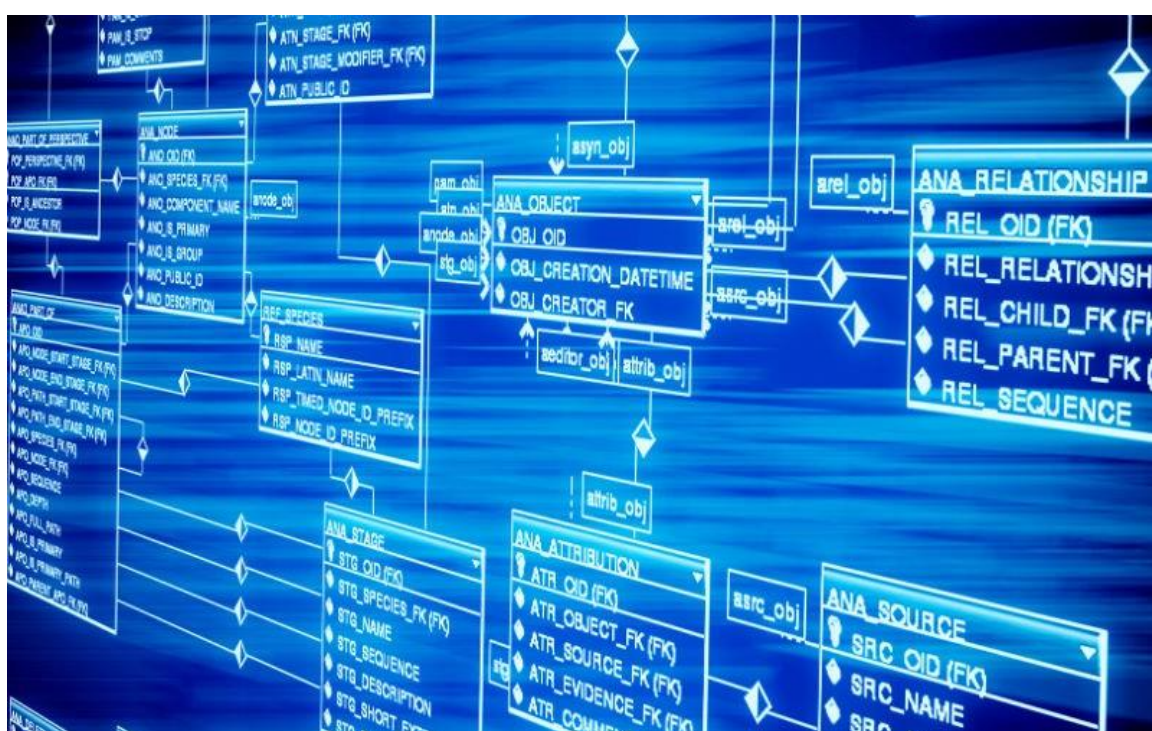




ΒΑΣΕΙΣ ΔΕΔΟΜΕΝΩΝ

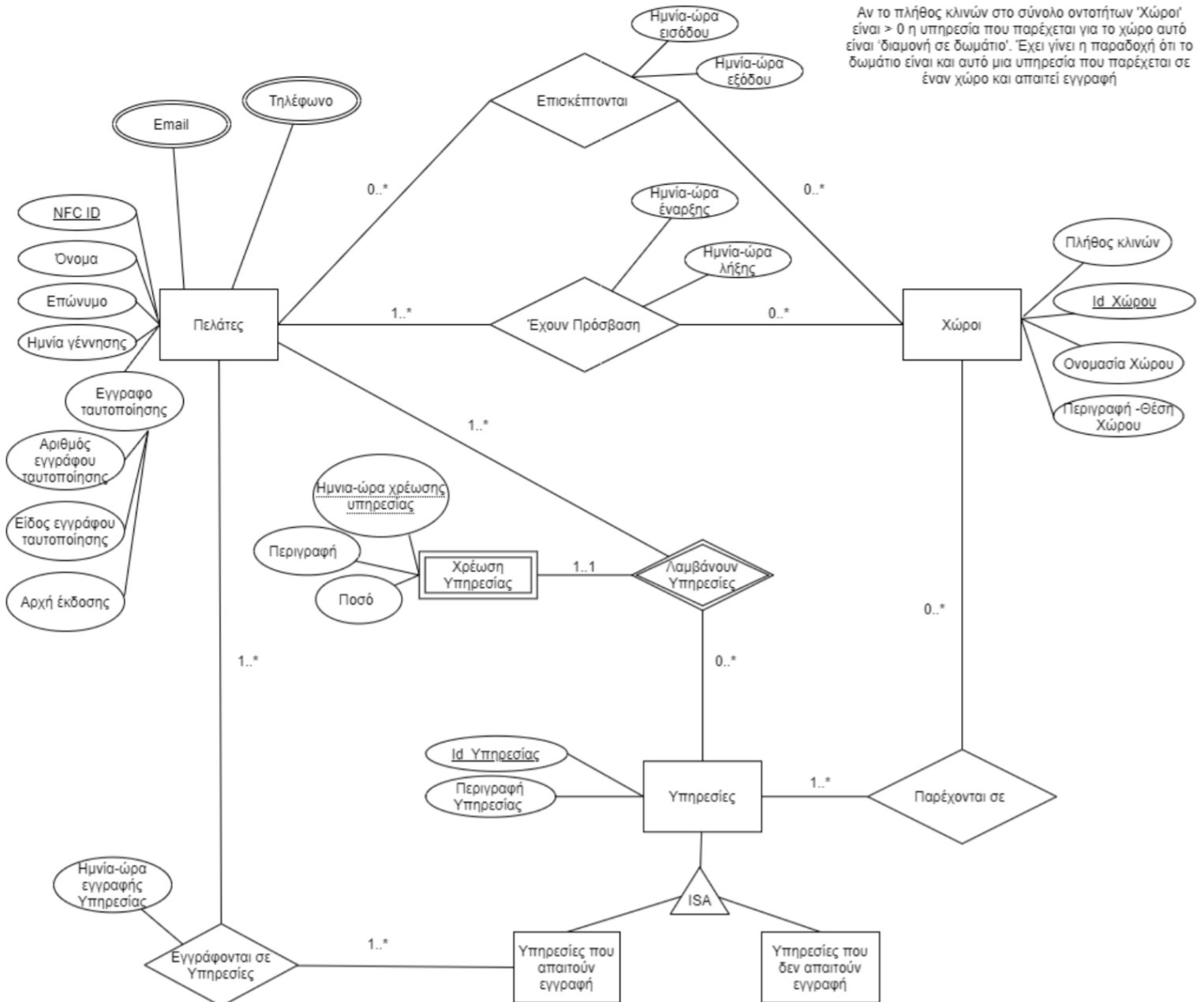
ΑΝΑΦΟΡΑ ΕΞΑΜΗΝΙΑΙΑΣ ΕΡΓΑΣΙΑΣ



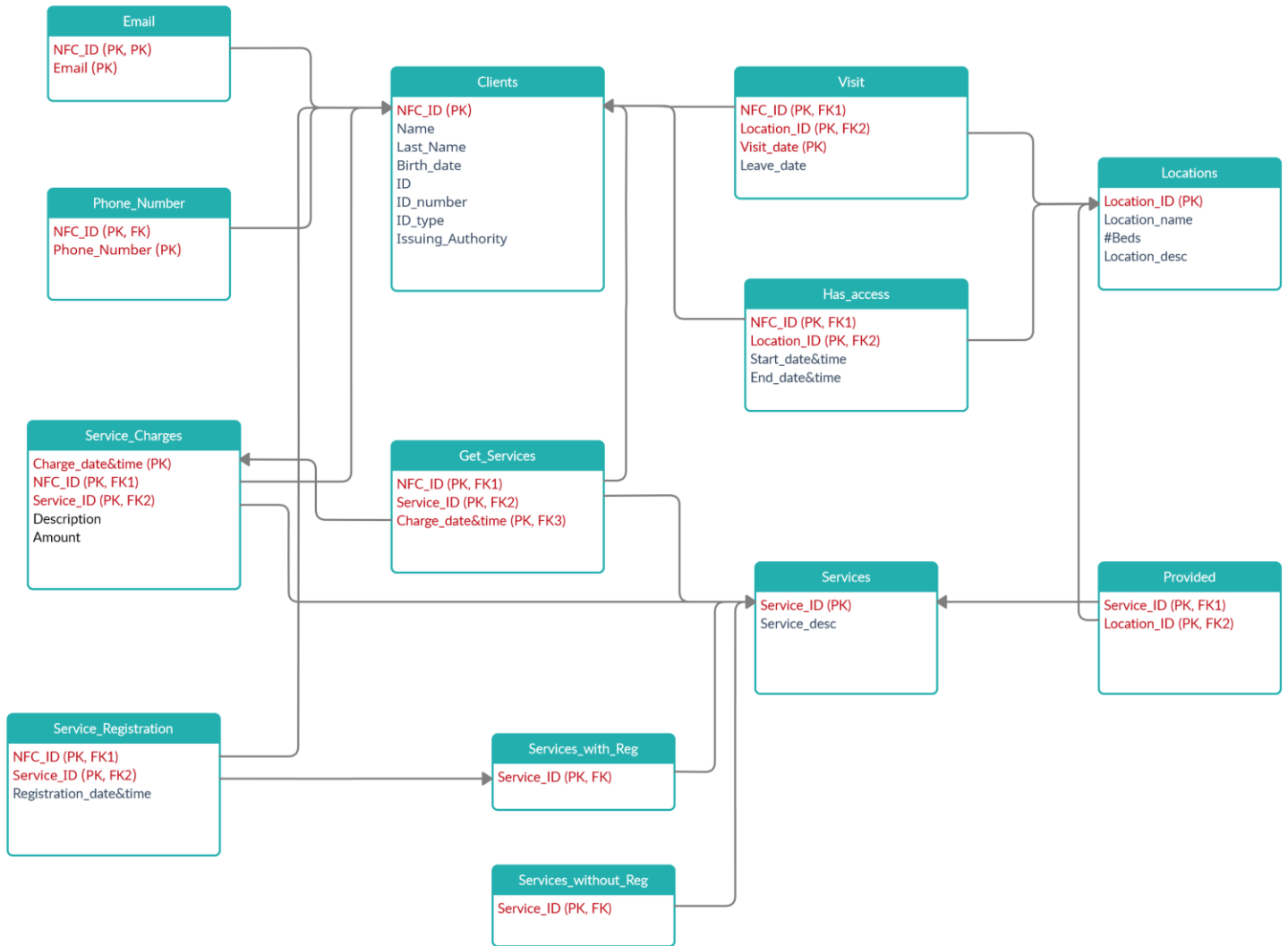
JUNE 21, 2021

ΘΟΔΩΡΗΣ ΑΡΑΠΗΣ – EL18028
ΚΡΙΣ ΚΟΥΤΣΗ – EL18905
ΑΡΙΑΔΝΗ ΚΑΖΔΑΓΛΗ – EL18838

Χρησιμοποιούμε το δοσμένο ER (απλή λύση).



1. Το σχεσιακό διάγραμμα είναι το ακόλουθο:



1a.

Περιορισμοί ακεραιότητας:

- Τα foreign keys που αναφέρονται στο NFC_ID του table clients θα διαγράφονται και θα ενημερώνονται “κασκοδικά” (κατά αλληλουχία), έτσι ώστε όταν ένας πελάτης φύγει από το ξενοδοχείο να μπορούν να διαγραφούν τα δεδομένα του από όλα τα tables και να αποφευχθεί η σύγχυση με τα δεδομένα νέων πελατών που ενδεχομένως να λάβουν το ίδιο NFC_ID.
- Τα foreign keys που αναφέρονται στο Service_ID του table services (όμοια και για το foreign key Service_ID του table service_registration που αναφέρεται στο Service_ID του table services_with_reg) θα ενημερώνονται “κασκοδικά”, αλλά η διαγραφή ενός Service_ID από το service δεν θα επιτρέπεται από κανένα table που περιέχει το αντίστοιχο foreign key. Η επιλογή αυτή έγινε καθώς δεν θέλουμε να γίνει διαγραφή κάποιας υπηρεσίας καθώς ο αριθμός και το είδος τους είναι καθορισμένος εξ αρχής και θεωρούμε πως δεν αλλάζει. Για τον ίδιο λόγο θέσαμε τους ίδιους περιορισμούς για τα foreign keys που αναφέρονται στο Location_ID του table locations.
- Το foreign key του table get_services που αναφέρεται στο Charge_date&time του table service_charges θα διαγράφεται και θα ενημερώνεται “κασκοδικά”, έτσι ώστε όταν κάποιος πελάτης φύγει να “αδειάσει” η βάση δεδομένων από τα δεδομένα για τις χρεώσεις του.

Κλειδιά:

| Tables | Primary keys | Αιτιολόγηση |
|----------------------|--------------------------------------|--|
| clients | NFC_ID | Είναι μοναδικό για κάθε πελάτη |
| visit | NFC_ID, Location_ID, Visit_date | Ο συνδυασμός τους δημιουργεί μοναδικά tuples |
| locations | Location_ID | Είναι μοναδικό για κάθε τοποθεσία |
| Email | NFC_ID, Email | Ο συνδυασμός τους δημιουργεί μοναδικά tuples |
| phone_number | NFC_ID, Phone_Number | Ο συνδυασμός τους δημιουργεί μοναδικά tuples |
| has_access | NFC_ID, Location_ID | Ο συνδυασμός τους δημιουργεί μοναδικά tuples |
| service_charges | NFC_ID, Service_ID | Ο συνδυασμός τους δημιουργεί μοναδικά tuples |
| get_services | NFC_ID, Service_ID, Charge_date&time | Ο συνδυασμός τους δημιουργεί μοναδικά tuples |
| services | Service_ID | Είναι μοναδικό για κάθε υπηρεσία |
| provided | Service_ID, Location_ID | Ο συνδυασμός τους δημιουργεί μοναδικά tuples |
| services_with_reg | Service_ID | Είναι μοναδικό για κάθε υπηρεσία με εγγραφή |
| services_without_reg | Service_ID | Είναι μοναδικό για κάθε υπηρεσία χωρίς εγγραφή |
| service_registration | NFC_ID, Service_ID | Ο συνδυασμός τους δημιουργεί μοναδικά tuples |

Αναφορική ακεραιότητα:

| Tables | Foreign keys | References(Table) |
|----------------------|---|---|
| phone_number | NFC_ID | NFC_ID(clients) |
| Email | NFC_ID | NFC_ID(clients) |
| get_services | NFC_ID, Service_ID, Charge_date&time | NFC_ID(clients), Service_ID(services), Charge_date&time(service_charges) |
| has_access | NFC_ID, Location_ID | NFC_ID(clients), Location_ID(locations) |
| provided | Service_ID, Location_ID | Service_ID(services), Location_ID(locations) |
| services_without_reg | Service_ID | Service_ID(services) |
| services_with_reg | Service_ID | Service_ID(services) |
| service_charges | NFC_ID, Service_ID | NFC_ID(clients), Service_ID(services) |
| service_registration | NFC_ID, Service_ID | NFC_ID(clients), Service_ID(services_with_reg) |
| visit | NFC_ID, Location_ID | NFC_ID(clients), Location_ID(locations) |

Περιορισμοί πεδίου τιμών:

- Το NFC_ID παίρνει τιμές από 1 έως 400, όσα είναι και τα δωμάτια
- Το Service_ID παίρνει τιμές από 1 έως 6, όσες είναι και οι υπηρεσίες
- Το Location_ID παίρνει τιμές από 1 έως 600, γιατί έχουμε 35 τοποθεσίες στο ισόγειο και 5 ορόφους (1ος οροφος → room 1XX, 5ος → room 5XX) Θεωρούμε ότι έχουμε απο 0 έως 4 κλίνες ανά τοποθεσία (Οι τοποθεσίες στο ισόγειο δεν έχουν κλίνες)

1b.

Τα επιπλέον ευρετήρια που ορίσαμε είναι:

- Το ευρετήριο `nfc_id_idx` στο `table service_charges` για το `NFC_ID` προκειμένου να μπορούμε να βρούμε όλες τις χρεώσεις κάποιου πελάτη πολύ γρήγορα, μιας και θα μπορεί να υπάρχουν πολλά διαφορετικά tuples με χρεώσεις για ένα συγκεκριμένο `NFC_ID`.
- Το ευρετήριο `visit_date_idx` στο `table visit` για το `Visit_date` προκειμένου να μπορούμε να βρούμε γρήγορα τους πελάτες που επισκέφτηκαν κάποιο `location` μια συγκεκριμένη χρονική στιγμή.
- Το ευρετήριο `Leave_date_idx` στο `table visit` για το `Leave_date` προκειμένου να μπορούμε να βρούμε γρήγορα τους πελάτες που έφυγαν από κάποιο `location` μια συγκεκριμένη χρονική στιγμή.

1c.

Για την ανάπτυξη της βάσης δεδομένων χρησιμοποιήσαμε την εφαρμογή XAMPP, η οποία μας συνέδεσε με την πλατφόρμα `phpmyadmin` μέσω της οποίας συνθέσαμε το `database`. Η εφαρμογή παρέχει ένα πολύ εύχρηστο GUI για την δημιουργία ενός `database`. Από γλώσσες, χρησιμοποιήσαμε την SQL για την λεπτομερέστερη επεξεργασία της βάσης δεδομένων (Η πλατφόρμα εξάγει αυτόματα τον SQL κώδικα οπότε η τροποποιήσεις έγιναν πάνω σε αυτόν τον κώδικα).

2.

Μπορείτε να δείτε τον κώδικα SQL στο αρχείο `"hotel.sql"` του φακέλου.