**9η Σειρά Ασκήσεων**

**Άσκηση 1:**

**α)** Η αρχιτεκτονική λογισμικού που αρμόζει στο συγκεκριμένο πρόβλημα είναι ο συνδυασμός της αρχιτεκτονικής πελάτη – εξυπηρετητή και της αρχιτεκτονικής τριών επιπέδων. Ποιο συγκεκριμένα σε ότι αφορά διαδικασίες ελέγχου χρησιμοποιείται η αρχιτεκτονική πελάτη – εξυπηρετητή, ενώ στις περιπτώσεις υποβολής αιτημάτων, πληρωμών κτλ. η αρχιτεκτονική τριών επιπέδων

Οι λόγοι για την χρήση της αρχιτεκτονικής πελάτη – εξυπηρετητή είναι οι εξής:

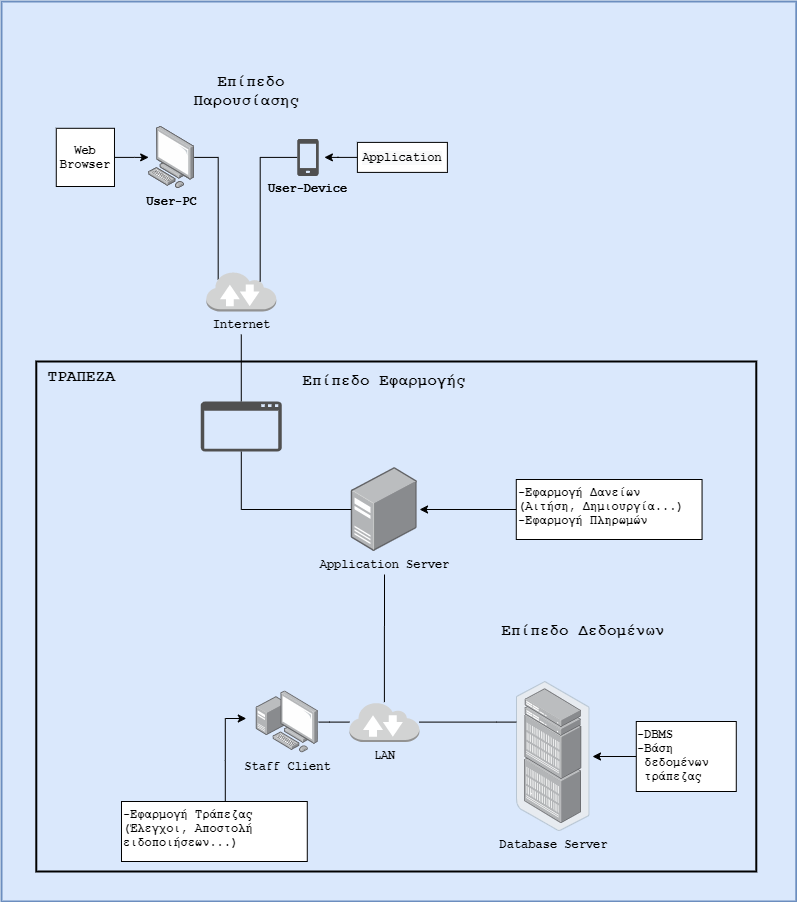
* Αξιοποίηση των υπολογιστικών πόρων των σταθμών εργασίας των υπαλλήλων της τράπεζας, Με αποτέλεσμα να υπάρχει μικρότερος φόρτος στον server
* Απαιτεί λιγότερη συντήρηση και έχει χαμηλότερο κόστος λειτουργίας

Οι λόγοι για την χρήση της αρχιτεκτονικής τριών επιπέδων είναι οι εξής:

* Δίνει την δυνατότητα αναβάθμισης του υλικού και λογισμικού ενός επιπέδου χωρίς να επηρεάζει τα υπόλοιπα
* Πιο αξιόπιστη και ανεξάρτητη λειτουργία των υποκείμενων διακομιστών και υπηρεσιών του συστήματος
* Οι πελάτες έχουν πρόσβαση στην εφαρμογή με την χρήση του διαδικτύου
* Μεγαλύτερη ασφάλεια καθώς η βάση δεδομένων δεν είναι άμεσα προσβάσιμη από το διαδίκτυο

**γ)** Οι τροπικές απαιτήσεις που πρέπει να ληφθούν υπόψη είναι οι εξής:

* Ο χρήστης έχει συμβατό browser στην συσκευή του
* Ο χρήστης έχει σταθερή πρόσβαση στο διαδίκτυο
* Η τράπεζα έχει εγκαταστήσει το απαραίτητο λογισμικό σε όλους τους υπολογιστές και διακομιστές που χρειάζεται
* Άρτιες δικτυακές υποδομές(τοπικές και διαδικτυακές) για γρήγορη εξυπηρέτηση των πελατών, αντοχή σε περιόδους αυξημένης κίνησης και γρήγορη επικοινωνία μεταξύ application server, database server και υπαλλήλων
* Εύχρηστο UI στις υπηρεσίες που χρησιμοποιούν οι πελάτες
* Περιορισμένη προσβασιμότητα στον database server για μεγαλύτερη ασφάλεια των δεδομένων

**β)**

**Άσκηση 2:**

