

ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ

**ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ
ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ**

ΒΑΣΕΙΣ ΔΕΔΟΜΕΝΩΝ

ΑΝΑΦΟΡΑ ΕΞΑΜΗΝΙΑΙΑΣ ΕΡΓΑΣΙΑΣ

Εαρινό εξάμηνο 2024

ΟΜΑΔΑ 150

Ονοματεπώνυμο: Παύλος Σελίμης

Αριθμός Μητρώου: 03120171

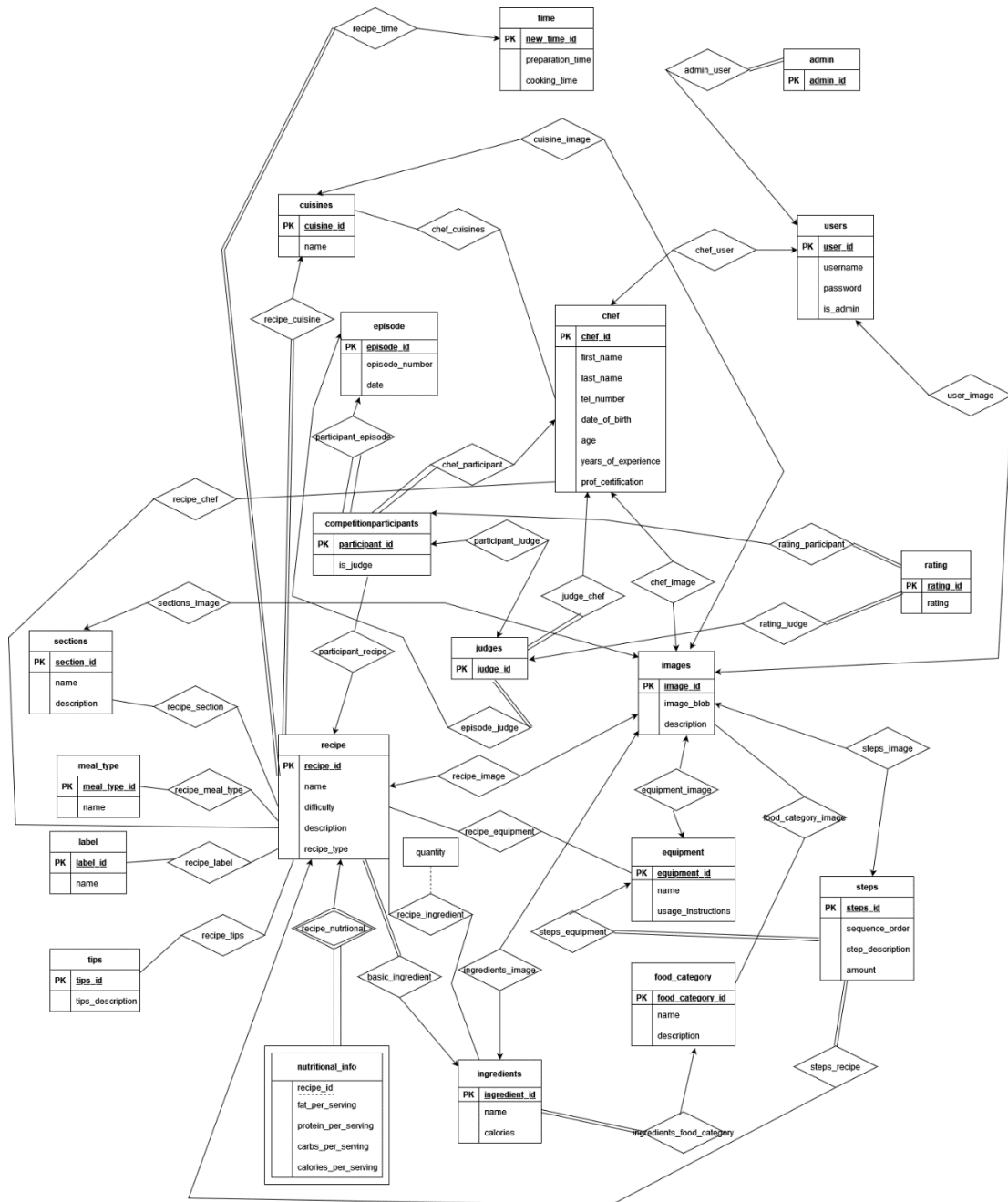
**Ονοματεπώνυμο: Ηλίας Αριστείδης
Βουγιούκας**

Αριθμός Μητρώου: 03120232

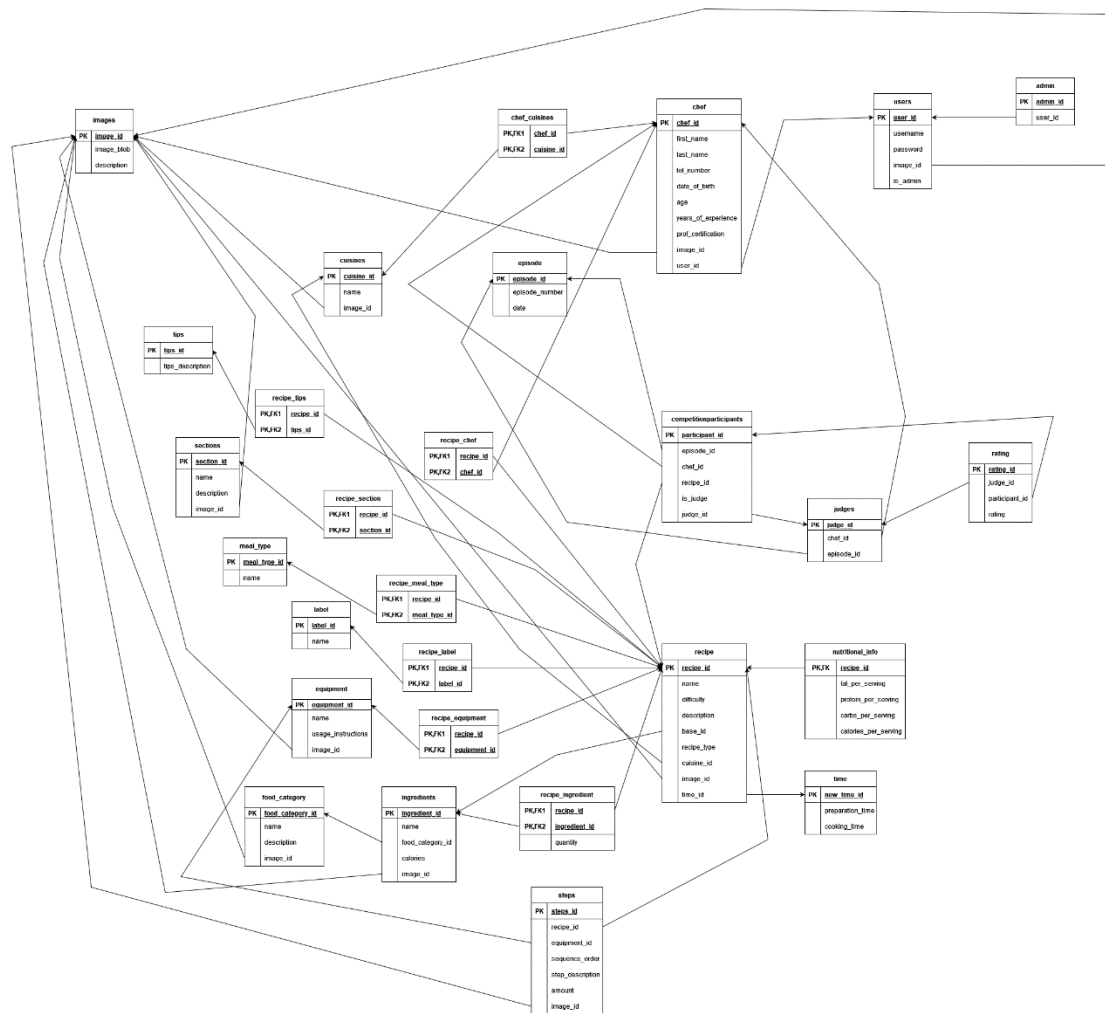
Ονοματεπώνυμο: Μάρκου Αθανάσιος

Αριθμός Μητρώου: 03123628

Διάγραμμα ER



Σχεσιακό Διάγραμμα



Σχολιασμός Σχεσιακού Διαγράμματος

Στην βάση μας περιέχονται 3 κύριες οντότητες : table chef , table recipe και table episode. Οι οντότητες που σχετίζονται με το chef table είναι τα tables cuisines μέσω του table chef_cuisines και users. Με το recipe table, σχετίζονται οι οντότητες cuisines, equipment, food_category, ingredients, label, meal_type, nutritional_info, sections, steps, time, tips μέσω και των αντίστοιχων ενδιάμεσων tables : recipe_equipment, recipe_ingredient, recipe_label, recipe_meal_type, recipe_section και recipe_tips. Με το episode table σχετίζονται οι οντότητες recipe_chef, judges, rating και competitionparticipants, οι οποίες περιέχουν πληροφορίες όσον αφορά την συμμετοχή των chef στα επεισόδια. Στις 4 αυτές οντότητες εισάγονται και τα δεδομένα για κάθε επεισόδιο μέσω της κλήρωσης (episodes_data_generator.py). Τέλος , έχουμε την οντότητα images , που περιέχει οποιαδήποτε εικόνα μπορεί να μας χρειαστεί για κάποια άλλη οντότητα , και την οντότητα admin , που σχετίζεται με το users table και έχει αποθηκευμένους τους administrators (εμείς στην βάση μας έχουμε έναν μόνο admin).

Indexing Βάσης Δεδομένων

Για να ορίσουμε τα indexes της βάσης αρχικά λαμβάνουμε υπόψιν ότι αυτά δημιουργούνται αυτόματα για τις κολώνες που αποτελούν primary keys για κάθε table. Αυτό είναι πολύ σημαντικό εφόσον τα primary keys γενικά αποτελούν IDs που χρησιμοποιούνται συνεχώς στα

διάφορα queries (WHERE clauses, JOIN ON κλπ).Επομένως είναι απαραίτητο η πρόσβαση σε κάθε primary key να είναι γρήγορη, κάτι που επιτυγχάνεται με τη χρήση ευρετηρίων.Το ίδιο ισχύει και για τα foreign keys, όπου επίσης δημιουργήθηκαν ευρετήρια αυτόματα για αυτά και χρησιμοποιούνται συνεχώς σε queries. Επιπλέον αναφέρεται πως όπου επιβάλλεται στο σχήμα περιορισμός UNIQUE δημιουργείται ευρετήριο για τα πεδία αυτά, οπότε δεν απαιτείται να δημιουργηθεί νέο αν αυτό απαιτηθεί για λόγους αναζήτησης.Έτσι, για την βελτιστοποίηση των queries δημιουργήσαμε ακόμα τα indexes :

date (date στο table episode) το οποίο χρησιμοποιείται σε αρκετά WHERE clauses

age (age στο table chef) το οποίο επίσης χρησιμοποιείται σε WHERE clause

idx_recipe_label_recipe_id_label_id (recipe_id, label_id στο table recipe_label) το οποίο είναι composite index και δημιουργήθηκε για το query 6

```
JOIN
  recipe_label r12 ON r11.recipe_id = r12.recipe_id AND r11.label_id
< r12.label_id
```

Όλα τα indexes φαίνονται αναλυτικά στο DDL script.

DDL & DML Scripts

Τα DDL και DML scripts δεν εμπεριέχονται στην αναφορά, καθώς περιέχουν πολλές γραμμές κειμένου, αλλά μπορούν να βρεθούν εύκολα μέσω του συνδέσμου του git repo της εφαρμογής μας :

https://github.com/paulosselimis/database_project.git

στα αρχεία DDL.sql και DML.sql αντίστοιχα.

Queries

Μέσω του παραπάνω συνδέσμου επίσης, στο αρχείο queries.py υπάρχουν 15 συναρτήσεις query1 – query15 , όπου φαίνονται ξεκάθαρα οι υλοποιήσεις των ζητούμενων queries.Επίσης, στην εκφώνηση μας ζητείται για το query 6 και query 8 να υπάρχει και εναλλακτικό Query Plan (πχ με force index), τα αντίστοιχα traces και τα συμπεράσματά μας από την μελέτη αυτών.

Για το query 6 :

Εναλλακτικό query plan (με force index) :

```
SELECT
  CONCAT(LEAST(r11.label_id, r12.label_id), '-', GREATEST(r11.label_id, r12.label_id)) AS label_pair,
  COUNT(*) AS pair_count
FROM
  recipe_label r11 FORCE INDEX (idx_recipe_label_recipe_id_label_id)
JOIN
  recipe_label r12 FORCE INDEX (idx_recipe_label_recipe_id_label_id)
  ON r11.recipe_id = r12.recipe_id AND r11.label_id < r12.label_id
JOIN
  competitionparticipants cp ON r11.recipe_id = cp.recipe_id
GROUP BY
  label_pair
ORDER BY
  pair_count DESC
LIMIT 3;
```

Μέσω του EXPLAIN statement έχουμε για το παραπάνω query :

id	select_type	table	type	possible_keys	key	key_len	ref	rows	Extra
1	SIMPLE	r11	index	idx_recipe_label_recipe_id_label_id	idx_recipe_label_recipe_id_label_id	8	NULL	403	Using index; Using temporary; Using filesort
1	SIMPLE	r12	ref	idx_recipe_label_recipe_id_label_id	idx_recipe_label_recipe_id_label_id	4	cooking_contest.r11.recipe_id	2	Using where; Using index
1	SIMPLE	cp	ref	recipe_id	recipe_id	5	cooking_contest.r11.recipe_id	5	Using index

Για το αρχικό query μέσω του EXPLAIN statement έχουμε :

id	select_type	table	type	possible_keys	key	key_len	ref	rows	Extra
1	SIMPLE	r11	index	recipe_id_label_id_idx_recipe_label_recipe_id_label_id	idx_recipe_label_recipe_id_label_id	8	NULL	403	Using index; Using temporary; Using filesort
1	SIMPLE	r12	ref	recipe_id_label_id_idx_recipe_label_recipe_id_label_id	idx_recipe_label_recipe_id_label_id	4	cooking_contest.r11.recipe_id	2	Using where; Using index
1	SIMPLE	cp	ref	recipe_id	recipe_id	5	cooking_contest.r11.recipe_id	5	Using index

Συμπεραίνουμε λοιπόν , ότι και στις δυο περιπτώσεις το query εκτελείται με την ίδια αποδοτικότητα , επιλέγονται αυτόματα λοιπόν τα σωστά indexes.

Για το query 8 :

Εναλλακτικό query plan (με force index) :

```
SELECT
    e.episode_number,
    cp.episode_id,
    COUNT(re.recipe_id) AS equipment_count
FROM
    competitionparticipants cp FORCE INDEX (episode_id)
JOIN
    episode e ON cp.episode_id = e.episode_id
JOIN
    recipe_equipment re FORCE INDEX (recipe_id) ON cp.recipe_id = re.recipe_id
GROUP BY
    cp.episode_id
ORDER BY
    equipment_count DESC
LIMIT 1;
```

Μέσω του EXPLAIN statement έχουμε για το παραπάνω query :

id	select_type	table	type	possible_keys	key	key_len	ref	rows	Extra
1	SIMPLE	e	index	PRIMARY	PRIMARY	4	NULL	60	Using temporary; Using filesort
1	SIMPLE	cp	ref	episode_id	episode_id	4	cooking_contest.e.episode_id	6	Using where
1	SIMPLE	re	ref	recipe_id	recipe_id	4	cooking_contest.cp.recipe_id	2	Using index

Για το αρχικό query μέσω του EXPLAIN statement έχουμε :

id	select_type	table	type	possible_keys	key	key_len	ref	rows	Extra
1	SIMPLE	e	index	PRIMARY	PRIMARY	4	NULL	60	Using temporary; Using filesort
1	SIMPLE	cp	ref	episode_id,recipe_id	episode_id	4	cooking_contest.e.episode_id	6	Using where
1	SIMPLE	re	ref	recipe_id	recipe_id	4	cooking_contest.cp.recipe_id	2	Using index

Συμπεραίνουμε λοιπόν , ότι και στις δυο περιπτώσεις το query εκτελείται με την ίδια αποδοτικότητα , επιλέγονται αυτόματα λοιπόν τα σωστά indexes.

Οδηγίες Εγκατάστασης

Βήμα 1ο - Κατέβασμα του repository μέσω git

Για την εγκατάσταση της εφαρμογής πρέπει πρώτα να γίνει clone τοπικά το git repo της εφαρμογής. Αυτό μπορεί να γίνει είτε μέσω της

εφαρμογής GitHub desktop, είτε μέσω κάποιου terminal με την εντολή `git clone https://github.com/paulosselimis/database_project.git` στο τοπικό directory που επιθυμούμε να εγκαταστήσουμε την εφαρμογή.

Βήμα 2ο - Εγκατάσταση της βάσης

Προκειμένου να εγκαταστήσουμε την βάση στον υπολογιστή μας χρειαζόμαστε έναν sql server (συγκεκριμένα χρησιμοποιήσαμε mysql μέσω xampp). Κάντε εγκατάσταση το XAMPP, ανοίξτε το XAMPP control panel και πατήστε start Apache και start MySQL. Στην συνέχεια, ανοίξτε το phpMyAdmin (<http://localhost/phpmyadmin/>) , δημιουργήστε μια νέα βάση δεδομένων με όνομα `cooking_contest` και κάνετε import το αρχείο `cooking_contest.sql`.

Τέλος, κάντε εγκατάσταση την Python και καντε install το **MySQL Connector** μέσω της εντολής : «`pip install mysql-connector-python`».

Το αρχείο `cooking_contest.sql` , ήδη περιέχει δεδομένα από τα αποτελέσματα των κληρώσεων. Αν θέλετε να κάνετε generate νέα δεδομένα , πρέπει να τρέξετε το αρχείο `delete_episodes_data.py`

(μέσω της εντολής: `python delete_episodes_data.py`) και ύστερα να

τρέξετε το αρχείο `episodes_data_generator.py` (μέσω της εντολής: `python episodes_data_generator.py`) έτσι ώστε να δημιουργηθούν και εισαχθούν νέα δεδομένα στην βάση όσον αφορά το κάθε επεισόδιο του διαγωνισμού μαγειρικής (το συγκεκριμένο πρόγραμμα υλοποιεί 60 κληρώσεις , εισάγει δηλαδή δεδομένα για 60 επεισόδια). Η βάση δεδομένων είναι έτοιμη.

Βήμα 3ο - Launch της εφαρμογής

Τρέξτε το αρχείο `login.py` (μέσω της εντολής: `python login.py`). Για την εκτέλεση των queries το αρχείο αυτό κάνει import το αρχείο `queries.py` το οποίο περιέχει τις 15 συναρτήσεις `query1 – query15`. Τρέχοντας λοιπόν το πρόγραμμα `login.py` , είστε έτοιμοι να χρησιμοποιήσετε την εφαρμογή μέσω του command line. Θα σας ζητηθεί username και password ώστε να κάνετε login στην εφαρμογή είτε ως user είτε ως admin , και στην συνέχεια ακολουθώντας τις οδηγίες που θα σας εμφανίζονται μπορείτε να κάνετε εύκολα χρήση της εφαρμογής.

Admin

Username : admin

Password : adminpassword

Users

Έχουμε στην βάση 70 users (όσοι και οι chef). Ενδεικτικά, για τον πρώτο user έχουμε :

Username : chef1

Password : password1

Git Repo

Το repository της εργασίας βάσης δεδομένων στο github:

https://github.com/paulosselimis/database_project.git