



Πανεπιστήμιο Πειραιώς

Τμήμα Ψηφιακών Συστημάτων

Π.Μ.Σ. Διδακτική της Τεχνολογίας και
Ψηφιακά Συστήματα
Κατεύθυνση Δικτυοκεντρικών Συστημάτων

Διπλωματική Εργασία

Ανάλυση, Σχεδιασμός και Υλοποίηση Πληροφοριακού
Συστήματος με χρήση SOA και Web Services

Πανάρετου Αλεξάνδρα-Αγγελική - ΜΕ09069

Επιβλέπων Καθηγητής: Μαρίνος Θεμιστοκλέους

Αθήνα, Δεκέμβριος 2013



Περιεχόμενα

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ



Ευρετήριο Εικόνων

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ



Περίληψη

Η παρούσα εργασία μελετά την ανάπτυξη και τη χρήση των Υπηρεσιών Παγκόσμιου Ιστού (Web Services) για την αυτοματοποίηση και απλούστευση των διαφόρων διαδικασιών. Σκοπός της παρούσας εργασίας είναι η δημιουργία μίας ηλεκτρονικής υπηρεσίας, η οποία αυτοματοποιεί τη διαδικασία αίτησης και χορήγησης αδειών για μία εταιρεία πληροφοριακών συστημάτων, την ABΓ Πληροφοριακά Συστήματα ΕΠΕ.

Η εφαρμογή που αναπτύχθηκε αποτελεί μία πρόταση για την αυτοματοποίηση των διαφόρων διεργασιών που πρέπει να πραγματοποιηθούν για την ολοκλήρωση της αίτησης και χορήγησης άδειας. Η διαδικασία αυτή περιλαμβάνει τις εξής υποδιαδικασίες:

1. Αίτηση Άδειας Εργαζομένου
2. Αξιολόγηση Άδειας Εργαζομένου

Οι παραπάνω υποδιαδικασίες, μπορούν να αναλυθούν περαιτέρω, ακολούθως:

- 1.1. Είσοδος στο Σύστημα και Συλλογή Δεδομένων Χρήστη
- 1.2. Παρουσίαση Αιτήσεων Εργαζομένου
- 1.3. Δημιουργία Νέας Αίτησης Άδειας Εργαζομένου
- 1.4. Ενημέρωση Υπεύθυνου Ανθρώπινου Δυναμικού

- 2.1. Είσοδος στο Σύστημα και Συλλογή Δεδομένων Χρήστη
- 2.2. Παρουσίαση Εκκρεμών Αιτήσεων Εργαζομένων
- 2.3. Αξιολόγηση (αποδοχή / απόρριψη) Αιτήσεων Εργαζομένων
- 2.4. Ενημέρωση Εργαζομένων

Η παραπάνω διαδικασία υλοποιήθηκε με βάση τις Υπηρεσιοστρεφείς Αρχιτεκτονικές (Service-Oriented Architectures) και συγκεκριμένα με χρήση του .NET Framework. Το .NET Framework είναι μία πλατφόρμα ανάπτυξης αντικειμενοστραφών εφαρμογών, ανοικτού κώδικα, η οποία προσφέρει μία εκτεταμένη βιβλιοθήκη κλάσεων και επιτρέπει σε εφαρμογές που είναι γραμμένες σε διαφορετικές γλώσσες να επικοινωνούν άμεσα μεταξύ τους, με τη χρήση μίας ενδιάμεσης γλώσσας (Common Language Infrastructure).



Επιπλέον, χρησιμοποιήθηκε το ASP .NET MVC 4 Framework, το οποίο εμπλουτίζει το .NET Framework με πρόσθετες κλάσεις. Το Framework αυτό στηρίζεται στην αρχιτεκτονική Model-View-Controller, η οποία γίνεται όλο και πιο διαδεδομένη, βάση των πλεονεκτημάτων που προσφέρει. Οι υπηρεσίες αναπτύχθηκαν στο Visual Studio 2010 χρησιμοποιώντας τη γλώσσα προγραμματισμού C#. Τέλος, η βάση της εφαρμογής στήθηκε σε Microsoft SQL Server 2008 R2.



1. Εισαγωγή

Στο κεφάλαιο αυτό πραγματοποιείται μία εισαγωγή στην τεχνολογία των Web Services, πάνω στην οποία έχει βασιστεί η παρούσα εργασία για τη δημιουργία μίας εφαρμογής. Στη συνέχεια του κεφαλαίου παρουσιάζεται το ερευνητικό πρόβλημα, ο σκοπός και οι αντικειμενικοί στόχοι της εργασίας, καθώς και η διάρθρωσή της.

1.1. Εισαγωγή στα Web Services

Οι υπηρεσίες διαδικτύου αποτελούν μία αναδυόμενη τεχνολογία, η οποία επιτρέπει σε ανόμοιες εφαρμογές, οι οποίες εκτελούνται σε διαφορετικές μηχανές, να ανταλλάσσουν δεδομένα και να συνεργάζονται μεταξύ τους, χωρίς να απαιτούνται επιπρόσθετα ιδιόκτητα λογισμικά ή τμήματα υλικού.

Μέχρι πρόσφατα, η δημιουργία και η παροχή υπηρεσιών στο διαδίκτυο γινόταν με ακαθόριστο τρόπο, ο οποίος διέφερε από επιχείρηση σε επιχείρηση. Έτσι, ενώ υπήρχε ένα αρκετά μεγάλο σύνολο από παρεχόμενες υπηρεσίες στο διαδίκτυο, για να μπορέσει κάποιος να τις χρησιμοποιήσει, έπρεπε για κάθε μία υπηρεσία να μελετήσει τον τρόπο λειτουργίας της, τον τρόπο με τον οποίο θα την καλέσει και γενικά να προσαρμόσει το σύστημά του, ώστε να είναι συμβατό με τον παροχέα της συγκεκριμένης υπηρεσίας.

Πλέον, τα Web Services έρχονται να αλλάξουν τα πράγματα, καθώς οι εφαρμογές που στηρίζονται στο παράδειγμα των υπηρεσιών διαδικτύου, μπορούν να ανταλλάσσουν μεταξύ τους δεδομένα, ασχέτως γλώσσας προγραμματισμού, πλατφόρμας ή πρωτοκόλλων επικοινωνίας. Οι περισσότερες επιχειρήσεις που δημιουργούν υπηρεσίες στο διαδίκτυο, βασίζονται στην αρχιτεκτονική των υπηρεσιών ιστού, όπως αυτή καθορίζεται από το W3C. Αυτό πρακτικά σημαίνει ότι επικρατεί ένας καθολικά αποδεκτός τρόπος δημιουργίας και χρήσης διαδικτυακών υπηρεσιών και η εκμετάλλευσή τους είναι πολύ ευκολότερη και αποδοτικότερη.



1.2. Ερευνητικό Πρόβλημα

Το πρόβλημα που καλείται να επιλύσει η παρούσα εργασία είναι η αυτοματοποίηση της διαδικασίας αίτησης και παροχής άδειας στους εργαζομένους μίας μικρής εταιρείας πληροφοριακών συστημάτων, την «ΑΒΓ Πληροφορικά Συστήματα Ε.Π.Ε.». Η εφαρμογή που δημιουργήθηκε, παρέχει τη δυνατότητα στους εργαζομένους της εταιρείας να βλέπουν τις διαθέσιμες ημέρες άδειας και να αιτούνται άδεια για συγκεκριμένες ημερομηνίες. Η αίτηση αυτή θα αξιολογείται από τον Υπεύθυνο Ανθρώπινου Δυναμικού και θα μπορεί να γίνεται αποδεκτή, ολικώς ή μερικώς, ή και να απορρίπτεται.

1.3. Σκοπός και Αντικειμενικοί Στόχοι

1.3.1. Σκοπός της Εργασίας

Σκοπός της παρούσας εργασίας είναι η μελέτη των Υπηρεσιοστρεφών Τεχνολογιών και Αρχιτεκτονικών και στη συνέχεια η ανάλυση, ο σχεδιασμός και η υλοποίηση μίας ηλεκτρονικής υπηρεσίας για την αυτοματοποίηση της διαδικασίας αίτησης και παροχής αδειών σε μία εταιρεία.

Η υπηρεσία αυτή απευθύνεται τόσο στους εργαζόμενους της εταιρείας που θέλουν να αιτηθούν άδεια, όσο και για τον Υπεύθυνο του Ανθρώπινου Δυναμικού, ο οποίος χρειάζεται να παρακολουθεί την κατάσταση του προσωπικού ανά πάσα στιγμή και να φροντίζει ότι οι άδειες παρέχονται σύμφωνα με τους κανονισμούς της εταιρείας και χωρίς να δημιουργούν πρόβλημα στην ορθή λειτουργία της.

1.3.2. Αντικειμενικοί Στόχοι της Εργασίας

Το αντικείμενο της εργασίας αποτελείται από τέσσερις βασικούς στόχους, οι οποίοι είναι οι ακόλουθοι:

- Η μελέτη της βιβλιογραφίας και η κατανόηση των βασικών εννοιών των Υπηρεσιοστρεφών Αρχιτεκτονικών και των Υπηρεσιών Παγκόσμιου Ιστού, καθώς και τα χαρακτηριστικά και τα οφέλη αυτών.



- Η ανάλυση και ο σχεδιασμός της Επιχειρησιακής Διαδικασίας Ηλεκτρονικής Διαχείρισης των Αδειών μίας εταιρείας, με βάση τις έννοιες που μελετήθηκαν, καθώς και η σύγκρισή του με την τωρινή διαδικασία που ακολουθείται.
- Η υλοποίηση της αυτοματοποιημένης υπηρεσίας Αίτησης και Διαχείρισης Αδειών για μία εταιρεία.
- Η περιγραφή, επεξήγηση και τεκμηρίωση της εφαρμογής που υλοποιήθηκε.

1.4. Δομή της Εργασίας

Η παρούσα εργασία αποτελείται από τις ακόλουθες ενότητες

- **Κεφάλαιο 1:** Στο πρώτο κεφάλαιο γίνεται μία παρουσίαση του ερευνητικού προβλήματος και επεξηγούνται οι αντικειμενικοί στόχοι και ο σκοπός της εργασίας. Επίσης, παρουσιάζεται περιληπτικά η δομή της εργασίας.
- **Κεφάλαιο 2:** Το δεύτερο κεφάλαιο περιλαμβάνει την ανασκόπηση της βιβλιογραφίας, η οποία χρησιμοποιήθηκε κατά τη σύνταξη της παρούσας εργασίας και την ανάπτυξη της εφαρμογής. Η εν λόγω βιβλιογραφία αφορά στις Υπηρεσιοστρεφείς Αρχιτεκτονικές (Service – Oriented Architectures) και στις υπηρεσίες παγκόσμιου ιστού (Web Services), καθώς και στα χαρακτηριστικά, τα πλεονεκτήματα και τα μειονεκτήματα αυτών.
- **Κεφάλαιο 3:** Στο τρίτο κεφάλαιο πραγματοποιείται μία εκτενέστερη ανάλυση του ερευνητικού προβλήματος, αναλύονται οι απαιτήσεις του συστήματος και περιγράφεται η προτεινόμενη λύση.
- **Κεφάλαιο 4:** Το τέταρτο κεφάλαιο περιλαμβάνει τον αρχιτεκτονικό σχεδιασμό της επιχειρηματικής διαδικασίας και τη διαγραμματική αναπαράστασή της. Επίσης, περιγράφει την υλοποίηση της εφαρμογής, αναφέροντας το σχεδιασμό της βάσης δεδομένων, τη δομή του project και τα προγραμματιστικά εργαλεία που χρησιμοποιήθηκαν.
- **Κεφάλαιο 5:** Το πέμπτο κεφάλαιο περιλαμβάνει την υλοποίηση και την παρουσίαση του συστήματος. Περιγράφεται η λειτουργικότητα και οι οθόνες του συστήματος (print screens) σε διάφορα πιθανά σενάρια χρήσης.
- **Κεφάλαιο 6:** Στο τελευταίο κεφάλαιο της εργασίας παρατίθενται τα συμπεράσματα και γίνεται αξιολόγηση της επιτυχίας των στόχων που



τέθηκαν στην εισαγωγή. Επιπλέον, παρατίθενται παρατηρήσεις για μελλοντικές βελτιώσεις και επεκτάσεις.

- **Βιβλιογραφία:** Η βιβλιογραφία περιλαμβάνει τις πηγές, ηλεκτρονικές και μη, οι οποίες χρησιμοποιήθηκαν κατά τη συγγραφή της παρούσας εργασίας.



2. Βιβλιογραφική Ανασκόπηση

Στο κεφάλαιο αυτό πραγματοποιείται μία ανασκόπηση της βιβλιογραφίας που χρησιμοποιήθηκε κατά τη συγγραφή της παρούσας εργασίας και κατά την υλοποίηση της ηλεκτρονικής υπηρεσίας Αίτησης και Χορήγησης Αδειών της εταιρείας «ΑΒΓ Πληροφορικά Συστήματα Ε.Π.Ε». Αναλύονται τα χαρακτηριστικά, τα πλεονεκτήματα και τα μειονεκτήματα των τεχνολογιών αυτών και αναφέρονται κάποιες περιπτώσεις, όπου τα Web Services βρίσκουν σήμερα εφαρμογή.

2.1. Υπηρεσίες Ιστού – Web Services

Τα Web Services είναι διαπλατφορμικές, αυτόνομες μονάδες ανοιχτού κώδικα, οι οποίες εκτελούν λειτουργίες που αυτοματοποιούν μέρος μίας επιχειρηματικής διαδικασίας. Τα Web Services μπορούν να περιγραφούν, να δημοσιευθούν, να εντοπιστούν, να συντονιστούν και να διαμορφωθούν με τη χρήση XML τεχνικών, με σκοπό να αναπτυχθούν διανεμημένες εφαρμογές με δυνατότητες συνεργασίας μεταξύ τους.

Σημαντικό χαρακτηριστικό των εφαρμογών που βασίζονται σε Web Services αποτελεί το γεγονός ότι είναι χαλαρής σύνδεσης μονάδες λογισμικού (loosely-coupled), το οποίο σημαίνει ότι μπορούν να ανταλλάσσουν δεδομένα μεταξύ τους, αδιαφορώντας για τη γλώσσα προγραμματισμού, την πλατφόρμα ή το πρωτόκολλο διαδικτύου που χρησιμοποιείται και χωρίς τη χρήση επιπρόσθετου λογισμικού ή υλικού. Αυτό επιτυγχάνεται χάρη στο τυποποιημένο σύστημα ανταλλαγής μηνυμάτων που χρησιμοποιούν, το οποίο βασίζεται στην XML τεχνολογία.

Επίσης, κάθε υπηρεσία διαδικτύου αποτελεί μία αυτοσυντηρούμενη (self contained) μονάδα λογισμικού. Αυτή η μονάδα περιγράφει η ίδια τα χαρακτηριστικά της, όπως τις διαθέσιμες λειτουργίες, τις παραμέτρους που χρησιμοποιεί, τους τύπους δεδομένων κ.τ.λ., ώστε να μπορούν εύκολα άλλες υπηρεσίες να προσδιορίσουν τη λειτουργικότητά της και να γνωρίζουν τον τρόπο με τον οποίο θα πρέπει να τη χρησιμοποιήσουν. Οι υπηρεσίες διαδικτύου διανέμονται μέσω internet, χρησιμοποιώντας ήδη υπάρχοντα πρωτόκολλα μεταφοράς, όπως το HTTP



και μπορούν να βρεθούν δυναμικά και να ενσωματωθούν εύκολα σε διάφορες άλλες εφαρμογές.

Τα Web Services στηρίζονται στο υπηρεσιοστρεφές μοντέλο, το οποίο υποστηρίζει τη συνύπαρξη και συνεργασία των παρακάτω οντοτήτων:

- **Πάροχος Υπηρεσίας (Service Provider):** Είναι η υπηρεσία που καλείται να υποστηρίξει τη λειτουργία άλλων υπηρεσιών ή εφαρμογών, εκτελώντας μία σειρά από ενέργειες. Λαμβάνει αιτήματα από άλλες υπηρεσίες ή εφαρμογές, τα οποία εκτελεί και επιστρέφει σε αυτές το αποτέλεσμα.
- **Πελάτης Υπηρεσίας (Service Requestor):** Είναι η υπηρεσία που καλεί έναν πάροχο υπηρεσίας, με σκοπό να θέσει κάποιο αίτημα προς εξυπηρέτηση. Μία υπηρεσία μπορεί ταυτόχρονα να είναι και πάροχος και πελάτης υπηρεσίας. Αυτό μπορεί να συμβεί εάν μία υπηρεσία, για να εξυπηρετήσει κάποιο αίτημα (ως πάροχος υπηρεσίας) χρειάζεται να καλέσει μία δεύτερη υπηρεσία για να πάρει ένα αποτέλεσμα (πελάτης υπηρεσίας).
- **Μητρώο Υπηρεσιών (Registry):** Αποτελεί ένα ευρετήριο καταχώρησης περιγραφών υπηρεσιών, με σκοπό τη διευκόλυνση εύρεσης και χρήσης των διαθέσιμων υπηρεσιών. Κάθε υπηρεσία, ενημερώνει το μητρώο για την ύπαρξή της και παρέχει πληροφορίες για τον τρόπο λειτουργίας της. Το μητρώο με τη σειρά του, δημιουργεί αντίστοιχες καταχωρήσεις, ανάμεσα στις οποίες, μπορεί ένας πελάτης υπηρεσίας να αναζητήσει μία υπηρεσία, βάσει συγκεκριμένων κριτηρίων αναζήτησης.

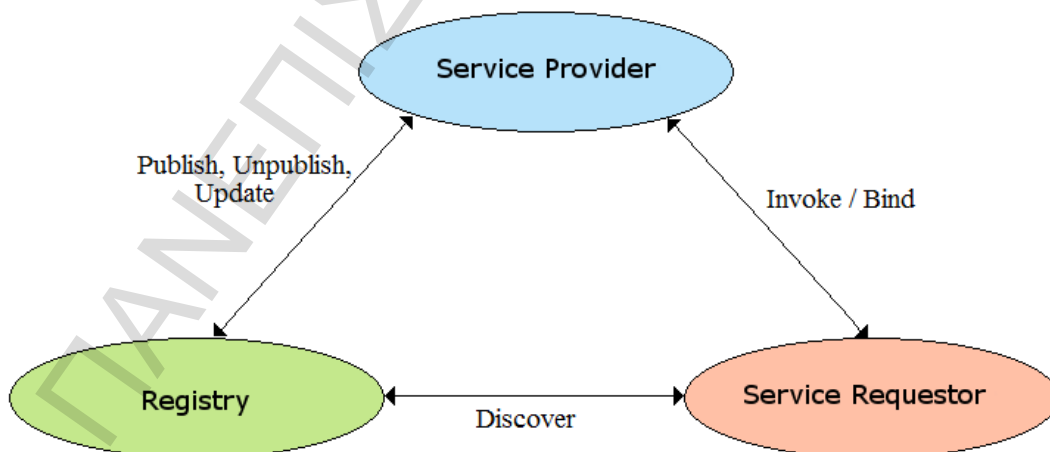
Η συνεργασία των παραπάνω οντοτήτων πραγματοποιείται μέσω μίας σειράς λειτουργιών, όπως αυτές περιγράφονται παρακάτω:

- **Λειτουργία Έκδοσης (Publish):** Η λειτουργία αυτή χρησιμοποιείται, ώστε ένας πάροχος υπηρεσίας να ενημερώσει τις άλλες υπηρεσίες και εφαρμογές για την ύπαρξή του. Μέσω αυτής της λειτουργίας περιγράφει και καταχωρεί την υπηρεσία του στο Μητρώο Υπηρεσιών. Κατά τη διαδικασία της περιγραφής, χρησιμοποιείται η γλώσσα περιγραφής WSDL και παρέχονται οι πληροφορίες του παρόχου, οι πληροφορίες της υπηρεσίας και κάποιες τεχνικές πληροφορίες, που αφορούν στην υλοποίηση και κλήση της υπηρεσίας. Κατά τη διαδικασία της

καταχώρησης, αποθηκεύονται οι παραπάνω πληροφορίες στο Μητρώο, ώστε να είναι προσβάσιμες από τους πελάτες υπηρεσιών.

- **Λειτουργία Αναζήτησης (Discover):** Η λειτουργία αυτή πραγματοποιείται, ώστε ένας πελάτης υπηρεσίας, με τη βοήθεια του μητρώου υπηρεσιών, να μπορέσει να βρει το κατάλληλο web service, που να καλύπτει τις ανάγκες του. Αρχικά, ο πελάτης υπηρεσίας υποβάλει ένα ερώτημα στο μητρώο υπηρεσιών, βάσει συγκεκριμένων κριτηρίων αναζήτησης. Από τη λίστα αποτελεσμάτων που θα λάβει, πραγματοποιεί την επιλογή, είτε χειροκίνητα, επιθεωρώντας αρχικά τη λίστα και κάνοντας στη συνέχεια την επιλογή, είτε αυτόματα, επιλέγοντας την υπηρεσία που ανταποκρίνεται καλύτερα στα κριτήρια αναζήτησης που τέθηκαν.
- **Λειτουργία Σύνδεσης και Εκτέλεσης (Bind and Invoke):** Η λειτουργία αυτή χρησιμοποιείται, ώστε να πραγματοποιηθεί η τελική σύνδεση μεταξύ πελάτη και παρόχου υπηρεσίας και να εκτελεστεί η απαιτούμενη εργασία. Σε αυτό το στάδιο χρησιμοποιούνται οι πληροφορίες που ανακτήθηκαν από το Μητρώο Υπηρεσιών, ώστε να πραγματοποιηθεί σωστά η σύνδεση και κλήση της υπηρεσίας. Η σύνδεση μπορεί να είναι είτε άμεση, μεταξύ πελάτη και παρόχου υπηρεσίας, είτε έμμεση, με τη βοήθεια ενός διαμεσολαβητή υπηρεσιών.

Η αλληλεπίδραση των παραπάνω οντοτήτων και η συσχέτιση των λειτουργιών που αναφέρθηκαν, παρουσιάζονται στο παρακάτω σχήμα:



Εικόνα 1: Αναπαράσταση οντοτήτων και λειτουργιών του υπηρεσιοστρεφούς μοντέλου των web services



Οι βασικές τεχνολογίες που χρησιμοποιούνται στα web services είναι οι ακόλουθες:

- XML (eXtensible Markup Language): Χρησιμοποιείται για την μεταφορά δεδομένων και συντελεί στην ανεξαρτητοποίηση των web services από κάθε λειτουργικό σύστημα, πλατφόρμα ή γλώσσα προγραμματισμού.
- SOAP (Simple Object Access Protocol): Αποτελεί πρότυπο κανάλι επικοινωνίας, το οποίο χρησιμοποιεί τεχνολογίες XML, ώστε να μπορεί να ανταλλαχθεί πάνω από ποικίλα δικτυακά πρωτόκολλα.
- WSDL (Web Services Description Languages): Αποτελεί μία πρότυπη περιγραφική γλώσσα, η οποία χρησιμοποιείται για την περιγραφή των παρεχόμενων υπηρεσιών.
- UDDI (Universal Description, Discovery and Integration): Καθορίζει ένα σύνολο από υπηρεσίες, οι οποίες θα υποστηρίζουν την περιγραφή και την ανακάλυψη των παρόχων web services, των διαθέσιμων web services και των τεχνικών διεπαφών, οι οποίες μπορούν να χρησιμοποιηθούν ώστε να έχει κάποιος χρήστης πρόσβαση σε αυτές τις υπηρεσίες.

Οι παραπάνω τεχνολογίες αναλύονται εκτενέστερα σε επόμενα κεφάλαια.

2.1.1. Τύποι Web Services

Τα Web Services διακρίνονται σε δύο βασικούς τύπους: τις απλές υπηρεσίες και τις σύνθετες υπηρεσίες.

- **Απλές Υπηρεσίες:** Χρησιμοποιούνται για να εκτελέσουν κάποια απλή εργασία. Λαμβάνουν ένα αίτημα, πραγματοποιούν μία σειρά από απλές ενέργειες, απαραίτητες για την λήψη του αποτελέσματος και στη συνέχεια αποστέλλουν την απόκρισή τους. Ουσιαστικά αναμένουν αιτήματα προς εξυπηρέτηση, χωρίς να κάνουν κάτι άλλο.
- **Σύνθετες Υπηρεσίες:** Χρησιμοποιούνται για να εκτελέσουν πιο σύνθετες εργασίες. Συνήθως περιλαμβάνουν κάποια λειτουργικότητα οι ίδιες, αλλά χρησιμοποιούνται κυρίως για να συντονίσουν άλλες υπηρεσίες, των οποίων ενσωματώνουν τη λειτουργικότητα, ώστε να καταλήξουν σε κάποιο αποτέλεσμα.



2.1.2. Ιεραρχία Web Services

Κατά καιρούς έχουν προταθεί διάφορα μοντέλα, τα οποία κατηγοριοποιούν τις υπηρεσίες παγκόσμιου ιστού. Παρακάτω παρουσιάζεται η προσέγγιση των Rose et al. (2008). Στο ανώτερο επίπεδο της ιεραρχίας βρίσκονται οι επιχειρησιακές διαδικασίες. Αυτές σπάνε σε μικρά τμήματα, τα οποία εκτελούνται από τις διάφορες υπηρεσίες και ολοκληρώνουν τελικά την επιχειρησιακή διαδικασία.

Τα web services χωρίζονται σε 6 διαφορετικές κατηγορίες, ανάλογα με τη λειτουργικότητα που προσφέρουν και τις ενέργειες που πραγματοποιούν.

- **Επιχειρησιακές Υπηρεσίες (Business Services):** Αποτελούν υψηλού επιπέδου υπηρεσίες, οι οποίες συνθέτουν τη λειτουργικότητά τους από υπηρεσίες κατώτερου επιπέδου. Είναι οι πιο σύνθετες από όλες τις υπηρεσίες και παρέχουν τη δυνατότητα επαναχρησιμοποίησης κοινών λειτουργιών, είτε εντός είτε εκτός ενός οργανισμού.
- **Τοπικές Υπηρεσίες (Domain Services):** Όπως προδίδει και το όνομά τους, οι τοπικές υπηρεσίες σχετίζονται συνήθως με συγκεκριμένες περιοχές και σπάνια χρησιμοποιούνται εκτός της περιοχής στην οποία ορίστηκαν.
- **Υπηρεσίες Ολοκλήρωσης (Integration Services):** Οι υπηρεσίες αυτές χρησιμοποιούνται για να ολοκληρώσουν υπάρχουσες εφαρμογές, όπως legacy systems, ERP applications κτλ. Η πολυπλοκότητα της λειτουργικότητάς τους ποικίλει ανάλογα με τη φύση των εφαρμογών που ολοκληρώνουν.
- **Εξωτερικές Υπηρεσίες (External Services):** Οι υπηρεσίες αυτές χρησιμοποιούνται για να διασφαλίσουν την επικοινωνία του οργανισμού με εξωτερικές προς αυτόν υπηρεσίες, οι οποίες είναι απαραίτητες για την ολοκλήρωση κάποιας διαδικασίας.
- **Υπηρεσίες Υποστήριξης (Utility Services):** Οι υπηρεσίες αυτές εκτελούν βασικές λειτουργίες και χρησιμοποιούνται από αρκετές υπηρεσίες και διαδικασίες.
- **Βασικές Υπηρεσίες (Foundation Services):** Οι υπηρεσίες αυτές εκτελούν πολύ απλές λειτουργίες και χρησιμοποιούνται για τη σύνθεση, ενορχήστρωση και χορογραφία των υπηρεσιών υψηλότερου επιπέδου.



2.1.3. Πλεονεκτήματα Web Services

Τα πλεονεκτήματα που μπορούν να προσφέρουν τα Web Services είναι πολλά, μερικά από τα οποία αναφέρονται παρακάτω:

- Διαλειτουργικότητα: Τα Web Services παρέχουν ανεξαρτησία, τόσο από το λειτουργικό σύστημα, όσο και από το υλικό (hardware). Οποιοδήποτε πρόγραμμα είναι συμβατό με την τεχνολογία αυτή, μπορεί να χρησιμοποιήσει αυτή την υπηρεσία.
- Ενσωμάτωση: Για την δημιουργία ενός web service σε κάποιο υπάρχον λογισμικό, δεν απαιτούνται αλλαγές στο μηχανισμό του συστήματος και η ενσωμάτωση πραγματοποιείται εύκολα και άμεσα.
- Διαθεσιμότητα και δημοσίευση: Οι πληροφορίες σχετικά με τα διαθέσιμα web services δημοσιεύονται στο διαδίκτυο, με αποτέλεσμα η εύρεση και η χρησιμοποίησή τους να αποτελεί μία πολύ εύκολη και γρήγορη διαδικασία.
- Τροποποίηση: Υπάρχει η δυνατότητα προσαρμογής μίας υπάρχουσας εφαρμογής, ώστε να ικανοποιεί τις επιχειρησιακές συνθήκες και τις ανάγκες μίας συγκεκριμένης εταιρείας.
- Επεκτασιμότητα: Ένα έτοιμο web service είναι πολύ εύκολο να επεκταθεί και να προστεθούν σε αυτό νέες λειτουργίες.
- Μικρό κόστος δημιουργίας και χρήσης: Το κόστος ενσωμάτωσης ενός ήδη υλοποιημένου web service σε κάποιο ιστότοπο ή δικτυακή εφαρμογή είναι πάρα πολύ μικρό. Επίσης, η μετατροπή μίας διαδικασίας σε online υπηρεσία είναι πολύ εύκολη και κοστίζει ελάχιστα.
- Μείωση πολυπλοκότητας: Οι αιτούμενοι και οι παροχείς ασχολούνται μόνο με τις διεπαφές για τη μεταξύ τους διασύνδεση, χωρίς να απαιτείται να γνωρίζουν τον τρόπο λειτουργίας της υπηρεσίας της απέναντι πλευράς.

2.1.4. Μειονεκτήματα Web Services

Πέραν του μεγάλου αριθμού πλεονεκτημάτων που παρουσιάζουν, τα web services εμφανίζουν και μία σειρά από μειονεκτήματα, τα οποία οφείλουν να σημειωθούν:



- Τα πρότυπα των web services για μερικές λειτουργίες όπως οι συναλλαγές, είναι ακόμη σε πρώιμο στάδιο, αντίθετα με άλλα συστήματα όπως η rmi και corba που βρίσκονται σε πολύ προχωρημένο επίπεδο.
- Έχουν χαμηλότερη απόδοση σε σχέση με άλλα κατακευκτωμένα συστήματα. Αυτό οφείλεται στην χρήση της τεχνολογίας XML για την επικοινωνία, η οποία δεν στοχεύει στη σαφήνεια της κωδικοποίησης, ούτε στην απόδοση της λεκτικής ανάλυσης, αλλά στην ευκολία της επικοινωνίας.
- Η χρήση του πρωτοκόλλου HTTP επιτρέπει την παράκαμψη κάποιων συστημάτων ασφαλείας, τα οποία σκοπό έχουν τον έλεγχο της επικοινωνίας μεταξύ των προγραμμάτων και την αποφυγή κακόβουλης χρήσης.

2.2. Υπηρεσιοστρεφείς Αρχιτεκτονικές - Service Oriented Architectures (SOA)

Η Υπηρεσιοστρεφής Αρχιτεκτονική (Service Oriented Architecture – SOA) είναι ένας λογικός τρόπος να σχεδιαστεί ένα σύστημα λογισμικού για να παρέχει υπηρεσίες στον τελικό χρήστη ή σε άλλες υπηρεσίες μοιρασμένες σε ένα δίκτυο, μέσω διεπαφών έκδοσης και εξερεύνησης. Η SOA προσφέρει μία ευέλικτη αρχιτεκτονική, η οποία παρέχει ένα πλαίσιο ολοκλήρωσης, πάνω στο οποίο οι αρχιτέκτονες λογισμικού μπορούν να σχεδιάσουν εφαρμογές χρησιμοποιώντας μία συλλογή από επαναχρησιμοποιήσιμα λειτουργικά τμήματα με ευκρινείς διεπαφές.

Ο Υπηρεσιοστρεφής Προγραμματισμός είναι ένα αναδυόμενο φαινόμενο, το οποίο περιέχει υπηρεσίες, όπως ο σχεδιασμός εφαρμογών, ενώ αντιπροσωπεύει τη συγχώνευση πολλαπλών τεχνολογιών, όπως, κατακευκτωμένα συστήματα, πληροφοριακά συστήματα, διαφορετικές γλώσσες προγραμματισμού και XML τεχνολογίες. Οι λειτουργίες της SOA διαχωρίζονται σε ευδιάκριτες μονάδες ή υπηρεσίες ιστού, οι οποίες ενώνονται με χαλαρές συνδέσεις μεταξύ τους και είναι προσβάσιμες πάνω από το δίκτυο, έτσι ώστε να μπορούν να συνδυαστούν και να επαναχρησιμοποιηθούν για την ανάπτυξη διαφόρων επιχειρησιακών εφαρμογών.



2.2.1. Επίπεδα Λειτουργίας SOA

Η Υπηρεσιοστρεφής Αρχιτεκτονική αποτελείται από 6 επίπεδα λειτουργίας, το καθένα από τα οποία χρησιμοποιεί τη λειτουργικότητα του παρακάτω επιπέδου, προσθέτοντας κάποια επιπλέον λειτουργικότητα. Τα επίπεδα αυτά, όπως φαίνονται και στην Εικόνα 2, είναι τα ακόλουθα:

Επίπεδο 1 – Business Domain: Σε αυτό το επίπεδο, όλες οι επιχειρηματικές διαδικασίες στοχεύουν σε ένα συγκεκριμένο επιχειρηματικό πεδίο.

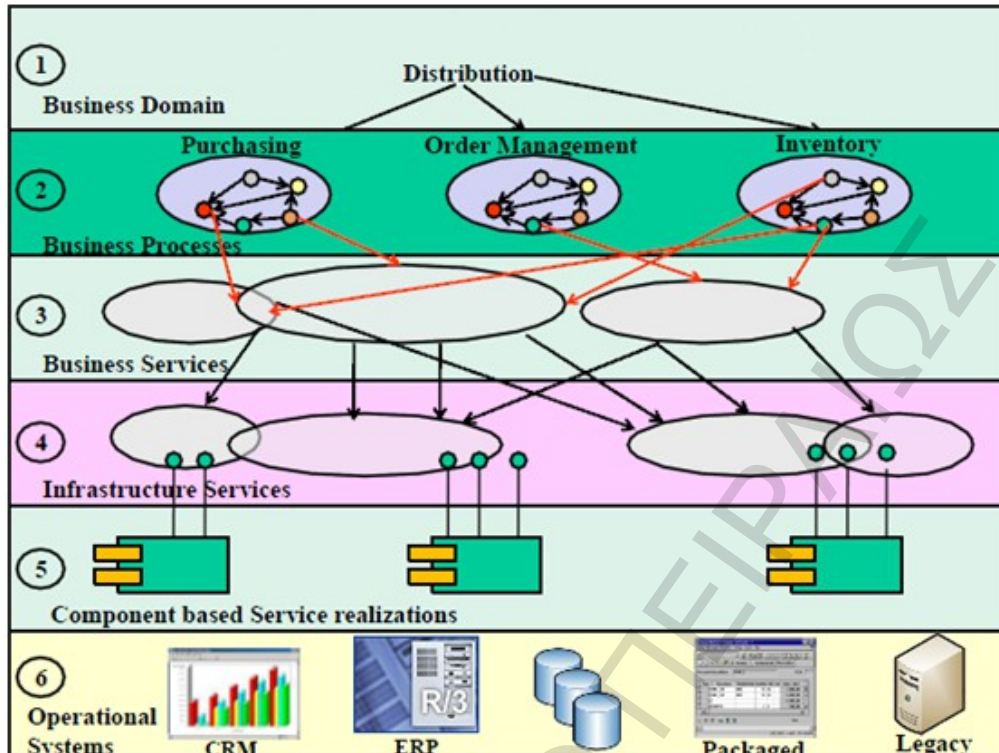
Επίπεδο 2 – Business Processes: Το επίπεδο αυτό έχει δημιουργηθεί από την υποδιαίρεση ενός επιχειρηματικού πεδίου σε μικρότερα τμήματα από βασικές επιχειρηματικές διαδικασίες.

Επίπεδο 3 – Business Services: Σε αυτό το επίπεδο περιέχονται οι υποδιαδικασίες στο χαμηλότερο βαθμό υποδιαίρεσης, με τις υπόλοιπες υποδιαδικασίες να αποτελούν μοναδικές (invisible-singular) επιχειρηματικές υπηρεσίες προς υλοποίηση.

Επίπεδο 4 – Infrastructure Services: Το επίπεδο αυτό αποτελείται από τις τεχνικές υπηρεσίες, οι οποίες παρέχουν το τεχνικό υπόβαθρο για τη δημιουργία, την παράδοση, τη συντήρηση και τον εφοδιασμό μοναδικών επιχειρηματικών υπηρεσιών.

Επίπεδο 5 – Component based Service realizations: Το επίπεδο αυτό αποτελεί την πραγματοποίηση του παράγοντα για την υλοποίηση υπηρεσιών από ήδη προϋπάρχουσες εφαρμογές και συστήματα.

Επίπεδο 6 – Operational Systems: Το επίπεδο αυτό χρησιμοποιείται από παράγοντες για την υλοποίηση επιχειρηματικών διαδικασιών και υπηρεσιών.



Εικόνα 2: Επίπεδα Λειτουργίας SOA

2.2.2. Πλεονεκτήματα SOA

Στο σημερινό απαιτητικό περιβάλλον, οι επιχειρήσεις αντιμετωπίζουν συνεχώς την πρόκληση να προσφέρουν περισσότερα με λιγότερους πόρους. Για το λόγο αυτό, ολοένα και περισσότεροι είναι εκείνοι που στρέφονται στις λύσεις που μπορούν να προσφέρουν οι Υπηρεσιοστρεφείς Αρχιτεκτονικές. Η SOA μπορεί να προσφέρει πολλά πλεονεκτήματα στις επιχειρήσεις που την υλοποιούν. Τα βασικότερα από αυτά, τα οποία την καθιστούν αποτελεσματική και ελκυστική, αναλύονται παρακάτω.

2.2.2.1. Επαναχρησιμοποίηση (Reusability)

Ένα από τα σημαντικότερα πλεονεκτήματα της SOA αποτελεί η δυνατότητα επαναχρησιμοποίησης των υπαρχουσών υπηρεσιών. Οι διάφοροι προγραμματιστές μπορούν να χρησιμοποιήσουν υπάρχοντα κώδικα υλοποίησης από άλλες



εφαρμογές, για την κάλυψη των δικών τους αναγκών. Η δυνατότητα αυτή οδηγεί σε σημαντική αποταμίευση χρόνου και κόστους κατά την ανάπτυξη μίας εφαρμογής.

Η δυνατότητα αυτή έχει και ένα βασικό μειονέκτημα. Ο επαναχρησιμοποιούμενος κώδικας έχει ξεχωριστά και συνήθως μοναδικά χαρακτηριστικά, μπορεί να έχει αναπτυχθεί σε διαφορετική γλώσσα προγραμματισμού και να χρησιμοποιεί διαφορετικά πρωτόκολλα και διεπαφές (interfaces). Όμως, στη SOA, το μόνο που απαιτείται για την επαναχρησιμοποίηση μίας υπηρεσίας είναι η γνώση της δημόσιας διεπαφής της.

2.2.2.2. Διαλειτουργικότητα (Interoperability)

Ένα άλλο σημαντικό πλεονέκτημα της SOA αποτελεί η διαλειτουργικότητα. Οι υπηρεσίες θα πρέπει να αναπτύσσονται με τέτοιο τρόπο, ώστε να αποφεύγονται οποιεσδήποτε τεχνολογικές εξαρτήσεις. Αυτό σημαίνει ότι οι πελάτες και οι πάροχοι των διαφόρων υπηρεσιών θα πρέπει να μπορούν να επικοινωνούν μεταξύ τους ανεξαρτήτως πλατφόρμας, συστήματος και γλώσσας προγραμματισμού.

Ο στόχος αυτός επιτυγχάνεται με την τυποποίηση του τρόπου επικοινωνίας μεταξύ τους. Η SOA περιλαμβάνει μία σειρά από πρωτόκολλα και τεχνολογίες, τα οποία είναι ευρέως διαδεδομένα και γενικώς αποδεκτά, και τα οποία την καθιστούν υλοποιούνται, ώστε να επιτευχθεί η επικοινωνία μεταξύ διαφορετικών πλατφορμών, συστημάτων ή γλωσσών προγραμματισμού.

2.2.2.3. Ευελιξία (Flexibility)

Οι υπηρεσίες που υλοποιούνται με βάση τη SOA παρέχουν μεγάλη ευελιξία, καθώς συνδέονται μεταξύ τους με χαλαρές συνδέσεις. Το γεγονός αυτό καθιστά δυνατή την τροποποίησή τους ανάλογα με τις επιχειρησιακές ανάγκες του οργανισμού με μικρό χρονικό και οικονομικό κόστος, χωρίς οι αλλαγές αυτές να επηρεάζουν τις υπόλοιπες υπηρεσίες του οργανισμού.



2.2.2.4. Αποδοτικότητα Κόστους (Cost Efficiency)

Η προσπάθεια ολοκλήρωσης των συστημάτων μίας επιχείρησης μπορεί να είναι αρκετά δαπανηρή και χρονοβόρα, καθώς απαιτείται εκτενής ανάλυση, προσεκτικός σχεδιασμός και προγραμματιστικός κόπος για την υλοποίηση των εργασιών. Με βάση τα πλεονεκτήματα που αναφέρθηκαν παραπάνω, είναι εμφανές ότι με τη χρήση της SOA, μία επιχείρηση μπορεί να καρπωθεί σημαντικά οικονομικά οφέλη.

2.3. XML (eXtensible Markup Language)

Η XML (eXtensible Markup Language) είναι μία γλώσσα σήμανσης, που περιέχει ένα σύνολο κανόνων για την ηλεκτρονική κωδικοποίηση κειμένων. Αναπτύχθηκε από μία Ομάδα Εργασίας υπό την επίβλεψη του διεθνούς οργανισμού World Wide Web Consortium (W3C) το 1996 και ορίζεται κυρίως στην προδιαγραφή XML 1.0.

Η XML αρχικά δημιουργήθηκε με σκοπό να ξεπεραστούν οι περιορισμοί της HTML (HyperText Markup Language), όχι όμως για να την αντικαταστήσει. Οι δύο γλώσσες έχουν δημιουργηθεί για διαφορετικούς σκοπούς, καθώς η HTML στοχεύει στον τρόπο με τον οποίο ένας web browser θα εμφανίσει τα δεδομένα, ενώ η XML επικεντρώνεται στον τρόπο μεταφοράς και αποθήκευσης των δεδομένων, δίνοντας έμφαση στο τι είδους δεδομένα είναι αυτά.

Η XML έχει σχεδιαστεί, δίνοντας έμφαση στην απλότητα και τη χρησιμότητα στο διαδίκτυο, ενώ μπορεί εύκολα να μεταφερθεί με τη χρήση πρωτοκόλλων διαδικτύου. Αποτελεί μία μορφοποίηση δεδομένων κειμένου, με ισχυρή υποστήριξη Unicode, αλλά χρησιμοποιείται ευρέως για την αναπαράσταση αυθαίρετων δομών δεδομένων, που μπορεί να προκύψουν στις διάφορες υπηρεσίες ιστού. Ουσιαστικά η XML παρέχει μία δενδροειδή δομή στα δεδομένα, χωρίς όμως να επιβάλλει τις λεπτομέρειες αυτής της δομής.

Σε αντίθεση με την HTML, η οποία περιλαμβάνει προκαθορισμένες ετικέτες (HTML elements), η XML παρέχει τη δυνατότητα στους χρήστες να καθορίζουν τις



ετικέτες. Η κάθε ετικέτα περικλείει κάποια πληροφορία, με σκοπό να ενημερώσει τον παραλήπτη του XML εγγράφου για αυτήν, καθώς και για τον τρόπο με τον οποίο θα πρέπει να τη διαχειριστεί.

Τα βασικά χαρακτηριστικά της γλώσσας είναι τα ακόλουθα:

- Η XML είναι δομημένη με σαφήνεια και ανεξάρτητη πλατφόρμας, με αποτέλεσμα να παρέχει τη δυνατότητα επικοινωνίας σε διαφορετικά υπολογιστικά συστήματα, τα οποία δεν θα μπορούσαν να επικοινωνήσουν μεταξύ τους με διαφορετικό τρόπο.
- Λόγω της φύσης της, είναι απρόσβλητη από τεχνολογικές αλλαγές.
- Υποστηρίζει ανεξαρτησία από τα δεδομένα και διαχωρίζει τα περιεχόμενα από τον τρόπο εμφάνισής τους και τον χειρισμό τους.
- Τα XML έγγραφα είναι αναγνώσιμα τόσο από ανθρώπους όσο και από μηχανές.
- Παρέχει τη δυνατότητα αναπαράστασης πολύπλοκων δομών δεδομένων, όπως λίστες, δέντρα, εγγραφές, ενώ μπορούν να περιγράψουν οποιονδήποτε τύπο δεδομένων.

Παρακάτω παρατίθεται ένα παράδειγμα XML εγγράφου:

```
<?xml version="1.0"?>
<collection>
  <song genre="rock">
    <title>Wind of Change</title>
    <artist>Scorpions</singer>
    <duration>4:34</duration>
  </song>
  <song genre="alternative">
    <title>Sadness</title>
    <artist>Enigma</singer>
    <duration>4:15</duration>
  </song>
  <!-- etc -->
</collection>
```



Το έγγραφο ξεκινά με μία προαιρετική δήλωση XML, η οποία το προσδιορίζει ως έγγραφο XML. Η παράμετρος *version*, καθορίζει την έκδοση της XML που χρησιμοποιείται στο έγγραφο. Παρακάτω περιγράφονται οι βασικές έννοιες που μπορούν να εμφανιστούν μέσα σε ένα XML έγγραφο:

- Στοιχεία (elements): Όμοια με αυτά της HTML, τα στοιχεία της XML αποτελούνται από ένα αρχικό tag (start tag), ένα τελικό tag (end tag) και δεδομένα ενδιάμεσα. Τα tags είναι case-sensitive και μπορούν να περιλαμβάνουν ένα ή περισσότερα χαρακτηριστικά (attributes). Επίσης, έχουν τη δυνατότητα να είναι κενά.
Π.χ. `<song genre="rock">Wind of change</song>`
- Χαρακτηριστικά (attributes): Αποτελούν επιπρόσθετη πληροφορία σχετικά με τα δεδομένα που χαρακτηρίζουν, χωρίς όμως να αποτελούν μέρος των δεδομένων. Για παράδειγμα, ο τύπος ενός εγγράφου δεν σχετίζεται με τα δεδομένα, όμως είναι σημαντικός για το πρόγραμμα που θα χρειαστεί να το επεξεργαστεί.
Π.χ. `<file type="doc">test.doc</file>`
- Σχόλια: Δεν αποτελούν μέρος των δεδομένων, αλλά χρησιμοποιούνται για την περιγραφή κάποιου τμήματος δεδομένων ή για την εισαγωγή σημειώσεων. Μπορούν να μπουν σε οποιοδήποτε σημείο του εγγράφου, ξεκινώντας με `<!--` και κλείνοντας με `-->`, όμοια με την HTML και μπορούν να περικλείουν όλους τους χαρακτήρες πλην των `&`, `<`, `>` και `"`. Για να προστεθούν αυτοί οι χαρακτήρες, χρησιμοποιούνται entity references (π.χ. `&`);
- Κείμενο: Κάθε στοιχείο περιλαμβάνει ακολουθίες χαρακτήρων Unicode, παρέχοντας τη δυνατότητα αποθήκευσης δεδομένων σε σχεδόν οποιαδήποτε γλώσσα.
- Έγγραφο: Αποτελεί μία μονάδα συσκευασίας XML και αποτελείται από ακριβώς ένα στοιχείο, το στοιχείο ρίζας (root element), το οποίο μπορεί να περιέχει και άλλα στοιχεία. Πολύ σημαντική είναι η σωστή δόμηση του XML εγγράφου (well-formed document), στην οποία βοηθά η σωστή σύνταξη. Το παράδειγμα που προηγήθηκε αποτελεί δείγμα ενός well-formed εγγράφου. Επιπλέον, το έγγραφο θα πρέπει να είναι έγκυρο (valid), δηλαδή να ικανοποιεί ένα σύνολο κανόνων που έχει ορίσει ο χρήστης, το οποίο περιλαμβάνεται σε κάποια προαιρετικά έγγραφα XML, όπως τα Document Type Definitions – DTDs και τα XML Schemas.



Ένα εξίσου σημαντικό στοιχείο της XML, αποτελούν τα XML Namespaces, τα οποία επιτρέπουν στους χρήστες να συνδυάζουν δύο ή περισσότερα XML κείμενα, χωρίς να συγκρούονται τυχόν ίδια ονόματα στοιχείων. Δημιουργούνται μοναδικά προθέματα, τα οποία μπαίνουν μπροστά από το όνομα του κάθε κοινού στοιχείου και ορίζουν τη σημασιολογία που θα πρέπει να ληφθεί υπόψη κατά την επεξεργασία του XML εγγράφου.

2.4. SOAP (Simple Object Access Protocol)

Το SOAP είναι ένα πολύ ελαφρύ πρωτόκολλο, το οποίο περιγράφεται ως Απλό Πρωτόκολλο Πρόσβασης Αντικειμένου και χρησιμοποιείται για την ανταλλαγή δομημένων πληροφοριών μεταξύ των web services. Βασίζεται στην XML για τη δομή των μηνυμάτων που ανταλλάσσονται, ενώ μπορεί να χρησιμοποιηθεί με διάφορα πρωτόκολλα μεταφοράς, όπως τα HTTP, HTTPS, SMTP και FTP.

Το SOAP παρέχει τη δυνατότητα προσθήκης κάποιων επιπλέον στοιχείων σε ένα μήνυμα, μέσω της χρήσης επικεφαλίδων. Στο SOAP μήνυμα που μεταφέρεται, προστίθενται κάποιες πληροφορίες, οι οποίες αφορούν τόσο στον καθορισμό και την περιγραφή του μηνύματος που μεταφέρεται, όσο και στον τρόπο με τον οποίο θα πρέπει να γίνει η επεξεργασία του μηνύματος από τους ενδιάμεσους κόμβους, μέχρι αυτό να φτάσει στον προορισμό του.

Τα βασικά χαρακτηριστικά του πρωτοκόλλου είναι τα ακόλουθα:

- *Απλότητα*: Η υλοποίηση του πρωτοκόλλου είναι απλή, με αποτέλεσμα να μειώνονται σημαντικά τόσο το κόστος υλοποίησης, όσο και η πολυπλοκότητα.
- *Ευελιξία*: Το SOAP είναι πολύ ευέλικτο, καθώς χρησιμοποιεί πρότυπα πρωτόκολλα μεταφοράς, όπως το HTTP και το SMTP, πράγμα που επιτρέπει τη χρήση του χωρίς να προκαλούνται θέματα ασφάλειας στην υποδομή και το δίκτυο μίας επιχείρησης.
- *Ανεξαρτησία*: Το πρωτόκολλο αυτό είναι ανεξάρτητο πλατφόρμας και γλώσσας προγραμματισμού, πράγμα που επιτρέπει με επιτυχία τη χρήση του σε εφαρμογές, οι οποίες θέλουν να επικοινωνήσουν μεταξύ τους, είναι όμως σχεδιασμένες σε διαφορετική πλατφόρμα ή γλώσσα προγραμματισμού.



- *Επεκτασιμότητα*: Η δομή του μηνύματος του πρωτοκόλλου, επιτρέπει την επέκτασή του από άλλα πρωτόκολλα και αναπτυσσόμενες τεχνολογίες, οι οποίες μπορούν να βελτιώσουν την αξιοπιστία και την ασφάλεια που προσφέρει το πρωτόκολλο κατά τη μεταφορά των μηνυμάτων.

Το μήνυμα SOAP έχει συγκεκριμένη δομή, η οποία αποτελείται από τα παρακάτω στοιχεία:

- *Φάκελος (Envelope)*: Αποτελεί υποχρεωτικό στοιχείο XML, το οποίο προσδιορίζει ότι το έγγραφο XML που μεταφέρεται, αποτελεί μήνυμα SOAP. Περιέχει τα κύρια δεδομένα του μηνύματος, συμπεριλαμβανομένων των πληροφοριών αποστολέα και παραλήπτη, καθώς και λεπτομέρειες για το ίδιο το μήνυμα.
- *Κεφαλίδα (Header)*: Αποτελεί προαιρετικό στοιχείο και περιλαμβάνει διάφορες επιπλέον πληροφορίες. Μπορεί να περιέχει πληροφορίες αυθεντικοποίησης, όπως όνομα χρήστη και συνθηματικό, αλλά και πληροφορίες συναλλαγής.
- *Σώμα (Body)*: Είναι υποχρεωτικό στοιχείο και περιλαμβάνει το πραγματικό μήνυμα προς μεταφορά.
- *Σφάλματα (Fault)*: Αποτελεί προαιρετικό στοιχείο, το οποίο περιλαμβάνει πληροφορίες σχετικά με την ορθότητα του μηνύματος και τα λάθη που προκλήθηκαν κατά την επεξεργασία του. Το στοιχείο αυτό περιλαμβάνεται στο σώμα του μηνύματος SOAP.

Παρακάτω, παρουσιάζεται ένα παράδειγμα της δομής ενός SOAP μηνύματος:

Φάκελος SOAP

```
<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope"
```

Κεφαλίδα SOAP

```
<soap:Header>
```

```
...
```

```
...
```

```
</soap:Header>
```

Σώμα SOAP



```
<soap:Body>  
  Σφάλματα SOAP  
  <soap:Fault>  
  ...  
  ...  
  </soap:Fault>  
</soap: Body >  
</soap:Envelope>
```

Ένα μήνυμα SOAP, μπορεί να περάσει από αρκετούς κόμβους μέχρι να φτάσει στον προορισμό του. Οι κόμβοι αποτελούν σημεία στο δίκτυο, τα οποία αναπαριστούν τη λογική που είναι υπεύθυνη για τη μεταβίβαση, λήψη, επεξεργασία ή/και αναμετάδοση ενός μηνύματος.

Για να επιτευχθεί η σωστή μεταφορά και επεξεργασία ενός SOAP μηνύματος, το πρωτόκολλο περιλαμβάνει ένα σύνολο από κανόνες, οι οποίοι ενσωματώνονται στην κεφαλίδα του μηνύματος. Οι κανόνες αυτοί αφορούν στον τρόπο με τον οποίο θα πρέπει να γίνεται η κωδικοποίηση κάθε διαφορετικού τύπου δεδομένων που αποστέλλεται σε μία εφαρμογή. Η ενσωμάτωση αυτών των κανόνων στο μήνυμα γίνεται με τον χαρακτηριστικό *encodingStyle*, το οποίο παραπέμπει σε ένα URL με τους κανόνες κωδικοποίησης. Επιπλέον, υπάρχει το χαρακτηριστικό *role*, το οποίο αναφέρει τον κόμβο, ο οποίος έχει τη δυνατότητα να επεξεργαστεί κάποιο τμήμα του μηνύματος.

Ένα εξίσου σημαντικό χαρακτηριστικό είναι και το *mustUnderstand*, το οποίο ενημερώνει εάν ο κόμβος που λαμβάνει το μήνυμα θα πρέπει να το καταλαβαίνει. Το χαρακτηριστικό αυτό είναι προαιρετικό, ενώ σε περίπτωση που κάποιο σημείο του μηνύματος δεν είναι κατανοητό, τότε θα πρέπει να συνταχθεί ένα μήνυμα λάθους. Τέλος, σημαντικό είναι και το χαρακτηριστικό *actor*, το οποίο αναφέρει τον εξυπηρετητή, ο οποίος μπορεί να διαχειριστεί την πληροφορία. Ο εξυπηρετητής που ορίζεται από το συγκεκριμένο χαρακτηριστικό, θα πρέπει να το αφαιρέσει, ενώ μπορεί να προσθέσει νέο πεδίο στον Header πριν το προωθήσει σε άλλο



εξυπηρετητή. Σε περίπτωση που η κεφαλίδα δεν περιλαμβάνει αυτό το χαρακτηριστικό, τότε ως actor θεωρείται ο παραλήπτης του μηνύματος.

2.4.1. Remote Procedure Calls (RPCs)

Τα μηνύματα SOAP έχουν τη δυνατότητα να υλοποιούν κλήσεις RPC (Remote Procedure Calls). Συγκεκριμένα, για την κλήση μίας μεθόδου ενός απομακρυσμένου αντικειμένου ενός web service, στέλνεται ένα μήνυμα SOAP (request), το οποίο περιλαμβάνει στο σώμα του το όνομα της μεθόδου και τις παραμέτρους που αυτή πρέπει να δεχθεί. Μετά την εκτέλεση της μεθόδου, τα αποτελέσματα επιστρέφονται στο πρόγραμμα το οποίο την κάλεσε, μέσω ενός άλλου μηνύματος SOAP (response). Η επικοινωνία πραγματοποιείται μεταξύ του πελάτη (SOAP Client) και του διακομιστή (SOAP Server), οι οποίοι είναι τμήματα του προγράμματος που αναλαμβάνουν τη μετατροπή από τη μορφή που είναι κατανοητή στην τοπική εφαρμογή αντίστοιχα σε μορφή SOAP XML φακέλου και το αντίστροφο. Με αυτό τον τρόπο λειτουργούν ως μεσάζοντες (proxy), μεταξύ των δύο επικοινωνούντων μηχανών.

2.4.2. SOAP Bindings

Όπως αναφέρθηκε και παραπάνω, τα μηνύματα SOAP μπορούν να μεταδίδονται μέσω διαφορετικών δικτυακών πρωτοκόλλων, όπως το HTTP, το SMTP κ.α., όμως το πιο διαδεδομένο είναι το HTTP. Αυτό συμβαίνει γιατί το πρωτόκολλο αυτό χρησιμοποιείται κατά κόρον στο διαδίκτυο, καθώς χρησιμοποιείται από τους browsers για την πλοήγηση των χρηστών. Με τη χρήση του συγκεκριμένου πρωτοκόλλου, τα δεδομένα μεταφέρονται εύκολα, με τον τρόπο που μεταφέρονται και οι ιστοσελίδες στο διαδίκτυο, χωρίς να εμποδίζονται από τυχόν τείχη ασφαλείας (firewalls) που μπορεί να έχουν οι υπολογιστές. Με τη χρήση του πρωτοκόλλου HTTP, το μήνυμα SOAP αποστέλλεται σαν μέρος του μηνύματος αίτησης ή απόκρισης HTTP (HTTP request ή response).

Η προδιαγραφή που ορίζει τον τρόπο με τον οποίο τα μηνύματα μεταφέρονται από έναν κόμβο σε έναν άλλο ονομάζεται σύνδεση SOAP (SOAP Binding). Μία προδιαγραφή περιγράφει τον τρόπο με τον οποίο ένα μήνυμα θα μεταφερθεί από έναν κόμβο σε έναν άλλο και δεν αναφέρεται σε ολόκληρη τη διαδρομή. Ωστόσο υπάρχει η δυνατότητα χρήσης της ίδιας προδιαγραφής σε ολόκληρη τη διαδρομή του μηνύματος.

2.5. REST (Representational State Transfer)

Το πρωτόκολλο REST αποτελεί ένα σχετικά νέο στυλ αρχιτεκτονικής, για τη δημιουργία διαδικτυακών εφαρμογών, το οποίο δημιουργήθηκε με σκοπό την απλούστευση της επικοινωνίας μεταξύ υπηρεσιών και πελατών υπηρεσιών. Το REST δημιουργήθηκε με σκοπό να αντικαταστήσει τους πολύπλοκους μηχανισμούς που χρησιμοποιούνταν, όπως CORBA, RPC, και SOAP, χρησιμοποιώντας το απλό HTTP πρωτόκολλο για την επικοινωνία.

Το πρωτόκολλο REST, αποτελεί μία «ελαφριά» εναλλακτική για την δημιουργία διαδικτυακών εφαρμογών, χάρις στην απλότητά του. Οι εφαρμογές που χρησιμοποιούν το πρωτόκολλο αυτό, εκμεταλλεύονται τις αιτήσεις (requests) του HTTP πρωτοκόλλου για όλες τους τις λειτουργίες, όπως δημιουργία ή ενημέρωση δεδομένων, ανάκτηση δεδομένων ή διαγραφή δεδομένων (Create/Read/Update/Delete). Παρά την απλότητά του όμως, το πρωτόκολλο έχει πλήρεις δυνατότητες και δεν υπάρχει κάτι, το οποίο να μπορεί να υλοποιηθεί με την αρχιτεκτονική SOAP, αλλά να μην μπορεί να υλοποιηθεί με την αρχιτεκτονική REST.

Τα βασικά χαρακτηριστικά του πρωτοκόλλου είναι τα ακόλουθα:

- *Απλότητα*: Η υλοποίηση του πρωτοκόλλου είναι απλή, με αποτέλεσμα να μειώνονται σημαντικά τόσο το κόστος υλοποίησης, όσο και η πολυπλοκότητα.
- *Ανεξαρτησία*: Το πρωτόκολλο αυτό είναι ανεξάρτητο πλατφόρμας και γλώσσας προγραμματισμού, πράγμα που επιτρέπει με επιτυχία τη χρήση του σε εφαρμογές, οι οποίες θέλουν να επικοινωνήσουν μεταξύ τους, είναι όμως σχεδιασμένες σε διαφορετική πλατφόρμα ή γλώσσα προγραμματισμού.



- *Βασισμένο σε υπάρχοντα standards:* Το REST πρωτόκολλο χρησιμοποιεί το HTTP πρωτόκολλο για την επικοινωνία και βασίζεται στα standards αυτού.
- *Δεν επηρεάζεται από Τοίχους Ασφαλείας:* Μπορεί εύκολα να χρησιμοποιηθεί με την ύπαρξη firewalls, χωρίς να υπάρχει πρόβλημα στην επικοινωνία.
- *Δεν περιλαμβάνει ενσωματωμένα χαρακτηριστικά ασφαλείας:* Επιπλέον ασφάλεια μπορεί να προστεθεί με τη χρήση υπαρχόντων λειτουργιών του HTTP πρωτοκόλλου, όπου αυτό απαιτείται. Για παράδειγμα, συχνά χρησιμοποιούνται username/password tokens, για ταυτοποίηση. Επίσης, σε περίπτωση που απαιτείται κρυπτογράφηση της επικοινωνίας, μπορεί να χρησιμοποιηθεί το HTTPS πρωτόκολλο.

Με σκοπό την καλύτερη κατανόηση της απλότητας του REST πρωτοκόλλου, σε σχέση με τη χρήση του SOAP πρωτοκόλλου, παρακάτω, παρατίθεται η σύγκριση κλήσης της ίδιας υπηρεσίας ιστού, με τη χρήση του SOAP και του REST πρωτοκόλλου αντίστοιχα.

SOAP

```
<?xml version="1.0"?>
<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope"
soap:encodingStyle=" http://www.w3.org/2003/05/soap-encoding">
  <soap:Body ab="http://www.mydomain.com/addressbook">
    <ab:GetUserDetails>
      <ab:UserID>111</ab:UserID>
    </ab:GetUserDetails>
  </soap:Body>
</soap:Envelope>
```

REST

<http://www.mydomain.com/addressbook/UserDetails/111>

Χρησιμοποιώντας το πρωτόκολλο SOAP, παρατηρούμε ότι απαιτείται η δημιουργία ενός σωστά δομημένου SOAP Envelope με τις απαραίτητες



πληροφορίες, ο οποίος αποστέλλεται στον server. Η απάντηση θα είναι ένα XML αρχείο, το οποίο θα αποσταλεί πίσω, μέσα σε ένα SOAP Response Envelope.

Αντίθετα, μέσω του REST πρωτοκόλλου, το μόνο που απαιτείται είναι η κλήση ενός URL με συγκεκριμένη δομή από τον browser. Το URL αυτό, αποστέλλεται στον server, μέσω μίας αίτησης GET, ενώ η απάντηση επιστρέφεται σε απλή μορφή, χωρίς να περιλαμβάνεται σε κάποιο αντίστοιχο φάκελο, όπου τα δεδομένα είναι έτοιμα προς επεξεργασία. Σε περίπτωση που απαιτείται η αποστολή περισσότερων από μία παραμέτρους, αυτό είναι εφικτό μέσω της προσθήκης παραμέτρων στο URL. Σε αυτή την περίπτωση θα πρέπει να χρησιμοποιηθεί η αίτηση POST του HTTP πρωτοκόλλου, όπου θα πρέπει να συμπεριληφθούν οι παράμετροι στο σώμα της αίτησης POST. Στο παρακάτω παράδειγμα φαίνεται η χρήση 2 παραμέτρων στο URL:

`http://www.mydomain.com/addressbook/UserDetails?
firstName=Alexandra&lastName=P`

Το πρωτόκολλο REST μπορεί εύκολα να χρησιμοποιήσει XML για την αποστολή αιτήσεων στον server, σε περίπτωση που απαιτείται η χρήση δομημένων δεδομένων. Ωστόσο, αυτό δε συνηθίζεται, καθώς η βασική δομή του πρωτοκόλλου προσφέρει όλες τις απαραίτητες δυνατότητες, χωρίς την επιπλέον πληροφορία που απαιτείται για τη σύνταξη του XML αρχείου. Οι απαντήσεις όμως του server, συνήθως αποστέλλονται σε μορφή XML, χωρίς αυτό να είναι δεσμευτικό, σε αντίθεση με το SOAP πρωτόκολλο. Άλλες μορφές απαντήσεων που λαμβάνονται από τον server είναι το CSV (Comma-Separated Values), ή JSON (JavaScript Object Notation). Η κάθε μία από τις παραπάνω μορφές έχει τα δικά της πλεονεκτήματα και μειονεκτήματα και μπορεί να εφαρμοστεί ανάλογα με τις ανάγκες της εκάστοτε υπηρεσίας.



2.6. WSDL (Web Services Description Languages)

Σε προηγούμενο κεφάλαιο περιγράφηκε το πρωτόκολλο SOAP, το οποίο βοηθά στην επίτευξη της επικοινωνίας μεταξύ ενός πελάτη υπηρεσίας και της υπηρεσίας που τον ενδιαφέρει. Το πρωτόκολλο αυτό καθορίζει κάποιους κανόνες που αφορούν στην επικοινωνία και προσφέρει μία συγκεκριμένη δομή για τα μηνύματα που ανταλλάσσονται. Όμως, για να μπορέσει ένας πελάτης να επικοινωνήσει με μία υπηρεσία θα πρέπει προηγουμένως να γνωρίζει κάποιες λεπτομέρειες για αυτήν, όπως τον τρόπο κλήσης της, τις μεθόδους που χρησιμοποιεί, τις παραμέτρους που απαιτεί και επιστρέφει, αλλά και πληροφορίες επικοινωνίας, όπως το πρωτόκολλο επικοινωνίας που θα πρέπει να χρησιμοποιήσει, το ακριβές μήνυμα που θα πρέπει να στείλει κ.τ.λ..

Σε αυτό το πρόβλημα, έρχεται να δώσει λύση η Γλώσσα Περιγραφής Υπηρεσιών Παγκόσμιου Ιστού (Web Services Description Language - WSDL). Η WSDL είναι μία γλώσσα βασισμένη στην XML, η οποία χρησιμοποιείται για να περιγράψει ένα Web Service και τους τρόπους επικοινωνίας με αυτό.

Το W3C ορίζει την WSDL ως: *«ένα σχήμα XML που περιγράφει τις δικτυακές υπηρεσίες σαν ένα σύνολο από τελικά σημεία που χειρίζονται μηνύματα τα οποία περιέχουν είτε πληροφορία προσανατολισμένη σε έγγραφα(document-oriented) είτε σε διαδικασίες (procedure-oriented). Οι λειτουργίες και τα μηνύματα περιγράφονται περιληπτικά και τότε δένονται σε ένα συγκεκριμένο πρωτόκολλο δικτύων και μορφή μηνυμάτων για να καθορίσουν ένα τελικό σημείο (endpoint). Πολλά σχετικά τελικά σημεία συνδυάζονται σε υπηρεσίες (services).»*.

Κάθε υπηρεσία παγκόσμιου ιστού έχει κάποιες δημόσιες διεπαφές (public interfaces), μέσω των οποίων επικοινωνούν μαζί της οι πελάτες υπηρεσίας. Η WSDL χρησιμοποιείται για να περιγράψει τις διεπαφές αυτές, περιλαμβάνοντας πληροφορίες όπως τις διαθέσιμες δημόσιες δραστηριότητες της υπηρεσίας, τα πρωτόκολλα XML μηνυμάτων που υποστηρίζουν τα web service, το είδος των δεδομένων, τη δέσμευση για το συγκεκριμένο πρωτόκολλο μεταφοράς που πρόκειται να χρησιμοποιηθεί και τη διεύθυνση για τον εντοπισμό της υπηρεσίας.



Η WSDL, όπως και η XML στην οποία στηρίζεται, είναι ανεξάρτητη πλατφόρμας και γλώσσας προγραμματισμού. Αποτελεί ένα είδος συμβολαίου μεταξύ του πελάτη υπηρεσίας και του παρόχου υπηρεσίας, καθώς περιγράφει όλες τις λεπτομέρειες που διέπουν τη μεταξύ τους επικοινωνία. Συγκεκριμένα, περιγράφει τι ακριβώς κάνει η υπηρεσία, τις μεθόδους που χρησιμοποιεί, τις παραμέτρους που απαιτεί και τα αποτελέσματα που επιστρέφει. Επίσης, περιλαμβάνει τις προδιαγραφές πρόσβασης στην υπηρεσία, όπως το πρωτόκολλο επικοινωνίας και το είδος της κωδικοποίησης που θα πρέπει να χρησιμοποιηθούν. Τέλος, αναφέρει που βρίσκεται η συγκεκριμένη υπηρεσία. Αυτό περιγράφεται από το όνομα και τη διεύθυνσή της, τα οποία είναι συνήθως ένα URL. Η περιγραφή αυτή μίας υπηρεσίας, καταχωρείται στο Μητρώο Υπηρεσιών (Registry), από όπου μπορεί να αναζητηθεί και να ανακτηθεί από τους Πελάτες Υπηρεσιών.

Οι παραπάνω πληροφορίες, ομαδοποιούνται σε ένα WSDL έγγραφο, σε δύο μέρη. Το πρώτο μέρος είναι το αφηρημένο (abstract), το οποίο περιγράφει τι κάνει η υπηρεσία, αναφέροντας τα εισερχόμενα και εξερχόμενα μηνύματα που ανταλλάσσονται, τις απαραίτητες παραμέτρους κ.τ.λ.. Το δεύτερο μέρος είναι το συγκεκριμένο (concrete), το οποίο περιλαμβάνει τις πληροφορίες σχετικά με το που και πως παρέχεται η υπηρεσία.

Κάποιες από τις πληροφορίες αυτές, περιγράφονται μέσα σε ένα WSDL έγγραφο, χρησιμοποιώντας συγκεκριμένα στοιχεία, όπως αυτά έχουν οριστεί από την προδιαγραφή της γλώσσας. Τα βασικότερα από αυτά είναι τα ακόλουθα:

- <types>: Αναφέρονται οι τύποι δεδομένων, οι οποίοι θα χρησιμοποιηθούν σε άλλα στοιχεία του εγγράφου αλλά και στα μηνύματα που ανταλλάσσονται. Ανήκει στο αφηρημένο μέρος.
- <message>: Περιγράφεται η μορφή του μηνύματος και το σύνολο των χρησιμοποιούμενων παραμέτρων που σχετίζονται με τις μεθόδους της υπηρεσίας. Κάθε στοιχείο message μέσα σε ένα WSDL έγγραφο θα πρέπει να έχει ένα μοναδικό όνομα. Ανήκει στο αφηρημένο μέρος.
- <portType>: Περιγράφει τη διασύνδεση της υπηρεσίας, καθορίζοντας ένα σύνολο λειτουργιών που αυτή υποστηρίζει. Οι λειτουργίες αυτές υποστηρίζονται από ένα ή περισσότερα τελικά σημεία. Ανήκει στο αφηρημένο μέρος.



- <port>: Αποτελεί ένα μοναδικό τελικό σημείο, το οποίο ορίζεται ως συνδυασμός μιας σύνδεσης και μιας διεύθυνσης δικτύου. Εμφανίζεται πάντα ως στοιχείο παιδί του στοιχείου service και ανήκει στο συγκεκριμένο μέρος.
- <operation>: Καθορισμός μιας λειτουργίας που υποστηρίζεται από μία υπηρεσία. Υπάρχουν 4 διαφορετικά είδη λειτουργιών:
 - i. Μίας κατεύθυνσης, όπου το τερματικό λαμβάνει ένα μήνυμα.
 - ii. Αίτησης - απάντησης, όπου το τερματικό λαμβάνει μία αίτηση και αποστέλλει μία αντίστοιχη απάντηση.
 - iii. Ζήτησης – απάντησης, όπου το τερματικό στέλνει ένα μήνυμα και λαμβάνει μία αντίστοιχη απάντηση.
 - iv. Ειδοποίησης, όπου το τερματικό στέλνει ένα ενημερωτικό μήνυμα.

Ανήκει στο αφηρημένο μέρος.

- <binding>: Ορίζει το πρωτόκολλο επικοινωνίας και τη μορφή των δεδομένων που απαιτούνται για τη διασύνδεση με ένα συγκεκριμένο τύπο τελικών σημείων. Το όνομα του binding θα πρέπει να είναι μοναδικό, καθώς μπορεί να υπάρχουν πολλά τέτοια στοιχεία σε ένα WSDL έγγραφο. Ανήκει στο συγκεκριμένο μέρος.
- <service>: Καθορίζει τη διεύθυνση επικοινωνίας για την υπηρεσία. Ανήκει στο συγκεκριμένο μέρος.

Η βασική δομή ενός WSDL εγγράφου, συμπεριλαμβανομένων των στοιχείων που περιγράφηκαν, φαίνεται παρακάτω:

```
<wsdl:definitions xmlns:wsdl=http://schemas.xmlsoap.org/wsdl
targetNamespace="your namespace"
xmlns:tns="your namespace"
xmlns:soapbind=http://schemas.xmlsoap.org/wsdl/soap">
  <wsdl:types>
    <xs:schema targetNamespace="your namespace"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
      </schema>
    </wsdl:types>
    <wsdl:message name="some operation input"> ... </wsdl:message>
    <wsdl:message name="some operation output"> ... </wsdl:message>
    <wsdl:portType name="your type name"> ... </wsdl:portType >
```



```
<wsdl:binding name=" your binding name" type="tns: the above stated
portType name" >
    <!-- define style and transport in general and use per operation -->
</wsdl:binding>
<wsdl:service name=" your binding name" type="tns: the above stated
portType name" >
    <!--define a port using the above binding and a URL -->
</wsdl:service>
</wsdl:definitions>
```

2.7. UDDI (Universal Description, Discovery and Integration)

Το UDDI, ως τεχνική προδιαγραφή, παρέχει μία μέθοδο για δημοσίευση και εύρεση των περιγραφών μίας υπηρεσίας. Ουσιαστικά αποτελεί μία κεντρική υπηρεσία καταλόγου, όπου καταχωρούνται οι υπηρεσίες παγκόσμιου ιστού . Στις αρχές του 2000 και καθώς η ιδέα των web services είχε αρχίσει να κερδίζει έδαφος, έγινε σαφές ότι μία τέτοια κεντρική υπηρεσία ήταν απαραίτητη, ώστε να μπορούν να αναπτυχθούν και να αξιοποιηθούν αποτελεσματικά τα διάφορα web services. Ένα τέτοιο πρότυπο καταλόγου θα έπρεπε να υποστηριχθεί τουλάχιστον από τις περισσότερες εταιρείες παροχής λογισμικού και να καταφέρει να υιοθετηθεί από τις διάφορες βιομηχανίες. Έτσι, η πρωτοβουλία του UDDI, η οποία ήταν αποτέλεσμα πολλών μηνών συνεργασίας μεταξύ των εταιρειών IBM, Microsoft και Ariba, ανακοινώθηκε επίσημα στις 6 Σεπτεμβρίου του 2000. Η υποστήριξη για το UDDI επεκτάθηκε πέραν των τριών αρχικών εταιρειών, και σήμερα αριθμεί πάνω από 310 εταιρείες.

Σκοπός του UDDI είναι να διευκολύνει την ανακάλυψη υπηρεσιών κατά το χρόνο σχεδίασης και εκτέλεσής τους. Αποτελεί όμως κάτι παραπάνω από έναν κατάλογο επιχειρήσεων και υπηρεσιών, καθώς ορίζει ένα σύνολο από δομές δεδομένων και προδιαγραφές API, τόσο για την αναζήτηση όσο και για την καταχώρηση εταιρειών και υπηρεσιών.



Η δομή των δεδομένων, τα οποία αποθηκεύονται στον κατάλογο UDDI είναι σε μορφή XML. Τα δεδομένα που συλλέγονται χωρίζονται στις ακόλουθες τρεις κατηγορίες:

- *Λευκές Σελίδες (White Pages)*: Περιλαμβάνουν γενικές πληροφορίες, όπως το όνομα, την περιγραφή, τα τηλέφωνα και τη διεύθυνση μίας εταιρείας, η οποία προσφέρει κάποια/ες υπηρεσία/ες.
- *Κίτρινες Σελίδες (Yellow Pages)*: Περιλαμβάνουν πληροφορίες, οι οποίες περιγράφουν μία υπηρεσία, χρησιμοποιώντας διαφορετικές κατηγοριοποιήσεις. Με αυτό τον τρόπο, δίνεται η δυνατότητα στους χρήστες να ανακαλύψουν μία υπηρεσία, αναζητώντας με βάση την κατηγορία που τους ενδιαφέρει.
- *Πράσινες Σελίδες (Green Pages)*: Περιλαμβάνουν λεπτομερείς τεχνικές πληροφορίες για μία υπηρεσία ιστού, επιτρέποντας σε ένα χρήστη, να ανακτήσει όλες τις απαραίτητες πληροφορίες, ώστε να υλοποιήσει μία εφαρμογή, η οποία να μπορεί να τη χρησιμοποιήσει.

2.7.1. Οι Δομές Δεδομένων του UDDI

Η κάθε καταχώρηση σε ένα μητρώο UDDI, είναι ένα XML έγγραφο, το οποίο αποτελείται από διαφορετικούς τύπους δεδομένων. Τα δεδομένα αυτά μπορούν να αφορούν στις επιχειρήσεις που προσφέρουν τις υπηρεσίες (*businessEntity*), στις ίδιες τις υπηρεσίες (*businessService*), στις συνδέσεις με υπηρεσίες (*bindingTemplate*) και στις προδιαγραφές των υπηρεσιών (*tModels*).

1. *Τύπος Δεδομένων businessEntity*: Περιλαμβάνει την περιγραφή και τις απαραίτητες πληροφορίες επικοινωνίας με κάθε οντότητα (οργανισμό, επιχείρηση κ.τ.λ.) η οποία παρέχει κάποια υπηρεσία παγκόσμιου ιστού.
2. *Τύπος Δεδομένων businessService*: Περιλαμβάνει πληροφορίες σχετικές με μία υπηρεσία παγκόσμιου ιστού ή με μία ομάδα συσχετιζόμενων μεταξύ τους υπηρεσιών, οι οποίες παρέχονται από το ίδιο *businessEntity*. Ένα *businessEntity* μπορεί να περιλαμβάνει πολλά *businessServices*, όμως ένα *businessService* μπορεί να ανήκει μόνο σε ένα *businessEntity*.



Σε κάθε `businessEntity` και `businessService`, ανατίθενται μοναδικά αναγνωριστικά, τα Universally Unique Identifiers (UUIDs), μέσω των οποίων αναγνωρίζονται μοναδικά σε όλα τα μητρώα UDDI.

3. *Τύπος Δεδομένων `bindingTemplate`*: Περιλαμβάνει όλες τις τεχνικές πληροφορίες που απαιτούνται για την εκτέλεση μίας υπηρεσίας. Ένα `businessService` μπορεί να περιλαμβάνει πολλά `bindingTemplates`, όμως ένα `bindingTemplate` μπορεί να ανήκει μόνο σε ένα `businessService`.
4. *Τύπος Δεδομένων `tModel`*: Χρησιμοποιείται για την αποθήκευση επιπλέον πληροφοριών, σχετικά με την παρεχόμενη υπηρεσία παγκόσμιου ιστού, όπως το `interface` και το πρωτόκολλο που χρησιμοποιούνται.

2.7.2. Διεπαφές Προγραμματισμού Εφαρμογών (Application Programming Interface - API)

Εκτός από τον ορισμό των δομών δεδομένων που αποθηκεύονται στον κατάλογο του UDDI, παρέχεται επίσης και ένας αριθμός από Διεπαφές Προγραμματισμού Εφαρμογών, ώστε να παρέχεται πρόσβαση σε ένα σύστημα UDDI. Οι διεπαφές αυτές είναι οι ακόλουθες:

- UDDI Inquiry: Αφορά στον εντοπισμό και την απόκτηση λεπτομερειών καταχώρησης σε ένα UDDI registry.
- UDDI Publication: Αφορά στη δημοσίευση και τροποποίηση πληροφοριών σε ένα UDDI registry.
- UDDI Security: Αφορά στον έλεγχο πρόσβασης σε ένα UDDI registry.
- UDDI Subscription: Δίνει τη δυνατότητα σε clients να εγγραφούν σε αλλαγές στην πληροφορία στο UDDI registry.
- UDDI Replication: Αφορά στη δημιουργία αντιγράφων πληροφοριών διαμέσου των κόμβων σε ένα UDDI registry.
- UDDI Custody and Ownership transfer: Αφορά στην αλλαγή του ιδιοκτήτη της πληροφορίας και της επιμέλειας από έναν κόμβο σε ένα άλλο, μέσα σε ένα UDDI registry.

Επιπλέον, παρέχονται και ένα σύνολο από APIs, από τους clients ενός UDDI συστήματος.



- UDDI Subscription Listener: Παρέχεται η δυνατότητα στον κόμβο UDDI να ενημερώνει τον πελάτη, όταν πραγματοποιούνται αλλαγές στον κατάλογο, για τις οποίες ο πελάτης έχει ζητήσει να ενημερώνεται μέσω του UDDI Subscription.
- UDDI Value Set: Χρησιμοποιείται για την κατηγοριοποίηση και το συσχετισμό των εννοιών που χρησιμοποιούνται.

2.7 Σύνοψη

Στα προηγούμενα κεφάλαια παρουσιάστηκαν αναλυτικά οι τεχνολογίες που απαιτούνται για την υλοποίηση ενός Web Service. Είδαμε πως η μία τεχνολογία βασίζεται στην άλλη για την υλοποίηση και την ορθή λειτουργία ενός Web Service, ενώ όλες βασίζονται στη γλώσσα XML, για την καταχώρηση και παρουσίαση των πληροφοριών που παρέχουν.

Συγκεκριμένα, η προδιαγραφή UDDI χρησιμοποιεί τη γλώσσα WSDL για να αποθηκεύει πληροφορίες σχετικές με τις καταχωρημένες εταιρείες και υπηρεσίες. Επιπλέον, η αναζήτηση και εγγραφή μίας πληροφορίας σε μία UDDI Registry, πραγματοποιείται μέσω μηνυμάτων SOAP, τα οποία χρησιμοποιούνται για την κλήση του αντίστοιχου API. Τα SOAP μηνύματα δεν είναι τίποτε παραπάνω από έναν απλό φάκελο, βασισμένο στην γλώσσα XML, ώστε να μεταφερθούν οι απαραίτητες πληροφορίες, καθώς και οι αντίστοιχοι κανόνες που διέπουν την επικοινωνία αυτή. Τα SOAP μηνύματα χρησιμοποιούνται επίσης και για την επικοινωνία μεταξύ του πελάτη και του παρόχου υπηρεσίας.



3. Περιγραφή Προβλήματος

Στο κεφάλαιο αυτό πραγματοποιείται μία εκτενέστερη περιγραφή του προβλήματος που καλείται να επιλύσει η παρούσα εργασία. Παρουσιάζεται η περιγραφή και η σύσταση της εταιρείας, στην οποία θα εφαρμοστεί η συγκεκριμένη εφαρμογή, καθώς και οι συνθήκες λειτουργίας της. Επίσης, παρουσιάζεται η περιγραφή και οι απαιτήσεις της εφαρμογής, καθώς και οι περιορισμοί και οι έλεγχοι που πρέπει να πραγματοποιούνται για τη σωστή λειτουργία και χρήση της.

3.1. Περιγραφή και Σύσταση Εταιρείας

Η εταιρεία «ΑΒΓ Πληροφοριακά Συστήματα Ε.Π.Ε.» αποτελεί μία μικρή εταιρεία πληροφοριακών συστημάτων με έδρα την Αθήνα. Τα άτομα που την απαρτίζουν είναι τα ακόλουθα:

- Ο κ. Δημήτρης Δ.: Είναι ο Διευθυντής της εταιρείας. Εκτελεί επίσης και χρέη Υπεύθυνου Ανθρώπινου Δυναμικού.
- Ο κ. Αντώνης Π.: Είναι ο Τεχνικός Διευθυντής της Εταιρείας και Project Manager.
- Οι κ. Πανάρετου Αλεξάνδρα και Π. Βίκυ: Έχουν το ίδιο αντικείμενο εργασίας και καλύπτουν τις θέσεις Software Engineer – Web Developer.
- Η κ. Κ. Μαρία: Πραγματοποιεί συλλογή και καταχώρηση δεδομένων (Data collection and data entry).

Οι εργαζόμενοι της εταιρείας απασχολούνται 5 μέρες την εβδομάδα (από Δευτέρα έως Παρασκευή), εκτός των εθνικών αργιών και δικαιούνται 2 ημέρες άδειας για κάθε μήνα εργασίας τους στην εταιρεία.

3.2. Περιγραφή και Απαιτήσεις Συστήματος

Στα πλαίσια της παρούσας εργασίας υλοποιήθηκε ένα σύστημα για την αυτοματοποίηση της διαδικασίας Αίτησης και Χορήγησης Αδειών για την εταιρεία «ΑΒΓ Πληροφοριακά Συστήματα Ε.Π.Ε.». Το σύστημα αυτό θα χρησιμοποιηθεί για την απλούστευση της διαδικασίας αυτής, την διευκόλυνση των χρηστών και για τον



καλύτερο και αξιοκρατικότερο έλεγχο τήρησης των περιορισμών που έχουν τεθεί από τον Διευθυντή της εταιρείας.

Η εφαρμογή απαιτεί δύο διαφορετικές οθόνες χρήσης. Η πρώτη οθόνη θα χρησιμοποιείται από τον Υπεύθυνο Ανθρώπινου Δυναμικού, ο οποίος θα έχει τη δυνατότητα να προσθέσει ή να αφαιρέσει κάποιον εργαζόμενο από το σύστημα, αλλά και να τροποποιήσει τα στοιχεία κάποιου υπάρχοντος εργαζομένου. Επιπλέον, θα μπορεί να διαχειρίζεται τις αιτήσεις για χορήγηση αδειών και να έχει μία συνολική εικόνα των αδειών των εργαζομένων. Κατά την αποδοχή ή απόρριψη κάποιας αίτησης άδειας από τον Υπεύθυνο Ανθρώπινου Δυναμικού, θα αποστέλλεται αυτόματα ενημερωτικό e-mail προς τον ενδιαφερόμενο.

Η δεύτερη οθόνη θα χρησιμοποιείται από τους ίδιους τους εργαζόμενους, οι οποίοι θα έχουν τη δυνατότητα να δουν και να τροποποιήσουν τα στοιχεία τους και να έχουν μία συνοπτική εικόνα των αδειών τους. Συγκεκριμένα, θα έχουν τη δυνατότητα να δουν πόσες μέρες άδειας δικαιούνται μέχρι την τρέχουσα ημέρα, πόσες από αυτές έχουν καταναλώσει και πόσες μένουν ακόμη διαθέσιμες. Επιπλέον, θα έχουν τη δυνατότητα υποβολής αίτησης άδειας για συγκεκριμένες ημερομηνίες που ενδιαφέρονται, σύμφωνα πάντα με το διαθέσιμο υπόλοιπο ημερών που διαθέτουν, αλλά και ελέγχου των απαντήσεων από τον Υπεύθυνο Ανθρώπινου Δυναμικού στις αιτήσεις τους.

Κατά την υποβολή αίτησης άδειας από έναν εργαζόμενο για συγκεκριμένες ημερομηνίες, το σύστημα θα αποστέλλει e-mail στον Υπεύθυνο Ανθρώπινου Δυναμικού και στον Project Manager, προς αποδοχή ή απόρριψη.

3.3. Περιορισμοί και Παραδοχές Συστήματος

Όπως κάθε εφαρμογή, έτσι και η συγκεκριμένη ηλεκτρονική υπηρεσία, περιλαμβάνει μία σειρά από περιορισμούς, οι οποίοι είναι απαραίτητοι για την ορθή και αδιάλειπτη λειτουργία της εταιρείας. Οι περιορισμοί αυτοί θα πρέπει να ενσωματωθούν στο σύστημα και να πραγματοποιούνται οι απαραίτητοι έλεγχοι ανά



περίπτωση. Επιπλέον, ορίζονται μία σειρά από παραδοχές, οι οποίες θα συντελέσουν στον σαφέστερο ορισμό της διαδικασίας.

Οι βασικοί περιορισμοί και οι παραδοχές του συστήματος είναι οι ακόλουθοι:

- Οι εθνικές αργίες και τα Σαββατοκύριακα δεν προσμετρούνται στις ημέρες αδειών που αιτείται κάποιος χρήστης του συστήματος. Εάν η αίτηση ενός χρήστη περιλαμβάνει τέτοιες περιπτώσεις, τότε θα πρέπει να υπολογίζεται ο καθαρός αριθμός ημερών άδειας, ο οποίος θα αφαιρείται από τον συνολικό αριθμό διαθέσιμων ημερών.
- Δεν υπάρχει κάποιος περιορισμός για την ταυτόχρονη παροχή άδειας στους εργαζόμενους της εταιρείας. Η αξιολόγηση των αιτήσεων των εργαζομένων, βρίσκεται στην κρίση του Υπεύθυνου Ανθρώπινου Δυναμικού, ο οποίος γνωρίζει τις απαιτήσεις της εταιρείας στις διάφορες χρονικές περιόδους και παραχωρεί τις άδειες ανάλογα με τις ανάγκες και το φόρτο εργασίας της εταιρείας.
- Οι εργαζόμενοι της εταιρείας δικαιούνται 2 ημέρες άδειας για κάθε μήνα του χρόνου. Ο αριθμός των δυνατών ημερών άδειας που μπορεί να αιτηθεί ένας εργαζόμενος εντός ενός ημερολογιακού έτους, αφορά σε όλες τις μέρες του συγκεκριμένου έτους που μπορεί να λάβει. Ο μέγιστος αριθμός ημερών άδειας που μπορεί να λάβει ένας εργαζόμενος είναι 24 ημέρες.
- Για το ημερολογιακό έτος, κατά το οποίο πραγματοποιήθηκε η πρόσληψη του εργαζομένου, οι διαθέσιμες ημέρες άδειας περιορίζονται με βάση τον παρακάτω αλγόριθμο:
 - Εάν η πρόσληψη του εργαζομένου έχει πραγματοποιηθεί από 1-10 του μήνα, ο εργαζόμενος δικαιούται 2 ημέρες άδειας για τον συγκεκριμένο μήνα και 2 ημέρες άδεια για κάθε επόμενο μήνα, μέχρι το τέλος του χρόνου.
 - Εάν η πρόσληψη του εργαζομένου έχει πραγματοποιηθεί από 11-20 του μήνα, ο εργαζόμενος δικαιούται 1 ημέρα άδειας για τον συγκεκριμένο μήνα και 2 ημέρες άδεια για κάθε επόμενο μήνα, μέχρι το τέλος του χρόνου.
 - Εάν η πρόσληψη του εργαζομένου έχει πραγματοποιηθεί από 21 μέχρι το τέλος του μήνα, ο εργαζόμενος δεν δικαιούται άδεια για τον



συγκεκριμένο μήνα, παρά μόνο 2 ημέρες άδεια για κάθε επόμενο μήνα, μέχρι το τέλος του χρόνου.

- Οι συνολικές μέρες άδειας που αιτείται ένας εργαζόμενος δεν θα πρέπει να υπερβαίνουν τον αριθμό των διαθέσιμων ημερών που του απομένουν για το συγκεκριμένο ημερολογιακό έτος.
- Οι διαθέσιμες ημέρες άδειας θα πρέπει να καταναλώνονται εντός του ίδιου έτους και δεν μπορούν να μεταφερθούν σε επόμενο έτος. Σε περίπτωση που ένας εργαζόμενος δεν έχει χρησιμοποιήσει όλες τις διαθέσιμες ημέρες άδειας, τότε θα πρέπει να αποζημιώνεται για κάθε ημέρα άδειας που δεν έχει λάβει, ανάλογα με την οικονομική πολιτική της εταιρείας.
- Όταν κάποιος εργαζόμενος αιτείται άδεια, η ενημέρωση θα πρέπει να αποστέλλεται αυτόματα μέσω e-mail στον Υπεύθυνο Ανθρώπινου Δυναμικού.
- Σε περίπτωση που μία αίτηση άδειας αξιολογηθεί από τον υπεύθυνο Ανθρώπινου Δυναμικού (αποδοχή ή απόρριψη), θα πρέπει να ενημερώνεται αυτόματα ο εργαζόμενος, μέσω e-mail.
- Ο Υπεύθυνος Ανθρώπινου Δυναμικού θα πρέπει να έχει τη δυνατότητα Αποδοχής, ολικής ή μερικής, αλλά και Απόρριψης μίας αίτησης, ανάλογα πάντα με την τρέχουσα διαθεσιμότητα ημερών για τον συγκεκριμένο εργαζόμενο.
- Σε περίπτωση που η αίτηση άδειας για κάποια συγκεκριμένη ημερομηνία, δεν έχει αξιολογηθεί από τον Υπεύθυνο Ανθρώπινου Δυναμικού, μέχρι την προηγούμενη ημέρα, τότε θα πρέπει να θεωρείται απορριφθείσα.



4. Αρχιτεκτονικός Σχεδιασμός

4.1. Περιγραφή Διαδικασιών

Η εταιρεία «ΑΒΓ Πληροφοριακά Συστήματα Ε.Π.Ε.» σκοπεύει να δημιουργήσει μία νέα επιχειρησιακή διαδικασία για την ηλεκτρονική αίτηση και χορήγηση αδειών του προσωπικού της. Η διαδικασία αυτή αποτελείται από 2 βασικές υποδιαδικασίες:

3. Αίτηση Άδειας Εργαζομένου
4. Αξιολόγηση Άδειας Εργαζομένου

Οι παραπάνω υποδιαδικασίες, μπορούν να αναλυθούν περαιτέρω, ακολούθως:

- 1.5. Είσοδος στο Σύστημα και Συλλογή Δεδομένων Χρήστη
- 1.6. Παρουσίαση Αιτήσεων Εργαζομένου
- 1.7. Δημιουργία Νέας Αίτησης Άδειας Εργαζομένου
- 1.8. Ενημέρωση Υπεύθυνου Ανθρώπινου Δυναμικού

- 4.1. Είσοδος στο Σύστημα και Συλλογή Δεδομένων Χρήστη
- 4.2. Παρουσίαση Εκκρεμών Αιτήσεων Εργαζομένων
- 4.3. Αξιολόγηση (αποδοχή / απόρριψη) Αιτήσεων Εργαζομένων
- 4.4. Ενημέρωση Εργαζομένων

Η κάθε υποδιαδικασία αποτελείται από ένα σύνολο διεργασιών και βημάτων. Παρακάτω αναλύονται οι υποδιαδικασίες.

Υποδιαδικασίες 1.1., 2.1. – Συλλογή Δεδομένων Χρήστη και Είσοδος στο Σύστημα

Η είσοδος χρήστη στο σύστημα αποτελεί μία υπηρεσία υποστήριξης, η οποία χρησιμοποιείται και σε άλλες διαδικασίες του συστήματος:

1.1.1. Login (Utility Service)

Υποδιαδικασία 1.2. – Παρουσίαση Αιτήσεων Εργαζομένου



Η παρουσίαση αιτήσεων εργαζομένου αποτελείται από τις ακόλουθες διεργασίες:

- 1.2.1. Συλλογή Αιτήσεων Εργαζομένου (Domain Service)
- 1.2.2. Μετάφραση λεκτικών πινάκων αιτήσεων (Foundation Service)
- 1.2.3. Εύρεση συνολικού αριθμού ημερών άδειας εργαζομένου για το επιλεγμένο έτος (Foundation Service)
- 1.2.4. Εύρεση αριθμού αποδεκτών ημερών άδειας εργαζομένου (Foundation Service)
- 1.2.5. Εύρεση υπολοίπου ημερών άδειας (Utility Service)

Υποδιαδικασία 1.3. – Νέα Αίτηση Άδειας

Η νέα αίτηση άδειας από τον εργαζόμενο αποτελείται από τις ακόλουθες διεργασίες:

- 1.3.1. Έλεγχος Εγκυρότητας Στοιχείων Αίτησης (Utility Service)
- 1.3.2. Υπολογισμός αριθμού ημερών άδειας αίτησης (Utility Service)
- 1.3.2. Έλεγχος διαθεσιμότητας ημερών (Utility Service)
- 1.3.3. Καταχώρηση Νέας Αίτησης (Foundation Service)

Υποδιαδικασία 2.2. – Παρουσίαση Εκκρεμών Αιτήσεων Εργαζομένων

Η παρουσίαση εκκρεμών αιτήσεων εργαζομένων αποτελείται από τις ακόλουθες διεργασίες:

- 2.2.1. Συλλογή Εκκρεμών Αιτήσεων Εργαζομένων (Domain Service)
- 2.2.2. Μετάφραση λεκτικών πινάκων αιτήσεων (Foundation Service)

Υποδιαδικασία 2.3. – Αξιολόγηση Αιτήσεων Εργαζομένων

Η αξιολόγηση αιτήσεων εργαζομένων πραγματοποιείται μόνο από τον Υπεύθυνο Ανθρώπινου Δυναμικού και αποτελείται από τις ακόλουθες διεργασίες:

- 2.3.1. Έλεγχος Εγκυρότητας Στοιχείων Αίτησης (Utility Service)



- 2.3.2. Υπολογισμός αριθμού αποδεκτών ημερών άδειας αίτησης (Utility Service)
- 2.3.2. Έλεγχος διαθεσιμότητας ημερών (Utility Service)
- 2.3.3. Καταχώρηση Αποδοχής Αίτησης (Foundation Service)
- 2.3.4. Καταχώρηση Απόρριψης Αίτησης (Foundation Service)

Web Services της εφαρμογής

Για τις υποδιαδικασίες που περιγράφηκαν παραπάνω, η εταιρεία χρησιμοποιεί έξι βασικές υπηρεσίες (Foundation Services):

- Μετάφραση λεκτικών πινάκων αιτήσεων, η οποία λαμβάνει έναν τύπο λεκτικού με την αντίστοιχη τιμή του και επιστρέφει το λεκτικό.
- Εύρεση συνολικού αριθμού ημερών άδειας εργαζομένου για το επιλεγμένο έτος, η οποία λαμβάνει την ημερομηνία πρόσληψης ενός εργαζομένου και το ημερολογιακό έτος για το οποίο ζητάμε τον αριθμό ημερών άδειας που μπορεί να λάβει ο εργαζόμενος και επιστρέφει το συνολικό αριθμό ημερών για το συγκεκριμένο έτος.
- Εύρεση αριθμού αποδεκτών ημερών άδειας εργαζομένου για το επιλεγμένο έτος, η οποία λαμβάνει το ημερολογιακό έτος για το οποίο ζητάμε τον αριθμό ημερών άδειας που έχει ήδη λάβει ο εργαζόμενος και επιστρέφει το συνολικό αριθμό αποδεκτών ημερών για το συγκεκριμένο έτος.
- Καταχώρηση Νέας Αίτησης, η οποία λαμβάνει τα στοιχεία μίας νέας αίτησης και τα καταχωρεί στη βάση.
- Ενημέρωση Υπάρχουσας Αίτησης, η οποία λαμβάνει το ID της αίτησης που θέλουμε να ενημερώσουμε και τα στοιχεία της αίτησης και τα ενημερώνει στη βάση.
- Εύρεση Επίσημων Εθνικών Αργιών, η οποία λαμβάνει ένα εύρος ημερομηνιών ή ένα έτος και επιστρέφει όλες τις διαθέσιμες επίσημες αργίες για το συγκεκριμένο χρονικό διάστημα.

Για την είσοδο του χρήστη στο σύστημα και τη συλλογή των στοιχείων του χρησιμοποιείται μία βοηθητική υπηρεσία (Utility Service), η οποία δέχεται ένα username – password, επιβεβαιώνει την ύπαρξη του χρήστη στο σύστημα και επιστρέφει όλα τα απαραίτητα στοιχεία του.



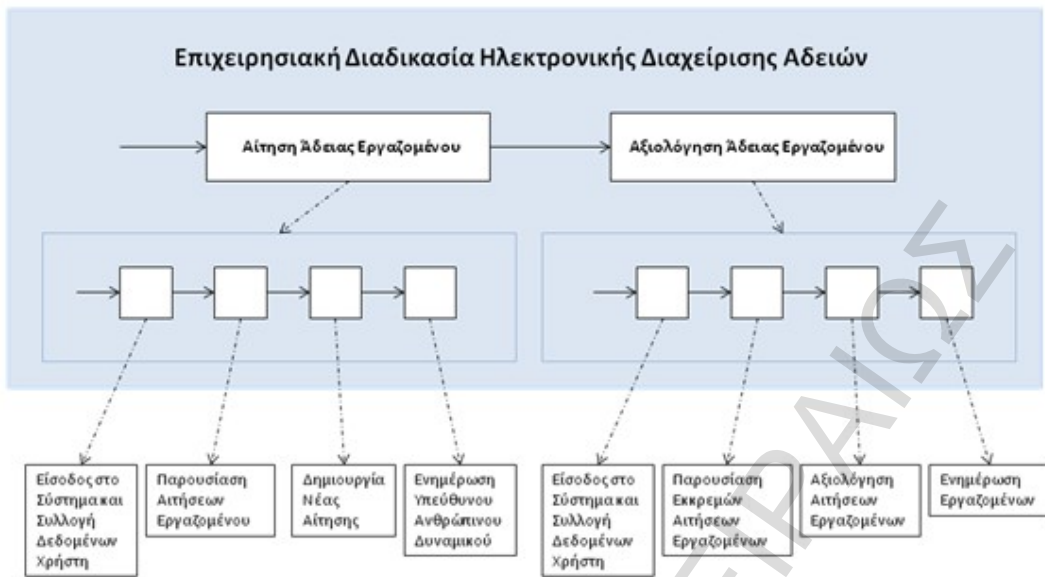
Για την εγγραφή ενός νέου εργαζομένου στο σύστημα χρησιμοποιείται μία σύνθετη υπηρεσία (Domain Service), η οποία λαμβάνει τα στοιχεία του εργαζομένου, ελέγχει τη διαθεσιμότητα του username και εισάγει τα στοιχεία αυτά στη βάση.

Για την εμφάνιση των αιτήσεων άδειας έχει δημιουργηθεί μία σύνθεση υπηρεσία (Domain Service), η οποία λαμβάνει τα κριτήρια φιλτραρίσματος και επιστρέφει τις εγγραφές αιτήσεων άδειας.

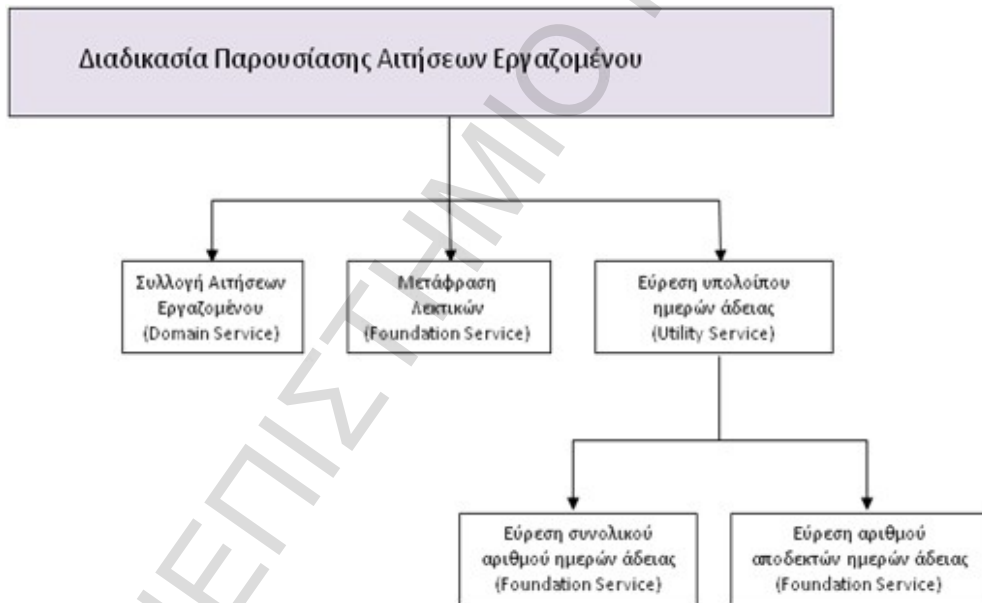
Για τον υπολογισμό του υπολοίπου ημερών άδειας εργαζομένου, έχει δημιουργηθεί μία βοηθητική υπηρεσία (Utility Service), η οποία με τη σειρά της καλεί τις 2 βασικές υπηρεσίες, εύρεση συνολικού αριθμού ημερών άδειας και εύρεση αποδεκτών ημερών άδειας και πραγματοποιεί τους απαραίτητους υπολογισμούς.

Για τον έλεγχο των στοιχείων μίας αίτησης άδειας, έχει χρησιμοποιηθεί μία βοηθητική υπηρεσία (Utility Service), η οποία ελέγχει την εγκυρότητα των ημερομηνιών της αίτησης. Στη συνέχεια καλείται η υπηρεσία υπολογισμού αριθμού ημερών άδειας της συγκεκριμένης αίτησης, η οποία με τη σειρά της καλεί μία βασική υπηρεσία εύρεσης των επίσημων αργιών για το συγκεκριμένο χρονικό διάστημα. Ο αριθμός των ημερών της συγκεκριμένης άδειας χρησιμοποιείται στη βοηθητική υπηρεσία ελέγχου διαθεσιμότητας ημερών, με βάση τις άδειες που έχει λάβει μέχρι στιγμής ο χρήστης. Οι παραπάνω έλεγχοι πραγματοποιούνται τόσο για την επεξεργασία της άδειας από τον ίδιο τον εργαζόμενο, όσο και κατά την επεξεργασία από τον Υπεύθυνο Ανθρώπινου Δυναμικού, για την αποδοχή της.

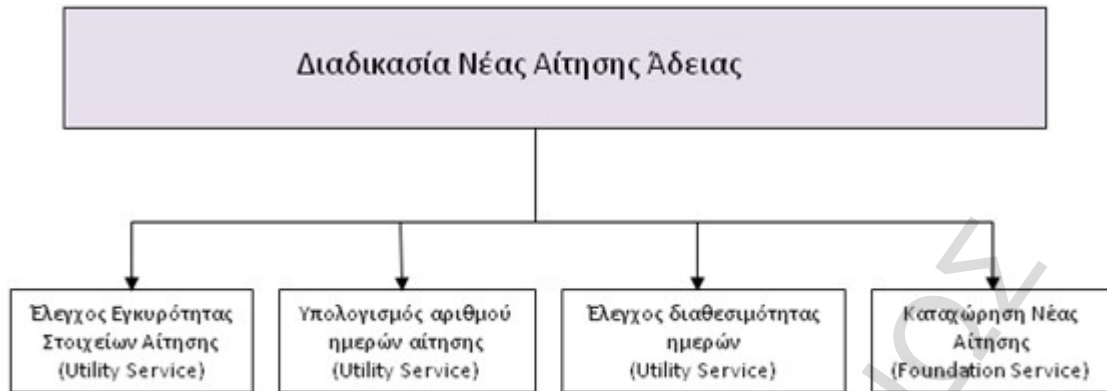
4.2. Ιεραρχικό Διάγραμμα



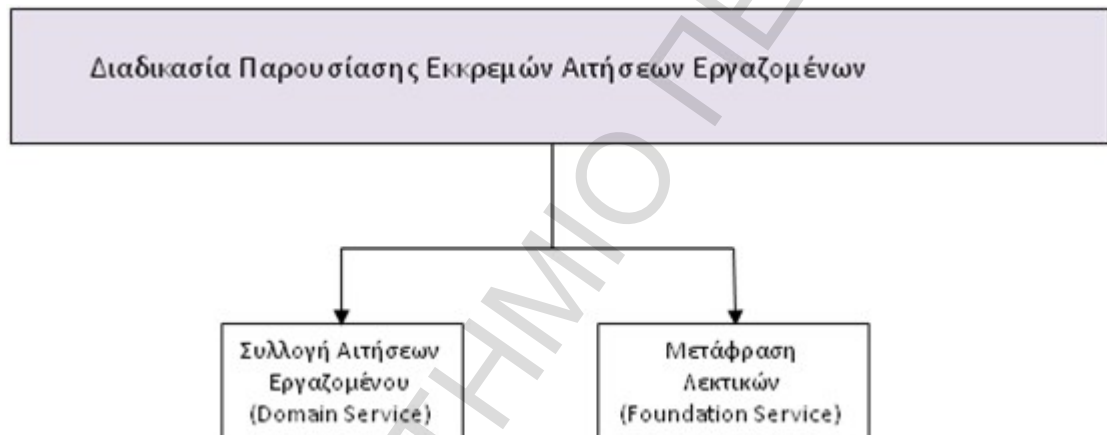
Εικόνα 3: Επιχειρησιακή Διαδικασία



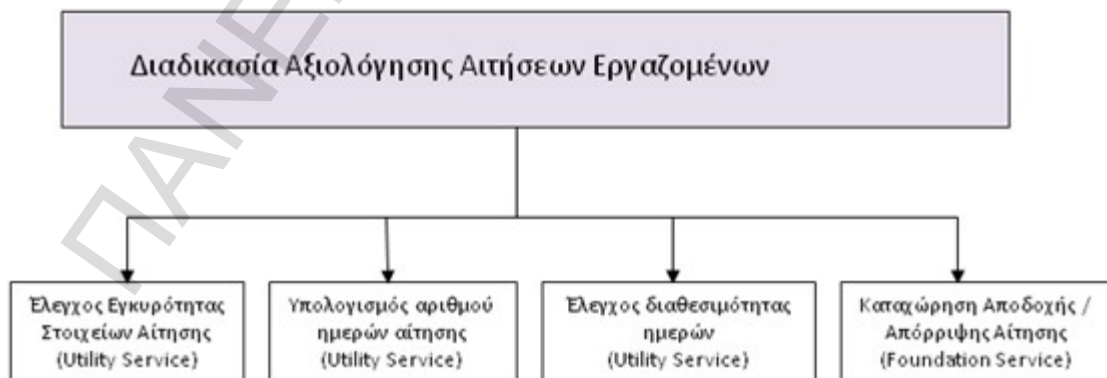
Εικόνα 4: Διαδικασία Παρουσίασης Αιτήσεων Εργαζομένου



Εικόνα 5: Διαδικασία Νέας Αίτησης Άδειας



Εικόνα 6: Διαδικασία Παρουσίασης Εκκρεμών Αιτήσεων Εργαζομένων



Εικόνα 7: Διαδικασία Αξιολόγησης Αιτήσεων Εργαζομένων

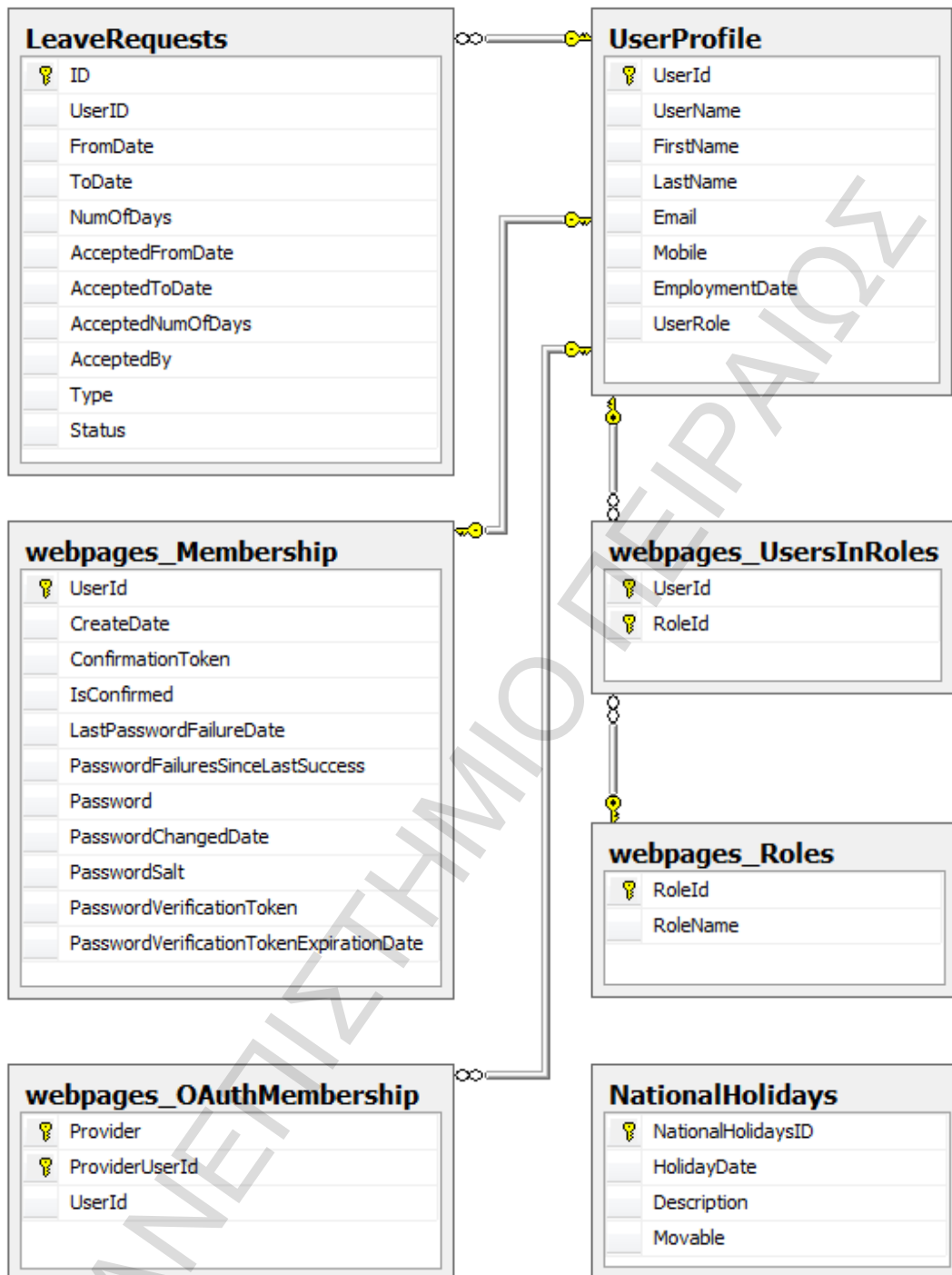


4.3. Σχεδιασμός Βάσης Δεδομένων

Η βάση του LMS Service αποτελείται από 7 πίνακες.

1. LeaveRequests: Περιλαμβάνει τις αιτήσεις άδειας των υπαλλήλων.
2. UserProfile: Περιλαμβάνει τα απαραίτητα στοιχεία κάθε χρήστη του συστήματος.
3. webpages_Membership: Περιλαμβάνει βοηθητικά στοιχεία σύνδεσης των χρηστών στο σύστημα.
4. webpages_UsersInRoles: Περιλαμβάνει τη σχέση μεταξύ χρηστών και κατηγοριών χρηστών του συστήματος.
5. webpages_Roles: Περιλαμβάνει τις δυνατές κατηγορίες χρηστών του συστήματος.
6. webpages_OAuthMembership: Περιλαμβάνει τη σχέση μεταξύ χρήστη και γκρουπ χρηστών. Έχει προστεθεί για μελλοντική χρήση.
7. NationalHolidays: Περιλαμβάνει όλες τις επίσημες εθνικές αργίες.

Οι πίνακες της βάσης δεδομένων δημιουργήθηκαν με τον Microsoft SQL Server 2008 R2. Ακολουθεί το διάγραμμα οντοτήτων – σχέσεων, το οποίο δημιουργήθηκε επίσης μέσω του Microsoft SQL Server 2008 R2.



Εικόνα 8: Διάγραμμα Οντοτήτων – Σχέσεων



4.4. Περιγραφή Συστήματος

Το σύστημα LMS Service υλοποιήθηκε σε περιβάλλον Microsoft Visual Studio 2010 με .NET Framework 4. Η γλώσσα προγραμματισμού που χρησιμοποιήθηκε είναι η C#. Αποτελεί ένα Restful Service, το οποίο έχει υλοποιηθεί με τη βοήθεια του ASP .NET MVC4 Framework.

Το ASP .NET MVC4 Framework αποτελεί ένα Framework για την υλοποίηση υπηρεσιών και διαδικτυακών εφαρμογών με τη βοήθεια του πρωτοκόλλου REST. Ακολουθεί την αρχιτεκτονική Model-View-Controller, η οποία ορίζεται ως εξής:

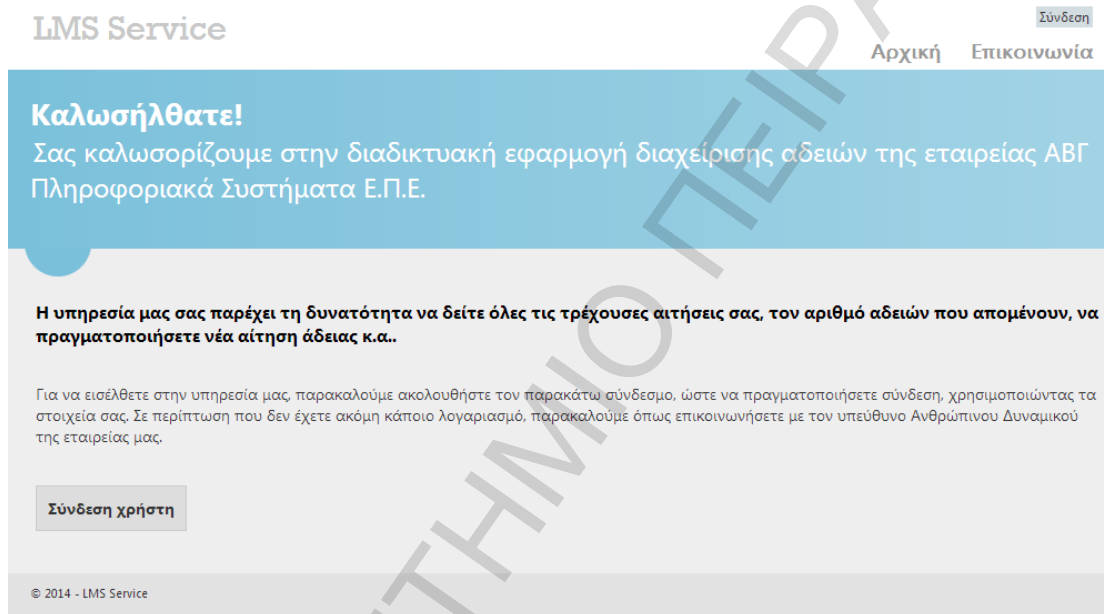
- Το Model περιλαμβάνει τη δομή των οντοτήτων που χρησιμοποιούνται από τους διάφορους Controllers για την επεξεργασία δεδομένων. Συνήθως τα Models αναπαριστούν πίνακες της βάσης δεδομένων.
- Το View χρησιμοποιείται για την παρουσίαση των δεδομένων. Το View δεν πρέπει να περιλαμβάνει κάποιο Business Logic, καθώς χρησιμοποιείται καθαρά για την αναπαράσταση των δεδομένων.
- Ο Controller χρησιμοποιεί ένα Model για τη συλλογή δεδομένων, βάσει συγκεκριμένων κριτηρίων και φίλτρων, τα οποία στη συνέχεια στέλνει σε ένα View.

Συνοπτικά, ο Controller χρησιμοποιεί ένα Model για να συλλέξει κάποια δεδομένα, τα οποία στη συνέχεια στέλνει στο View, ώστε να τα παρουσιάσει με κατάλληλο τρόπο.

Η εφαρμογή αποτελείται από το project LMSServices, το οποίο περιλαμβάνει τα Models, τα Views, τους Controllers αλλά και όλα τα βοηθητικά στοιχεία για τις ιστοσελίδες που δημιουργήθηκαν, όπως τα απαραίτητα scripts και τα stylesheets. Στους Controllers περιλαμβάνονται και τα Restful Services που έχουμε υλοποιήσει στα πλαίσια της εφαρμογής.

5. Εγχειρίδιο Χρήσης

Η αρχική σελίδα της εφαρμογής περιλαμβάνει κάποια πληροφοριακά στοιχεία της εφαρμογής και της εταιρείας και τη βασική δυνατότητα εισόδου στο σύστημα. Δεν υπάρχει η άμεση δυνατότητα εγγραφής στην εφαρμογή, καθώς αυτή η λειτουργία θα πρέπει να πραγματοποιείται αποκλειστικά από τον Υπεύθυνο Ανθρώπινου Δυναμικού της εταιρείας, ο οποίος στη συνέχεια παραχωρεί τα απαραίτητα στοιχεία πρόσβασης στον εργαζόμενο.



Εικόνα 9: Αρχική οθόνη εφαρμογής

Επιλέγοντας το σύνδεσμο «Σύνδεση» (πάνω δεξιά) ή το κουμπί «Σύνδεση Χρήστη» που βρίσκεται στην οθόνη, ο χρήστης έχει τη δυνατότητα να πραγματοποιήσει είσοδο στο σύστημα. Σε αυτή την οθόνη ο χρήστης θα πρέπει να συμπληρώσει το username και το password που του έχει δοθεί από τον Υπεύθυνο Ανθρώπινου Δυναμικού. Σε περίπτωση που πληκτρολογήσει λάθος κάποιο στοιχείο, θα εμφανιστεί αντίστοιχο μήνυμα στην οθόνη, προτρέποντας το χρήστη να προσπαθήσει ξανά. Επιπλέον, υπάρχει η δυνατότητα αποθήκευσης του κωδικού (επιλέγοντας το checkbox “Remember me?”, ώστε να μην χρειάζεται να πληκτρολογεί τα στοιχεία κάθε φορά ο χρήστης.

LMS Service

Σύνδεση

Αρχική Επικοινωνία

Σύνδεση

User name

Password

Remember me?

Σύνδεση

© 2014 - LMS Service

Εικόνα 10: Είσοδος Εργαζομένου

Μετά την επιτυχή εισαγωγή του χρήστη στο σύστημα, εμφανίζεται η παρακάτω οθόνη ως αρχική σελίδα, η οποία παρουσιάζει τις δυνατότητες του χρήστη. Οι επιλογές αυτές διαφέρουν, ανάλογα με τον τύπο του χρήστη.

LMS Service

Γεια σου, Alexa! Αποσύνδεση

Αρχική Επικοινωνία

Καλωσήλθατε!

Σας καλωσορίζουμε στην διαδικτυακή εφαρμογή διαχείρισης αδειών της εταιρείας ABΓ Πληροφορικά Συστήματα Ε.Π.Ε.

Η υπηρεσία μας σας παρέχει τη δυνατότητα να δείτε όλες τις τρέχουσες αιτήσεις σας, τον αριθμό αδειών που απομένουν, να πραγματοποιήσετε νέα αίτηση άδειας κ.α..

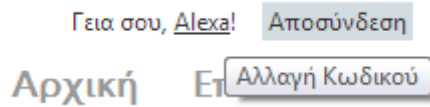
[Εμφάνιση Αιτήσεων](#)

[Δημιουργία νέας αίτησης](#)

© 2014 - LMS Service

Εικόνα 11: Αρχική οθόνη συνδεδεμένου χρήστη

Πατώντας το σύνδεσμο που βρίσκεται στο όνομα του χρήστη, στο πάνω δεξί μέρος της οθόνης, ο χρήστης έχει τη δυνατότητα να τροποποιήσει τον κωδικό πρόσβασης του στο σύστημα.



Εικόνα 12: Σύνδεσμος αλλαγής κωδικού πρόσβασης χρήστη

Η οθόνη που εμφανίζεται είναι η ακόλουθη:

A screenshot of the 'LMS Service' user interface. The page title is 'LMS Service'. The navigation menu includes 'Γεια σου, Alexa!', 'Αποσύνδεση', 'Αρχική', and 'Επικοινωνία'. The main content area is titled 'Αλλαγή Κωδικού Πρόσβασης' (Change Password). Below the title, it says 'Είστε συνδεδεμένος ως Alexa.' (You are logged in as Alexa.). There are three input fields: 'Παλιός κωδικός' (Old password), 'Νέος κωδικός' (New password), and 'Επιβεβαίωση Κωδικού' (Confirm password). A button labeled 'Αλλαγή Κωδικού' (Change Password) is at the bottom. The footer contains the text '© 2014 - LMS Service'.

Εικόνα 13: Οθόνη αλλαγής κωδικού πρόσβασης χρήστη

Πατώντας το σύνδεσμο «Εμφάνιση Αιτήσεων» που βρίσκεται στην αρχική σελίδα, ο χρήστης μεταφέρεται στην οθόνη παρουσίασης των αιτήσεων του ανά έτος. Σε αυτή την οθόνη μπορεί να δει την κατάσταση όλων των αιτήσεων του, ενώ για αυτές που είναι ακόμη εκκρεμείς (δεν έχουν αξιολογηθεί από τον Υπεύθυνο Ανθρώπινου Δυναμικού), έχει τη δυνατότητα να τις επεξεργαστεί, ώστε να αλλάξει τις ημερομηνίες της αίτησης ή και να τις διαγράψει. Σε περίπτωση που μία αίτηση έχει αξιολογηθεί ή είναι παρελθόντος χρόνου, δεν μπορεί να την επεξεργαστεί ή να την διαγράψει.

Σε αυτή την οθόνη, ο χρήστης μπορεί να δει επίσης τις συνολικές ημέρες άδειας που μπορεί να λάβει για το επιλεγμένο έτος, τις ημέρες άδειας που έχουν γίνει αποδεκτές μέχρι στιγμής και το υπόλοιπο ημερών που του απομένει.

LMS Service Για σου, **Αλεξά** | Αποσύνδεση
Αρχική | Επικοινωνία

Αιτήσεις Αδειας Χρήστη

Έτος Παρουσίασης: 2014

Από	Έως	Τύπος Άδειας	Αριθμός Ημερών	Κατάσταση	
01-01-2014	07-01-2014	Άδεια Διακοπών	3	Αποδεκτή	Λεπτομέρειες
13-01-2014	13-01-2014	Άδεια Διακοπών	1	Αποδεκτή	Λεπτομέρειες
03-02-2014	07-02-2014	Άδεια Διακοπών	5	Εκκρεμεί	Λεπτομέρειες Επεξεργασία Διαγραφή

[Νέα Αίτηση](#)

Συνολικές ημέρες άδειας για το έτος 2014	Αποδεκτές ημέρες άδειας	Υπόλοιπο ημερών άδειας
24	4	20

© 2014 - LMS Service

Εικόνα 14: Εμφάνιση αιτήσεων χρήστη ανά έτος

Η αλλαγή του έτους για το οποίο εμφανίζονται όλες οι αιτήσεις, πραγματοποιείται από το select box που βρίσκεται στο πάνω μέρος της οθόνης. Εκεί εμφανίζονται όλα τα έτη από το έτος πρόσληψης του συγκεκριμένου υπαλλήλου, έως και ένα χρόνο μετά το τρέχον έτος.

Αιτήσεις Αδειας Χρήστη

Έτος Παρουσίασης: 2014

Από	Έως	Τύπος Άδειας	Αριθμός Ημερών	Κατάσταση
01-01-2014	07-01-2014			
13-01-2014	13-01-2014	Άδεια Διακοπών	1	

Εικόνα 15: Επιλογή έτους εμφάνισης αιτήσεων

Σε περίπτωση που μία άδεια έχει γίνει μερικώς αποδεκτή, στον πίνακα παρουσίασης αιτήσεων εμφανίζονται με μικρά γράμματα οι ημερομηνίες αίτησης του χρήστη, για λόγους ιστορικού, όπως φαίνεται παρακάτω:

Από	Έως	Τύπος Άδειας	Αριθμός Ημερών	Κατάσταση	
22-08-2013	01-09-2013 (Αίτηση: 02-09-2013)	Άδεια Διακοπών	7 (Αίτηση: 8)	Αποδεκτή	Λεπτομέρειες

Εικόνα 16: Εμφάνιση στοιχείων αρχικής αίτησης, εφόσον η αίτηση έχει γίνει μερικώς μόνο αποδεκτή

Στην παρακάτω εικόνα εμφανίζεται η φόρμα επεξεργασίας αίτησης χρήστη. Σε αυτή την οθόνη ο χρήστης μπορεί να δει τα πλήρη στοιχεία της αίτησης, αλλά μπορεί να τροποποιήσει μόνο τις ημερομηνίες έναρξης και λήξης της άδειας. Πατώντας το κουμπί «Αποθήκευση», τα νέα στοιχεία της αίτησης αποθηκεύονται στη βάση, μόνο εφόσον ικανοποιούν τους απαραίτητους ελέγχους, οι οποίοι πραγματοποιούνται αυτόματα.

LMS Service

Γεια σου, [Alexa!](#) [Αποσύνδεση](#)

[Αρχική](#) [Επικοινωνία](#)

Επεξεργασία Χρήστης

Πανάρετου Αλεξάνδρα

Από
2014-02-03

Έως
2014-02-07

Τύπος Άδειας
Άδεια Διακοπών

Αριθμός Ημερών
5

Αποθήκευση

[Επιστροφή στη Λίστα](#)

© 2014 - LMS Service

Εικόνα 17: Φόρμα επεξεργασίας αίτησης χρήστη

Ο χρήστης έχει τη δυνατότητα να πραγματοποιήσει μία νέα αίτηση άδειας, ακολουθώντας το σύνδεσμο «Νέα Αίτηση» που βρίσκεται στην οθόνη εμφάνισης αιτήσεων ή στην αρχική σελίδα. Η φόρμα νέας αίτησης είναι η ακόλουθη:

LMS Service

Γεια σου, Alex! Αποσύνδεση

Αρχική Επικοινωνία

Νέα Αίτηση

Από

Έως

Τύπος Άδειας

Άδεια Διακοπών

Αίτηση

[Επιστροφή στη Λίστα](#)

© 2014 - LMS Service

Εικόνα 18: Φόρμα νέας αίτησης

Σε περίπτωση που τα στοιχεία της αίτησης είναι σωστά και ικανοποιούν τις απαραίτητες συνθήκες, τότε η νέα αίτηση προστίθεται στον πίνακα παρουσίασης αιτήσεων, με κατάσταση «Εκκρεμεί», ενώ αυτόματα αποστέλλεται ενημερωτικό e-mail στον Υπεύθυνο Ανθρώπινου Δυναμικού, ώστε να ενημερωθεί για την νέα αίτηση, την οποία θα πρέπει να αξιολογήσει.

Η αρχική οθόνη του Υπεύθυνου Ανθρώπινου Δυναμικού είναι η ακόλουθη:

LMS Service

Γεια σου, ddi! Αποσύνδεση

Αρχική Εγγραφή Νέου Χρήστη Επικοινωνία

Καλωσήλθατε!

Σας καλωσορίζουμε στην διαδικτυακή εφαρμογή διαχείρισης αδειών της εταιρείας ABΓ Πληροφορικά Συστήματα Ε.Π.Ε.

Η υπηρεσία μας σας παρέχει τη δυνατότητα να δείτε όλες τις τρέχουσες αιτήσεις σας, τον αριθμό αδειών που απομένουν, να πραγματοποιήσετε νέα αίτηση άδειας κ.α..

Διαχείριση Αδειών Χρηστών

[Εμφάνιση Αιτήσεων σε εκκρεμότητα](#)

Διαχείριση Προσωπικών Αιτήσεων

[Εμφάνιση Αιτήσεων](#)

[Δημιουργία νέας αίτησης](#)

© 2014 - LMS Service

Εικόνα 19: Αρχική οθόνη Υπεύθυνου Ανθρώπινου Δυναμικού

Μπορούμε να δούμε ότι υπάρχουν οι ίδιες επιλογές που έχει ένας απλός εργαζόμενος, με την προσθήκη των συνδέσμων «Εγγραφή Νέου Χρήστη» και «Εμφάνιση Αιτήσεων σε εκκρεμότητα».

Ο σύνδεσμος «Εγγραφή Νέου Χρήστη» εμφανίζει την παρακάτω φόρμα, μέσω της οποίας ο Υπεύθυνος Ανθρώπινου Δυναμικού μπορεί να προσθέσει έναν νέο εργαζόμενο στην εταιρεία.

LMS Service Γεια σου dd! Αποσύνδεση

Αρχική Εγγραφή Νέου Χρήστη Επικοινωνία

Εγγραφή Νέου Χρήστη.

Όνομα Χρήστη

Κωδικός

Επιβεβαίωση Κωδικού

Όνομα

Επώνυμο

Ημερομηνία Πρόσληψης

Email

Κινητό Τηλέφωνο

Τύπος Υπαλλήλου
Human Resources

© 2014 - LMS Service

Εικόνα 20: Φόρμα προσθήκης νέου χρήστη

Κατά την υποβολή της φόρμας πραγματοποιείται έλεγχος εγκυρότητας των δεδομένων και έλεγχος διαθεσιμότητας του Ονόματος Χρήστη. Σε περίπτωση που αυτοί είναι επιτυχείς, τότε προστίθεται ο νέος χρήστης στο σύστημα, ο οποίος ενημερώνεται αυτόματα με e-mail, στο e-mail που έχει καταχωρηθεί για τον συγκεκριμένο χρήστη.

Ο σύνδεσμος «Εμφάνιση Αιτήσεων σε εκκρεμότητα» παρουσιάζει όλες τις εκκρεμείς αιτήσεις των χρηστών του συστήματος, όπως φαίνεται στην παρακάτω οθόνη:

The screenshot shows the LMS Service interface. At the top, there is a navigation bar with links for 'Αρχική', 'Εγγραφή Νέου Χρήστη', and 'Επικοινωνία'. Below the navigation bar, the main heading is 'Αιτήσεις Χρηστών σε εκκρεμότητα'. A table displays the following data:

Χρήστης	Από	Έως	Τύπος Άδειας	Αριθμός Ημερών	Κατάσταση	
Πανάρετου Αλεξάνδρα	03-02-2014	07-02-2014	Άδεια Διακοπών	5	Εκκρεμεί	Λεπτομέρειες Επεξεργασία Απόρριψη
Δ. Δημήτρης	07-07-2014	10-07-2014	Άδεια Διακοπών	4	Εκκρεμεί	Λεπτομέρειες Επεξεργασία Απόρριψη
Π. Βίκυ	27-01-2014	31-01-2014	Άδεια Διακοπών	5	Εκκρεμεί	Λεπτομέρειες Επεξεργασία Απόρριψη
Π. Αντώνης	10-02-2014	14-02-2014	Άδεια Διακοπών	5	Εκκρεμεί	Λεπτομέρειες Επεξεργασία Απόρριψη

© 2014 - LMS Service

Εικόνα 21: Οθόνη εμφάνισης αιτήσεων σε εκκρεμότητα

Από αυτή την οθόνη ο χρήστης έχει τη δυνατότητα να δει τις λεπτομέρειες μίας αίτησης, να πραγματοποιήσει επεξεργασία, ώστε να ορίσει τις ημερομηνίες αποδοχής ή να απορρίψει την αίτηση. Η φόρμα επεξεργασίας της αίτησης εμφανίζεται στην παρακάτω εικόνα. Όλες οι ημερομηνίες καταχωρούνται με τη βοήθεια ενός ημερολογίου (calendar popup).

Ο Υπεύθυνος Ανθρώπινου Δυναμικού δεν έχει τη δυνατότητα να τροποποιήσει τις ημερομηνίες αίτησης των άλλων χρηστών, παρά μόνο να καταχωρήσει τις ημερομηνίες αποδοχής. Οι ημερομηνίες αποδοχής πρέπει να περικλείονται στις ημερομηνίες αίτησης. Κατά τη στιγμή της αποδοχής μίας αίτησης πραγματοποιείται ξανά ο έλεγχος για τη δυνατότητα λήψης ημερών άδειας από τον χρήστη.

Σε περίπτωση που αλλάξει η κατάσταση μίας αίτησης άδειας, ο χρήστης ενημερώνεται αυτόματα μέσω e-mail.



LMS Service Γεια σου dd! [Αποσύνδεση](#)

[Αρχική](#) [Εγγραφή Νέου Χρήστη](#) [Επικοινωνία](#)

Επεξεργασία

Χρήστης
Πανάρετου Αλεξάνδρα

Από

Έως

Αποδεκτή από

Αποδεκτή έως

Τύπος Άδειας January 2014

Αριθμός Ημερ

Su	Mo	Tu	We	Th	Fr	Sa
			1	2	3	4
5	6	7	8	9	10	11
12	13	14	15	16	17	18
19	20	21	22	23	24	25
26	27	28	29	30	31	

Αποθήκευση

[Επιστροφή στη Λίστα Αιτήσεων σε εκκρεμότητα](#)
[Επιστροφή στη Λίστα Προσωπικών Αιτήσεων](#)

© 2014 - LMS Service

Εικόνα 22: Φόρμα επεξεργασίας αίτησης από τον Υπεύθυνο Ανθρώπινου Δυναμικού



6. Συμπεράσματα

6.1. Αξιολόγηση Εφαρμογής – Συμπεράσματα

Σε μία προσπάθεια αξιολόγησης της εφαρμογής που υλοποιήθηκε, μπορούμε να πούμε ότι επιτεύχθηκε η αυτοματοποίηση της διαδικασίας αίτησης και χορήγησης αδειών για την εταιρεία ΑΒΓ Πληροφοριακά Συστήματα ΕΠΕ. Ο χρήστης λαμβάνει τα στοιχεία πρόσβασης από τον Υπεύθυνο Ανθρώπινου Δυναμικού και εισέρχεται στο σύστημα, όπου μπορεί να δει όλες τις αιτήσεις που έχει πραγματοποιήσει, καταχωρημένες ανά έτος, να δει τις ημέρες άδειας που δικαιούται, που έχει λάβει και που του απομένουν ανά έτος και να πραγματοποιήσει μία νέα αίτηση, με βάση το υπόλοιπο ημερών του. Από την άλλη μεριά, ο Υπεύθυνος Ανθρώπινου Δυναμικού έχει αντίστοιχες δυνατότητες, με κάποιες επιπλέον, λόγω του ρόλου του στο σύστημα. Συγκεκριμένα, ο Υπεύθυνος Ανθρώπινου Δυναμικού έχει τη δυνατότητα να προσθέσει έναν νέο εργαζόμενο στο σύστημα και να αξιολογήσει τις εκκρεμείς αιτήσεις των χρηστών, αποδέχοντας ή απορρίπτοντάς τες.

Η επιχειρησιακή διαδικασία αναλύθηκε και σχεδιάστηκε με βάση την αρχιτεκτονική SOA, χρησιμοποιώντας το REST πρωτόκολλο. Από αυτή τη σχεδίαση προέκυψαν και τα web services της εφαρμογής, τα οποία τεκμηριώθηκαν με κατάλληλα ιεραρχικά διαγράμματα. Η υλοποίηση της εφαρμογής πραγματοποιήθηκε με οδηγό τα ιεραρχικά διαγράμματα.

6.2. Μελλοντικές Βελτιώσεις

Η υλοποίηση και λειτουργία του αυτοματοποιημένου συστήματος ήταν επιτυχής. Παρ' όλα αυτά, υπάρχουν κάποια θέματα τα οποία αξίζει να σημειωθούν και σχετίζονται τόσο με την βελτίωση όσο και με την εισαγωγή νέων υπηρεσιών στην εφαρμογή. Παρακάτω αναφέρονται οι βελτιώσεις της εφαρμογής:



Γραφική Απεικόνιση Ημερών Άδειας σε Ημερολόγιο

Ο χρήστης θα μπορεί να βλέπει όλες τις αιτήσεις που έχει πραγματοποιήσει σε ένα ετήσιο ημερολόγιο, το οποίο θα χρησιμοποιεί συγκεκριμένα χρώματα για την αναπαράσταση των διαφορετικών καταστάσεων μίας αίτησης. Στο ίδιο ημερολόγιο θα εμφανίζονται και οι επίσημες αργίες. Με αυτό τον τρόπο θα μπορεί να έχει άμεσα μια πληρέστερη εικόνα των αιτήσεών του.

Δημιουργία Mobile Εφαρμογής

Ο χρήστης θα μπορεί να βλέπει όλες τις αιτήσεις που έχει πραγματοποιήσει, αλλά και να πραγματοποιήσει μία νέα αίτηση από κάποια κινητή συσκευή (smartphone/tablet κ.τ.λ.), μέσω της αντίστοιχης εφαρμογής. Με αυτό τον τρόπο προστίθεται ευελιξία στο σύστημα. Αυτό θα είναι περισσότερο χρήσιμο για τον Υπεύθυνο Ανθρώπινου Δυναμικού, ο οποίος θα μπορεί άμεσα να διαχειρίζεται τις αιτήσεις των χρηστών.

Διασύνδεση με SMS API

Οι χρήστες του συστήματος θα μπορούν να ενημερώνονται άμεσα, στο κινητό τους τηλέφωνο, σχετικά με την αλλαγή κατάστασης κάποιας άδειας. Αυτό προϋποθέτει την συμφωνία, επί πληρωμή, με κάποια εταιρεία κινητής τηλεφωνίας.

Χρήση Ajax Components

Σε μία προσπάθεια αναβάθμισης της εφαρμογής και καλύτερης εξυπηρέτησης των χρηστών, θα μπορούσαν να τροποποιηθούν τα βήματα καταχώρησης και επεξεργασίας αίτησης, με τρόπο ώστε ο έλεγχος εγκυρότητας των δεδομένων της αίτησης να πραγματοποιείται αυτόματα μέσω Ajax, χωρίς να απαιτείται το πάτημα του κουμπιού και η ενημέρωση της σελίδας.



ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ



Βιβλιογραφία

- Θεμιστοκλέους Μ., Μαντζάνα Β., (2010), *Υπηρεσίες Παγκόσμιου Ιστού και Υπηρεσιοστρεφείς Αρχιτεκτονικές (Web Services and SOA)*, Αυτοέκδοση
- Θεμιστοκλέους Μ., (2010), *Σημειώσεις Μαθήματος «Υπηρεσιοστρεφείς Τεχνολογίες και Αρχιτεκτονικές»*, Κατεύθυνση Δικτυοκεντρικών Συστημάτων, Τμήμα Ψηφιακών Συστημάτων, Πανεπιστήμιο Πειραιώς
- Erl Thomas, (2004), *Service-oriented architecture: A field guide to integrating XML and Web Services*, Prentice Hall Professional Technical Reference
- Ναθαναήλ Η., (2010), *Μεταπτυχιακή Εργασία: Ανάλυση, Σχεδιασμός και Δημιουργία Ηλεκτρονικού Καταστήματος*, Τμήμα Ψηφιακών Συστημάτων, Πανεπιστήμιο Πειραιώς
- Newcomer E. Lomow G., (2005), *Understanding SOA with Web Services*, Addison-Wesley
- Κυριακούλιας Κωνσταντίνος, Ντελής Επαμεινώνδας, (2006), *Τεχνολογία Web Service*, Τμήμα Εφαρμογών Πληροφορικής Στη Διοίκηση Και Στην Οικονομία
- Δανίδου Ιωάννα (2005), *Ανάπτυξη Συστήματος Συμβουλευτικών Τουριστικών Υπηρεσιών με τη χρήση Υπηρεσιών Ιστού*, Τμήμα Πληροφορικής, Πανεπιστήμιο Κύπρου

Ηλεκτρονικές Πηγές

- <http://msdn.microsoft.com/en-us/library/aa480021.aspx>
- <http://www.w3schools.com/webservices/default.asp>
- <http://www.asp.net/mvc/tutorials/mvc-4>
- <http://www.w3.org/TR/ws-arch/>
- <http://stackoverflow.com>