther: Wakanda Forever	11/20/2022	€ 0.15
4969	EPISODE	Breaking Bad: S5 E16	11/20/2022	€ 0.10
4968	EPISODE	Arcane: S1 E1	11/20/2022	€ 0.10
4965	FILM	Pulp Fiction	11/17/2022	€ 0.30
4967	FILM	The Lord of the Rings: The Two Towers	11/17/2022	€ 0.15
4966	FILM	The Lord of the Rings: The Fellowship of the Ring	11/17/2022	€ 0.30
4963	FILM	The Lord of the Rings: The Fellowship of the Ring	11/17/2022	€ 0.30
4962	FILM	Pulp Fiction	11/17/2022	€ 0.30
4964	FILM	The Lord of the Rings: The Two Towers	11/17/2022	€ 0.15
4961	FILM	The Lord of the Rings: The Two Towers	11/17/2022	€ 0.15

Profile updated.

Βλέπουμε στην εικόνα πως το όνομα χρήστη άλλαξε πάνω στο κέντρο, καθώς και δεν υπάρχει ποια κουμπί να μεταβεί στις σειρές αφού αλλάξαμε και τον τύπο εγγραφής. Αν προσπαθήσει να μεταβεί εκεί μέσω URL θα γίνει αυτόματα redirected στη σελίδα των ταινιών. Οι αλλαγές αυτές προφανώς εμφανίζονται και στην βάση.

	id	first_name	last_name	email	created_at	is_active	sub_type	address_id
1	1	Updated	Name	customer@mail.com	2022-11-17 20:13:15	1	ONLY_FILMS	21

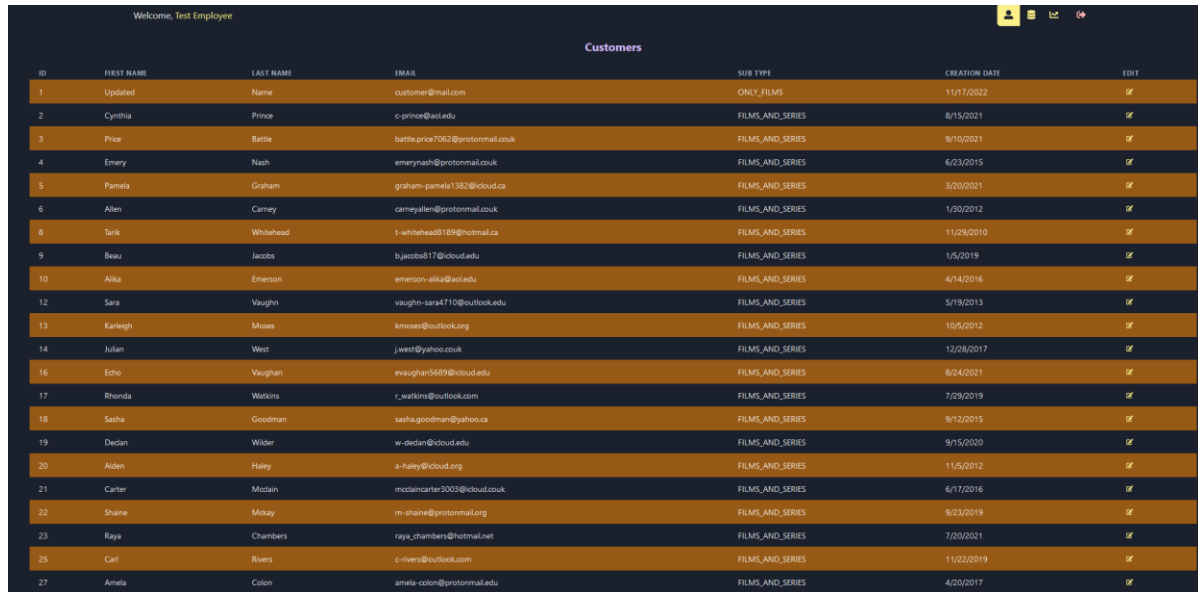
Στην εικόνα μπορούμε να δούμε επίσης και τις ενοικιάσεις που κάναμε παραπάνω.

– Υπάλληλος (employee)

Στο πάνω μέρος εμφανίζεται navigation bar, χαιρετισμός στον υπάλληλο καθώς και κουμπί για logout.

- Σελίδα **customers**

Εμφανίζεται λίστα με τους πελάτες και ο υπάλληλος μπορεί να πατήσει πάνω στο κουμπί **edit** στα δεξιά και να αλλάξει τα δεδομένα του εκάστοτε χρήστη.



ID	FIRST NAME	LAST NAME	EMAIL	SUB TYPE	CREATION DATE	EDIT
1	Updated	Name	customer@mail.com	ONLY FILMS	11/17/2022	
2	Cynthia	Prince	c.prince@ed.edu	FILMS_AND_SERIES	8/15/2021	
3	Price	Battle	battle.price7062@protonmail.co.uk	FILMS_AND_SERIES	9/10/2021	
4	Emery	Nash	emerynash@protonmail.co.uk	FILMS_AND_SERIES	6/23/2015	
5	Pamela	Graham	graham-pamela1382@icloud.ca	FILMS_AND_SERIES	3/20/2021	
6	Allen	Carney	carneyallen@protonmail.co.uk	FILMS_AND_SERIES	1/30/2012	
8	Tarik	Whitehead	t.whitehead183@hotmail.ca	FILMS_AND_SERIES	11/29/2010	
9	Beau	Jacobs	b.jacobsd17@icloud.edu	FILMS_AND_SERIES	1/5/2019	
10	Arika	Emerson	emerson-arika@ed.edu	FILMS_AND_SERIES	4/14/2016	
12	Sara	Vaughn	vaughn-sara4710@outlook.edu	FILMS_AND_SERIES	5/19/2013	
13	Karligh	Moses	kmoses@outlook.org	FILMS_AND_SERIES	10/5/2012	
14	Julian	West	j.west@yahoo.co.uk	FILMS_AND_SERIES	12/28/2017	
16	Echo	Vaughan	evauhan1669@icloud.edu	FILMS_AND_SERIES	8/04/2021	
17	Rhonda	Watkins	r.watkins@outlook.com	FILMS_AND_SERIES	7/29/2019	
18	Sasha	Goodman	sasha.goodman@yahoo.ca	FILMS_AND_SERIES	9/12/2015	
19	Declan	Wilder	w-declan@icloud.edu	FILMS_AND_SERIES	9/15/2020	
20	Aiden	Haley	a-haley@icloud.org	FILMS_AND_SERIES	11/5/2012	
21	Carter	Mccain	mccaincarter3003@icloud.co.uk	FILMS_AND_SERIES	6/17/2016	
22	Shane	Miday	m-shane@protonmail.org	FILMS_AND_SERIES	9/23/2019	
23	Raya	Chambers	raya_chambers@hotmail.net	FILMS_AND_SERIES	7/20/2021	
25	Carl	Rivers	c.rivers@outlook.com	FILMS_AND_SERIES	11/22/2019	
27	Amela	Cotton	amela.cotton@protonmail.edu	FILMS_AND_SERIES	4/20/2017	

Όταν ο υπάλληλος επεξεργάζεται έναν χρήστη μπορεί να δει και να αλλάξει όλες τις πληροφορίες (εκτός του email) καθώς και τη λίστα με τις ενοικιάσεις του χρήστη.



Welcome, Test Employee

First Name

Updated

Last Name

Name

Email

customer@mail.com

Country

Indonesia

City

Brussel

Address

P.O. Box 646, 9978 Egestas. Av., Baden W?rttemberg, D878 9TW, 1-881-868-6813

Subscription Type

☒ Only Films ☐ Only Series ☐ Films & Series

ID: 1

Creation Date: 11/17/2022

Update

Order History

ORDER ID	ORDER TYPE	TITLE	ORDER DATE	PRICE
4979	EPISODE	Game of Thrones: S3 E2	11/21/2022	€ 0.15
4978	FILM	The Lord of the Rings: The Return of the King	11/21/2022	€ 0.30
4977	EPISODE	Breaking Bad: S2 E3	11/21/2022	€ 0.08
4976	FILM	Demon Slayer - Kimetsu no Yaiba- The Movie: Mugen Train	11/21/2022	€ 0.30
4975	EPISODE	House of the Dragon: S1 E7	11/21/2022	€ 0.15
4974	EPISODE	Arcane: S1 E7	11/20/2022	€ 0.10
4973	FILM	The Lord of the Rings: The Return of the King	11/20/2022	€ 0.15
4972	EPISODE	Fullmetal Alchemist: Brotherhood: S1 E63	11/20/2022	€ 0.10
4971	FILM	The Matrix	11/20/2022	€ 0.30

• Σελίδα **data**

Στη σελίδα αυτή εμφανίζεται νέο navigation bar με το οποίο ο υπάλληλος μπορεί να δει λίστες με τα δεδομένα που είναι αποθηκευμένα μέσα στην βάση. Η δημιουργία και η επεξεργασία δεδομένων δεν έχει υλοποιηθεί.

Welcome, Test Employee

Films

Series

Seasons

Episodes

Categories

Languages

Actors

Addresses

Cities

Countries

+

ID	TITLE	SEASON	SERIES
22	Welcome to the Playground	1	Arcane
23	Some Mysteries Are Better Left Unsolved	1	Arcane
24	The Base Violence Necessary for Change	1	Arcane
25	Happy Progress Day!	1	Arcane
26	Everybody Wants to Be My Enemy	1	Arcane
27	When These Walls Come Tumbling Down	1	Arcane
28	The Boy Savior	1	Arcane
29	Oil and Water	1	Arcane
30	The Monster You Created	1	Arcane
31	DOG & CHAINSAW	1	Chainsaw Man
32	ARRIVAL IN TOKYO	1	Chainsaw Man
33	MEOWY'S WHEREABOUTS	1	Chainsaw Man
34	RESCUE	1	Chainsaw Man
35	GUN DEVIL	1	Chainsaw Man
36	KILL DENJI	1	Chainsaw Man
37	Episode 7	1	Chainsaw Man

• Σελίδα **sales**

Εμφανίζονται δύο λίστες με τις ταινίες και σειρές με τις περισσότερες ενοικιάσεις τον τελευταίο μήνα αντίστοιχα.

Welcome, Test Employee

ID	TITLE	RENTALS
12	The Lord of the Rings: The Two Towers	4
10	Pulp Fiction	3
11	The Lord of the Rings: The Fellowship of the Ring	3
1	Black Panther: Wakanda Forever	2
20	Avengers: Endgame	2

Film Rentals

ID	TITLE	RENTALS
8	Game of Thrones	15
9	Fullmetal Alchemist: Brotherhood	8
6	Breaking Bad	5
7	The Lord of the Rings: The Rings of Power	4
2	Arcane	3

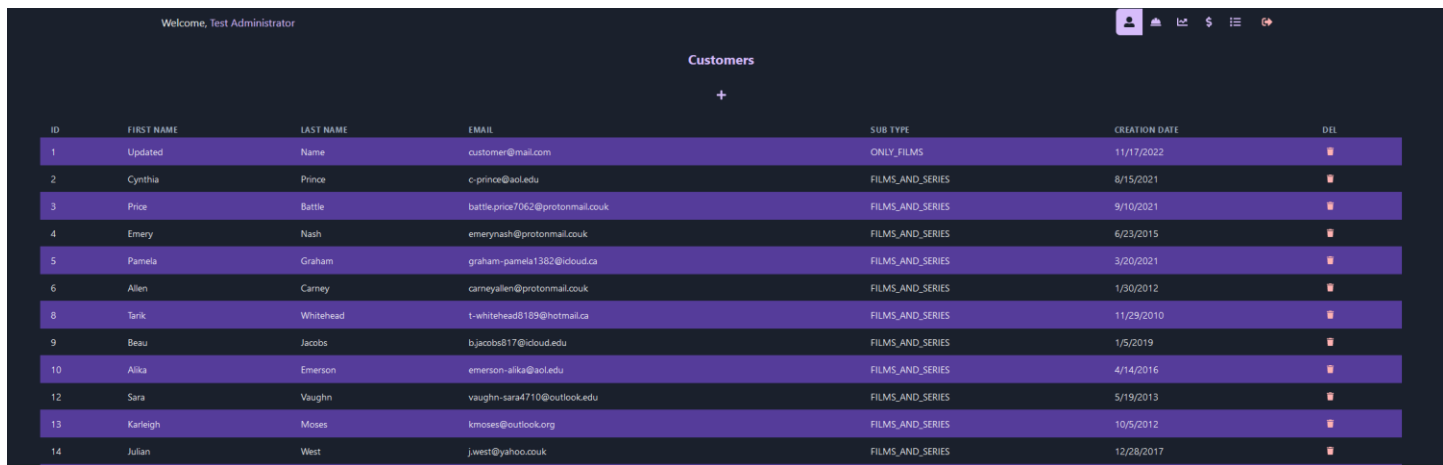
Series Rentals

– Διαχειριστής (administrator)

Στο πάνω μέρος εμφανίζεται navigation bar, χαιρετισμός στον υπάλληλο καθώς και κουμπί για logout.

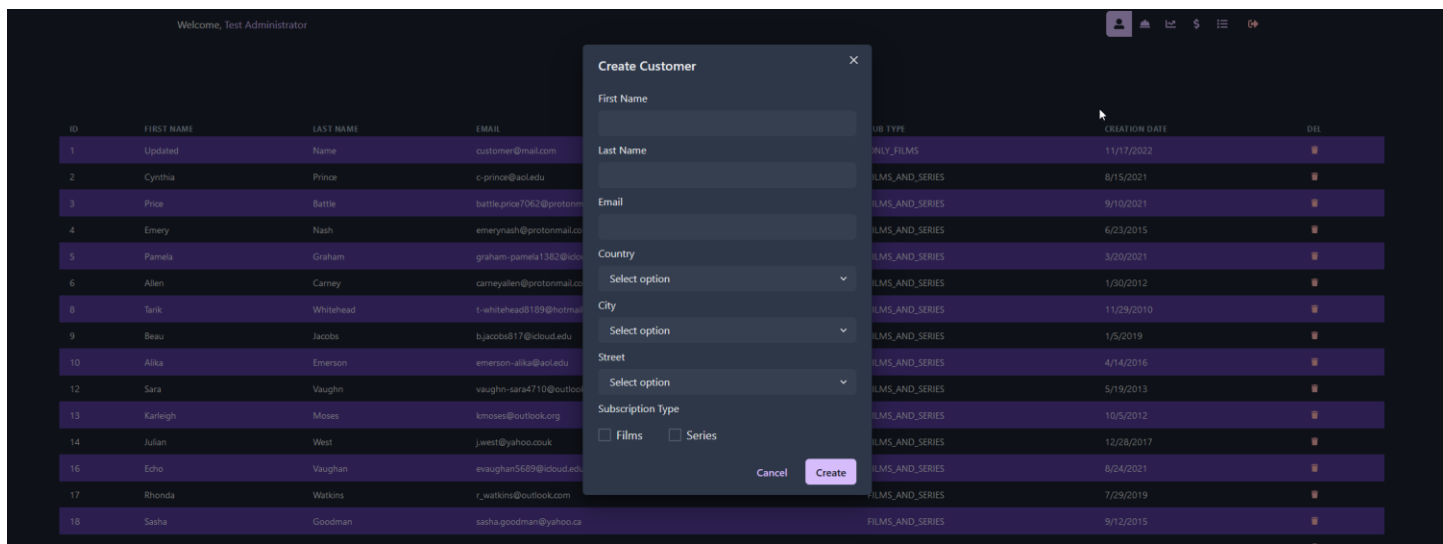
• Σελίδα customers

Εμφανίζεται λίστα με όλους τους πελάτες της εφαρμογής.



ID	FIRST NAME	LAST NAME	EMAIL	SUB TYPE	CREATION DATE	DEL
1	Updated	Name	customer@mail.com	ONLY FILMS	11/17/2022	
2	Cynthia	Prince	c-prince@aol.edu	FILMS_AND_SERIES	8/15/2021	
3	Price	Battle	battle.price7062@protonmail.co.uk	FILMS_AND_SERIES	9/10/2021	
4	Emery	Nash	emerynash@protonmail.co.uk	FILMS_AND_SERIES	6/23/2015	
5	Pamela	Graham	graham-pamela1382@icloud.ca	FILMS_AND_SERIES	3/20/2021	
6	Allen	Carney	carneyallen@protonmail.co.uk	FILMS_AND_SERIES	1/30/2012	
8	Tarik	Whitehead	t-whitehead8189@hotmail.ca	FILMS_AND_SERIES	11/29/2010	
9	Beau	Jacobs	b.jacobs817@icloud.edu	FILMS_AND_SERIES	1/5/2019	
10	Alika	Emerson	emerson-aliika@aol.edu	FILMS_AND_SERIES	4/14/2016	
12	Sara	Vaughn	vaughn-sara4710@outlook.edu	FILMS_AND_SERIES	5/19/2013	
13	Karleigh	Moses	kmoses@outlook.org	FILMS_AND_SERIES	10/5/2012	
14	Julian	West	j.west@yahoo.co.uk	FILMS_AND_SERIES	12/28/2017	

Ο διαχειριστής έχει τη δυνατότητα να δημιουργήσει χρήστη με τη χρήση του κουμπιού + και με εισαγωγή των αντίστοιχων δεδομένων.



Create Customer

First Name

Last Name

Email

Country

Select option

City

Select option

Street

Select option

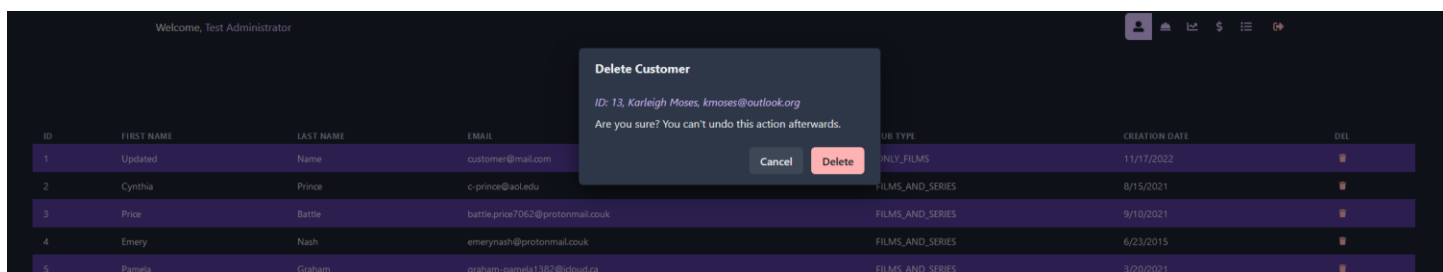
Subscription Type

☐ Films ☐ Series

Cancel Create

Με τη δημιουργία νέου πελάτη εμφανίζεται feedback στο διαχειριστή και ενημερώνεται ο πίνακας

Ο διαχειριστής έχει τη δυνατότητα να διαγράψει κάποιον χρήστη με το αντίστοιχο κουμπί οπότε και εμφανίζεται αυτό το dialog για έλεγχο.



Delete Customer

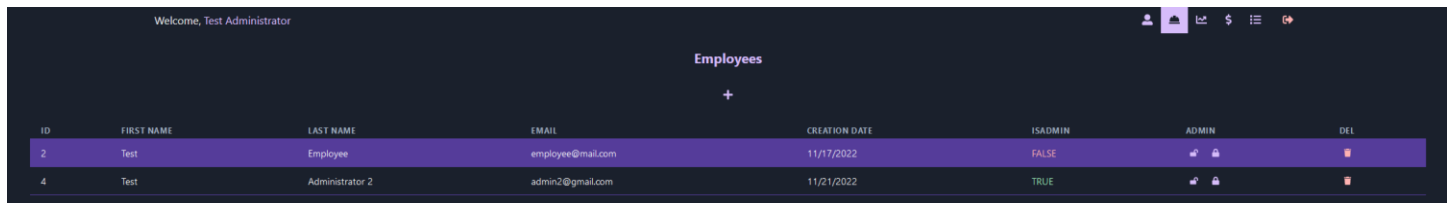
ID: 13, Karleigh Moses, kmoses@outlook.org

Are you sure? You can't undo this action afterwards.

Cancel Delete

- Σελίδα **employees**

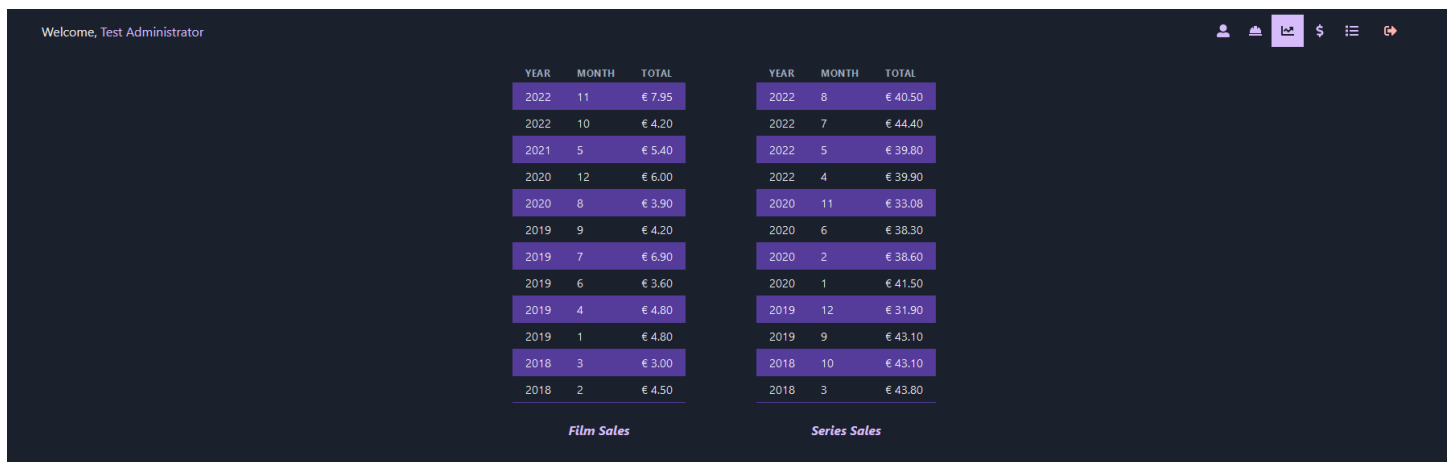
Η λειτουργία είναι ακριβώς ίδια με αυτή της λίστας των customers με τη μόνη διαφορά πως υπάρχει η επιλογή να αλλάξει έναν υπάλληλο από υπάλληλο σε διαχειριστή και το αντίθετο.



ID	FIRST NAME	LAST NAME	EMAIL	CREATION DATE	ISADMIN	ADMIN	DEL
2	Test	Employee	employee@mail.com	11/17/2022	FALSE		
4	Test	Administrator 2	admin2@gmail.com	11/21/2022	TRUE		

- Σελίδα **sales**

Εμφανίζονται οι πωλήσεις ανά μήνα από την πρώτη πώληση για ταινίες και σειρές αντίστοιχα.



YEAR	MONTH	TOTAL
2022	11	€ 7.95
2022	10	€ 4.20
2021	5	€ 5.40
2020	12	€ 6.00
2020	8	€ 3.90
2019	9	€ 4.20
2019	7	€ 6.90
2019	6	€ 3.60
2019	4	€ 4.80
2019	1	€ 4.80
2018	3	€ 3.00
2018	2	€ 4.50

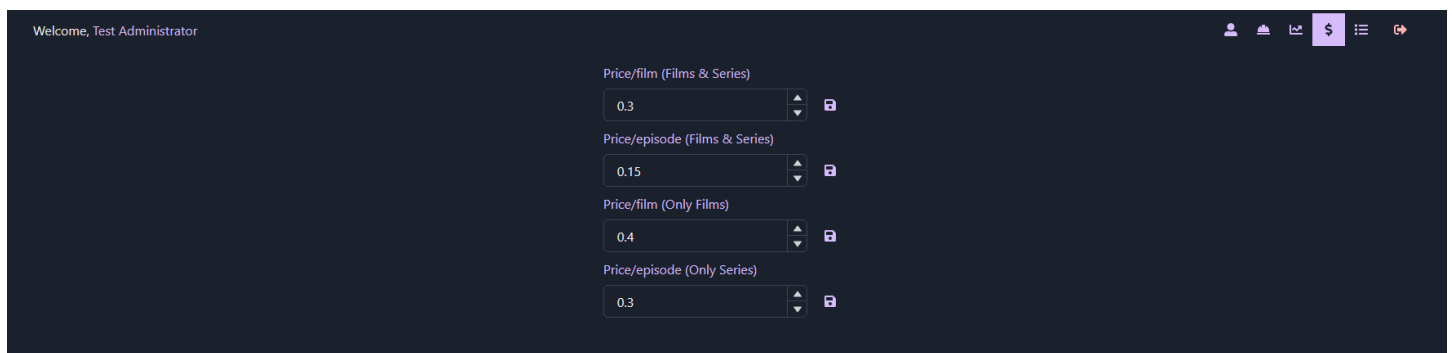
Film Sales

YEAR	MONTH	TOTAL
2022	8	€ 40.50
2022	7	€ 44.40
2022	5	€ 39.80
2022	4	€ 39.90
2020	11	€ 33.08
2020	6	€ 38.30
2020	2	€ 38.60
2020	1	€ 41.50
2019	12	€ 31.90
2019	9	€ 43.10
2018	10	€ 43.10
2018	3	€ 43.80

Series Sales

- Σελίδα **pricing**

Εμφανίζονται 4 number input τα οποία μπορεί να χρησιμοποιήσει ο διαχειριστής για να αλλάξει τις τιμές για τις ενοικιάσεις.



Price/film (Films & Series)

0.3

Price/episode (Films & Series)

0.15

Price/film (Only Films)

0.4

Price/episode (Only Series)

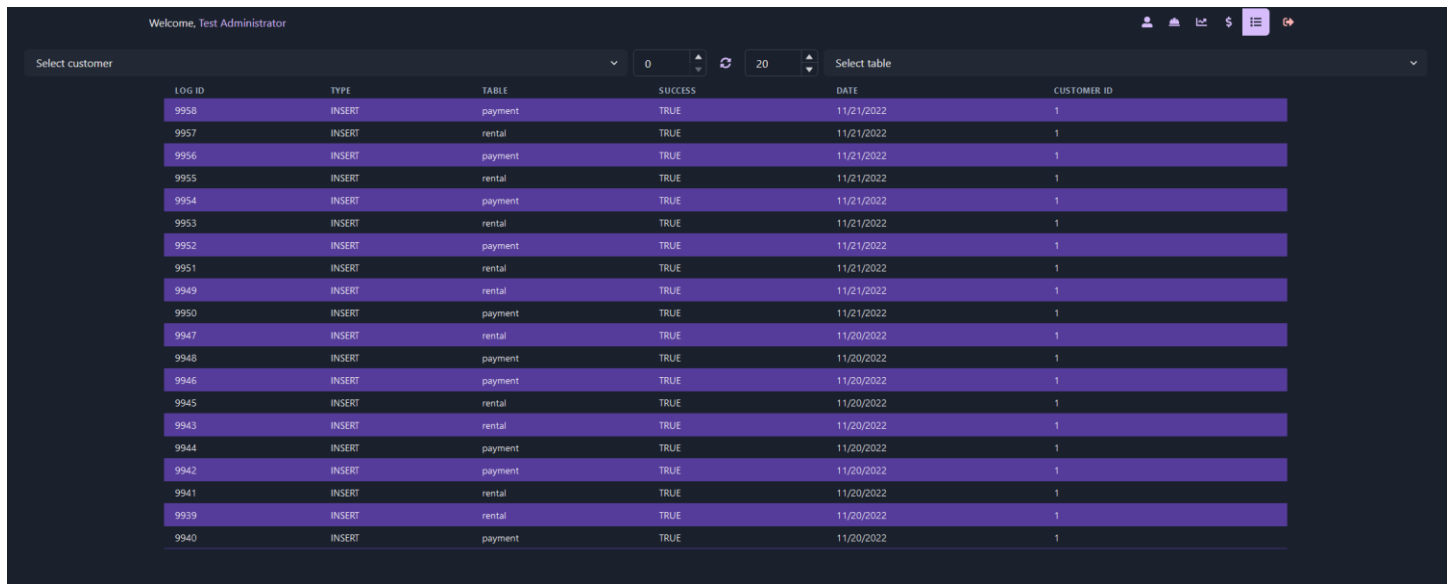
0.3

Οι αλλαγές εμφανίζονται και άμεσα στη βάση.

	id	sub_type	show_type	amount	created_at
1	9	FILMS_AND_SERIES	EPISODE	0.25	2022-11-21 22:00:03

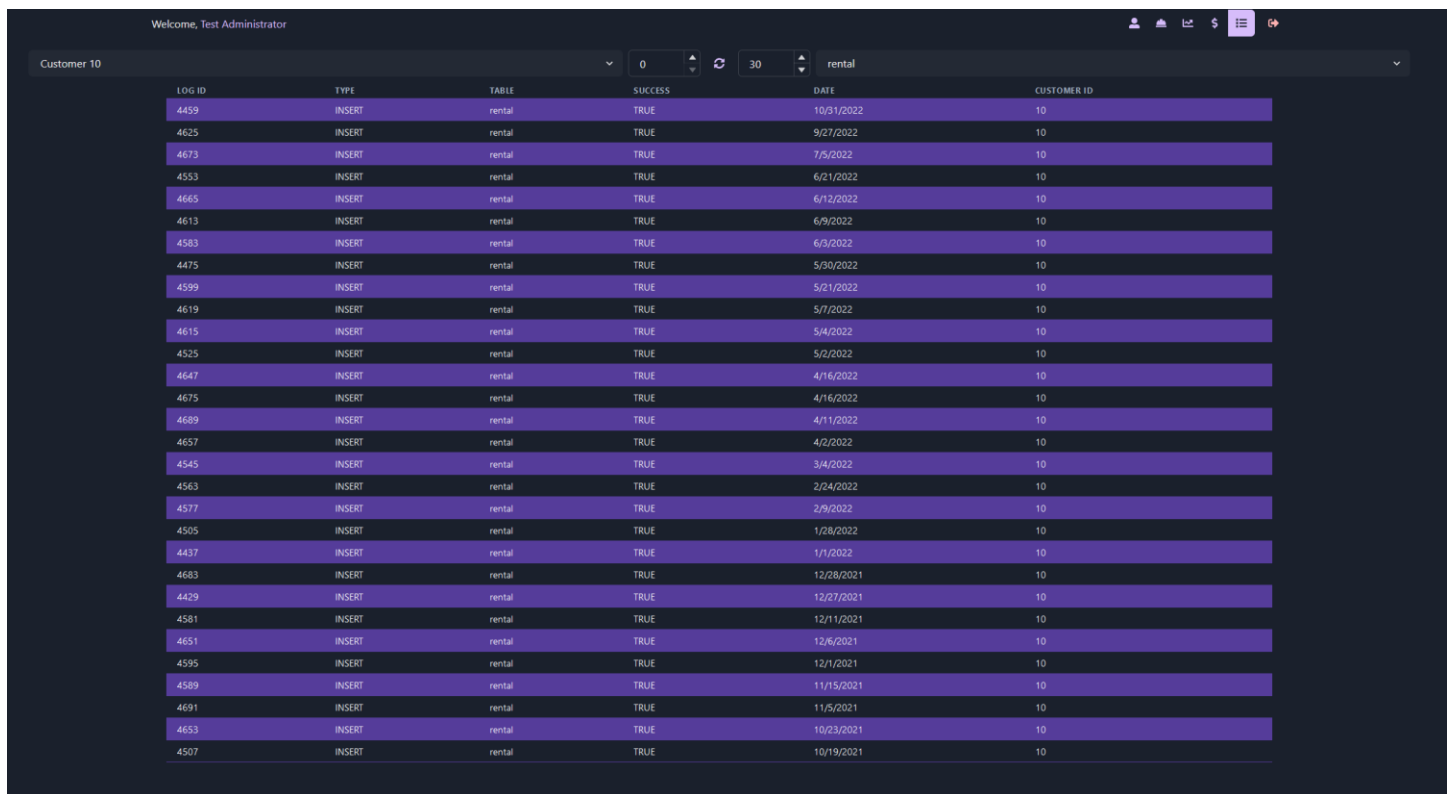
- Σελίδα logs

Εμφανίζεται πίνακας με τις ενέργειες που υπάρχουν στον πίνακα **customer_log**. Ο διαχειριστής έχει τη δυνατότητα να φιλτράρει ανά χρήστη και ανά πίνακα της βάσης, καθώς και να ελέγξει τον αριθμό των αποτελεσμάτων που εμφανίζεται καθώς ο αριθμός των γραμμών στον πίνακα είναι πολύ μεγάλος.



LOG ID	TYPE	TABLE	SUCCESS	DATE	CUSTOMER ID
9958	INSERT	payment	TRUE	11/21/2022	1
9957	INSERT	rental	TRUE	11/21/2022	1
9956	INSERT	payment	TRUE	11/21/2022	1
9955	INSERT	rental	TRUE	11/21/2022	1
9954	INSERT	payment	TRUE	11/21/2022	1
9953	INSERT	rental	TRUE	11/21/2022	1
9952	INSERT	payment	TRUE	11/21/2022	1
9951	INSERT	rental	TRUE	11/21/2022	1
9949	INSERT	rental	TRUE	11/21/2022	1
9950	INSERT	payment	TRUE	11/21/2022	1
9947	INSERT	rental	TRUE	11/20/2022	1
9948	INSERT	payment	TRUE	11/20/2022	1
9946	INSERT	payment	TRUE	11/20/2022	1
9945	INSERT	rental	TRUE	11/20/2022	1
9943	INSERT	rental	TRUE	11/20/2022	1
9944	INSERT	payment	TRUE	11/20/2022	1
9942	INSERT	payment	TRUE	11/20/2022	1
9941	INSERT	rental	TRUE	11/20/2022	1
9939	INSERT	rental	TRUE	11/20/2022	1
9940	INSERT	payment	TRUE	11/20/2022	1

Για παράδειγμα αυτές είναι οι τελευταίες 30 εγγραφές του χρήστη 10 στο πίνακα rental.



LOG ID	TYPE	TABLE	SUCCESS	DATE	CUSTOMER ID
4459	INSERT	rental	TRUE	10/31/2022	10
4625	INSERT	rental	TRUE	9/27/2022	10
4673	INSERT	rental	TRUE	7/5/2022	10
4553	INSERT	rental	TRUE	6/21/2022	10
4665	INSERT	rental	TRUE	6/12/2022	10
4613	INSERT	rental	TRUE	6/9/2022	10
4583	INSERT	rental	TRUE	6/3/2022	10
4475	INSERT	rental	TRUE	5/30/2022	10
4599	INSERT	rental	TRUE	5/21/2022	10
4619	INSERT	rental	TRUE	5/7/2022	10
4615	INSERT	rental	TRUE	5/4/2022	10
4525	INSERT	rental	TRUE	5/2/2022	10
4647	INSERT	rental	TRUE	4/16/2022	10
4675	INSERT	rental	TRUE	4/16/2022	10
4689	INSERT	rental	TRUE	4/11/2022	10
4657	INSERT	rental	TRUE	4/2/2022	10
4545	INSERT	rental	TRUE	3/4/2022	10
4563	INSERT	rental	TRUE	2/24/2022	10
4577	INSERT	rental	TRUE	2/9/2022	10
4505	INSERT	rental	TRUE	1/28/2022	10
4437	INSERT	rental	TRUE	1/1/2022	10
4683	INSERT	rental	TRUE	12/28/2021	10
4429	INSERT	rental	TRUE	12/27/2021	10
4581	INSERT	rental	TRUE	12/11/2021	10
4651	INSERT	rental	TRUE	12/6/2021	10
4595	INSERT	rental	TRUE	12/1/2021	10
4589	INSERT	rental	TRUE	11/15/2021	10
4691	INSERT	rental	TRUE	11/5/2021	10
4653	INSERT	rental	TRUE	10/23/2021	10
4507	INSERT	rental	TRUE	10/19/2021	10

Εδώ ολοκληρώνεται η αναφορά για το Project του Εργαστηρίου Βάσεων Δεδομένων. Σας ευχαριστούμε πολύ για την προσοχή σας.