

Ατομική Διπλωματική Εργασία

**Δημιουργία διαδικτυακής πλατφόρμας
για έργα του Σαίξπηρ και σχετικές μεταφράσεις τους**

Σωκράτης Γιαννακού

ΠΑΝΕΠΙΣΤΗΜΙΟ ΚΥΠΡΟΥ



ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ

Μάιος 2021

ΠΑΝΕΠΙΣΤΗΜΙΟ ΚΥΠΡΟΥ
ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ

**Δημιουργία διαδικτυακής πλατφόρμας
για έργα του Σαίξπηρ και σχετικές μεταφράσεις τους**

Σωκράτης Γιαννακού

Επιβλέπουσα Καθηγήτρια
Γεωργία Καπιτσάκη

Η Ατομική Διπλωματική Εργασία υποβλήθηκε προς μερική εκπλήρωση των απαιτήσεων απόκτησης του πτυχίου Πληροφορικής του Τμήματος Πληροφορικής του Πανεπιστημίου Κύπρου

Μάιος 2021

Ευχαριστίες

Θα ήθελα να ευχαριστήσω την επιβλέπουσα καθηγήτρια μου Δρ. Γεωργία Καπιτσάκη που μου έδωσε τη δυνατότητα να εργαστώ στο συγκεκριμένο θέμα καθώς επίσης και για την εξαιρετική συνεργασία, καθοδήγηση και άμεση ανταπόκριση σε οποιαδήποτε βοήθεια χρειάστηκα.

Θα ήθελα επίσης να ευχαριστήσω τον καθηγητή Δρ. Κωνσταντίνο Κρίτση ο οποίος ανέθεσε το θέμα της Ατομικής Διπλωματικής μου Εργασίας και παρευρισκόταν σε κάθε συνάντηση που αφορούσε την ΑΔΕ μου, βοηθώντας με να καταλάβω καλύτερα τις απαιτήσεις που προέκυψαν.

Τέλος θα ήθελα να ευχαριστήσω θερμά την οικογένεια και τους φίλους μου που ήταν πάντα δίπλα μου και με υποστήριζαν συνεχώς καθ' όλη τη διάρκεια των σπουδών μου.

Περίληψη

Σε αυτή την Ατομική Διπλωματική Εργασία αρχικά γίνεται μια γενική εισαγωγή στο κύριο θέμα της εργασίας, το οποίο είναι η υλοποίηση μιας εκσυγχρονισμένης πλατφόρμας όπου διαφυλάσσονται τα έργα του Σαίξπηρ και οι σχετικές τους μεταφράσεις. Αναφέρονται οι απαιτήσεις και οι λόγοι που οδήγησαν στην ανάπτυξη μιας τέτοιας πλατφόρμας, καθώς και οι λειτουργίες που πρέπει να παρέχονται προς τους χρήστες.

Στη συνέχεια γίνεται μια σύντομη μελέτη και ανάλυση σε ορισμένα υπάρχοντα συστήματα, με αναφορά στα μειονεκτήματά τους, που δημιούργησαν και την ανάγκη δημιουργίας ενός καλύτερου, πιο ολοκληρωμένου συστήματος. Παρουσιάζεται όμως πιο αναλυτικά ένα από αυτά, το οποίο χρησιμοποιήθηκε και ως θεμέλιο για την υλοποίηση της ζητούμενης πλατφόρμας.

Έπειτα γίνεται περιγραφή της μεθοδολογίας που ακολουθήθηκε για την δημιουργία της πλατφόρμας. Εξηγούνται οι τεχνολογίες και τα εργαλεία που χρησιμοποιήθηκαν, όπως και όλες οι τελικές λειτουργίες που προσφέρονται στους χρήστες.

Τέλος, δίνονται τα συμπεράσματα και αξιολογείται η πλατφόρμα στο κατά πόσο ανταποκρίνεται στις απαιτήσεις. Προτείνονται βελτιστοποιήσεις και ιδέες για μελλοντικές αναβαθμίσεις του συστήματος.

Περιεχόμενα

Κεφάλαιο 1	Εισαγωγή.....	1
	1.1 Γενική Εισαγωγή	1
	1.2 Περιγραφή και Στόχος της Διπλωματικής Εργασίας	2
	1.3 Δομή Διπλωματικής Εργασίας	3
Κεφάλαιο 2	Υπάρχοντα συστήματα διατήρησης έργων του Σαίξπηρ.....	5
	2.1 The Complete Works of William Shakespeare (mit.edu)	5
	2.2 No Sweat Shakespeare	6
	2.3 The Complete Works of Shakespeare	7
	2.4 LitCharts	8
	2.5 Open Source Shakespeare	9
Κεφάλαιο 3	Απαιτήσεις συστήματος	12
	3.1 Λειτουργικές Απαιτήσεις	12
	3.2 Μη Λειτουργικές Απαιτήσεις	14
Κεφάλαιο 4	Τεχνολογίες και Εργαλεία.....	16
	4.1 Επαναχρησιμοποίηση Λογισμικού	16
	4.2 Perl language	17
	4.2.1 DBI Module	17
	4.3 Web Application	17
	4.4 Front-end	18
	4.4.1 HTML	19
	4.4.2 CSS	19
	4.4.3 JavaScript	19
	4.4.4 React JS	19
	4.4.5 React Modules	20
	4.4.6 Bootstrap	21
	4.5 Back-End	21
	4.5.1 Node.js	21

4.5.2 Express.js	22
4.5.3 Node Modules	22
4.6 Βάση Δεδομένων – MySQL	23
4.7 RESTful API	24
4.7.1 Postman	24
4.8 npm	25
4.9 Component Diagram	25
Κεφάλαιο 5 Προετοιμασία Κειμένων και Βάσης Δεδομένων.....	27
5.1 Σχεδίαση Βάσης Δεδομένων	27
5.2 Οργάνωση Κειμένων	28
5.3 Parsing	30
Κεφάλαιο 6 Υλοποίηση της Web εφαρμογής Dramatica.....	37
6.1 Ρύθμιση Περιβάλλοντος	37
6.2 Υλοποίηση λειτουργιών και Συστατικών Πλατφόρμας	39
6.3 Deployment	44
6.4 Παρουσίαση Τελικού Συστήματος	45
Κεφάλαιο 7 Συμπεράσματα	50
7.1 Αξιολόγηση και Έλεγχος Πλατφόρμας	50
7.2 Αξιολόγηση Διπλωματικής Εργασίας	51
7.3 Μελλοντικές Επεκτάσεις	52
Βιβλιογραφία	53
Παράρτημα Α	56

Κεφάλαιο 1

Εισαγωγή

1.1 Γενική Εισαγωγή	1
1.2 Περιγραφή και Στόχος της Διπλωματικής Εργασίας	2
1.3 Δομή Διπλωματικής Εργασίας	3

1.1 Γενική Εισαγωγή

Πλέον ζούμε σε μια εποχή όπου η τεχνολογία συμβάλλει σε όλους τους επιστημονικούς κλάδους, ένας τέτοιος κλάδος είναι ο φιλολογικός. Πιο συγκεκριμένα ο Δρ Κωνσταντίνος Κρίτσης, μέλος του Τμήματος των Αγγλικών Σπουδών στο Πανεπιστήμιο Κύπρου, ασχολείται με την μελέτη των έργων του Σαίξπηρ και των Ελληνικών μεταφράσεων τους. Ο λόγος που ο κ. Κρίτσης στράφηκε προς την τεχνολογία και την επιστήμη της Πληροφορικής είναι για να μπορεί να εκτελεί τις εργασίες του ευκολότερα. Για παράδειγμα μια τυπική του εργασία είναι η μελέτη χειρόγραφων έργων, η σύγκριση ανάμεσα σε διάφορες μεταφράσεις των έργων από διάφορους μεταφραστές ως προς το περιεχόμενο και ως προς τη δομή. Τα έγγραφα αυτά όμως δεν είναι εύκολα προσβάσιμα, δεν είναι καλά οργανωμένα και συνήθως περιέχουν λάθη, λόγω του ότι είναι τόσο παλιά. Ως αποτέλεσμα η εργασία των ερευνητών γίνεται αρκετά δυσκολότερη και κουραστική.

Με τη βοήθεια της τεχνολογίας μπορούν να λυθούν τα πιο πάνω προβλήματα. Τα έγγραφα πλέον υπάρχουν συγκεντρωμένα σε ένα διαδικτυακό χώρο, ώστε να υπάρχει εύκολη πρόσβαση. Η πληροφορία πλέον παρουσιάζεται δομημένη και οργανωμένη. Πέραν από την παρουσίαση της πληροφορίας, αυτοματοποιούνται μερικές τυπικές εργασίες, οι οποίες ολοκληρώνονται από το σύστημα σε ελάχιστα δευτερόλεπτα και με ελάχιστα ή καθόλου σφάλματα. Αντίθετα εάν αυτές οι τυπικές εργασίες πραγματοποιούνται από ανθρώπους, πολύ

πιθανόν να υπάρχουν λάθη και σίγουρα ο χρόνος εκτέλεσης τους θα ήταν εκθετικά μεγαλύτερος. Με λίγα λόγια αφαιρείται ο ανθρώπινος παράγοντας.

1.2 Περιγραφή και Στόχος της Διπλωματικής Εργασίας

Ο Παγκόσμιος Ιστός λειτουργεί εδώ και 30 χρόνια και συνεχίζει να παρέχει αμέτρητες πληροφορίες και εργαλεία. Ο κ. Κρίτσης βρήκε πολλά συστήματα που συσχετίζονται με τη μελέτη των έργων του Σαίξπηρ. Κανένα από αυτά όμως δεν ικανοποιεί πλήρως τις ανάγκες του. Τα περισσότερα υπάρχοντα συστήματα είναι προγραμματισμένα σε παλιές πλατφόρμες, πράγμα που σημαίνει ότι δεν είναι ιδιαίτερα ελκυστικά ή εύκολα στη χρήση τους. Επιπλέον παρέχουν περιορισμένες πληροφορίες και κυρίως δεν έχουν Ελληνικές μεταφράσεις. Στο Κεφάλαιο 2 θα αναφερθώ πιο συγκεκριμένα σε αυτά τα συστήματα.

Εγώ καλούμαι να υλοποιήσω ένα σύγχρονο σύστημα που να καλύπτει όλες τις ανάγκες των ερευνητών. Αρχικά κοίταξα τα υπάρχοντα συστήματα, κατανοώντας τα προβλήματα που αντιμετωπίζουν. Έπειτα μελέτησα και κατανόησα πολλές σύγχρονες τεχνολογικές έννοιες, τις οποίες εφάρμοσα στην δημιουργία του δικού μου συστήματος. Έδωσα ιδιαίτερη σημασία στην ευχρηστία, στην ασφάλεια και στην επεκτασιμότητα. Θα εξηγήσω αυτές ιδιότητες στο Κεφάλαιο 3. Το τελικό σύστημα προσφέρει στο χρήστη εύκολη πρόσβαση σε έργα του Σαίξπηρ και σε Ελληνικές μεταφράσεις.

Στα πλαίσια της Ατομικής Διπλωματικής Εργασίας μου, ασχολήθηκα κυρίως με το έργο ‘Ρωμαίος και Ιουλιέτα’ και με τις αντίστοιχες μεταφράσεις του Βικέλα και του Μπελιέ. Οι Ελληνικές μεταφράσεις και το αυθεντικό Αγγλικό κείμενο βρίσκονται οργανωμένες σε μια βάση δεδομένων. Ο χρήστης μέσα από μια ηλεκτρονική συσκευή, συνδέεται στο διαδίκτυο και στη συνέχεια στην πλατφόρμα. Η πλατφόρμα λειτουργεί ως μέσω διπροσωπίας του χρήστη. Ο χρήστης εισάγει input, κάνει δηλαδή αιτήματα και η πλατφόρμα τα επεξεργάζεται και επικοινωνεί με τη βάση δεδομένων. Αν τα αιτήματα είναι σωστά ή μπορούν να πραγματοποιηθούν, τότε εκτελούνται και επιστρέφονται τα αποτελέσματα προς στο χρήστη. Η κύρια λειτουργία της πλατφόρμας είναι αφού γίνει η επιλογή ενός έργου από το χρήστη και μιας μετάφρασης του ίδιου έργου, να παρουσιάζονται παράλληλα τα λόγια των χαρακτήρων στις δυο εκδοχές του κειμένου. Επιπρόσθετα επιτρέπεται η παρουσίαση του αυθεντικού Αγγλικού κειμένου με δυο άλλες μεταφράσεις, ώστε ο χρήστης να έχει ταυτόχρονα τρία κείμενα στην οθόνη του. Μια άλλη πολύ σημαντική λειτουργία του συστήματος είναι μια απλή αναζήτηση. Ο χρήστης εισάγει οποιαδήποτε λέξη, στα Αγγλικά ή στα Ελληνικά και το σύστημα βρίσκει όλες τις εμφανίσεις της λέξης και τις κάνει highlight.

Μέσα στα αποτελέσματα της αναζήτησης φαίνονται και οι αντίστοιχες μεταφράσεις στα Ελληνικά ή στα Αγγλικά, δηλαδή η αντίθετη γλώσσα από την οποία έγινε η αναζήτηση, ώστε να φαίνεται το χωρίο που περιέχει την μεταφρασμένη λέξη που αναζητήθηκε. Παράλληλα ο χρήστης έχει πρόσβαση σε μια πιο εξελιγμένη αναζήτηση, το “Advanced Search”, όπου μπορεί να χρησιμοποιήσει φίλτρα για την αναζήτηση του. Για παράδειγμα το όνομα της μετάφρασης, το όνομα του χαρακτήρα που λέει τις γραμμές στις οποίες περιέχεται η λέξη που αναζητήθηκε, ο αριθμός Σκηνής και ο αριθμός Πράξης. Τέλος υπάρχει η δυνατότητα μεταφόρτωσης ενός αρχείου από την πλευρά του χρήστη προς τον server. Ο διαχειριστής του συστήματος λαμβάνει το αρχείο και κρίνει αν είναι χρήσιμο προς τη βελτίωση του συστήματος, για παράδειγμα μια μετάφραση του Ρωμαίου και της Ιουλίετας από έναν άλλο μεταφραστή που δεν υπάρχει στη βάση. Τότε μπορεί να γίνει επεξεργασία του κειμένου αυτού και να εμπλουτιστεί το διαθέσιμο υλικό της πλατφόρμας.

1.3 Δομή Διπλωματικής Εργασίας

Στο παρόν κεφάλαιο υπάρχει ένα γενικό υπόβαθρο στο θέμα, ο στόχος της ΑΔΕ και η περιγραφή του τελικού συστήματος που υλοποιήθηκε.

Κεφάλαιο 2:

Στο δεύτερο κεφάλαιο γίνεται μια μελέτη στα υπάρχοντα συστήματα που παρέχουν πληροφορίες των έργων του Σαίξπηρ και παρουσιάζονται τα προβλήματα τους.

Κεφάλαιο 3:

Σε αυτό το κεφάλαιο καταγράφονται οι ανάγκες του συστήματος. Πρόκειται για τις λειτουργικές απαιτήσεις, για το τι θα κάνει το σύστημα και για τις μη λειτουργικές απαιτήσεις, δηλαδή για τυχόν περιορισμούς ή μετρικές που πρέπει να ληφθούν υπόψη.

Κεφάλαιο 4:

Στο τέταρτο κεφάλαιο παρουσιάζονται συνοπτικά όλα τα εργαλεία, οι βιβλιοθήκες και οι έννοιες που συνείσφεραν στην τελική ανάπτυξη του συστήματος.

Κεφάλαιο 5:

Έπειτα γίνεται αναφορά στη μεθοδολογία που εφαρμόστηκε για την υλοποίηση του συστήματος. Εξηγείται η διαδικασία προετοιμασίας των κειμένων και η δημιουργία της Βάσης Δεδομένων.

Κεφάλαιο 6:

Στη συνέχεια γίνεται μια επίδειξη της πλατφόρμας και των λειτουργιών της. Προβάλλονται τα συστατικά, οι τελικές οθόνες και το πως έγινε η υλοποίηση του front-end, του back-end και το deployment.

Κεφάλαιο 7:

Τέλος γίνεται μια σύντομη αξιολόγηση του συστήματος, εκτίθενται πιθανά προβλήματα και προτείνονται μελλοντικές επεκτάσεις.

Κεφάλαιο 2

Υπάρχοντα συστήματα διατήρησης έργων του Σαίξπηρ

2.1 The Complete Works of William Shakespeare (mit.edu)	5
2.2 No Sweat Shakespeare	6
2.3 The Complete Works of Shakespeare	7
2.4 LitCharts	8
2.5 Open Source Shakespeare	9

Για την προετοιμασία αυτού του κεφαλαίου αναζήτησα άλλα, υπάρχοντα συστήματα διατήρησης έργων του Σαίξπηρ στον παγκόσμιο ιστό. Πήρα τα 5 πρώτα αποτελέσματα αναζήτησης στο Google και έλεγξα το κατά πόσο ανταποκρίνονται στις απαιτήσεις μας.

2.1 The Complete Works of William Shakespeare (mit.edu)

Η πρώτη μου επίσκεψη ήταν στο The Complete Works of William Shakespeare (mit.edu). Αυτός ο ιστότοπος λειτουργεί από το 1993 και παρέχει μια ολοκληρωμένη συλλογή με όλα τα έργα του Σαίξπηρ [1].


The Complete Works of William Shakespeare

Welcome to the Web's first edition of the Complete Works of William Shakespeare. This site has offered Shakespeare's plays and poetry to the Internet community since 1993.

For other Shakespeare resources, visit the [Mr. William Shakespeare and the Internet](#) Web site.

The original electronic source for this server was the Complete Moby(tm) Shakespeare. The HTML versions of the plays provided here are placed in the public domain.

[Older news items](#)



Comedy	History	Tragedy	Poetry
All's Well That Ends Well	Henry IV, part 1	Antony and Cleopatra	The Sonnets
As You Like It	Henry IV, part 2	Coriolanus	A Lover's Complaint
The Comedy of Errors	Henry V	Hamlet	The Rape of Lucrece
Cymbeline	Henry VI, part 1	Julius Caesar	Romeo and Juliet
Love's Labour's Lost	Henry VI, part 2	King Lear	Funeral Elegy for W.S.
Measure for Measure	Henry VI, part 3	Macbeth	
The Merry Wives of Windsor	Henry VIII	Othello	
The Merchant of Venice	King John	Romeo and Juliet	
A Midsummer Night's Dream	Richard II	Timon of Athens	
Much Ado About Nothing	Richard III	Titus Andronicus	
Pericles, Prince of Tyre			
Taming of the Shrew			
The Tempest			
Troilus and Cressida			
Twelfth Night			
...			

Σχήμα 2.1 Η ιστοσελίδα The Complete Works of William Shakespeare – αρχική

Romeo and Juliet
[Shakespeare homepage](#) | [Romeo and Juliet](#) | Act 1, Scene 1
[Previous scene](#) | [Next scene](#)

SCENE I. Verona. A public place.

Enter SAMPSON and GREGORY, of the house of Capulet, armed with swords and bucklers

SAMPSON

Gregory, o' my word, we'll not carry coals.

GREGORY

No, for then we should be colliers.

SAMPSON

I mean, an we be in choler, we'll draw.

GREGORY

Ay, while you live, draw your neck out o' the collar.

SAMPSON

I strike quickly, being moved.

GREGORY

But thou art not quickly moved to strike.

SAMPSON

A dog of the house of Montague moves me.

GREGORY

To move is to stir; and to be valiant is to stand:
therefore, if thou art moved, thou runn'st away.

Σχήμα 2.2 Η ιστοσελίδα The Complete Works of William Shakespeare – κείμενο

Πρόκειται για μια λιτή, εύκολη στη χρήση ιστοσελίδα, αλλά από την πρώτη ματιά φαίνεται το πόσο παλιά είναι.

Προβλήματα

- Η ανάγνωση δεν είναι ευχάριστη
- Το υλικό είναι περιορισμένο
- Τα κείμενα χρειάζονται αρκετό χρόνο για να φορτωθούν

2.2 No Sweat Shakespeare

Η No Sweat Shakespeare είναι μια πιο σύγχρονη πλατφόρμα, με εμπλουτισμένο υλικό. Τα κείμενα φορτώνονται σχετικά γρήγορα. Περιέχονται πληροφορίες για τα έργα, τους χαρακτήρες, για διάσημες φράσεις και επίσης υπάρχουν σύνδεσμοι για επιπλέον υλικό [2]. Παρουσιαστικά υπάρχει μια βελτίωση σε σχέση με την προηγούμενη ιστοσελίδα και το υλικό είναι πολύ περισσότερο.

The screenshot shows the 'No Sweat Shakespeare' website. The header is teal with the site's logo and social media icons. A navigation bar below the header contains links: Home, Resources, Plays, Sonnets, Quotes, Stratford, ebooks, Blog, Shop, and a search icon. The main content area is titled 'All's Well That Ends Well Text: Act 1, Scene 1'. A paragraph explains that the text is long and split into scenes per page. Below this, the text of Act 1, Scene 1 is displayed, starting with 'Enter BERTRAM, the COUNTESS of Rousillon, HELENA, and LAFEU, all in black'. A sidebar on the right lists various links, including 'All's Well That Ends Well Setting', 'All's Well That Ends Well Script', and a list of links for each act and scene of the play.

Σχήμα 2.3 Η ιστοσελίδα No Sweat Shakespeare

Προβλήματα

- Εξακολουθεί όμως να μην μας ικανοποιεί αφού δεν υπάρχουν μεταφράσεις των έργων σε άλλες γλώσσες.
- Η αναζήτηση λέξεων μέσα σε κείμενα δεν είναι άμεση και χρειάζεται ένα αρκετά μεγάλο χρονικό διάστημα για να ολοκληρωθεί.
- Πολλές περιττές πληροφορίες σε πολλά σημεία της σελίδας, π.χ. διαφημίσεις και προώθηση του κατόχου της πλατφόρμας.

2.3 The Complete Works of Shakespeare

Το The Complete Works of Shakespeare έχει ένα απλό και μινιμαλιστικό design. Περιέχει απλά τα έργα του Σαίξπηρ και τίποτα περισσότερο [3]. Επίσης φαίνεται ότι είναι μια από τις πιο σύγχρονες ιστοσελίδες. Η απόδοση είναι καλή, αν και το μόνο κριτήριο που μπορεί να ορίσει κάποιος είναι η ταχύτητα φόρτωσης των κειμένων.

Romeo and Juliet

THE FULL TEXT
BY WILLIAM SHAKESPEARE

ACT I

PROLOGUE

Two households, both alike in dignity,
In fair Verona, where we lay our scene,
From ancient grudge break to new mutiny,
Where civil blood makes civil hands unclean.
From forth the fatal loins of these two foes

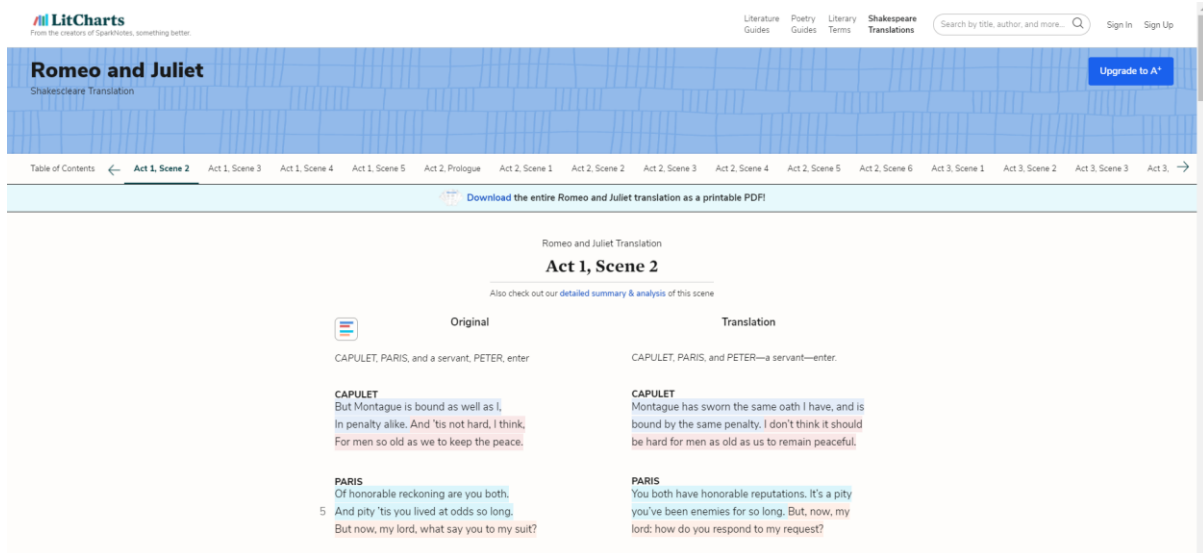
Σχήμα 2.4 Η ιστοσελίδα The Complete Works of Shakespeare

Προβλήματα

- Όχι μεταφράσεις
- Φτωχό περιεχόμενο και διαθέσιμες λειτουργίες
- Κατά την ανάγνωση των κειμένων δεν υπάρχει πλοήγηση στις Πράξεις και τις Σκηνές του έργου. Ο χρήστης έχει απέναντί του ένα μεγάλο αρχείο κειμένου και θα πρέπει ο ίδιος να βρει το σημείο που θέλει να διαβάσει κάνοντας scroll.

2.4 LitCharts

Μια ακόμη πλατφόρμα είναι η LitCharts. Πρόκειται για μια μεγάλη online βιβλιοθήκη που το υλικό της είναι κυρίως οδηγοί – guides για θεατρικά έργα, ποιήματα, λογοτεχνικά βιβλία πολλών συγγραφέων [4]. Οι οδηγοί είναι μια ωραία προσθήκη. Περιγράφονται τα έργα, οι χαρακτήρες, οι θεματικές ενότητες και περιέχεται και μια περίληψη του έργου. Όσον αφορά τα έργα του Σαίξπηρ υπάρχουν ολόκληρα τα έργα και χωρίζονται σωστά σε τμήματα. Κάτι ενδιαφέρον που δεν είδαμε μέχρι τώρα σε άλλες πλατφόρμες είναι η παρουσίαση των αυθεντικών έργων στα Early Modern English δίπλα από την μετάφραση στη σύγχρονη Αγγλική γλώσσα. Έχουμε δηλαδή τα έργα στην γλώσσα που έγραψε ο Σαίξπηρ γύρω στο 1600 μ.Χ.



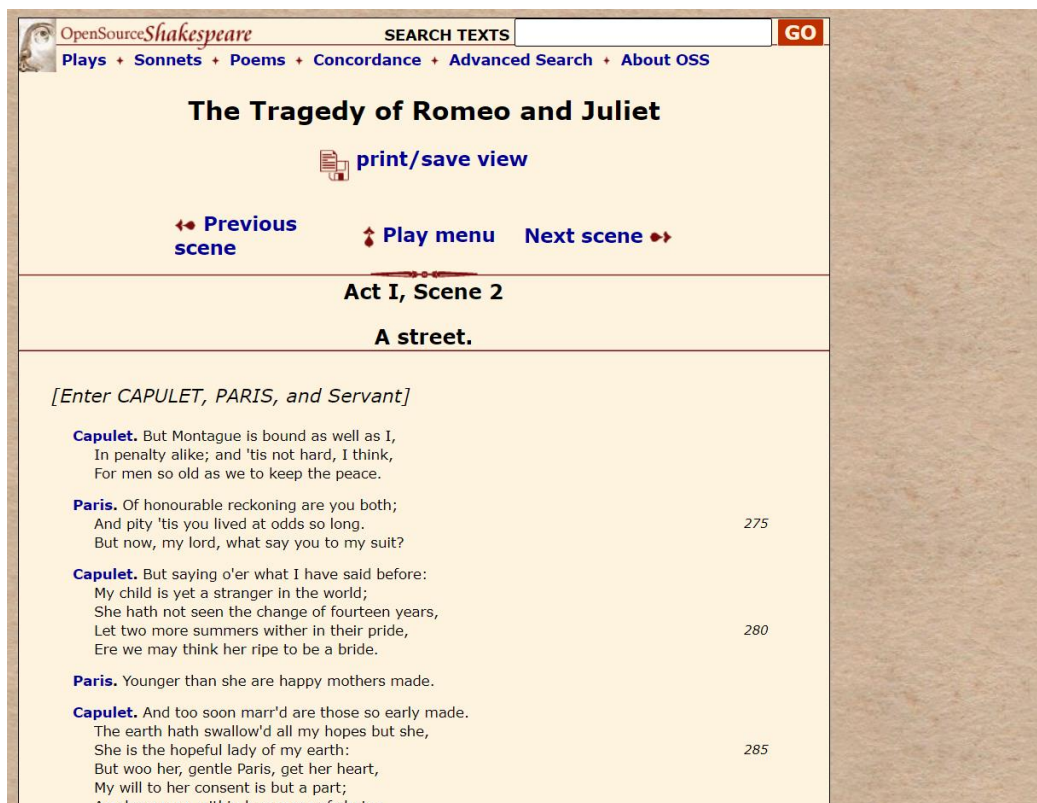
Σχήμα 2.5 Η ιστοσελίδα LitCharts

Προβλήματα

- Όχι Ελληνικές μεταφράσεις
- Πρόσεξα ένα bug όπου κατά την αναζήτηση ενός έργου αρχικά τυπώνεται μήνυμα ότι δεν υπάρχει το συγκεκριμένο έργο, αλλά μετά από λίγο φορτώνει αποτελέσματα.
- Καθώς πλοηγούσα την ιστοσελίδα, σε διάφορες χρονικές στιγμές εμφανιζόταν στην οθόνη φόρμα εγγραφής και σταματούσε οποιαδήποτε ενέργεια προσπαθούσα να κάνω. Για εμένα αυτός είναι κακός σχεδιασμός ιστοσελίδας και επηρεάζεται αρνητικά η εμπειρία του χρήστη.

2.5 Open Source Shakespeare

Η τρίτη και τελευταία πλατφόρμα που θα δούμε είναι το Open Source Shakespeare (OSS). Αν και έχουμε ακόμη μια παλιά ιστοσελίδα, η συγκεκριμένη παρέχει πολλές, χρήσιμες πληροφορίες, πέραν των αυθεντικών κειμένων. Υπάρχει μια ισχυρή αναζήτηση που μας δίνει αποτελέσματα σε ικανοποιητικούς χρόνους. Παράλληλα ο χρήστης μπορεί να εκτελέσει τις εργασίες του γρήγορα και εύκολα [5].




Σχήμα 2.6 Η ιστοσελίδα OpenSource Shakespeare - κείμενο

Προβλήματα

- Πάλι, δεν υπάρχουν μεταφράσεις σε άλλες γλώσσες.
- Θα μπορούσε να βελτιωθεί η ταχύτητα αναζήτησης έστω και λίγο

Η OSS όμως έχει και ένα άλλο εξαιρετικό θετικό σημείο. Είναι open source [6], δηλαδή παρέχει τον κώδικα με τον οποίο είναι υλοποιημένη προς τους χρήστες, οι οποίοι μπορούν να τον δουν, να τον επεξεργαστούν ή να τον αλλάξουν. Ο δημιουργός της OSS εξηγεί λεπτομερώς την διαδικασία υλοποίησης της και έχει ανεβάσει ολόκληρη τη βάση δεδομένων που περιέχει όλα τα έργα, τους χαρακτήρες και άλλα πολλά. Όπως και πολλοί άλλοι προγραμματιστές που προσδιορίζουν τα συστήματά τους ως open source, ο δημιουργός της OSS αναφέρει συγκεκριμένα ότι δεν επιτρέπεται η χρήση του κώδικα του με κερδοφόρο σκοπό. Θα ήθελε να διατηρηθεί το σύστημα του ως μια δωρεάν, μη κερδοφόρα πηγή πληροφόρησης για τους ερευνητές και τους απλούς θαυμαστές του Σαίξπηρ. Για την υλοποίηση του συστήματος για την Ατομική μου Διπλωματική Εργασία, μεταφόρτωσα και επεξεργάστηκα τον κώδικα της βάσης δεδομένων, τον κώδικα του parser (του οποίου η λειτουργία θα εξηγηθεί στη συνέχεια) και εμπνεύστηκα από τον τρόπο που παρουσιάζει τα κείμενα και εκτελεί την αναζήτηση.


OpenSource*Shakespeare*

SEARCH TEXTS
GO

[Plays](#) + [Sonnets](#) + [Poems](#) + [Concordance](#) + [Advanced Search](#) + [About OSS](#)

APPENDIX B: Marked-up play text
Table of Contents

APPENDIX C: Parser source code

```
#####
# Shakespeare text parser
#####
# Eric M. Johnson
# July 12, 2003
#
# January 30, 2004: modified to use new database schema
#
# "Sections" = Acts
# "Chapters" = Scenes
#####

# begin timing the script
$begin_time = time();

#####
# subroutine to add lines to database
#####

sub linewrite {
    $writepara = $_[0];
    $writeparanum = $_[1];
    $writeparatype = $_[2];
    $writeparasection = $_[3];
    $writeparachapter = $_[4];
```

Σχήμα 2.7 Η ιστοσελίδα OSS – Ανοικτός κώδικας

Κεφάλαιο 3

Απαιτήσεις συστήματος

3.1 Λειτουργικές Απαιτήσεις	12
3.2 Μη Λειτουργικές Απαιτήσεις	14

3.1 Λειτουργικές Απαιτήσεις

Στην αρχή του κύκλου ζωής του κάθε λογισμικού υπάρχει η διαδικασία εξαγωγής απαιτήσεων [7]. Οι λειτουργικές απαιτήσεις είναι οι υπηρεσίες που θα πρέπει να παρέχει το σύστημα. Στην περίπτωση της ΑΔΕ μου, οι απαιτήσεις μου δόθηκαν σε γραπτή μορφή και ήταν οι πιο κάτω:

1. Κυριότερος στόχος του συστήματος είναι η παρουσίαση όλων των διαθέσιμων, “open source” έργων του Σαίξπηρ. Οι χρήστες θα μπορούν να διαβάσουν όλα τα έργα του Σαίξπηρ μαζεμένα σε ένα ιστότοπο. Τα έργα θα υπάρχουν στην αρχέτυπη τους μορφή, δηλαδή στην Αγγλική γλώσσα. Επιπλέον η πλατφόρμα θα παρέχει μια ή περισσότερες μεταφράσεις για μερικά από τα έργα στην Ελληνική γλώσσα. Σε ένα μεταγενέστερο στάδιο ίσως προστεθούν μεταφράσεις και σε άλλες γλώσσες.
2. Εξαιρετικά σημαντική είναι η παράλληλη παρουσίαση των Αγγλικών πρωτότυπων έργων με τις διαθέσιμες Ελληνικές μεταφράσεις. Έλληνες ερευνητές των έργων του Σαίξπηρ εξέφρασαν την ανάγκη για μια εύχρηστη πλατφόρμα, η οποία να τους παρέχει τα έργα μαζί με τις μεταφράσεις στοιχισμένα σε ένα χώρο. Αυτή η λειτουργία θα τους διευκόλυνε πολύ στην μελέτη και στη σύγκριση των έργων με τις αντίστοιχες μεταφράσεις ή στη σύγκριση των μεταφράσεων. Δεν υπάρχει κάποια γνωστή πλατφόρμα στο διαδίκτυο που να καλύπτει αυτή την απαίτηση, γεγονός που καθιστά το σύστημα πρωτοπόρο.

3. Πέραν από τη μια προς μια παρουσίαση των κειμένων. Αγγλικού – Ελληνικού, θα δίνεται η δυνατότητα παρουσίασης του Αγγλικού μαζί με 2 ή ακόμα και περισσότερες Ελληνικές μεταφράσεις. Θα πρέπει ωστόσο να διατηρηθεί η όμορφη και ευανάγνωστη παρουσίαση των κειμένων, άρα θα πρέπει να περιοριστεί ο αριθμός των κειμένων που θα φαίνονται στην οθόνη κάθε φορά. Κατά τη γνώμη μου δεν πρέπει να ξεπερνά τα τρία κείμενα, δηλαδή το αυθεντικό μαζί με δύο άλλες μεταφράσεις.
4. Ο χρήστης θα μπορεί να κάνει αναζήτηση σε όλα τα έργα του Σαίξπηρ, όπως επίσης και σε όλες τις μεταφράσεις. Θα υπάρχει δυνατότητα αναζήτησης οποιασδήποτε λέξης ή φράσης, εάν γίνει η εισαγωγή τους χωρίς κανένα σφάλμα. Θα παρουσιάζονται όλες οι εμφανίσεις τις λέξης. Θα εμφανίζονται με αλφαβητική σειρά τα έργα που τις περιέχουν, τον μεταφραστή του εκδοχής του έργου, τον χαρακτήρα που είπε τη λέξη, τον αριθμό Πράξης, τον αριθμός Σκηνής και τέλος το απόσπασμα στο οποίο βρίσκεται η λέξη, με τη λέξη να είναι highlighted. Θα μπορούν να αναζητούνται λέξεις στα Ελληνικά ή και στα Αγγλικά.
5. Επιπρόσθετα θα υπάρχει μια συνδυαστική αναζήτηση, όπου ο χρήστης θα μπορεί να εισάγει το όνομα του έργου, του μεταφραστή και το όνομα του χαρακτήρα, τους αριθμούς Πράξης και Σκηνής, έτσι ώστε να περιορίσει την αναζήτηση και να βάλει κάποια όρια που ο ίδιος επιθυμεί. Ξανά η αναζήτηση μπορεί να γίνει και στις δύο γλώσσες.
6. Ο χρήστης θα μπορεί να ανεβάζει επιπλέον μεταφράσεις στην πλατφόρμα. Αρχικά θα μεταφορτώσει το αρχείο της μετάφρασης, το οποίο θα περάσει από τη διαδικασία του parsing και έπειτα θα τοποθετηθεί στη βάση δεδομένων, για να μπορεί στο μέλλον να είναι προσβάσιμο από τους χρήστες. Η πιο πάνω απαίτηση θα πρέπει να εξακριβωθεί με ιδιαίτερη προσοχή, καθώς η πρόσβαση ενός χρήστη (ακόμα και έμπιστου) στη βάση δεδομένων του συστήματος, είναι αρκετά επικίνδυνη. Το πιο πιθανό είναι ότι θα υπάρχει κάποιος περιορισμός κατά την μεταφόρτωση του αρχείου και πριν την επεξεργασία του κειμένου θα γίνεται ένας γρήγορος έλεγχος από το διαχειριστή του συστήματος.
7. Σε ορισμένα έργα και μεταφράσεις θα υπάρχει σύνδεσμος (link) με θεατρικές αναπαραστάσεις. Θα υπάρχουν πληροφορίες για την παραγωγή, όπως η ημερομηνία της εκτέλεσης, το όνομα του σκηνοθέτη και τα ονόματα των ηθοποιών. Πιθανόν να

υπάρχει σύνδεσμος σε τυχόν κρητικές στοχευμένες στις παραστάσεις, ως επίσης και σύνδεσμος σε ηχητικά αποσπάσματα ή βίντεο. Αυτή η απαίτηση δεν ολοκληρώθηκε στα πλαίσια αυτής της ΑΔΕ, επειδή θα χρειαζόντουσαν περισσότεροι πόροι.

3.2 Μη Λειτουργικές Απαιτήσεις

Οι μη λειτουργικές απαιτήσεις αφορούν τις ιδιότητες και τους τεχνικούς περιορισμούς του συστήματος. Το σύστημα για να γίνει αποδεκτό θα πρέπει να τις πληρεί.

1. Το σύστημα δεν μπορεί να χρησιμοποιηθεί ως μέσο κερδοφορίας. Όπως προανέφερα αυτή είναι μια απαραίτητη ιδιότητα που θα έχει το σύστημα, σε καμία περίπτωση δεν μπορεί να πουληθεί, για παράδειγμα ως υπηρεσία, αφού είναι ενάντια στις αρχές που έθεσε ο κάτοχος ενός μέρους του κώδικα.
2. Usability - Ευχρηστία: Μια εξαιρετικά σημαντική ιδιότητα που ορίζεται στο ISO9241 ως η ιδιότητα ενός συστήματος, συσκευής ή λογισμικού που επιτρέπει στο χρήστη να εκτελεί κάποια εργασία αποδοτικά, αποτελεσματικά και να του δίνει μια υποκειμενική ευχαρίστηση [8]. Ο χρήστης πρέπει με την είσοδο του στην πλατφόρμα να γνωρίζει τι πρέπει να κάνει, να καθοδηγείται και να μπορεί να εκτελεί οποιαδήποτε εργασία εύκολα. Αυτό μπορεί να επιτευχθεί με τη χρήση σωστών μενού, κουμπιών, φορμών και άλλων συστατικών της πλατφόρμας, δηλαδή συστατικά τα οποία είναι ορατά, με χρώματα και μέγεθος που να ταιριάζουν. Επιπλέον μετά από κάθε πράξη του χρήστη κατά την οποία δεν πήρε αποτελέσματα, να παίρνει μηνύματα λάθους ώστε να μπορεί να διορθώσει ή να αλλάξει την τυχόν λανθασμένη επιλογή που έχει κάνει.
3. Security – Ασφάλεια: Πρέπει να παρέχεται προστασία προς το σύστημα από επιθέσεις κακόβουλων χρηστών. Μια τέτοια επίθεση είναι τα SQL injections [9], η εισαγωγή malicious κώδικα ως είσοδο στο σύστημα, οποίος μεταφράζεται ως κώδικας SQL για τη βάση δεδομένων. Ως αποτέλεσμα μπορούν να διαγραφτούν πίνακες, να αλλάξουν δεδομένα ή να χαθεί ολόκληρη η βάση. Ένας καλός τρόπος αντιμετώπισης αυτών των επιθέσεων είναι η χρήση των wildcard characters (?). Αυτό το Αγγλικό ερωτηματικό μπαίνει στη θέση μιας μεταβλητής εισόδου για ένα query. Με λίγα λόγια η είσοδος του χρήστη δεν μπαίνει απευθείας μέσα στο query το οποίο κατευθύνεται ύστερα στη βάση, αλλά αποθηκεύεται τοπικά σε μια νέα μεταβλητή που αρχικά έχει ως placeholder το σύμβολο ?.

4. High Availability – υψηλή Διαθεσιμότητα: Ιδανικά θέλουμε το σύστημα να είναι ενεργό συνεχώς, εκτός στις χρονικές περιόδους όπου θα γίνεται κάποιο είδος αναβάθμισης. Επίσης να μπορεί να δέχεται ένα μεγάλο αριθμό ταυτόχρονων συνδεδεμένων χρηστών, έτσι ώστε να οι χρήστες να έχουν εύκολη πρόσβαση οποιαδήποτε στιγμή επιθυμούν.
5. Scalability – Επεκτασιμότητα: Όπως θα αναφέρω και στο τελευταίο κεφάλαιο, υπάρχει πολύς χώρος για βελτίωση. Θα πρέπει από τα πρώτα στάδια υλοποίησης του συστήματος να γραφτεί κώδικας ο οποίος να μπορεί επεκταθεί, να επαναχρησιμοποιηθεί και να μην προκαλούνται ανεπιθύμητα αποτελέσματα σε άλλα μέρη. Για παράδειγμα, οι λίστες dropdown να είναι φτιαγμένες με τέτοιο τρόπο ώστε με την εισαγωγή νέου στοιχείου να το εμφανίζει, χωρίς να πρέπει να γίνει κάποια αλλαγή στον κώδικα.
6. Performance – Απόδοση: Ένα ισχυρό σύστημα είναι πάντοτε προτιμότερο παρά ένα πιο αδύναμο αντίπαλο σύστημα. Ο αριθμός πολλών ενεργών χρηστών δεν πρέπει να επηρεάζει την απόδοση απέναντι στον κάθε ένα από τους χρήστες. Ο χρόνος απόκρισης πρέπει να είναι ελάχιστος για να μπορεί να εκτελείται η κάθε εργασία γρήγορα. Αυτό μπορεί να επιτευχθεί με τη χρήση έξυπνων αλγόριθμων και γενικά έξυπνη τοποθέτηση κώδικα στο σύστημα. Ένα παράδειγμα είναι η αποφυγή πολλών αχρείαστων αναδρομικών κλήσεων.
7. Error prevention – Αποτροπή λαθών: Ο χρήστης όσο γίνεται θα πρέπει να αποτρέπεται από το να κάνει λάθη, όπως για παράδειγμα να καταλήγει σε λάθος συνδέσμους, να παγώνει η οθόνη, η να μην φορτώνουν κάποια components της πλατφόρμας. Αυτά τα προβλήματα λύνονται μέσα από τη σωστή εγγραφή του κώδικα. Μπορούμε να περιορίσουμε όλες του τις κινήσεις για παράδειγμα να μην μπορεί να πάει σε ένα σύνδεσμο χωρίς να περάσει από έναν άλλο σύνδεσμο ή να πρέπει να πατήσει ένα κουμπί πρώτα.
8. Portability – Φορητότητα: Η ιστοσελίδα θα μπορεί να είναι προσβάσιμη από όλους τους συνήθεις φυλλομετρητές, όπως τους Microsoft Edge, Google Chrome, Mozilla Firefox αλλά και από διάφορες συσκευές εκτός από υπολογιστές, όπως smartphones και tablets.

Κεφάλαιο 4

Τεχνολογίες και Εργαλεία

4.1 Επαναχρησιμοποίηση Λογισμικού	16
4.2 Perl language	17
4.2.1 DBI Module	17
4.3 Web Application	17
4.4 Front-End	18
4.4.1 HTML	19
4.4.2 CSS	19
4.4.3 JavaScript	19
4.4.4 React JS	19
4.4.5 React Modules	20
4.4.6 Bootstrap	21
4.5 Back-End	21
4.5.1 Node.js	21
4.5.2 Express.js	22
4.5.3 Node Modules	22
4.6 Βάση Δεδομένων -MySQL	23
4.7 RESTful API	24
4.7.1 Postman	24
4.8 npm	25
4.9 Component Diagram	25

4.1 Επαναχρησιμοποίηση Λογισμικού

Όπως είχα αναφέρει προηγουμένως, ήθελα από την αρχή να ακολουθήσω μια προσέγγιση επαναχρησιμοποίησης λογισμικού [10]. Αρχικά προσπάθησα να καταλάβω τη μεθοδολογία που εφάρμοσε ο δημιουργός του OSS και στη συνέχεια κατέβασα τον Ανοικτό Κώδικα για

την βάση δεδομένων που περιέχει τους πίνακες και τα δεδομένα. Διέγραψα ορισμένους πίνακες, άλλους τους επεξεργάστηκα και δημιούργησα και πολλούς καινούργιους. Εκτός από τον κώδικα της βάσης, στο OSS υπάρχει και ένας αρκετά παλιός κώδικας στη γλώσσα Perl, ο οποίος ασχολείται με το Parsing του κειμένου, μια πολύ βασική διαδικασία μαρκαρίσματος του κειμένου, για να μπορεί να διασπαστεί σε διάφορα μέρη και να τοποθετηθεί στη βάση δεδομένων σε πολλαπλούς πίνακες. Θα επανέλθω στο parsing με μια πιο αναλυτική εξήγηση στο επόμενο κεφάλαιο. Ο συγκεκριμένος κώδικας 15 χρόνια μετά την δημιουργία του δεν λειτουργεί και χρειάστηκε να τον αλλάξω. Αντικατέστησα παλιά modules με πιο σύγχρονα και τον προσαρμόσα στα δικά μου δεδομένα και κυρίως για να δουλεύει με τα ελληνικά κείμενα.

4.1 Perl language

Η γλώσσα προγραμματισμού Perl είναι μια από τις παλαιότερες γλώσσες προγραμματισμού, η οποία σήμερα χρησιμοποιείται πολύ σπάνια σε σύγχρονα συστήματα. Μετά από έρευνα κατέληξα ότι τελικά θα διατηρήσω τον κώδικα σε αυτή την παλιά γλώσσα. Πρώτα, επειδή ήθελα να παραμείνω στην προσέγγιση της επαναχρησιμοποίησης λογισμικού. Επίσης όμως για το ότι η Perl υπερέχει στην επεξεργασία κειμένου, αυτό ακριβώς που θέλω να πετύχω με το parsing [11].

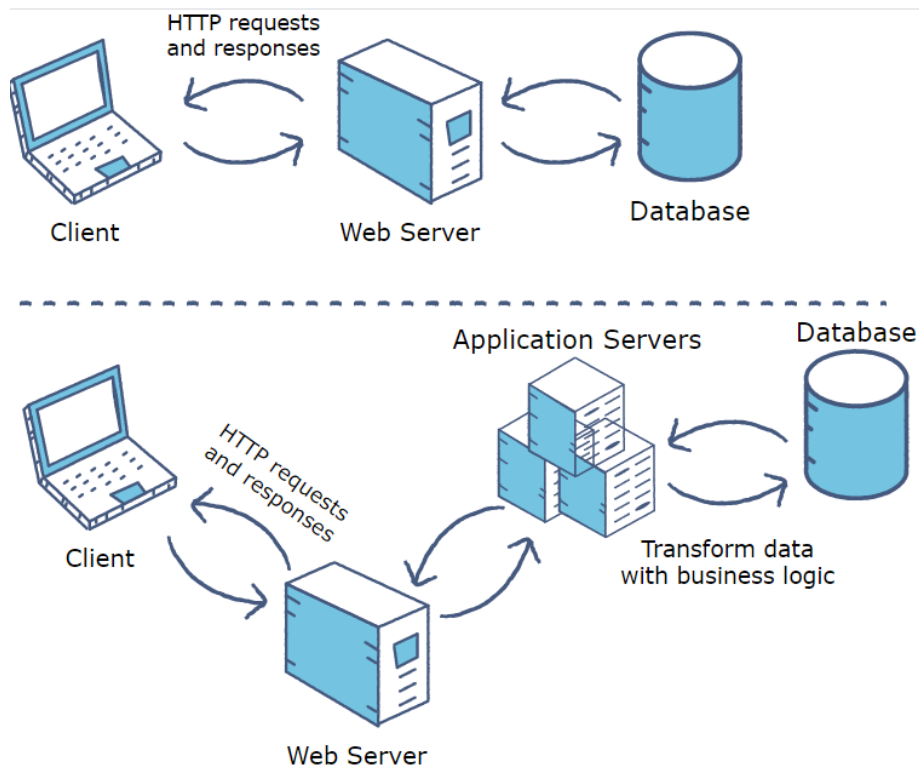
4.2.1 DBI Module

Όπως είπα πιο πάνω για λειτουργήσει ο κώδικας της Perl έπρεπε να κατεβάσω την τελευταία της έκδοση και να κάνω αλλαγές στον κώδικα. Μια αλλαγή είναι η εγκατάσταση της βιβλιοθήκης DBI (Database Independent Interface), η οποία προσφέρει επικοινωνία μεταξύ του κώδικα της Perl με την βάση δεδομένων, δηλαδή γίνεται σύνδεση και επιτρέπει την εκτέλεση queries.

4.3 Web Application

Η πλατφόρμα που υλοποιήθηκε έχει μορφή Web Application, είναι ένα λογισμικό αρχιτεκτονικής client-server, το οποίο εκτελεί λειτουργίες μέσα από το Διαδίκτυο. Ο χρήστης χρησιμοποιώντας ένα web browser επικοινωνεί με τη διεπιφάνεια του Web Application και στέλνει αιτήματα προς τον web server. Ο web server με τη σειρά του προωθεί το αίτημα στον web application server, ο οποίος έχει περισσότερες δυνατότητες. Ο web application server μπορεί να εκτελέσει το αίτημα, να το επεξεργαστεί τα δεδομένα και

να επικοινωνήσει με τη βάση δεδομένων. Αφού γίνουν οι κατάλληλες εργασίες, ο web application server επιστρέφει τα αποτελέσματα στον web server που απαντά πίσω στον browser του χρήστη με τα ζητούμενα [12]. Οι διαδικτυακές εφαρμογές σε αντίθεση με τις ιστοσελίδες προσφέρουν περισσότερη διαδραστικότητα στους χρήστες, καθώς ένα μεγάλο μέρος των υπολογισμών γίνεται από τα browsers, στην πλευρά του client. Για την κατασκευή των Web Applications χρησιμοποιούνται πολύ συχνά διάφορα frameworks, λογισμικά που αποτελούνται από πολλά συστατικά και διευκολύνουν την ανάπτυξη λογισμικού. Συνήθως φροντίζουν τις λειτουργικότητες χαμηλού επιπέδου.



Σχήμα 4.1 Web Site (πάνω) και Web Application (κάτω) [13]

4.4 Front-End

Όταν μιλούμε για front-end development μιλούμε για την ανάπτυξη λογισμικού που αφορά το τι βλέπει ο χρήστης και με το τι μπορεί να αντιδράσει άμεσα, πχ τα κουμπιά, τις φόρμες και τις εικόνες. Αποτελείται κυρίως από το συνδυασμό τεχνολογιών, όπως HTML, client-side JavaScript, CSS.

4.4.1 HTML

HTML ή αλλιώς HyperText Markup Language είναι η πιο γνωστή γλώσσα για markup που χρησιμοποιείται στον Παγκόσμιο Ιστό. Είναι υπεύθυνη για την παρουσίαση δομημένης πληροφορίας και διαβάζεται από τους φυλλομετρητές για να παρουσιάσουν τα διάφορα στοιχεία μιας σελίδας [14]. Μερικά στοιχεία της HTML είναι το , για εικόνες, το <p> για παραγράφους και το <a> για υπερσυνδέσμους.

4.4.2 CSS

Σε συνδυασμό με την HTML χρησιμοποιούνται πάντοτε τα Cascading Style Sheets [15], γλώσσα περιγραφής και μορφοποίησης των στοιχείων της HTML, ώστε να είναι πιο όμορφα ή πιο κατανοητά προς τους χρήστες. Για παράδειγμα η στοίχιση ενός κειμένου, η δημιουργία περιγράμματος σε μια εικόνα ή το χρωμάτισμα ενός κουμπιού.

4.4.3 JavaScript

Η JavaScript είναι η πιο δημοφιλής scripting γλώσσα για ιστοσελίδες και για διαδικτυακές εφαρμογές. Όταν χρησιμοποιείται στο front-end, μπορεί να δώσει λειτουργικότητα σε κάθε στοιχείο της σελίδας. Μπορεί να δημιουργήσει ή να αναβαθμίσει περιεχόμενο δυναμικά, παίρνοντας για παράδειγμα τις εισόδους του χρήστη. Χτισμένα πάνω στην JavaScript είναι ισχυρά APIs (Application Programming Interfaces), βιβλιοθήκες κώδικα που είναι έτοιμος στη χρήση και δίνουν στον προγραμματιστή τη δυνατότητα υλοποίησης έτοιμων λειτουργιών, που διαφορετικά θα χρειάζονταν πολύ χρόνο να αναπτυχθούν [16]. Ένα παραδείγματα χρήσης JavaScript είναι η εμφάνιση μιας εικόνας με το πάτημα ενός κουμπιού από το χρήστη.

4.4.4 React JS

Η React JS είναι μια ευέλικτη και αποτελεσματική βιβλιοθήκη της JavaScript η οποία χρησιμοποιείται για τη δημιουργία του user interface σε ένα web application. Η react χωρίζει τη διεπιφάνεια του χρήστη σε πολλαπλά components, κομμάτια κώδικα που μπορούν να επαναχρησιμοποιηθούν και να επεξεργαστούν ξεχωριστά. Με αυτό τον τρόπο η δημιουργία δυναμικών εφαρμογών είναι ευκολότερη και προσφέρεται περισσότερη λειτουργικότητα. Επίσης η εύρεση των σφαλμάτων είναι ευκολότερη. Τα components έχουν καταστάσεις-states, αντικείμενα που περιέχουν πληροφορίες για τα components και αλλάζουν ανάλογα με

κάποιες δράσεις στο σύστημα, για παράδειγμα εάν ο χρήστης πατήσει ένα κουμπί. Οι καταστάσεις κρίνουν το πως θα γίνει rendered το component, το πως θα εμφανιστεί στην οθόνη ως κώδικας html. Η απόδοση του συστήματος βελτιώνεται αφού δεν χρειάζεται να φορτώνεται ολόκληρη η σελίδα κάθε φορά, γίνεται μόνο το component που υπέστηκε αλλαγές. Αυτό επιτυγχάνεται με τη χρήση της JSX, JavaScript Syntax Extension, που μας επιτρέπει να γράφουμε κώδικα html στο ίδιο αρχείο που περιέχει κώδικα JavaScript [17].

```
class Square extends React.Component {
  render() {
    return (
      <button className="square" onClick={function() { alert('click'); }}>
        {this.props.value}
      </button>
    );
  }
}
```

Σχήμα 4.2 Κώδικας React Component γραμμένος με σύνταξη JSX

4.4.5 React Modules

Η React JS παρέχει πολυάριθμες βιβλιοθήκες που κάνουν την υλοποίηση του κώδικα ευκολότερη.

React Router

Η React Router είναι μια συλλογή από components πλοήγησης που ενώνουν με συνδέσμους τα διάφορα components της εφαρμογής, αλλάζοντας για παράδειγμα το browser URL [18]. Ένα παράδειγμα είναι πατώντας σε ένα κουμπί που έχει Link component, να μεταβαίνει σε ένα άλλο React component με διαφορετικό URL.

React axios

Η React axios είναι ένα component που διαχειρίζεται τα http requests και responses του χρήστη, χρησιμοποιώντας απλές μεθόδους όπως get και post [19]. Μέσα στις get και post μπαίνει ως παράμετρος το URL στο οποίο βρίσκεται ο πόρος που ζήτησε ο χρήστης. Τα responses αντιμετωπίζονται με την έννοια των promises, αντικείμενα της JavaScript τα οποία μπορεί να παράγουν μια τιμή στο μέλλον. Υπάρχουν τρία πιθανά αποτελέσματα, η

πραγματοποίηση του promise και η επιστροφή αποτελέσματος, η απόρριψη του promise και η επιστροφή μηνύματος λάθους και η αναμονή. Με αυτό τον τρόπο η υλοποίηση είναι ευκολότερη και έχουμε καλύτερη διαχείριση σφαλμάτων.

```
componentDidMount() {  
  axios.get('http://localhost:3333/items')  
    .then(response => this.setState({items: response.data}))  
    .catch(err => console.log(err))  
}
```

Σχήμα 4.3 Κώδικας axios – http get request

4.4.6 Bootstrap

Πρόκειται για το πιο δημοφιλές, δωρεάν framework για front-end development, που στοχεύει στην υλοποίηση responsive εφαρμογών. Αποτελείται από πολλά έτοιμα components με σταθερά καλή σχεδίαση και ευκολία στη χρήση τους [20]. Για παράδειγμα για το παρόν σύστημα χρησιμοποίησα έτοιμο menu drop down, του οποίου πρόσθεσα λειτουργικότητα με JavaScript.

4.5 Back-End

Εκτός από το τι βλέπει ο χρήστης υπάρχει και η έννοια του back-end, όλα τα μέρη του συστήματος που είναι κρυμμένα προς αυτόν. Έχουμε να κάνουμε με διεργασίες που γίνονται στην πλευρά του server, όπως είναι η επικοινωνία με τη βάση δεδομένων, η επεξεργασία ενός request και η διαχείριση της μεταφόρτωσης των αρχείων.

4.5.1 Node.js

Το Node είναι μια πλατφόρμα ανάπτυξης λογισμικού χτισμένη σε JavaScript. Χρησιμοποιείται για την υλοποίηση του back-end όπως ανέφερα πιο πάνω. Ακολουθεί ένα μοντέλο non-blocking I/O και μπορεί να εκτελεί πολλαπλές διεργασίες ταυτόχρονα χωρίς να περιμένει κάποιο input ή output για να συνεχίσει [21].

4.5.2 Express

Το Express είναι ένα open-source framework για το Node.js που χρησιμοποιείται στο χτίσιμο web applications σε συνδυασμό με το front-end με ιδιαίτερη ευκολία. Πιο συγκεκριμένα εγκαθιστά ένα server-listener, που τρέχει συνέχεια και περιμένει να ‘ακούσει’ requests [22].

```
✓ app.listen(3001, () => {  
  |   console.log("running on port 3001");  
  | }  
})
```

Σχήμα 4.4 Κώδικας Express, server running

Όταν ληφθεί κάποιο request (req), το επεξεργάζεται, συνήθως κατασκευάζει ένα sql query που το στέλνει στη βάση δεδομένων, παίρνει αποτελέσματα πίσω από τη βάση και τελικά κατασκευάζει το response (res) που επιστρέφει στο χρήστη.

```
✓ app.get('/api/getPlays', (req, res) => {  
  |   const sql = "SELECT WorkID, Title FROM works ORDER BY Title Asc";  
  |   db.query(sql, (err, result) => {  
  |     |   res.send(result);  
  |     | }  
  |   });  
  | }  
})
```

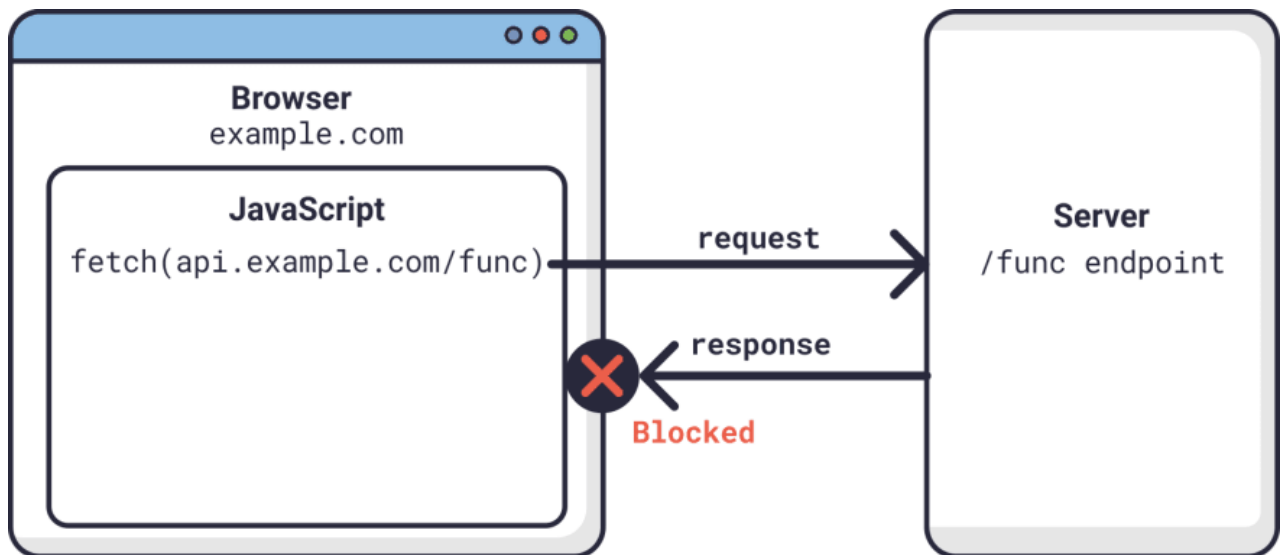
Σχήμα 4.5 Κώδικας Express, request handling

4.5.3 Node Modules

Όπως και η React JS, η Node και η Express υποστηρίζονται από πολλές χρήσιμες βιβλιοθήκες.

Cross-Origin Resource Sharing (CORS)

Το CORS είναι ένας μηχανισμός στο header των HTTP requests – responses που επιτρέπει στον server να μοιράζεται πόρους (εικόνες, κείμενο) σε κάποιο άλλο domain πέρα από το δικό του, ώστε ο browser που ζήτησε αυτούς τους πόρους να μπορεί να τους φορτώσει [23].



Σχήμα 4.6 CORS [24]

Express-fileupload

Το Express-fileupload διαχειρίζεται http requests, μέσα από τα οποία εντοπίζει φορτωμένα αρχεία, τα εξάγει προς τα έξω και επιτρέπει στο server να τα διαχειριστεί [25].

nodemon

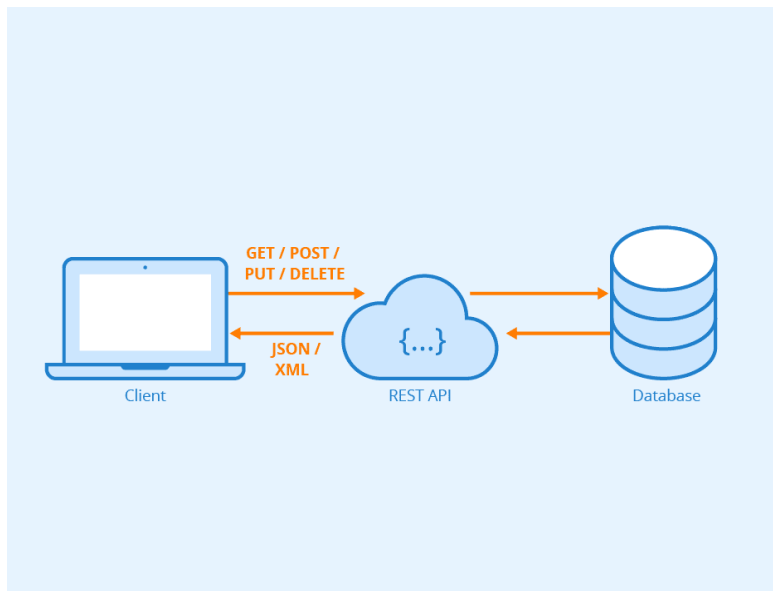
Με το nodemon μπορώ να έχω τον server να τρέχει και ταυτόχρονα να αλλάζω τον κώδικα στη Node.js. Μόλις εντοπίσει αλλαγές κάνει αυτόματα επανεκκίνηση και ο server συνεχίζει να τρέχει κανονικά, εκτός αν προέκυψε κάποιο σφάλμα κατά την αλλαγή του κώδικα [26].

4.6 Βάση Δεδομένων - MYSQL

Ένα αναγκαίο στοιχείο σε κάθε web application είναι η βάση δεδομένων, μια συλλογή από σχετιζόμενα δεδομένα. Τα δεδομένα είναι οργανωμένα σε πίνακες και μπορούν να εκτελεστούν επάνω τους πολλές πράξεις υπό την μορφή των queries στη γλώσσα SQL. Για την υλοποίηση της βάσης δεδομένων επέλεξα την MySQL [27]. Επέλεξα μια σχεσιακή βάση δεδομένων για να μπορώ να οργανώσω σε πίνακες τον μεγάλο όγκο δεδομένων που έχουν τα κείμενα και για να μπορώ να κατασκευάσω πολύπλοκα queries για τις λειτουργίες της πλατφόρμας, όπως θα δούμε και στη συνέχεια.

4.7 RESTful API

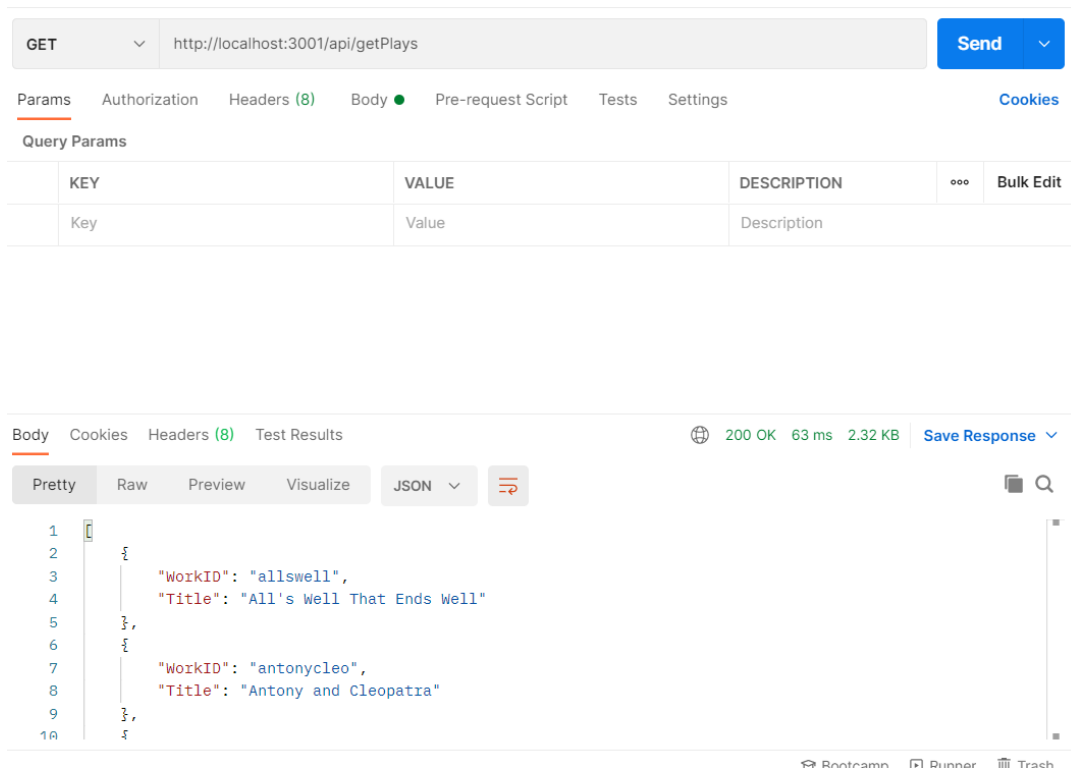
REST είναι η σύγχρονη αρχιτεκτονική που υιοθετούν οι υπηρεσίες για την επικοινωνία μέσω του HTTP Protocol. Βασική αρχή είναι ότι κάθε αντικείμενο είναι ένας πόρος ο οποίος προσδιορίζεται από ένα URL και πάνω στον οποίο μπορούν να εκτελεστούν οι HTTP μέθοδοι (POST, GET, PUT, DELETE). Τα URLs των πόρων καθώς και οι επιτρεπτές πράξεις είναι ορατές στο χρήστη μέσω ενός API (Application User Interface) [28]. Σε αυτή την ΑΔΕ, το API είναι κρυμμένο σε ένα χαμηλότερο επίπεδο κάτω από τα διάφορα κουμπιά της πλατφόρμας. Μέσα από τα axios της React, οι ενέργειες του χρήστη χτυπούν το API και στέλνεται το request προς το backend, στην Express. Η Express επεξεργάζεται το αίτημα, επικοινωνεί με τη βάση και επιστρέφει πίσω το αποτέλεσμα με τον ίδιο τρόπο που είχε φτάσει το αίτημα.



Σχήμα 4.7 RESTful API

4.7.1 Postman

Το Postman είναι ένα πολύ χρήσιμο εργαλείο που επιτρέπει στους χρήστες να δοκιμάζουν τα APIs τους [29]. Στην δική μου περίπτωση έχω την τυπική διαδικασία για να κάνω ένα request όπως ανέφερα και πιο πάνω, Browser => React => Express => MySQL => Express => React => Browser. Αν υπάρχει κάποιο σφάλμα είναι αρκετά δύσκολο να καταλάβω σε πιο σημείο υπάρχει πρόβλημα. Με το Postman η αποσφαλμάτωση έγινε πολύ επειδή μπορώ να δοκιμάζω άμεσα τις κλήσεις στο API και να παίρνω αποτελέσματα.



Σχήμα 4.8 Postman get request με αποτελέσματα

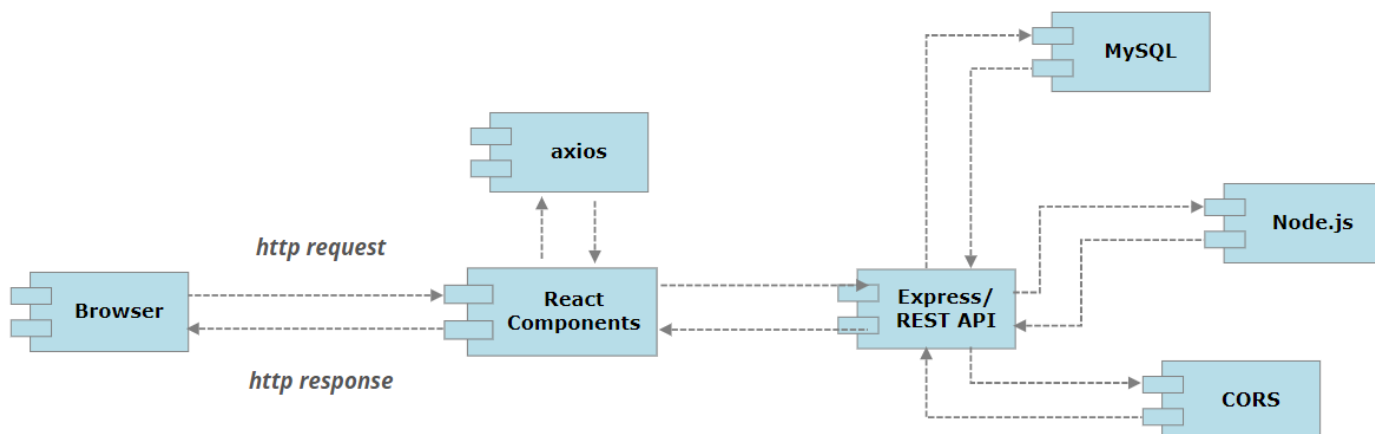
4.8 npm

Το npm είναι πρόγραμμα διαχείρισης πακέτων για την JavaScript και τις σχετικές τις πλατφόρμες. Εγκαθίσταται μαζί με το Node.js και με τη χρήση του απέκτησα όλα τα modules και τις εξαρτήσεις που χρειάζονται για αυτό το project [30].

4.9 Component Diagram

Major Components:

1. React – render User Interface
2. Axios – make HTTP requests from browser and Nodejs
3. Node.js – Web Server, handles data communication
4. Express.js – APIs, handles HTTP requests
5. Cors – allows server to request resources from an outside domain
6. MySQL – RDBMS



Σχήμα 4.9 Component Diagram

Κεφάλαιο 5

Προετοιμασία Κειμένων και Βάσης Δεδομένων

5.1 Σχεδίαση Βάσης Δεδομένων	27
5.2 Οργάνωση Κειμένων	28
5.3 Parsing	30

5.1 Σχεδίαση Βάσης Δεδομένων

Όπως είχα ήδη αναφέρει, το πρώτο βήμα στην υλοποίηση της πλατφόρμας ήταν να μάθω το πως ο δημιουργός του OpenSource Shakespeare έφτιαξε τη βάση δεδομένων του και επίσης να κατεβάσω τον κώδικα της βάσης, ο οποίος είναι ανοικτός κώδικας. Για τη δημιουργία της βάσης χρησιμοποίησα το λογισμικό MySQL Workbench [31]. Από την Β.Δ. του OSS είχα κρατήσει τους παρακάτω πίνακες, αφαίρεσα μερικές στήλες που πίστευα ότι δεν θα χρειαστώ και κράτησα όλα τους τα δεδομένα:

- works – Περιέχει ονομαστικά όλα τα έργα του Σαίξπηρ
- chapters – Περιέχει τον τίτλο της κάθε σκηνής για κάθε έργο
- characters – Περιέχει όλους τους χαρακτήρες σε κάθε έργο
- paragraphs – Περιέχει όλες τις παραγράφους του κάθε έργου, δηλαδή ολόκληρο το κείμενο

Ο κύριος στόχος της ΑΔΕ μου είναι η παράλληλη παρουσίαση των Αγγλικών με των Ελληνικών κειμένων, ή ακόμα καλύτερα η διασύνδεση τους. Το επόμενο βήμα ήταν να δημιουργήσω τους 4 αντίστοιχους πίνακες για τα Ελληνικά κείμενα:

- worksGreek
- chaptersGreek
- charactersGreek
- paragraphsGreek

Ο κάθε πίνακας χρειάζεται ένα σύνδεσμο στον αντίστοιχο Αγγλικό πίνακα, ένα foreign key.

5.2 Οργάνωση κειμένων

SCENE I. Verona. A public place.

Enter SAMPSON and GREGORY, of the house of Capulet, armed with swords and bucklers

SAMPSON

Gregory, o' my word, we'll not carry coals.

GREGORY

No, for then we should be colliers.

SAMPSON

I mean, an we be in choler, we'll draw.

GREGORY

Ay, while you live, draw your neck out o' the collar.

Σχήμα 5.0 Αγγλικό αρχείο κειμένου, Ρωμαίος και Ιουλιέτα

Στο πιο πάνω απόσπασμα αρχικά έχουμε αλλαγή σκηνής μαζί με τον τίτλο της σκηνής. Μετά υπάρχει μια γραμμή που χρησιμεύει ως σκηνοθετική οδηγία. Το υπόλοιπο κείμενο είναι η συνομιλία 2 χαρακτήρων, του Sampson και του Gregory. Το θέμα της ΑΔΕ είναι τα θεατρικά έργα, όπως ο Ρωμαίος και η Ιουλιέτα, που πάντοτε αποτελούνται από διαλόγους χαρακτήρων. Ο τρόπος που οργανώνω, διασπάω και διαχειρίζομαι τα κείμενα είναι ως παραγράφους, όπου η κάθε παράγραφος είναι τα λόγια του κάθε χαρακτήρα. Το πιο πάνω απόσπασμα είναι αποθηκευμένο με αυτόν τον τρόπο στη βάση, κάτω από τον πίνακα paragraphs:

WorkID	ParagraphID	CharID	PlainText
romeojuliet	886860	sampson	Gregory, o' my word, we'll not carry coals.
romeojuliet	886862	gregory	No, for then we should be colliers.
romeojuliet	886863	sampson	I mean, an we be in choler, we'll draw.
romeojuliet	886864	gregory	Ay, while you live, draw your neck out o' the collar.

Σχήμα 5.1 paragraphs table

Για να μπορώ όμως να συνδέσω τα κείμενα των 2 γλωσσών έπρεπε να προσθέσω μια νέα στήλη και στους 2 πίνακες που να καταγράφει τον αριθμό παραγράφου, δηλαδή τον αριθμό εμφάνισης χαρακτήρα. Ονόμασα το νέο πεδίο CharLine, η βασική ιδέα είναι όταν τα

CharLine του Αγγλικού και του Ελληνικού κειμένου είναι ίσα, τότε έχουμε αντιστοίχιση των 2 παραγράφων. Μια αλλαγή που έπρεπε να γίνει, ήταν η αγνόηση των σκηνοθετικών οδηγιών, όπως είδαμε και πιο πάνω, όπως και τα λόγια του χορού που δεν υπάρχουν σε όλες τις Ελληνικές μεταφράσεις.

WorkID	ParagraphID	CharID	PlainText	CharLine
romeojuliet	886860	sampson	Gregory, o' my word, we'll not carry coals.	1
romeojuliet	886862	gregory	No, for then we should be colliers.	2
romeojuliet	886863	sampson	I mean, an we be in choler, we'll draw.	3
romeojuliet	886864	gregory	Ay, while you live, draw your neck out o' the collar.	4

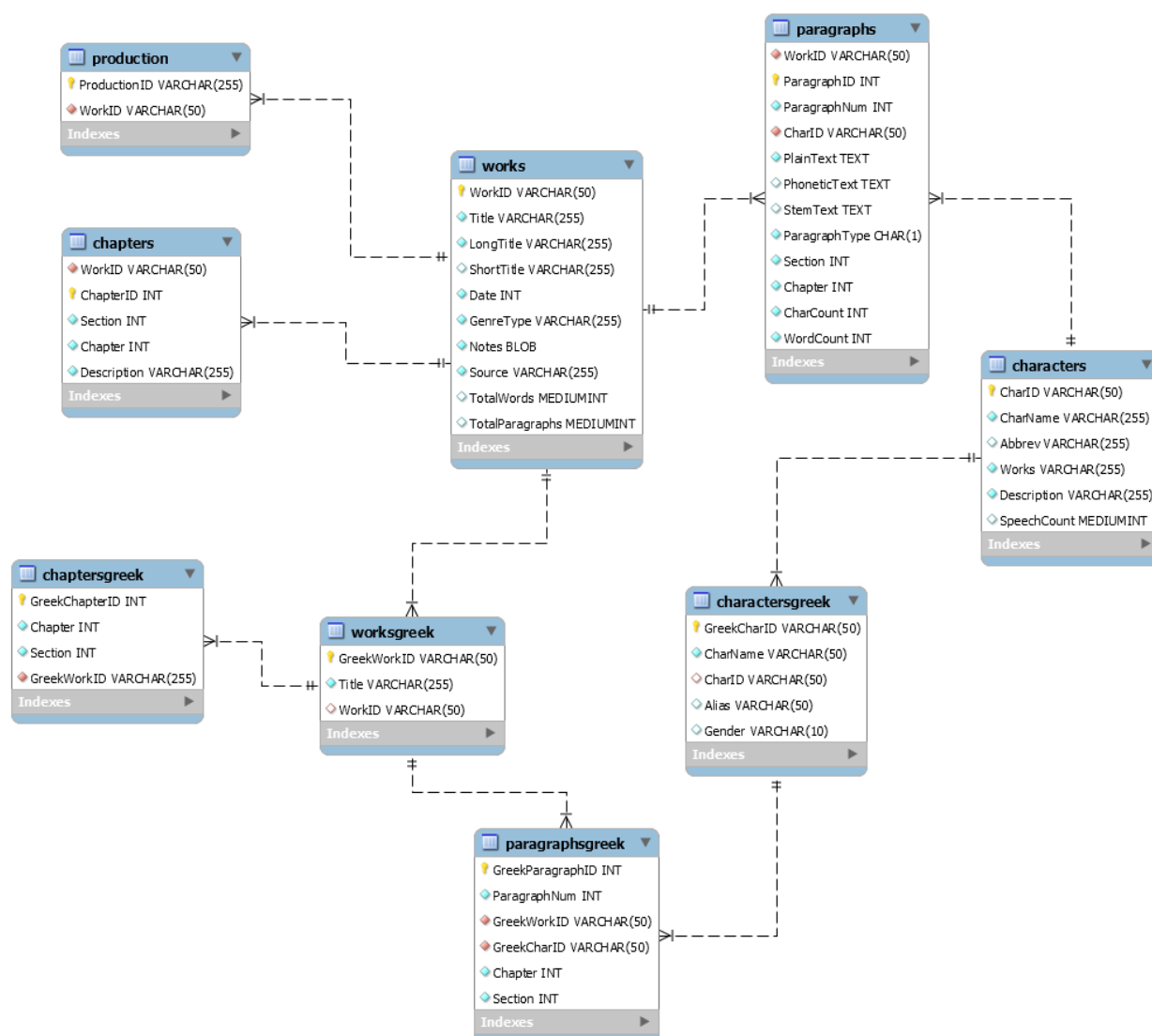
Σχήμα 5.2 paragraphs table με την προσθήκη μετρητή παραγράφων CharLine

GreekWorkID	GreekParagraphID	GreekCharID	PlainText	CharLine
vikelasromeo	55656	sampson	Γρηγόρη, μα την πίστιν μου ούτ' ε...	1
vikelasromeo	55657	gregory	Όχι βέβαια! μήπως είμεθα γαιδούρ...	2
vikelasromeo	55658	sampson	Θέλω να ειπώ, αν μας θυμώσουν ...	3
vikelasromeo	55659	gregory	Κύτταξε όμως να μη πηδός πολλά ...	4

Σχήμα 5.3 paragraphsgreek table με CharLine

Μια ακόμα αλλαγή ήταν να τοποθετηθεί στον πίνακα worksGreek ένα επιπλέον πεδίο για την αποθήκευση του μεταφραστή. Παρακάτω φαίνεται το διάγραμμα ολόκληρης της Βάσης Δεδομένων.

Database diagram (ER)



Σχήμα 5.4 Database diagram

5.3 Parsing

Η δομή της βάσης δεδομένων είναι έτοιμη, οι πίνακες των Αγγλικών κειμένων έχουν δεδομένα αλλά οι πίνακες των Ελληνικών είναι κενοί. Στη διάθεση μου έχω τις Ελληνικές μεταφράσεις σε μορφή κειμένου και τώρα πρέπει να διασπάσω αυτά τα κείμενα για εισέλθουν σωστά στους πίνακες που ανέφερα. Αυτή η διαδικασία επιτυγχάνεται με το parsing, σε 2 στάδια. Πρώτα το αρχείο κειμένου γίνεται parsed, δηλαδή μαρκάρεται με συγκεκριμένα σύμβολα και γενικά αλλάζει η δομή του κειμένου. Στο δεύτερο στάδιο τα σύμβολα διαβάζονται και στο καθένα εκτελούνται κάποιες πράξεις με αποτέλεσμα να τοποθετηθεί το κείμενο χωρισμένο μέσα στη βάση. Για το πρώτο στάδιο χρησιμοποιώ το

πρόγραμμα στην Perl που ονόμασα parser.pl και για το δεύτερο στάδιο χρησιμοποιώ ένα άλλο πρόγραμμα και πάλι στην Perl, με το όνομα insertion.pl. Για να γίνει κατανοητή όλη η διαδικασία, επινόησα την πιο κάτω μεθοδολογία. Επίσης αναφέρω ξανά ότι στα πλαίσια της Ατομικής Διπλωματικής Εργασίας μου, ασχολήθηκα με το έργο Ρωμαίος και Ιουλιέτα και τις μεταφράσεις του Βικέλα και του Μπελιέ.

Μεθοδολογία Parsing

1. Εισαγωγή στον πίνακα worksGreek το όνομα του έργου, του μεταφραστή και ως ξένο κλειδί το κλειδί του αντίστοιχου Αγγλικού έργου.

GreekWorkID	Title	WorkID	Translator
vikelasromeo	ΡΩΜΑΙΟΣ ΚΑΙ ΙΟΥΛΙΕΤΑ	romeojuliet	Βικέλας

Σχήμα 5.5 Εισαγωγή εγγραφής στον πίνακα worksgreek

2. Ανοίγω το αρχείο κειμένου, βρίσκω όλους τους χαρακτήρες του έργου με τις διάφορες συντομογραφίες που μπορεί να έχουν και τους εισάγω στον πίνακα greekCharacters, βάζοντας επίσης και για ξένο κλειδί το κλειδί του αντίστοιχου χαρακτήρα του Αγγλικού πίνακα characters.

GreekCharID	CharName	GreekWorkID	CharID	Alias	Alias2
abraham	ΑΒΡΑΑΜ, υπηρέτης του Μοντέκη	vikelasromeo	abraham-rj	ΑΒΡΑΑΜ	NULL
apothecary	ΦΑΡΜΑΚΟΠΩΛΗΣ	vikelasromeo	apothecary	ΦΑΡΜΑΚΟΠΩΛΗΣ	NULL
balthasar	ΒΑΛΤΑΣΣΑΡ, υπηρέτης του Ρωμαιο	vikelasromeo	balthasar	ΒΑΛΤΑΣΣΑΡ	NULL
benvolio	ΜΠΕΜΒΟΛΙΟΣ, ανεψιός του Μοντέκ...	vikelasromeo	benvolio	ΜΠΕΜΒΟΛΙΟΣ	NULL
blank	blank	vikelasromeo	blank	blank	NULL
capulet	ΚΑΠΟΥΛΕΤΟΣ, αρχηγός της μιας απ...	vikelasromeo	capulet	ΚΑΠΟΥΛΕΤΟΣ	NULL
citizen-rj	ΠΟΛΙΤΗΣ	vikelasromeo	1citizen	ΕΙΣ ΠΟΛΙΤΗΣ	Ο ΠΟΛΙΤΗΣ
friarjohn	ΠΑΤΕΡ ΙΩΑΝΝΗΣ, καλόγηρος	vikelasromeo	friarjohn	ΠΑΤΕΡ ΙΩΑΝΝΗΣ	NULL
friarlaurence	ΠΑΤΕΡ ΛΑΥΡΕΝΤΙΟΣ, καλόγηρος	vikelasromeo	friarlaurence	ΛΑΥΡΕΝΤΙΟΣ	NULL
gregory	ΓΡΗΓΟΡΗΣ, υπηρέτης του Καπουλέτου	vikelasromeo	gregory	ΓΡΗΓΟΡΗΣ	ΓΚΡΕΓΚΟΡΤ
juliet	ΙΟΥΛΙΕΤΑ, θυγάτηρ του Καπουλέτου	vikelasromeo	juliet	ΙΟΥΛΙΕΤΑ	NULL
ladycapulet	Η ΣΥΖΥΓΟΣ του Καπουλέτου	vikelasromeo	ladycapulet	ΚΑΠΟΥΛΕΤΑΙΝΑ	ΚΑΠΟΥΛΕΤΟΣ
ladymontague	Η ΣΥΖΥΓΟΣ του Μοντέκη	vikelasromeo	ladymont...	ΜΟΝΤΕΚΑΙΝΑ	NULL
mercutio	ΜΕΡΚΟΥΤΙΟΣ, συγγενής του πρίγκη...	vikelasromeo	mercutio	ΜΕΡΚΟΥΤΙΟΣ	NULL
montague	ΜΟΝΤΕΚΗΣ, αρχηγός της μιας από π...	vikelasromeo	montague	ΜΟΝΤΕΚΗΣ	NULL
musician-a	Α ΜΟΥΣΙΚΟΣ	vikelasromeo	1musician-rj	Α' ΜΟΥΣΙΚΟΣ	NULL
musician-b	Β ΜΟΥΣΙΚΟΣ	vikelasromeo	2musician-rj	Β' ΜΟΥΣΙΚΟΣ	NULL
musician-c	Γ ΜΟΥΣΙΚΟΣ	vikelasromeo	3musician-ri	Γ' ΜΟΥΣΙΚΟΣ	NULL

Σχήμα 5.6 Εισαγωγή χαρακτήρων στον πίνακα charactersgreek

- \\([0-9]+\\)), έτσι ώστε να φύγουν οι αναφορές από το κείμενο πχ [20], [32]

- Αντικατάσταση ' με '
- Αντικατάσταση — με -
- Αντικατάσταση ' με '

Εντολή: \$ Perl parser.pl vikelasromeo.

- Το parser.pl πρώτα συνδέεται με τη βάση μέσω του module DBI.
- Μετά βρίσκει όλους τους χαρακτήρες του έργου που δόθηκε ως argument, τοποθετώντας τους σε ένα πίνακα.

Σχήμα 5.7 κώδικας parser.pl – κατασκευή πίνακα χαρακτήρων

- Έπειτα ανοίγει το αρχείο κειμένου χρησιμοποιώντας κωδικοποίηση utf-8 και αφαιρεί τους περιττούς κενούς χαρακτήρες (whitespace).

- Τέλος μαρκάρει το κείμενο, διαβάζοντας πρόταση-πρόταση. Αν εντοπίσει όνομα χαρακτήρα ίσο με κάποιο από τα ονόματα που έχει ο πίνακας που δημιουργήθηκε στην αρχή του προγράμματος, βάζει το σύμβολο % στην αρχή της πρότασης.
- Όταν φτάσει στην επόμενη γραμμή αν δεν βρει όνομα χαρακτήρα, τότε σημαίνει ότι τα λόγια του προηγούμενου χαρακτήρα συνεχίζονται και γίνεται μαρκάρισμα με το σύμβολο ^.
- Στις γραμμές που αρχίζουν με παρένθεση τοποθετεί το σύμβολο &, δηλώνει ότι είναι σκηνοθετική οδηγία ή τίτλος σκηνής.

```

if (/^\s*$/){
}

else{
    $flag=0;
    $line2=$line;
    $line2 =~ s/^\s+//;

    @spl = split(' ', $line2, 2);
    $character=$spl[0];
    $restLine=$spl[1];
    for my $i (@chars){
        if ($i."." eq $character){
            $line2="%'.$line2;
            $flag=1;
            last;
        }
        if ($i eq $character){
            $line2="%'.$character.'.'.$restLine;
            $flag=1;
            last;
        }
    }
    $character='';
    if ($flag == 0){
        $line2='^'.$line2;
    }
}

```

Σχήμα 5.8 κώδικας parser.pl – μαρκάρισμα κειμένου

6. Χειροκίνητο – Manual πέραςμα του μαρκαρισμένου κειμένου για τοποθέτηση \$ συμβόλων όπου ξεκινά νέα Πράξη ή νέα σκηνή και γρήγορη εύρεση λαθών.

ΣΚΗΝΗ ΠΡΩΤΗ (3).

Πλατεία.
(Εισέρχονται ο ΣΑΜΨΩΝ και ο ΓΡΗΓΟΡΗΣ φέροντες ξίφη και ασπίδας).

ΣΑΜΨΩΝ
Γρηγόρη, μα την πίστιν μου ούτε' εγώ τρώγω άχυρα,
ούτε συ! (4)

ΓΡΗΓΟΡΗΣ
Όχι βέβαια! μήπως είμεθα γαϊδούρια;

ΣΑΜΨΩΝ
Θέλω να ειπώ, αν μας θυμώσουν να πηδήσωμεν κατ'
επάνω των.

ΓΡΗΓΟΡΗΣ
Κύτταξε όμως να μη πηδάς πολλά παλούκια.

Σχήμα 5.9 Αρχείο κειμένου μετάφρασης

```
$SECTION 1.
$CHAPTER 1. Πλατεία.
&.(Εισέρχονται ο ΣΑΜΨΩΝ και ο ΓΡΗΓΟΡΗΣ φέροντες ξίφη και ασπίδας).
%ΣΑΜΨΩΝ. Γρηγόρη, μα την πίστιν μου ούτε' εγώ τρώγω άχυρα,
^ούτε συ!
%ΓΡΗΓΟΡΗΣ. Όχι βέβαια! μήπως είμεθα γαϊδούρια;
%ΣΑΜΨΩΝ. Θέλω να ειπώ, αν μας θυμώσουν να πηδήσωμεν κατ'
^επάνω των.
%ΓΡΗΓΟΡΗΣ. Κύτταξε όμως να μη πηδάς πολλά παλούκια.
```

Σχήμα 5.10 Αρχείο κειμένου μετάφρασης – μετά το μαρκάρισμα

- Μέσα από το command line τρέχω το insertion.pl με είσοδο το όνομα του νέου parsed αρχείου και το όνομα που δώσαμε στο κλειδί του έργου στον πίνακα worksGreek.

Εντολή: \$ Perl insertion.pl vikelasromeo

- Το insertion.pl αρχικά ενώνεται με τη βάση μέσω του DBI.
- Διαγράφει τα περιεχόμενα των πινάκων της βάσης chaptersGreek, paragraphsGreek για το επιλεγόμενο έργο.
- Περνά γραμμή-γραμμή το μαρκαρισμένο κείμενο.

- Εάν βρει το σύμβολο \$, ακολουθούμενο από ‘Section’ ή ‘Chapter’, ενημερώνει τον πίνακα chaptersGreek με την αλλαγή Πράξης ή Σκηνής.
- Αν βρει το σύμβολο %, παίρνει τη συντομογραφία – Alias του χαρακτήρα που μιλά, την συγκρίνει με τους χαρακτήρες που υπάρχουν στον πίνακα charactersGreek, και ενημερώνει τον πίνακα paragraphsGreek με τα λόγια του χαρακτήρα, αυξάνοντας και την τιμή του CharLine κατά 1.
- Εάν βρει το σύμβολο ^, τότε κάνει append την ακολουθούμενη γραμμή στα λόγια του πιο πρόσφατου χαρακτήρα που βρέθηκε.
- Στο τέλος με λίγη τροποποίηση στον κώδικα, βρίσκει τον αριθμό γραμμής στην περίπτωση που λείπουν λόγια από τα Ελληνικά κείμενα, ενώ τα λόγια υπάρχουν στα Αγγλικά. Μπαίνει το κενό και προχωρεί ο counter του αριθμού παραγράφων (LineCount). Έτσι συνεχίζει να υπάρχει σύνδεση στις παραγράφους των 2 κειμένων 1 προς 1.

```

✓ sub charfinder {
    $tempcharline = $_[0];
    $tempcharparagraphtype = $_[1];

    if ($tempcharparagraphtype ne 's') {
        # get the chartemp value
        $pdloc = index($tempcharline, ".");
        $chartemp = substr($tempcharline, 0, $pdloc);
        $tempcharline = substr($tempcharline, $pdloc + 2);

        $charid = '';
        # print "$chartemp";

        $chartemp =~ s/^\s+|\s+$//g;
        $chartemp =~ s/\'/\'\'/g;
        # get character info from db
        $getcharinfo = $db->prepare( "SELECT *
        FROM charactersgreek
        WHERE GreekWorkID
        LIKE '%$currentwork%'
        AND Alias='$chartemp'
        OR Alias2='$chartemp'");
        $getcharinfo->execute() or die $DBI::errstr;
    }
}

```

Σχήμα 5.11 Κώδικας insertion.pl – Εύρεση χαρακτήρα από τη ΒΔ που να έχει ίδιο Alias

GreekWorkID	GreekParagraphID	GreekCharID	PlainText	Section	Chapter	CharLine
vikelasromeo	55656	sampson	Γρηγόρη, μα την πίστιν μου ούτ' ε...	1	1	1
vikelasromeo	55657	gregory	Όχι βέβαια! μήπως είμεθα γαϊδούρ...	1	1	2
vikelasromeo	55658	sampson	Θέλω να ειπώ, αν μας θυμώσουν ...	1	1	3
vikelasromeo	55659	gregory	Κύτταξε όμως να μη πηδός πολλά ...	1	1	4

Σχήμα 5.12 Πίνακας paragraphsgreek – μετά το insertion.pl

GreekChapterID	GreekWorkID	Section	Chapter	Description
1367	vikelasromeo	1	1	Πλατεία.
1368	vikelasromeo	1	2	Οδός έμπροσθεν της οικίας του Καπουλέτου.
1369	vikelasromeo	1	3	Θάλαμος εν τη οικία του Καπουλέτου.
1370	vikelasromeo	1	4	Οδός εις Βερώναν.
1371	vikelasromeo	1	5	Αίθουσα εν τη, οικία του Καπουλέτου.
1372	vikelasromeo	2	1	Πλατεία παρά τον κήπον του Καπουλέτου.

Σχήμα 5.13 Πίνακας chaptersgreek – μετά το insertion.pl

Η βάση δεδομένων είναι πλέον έτοιμη και απομένει η υλοποίηση του Web Application.

Κεφάλαιο 6

Υλοποίηση της Web εφαρμογής Dramatica

6.1 Ρύθμιση Περιβάλλοντος	37
6.2 Υλοποίηση Λειτουργιών και Συστατικών Πλατφόρμας	39
6.3 Deployment	44
6.4 Παρουσίαση Τελικού Συστήματος	45

6.1 Ρύθμιση Περιβάλλοντος

Αρχικά έπρεπε να εγκαταστήσω τις τεχνολογίες που ανέφερα στο Κεφάλαιο 4. Πρώτα κατέβασα το Node.js μαζί με το npm, για το back-end της εφαρμογής και για την εγκατάσταση βιβλιοθηκών. Στη συνέχεια εγκατέστησα το Visual Studio Code ως το πρόγραμμα επεξεργασίας πηγαίου κώδικα. Το VS Code έρχεται μαζί με γραμμή εντολών την οποία χρησιμοποίησα για πολλούς σκοπούς [32].

Δημιουργία εφαρμογής React / React Web Server

Αφού άνοιξα το Visual Studio Code, επέλεξα ένα φάκελο για το project. Μέσα στο φάκελο δημιούργησα 2 νέους φακέλους και τους έδωσα ονόματα client και server. Μετέβηκα στο φάκελο client και έγραψα στη γραμμή εντολής την εντολή δημιουργίας εφαρμογής React μαζί με web server.

Εντολή: \$ npm create-react-app

Η εφαρμογή δημιουργείται μαζί με διάφορα αρχεία, αυτά που κράτησα είναι τα ακόλουθα:

- App.js – Εδώ υπάρχει ο περισσότερος κώδικας, όλα τα components, οι συναρτήσεις και οι λειτουργικότητες. Υπάρχει
- Index.js – Περιέχει τον κώδικα που καλείται πρώτος με την εκκίνηση του React Web Server. Κάνει render το component App.js, που περιέχει όλα τα

άλλα components. Ο κώδικας είναι JavaScript, HTML και όταν συνδυαστούν JSX syntax.

- App.css – Κώδικας CSS για τα html elements
- Index.html – Η μια και μοναδική σελίδα HTML που χρησιμοποιεί η React, αφού κάθε φορά την κάνει render αλλά επιλέγοντας να εμφανίσει διαφορετικά React components.
- package.json – Περιέχει όλα τα dependencies που υπάρχουν με την εγκατάσταση της εφαρμογής ως επίσης και όλα τα μελλοντικά dependencies που θα αναφέρω στη συνέχεια.

Δημιουργία Node Application Server

Μετά την δημιουργία της εφαρμογής React άλλαξα directory και μετέβηκα στο φάκελο server. Δημιούργησα ένα αρχείο JavaScript με το όνομα server.js .

Εγκατάσταση πακέτων/modules

Από το command line εγκατέστησα όλα τα απαραίτητα modules, τα οποία ανέφερα στο Κεφάλαιο 4. Η πρώτη εντολή είναι για το client directory ενώ η δεύτερη για το server directory.

Εντολή: \$ npm install axios react-router-dom react-bootstrap bootstrap

Εντολή: \$ npm install express body-parser cors nodemon mysql express-fileupload

Εκκίνηση React Web Server / Node Application Server

Υλοποίησα τον Application Server Node.js με τη χρήση κώδικα υψηλού επιπέδου από το Express framework, ορίζοντας listener στο port 3001 και ενώνοντας τον με τη βάση δεδομένων. Υπενθυμίζω ότι ο server είναι local host και λειτουργεί μόνο στη συσκευή στην οποία τρέχει

```

✓ const db = mysql.createConnection({
  host: 'localhost',
  user: 'root',
  password: 'password',
  database: 'dramaticadb',
  port: 3306
});

✓ app.listen(3001, () => {
  console.log("running on port 3001");
})

```

Σχήμα 6.1 κώδικας server.js

Τώρα μπορώ να ξεκινήσω τους 2 servers:

Εντολή: **\$ npm start (react)**

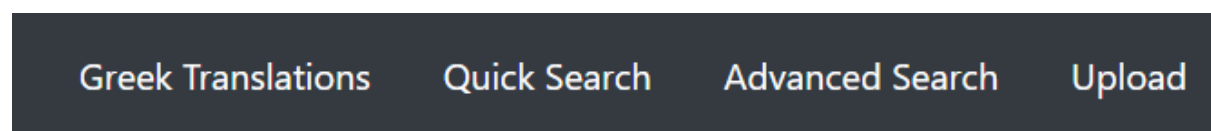
Εντολή: **\$ npm run devStart (node)**

6.2 Υλοποίηση Λειτουργιών και Συστατικών

Όπως είχα αναφέρει προηγουμένως, υπάρχει μόνο μια σελίδα HTML και επάνω τις φορτώνονται τα διάφορα React components. Για λειτουργία συνδέσμων και τη μετάβαση μεταξύ των στοιχείων της σελίδας, χρειάζεται ένας React Router με Links, ο οποίος καθορίζει τα URLs.

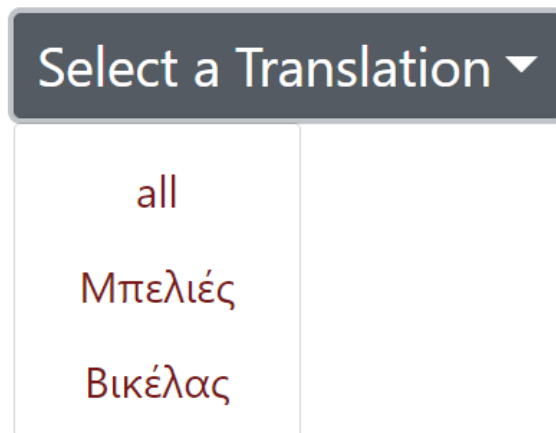
Τα κύρια components της πλατφόρμας είναι:

- Navbar



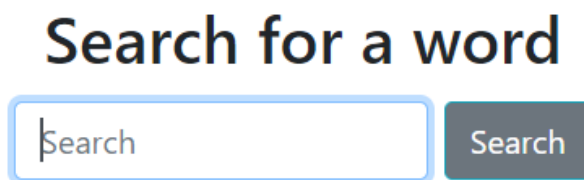
Σχήμα 6.2 Navbar

- Dropdown Lists



Σχήμα 6.3 Dropdown Menu

- Search form and button



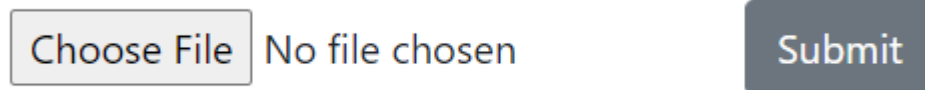
Σχήμα 6.4 Search component

- Search results table

#	Title	Scene	Character	Text
0	Romeo and Juliet	1,1	BENVOLIO	Madam, an hour before the worshipp'd sun Peer'd forth the golden winc sycamore That westward rooteth from the city's side, So early walkin ^g d wood: I, measuring his affections by my own, That most are busied wher from me.
1	Romeo and Juliet	1,1	ROMEO	Alas, that love, whose view is muffled still, Should, without eyes, see pat it all. Here's much to do with hate, but more with love. Why, then, O bra Mis-shapen chaos of well-seeming forms! Feather of lead, bright smoke, this. Dost thou not laugh?
2	Romeo and Juliet	1,1	ROMEO	Why, such is love's transgression. Griefs of mine own lie heavy in my bre shown Doth add more grief to too much of mine own. Love is a smoke r nourish'd with lovers' tears: What is it else? a madness most discreet, A c

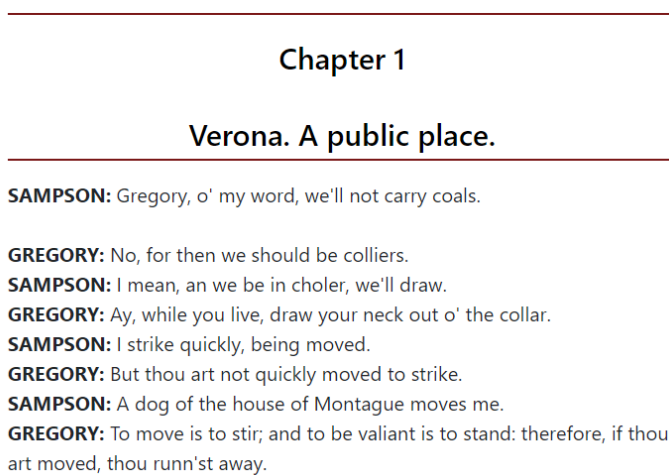
Σχήμα 6.5 Αποτελέσματα αναζήτησης λέξης

- Upload component



Σχήμα 6.6 upload component

- Text components



Chapter 1

Verona. A public place.

SAMPSON: Gregory, o' my word, we'll not carry coals.

GREGORY: No, for then we should be colliers.

SAMPSON: I mean, an we be in choler, we'll draw.

GREGORY: Ay, while you live, draw your neck out o' the collar.

SAMPSON: I strike quickly, being moved.

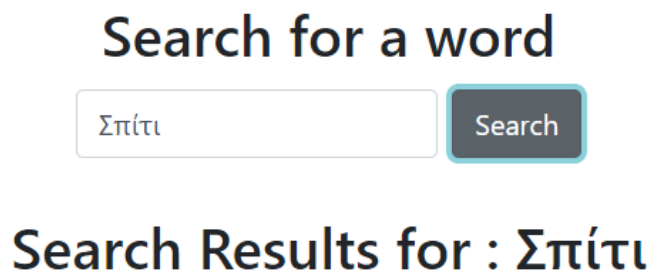
GREGORY: But thou art not quickly moved to strike.

SAMPSON: A dog of the house of Montague moves me.

GREGORY: To move is to stir; and to be valiant is to stand: therefore, if thou art moved, thou runn'st away.

Σχήμα 6.7 Text component

Παράδειγμα αναζήτησης λέξης στο Search component:



Search for a word

Search Results for : Σπίτι

Σχήμα 6.8 Παράδειγμα αναζήτησης λέξης

Ο χρήστης μεταβαίνει στην ιστοσελίδα της πλατφόρμας Dramatica και αναζητεί τη λέξη 'Σπίτι'. Τα components λειτουργούν με states. Για παράδειγμα αν ο χρήστης γράψει μια λέξη στην φόρμα αναζήτησης, τότε το state της φόρμας αλλάζει και πλέον έχει μέσα μια τιμή. Στη γραμμή 727 του πιο κάτω κώδικα φαίνεται η δήλωση του state και πιο κάτω ο τρόπος με τον οποίο καλείται η συνάρτηση αλλαγής του state, η setValue. Η τιμή του state, η value, είναι η τιμή εισόδου του χρήστη, δηλαδή η λέξη Σπίτι.

```
724 const Search = () => {
725
726
727   const [value, setValue] = useState(),
728   onInput = ({target:{value}}) => setValue(value),
729   onFormSubmit = e => {
730     e.preventDefault()
731     setValue()
732   }
733
734   let empty=true;
735   const [searchedWord, setSearchWord] = useState();
736   const [searchResults, setSearchResults] = useState([]);
737 }
```

Σχήμα 6.9 κώδικας React - states

```
useEffect(() => {
  console.log(searchedWord)
  Axios.get('http://localhost:3001/api/searchWord', {
    params: {
      word:searchedWord
    }
  })
  .then((response) => {
    setSearchResults(response.data);
    if (response.data.length===0){
      empty=true;
    }
    else{
      empty=false;
    }
  })
})
```

Σχήμα 6.10 React -axios

Στο ίδιο component έχουμε τη συνάρτηση useEffect η οποία κατά το rendering του component και με τη βοήθεια της βιβλιοθήκης axios, κάνει get request στο url searchWord και έχει ως παράμετρο την τιμή Σπίτι, που είδαμε πιο πάνω.


```

app.get('/api/searchWord', (req, res) => {
  let word = req.query.word;
  let replacement = '%'+word+'%';
  let replacement2 = '%'+word+'%';
  const sql="SELECT Title,Section,Chapter,Translator,Abbrev
  db.query(sql,[replacement,replacement2],(err,result) => {
    res.send(result);
  });
})

```

Σχήμα 6.11 κώδικας node js/express – http get request handling

Από την πλευρά του server node.js, αντιμετωπίζεται το get request του χρήστη. Ετοιμάζεται κώδικας sql με την παράμετρο word, την λέξη που αναζήτησε ο χρήστης. Το query στέλνεται στη βάση δεδομένων. Αν δεν υπάρξει σφάλμα επιστρέφονται τα αποτελέσματα πίσω στον server node.js , που με τη σειρά του τα προωθεί στο web server της React.

```

SELECT Title,Section,Chapter,Translator,Abbrev as 'Char',PlainText
FROM dramaticadb.paragraphs
INNER JOIN works
ON works.WorkID=paragraphs.WorkID
INNER JOIN characters
ON characters.CharID=paragraphs.CharID
where paragraphs.CharID!='xxx' && paragraphs.CharID='chorus-rj' && works.WorkID='romeojuliet'
&& PlainText LIKE ? UNION ALL SELECT Title,Section,Chapter,Translator,Alias AS 'Char',PlainText
FROM dramaticadb.paragraphsgreek
INNER JOIN worksgreek
ON worksgreek.GreekWorkID=paragraphsgreek.GreekWorkID
INNER JOIN charactersgreek
ON charactersgreek.GreekCharID=paragraphsgreek.GreekCharID
where paragraphsgreek.GreekCharID!='xxx' && worksgreek.WorkID='romeojuliet'
&& PlainText like ?
ORDER BY Title, Section, Chapter ASC

```

Σχήμα 6.12 Query αναζήτησης μιας λέξης

Πίσω στο component Search, επιστρέφονται τα αποτελέσματα σε μορφή λίστας και τελικά επιστρέφονται στην οθόνη ως κώδικας html. Στην περίπτωση του Search component δημιουργείται δυναμικά πίνακας.

```

<table className="table mg-left">
<thead className="thBGColor" >
  <tr>
    <th className="thColor" scope="col">#</th>
    <th className="thColor" scope="col">Title</th>
    <th className="thColor" scope="col">Scene</th>
    <th className="thColor" scope="col">Character</th>
    <th className="thColor" scope="col">Text</th>
  </tr>
</thead>
<tbody>
  {searchResults.map( (val,counter)=> {
    let translator="";
    let text=val.PlainText;
    text= text.replace(/\[p]/g, '\n');

```

Σχήμα 6.13 κώδικας React – render component

Search for a word

Σπίτι Search

Search Results for : Σπίτι

#	Title	Scene	Character	Text
0	ΡΩΜΑΙΟΣ ΚΑΙ ΙΟΥΔΑΙΟΣ-Μπελίας	1,1	ΠΡΙΓΚΙΠΑΣ	Ανυπόταχοι άνθρωποι, εχθροί της ειρήνης, που μαγαρίζετε το ατσάλι των σπαθιών σας με το αίμα των συμπολιτών σας! Εί, δεν μ' ακούν; Εί! Δεν είστε άνθρωποι εσείς, είστε θηρία ανήμερα, που σβήνετε τη φλόγα της ολέθριας μανίας σας στις πορφύρες πηγές που αναβλύζουν απ' τις φλέβες σας! Για να μην το πληρώσετε με άγρια βασανιστήρια, αφήστε τ' άνομα σπαθιά σας απ' τα ματωμένα χέρια σας κι ακούστε την οργισμένη απόφαση του ηγεμόνα σας. Τρεις εμφύλιες συγκρούσεις με ασήμαντες αιτίες έχετε προκαλέσει, γερο-Καπούλέτο και γερο-Μοντέγο, που ταραζάνε την ησυχία της πόλης μας και κάνουν τους σεβαστούς πολίτες της Βερόνας να βγάλουνε τα επιβλητικά κοσμήματά τους και ν' αρπάζουνε στα γερασμένα χέρια τους κοντάρια σκουριασμένα απ' την πολυχρονη ειρήνη για να σταματήσουν το δικό σας αρρωστημένο μίσος! Εάν ταραζέτε άλλη μία φορά τους δρόμους μας, θα το πληρώσετε με τη ζωή σας και οι δύο! Ετούτη τη φορά, όλοι οι υπόλοιποι διαλυθείτε. Καπουλέτο: εσύ θα 'ρθεις μαζί μου, κι εσύ εσύ, Μοντέγο, σε περιμένω το απόγευμα στο σπίτι μου, όπου δικάζω, κι εκεί θα μάθεις τις περαιτέρω αποφάσεις μου. Για τελευταία φορά: αν δεν διαλυθείτε αμέσως, θα πεθάνετε!
1	ΡΩΜΑΙΟΣ ΚΑΙ ΙΟΥΔΑΙΟΣ-Βυκίλας	1,1	ΣΑΜΨΩΝ	Κι' ένα σκυλί από το σπίτι του Μοντέκο με φέρνει άνω κάτω!
2	ΡΩΜΑΙΟΣ ΚΑΙ ΙΟΥΔΑΙΟΣ	1,1	ΣΑΜΨΩΝ	Δεν σαλεύω κ' εγώ αν απαντήσω κ' ένα σκυλί απ' εκείνο το σπίτι ή άνδρα, ή γυναίκα τύχω δεν σαλεύω!

Σχήμα 6.14 Εμφάνιση αποτελεσμάτων αναζήτησης

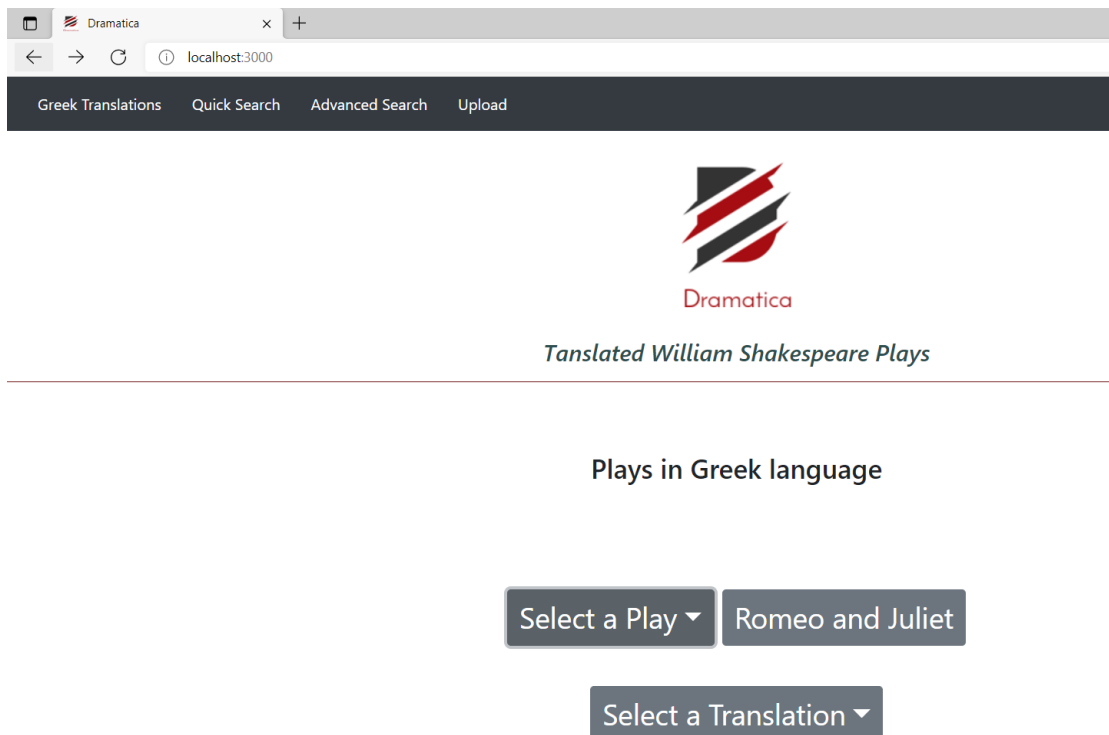
6.3 Deployment

Για να μπορεί κάποιος χρήστης να έχει πρόσβαση στην πλατφόρμα θα πρέπει πρώτα να γίνει το deployment της πλατφόρμας σε ένα cloud server. Δοκίμασα την δυνατότητα του deployment της εφαρμογής Dramatica στο server που παρέχει το Πανεπιστήμιο Κύπρου, ακολουθώντας τα πιο κάτω βήματα.

- Εκτέλεση της εντολής '\$ npm run build' στην κονσόλα. Δημιουργείται φάκελος build στο project, μέσα στον οποίο ο κώδικας μεταγλωττίστηκε και μπορεί να τρέξει.
- Άλλαξα στο αρχείο package.json, που βρίσκεται στο project directory της React JS, το dependency URL στη διεύθυνση του χώρου στο server του Πανεπιστημίου που μου δόθηκε.
- Δημιούργησα ένα .htaccess αρχείο για το configuration του Apache Server.
- Έκανα export και import τη βάση δεδομένων
- Μετέφερα στο server τον φάκελο build, το .htaccess και το αρχείο server.js του server node.js
- Εκτέλεσα την εντολή '\$ Node server.js' για την εκκίνηση του server node.js

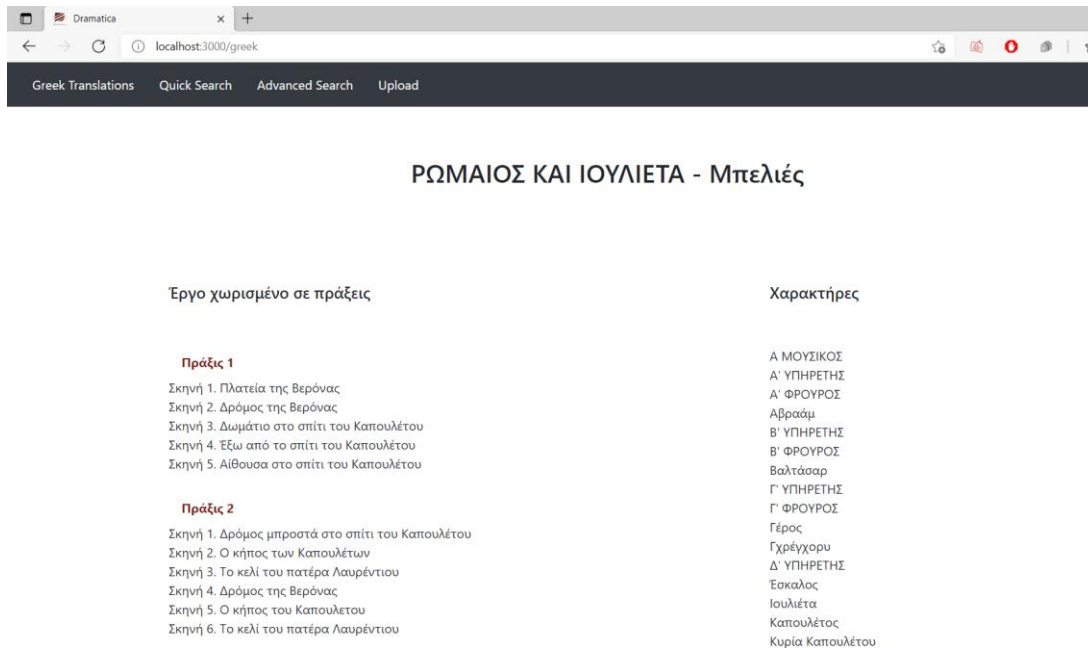
6.4 Παρουσίαση Τελικού Συστήματος

- Homepage



Σχήμα 6.15 Dramatica - homepage

- Preview Έργου

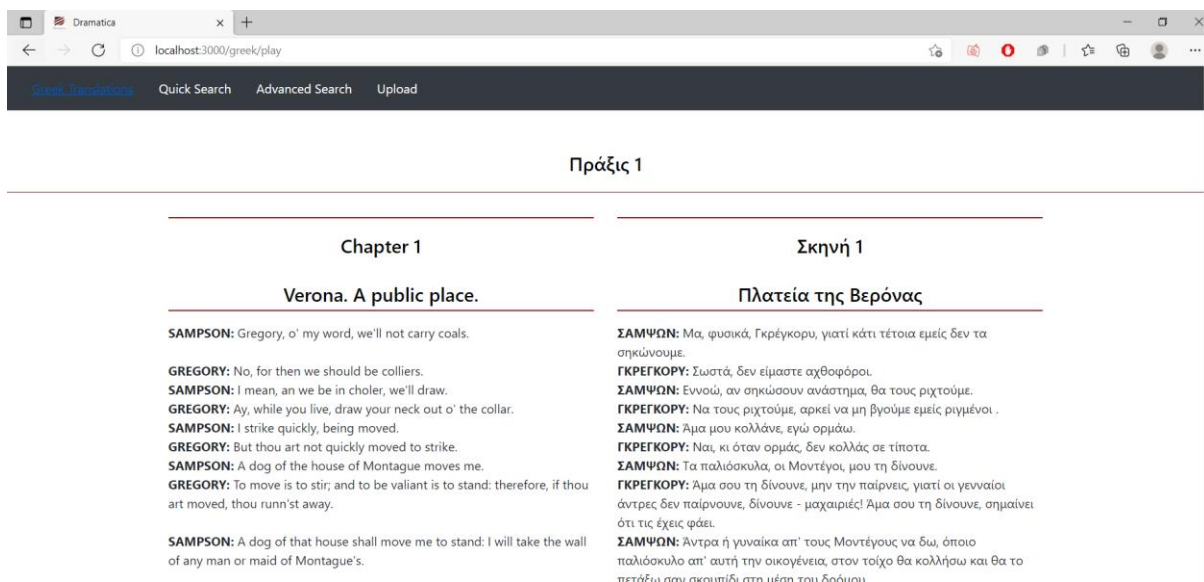


The screenshot shows the Dramatica website interface. At the top, there's a navigation bar with links: Greek Translations, Quick Search, Advanced Search, and Upload. The main heading is 'ΡΩΜΑΙΟΣ ΚΑΙ ΙΟΥΛΙΕΤΑ - Μπελιές'. Below this, there are two columns: 'Έργο χωρισμένο σε πράξεις' (Work divided into acts) and 'Χαρακτήρες' (Characters). The 'Έργο' column lists two acts with their respective scenes. The 'Χαρακτήρες' column lists the characters and their roles.

Έργο χωρισμένο σε πράξεις	Χαρακτήρες
Πράξις 1 Σκηνή 1. Πλατεία της Βερόνας Σκηνή 2. Δρόμος της Βερόνας Σκηνή 3. Δωμάτιο στο σπίτι του Καπουλέτου Σκηνή 4. Εξω από το σπίτι του Καπουλέτου Σκηνή 5. Αίθουσα στο σπίτι του Καπουλέτου	Α ΜΟΥΣΙΚΟΣ Α' ΥΠΗΡΕΤΗΣ Α' ΦΡΟΥΡΟΣ Αβραάμ Β' ΥΠΗΡΕΤΗΣ Β' ΦΡΟΥΡΟΣ Βολτάσαρ Γ' ΥΠΗΡΕΤΗΣ Γ' ΦΡΟΥΡΟΣ Γέρος Γκρέγκορυ Δ' ΥΠΗΡΕΤΗΣ Έσκαλος Ιουλιέτα Καπουλέτος Κυρία Καπουλέτου

Σχήμα 6.16 Dramatica - Work Preview

- Προβολή Αγγλικού κειμένου με μια Ελληνική μετάφραση



The screenshot shows the Dramatica website interface. At the top, there's a navigation bar with links: Greek Translations, Quick Search, Advanced Search, and Upload. The main heading is 'Πράξις 1'. Below this, there are two columns: 'Chapter 1' and 'Σκηνή 1'. The 'Chapter 1' column contains the English text of the first act. The 'Σκηνή 1' column contains the Greek text of the first act, including the characters' names and their roles.

Chapter 1	Σκηνή 1
Verona. A public place. SAMPSON: Gregory, o' my word, we'll not carry coals. GREGORY: No, for then we should be colliers. SAMPSON: I mean, an we be in choler, we'll draw. GREGORY: Ay, while you live, draw your neck out o' the collar. SAMPSON: I strike quickly, being moved. GREGORY: But thou art not quickly moved to strike. SAMPSON: A dog of the house of Montague moves me. GREGORY: To move is to stir; and to be valiant is to stand: therefore, if thou art moved, thou runn'st away. SAMPSON: A dog of that house shall move me to stand: I will take the wall of any man or maid of Montague's.	Πλατεία της Βερόνας ΣΑΜΨΩΝ: Μα, φυσικά, Γκρέγκορυ, γιατί κάτι τέτοια εμείς δεν τα σηκώνουμε. ΓΚΡΕΓΚΟΡΥ: Σωστά, δεν είμαστε αχθοφόροι. ΣΑΜΨΩΝ: Εννοώ, αν σηκώσουν ανάστημα, θα τους ριχτούμε. ΓΚΡΕΓΚΟΡΥ: Να τους ριχτούμε, αρκεί να μη βγούμε εμείς ριγμένοι. ΣΑΜΨΩΝ: Άμα μου κολλάνε, εγώ ορμάω. ΓΚΡΕΓΚΟΡΥ: Ναι, κι όταν ορμάς, δεν κολλάς σε τίποτα. ΣΑΜΨΩΝ: Τα παλιόσκυλα, οι Μοντέγοι, μου τη δίνουνε. ΓΚΡΕΓΚΟΡΥ: Άμα σου τη δίνουνε, μην την παίρνεις, γιατί οι γενναίοι άντρες δεν παίρνουνε, δίνουνε - μαχαίριές! Άμα σου τη δίνουνε, σημαίνει ότι τις έχεις φάει. ΣΑΜΨΩΝ: Άντρα ή γυναίκα απ' τους Μοντέγους να δω, όποιο παλιόσκυλο απ' αυτή την οικογένεια, στον τοίχο θα κολλήσω και θα το πετάξω σου περπατώντας στο μέση του δρόμου!

Σχήμα 6.17 Dramatica – 2 texts presentation

- Προβολή Αγγλικού κειμένου με δυο Ελληνικές μεταφράσεις

Greek Translations
Quick Search
Advanced Search
Upload

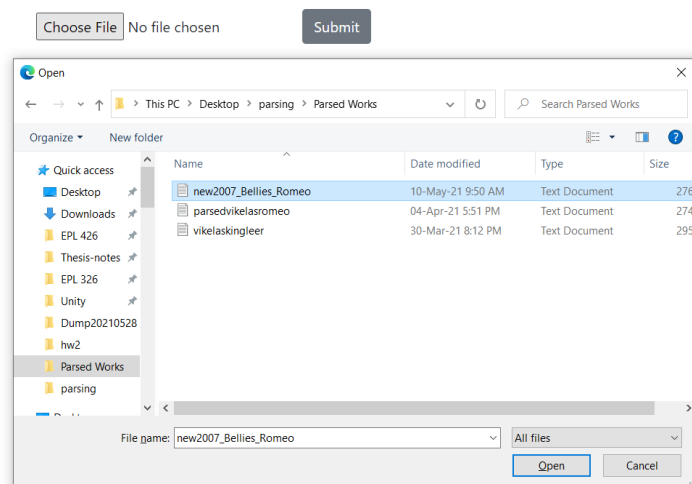
Πράξις 1

Chapter 1	Σκηνή 1	Σκηνή 1
Verona. A public place.	Πλατεία.	Πλατεία της Βερόνας
<p>SAMPSON: Gregory, o' my word, we'll not carry coals.</p> <p>GREGORY: No, for then we should be colliers.</p> <p>SAMPSON: I mean, an we be in choler, we'll draw.</p> <p>GREGORY: Ay, while you live, draw your neck out o' the collar.</p> <p>SAMPSON: I strike quickly, being moved.</p> <p>GREGORY: But thou art not quickly moved to strike.</p> <p>SAMPSON: A dog of the house of Montague moves me.</p>	<p>ΣΑΜΨΩΝ: Γρηγόρη, μα την πίστιν μου ούτ' εγώ τρώγω άχυρα, ούτε συ!</p> <p>ΓΡΗΓΟΡΗΣ: Όχι βέβαια! μήπως είμεθα γαιδούρια;</p> <p>ΣΑΜΨΩΝ: Θέλω να ειπώ, αν μας θυμώσουν να πηδήσωμεν κατ' επάνω των.</p> <p>ΓΡΗΓΟΡΗΣ: Κύτταξε όμως να μη πηδάς πολλά παλούκια.</p> <p>ΣΑΜΨΩΝ: Δεν αργώ να κτυπήσω, αν πάρω φωτιά!</p> <p>ΓΡΗΓΟΡΗΣ: Αλλ' αργείς να πάρης φωτιά, διά να κτυπήσης.</p> <p>ΣΑΜΨΩΝ: Κι' ένα σκυλί από το σπίτι του Μοντέκη με φέρνει άνω κάτω!</p>	<p>ΣΑΜΨΩΝ: Μα, φυσικά, Γκρέγκορι, γιατί κάτι τέτοια εμείς δεν τα σηκώνουμε.</p> <p>ΓΚΡΕΓΚΟΡΥ: Σωστά, δεν είμαστε αχθοφόροι.</p> <p>ΣΑΜΨΩΝ: Εννοώ, αν σηκώσουν ανάστημα, θα τους ριχτούμε.</p> <p>ΓΚΡΕΓΚΟΡΥ: Να τους ριχτούμε, αρκεί να μη βγούμε εμείς ριγμένοι.</p> <p>ΣΑΜΨΩΝ: Άμα μου κολλάνε, εγώ ορμάω.</p> <p>ΓΚΡΕΓΚΟΡΥ: Ναι, κι όταν ορμάς, δεν κολλάς σε τίποτα.</p> <p>ΣΑΜΨΩΝ: Τα παλιόσκυλα, οι Μοντέγοι, μου τη δίνουνε.</p>

Σχήμα 6.18 Dramatica – 3 texts presentation

- Επιτυχημένη μεταφόρτωση αρχείου

Upload a Translation



Σχήμα 6.19 Dramatica – file upload

Upload a Translation

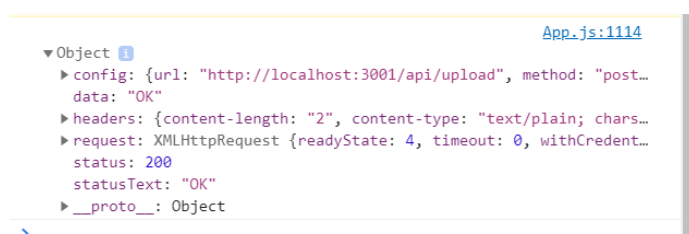
Choose File

new2007_Be...s_Romeo.txt

Submit

Upload Successful!

Σχήμα 6.20 Dramatica – successful upload



Σχήμα 6.21 Dramatica – browser log / successful upload

- Απλή Αναζήτηση Αγγλικής λέξης

Search for a word

house

Search

Search Results for : house

#	Title	Scene	Character	Text
0	Romeo and Juliet	1,1	SAMPSON	A dog of the house of Montague moves me.
1	Romeo and Juliet	1,1	SAMPSON	A dog of that house shall move me to stand: I will take the wall of any man or maid of Montague's.
2	Romeo and Juliet	1,1	GREGORY	'Tis well thou art not fish; if thou hadst, thou hadst been poor John. Draw thy tool! here comes two of the house of the Montagues.

Σχήμα 6.22 Dramatica – Simple search

- Σύνθετη αναζήτηση Ελληνικής λέξης

Advanced Search

Select Play ▾

All Plays

Select Character ▾

Juliet

Select Act ▾

5

Select Scene ▾

3

Search

#	Title	Scene	Character	Greek Text	English Text
0	ΡΩΜΑΙΟΣ ΚΑΙ ΙΟΥΛΙΕΤΑ- Βικέλας	5,3	ΙΟΥΛΙΕΤΑ	Πάτερ, ω πάτερ μου παρήγορε, ο άνδρας μου πώς είναι; Το ενθυμούμαι καθαρά πού έπρεπε να ήμαι· ηξεύρω πού ευρίσκομαι. Πού είναι ο Ρωμαίος ;	O comfortable friar! where is my lord? I do remember well where I should be, And there I am. Where is my Romeo?
1	ΡΩΜΑΙΟΣ ΚΑΙ ΙΟΥΛΙΕΤΑ- Μπελιές	5,3	ΙΟΥΛΙΕΤΑ	Αχ, τι παρηγοριά που είστε δω, πατέρα. Πού είναι ο κύριός μου; Ξέρω καλά πού έπρεπε να είμαι, κι εδώ βρίσκομαι. Πού είναι ο Ρωμαίος μου;	O comfortable friar! where is my lord? I do remember well where I should be, And there I am. Where is my Romeo?

Σχήμα 6.23 Dramatica – Advanced Search

Κεφάλαιο 7

Συμπεράσματα

7.1 Αξιολόγηση και Έλεγχος Πλατφόρμας	50
7.2 Αξιολόγηση Διπλωματικής Εργασίας	51
7.3 Μελλοντικές Επεκτάσεις	52

Αξιολόγηση και Έλεγχος Πλατφόρμας

Σε πρώτο στάδιο έκανα εγώ προσωπικά ελέγχους για πιθανά σφάλματα ή bugs. Έλεγχα για παράδειγμα ότι κάθε API δουλεύει όπως πρέπει, έλεγχα για τους συνδέσμους ότι δεν προκύπτει κάποιο ανεπιθύμητο αποτέλεσμα και δοκίμασα πολλά διαφορετικά inputs για τη λειτουργία της αναζήτησης. Έτρεξα την εφαρμογή από τους browsers Google Chrome, Microsoft Edge, Mozilla Firefox και από smartphone Xiaomi Redmi.

Σε ένα δεύτερο στάδιο στη φάση της αξιολόγησης, μέσω online κλήσης έκανα μια επίδειξη του συστήματος στον κ. Κρίτη, αφού είναι ο άνθρωπος που μας έθεσε τις βασικές απαιτήσεις και αποτελεί μέρος σε από τα κύρια target groups, τους ερευνητές. Η γνώμη του και τα σχόλια του είναι λοιπόν ιδιαίτερα σημαντικά.

Σαν αρχικό σύστημα και περιβάλλον, που καλύπτει τις βασικές απαιτήσεις που θέσαμε, ο κ. Κρίτης ήταν ευχαριστημένος. Πρότεινε όμως τις εξής διορθώσεις, τις οποίες θα υλοποιήσω:

1. Το Αγγλικό κείμενο στην αναζήτηση να είναι πάντα πριν τα Ελληνικά κείμενα
2. Οι μεταφράσεις να διαχωρίζονται οριζόντια και όχι κάθετα
3. Η αρίθμηση των αποτελεσμάτων να ξεκινά από το 1 και όχι από το 0
4. Ο αριθμός Πράξης να είναι στα λατινικά, όχι με αραβική αρίθμηση
5. Εφόσον υπάρχει σαν δεδομένο, να υπάρχει η ημερομηνία μετάφρασης για κάθε μετάφραση

Εξέφρασε επιπρόσθετα μια αδυναμία του συστήματος. Όταν του εξήγησα τη διαδικασία αποθήκευσης κειμένων στη Βάση Δεδομένων, τον προβλημάτισε το ότι ο προγραμματιστής ή ο διαχειριστής του συστήματος επιβαρύνεται με το φόρτο εργασίας του ερευνητή. Όχι μόνο αυτή η διαδικασία είναι δύσκολη και εκνευριστική για τον προγραμματιστή, αλλά πιθανόν να κάνει λάθη κατά τη manual επεξεργασία, γιατί δε γνωρίζει τα κείμενα το ίδιο καλά όσο οι ερευνητές. Στην περίπτωση μετάφρασης των κειμένων σε άλλες γλώσσες όπως τα Ρωσικά τότε ο προγραμματιστής ίσως να μην μπορεί καν να ολοκληρώσει τη μετάφραση. Άρα θα χρειαστεί μια έξυπνη προσέγγιση για τη λύση αυτού του προβλήματος.

Τέλος ο κ. Κρίτσης πρότεινε τη δυνατότητα αναζήτησης σε περισσότερες μεταφράσεις ταυτόχρονα. Στα πλαίσια της ΑΔΕ υπάρχει το Αγγλικό κείμενο μαζί με 2 Ελληνικές μεταφράσεις, ο κ. Κρίτσης πρότεινε να μπορούν να παρουσιάζονται τα αποτελέσματα 10, 20 ή ακόμα και περισσότερων μεταφράσεων, εάν εφόσον υπάρχουν στη Βάση Δεδομένων.

7.1 Αξιολόγηση Εργασίας

Ο στόχος της Διπλωματικής μου Εργασίας ήταν η κατασκευή ενός χρήσιμου εργαλείου που να μπορεί να καλύπτει τις ανάγκες των ερευνητών του Σαίξπηρ, έτσι ώστε να φύγουν από το παλιό σύστημα της χειρόγραφης μελέτης και να επωφεληθούν από την τεχνολογία. Πιστεύω πως από την αρχή χρησιμοποίησα τις κατάλληλες τεχνικές και μεθόδους. Σε κάθε στάδιο σκεφτόμουν πώς η κάθε αλλαγή που θα έκανα θα επηρέαζε την μελλοντική κατάσταση του συστήματος και με ποιο τρόπο θα μπορούσα να κάνω βελτιώσεις. Ήταν σημαντικό να κατανοήσω καλά τις απαιτήσεις, κάτι που κατάφερα μέσα από τις συναντήσεις που συμμετείχα μαζί με την κ. Καπιτσάκη και τον κ. Κρίτση.

Μπορώ να πω ότι συνάντησα αρκετές δυσκολίες και ίσως έκανα κάποιες λάθος επιλογές που μου κόστισαν πολύ σε χρόνο. Επέλεξα να χρησιμοποιήσω εργαλεία, γλώσσες προγραμματισμού και βιβλιοθήκες τις οποίες δεν ξαναχρησιμοποίησα στο παρελθόν. Για παράδειγμα η Perl, η Node αλλά κυρίως η React, γιατί ο τρόπος που λειτουργεί είναι πολύ διαφορετικός από μια ιστοσελίδα χτισμένη με JavaScript και php, τεχνολογίες που διδάχτηκα και στις σπουδές μου. Επίσης ήρθα αντιμέτωπος με προκλήσεις σχετικά με την κάλυψη των απαιτήσεων. Συγκεκριμένα για την τριπλή προβολή κειμένου και την εξειδικευμένη αναζήτηση χρειάστηκε να υλοποιήσω μεγάλα και πολύπλοκα queries.

Επίσης η επινόηση και η εφαρμογή ολόκληρης της μεθοδολογίας μου πήρε αρκετό χρόνο. Ακόμη και τώρα υπάρχει ένα μεγάλο πρόβλημα. Οι Ελληνικές μεταφράσεις βρίσκονται σε

μια αρκετά κακή κατάσταση σε σχέση με τα Αγγλικά κείμενα. Υπάρχουν πολλά λανθασμένα σύμβολα κάτι που κάνει εξαιρετικά δύσκολη την αυτοματοποίηση της διαδικασίας του parsing που εξήγησα στο Κεφάλαιο 5. Για εμένα που έμαθα πλέον καλά τη διαδικασία, δεν υπάρχει κάποια δυσκολία. Αν μεταφέρω τις γνώσεις μου σε έναν επαγγελματία προγραμματιστή, πιστεύω πως μπορεί να ανεβάσει περισσότερα κείμενα και να βελτιώσει την πλατφόρμα. Ένας όμως ερευνητής του Σαίξπηρ, δεν θα μπορεί να τα ανεβάσει ο ίδιος. Το θετικό είναι ότι οι μεταφράσεις είναι λίγες και περιορισμένες, άρα μπορούν να μουν όλες στην αρχή από ένα άτομο.

Παρά τις δυσκολίες, έχω να πω ότι τελειώνοντας αυτή την Διπλωματική Εργασία, νιώθω καλύτερος και βελτιωμένος. Έμαθα από κάποια λάθη που έκανα. Απέκτησα πολλές γνώσεις στην περιοχή της Τεχνολογίας Λογισμικού και συγκεκριμένα στο Web Development. Όσον αφορά το αποτέλεσμα της Web Εφαρμογής, μπορώ να πω ότι είμαι ευχαριστημένος. Οι ζητούμενες λειτουργίες έχουν υλοποιηθεί επιτυχώς και η πλατφόρμα είναι ελκυστική και αποδοτική.

7.2 Μελλοντικές Επεκτάσεις

Ενώ είμαι ευχαριστημένος με το αποτέλεσμα, είμαι σίγουρος ότι η πλατφόρμα μπορεί να βελτιωθεί σημαντικά. Αρχικά, κάτι που ήθελα να υλοποιήσω αλλά δεν τα κατάφερα λόγω της έλλειψης γνώσεων που έχω στην React, ήταν ένα σύστημα login. Θα ήθελα να υπάρχουν ειδικά user accounts για τους έμπιστους χρήστες και να υπάρχει ένα επιπλέον στρώμα ασφάλειας.

Φυσικά μπορούν να μουν περισσότερες Ελληνικές μεταφράσεις έργων του Σαίξπηρ, αφού ασχολήθηκα μόνο με τις 2 μεταφράσεις του Ρωμαίου και της Ιουλιέτας και σε ένα πιο αρχικό στάδιο με τον Βασιλεία Αηρ. Τα Αγγλικά κείμενα είναι ήδη όλα στη βάση.

Εκτός από Ελληνικές μεταφράσεις θα ήταν καλό να μουν και άλλες γλώσσες και να γίνεται σύγκριση μεταξύ των διάφορων μεταφράσεων.

Μια ακόμη επιθυμία του κ. Κρίτση ήταν να υπάρχουν στα έργα σύνδεσμοι με πληροφορίες και βίντεο θεατρικών παραστάσεων. Η απαίτηση αυτή όμως απορρίφθηκε σε ένα αρχικό στάδιο, λόγω του περιορισμένου χρόνου και της έλλειψης υλικού.

Βιβλιογραφία

- [1] The Complete Works of William Shakespeare
<http://shakespeare.mit.edu/>
- [2] No Sweat Shakespeare
<https://www.nosweatshakespeare.com/>
- [3] The Complete Works of Shakespeare
<https://www.thecompleteworksofshakespeare.com/home>
- [4] LitCharts
<https://www.litcharts.com/>
- [5] OpenSource Shakespeare
<https://www.opensourceshakespeare.org/>
- [6] What is open source software?
<https://opensource.com/resources/what-open-source>
- [7] Sommerville, I., (2016). Software Engineering 10th Edition
PEARSON.
- [8] Usability – ISO9241
<https://www.iso.org/obp/ui/#iso:std:iso:9241:-210:ed-2:v1:en>
- [9] What is SQL Injection?
<https://portswigger.net/web-security/sql-injection>
- [10] Sarjeant, J. V., & Andrews A. A. (2015). Comparing Reuse Strategies in Different Development Environments.
<https://www.sciencedirect.com/science/article/pii/S0065245814000035>
- [11] Perl in 2020: Is It Still Worth Learning Now?

<https://somedudesays.com/2020/02/perl-in-2020-is-it-still-worth-learning/>

- [12] Gibb, R. (2016). What is a Web Application?
<https://blog.stackpath.com/web-application/>
- [13] Web Server vs Application Server
<https://www.educative.io/edpresso/web-server-vs-application-server>
- [14] HTML: HyperText Markup Language
<https://developer.mozilla.org/en-US/docs/Web/HTML>
- [15] CSS: Cascading Style Sheets
<https://developer.mozilla.org/en-US/docs/Web/CSS>
- [16] JavaScript
<https://developer.mozilla.org/en-US/docs/Web/JavaScript>
- [17] Sufiyan, T. (2021). What is ReactJS: Introduction To React and Its Features
<https://www.simplilearn.com/tutorials/reactjs-tutorial/what-is-reactjs>
- [18] React Router
<https://reactrouter.com/web/guides/quick-start>
- [19] React axios
<https://www.npmjs.com/package/react-axios>
- [20] Bootstrap
<https://getbootstrap.com>
- [21] Node.js
<https://nodejs.org/en/about/>
- [22] Express
<https://expressjs.com/>
- [23] CORS

<https://developer.mozilla.org/en-US/docs/Web/HTTP/CORS>

- [24] Ramirez, N. (2019). How Does CORS work?
<https://www.haproxy.com/blog/enabling-cors-in-haproxy/>

- [25] Express-fileupload
<https://www.npmjs.com/package/express-fileupload>

- [26] nodemon
<https://www.npmjs.com/package/nodemon>

- [27] MySQL
www.mysql.com

- [28] RESTful Services
<https://www.guru99.com/restful-web-services.html>

- [29] Postman
<https://www.postman.com/>

- [30] npm
<https://www.npmjs.com/>

- [31] MySQL Workbench
<https://www.mysql.com/products/workbench/>

- [32] Visual Studio Code
<https://code.visualstudio.com/>

Παράρτημα Α

Σύνδεσμος Web App – Dramatica: <http://thesis.in.cs.ucy.ac.cy/dramatica/>

Σύνδεσμος GitHub με τον κώδικα της εφαρμογής: [socyi/Dramatica · GitHub](https://github.com/socyi/Dramatica)