



Εθνικό Μετσόβιο Πολυτεχνείο
Σχολή Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών
Τομέας Τεχνολογίας Πληροφορικής και Υπολογιστών

NFC-enabled πρόσβαση σε δωμάτια και υπηρεσίες ξενοδοχείου

Αναφορά εξαμηνιαίας εργασίας

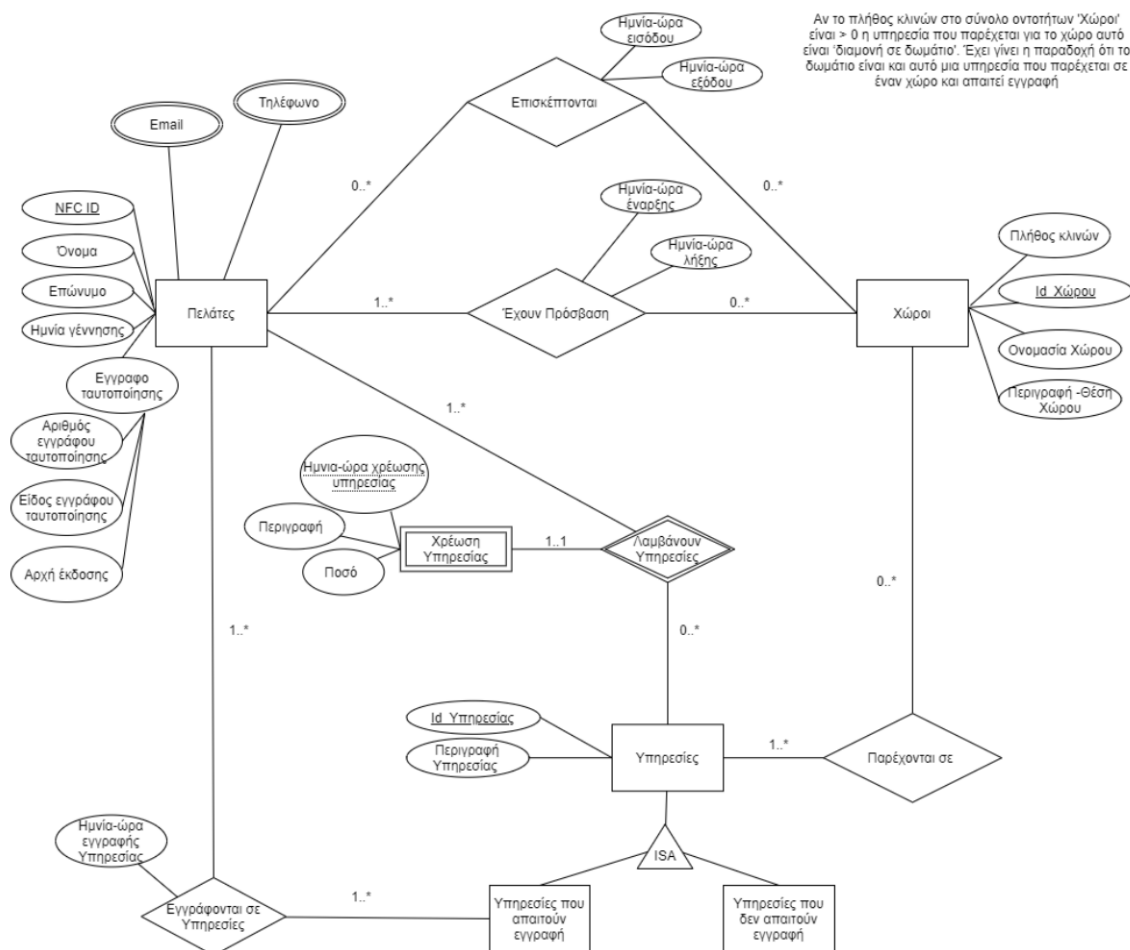
Στο μάθημα “**Βάσεις Δεδομένων**” 2020-21

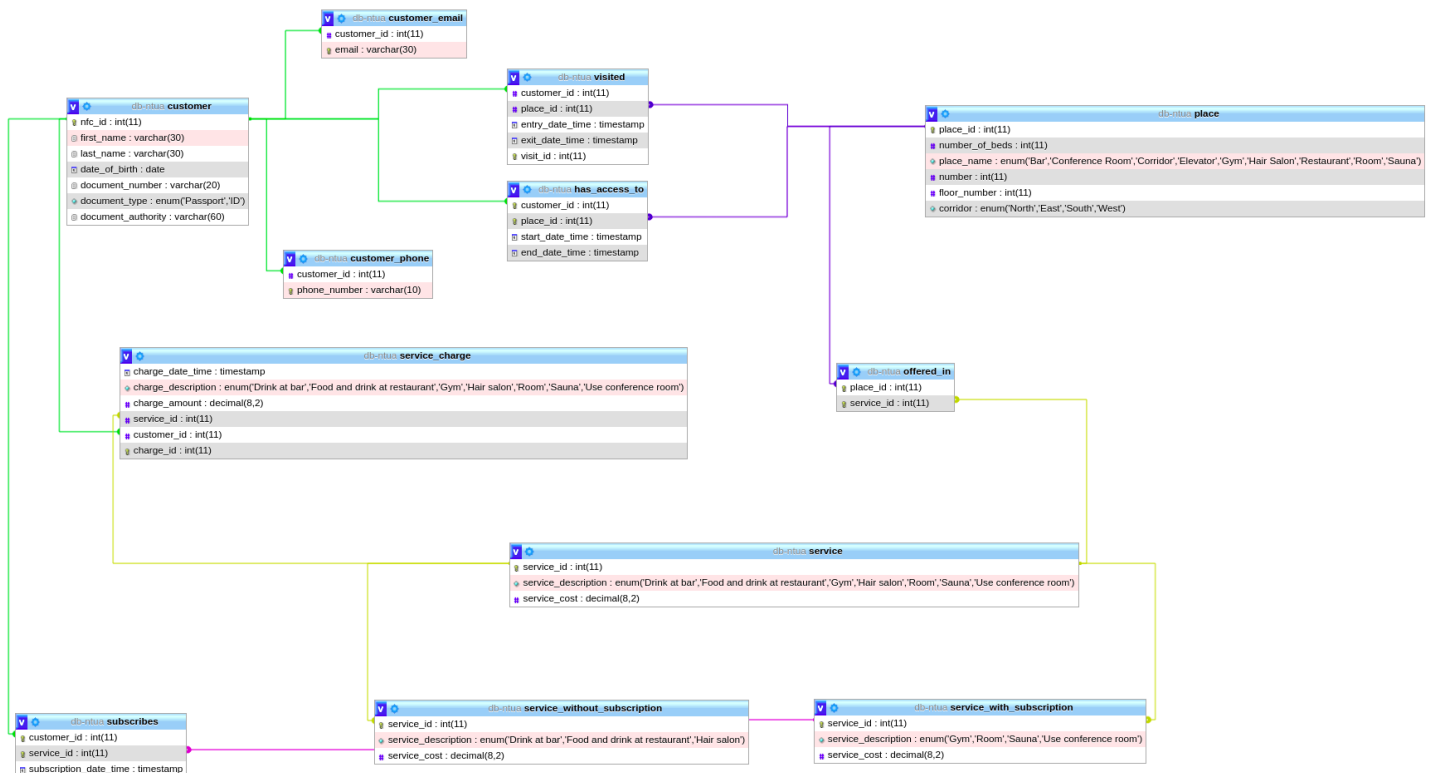
Ημερομηνία παράδοσης: Ιούνιος 2021

Μέλη της ομάδας Μ:	
Νικόλαος Παγώνας	<i>el18175</i>
Αναστάσιος Παπαζαφειρόπουλος	<i>el18079</i>
Νικήτας Τσίννας	<i>el18187</i>

1. Το σχεσιακό διάγραμμα της βάσης δεδομένων

Βασιστήκαμε στο ER της απλής λύσης που μας δόθηκε. Παρακάτω ακολουθεί το ER καθώς και το σχεσιακό διάγραμμα που δημιουργήσαμε για την υλοποίηση της βάσης (το σχεσιακό διάγραμμα φαίνεται σε full-scale στο αρχείο relational-diagram.png):





a. Περιορισμοί που έχουν οριστεί, ανά κατηγορία, με σύντομη αιτιολόγηση για την κάθε επιλογή

- 1η κατηγορία: **primary keys**

Ακολουθώντας και την λογική του ER διαγράμματος (απλή λύση) που μας δόθηκε, στους περισσότερους πίνακες ορίσαμε κάποιο identifier (ουσιαστικά έναν μοναδικό ακέραιο αριθμό) που αποτελούσε το primary key ώστε να έχουμε καλύτερη εποπτεία. Για παράδειγμα, στον πίνακα service, ορίσαμε ως primary key το service_id. Εξαίρεση σε αυτή την πρακτική αποτέλεσαν οι πίνακες customer_phone και customer_email με primary keys: το phone_number και το email αντίστοιχα.

- 2η κατηγορία: **foreign keys**

Στον πίνακα **customer_phone** έχουμε foreign key customer_id με αναφορά στο nfc_id του πίνακα **customer** και μάλιστα με: *on delete cascade* και *on update cascade*, γιατί σε περίπτωση που διαγραφεί ένας πελάτης από τη βάση δε μας ενδιαφέρει να κρατήσουμε το τηλέφωνό του.

Ομοίως στον πίνακα **customer_email**.

Στον πίνακα **service_with_subscription** έχουμε foreign key service_id με αναφορά στο service_id του πίνακα **service** και πάλι με: *on delete cascade* και *on update cascade*.

cascade, γιατί αν διαγραφεί μια υπηρεσία από τη βάση πρέπει να διαγραφεί και από τον υποπίνακα των υπηρεσιών που απαιτούν εγγραφή, αν ανήκει σε αυτόν.

Ομοίως στον πίνακα **service_without_subscription** για τις υπηρεσίες που δεν απαιτούν εγγραφή.

Στον πίνακα **subscribes** έχουμε foreign keys *customer_id* και *service_id* με αναφορά στο *nfc_id* του πίνακα **customer** και στο *service_id* του πίνακα **service_without_subscription** με αμφότερα να ακολουθούν: *on delete cascade* και *on update cascade*, καθώς δε μας ενδιαφέρει να κρατήσουμε τις εγγραφές ενός πελάτη που δεν υπάρχει στη βάση και αντίστοιχα σε μια υπηρεσία που έχει διαγραφεί.

Παρομοίως στον πίνακα **service_charge** με τις εξής διαφορές:

Το foreign key *service_id* αναφέρεται στο *service_id* του πίνακα **service**.

Αμφότερα τα foreign keys ακολουθούν *on delete set null*, καθώς ακόμα και σε περίπτωση που διαγραφεί είτε ο πελάτης, είτε η υπηρεσία από τη βάση θέλουμε να έχουμε μια εποπτεία σχετικά με την υπηρεσία που έχει επιφέρει τα περισσότερα έσοδα στο ξενοδοχείο για κάποια περίοδο.

Στον πίνακα **visited** έχουμε foreign keys *customer_id* με αναφορά στο *nfc_id* του **customer** με *on delete set null* και *on update cascade*, γιατί κι εδώ ακόμη και στην περίπτωση που διαγραφεί ένας πελάτης από τη βάση θέλουμε να έχουμε εποπτεία των πιο πολυσύχναστων χώρων του ξενοδοχείου. Το άλλο foreign key είναι το *place_id* που αναφέρεται στο *place_id* του πίνακα **place** με: *on delete cascade* και *on update cascade*, καθώς οι αλλαγές στη βάση ως προς τους χώρους του ξενοδοχείου θέλουμε να μεταφερθούν και στη σχέση **visited**.

Παρομοίως και στον πίνακα **has_access_to** με τις εξής διαφορές:

Το foreign key *customer_id* ακολουθεί *on delete cascade*, καθώς αν διαγραφεί κάποιος πελάτης δε μας ενδιαφέρει να κρατάμε σε ποιες τοποθεσίες έχει πρόσβαση.

Το foreign key *place_id* ακολουθεί *on delete restrict*, καθώς επειδή έχουμε κάνει την παραδοχή ότι το κάθε δωμάτιο είναι ξεχωριστός χώρος και υπηρεσία, δε θέλουμε να διαγραφεί κάποιο δωμάτιο ενώ μέσα μένει πελάτης.

Τέλος, στον πίνακα **offered_in** έχουμε foreign keys *place_id* με αναφορά στο *place_id* του **place** και *service_id* με αναφορά στο *service_id* του **service**. Αμφότερα τα foreign keys ακολουθούν *on delete cascade* και *on update cascade*, καθώς οι ενημερώσεις και οι διαγραφές στις τοποθεσίες και στις υπηρεσίες θέλουμε να περνούν και στη σχέση **offered_in**.

- 3η κατηγορία: **null/not null**

Ως γενική πρακτική ακολουθήσαμε ότι μία ποσότητα δε μπορεί να γίνει null, εκτός από κάποιες συγκεκριμένες που αναφέρονται παρακάτω. Αυτές είναι: *customer_id*,

service_id (όταν δε χρησιμοποιούνται ως primary keys), exit_date_time (με τη λογική ότι θεωρείται null όση ώρα ο πελάτης παραμένει στην υπό εξέταση τοποθεσία πριν αποχωρήσει), corridor (null εάν η τοποθεσία βρίσκεται στο ισόγειο)

- 4η κατηγορία: **πεδίο τιμών**

Εδώ θα αναφερθούμε σε συγκεκριμένα παραδείγματα, καθώς τα περισσότερα είναι προφανή. Το place_name είναι τύπου enum με τις ακόλουθες τιμές: ("Bar", "Conference Room", "Corridor", "Elevator", "Gym", "Hair Salon", "Restaurant", "Room", "Sauna"). Ομοίως, το corridor με τις ακόλουθες τιμές: ("North", "East", "South", "West"). Το service_cost είναι numeric(8,2), δηλαδή αριθμός με το πολύ 8 ψηφία από τα οποία 2 είναι δεκαδικά. Οι start_date_time και end_date_time είναι timestamps γιατί θέλουμε να κρατάμε και την ημερομηνία και την ώρα. Ομοίως, και οι entry_date_time και exit_date_time.

- 5η κατηγορία: **περιορισμοί οριζόμενοι από τον χρήστη**

- Έλεγχος (και σύμβαση παράλληλα) ότι ο αριθμός ταυτότητας αποτελείται από 8 χαρακτήρες και ο αριθμός διαβατηρίου από 9.
- Έλεγχος ότι ο αριθμός τηλεφώνου αποτελείται από 10 ψηφία.
- Έλεγχος ότι η διεύθυνση email ακολουθεί την εξής μορφή: "%_@_%._%".
- Έλεγχος ότι ο αριθμός των τοποθεσιών υπακούει στους περιορισμούς της εκφώνησης, δηλαδή ο προσδιοριστικός αριθμός number είναι από 1 μέχρι 6 για τα Bars, από 1 μέχρι 5 για τους ανελκυστήρες κ.ο.κ.
- Έλεγχος ότι έχουμε μέχρι 5 ορόφους δωματίων.
- Έλεγχος μη αρνητικού κόστους.
- Έλεγχος μη αρνητικής χρέωσης υπηρεσίας.

b. Το σύστημα και οι γλώσσες προγραμματισμού που χρησιμοποιήθηκαν για την ανάπτυξη της εφαρμογής

Για την ανάπτυξη της εφαρμογής χρησιμοποιήσαμε το πακέτο προγραμμάτων "XAMPP" το οποίο περιέχει εξυπηρετητή ιστοσελίδων https "Apache", το σύστημα διαχείρισης σχεσιακών βάσεων δεδομένων "MySQL" καθώς και έναν διερμηνέα για κώδικα γραμμένο σε γλώσσες προγραμματισμού PHP.

Επομένως, το σύστημα αναπτύχθηκε μέσω των γλωσσών:

- **SQL** αρχικά για την δημιουργία των πινάκων και όψεων καθώς και για τις εισαγωγές δεδομένων στην βάση και έπειτα για την υλοποίηση των SQL ερωτημάτων που χρησιμοποιήθηκαν ενσωματωμένα στον κώδικα php του UI.
- **PHP** για την σύνδεση της διεπαφής του χρήστη με την βάση μέσω της επέκτασης "MySQLi (ή "MySQL improved") που διατίθεται, η οποία αποτελεί

έναν relational database driver.

- **HTML** σε συνδυασμό με php για την ανάπτυξη του User Interface μέσω της οποίας αναπτύχθηκε η λειτουργικότητα της σελίδας και ο σκελετός εμφάνισής της.
- **CSS** για την μορφοποίηση της σελίδας.

c. Αναλυτικά βήματα για το πώς θα μπορούσε να εγκαταστήσει κάποιος την εφαρμογή από την αρχή

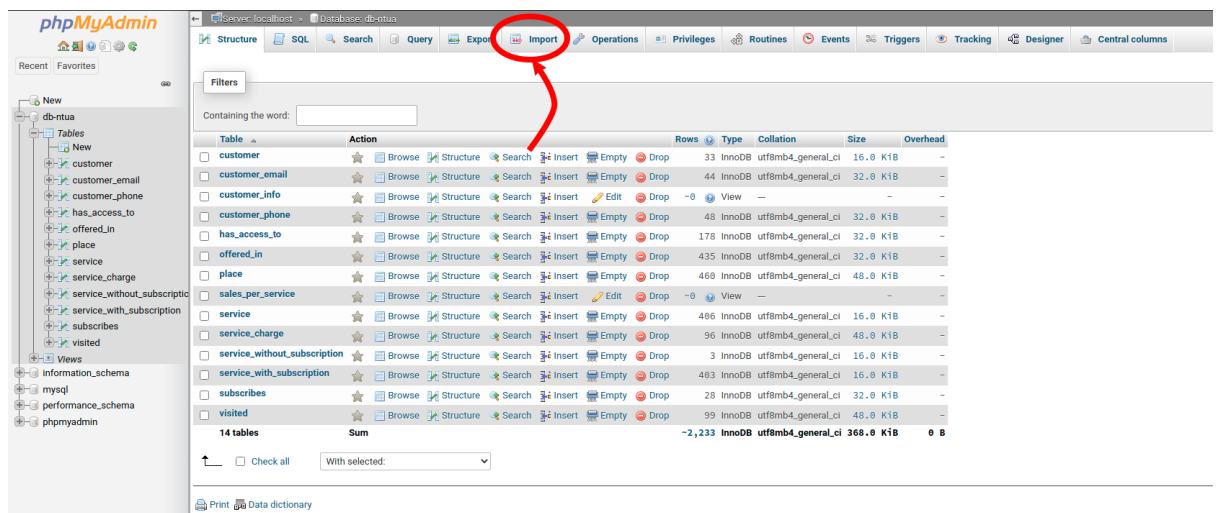
Τα παρακάτω βήματα υποθέτουν ότι έχετε κατεβάσει όλα τα αρχεία της υποβολής (βρίσκονται στον φάκελο Group_M)

1. Κατεβάστε και εγκαταστήστε το εργαλείο XAMPP:

<https://www.apachefriends.org/download.html>

2. Αντιγράψτε τον φάκελο Group_M μέσα στον φάκελο htdocs του XAMPP. Αυτό γίνεται για να μπορεί να τρέξει ο Apache τα αρχεία της εφαρμογής. Για Windows το default directory είναι **C:\xampp\htdocs**, ενώ για Linux το default directory είναι **/opt/lampp/htdocs**.

3. Γράφοντας **localhost/phpmyadmin** σε έναν browser, θα σας εμφανιστεί το home page του phpmyadmin. Πατώντας import στο πάνω μέρος της σελίδας, μπορείτε να διαλέξετε ένα sql αρχείο και να το εκτελέσετε. Κάντε import το αρχείο **setup.sql** (βρίσκεται μέσα στον φάκελο "Group_M/SQL") και πατήστε "Go" κάτω δεξιά. Αυτό θα στήσει την βάση τοπικά, και θα της δώσει το default name "db-ntua". **Μην αλλάξετε** το όνομα, γιατί θα δημιουργήσει θέμα στη σύνδεση με τη βάση.



Databases SQL Status User accounts Export Import Settings Replication Variables Charsets Engines

Importing into the current server

File to import:

File may be compressed (gzip, bzip2, zip) or uncompressed.
A compressed file's name must end in `[.format].[compression]`. Example: `.sql.zip`

Browse your computer: **Choose File** No file chosen (Max: 40MiB)

You may also drag and drop a file on any page.

Character set of the file: utf-8

Partial import:

☒ Allow the interruption of an import in case the script detects it is close to the PHP timeout limit. (This might be a good way to import large files, however it can break transactions.)

Skip this number of queries (for SQL) starting from the first one: 0

Other options:

☒ Enable foreign key checks

Format:

4. Έχοντας τον apache να τρέχει στον localhost, πληκτρολογήστε σε έναν browser: localhost/Group_M/UI/index.php

Θα πρέπει να σας εμφανιστεί αυτή η σελίδα:


Hotel Database online

Πελατολόγιο Πωλήσεις Υπηρεσίες και Χρήσεις Ιστορικό Επισκέψεων Εργαλείο Εντοπισμού Κρουσμάτων Στατιστικά Σχετικά

Καλώς ορίσατε

Σχολή ΗΜΜΥ, ΕΜΠ

Εξαμηνιαία εργασία στο μάθημα Βάσεις Δεδομένων του εξαμήνου



Οδηγός

Για τα αντίστοιχα ερωτήματα πηγαίντε στο:

- Ερώτημα 7: "Υπηρεσίες και Χρήσεις"
- Ερώτημα 8: "Πωλήσεις και Πελατολόγιο"
- Ερώτημα 9: "Ιστορικό Επισκέψεων"
- Ερώτημα 10: "Εργαλείο Εντοπισμού Κρουσμάτων"
- Ερώτημα 11: "Στατιστικά"

Αν εμφανίστηκε αυτή η σελίδα, τότε έχετε εγκαταστήσει επιτυχώς την εφαρμογή.

2. Όλος ο SQL κώδικας που έχει γραφεί για την υποστήριξη της εφαρμογής

Όλος ο SQL κώδικας βρίσκεται στον φάκελο SQL και περιέχει:

- Το αρχείο setup.sql, το οποίο περιλαμβάνει όλους τους ορισμούς των tables, views, triggers και insertions. Αυτό το αρχείο περιέχει όλη την πληροφορία για να στηθεί η βάση από την αρχή (για παράδειγμα, μέσω της επιλογής import του phpmyadmin).
- Το αρχείο queries.sql, το οποίο περιέχει τα ερωτήματα που χρησιμοποιήθηκαν για την λειτουργικότητα του UI. Επειδή όμως αυτά τα ερωτήματα βρίσκονται στα αρχεία της rhr, εν γένει περιέχουν μεταβλητές, οι οποίες εξαρτώνται από το input του χρήστη. Τα ερωτήματα που παρατίθενται σε αυτό το αρχείο περιέχουν μεταβλητές που περικλείονται από '<' και '>' (για παράδειγμα <var>). Για να μπορεί λοιπόν να τρέξει κάποιο ερώτημα εκτός του UI θα πρέπει οι μεταβλητές αυτές να αντικατασταθούν κατάλληλα, όπως ορίζουν οι επεξηγήσεις που βρίσκονται σε μορφή σχολίων μέσα στο queries.sql .

3. Όλα τα αρχεία τεχνοδιαμόρφωσης (configuration files) που χρειάζονται για να εγκατασταθεί από την αρχή η εφαρμογή

Τα αρχεία τεχνοδιαμόρφωσης βρίσκονται στον φάκελο UI. Περιέχουν rhr αρχεία, css αρχεία, εικόνες. Δεν χρειάζεται κάποια παραπάνω επεξεργασία αυτών των αρχείων, αρκεί να ακολουθηθούν οι οδηγίες εγκατάστασης.

4. Video αναλυτικής παρουσίασης συστήματος

Το Video όπου παρουσιάζεται αναλυτικά το σύστημα και η λειτουργικότητά του συστήματος περιέχεται μέσα στο zip με ονομασία αρχείου "video_Group_M.zip". Παρακάτω ακολουθεί ο πίνακας με τα timestamp των ερωτημάτων όπου έχει δοθεί απάντηση (βρίσκεται και σε pdf αρχείο, στο video_Group_M.zip):

Ερώτημα	Αρχή	Τέλος
(a)	0:07	0:25
(b)	0:28	1:35
(c)	1:38	3:42
(e)	3:43	5:09
(f).7	5:10	6:20
(f).9	6:42	7:08

(f).10	7:10	7:58
(f).11	8:00	8:26
(g)	8:29	8:45