



ΑΛΕΞΑΝΔΡΕΙΟ Τ.Ε.Ι. ΘΕΣΣΑΛΟΝΙΚΗΣ
ΣΧΟΛΗ ΤΕΧΝΟΛΟΓΙΚΩΝ ΕΦΑΡΜΟΓΩΝ
ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ



ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

ΣΧΕΔΙΑΣΗ ΚΑΙ ΑΝΑΠΤΥΞΗ ΣΥΣΤΗΜΑΤΟΣ ΠΡΟΣΒΑΣΙΜΟΤΗΤΑΣ ΓΙΑ ΑΜΕΑ

Της φοιτήτριας
Νιβορλή Πασχαλιάς
Αρ. Μητρώου: 05/2849

Επιβλέπων καθηγητής
Μιχάλης Σαλαμπάσης

Θεσσαλονίκη, Φεβρουάριος 2012

Ευχαριστίες

Θέλω να ευχαριστήσω τον Janik La Grange για την πολύτιμη βοήθειά του που ήταν τόσο μέσω σχολίων και παρατηρήσεων όσο και μέσω ψυχολογικής υποστήριξης κατά την εκπόνηση αυτής της πτυχιακής εργασίας. Επίσης θέλω να ευχαριστήσω την Αναστασία Τάντση για την βοήθειά της και τις διορθώσεις της κυρίως σε γραμματικά και συντακτικά θέματα που αφορούσαν την πτυχιακή. Τέλος θα ήθελα να ευχαριστήσω όσους ήταν δίπλα μου και με στήριξαν όλο αυτό το διάστημα.

Πίνακας περιεχομένων

Πίνακας περιεχομένων	3
Ευρετήριο Σχημάτων	5
Ευρετήριο Πινάκων	6
1. Εισαγωγή	7
2. Front ends.....	10
2.1. Front end γλώσσες προγραμματισμού.....	10
2.1.1. HTML και XHTML	11
2.1.2. Javascript.....	15
2.1.3. CSS.....	17
2.2. Front end frameworks.....	21
2.2.1. jQuery	21
2.2.2. Easy.....	25
3. Javascript και HTML generator Frameworks	27
3.1. Google Web Toolkit.....	27
3.1.1. GWT και προσβασιμότητα	38
3.2. Pyjamas	40
3.2.1. Σύγκριση του Pyjamas με το GWT	42
3.3. Morfik BX	43
3.3.1. Σύγκριση του Morfik BX με το GWT	45
3.4. Σύγκριση των Javascript και HTML generator frameworks με τα υπόλοιπα.....	45
4. Προσβασιμότητα στον παγκόσμιο ιστό.....	47
4.1. Διάφορα είδη αναπηρίας	52
4.1.1. Προβλήματα όρασης.....	52
4.1.2. Προβλήματα Ακοής	55
4.1.3. Σωματικές αναπηρίες.....	56
4.1.4. Λεκτικά προβλήματα	57
4.1.5. Γνωστική και νευρολογική αναπηρία	57
4.1.6. Πολλαπλές διαταραχές	60
4.1.7. Συμπτώματα που σχετίζονται με την γήρανση	61
4.2. Τυφλοί χρήστες.....	61
4.2.1. Screen readers	62

4.2.2. Περιορισμοί των screen readers	63
4.2.3. Προσβασιμότητα πληκτρολογίου	64
4.2.4. Βασικές έννοιες	65
5. Εφαρμογή	68
5.1. Τα προβλήματα που παρουσιάζονται	69
5.2. Οι δοθείσες λύσεις.....	71
6. Συμπεράσματα.....	75
Βιβλιογραφία	77
Websites	77
Βιβλία.....	78

Ευρετήριο Σχημάτων

Εικόνα 1 – New Web Application Project	36
Εικόνα 2 – Create a Web Application Project	37
Σχήμα 3 – Μετατροπή από python σε Javascript	41
Εικόνα 4 – Braille συσκευή	49
Εικόνα 5 – Ειδικά πληκτρολόγια	50
Εικόνα 6 – Χαμηλή οξύτητα, όραση τούνελ και απώλεια κεντρικού οράματος	53
Εικόνα 7 – Κουμπί check availability	69
Εικόνα 8 – Κουμπί check availability και μήνυμα λάθους	70
Εικόνα 9 – Κουμπί check availability και is available μήνυμα	70

Ευρετήριο Πινάκων

Πίνακας 1 – Κατάλογοι του GWT	28
Πίνακας 2 – Αρχεία του GWT	26
Πίνακας 3 - Βασικές έννοιες που αφορούν την τύφλωση	65

1. Εισαγωγή

Με το πέρασμα των χρόνων όλο και περισσότερη έμφαση δίνεται στην δυνατότητα πρόσβασης όλων των χρηστών στον παγκόσμιο ιστό. Άλλωστε, όπως ανέφερε και ο Tim Berners-Lee, ο εφευρέτης του παγκόσμιου ιστού, “Η δύναμη του παγκόσμιου ιστού βρίσκεται στην καθολικότητά του. Η πρόσβαση από τον καθένα ανεξάρτητα από την αναπηρία που αντιμετωπίζει είναι μία ουσιώδης έννοιά του.”

Προσβασιμότητα στον παγκόσμιο ιστό σημαίνει ότι ακόμα και οι άνθρωποι με ειδικές ανάγκες και αναπηρία μπορούν να τον χρησιμοποιήσουν. Πιο συγκεκριμένα ότι μπορούν να αντιληφθούν, να καταλάβουν, να πλοηγηθούν και να αλληλεπιδράσουν με τον παγκόσμιο ιστό ανεξάρτητα από το είδος της αναπηρίας τους. Μερικά από τα είδη αναπηρίας που μπορούν να αναφερθούν είναι τα παρακάτω:

- Οπτική αναπηρία, στην οποία συμπεριλαμβάνεται η ολική τύφλωση, η χαμηλή όραση και η αχρωματοψία
- Βλάβη στην ακοή, στην οποία συμπεριλαμβάνεται η κώφωση και η βαρηκοΐα
- Φυσική αναπηρία, στην οποία συμπεριλαμβάνεται η δυσκινησία
- Προφορική αναπηρία
- Γνωστική και νευρολογική αναπηρία, στην οποία συμπεριλαμβάνεται η δυσλεξία και η δυσαριθμησία, το σύνδρομο ελλειμματικής προσοχής, οι μαθησιακές δυσκολίες, η απώλεια μνήμης, η νοητική καθυστέρηση και οι επιληπτικές διαταραχές
- Πολλαπλή αναπηρία
- Καταστάσεις που σχετίζονται με την γήρανση

Επίσης η προσβασιμότητα σχετίζεται και με προσωρινή αναπηρία όπως για παράδειγμα ένα σπασμένο χέρι ή ακόμα και με ηλικιωμένους ανθρώπους καθώς οι ικανότητές τους αλλάζουν.

Στην συγκεκριμένη πτυχιακή θα εστιάσουμε στους ανθρώπους με προβλήματα όρασης και κυρίως στους τυφλούς χρήστες. Η τύφλωση αναφέρεται σε ανθρώπους που έχουν ολική και μόνιμη έλλειψη όρασης και στα δύο μάτια.

Για να έχουν πρόσβαση στον παγκόσμιο ιστό, πολύ άνθρωποι που είναι τυφλοί βασίζονται στους screen readers που είναι λογισμικό που διαβάζει κείμενο από την οθόνη του υπολογιστή και το εξάγει σε προφορική ομιλία ή/και σε braille μορφή που ανανεώνεται. Μερικοί τυφλοί χρησιμοποιούν επίσης text-based browsers, όπως ο Lynx, ή voice browsers αντί για browsers με γραφικό περιβάλλον σε συνδυασμό με κάποιο screen reader. Επίσης πολλές φορές χρησιμοποιούν στρατηγικές γρήγορης πλοήγησης όπως η πλοήγηση μέσα στις κεφαλίδες και στους υπερσυνδέσμους μιας ιστοσελίδας μέσω του πλήκτρου tab. Όλ' αυτά όμως θα τα αναφέρουμε πιο αναλυτικά σε μετέπειτα κεφάλαιο που θα αναφερθούμε διεξοδικά στην προσβασιμότητα και κυρίως στους τυφλούς χρήστες.

Για να διασφαλίσουμε ότι υπάρχει καλή προσβασιμότητα σε όλες τις λειτουργίες της εφαρμογής και για όλα τα είδη χρηστών πρέπει να αναπτυχθούν φιλικά user interfaces. Για να επιτευχθεί αυτό είναι απαραίτητη η χρήση front-ends τεχνολογιών.

Στο κεφάλαιο 2, θα αναφερθούμε σε front end γλώσσες προγραμματισμού και frameworks. Θα δώσουμε έναν ορισμό για το τι σημαίνει front end και θα αναφέρουμε σε μερικές front end γλώσσες προγραμματισμού, όπως η HTML και η CSS, και σε μερικά front end frameworks που μπορούν να χρησιμοποιηθούν για τη δημιουργία web εφαρμογών στον παγκόσμιο ιστό. Τέλος θ' ασχοληθούμε με το πώς μέσω αυτών των frameworks μπορούμε να έχουμε καλό usability καθώς και με τις λύσεις που μπορούν να δώσουν στην προσβασιμότητα ατόμων με ειδικές ανάγκες στον παγκόσμιο ιστό.

Στο κεφάλαιο 3 θα αναλύσουμε το Google Web Toolkit (GWT), ένα εργαλείο της Google για δημιουργία browser-based εφαρμογών. Το GWT δίνει την δυνατότητα στους προγραμματιστές να γράφουν AJAX εφαρμογές σε Java και μετά να κάνουν compile τον πηγαίο κώδικα σε Javascript που τρέχει σε όλους τους browsers, συμπεριλαμβάνοντας browsers για Android κινητά και iPhone. Επίσης θ' αναφερθούμε στο Pyjamas που είναι ένα εργαλείο που βασίστηκε στο GWT αλλά χρησιμοποιεί γλώσσα python αντί της Java καθώς και στο Morfik BX που

χρησιμοποιεί Basic. Θα παρουσιάσουμε τις ιδιαιτερότητές τους και πώς λειτουργούν, ενώ παράλληλα θα τα συγκρίνουμε και μεταξύ τους.

Επίσης, ένα ακόμα ζήτημα που θα αντιμετωπιστεί στην παρούσα πτυχιακή, όπως ήδη αναφέραμε, είναι η προσβασιμότητα. Στο κεφάλαιο 4 θα εξηγήσουμε τι ακριβώς εννοούμε όταν αναφερόμαστε σε προσβασιμότητα και θα αναφερθούμε στις διάφορες κατηγορίες ανθρώπων με ειδικές ανάγκες και στα προβλήματα που αντιμετωπίζουν κατά την πλοήγησή τους στον παγκόσμιο ιστό. Τέλος θα κάνουμε μία εκτενέστερη αναφορά στους τυφλούς χρήστες. Θα αναφερθούμε σε διάφορα εργαλεία και προγράμματα που χρησιμοποιούν και καθιστούν δυνατή τη πρόσβασή τους στο διαδίκτυο.

Στο κεφάλαιο 5 θα αναφερθούμε σε μία εφαρμογή που είναι σχεδιασμένη και υλοποιημένη με το GWT. Η συγκεκριμένη εφαρμογή είναι ένα case study που αφορά τυφλούς χρήστες και τα προβλήματα που αυτοί αντιμετωπίζουν στην δημιουργία ενός λογαριασμού e-mail κατά την διαδικασία της εγγραφής τους. Αυτό το case study βασίστηκε σε ένα υπάρχον βίντεο και μέσα από αυτή την εφαρμογή παρέχονται λύσεις και βελτιώσεις στα προβλήματα που αντιμετωπίζουν.

Τέλος, στο κεφάλαιο 6 θα αναφερθούν τα προβλήματα που παρουσιάστηκαν κατά την διάρκεια αυτής της πτυχιακής καθώς και τα συμπεράσματα που προέκυψαν απ' αυτήν.

2. Front ends

Στο κεφάλαιο αυτό θα ασχοληθούμε με τις front end γλώσσες προγραμματισμού και frameworks. Τι ακριβώς εννοούμε όμως όταν αναφερόμαστε σε front ends και σε frameworks; Στην επιστήμη των υπολογιστών όταν αναφερόμαστε σε front ends αναφερόμαστε στο client side μέρος της εφαρμογής το οποίο καθορίζει το οπτικό κομμάτι καθώς και την συμπεριφορά της εφαρμογής. Επίσης, το front end κομμάτι μιας εφαρμογής ασχολείται με την cross-browser, την cross-platform και την cross-device λειτουργικότητα της εφαρμογής καθώς και με την προσβασιμότητα. Το web application framework είναι ένα σύνολο από βιβλιοθήκες και κλάσεις για την δημιουργία δυναμικών ιστοσελίδων, web εφαρμογών και web services.

Επίσης, στο κεφάλαιο αυτό θ' αναφερθούμε σε όλ' αυτά σε συνδυασμό με την προσβασιμότητα. Η W3C (World Wide Web Consortium) είναι μία διεθνής κοινότητα όπου οι οργανώσεις - μέλη, το πλήρης απασχόλησης προσωπικό και το κοινό δουλεύουν για να αναπτύξουν πρότυπα στον παγκόσμιο ιστό. Η πρωτοβουλία για την πρόσβαση στο web (WAI - Web Accessibility Initiative) έχει δημοσιεύσει οδηγίες προσβασιμότητας (WCAG - Web Content Accessibility Guidelines) για το web που βοηθούν στην δημιουργία web περιεχομένου που είναι προσβάσιμο από ανθρώπους με αναπηρία. Μέσα σ' αυτές τις οδηγίες μπορεί να βρει κανείς κάποιους κανόνες και πώς μπορούν αυτοί να τηρηθούν και να υλοποιηθούν μέσω διαφορετικών τεχνικών και γλωσσών.

2.1. Front end γλώσσες προγραμματισμού

Σ' αυτό το υποκεφάλαιο θ' ασχοληθούμε με τις front end γλώσσες προγραμματισμού. Οι βασικότερες απ' αυτές είναι οι HTML, Javascript, CSS, XHTML, DHTML και Flash.

2.1.1. HTML και XHTML

Τα αρχικά HTML σημαίνουν HyperText Markup Language, δηλαδή γλώσσα σήμανσης υπερκειμένου. Η HTML χρησιμοποιείται μόνο για να προσδιορίσει το περιεχόμενο του εγγράφου και τίποτα παραπάνω. Αυτός είναι και ο λόγος που δεν θεωρείται γλώσσα προγραμματισμού. Οι γλώσσες προγραμματισμού είναι πολύ πιο ισχυρές καθώς μπορείς να υλοποιήσεις μ' αυτές όλα τα είδη των χρηστών, εργασιών και λογικών δεδομένων. Η HTML δεν είναι τίποτα παραπάνω από μια γλώσσα σημάτων. Γλώσσες σημάτων σαν την HTML είναι η LaTeX, η XML, η XHTML, η Textile και πολλές άλλες.

Υπάρχουν πολλές εκδόσεις της HTML. Τον Ιανουάριο του 2000 προστέθηκαν μερικοί πιο αυστηροί κανόνες στην HTML 4.01 και έτσι δημιουργήθηκε η XHTML (Extensible HyperText Markup Language). Ουσιαστικά η XHTML είναι η υλοποίηση της HTML 4.01 ως εφαρμογή της XML.

Οι κυριότερες διαφορές ανάμεσα στην HTML και την XHTML είναι οι εξής:

- Τα XHTML στοιχεία πρέπει να είναι σωστά φωλιασμένα
- Τα XHTML στοιχεία πρέπει πάντα να κλείνουν
- Τα XHTML στοιχεία πρέπει να γράφονται με μικρά γράμματα
- Τα XHTML αρχεία πρέπει να έχουν ένα στοιχείο ρίζα

Όπως φαίνεται και απ' τα ονόματά τους, η HTML και η XHTML είναι γλώσσες σημάτων, το οποίο μπορεί να ακούγεται περίπλοκο μέχρι να σκεφτεί κανείς ότι ουσιαστικά συναντάμε σημάσεις κάθε μέρα. Όταν δημιουργούμε ένα έγγραφο σε έναν επεξεργαστή κειμένου μπορούμε να προσθέσουμε στυλ και μορφοποίηση στο κείμενο για να εξηγήσουμε καλύτερα την δομή του. Για παράδειγμα, μπορούμε να ξεχωρίσουμε τις επικεφαλίδες από το κυρίως κείμενο του κειμένου χρησιμοποιώντας στυλ επικεφαλίδας, το οποίο είναι συνήθως μεγαλύτερου μεγέθους. Μπορούμε να χρησιμοποιήσουμε το κουμπί enter για να ξεκινήσουμε μία νέα παράγραφο. Μπορούμε να εισάγουμε πίνακες μέσα στο κείμενό μας για να βάλουμε μέσα δεδομένα ή να δημιουργήσουμε λίστες από κουκίδες για να διατηρήσουμε μία σειρά σε σχετικά πράγματα κ.ο.κ.. Ο κύριος σκοπός χρήσης αυτών των σημάτων είναι για να παρέχει μία δομή στο κείμενο που θα το κάνει πιο εύκολα αναγνώσιμο.

Όταν χρησιμοποιούμε σημάνσεις για αρχεία του web εκτελούμε μια παρόμοια διαδικασία, μόνο που χρησιμοποιούμε αντικείμενα τα οποία ονομάζονται σημάνσεις. Αυτό ισχύει τόσο για την HTML όσο και για την XHTML.

Όσον αφορά την προσβασιμότητα, η WAI αναζητούσε λύσεις σε 4 προβλήματα:

1. Τεχνικές ανάπτυξης κώδικα που προκαλούν προβλήματα σε χρήστες που χρησιμοποιούν screen readers, audio browsers και text-only browsers.
2. Μη δομημένες σελίδες που προκαλούν σύγχυση στους χρήστες και όρια στην πλοήγηση.
3. Τεχνικές που βασίζονται σε δομημένα στοιχεία για μορφοποίηση ή παρουσίαση προσθέτοντας μία σήμανση στις συστάσεις της HTML.
4. Υπερβολική χρήση οπτικών μέσων αναπαράστασης της πληροφορίας χωρίς να υπάρχουν text-based εναλλακτικές λύσεις.

Όλ' αυτά λήφθηκαν υπ' όψιν στην δημιουργία της HTML 4.0 και έτσι είχαμε πολλά νέα στοιχεία. Παρακάτω αναφέρονται μερικά απ' αυτά:

- Τα στοιχεία `abbr` και `acronym` χρησιμοποιούντε σε συνδυασμό με την ιδιότητα `lang` για να υποδηλώσουν μία αλλαγή στην φυσική γλώσσα ενός εγγράφου. Για άτομα που χρησιμοποιούν speech-based browsers, αυτές οι αλλαγές καθιστούν πιο εύκολο τον προσδιορισμό του νοήματος μίας συντομογραφίας ή ενός ακρωνύμου καθώς και τον προσδιορισμό αλλαγής γλώσσας στο κείμενο.
- Η `blockquote` μέθοδος για διαχωρισμό κειμένου που είναι σε παράθεση έχει ενισχυθεί με την εισαγωγή του νέου στοιχείου `q` για inline παραθέσεις.
- Τα στοιχεία `ins` και `del` είναι νέα στοιχεία που δηλώνουν εισαγωγή και διαγραφή από το κείμενο. Αυτές οι ιδιότητες είναι σημαντικές για να μπορούν να τονιστούν οι αλλαγές κειμένου σε επίσημα έγγραφα. Για παράδειγμα: `My favorite color is blue <ins>red</ins>!`
- Οι φόρμες περιλαμβάνουν λογική ομαδοποίηση των στοιχείων

Επίσης, στις οδηγίες προσβασιμότητας περιεχομένου web που εξέδωσε η WAI υπάρχουν τεχνικές που μπορούν να υλοποιηθούν μέσω της HTML.

Στη συνέχεια θ' αναφερθούμε σε μερικά παραδείγματα προσβασιμότητας μέσω της HTML για:

- Πίνακες
 1. Η ιδιότητα `summary` ενός πίνακα παρέχει μία μη οπτική περίληψη των περιεχομένων του πίνακα, πράγμα το οποίο μπορεί να είναι χρήσιμο σε άτομα που έχουν πρόσβαση στον πίνακα χρησιμοποιώντας βοηθητική τεχνολογία.
 2. Η ιδιότητα `scope` των `th`/`td` προσδιορίζει ποιιά κελιά του πίνακα σχετίζονται με κάποια συγκεκριμένη κεφαλίδα
 3. Η ιδιότητα `headers` των `th`/`td` προσδιορίζει ποιες κεφαλίδες είναι σχετικές με ένα δοθέν κελί του πίνακα, παρέχοντας το περιεχόμενο των δεδομένων για χρήστες `non-visual browsers`.
- Φόρμες
 3. Είναι καλό να γίνεται ομαδοποίηση των στοιχείων της φόρμας. Τα στοιχεία `fieldset` και `legend` ομαδοποιούν λογικά συσχετιζόμενα στοιχεία της φόρμας, ενώ μπορούμε να παρέχουμε ένα τίτλο μέσω του στοιχείου `legend`
 4. Το στοιχείο `label` συνδέει ένα στοιχείο της φόρμας με το αντίστοιχο περιγραφικό του κείμενο με ξεκάθαρο τρόπο. Είναι μία πολύ καλή βοήθεια για τυφλούς χρήστες που χρησιμοποιούν `screen readers` για να αλληλεπιδράσουν με φόρμες
- Εικόνες και πολυμέσα
 1. Η ιδιότητα `alt` παρέχει εναλλακτικό κείμενο για μία σημαντική εικόνα. Μπορεί να εφαρμοστεί στο `img` στοιχείο ή σε ένα `input` τύπου εικόνας (π.χ. `<input type="image" src="submit.jpg" alt="Submit your details"/>`)
 2. Αν η εικόνα είναι αρκετά πολύπλοκη και η περιγραφή της δεν χωράει στην ιδιότητα `alt` μπορούμε να χρησιμοποιήσουμε την ιδιότητα `longdesc` που παρέχει σύνδεση με επιπλέον πληροφορίες σχετικές με την εικόνα, που βρίσκονται σε ένα διαφορετικό αρχείο κειμένου. (π.χ. ``)

3. Αν έχουμε ένα βίντεο ή ένα ηχητικό κομμάτι πρέπει να παρέχουμε ένα βοηθητικό κείμενο με το ίδιο περιεχόμενο.
- Σύνδεσμοι και πλοήγηση
 1. Πρέπει να είμαστε βέβαιοι ότι οι σύνδεσμοι είναι ευδιάκριτοι ώστε οι χρήστες να βρίσκουν εύκολα που πρέπει να κάνουν κλικ και να μπορούν να τους εντοπίσουν με μια γρήγορη ματιά
 2. Πρέπει να χρησιμοποιούμε κείμενο ή εικόνες στους συνδέσμους που να περιγράφουν τι θα δούνε οι επισκέπτες αν κάνουν κλικ πάνω τους (δεν πρέπει να χρησιμοποιούμε φράσεις όπως «κάνε κλικ εδώ»). Όταν χρησιμοποιούμε εικόνες σ' έναν σύνδεσμο πρέπει να χρησιμοποιούμε εναλλακτικό κείμενο που να περιγράφει αυτό που θα δούνε οι χρήστες αν κάνουν κλικ πάνω του.
 3. Μπορούμε να χρησιμοποιούμε την ιδιότητα `title` σε ένα στοιχείο `<a>` για να παρέχουμε επιπλέον πληροφορίες στον επισκέπτη.
 4. Πρέπει να αποφεύγουμε να ανοίγουμε συνδέσμους σε νέα παράθυρα καθώς αυτό μπορεί να μπερδέψει τους χρήστες είτε επειδή δεν θα δουν το νέο παράθυρο που ανοίγει (τυφλοί χρήστες) είτε επειδή θα έχουν πάρα πολλά παράθυρα ανοιχτά.
 5. Πρέπει να αποφεύγουμε τους συνδέσμους που απαιτούν από τον χρήστη να έχει ακριβή έλεγχο του ποντικιού.
 - Δομή
 1. Πρέπει να χρησιμοποιούμε τα διάφορα επίπεδα κεφαλίδων (από `<h1>` έως `<h6>`) για να οργανώνουμε τις σελίδες μας και να τους δίνουμε δομή. Επίσης, οι κεφαλίδες προσφέρουν στους χρήστες που χρησιμοποιούν βοηθητική τεχνολογία όπως οι screen readers μία επιπλέον και γρήγορη μέθοδο για πλοήγηση μέσα στο έγγραφο μεταφέροντας τους από κεφαλίδα σε κεφαλίδα
 2. Πρέπει να προσπαθούμε να χρησιμοποιούμε σχετικές μονάδες μέτρησης του κειμένου και όχι απόλυτα μεγέθη ώστε να μπορούν εύκολα οι χρήστες να αυξομειώνουν το μέγεθος του κειμένου στον φυλλομετρητή τους.
 - Γενικού σκοπού

1. Είναι καλό να χρησιμοποιούμε μια DOCTYPE δήλωση για να προσδιορίζουμε ποια έκδοση της HTML ή της XHTML χρησιμοποιούμε.
2. Πρέπει να καθορίζουμε την γλώσσα του κειμένου στο `<body>` στοιχείο. Αν η γλώσσα αλλάζει στην μέση του κειμένου πρέπει να το καθορίζουμε μέσω των χαρακτηριστικών `lang` ή `xml:lang`.
3. Ένας καλογραμμένος τίτλος (`<title>`), παρόλο που δεν είναι ένα χαρακτηριστικό προσβασιμότητας, είναι το πρώτο πράγμα που θα διαβαστεί από ένα πρόγραμμα screen reader, γι' αυτό είναι μια καλή ευκαιρία για να εξηγήσουμε τι ακολουθεί.

2.1.2. Javascript

Η Javascript είναι η scripting γλώσσα του διαδικτύου και προσθέτει διαδραστικότητα στις HTML σελίδες. Μπορεί είτε να ενσωματωθεί σε μία ιστοσελίδα είτε σε ένα εξωτερικό script αρχείο και δίνει την δυνατότητα στους σχεδιαστές ιστοσελίδων να εκτελούν εργασίες όπως οι παρακάτω:

- Να διαβάζουν στοιχεία από έγγραφα και να γράφουν νέα στοιχεία και κείμενο σε έγγραφα.
- Να γράφουν και να διαβάζουν HTML στοιχεία. Μέσω της Javascript έχουν την δυνατότητα να διαβάζουν και να αλλάζουν τα περιεχόμενα ενός HTML στοιχείου.
- Να εκτελούν μαθηματικούς υπολογισμούς των δεδομένων.
- Να αντιδρούν σε γεγονότα. Η Javascript μπορεί να εκτελεστεί αν συμβεί ένα γεγονός, όπως για παράδειγμα όταν φορτωθεί μια σελίδα ή όταν ο χρήστης κάνει κλικ πάνω σε ένα HTML στοιχείο.
- Να ανακτούν την τρέχουσα ημερομηνία και ώρα από τον υπολογιστή του χρήστη ή την ημερομηνία και ώρα που ένα έγγραφο τροποποιήθηκε για τελευταία φορά.
- Να προσδιορίσουν το μέγεθος της οθόνης του χρήστη, την έκδοση του φυλλομετρητή ή την ανάλυση της οθόνης. Έτσι μπορούν για παράδειγμα να βρουν τί φυλλομετρητή χρησιμοποιεί ο χρήστης και, ανάλογα με τον

φυλλομετρητή, να φορτώσουν μια άλλη σελίδα που να είναι ειδικά σχεδιασμένη γι' αυτόν τον φυλλομετρητή.

- Να επικυρώνουν τα δεδομένα και να εκτελούν ενέργειες βασισμένες σε καταστάσεις. Η Javascript μπορεί να χρησιμοποιηθεί για να ειδοποιήσει τους χρήστες αν εισάγουν λάθος δεδομένα σε μια φόρμα ή για να επικυρώσει τα δεδομένα μιας φόρμας πριν να υποβληθεί στον server. Αυτό αποτρέπει επιπλέον επεξεργασία από την μεριά του server.
- Να δημιουργήσουν cookies. Η Javascript μπορεί να χρησιμοποιηθεί για να αποθηκεύσει και να ανακτήσει πληροφορίες στον υπολογιστή του χρήστη.

Παρ' όλες τις δυνατότητες που έχει η Javascript μπορεί να προκαλέσει κάποια προβλήματα στην προσβασιμότητα. Αυτά τα προβλήματα περιλαμβάνουν τα εξής:

- Πλοήγηση. Αδυναμία ή δυσκολία πλοήγησης χρησιμοποιώντας πληκτρολόγιο ή βοηθητικές τεχνολογίες.
- Κρυφό περιεχόμενο. Παρουσίαση περιεχομένου ή λειτουργικότητα που δεν είναι προσβάσιμη μέσω βοηθητικών τεχνολογιών.
- Έλεγχος χρήστη. Έλλειψη ελέγχου από τον χρήστη στις αυτοματοποιημένες αλλαγές περιεχομένου.
- Σύγχυση/Αποπροσανατολισμός. Αλλαγή ή απενεργοποίηση της συνηθισμένης λειτουργίας του φυλλομετρητή ή πυροδότηση κάποιου γεγονότος εν άγνοια του χρήστη.

Μια σελίδα που περιέχει Javascript είναι πλήρως προσβάσιμη αν η λειτουργικότητα του script είναι ανεξάρτητη από συσκευές και η πληροφορία (το περιεχόμενο) είναι προσβάσιμη από τις βοηθητικές τεχνολογίες. Δυστυχώς, δεν υπάρχει ένας ενιαίος τρόπος για να αντιμετωπιστούν όλα τα προβλήματα προσβασιμότητας της Javascript. Ο μόνος τρόπος για να διαβεβαιωθεί η προσβασιμότητα της Javascript είναι ελέγχοντας το κάθε script ξεχωριστά και αναπτύσσοντας μια μοναδική λύση στα προβλήματα προσβασιμότητας για καθένα απ' αυτά. Οι σχεδιαστές πρέπει να γνωρίζουν τα προβλήματα προσβασιμότητας της Javascript και να εφαρμόσουν τεχνικές οι οποίες παρέχουν τουλάχιστον ένα από τα παρακάτω:

1. Μέσω της Javascript να διαβεβαιώσουν ότι το script είναι προσβάσιμο

2. Να παρέχουν έναν εναλλακτικό τρόπο πρόσβασης που δεν περιέχει Javascript.

Η χρήση της Javascript καθ' αυτή δεν σημαίνει ότι η σελίδα δεν είναι προσβάσιμη. Αντίθετα, πολλές φορές, η χρήση της μπορεί να αυξήσει την προσβασιμότητα της σελίδας.

Μερικές φορές η Javascript χρησιμοποιείται για την δημιουργία οπτικών στοιχείων του interface τα οποία δεν επηρεάζουν την προσβασιμότητα. Συχνά η Javascript χρησιμοποιείται για image rollovers στα οποία μια εικόνα αντικαθιστάται με μια άλλη όταν ο δείκτης του ποντικιού βρεθεί πάνω της. Τέτοιες χρήσεις της Javascript δεν απαιτούν εναλλακτικούς τρόπους πρόσβασης μιας και δεν προσθέτουν επιπλέον λειτουργικότητα στη σελίδα.

Για να διασφαλίσουμε προσβασιμότητα μέσω της Javascript πρέπει να λάβουμε υπ' όψιν μας τα ακόλουθα ζητήματα:

- Πρέπει να χρησιμοποιούμε μόνο χειριστές γεγονότων (event handlers) που είναι ανεξάρτητοι από τις συσκευές (για παράδειγμα να μην απαιτείται μόνο η χρήση ποντικιού).
- Το περιεχόμενο και οι λειτουργίες που παρέχονται μέσω του script πρέπει να είναι προσβάσιμα από τις βοηθητικές τεχνολογίες.
- Οι σελίδες που χρησιμοποιούνε scripts πρέπει να είναι πλήρως προσπελάσιμες μέσω του πληκτρολογίου.
- Η Javascript δεν πρέπει να αλλάζει ή να υπερβαίνει τις προκαθορισμένες λειτουργίες του φυλλομετρητή με τρόπο που να προκαλεί σύγχυση.
- Όταν η Javascript δεν παρέχει πλήρη προσβασιμότητα, αυτό πρέπει να γίνεται με κάποιον άλλο τρόπο.

2.1.3. CSS

Τα αρχικά CSS σημαίνουν Cascading Style Sheet. Η CSS είναι η γλώσσα που χρησιμοποιείται για περιγραφή των web σελίδων, συμπεριλαμβάνοντας χρώματα, διάταξη και γραμματοσειρά. Ουσιαστικά η CSS καθορίζει πώς θα εμφανιστούν τα HTML στοιχεία, ενώ επιτρέπει την προσαρμογή της παρουσίασης σε

διαφορετικούς τύπους συσκευών. Ο διαχωρισμός της HTML και της CSS κάνει πιο εύκολη την συντήρηση των ιστοσελίδων, την απόκτηση ίδιου στυλ σε πολλές ιστοσελίδες καθώς και την χρήση πολλών διαφορετικών στυλ. Αυτό αναφέρεται σαν διαχωρισμός της δομής (ή του περιεχομένου) από την παρουσίαση.

Υπάρχουν τρεις τρόποι με τους οποίους μπορούμε να προσθέσουμε μορφοποίηση στις ιστοσελίδες:

- External style sheet
- Internal style sheet
- Inline style sheet

Στην συνέχεια θα αναλύσουμε καθέναν απ' αυτούς.

External style sheet: Είναι ο ιδανικός τρόπος όταν ένα στυλ εφαρμόζεται σε πολλές σελίδες. Μέσω του εξωτερικού style sheet μπορούμε να αλλάξουμε την εμφάνιση ενός ολόκληρου web site αλλάζοντας απλά ένα αρχείο. Κάθε σελίδα πρέπει να συνδέεται με το style sheet μέσω της σήμανσης <link>. Αυτός ο τρόπος παρέχει αρκετά προτερήματα:

- Οι σχεδιαστές ιστοσελίδων μπορούν εύκολα να χρησιμοποιήσουν το ίδιο style sheet σε οποιονδήποτε αριθμό HTML εγγράφων
- Οι σχεδιαστές ιστοσελίδων μπορούν εύκολα να αλλάξουν το style sheet χωρίς να χρειάζονται αλλαγές στο αρχείο
- Οι user agents μπορούν να φορτώσουν style sheets επιλεκτικά (βασισμένοι στις προδιαγραφές του συστήματος στο οποίο θα εμφανιστεί)

Ακολουθεί ένα παράδειγμα χρήσης εξωτερικού style sheet (example.css):

```
p {
  color: red;
  font-size: 120%;
}
```

Μαζί με τον τρόπο σύνδεσής του (example.html):

```
<head>
  <link rel="stylesheet" type="text/css" href="example.css">
</head>
```

Internal style sheet: Είναι ο τρόπος που εφαρμόζονται οι style sheet κανόνες στο head κομμάτι του HTML εγγράφου χρησιμοποιώντας την σήμανση <style>. Πρέπει να χρησιμοποιείται όταν ένα μόνο αρχείο έχει το δικό του μοναδικό στυλ. Ακολουθεί ένα παράδειγμα χρήσης αυτού του τύπου style sheet (example.html):

```
<head>
  <style>
    p { color: red; font-size: 120%; }
  </style>
</head>
<body>
  <p>This is a paragraph</p>
</body>
```

Inline style sheet: Η σήμανση αρχής μπορεί να περιλαμβάνει κανόνες style sheet απευθείας στο HTML έγγραφο μέσω της ιδιότητας style.

Ακολουθεί ένα παράδειγμα τέτοιου τύπου (example.html):

```
<p style="color: red; font-size: 120%; ">This is a
paragraph</p>
```

Αν κάποιες ιδιότητες έχουν οριστεί σε πολλά style sheets για τον ίδιο επιλογέα, οι τιμές θα κληρονομηθούν από το πιο ειδικό style sheet στο πιο γενικό.

Αν για παράδειγμα ένα εξωτερικό style sheet έχει τις εξής ιδιότητες για τον επιλογέα h3:

```
h3
{
color:red;
text-align:left;
font-size:8pt;
}
```

και ένα εσωτερικό style sheet έχει τις εξής ιδιότητες για τον επιλογέα h3:

```
h3
{
text-align:right;
```

```
font-size:20pt;  
}
```

τότε αν η σελίδα με το εσωτερικό style sheet συνδέεται με το εξωτερικό style sheet, ο επιλογέας h3 θα πάρει τις τιμές:

```
color:red;  
text-align:right;  
font-size:20pt;
```

όπου το χρώμα κληρονομείται από το εξωτερικό style sheet και το text-align και το font-size από το εσωτερικό style sheet.

Η ιεραρχία κληρονομικότητας είναι η εξής:

1. Inline style
2. Internal style sheet
3. External style sheet
4. Browser default

με το inline style να είναι το πιο ισχυρό.

Επίσης, μέσω της CSS μπορούμε να σχεδιάσουμε μια σελίδα με καλύτερη προσβασιμότητα. Στην συνέχεια θα αναλύσουμε μερικά χαρακτηριστικά της που επηρεάζουν την προσβασιμότητα.

Για να βεβαιωθούμε ότι οι χρήστες ελέγχουν τα styles, η CSS2 αλλάζει την σημασία του χειριστή `!important` που έχει οριστεί στην CSS1. Στη CSS1 οι σχεδιαστές αποφάσιζαν πάντα για το τελικό στυλ, αντίθετα στην CSS2 αν το style sheet ενός χρήστη περιέχει το `!important` παίρνει προτεραιότητα σε σχέση με οποιονδήποτε κανόνα style sheet του σχεδιαστή. Αυτό είναι μία πολύ βασική επιλογή για χρήστες που χρειάζονται ή πρέπει να αποφύγουν συγκεκριμένους συνδυασμούς χρωμάτων ή χρήστες που χρειάζονται μεγάλη γραμματοσειρά.

Οι ακουστικές ιδιότητες της CSS2 προσφέρουν πληροφορίες σε τυφλούς χρήστες και χρήστες voice browsers, όπως οι γραμματοσειρές παρέχουν οπτικές πληροφορίες. Για παράδειγμα μέσω της χρήσης του `voice-family`, το οποίο είναι κάτι σαν “ακουστική γραμματοσειρά”, ένας χρήστης μπορεί να καταλάβει ότι το περιεχόμενο που ακούει είναι παραδείγματος χάριν κεφαλίδα.

Η CSS2 επιτρέπει στους χρήστες να έχουν πρόσβαση στο περιεχόμενο ή στη πληροφορία ενός στοιχείου με εναλλακτικό τρόπο, χρησιμοποιώντας τα παρακάτω μαζί:

- επιλογέας ιδιοτήτων
- η μέθοδος `attr()` και η ιδιότητα `content`
- τα `before` και `after` ψευδοστοιχεία

Στο ακόλουθο παράδειγμα η τιμή του χαρακτηριστικού `alt` για το `IMG` στοιχείο φορτώνεται μετά την εικόνα (οπτικά, ηχητικά κ.τ.λ.)

```
IMG:after { content: attr(alt) }
```

Τα `media types` του CSS2 σε συνδυασμό με τους `@media` κανόνες δίνουν την δυνατότητα στους σχεδιαστές και χρήστες να δημιουργήσουν `style sheets` τα οποία επιτρέπουν στα έγγραφα να φορτωθούν πιο κατάλληλα σε συγκεκριμένες συσκευές. Αυτά τα `style sheets` προσαρμόζουν το περιεχόμενο για παρουσίαση σε συσκευές Braille, tty, οθόνες και χαρτί. Για παράδειγμα:

```
@media print {  
    /* style sheet for print goes here */  
}
```

2.2. Front end frameworks

Σ' αυτό το σημείο θα αναφερθούμε σε δυο front end frameworks, το jQuery και το Easy. Η χρήση αυτών των frameworks διευκολύνει τους σχεδιαστές ιστοσελίδων στην ανάπτυξη κώδικα και στη σχεδίαση. Αυτό επιτυγχάνεται μέσω κάποιων βιβλιοθηκών και προτύπων τα οποία περιλαμβάνονται σ' αυτά τα frameworks.

2.2.1. jQuery

Το jQuery είναι μία βιβλιοθήκη με Javascript functions. Είναι γρήγορη και συνοπτική με αποτέλεσμα να μας δίνει την δυνατότητα να εκτελούμε σύνθετες και

πολύπλοκες διαδικασίες χωρίς να απαιτεί πολύ κώδικα. Το jQuery παρέχει τις παρακάτω δυνατότητες:

- την επιλογή και τον χειρισμό των HTML στοιχείων
- τον χειρισμό της CSS
- τον χειρισμό των HTML γεγονότων
- την δημιουργία εφέ και animation της Javascript
- την διάσχυση και τροποποίηση του HTML DOM
- την χρήση AJAX

Η βιβλιοθήκη jQuery αποθηκεύεται σε ένα αρχείο Javascript και περιέχει όλες τις μεθόδους jQuery. Μπορούμε να την προσθέσουμε στην σελίδα μας με τον ακόλουθο τρόπο:

```
<head>  
<script type="text/javascript" src="jquery.js"></script>  
</head>
```

Η σύνταξη του jQuery έχει φτιαχτεί με τέτοιο τρόπο ώστε να επιτρέπει την επιλογή HTML στοιχείων και την τροποποίησή τους. Η βασική του σύνταξη είναι η εξής:

```
$(επιλογέας).ενέργεια(), όπου
```

το σύμβολο του δολαρίου προσδιορίζει το jQuery,
ο επιλογέας την εύρεση των HTML στοιχείων
και η ενέργεια() προσδιορίζει την ενέργεια που θα πραγματοποιηθεί στα στοιχεία.

Οι επιλογείς του jQuery μας επιτρέπουν επιλέξουμε HTML στοιχεία με βάση το όνομα του στοιχείου, μια ιδιότητά του ή μέσω του περιεχομένου του.

Επιλογή με βάση το όνομα του στοιχείου: Μέσω αυτού του τρόπου μπορούμε να επιλέξουμε στοιχεία που είναι του ίδιου τύπου, ανήκουν στην ίδια κλάση ή έχουν το ίδιο id. Ακολουθούν μερικά παραδείγματα:

```
$("p") που επιλέγει όλες τις παραγράφους (δηλαδή τα <p> στοιχεία)
```

```
$("p.intro") που επιλέγει όλες τις παραγράφους της κλάσης intro  
(class="intro")
```

`$("#demo")` που επιλέγει όλες τις παραγράφους με `id="demo"`.

Επιλογή με βάση μια ιδιότητα του στοιχείου: Σ' αυτή την περίπτωση το jQuery χρησιμοποιεί XPath εκφράσεις για να επιλέξει στοιχεία που έχουν την συγκεκριμένη ιδιότητα. Ακολουθούν μερικά τέτοια παραδείγματα:

`("[href]")` που επιλέγει όλα τα στοιχεία που έχουν την ιδιότητα `href`

`("[href='#']")` που επιλέγει όλα τα στοιχεία που έχουν την ιδιότητα `href` ίση με `"#"`

`("[href!='#']")` που επιλέγει όλα τα στοιχεία που η `href` ιδιότητά τους δεν είναι ίση με `"#"`

`("[href$='.jpg']")` που επιλέγει όλα τα στοιχεία που η `href` ιδιότητά τους τελειώνει σε `".jpg"`

Επιλογή με βάση το περιεχόμενο του στοιχείου: Μ' αυτόν τον τρόπο μπορούμε να επιλέξουμε όλα τα στοιχεία που έχουν ένα συγκεκριμένο περιεχόμενο. Για παράδειγμα:

`(":contains('Some text'))` που επιλέγει όλα τα στοιχεία που περιέχουν το κείμενο `'Some text'`

Μπορούμε να αναφερόμαστε στο κύριο jQuery αντικείμενο με το αναγνωριστικό όνομα `jQuery` ή με το σήμα του δολαρίου (`$`). Ας δούμε για παράδειγμα την ακόλουθη κλήση του jQuery αντικειμένου:

```
//Διάλεξε όλα τα στοιχεία της κλάσης: "someClass" και
//εφάρμοσε κόκκινο όριο σε ολ' αυτά
jQuery(".someClass").css("border", "1px solid red");
```

```
//Διάλεξε ένα αντικείμενο με id: someId και εισήγαγε δυναμικά
//HTML σ' αυτό
jQuery("#someId").html("<b>So bold!</b>");
```

```
//Ακριβώς το ίδιο με το από πάνω αλλά χρησιμοποιώντας $:
$("#someId").html("<b>So bold!</b>");
```

Αυτό το παράδειγμα μας δείχνει πόσο ισχυρή μπορεί να είναι μία δήλωση μέσω του jQuery. Δεν χρειάζεται να γράψει κανείς τις δικές του

`document.getElementById` λειτουργίες. Με μόνο μια γραμμή κώδικα το jQuery διαλέγει όλα τα στοιχεία που χρειάζονται σκανάροντας ολόκληρο το DOM και εφαρμόζει το επιθυμητό αποτέλεσμα. Επίσης είναι καλό να τονίσουμε ότι το jQuery φροντίζει και για cross-browser συμβατότητα.

Πολλοί Javascript προγραμματιστές θέλουν να εκτελείται κάποιος κώδικας στην `window.onload` συνάρτηση υπερφορτώνοντας το onload γεγονός της `<body>` σήμανσης. Αυτό σημαίνει ότι ο κώδικάς μας θα εκτελεστεί όταν η σελίδα φορτωθεί πλήρως. Δεν θέλουμε να εκτελέσουμε Javascript κώδικα σε μια σελίδα που ακόμα φορτώνεται καθώς αυτό θα μπορούσε να οδηγήσει σε λάθη. Για παράδειγμα να προσπαθήσει να κρύψει ένα στοιχείο που δεν υπάρχει ή να προσπαθήσει να βρει το μέγεθος μιας εικόνας που δεν έχει ακόμα φορτωθεί. Για να μας παρέχει αυτήν την λειτουργικότητα, το jQuery μας δίνει μια νέα μέθοδο που ονομάζεται `ready` και μπορεί να κληθεί από το βασικό αντικείμενο jQuery. Όταν γράφουμε ένα jQuery plugin χρησιμοποιούμε την συνάρτηση `ready` για να βεβαιωθούμε ότι είμαστε έτοιμοι να εκτελέσουμε το plugin μας. Ακολουθεί ένα τέτοιο παράδειγμα:

```
<script type="text/javascript">
$(document).ready(function()
{
//Το DOM έχει κατασκευαστεί
//Εδώ θα αρχικοποιήσουμε και θα τρέξουμε το plugin μας
});
</script>
```

Ας υποθέσουμε ότι έχουμε ένα plugin που ονομάζεται `Shuffle`. Τότε ο κώδικας θα έμοιαζε με τον ακόλουθο:

```
<script type="text/javascript">
//Ένας τρόπος για να αρχικοποιήσουμε τον κώδικα του plugin
$(document).ready(function()
{
jQuery.Shuffle.initialize( "monalisa.jpg", 5, 8, 67, 1500);
jQuery.Shuffle.run();
});
</script>
```


Το στυλ γραφής κώδικα εξαρτάται αποκλειστικά από τον προγραμματιστή αλλά εξαρτάται και από το τι θέλει να πετύχει. Είναι εύκολο να σκεφτούμε ότι ένα plugin πρέπει να γράφεται, να αρχικοποιείται και να εκτελείται με ορισμένο τρόπο κάθε φορά αλλά αυτό δεν είναι αληθές. Ο λόγος που βλέπουμε πολλά διαφορετικά στυλ και συντακτικές διαφορές στα jQuery plugin οφείλεται στο ότι οι προγραμματιστές προσπαθούν να πραγματοποιήσουν διαφορετικά πράγματα.

Τέλος, θα πούμε κάποια πράγματα για το jQuery UI (User Interface). Το jQuery UI περιλαμβάνει ένα περιεκτικό σύνολο από plugins που σχετίζονται με αλληλεπιδράσεις, widgets και οπτικά εφέ που χρησιμοποιούν αρχιτεκτονική καθοδηγούμενη από γεγονότα λαμβάνοντας υπ' όψιν τους τα web πρότυπα, την προσβασιμότητα και τον φιλικό προς τον χρήστη σχεδιασμό. Όλα τα plugins είναι συμβατά με IE 6.0+, Firefox 3+, Safari 3.1+, Opera 9.6+ και Google Chrome.

Ακολουθούν οι κατηγορίες με τα συστατικά που μας παρέχει το jQuery UI:

- Αλληλεπιδράσεις που περιλαμβάνουν σύνθετες συμπεριφορές όπως drag and drop, resizing, selection και sorting με τις αντίστοιχες συναρτήσεις (draggable, droppable, resizable, selectable, sortable).
- Widgets που αποτελούν μια λίστα με στοιχεία που καθένα απ' αυτά είναι πλήρως προσαρμόσιμο μέσω των ιδιοτήτων του. Τέτοια widgets είναι τα accordion, autocomplete, button, datepicker, dialog, progressbar, slider και tabs.
- Εφέ που χρησιμοποιούνται για τις animated μεταβιβάσεις και για την ευκολότερη δημιουργία αλληλεπιδράσεων. Περιλαμβάνει τις μεθόδους effect, show, hide, toggle, color animation, add class, remove class, toggle class, switch class.
- Utilities που είναι εργαλεία χαμηλού επιπέδου με σκοπό την ανάπτυξη αλληλεπιδράσεων, widgets και εφέ. Περιλαμβάνουν τα position και widget.

2.2.2. Easy

Το Easy είναι ένα CSS/HTML/Javascript framework. Η βασική ιδέα στην οποία στηρίζεται είναι η μείωση του χρόνου που χρειάζεται για τον σχεδιασμό του βασικού HTML προτύπου επαναχρησιμοποιώντας τις ίδιες τεχνικές κώδικα.

Ο κύριος στόχος αυτού του framework είναι να παρέχει μία σταθερή βάση για την δημιουργία front end εφαρμογών. Μπορεί να βοηθήσει στη γρήγορη δημιουργία ενός κύριου προτύπου, προσπερνώντας τα πρώτα βήματα, χωρίς να υπάρχει ανησυχία για ζητήματα που αφορούν την αναπαράσταση (rendering) στον φυλλομετρητή, συμπεριλαμβάνοντας πολυάριθμα plugins σε Javascript για την εκτέλεση απλών εργασιών.

Το Easy είναι κάτι παραπάνω από ένα CSS framework. Περιλαμβάνει ενσωματωμένες διαδραστικές λειτουργίες που είναι τόσο εύκολες στην χρήση που δεν χρειάζεται κανείς να ξέρει Javascript. Επίσης, περιλαμβάνει μια βιβλιοθήκη με έτοιμα κομμάτια CSS/HTML κώδικα τα οποία μπορεί κανείς απλά να τα επικολλήσει στο κύριό του πρότυπο.

Το πακέτο του Easy front end framework περιλαμβάνει:

- Ένα άδειο HTML έγγραφο που ονομάζεται index.html και περιλαμβάνει συνδέσεις με τα απαραίτητα αρχεία
- Ένα demo HTML έγγραφο που ονομάζεται demo.html με εφαρμοσμένα τα περισσότερα χαρακτηριστικά
- CSS αρχεία τόσο για οθόνη όσο και για εκτύπωση, με ονόματα: easy.css και easyprint.css
- Ένα αρχείο Javascript που ονομάζεται easy.js και περιλαμβάνει όλες τις ενσωματωμένες λειτουργίες που παρέχει το Easy
- Ένα αρχείο Javascript που ονομάζεται main.js που χρησιμοποιείται για να συμπεριλάβει κώδικα Javascript που έχει ήδη συσταθεί ή/και για προσθήκη κώδικα Javascript γραμμένο απ' τον προγραμματιστή
- Ένα jQuery αρχείο που ονομάζεται jquery.js
- Έναν άδειο φάκελο που ονομάζεται images και προορίζεται για αποθήκευση των εικόνων
- Έναν φάκελο που ονομάζεται library και περιλαμβάνει έτοιμα κομμάτια HTML κώδικα που μπορεί να χρησιμοποιήσει ο προγραμματιστής για το σχεδιασμό του κυρίως προτύπου. Επίσης περιλαμβάνει αρκετά έτοιμα πρότυπα με διαφορετική διάταξη.
- Ένα αρχείο κειμένου που περιλαμβάνει την επαγγελματική άδεια χρήσης και ονομάζεται license.txt

3. Javascript και HTML generator Frameworks

Στο κεφάλαιο αυτό θ' αναφερθούμε στα Javascript και HTML generator frameworks. Τα frameworks αυτά παράγουν κώδικα HTML και Javascript. Το Google Web Toolkit (GWT) είναι ένα από τα κυρίαρχα frameworks αυτής της κατηγορίας. Ο κώδικας γράφεται εξ' ολοκλήρου σε Java και υπάρχουν compilers που μετατρέπουν τον Java κώδικα σε HTML και Javascript. Θα αναφερθούμε κυρίως στο GWT αλλά και σε κάποια άλλα frameworks, όπως το Pyjamas και το Morfik BX, με τα οποία μπορούμε να γράψουμε κώδικα σε Python και Basic αντίστοιχα και στην συνέχεια να τον μεταγλωττίσουμε σε HTML και Javascript. Στο τέλος του κεφαλαίου θα δούμε και μια σύγκριση των τριών αυτών frameworks.

3.1. Google Web Toolkit

Όπως ήδη αναφέραμε στην εισαγωγή αυτής της πτυχιακής το GWT είναι ένα εργαλείο της Google για σχεδιασμό AJAX εφαρμογών. Είναι λογισμικό ανοιχτού κώδικα, είναι εντελώς δωρεάν και χρησιμοποιείται από χιλιάδες προγραμματιστές παγκοσμίως.

Μέσω του GWT ένας προγραμματιστής μπορεί να γράψει κώδικα σε Java και να τον μεταγλωττίσει σε Javascript που τρέχει σε όλους τους φυλλομετρητές. Ο κύριος στόχος του είναι να επιτρέψει την δημιουργία web εφαρμογών, χωρίς να χρειάζεται οι προγραμματιστές να γνωρίζουν το XMLHttpRequest, JavaScript και τις ιδιορρυθμίες του κάθε προγράμματος πλοήγησης.

Για να το κάνει αυτό μας παρέχει ένα σύνολο από Java APIs και Widgets. Ωστόσο δεν μας δημιουργεί περιορισμούς, με το GWT μπορούμε να κάνουμε ότι μπορούμε και με το DOM ενός φυλλομετρητή και την JavaScript.

Επίσης μπορούμε να χρησιμοποιήσουμε κάποιο IDE (Integrated Development Environment) της προτίμησής μας και να εκδώσουμε, να ανανεώσουμε και να δούμε την σελίδα μας όπως ακριβώς κάναμε με την Javascript, ενώ παράλληλα

μπορούμε να δούμε μεταβλητές, να βάλουμε breakpoints και να χρησιμοποιήσουμε όλα τα εργαλεία αποσφαλμάτωσης που μας παρέχει η Java. Το μόνο που χρειάζεται είναι να εγκαταστήσουμε το GWT SDK ή το Google Plugin για το Eclipse.

Στους δυο πίνακες που ακολουθούν φαίνονται οι σημαντικότεροι κατάλογοι και αρχεία που δημιουργούνται αυτόματα όταν δημιουργούμε μία καινούργια GWT εφαρμογή με όνομα MyProject, ενώ υπάρχει και μια μικρή περιγραφή για καθένα απ' αυτά.

Κατάλογοι	Περιγραφή
/src/com/google/gwt/sample/myproject	Περιέχει τον ορισμό του module του GWT και τα αρχικά αρχεία εφαρμογής
/test/com/google/gwt/sample/myproject	Περιέχει ένα κατάλογο δοκιμών του JUnit και μια αρχική κλάση ελέγχου
/war	Περιέχει στατικές πηγές όπως αρχεία εικόνων, style sheets και HTML host pages
/war/WEB-INF	Περιέχει τα αρχεία Java της web εφαρμογής
/war/WEB-INF/lib	Περιέχει τις Java βιβλιοθήκες της web εφαρμογής

Πίνακας 1 – Κατάλογοι του GWT

Αρχεία	Περιγραφή
MyProject.gwt.xml	Περιέχει τον ορισμό του module του GWT
MyProject.html	Περιέχει το host page
MyProject.css	Περιέχει το style sheet της εφαρμογής
web.xml	Είναι το αρχείο που περιγράφει την Java web εφαρμογή
MyProject.java	Είναι η entry point κλάση του GWT
GreetingService.java, GreetingServiceAsync.java, GreetingServiceImpl.java	Περιέχει τα δείγματα RPC (Remote Procedure Call) κλάσεων του GWT

gwt-servlet.jar	Είναι η runtime βιβλιοθήκη του διακομιστή του GWT
MyProjectTest.java	Είναι η αρχική περίπτωση δοκιμής του MyProject

Πίνακας 2 – Αρχεία του GWT

Για την καλύτερη κατανόηση των παρακάτω, στο σημείο αυτό θ' αναφέρουμε τί είναι το standards mode και τί το quirks mode. Στην επιστήμη των υπολογιστών το quirks mode αναφέρεται σε μια τεχνική που χρησιμοποιείται από κάποιους φυλλομετρητές έτσι ώστε να υπάρχει συμβατότητα με σελίδες που έχουν σχεδιαστεί για παλιότερες εκδόσεις φυλλομετρητών, αντί να ακολουθούν πιστά τις οδηγίες W3C και IETF που χρησιμοποιούνται όταν σχεδιάζουμε σε standards mode.

Για να προσφέρει καλύτερη συμβατότητα ανάμεσα στους διάφορους φυλλομετρητές, το GWT θέτει την doctype δήλωση σε HTML 4.01 Transitional. Αυτό με την σειρά του ορίζει την μηχανή ερμηνεύσης του φυλλομετρητή σε quirks mode. Αν παρόλ' αυτά θέλει κάποιος να ερμηνεύσει την εφαρμογή του σε standards mode, υπάρχει ένα σύνολο από άλλα doctypes που μπορούν να χρησιμοποιηθούν ώστε να αναγκάσουν την αναπαράσταση του φυλλομετρητή να γίνει με αυτόν τον τρόπο. Γενικά, οι εφαρμογές του GWT δουλεύουν εξίσου καλά σε standards mode όπως και σε quirks mode, αλλά σε μερικές περιπτώσεις η χρήση κάποιων widgets και panels μπορεί να μην ερμηνευτεί σωστά. Αυτό το πρόβλημα έχει βελτιωθεί αρκετά από την έκδοση 1.5 του GWT και η δουλειά συνεχίζεται ώστε να βρεθεί λύση στο πρόβλημα αυτό μια για πάντα.

Για να φτιάξουμε μια web εφαρμογή το GWT μας παρέχει ένα σύνολο από widgets και panels. Στην συνέχεια ακολουθεί μία λίστα με τα διαθέσιμα widgets:

- Button που είναι ένα κλασσικό κουμπί.
- PushButton που είναι ένα κλασσικό κουμπί αλλά μας παρέχει την δυνατότητα να προσαρμόσουμε το στυλ του.
- RadioButton που είναι ένα κλασσικό radio button.
- CheckBox που είναι ένα κλασσικό check box.
- DatePicker που είναι ένα κλασσικό date picker.

- `ToggleButton` που είναι ένα κουμπί που μας επιτρέπει να αλλάξουμε το στυλ του ενώ παράλληλα επιτρέπει στον χρήστη να το εναλλάσσει μεταξύ των `up` και `down` καταστάσεων.
- `TextBox` που είναι ένα απλό `text box` μίας γραμμής.
- `PasswordTextBox` που είναι ένα `text box` που καλύπτει την είσοδό του ώστε να εμποδίζονται οι υποκλοπές.
- `TextArea` που είναι ένα `text box` που επιτρέπει την είσοδο περισσότερων της μιας γραμμής.
- `Hyperlink` που είναι ένας υπερσύνδεσμος.
- `ListBox` που εμφανίζει στον χρήστη μία λίστα από επιλογές είτε με την μορφή `list box` είτε με την μορφή `drop-down list`.
- `CellList` που είναι μία λίστα κελιών μίας στήλης.
- `MenuBar` που είναι ένα `menu bar` που μπορεί να περιέχει οποιονδήποτε αριθμό στοιχείων `menu` και καθένα από αυτά μπορεί είτε να πυροδοτήσει ένα `command` είτε να ανοίξει ένα υπο-μενού. Το `command` περιλαμβάνει μια ενέργεια για μεταγενέστερη εκτέλεση.
- `Tree` που είναι ένα ιεραρχικό δέντρο που περιέχει `Treeltems`. Καθένα απ' αυτά μπορεί ν' ανοίξει, να κλείσει και να επιλεγεί απ' τον χρήστη.
- `CellTree` που είναι ένας τρόπος εμφάνισης ενός δέντρου. Αυτό το `widget` δουλεύει μόνο αν η HTML σελίδα έχει μια σαφή `<!DOCTYPE>` δήλωση.
- `SuggestBox` που είναι ένα `text box` ή ένα `text area` που εμφανίζει στο χρήστη ένα σύνολο από επιλογές που έχουν οριστεί από πριν.
- `RichTextArea` που είναι ένας `text editor` που επιτρέπει σύνθετο στυλ και μορφή.
- `FlexTable` που είναι ένας ευέλικτος πίνακας που δημιουργεί κελιά κατά ζήτηση. Τα κελιά αυτού του πίνακα μπορούν να διασχίζουν πολλαπλές γραμμές και στήλες.
- `Grid` που είναι ένα ορθογώνιο πλέγμα που τα κελιά του μπορεί να περιλαμβάνουν κείμενο, `html` ή άλλα `widgets`.
- `CellTable` που είναι μια πινακοειδής όψη που υποστηρίζει γραμμές και στήλες.
- `CellBrowser` που είναι μια δένδροειδής όψη που επιτρέπει την περιήγηση και μπορεί κάθε φορά να είναι ανοιχτός μόνο ένας κόμβος ανά επίπεδο.

Αυτό το widget δουλεύει μόνο αν η HTML σελίδα έχει μια σαφή <!DOCTYPE> δήλωση.

- TabBar που είναι μια οριζόντια μπάρα από καρτέλες που συχνά χρησιμοποιείται σαν μέρος ενός TabPanel.
- DialogBox που είναι μια pop up φόρμα που έχει επικεφαλίδα στο πάνω μέρος και μπορεί να μετακινηθεί από τον χρήστη.

Παρόλ' αυτά αν δεν βρίσκουμε κάποιο widget που να καλύπτει τις απαιτήσεις μας μπορούμε να δημιουργήσουμε ένα δικό μας. Το νέο widget μπορεί να προκύψει από μία σύνθεση υπάρχοντων widgets ή μπορούμε να το δημιουργήσουμε απ' την αρχή χρησιμοποιώντας Java ή Javascript.

Στην λίστα που μόλις αναφέραμε τα widgets έχουν το προεπιλεγμένο στυλ, όμως μπορούμε να αλλάξουμε την εμφάνισή τους μέσω της CSS.

Το GWT μας παρέχει ένα σύνολο από panels για να μπορέσουμε να καθορίσουμε την διάταξη των widgets στην σελίδα. Ένα panel μπορεί να έχει μέσα του άλλα panels. Αυτό είναι ανάλογο με την διάταξη μιας HTML ιστοσελίδας χρησιμοποιώντας div στοιχεία ή πίνακες τον έναν μέσα στον άλλο.

Στην συνέχεια ακολουθεί μία λίστα με όλα τα διαθέσιμα panels:

- PopUpPanel που είναι ένα panel που εμφανίζεται πάνω από τα άλλα widgets. Επικαλύπτει την περιοχή χρήστη του φυλλομετρητή και οποιοδήποτε άλλο popUp δημιουργήθηκε προγενέστερα.
- StackPanel που είναι ένα panel που στοιβάζει τα παιδιά του κάθετα, εμφανίζοντας μόνο ένα κάθε φορά. Σ' αυτό το panel κάθε παιδί έχει κεφαλίδα και ο χρήστης μπορεί να κάνει κλικ πάνω σ' αυτήν για να εμφανιστεί. Αυτό το panel δουλεύει μόνο σε quirks mode. Αν η εφαρμογή μας είναι σε standards mode πρέπει να χρησιμοποιήσουμε το StackLayoutPanel.
- StackLayoutPanel που είναι ένα panel που επίσης στοιβάζει τα παιδιά του κάθετα εμφανίζοντας μόνο ένα κάθε φορά. Όπως και στο προηγούμενο panel έτσι και σ' αυτό κάθε παιδί έχει μία κεφαλίδα στην οποία μπορεί να κάνει κλικ ο χρήστης για να εμφανιστεί. Η διαφορά του με το προηγούμενο

είναι ότι δουλεύει μόνο σε standards mode, που σημαίνει ότι η HTML σελίδα στην οποία τρέχει πρέπει να έχει μια σαφή <!DOCTYPE> δήλωση.

- HorizontalPanel που είναι ένα panel που τοποθετεί όλα του τα widgets σε μία μονή οριζόντια στήλη.
- VerticalPanel που είναι ένα panel που τοποθετεί όλα του τα widgets σε μια μονή κάθετη στήλη.
- FlowPanel που είναι ένα panel που τοποθετεί τα widgets του χρησιμοποιώντας την προεπιλεγμένη HTML διάταξη.
- VerticalSplitPanel που είναι ένα panel που έχει δύο widgets σε μια μονή κάθετη στήλη και επιτρέπει στον χρήστη να αλλάζει το ύψους του κάθε widget. Στα widgets που βρίσκονται μέσα στο VerticalSplitPanel θα προστεθούν αυτόματα μπάρες κύλισης όταν χρειαστεί. Αυτό το panel δουλεύει μόνο σε quirks mode. Αν η εφαρμογή είναι σε standards mode πρέπει να χρησιμοποιήσουμε το SplitLayoutPanel.
- HorizontalSplitPanel που είναι ένα panel που έχει δυο widgets σε μια μονή οριζόντια στήλη και επιτρέπει στον χρήστη να αλλάζει το πλάτους του κάθε widget. Στα widgets που βρίσκονται μέσα σ' αυτό το panel θα προστεθούν αυτόματα μπάρες κύλισης αν χρειαστεί. Και αυτό το panel δουλεύει μόνο σε quirks mode και αν η εφαρμογή μας είναι σε standards mode πρέπει να χρησιμοποιήσουμε το SplitLayoutPanel.
- SplitLayoutPanel που είναι ένα panel που προσθέτει διαχωριστές που μπορούν να καθοριστούν απ' τον χρήστη μεταξύ όλων των widget-παιδιών.
- DockPanel που είναι ένα panel που αφήνει τα widget-παιδιά του να είναι συνδεδεμένα στις εξωτερικές πλευρές και αφήνει το τελευταίο widget να καταλάβει τον χώρο που βρίσκεται στο κέντρο. Αυτό το widget έχει περιορισμούς στο standards mode που δεν υπάρχουν στο quirks mode. Στο widget-παιδί δεν μπορούμε να χρησιμοποιήσουμε ποσοστά για να αλλάξουμε το μέγεθός του. Για παράδειγμα, αν ορίσουμε το ύψος του widget-παιδιού σε 100% αυτό δεν θα οδηγήσει σε κάλυψη όλου του δυνατού ύψους στο παιδί.
- DockLayoutPanel που είναι ένα panel που αφήνει τα widget-παιδιά του να είναι συνδεδεμένα στις εξωτερικές πλευρές και αφήνει το τελευταίο widget να καταλάβει τον χώρο που βρίσκεται στο κέντρο. Αυτό το widget δουλεύει

μόνο σε standards mode, το οποίο απαιτεί στην HTML σελίδα να υπάρχει μία σαφής <!DOCTYPE> δήλωση.

- TabPanel που είναι ένα panel που αναπαριστά ένα σύνολο από σελίδες χωρισμένες σε καρτέλες και καθεμία απ' αυτές περιέχει ένα άλλο widget. Τα widgets παιδιά φαίνονται καθώς ο χρήστης επιλέγει μια απ' τις καρτέλες που σχετίζονται μ' αυτά. Και αυτό το widget δουλεύει μόνο σε quirks mode και αν η εφαρμογή μας είναι σε standards mode πρέπει να χρησιμοποιήσουμε το TabLayoutPanel.
- TabLayoutPanel που είναι ένα panel που αναπαριστά ένα σύνολο από σελίδες χωρισμένες σε καρτέλες και καθεμία απ' αυτές περιέχει ένα άλλο widget. Τα widget-παιδιά φαίνονται καθώς ο χρήστης επιλέγει μια απ' τις καρτέλες που σχετίζονται μ' αυτά. Οι καρτέλες μπορεί να περιέχουν κείμενο, HTML ή widgets. Αυτό το widget δουλεύει μόνο σε standards mode, το οποίο προαπαιτεί η HTML σελίδα στην οποία τρέχει να έχει μία σαφή <!DOCTYPE> δήλωση.
- DisclosurePanel που είναι ένα widget που αποτελείται από μια κεφαλίδα και ένα panel που περιλαμβάνει το περιεχόμενο, και αποκαλύπτει το περιεχόμενο όταν ο χρήστης κάνει κλικ στην κεφαλίδα.

Υπάρχει ένα ακόμα panel που δεν είναι ορατό στην διεπιφάνεια του χρήστη, αυτό είναι το Root panel. Το Root panel είναι αυτό που περιέχει όλα τα δυναμικά στοιχεία της εφαρμογής. Είναι στην κορυφή οποιασδήποτε ιεραρχίας διεπιφάνειας χρήστη στο GWT. Υπάρχουν δύο τρόποι με τους οποίους μπορούμε να χρησιμοποιήσουμε ένα Root panel: είτε δημιουργώντας όλο το σώμα της σελίδας, είτε δημιουργώντας ειδικά στοιχεία που ενσωματώνονται στο σώμα.

Το Root panel δουλεύει αντιστοιχίζοντας το <body> ή κάποιο άλλο στοιχείο στην HTML σελίδα. Σαν προεπιλογή (η οποία ισχύει όταν δεν έχουμε κρατήσει κάποια θέση στην host σελίδα) το Root panel αντιστοιχίζεται με το <body> στοιχείο, όπως φαίνεται στο παράδειγμα παρακάτω:

```
RootPanel.get() //Προεπιλογή. Αντιστοιχίζει το HTML body στοιχείο
```

Έτσι το περιεχόμενο του RootPanel θα αντικαταστήσει το κενό <body> της host σελίδας.

Παρόλ' αυτά μπορούμε ν' αντιστοιχίσουμε οποιοδήποτε στοιχείο αν του δώσουμε ένα id και καλώντας το Root panel περάσουμε το id αυτό σαν παράμετρο, όπως φαίνεται στο παράδειγμα που ακολουθεί:

```
RootPanel.get("someElement") //Αντιστοιχίζει το HTML στοιχείο με id  
//"someElement"
```

Σ' αυτή τη περίπτωση θα βρεθεί το στοιχείο με αυτό το id και θα αντικατασταθεί με το περιεχόμενο του RootPanel.

Μία host σελίδα μπορεί να περιλαμβάνει παραπάνω του ενός Root panels. Για παράδειγμα, αν θέλουμε να ενσωματώσουμε πολλαπλά GWT widgets και panels σε μία host σελίδα, καθένα απ' αυτά πρέπει να υλοποιηθεί ανεξάρτητα από τα υπόλοιπα, αντιστοιχίζοντας το δικό του Root panel.

Ο κώδικας Java σε κάθε GWT εφαρμογή βρίσκεται στο αρχείο: MyProject/src/com/google/gwt/sample/myproject/client/MyProject.java. Αυτό το αρχείο περιέχει την MyProject κλάση. Αυτή η κλάση υλοποιεί την διεπαφή EntryPoint και περιλαμβάνει την μέθοδο onModuleLoad. Επειδή η κλάση MyProject έχει οριστεί σαν entry point κλάση, όταν εκτελείται το MyProject η onModuleLoad μέθοδος θα κληθεί.

Το GWT είναι event-based. Αυτό σημαίνει ότι ο κώδικας εκτελείται επειδή κάποιο γεγονός προκλήθηκε. Συνήθως αυτό το γεγονός πυροδοτείται από κάποια ενέργεια του χρήστη καθώς χρησιμοποιεί το ποντίκι ή το πληκτρολόγιο για να αλληλεπιδράσει με την διεπιφάνεια της εφαρμογής.

Για να γράψουμε ένα γεγονός, περνάμε ένα συγκεκριμένο event handler interface στο κατάλληλο widget. Ένα event handler interface ορίζει μια ή περισσότερες μεθόδους της οποίες το widget με την σειρά του καλεί για να δημιουργήσει το γεγονός.

Αν έχουμε μια μεγάλη εφαρμογή με πολλά γεγονότα είναι καλύτερα να έχουμε μια κλάση που να υλοποιεί το event handler interface και να διαχειρίζεται γεγονότα που έρχονται από πολλαπλούς εκδότες γεγονότων.

Μέσα στο αρχείο MyProject.css, περιέχονται όλοι οι style sheet κανόνες για μια εφαρμογή. Όπως συμβαίνει με κάθε web σελίδα, έτσι και εδώ, μπορούμε να καθορίσουμε πολλαπλά style sheets. Μπορούμε να έχουμε μια λίστα με style

sheets σε σειρά κληρονομικότητας, δηλαδή με τον πιο εξειδικευμένο κανόνα στυλ καταγεγραμμένο στο τελευταίο style sheet.

Επίσης, μπορούμε να έχουμε πρωτεύοντα και δευτερεύοντα στυλ. Στο GWT, κάθε κλάση widget έχει ένα σχετικό όνομα στυλ (όπως είναι για παράδειγμα το gwt-Button) που το υποχρεώνει να ακολουθήσει ένα CSS κανόνα. Αυτό είναι το πρωτεύον στυλ του widget. Το δευτερεύον στυλ εφαρμόζεται σε ένα μόνο widget. Αυτό γίνεται καθορίζοντας πρώτα τον κανόνα στυλ στο MyProject.css και καθορίζοντας στην συνέχεια τα χαρακτηριστικά της HTML κλάσης προγραμματιστικά μέσα στο MyProject.java αρχείο. Ακολουθεί ένα τέτοιο παράδειγμα για καλύτερη κατανόηση.

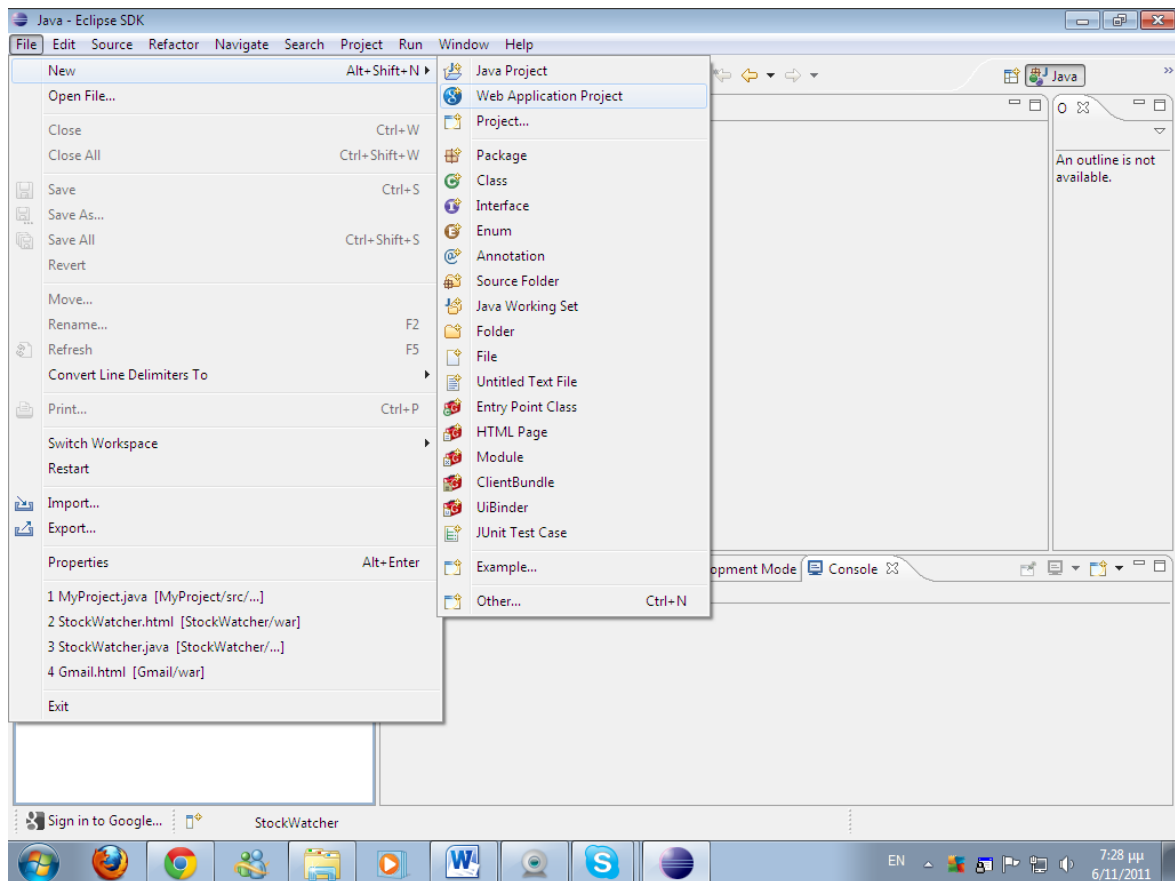
Στο MyProject.css ορίζουμε το στυλ:

```
.addPanel {  
    margin: 10px 0px 15px 0px;  
}
```

Στο MyProject.java κάνουμε την συσχέτιση:

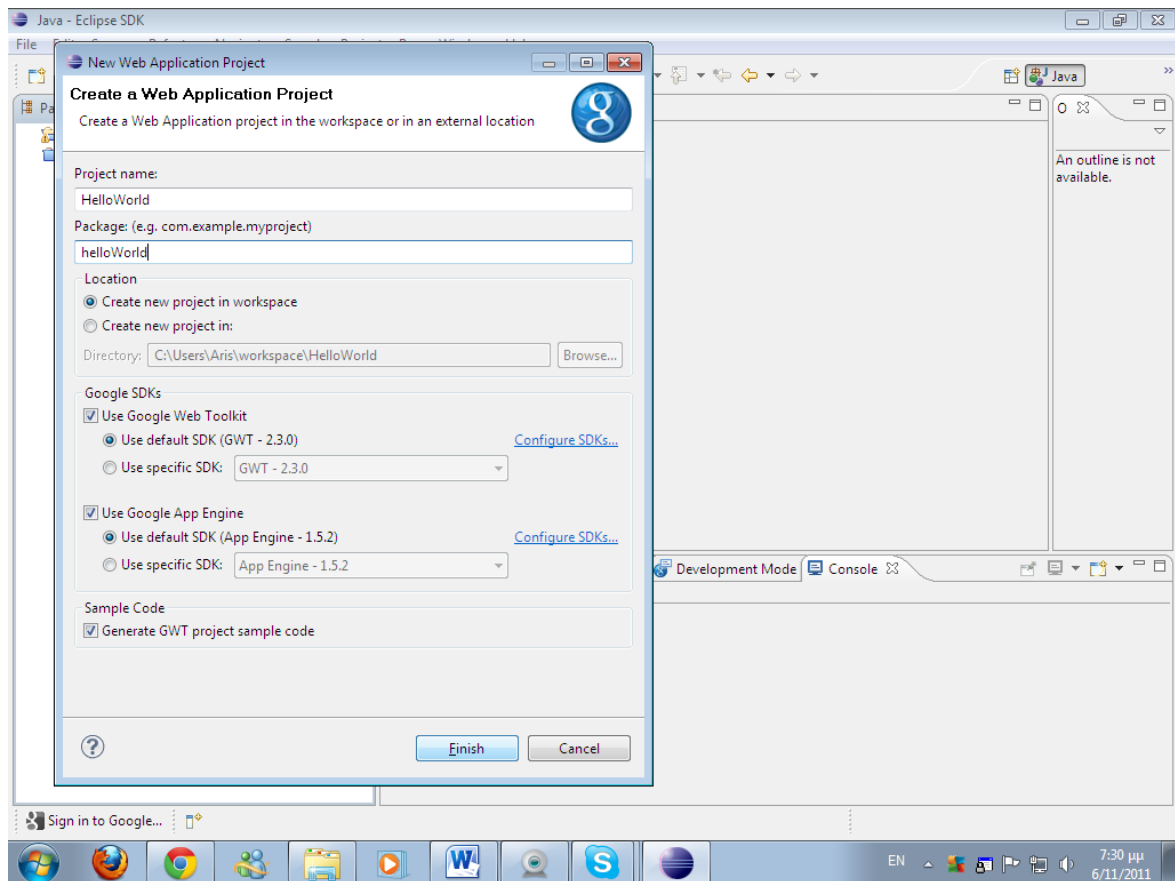
```
somePanel.addStyleName("addPanel");
```

Τέλος θα δείξουμε πώς μπορούμε να κάνουμε μια μικρή εφαρμογή, το Hello World από την αρχή μέσω του GWT. Αρχικά δημιουργούμε ένα νέο Web Application Project, ακολουθώντας τα: File – New – Web Application Project, όπως φαίνεται στην εικόνα που ακολουθεί.



Εικόνα 1 – New Web Application Project

Στη συνέχεια ονομάζουμε την εφαρμογή HelloWorld και το πακέτο helloWorld, ενώ παράλληλα κάνουμε κλικ στο Create new project in workspace, στο Use default SDK (GWT - 2.3.0), στο Use default SDK (App Engine - 1.5.2.) και στο Generate GWT project sample code, όπως φαίνεται στην Εικόνα 2.



Εικόνα 2 – Create a Web Application Project

Στη συνέχεια ανοίγουμε το αρχείο HelloWorld.java, σβήνουμε οτι υπάρχει μέσα και γράφουμε τα εξής:

```
package helloWorld.client;
import com.google.gwt.core.client.EntryPoint;
import com.google.gwt.user.client.Window;
import com.google.gwt.user.client.ui.Button;
import com.google.gwt.user.client.ui.ClickListener;
import com.google.gwt.user.client.ui.RootPanel;
import com.google.gwt.user.client.ui.Widget;
public class HelloWorld implements EntryPoint {
    public void onModuleLoad() {
        Button b = new Button("Click me", new ClickListener() {
            public void onClick(Widget sender) {
                Window.alert("Hello World");
            }
        });
        RootPanel.get("myData").add(b);
    }
}
```

Ενώ στο αρχείο HelloWorld.html γράφουμε:

```
<html>
  <head>
    <title>Hello World Project</title>
  </head>
  <body>
    <div id="myData"></div>
  </body>
</html>
```

3.1.1. GWT και προσβασιμότητα

Ένα από τα βασικά χαρακτηριστικά του GWT είναι ότι μας παρέχει κάποιες τεχνικές που σχετίζονται με την προσβασιμότητα και έτσι μας επιτρέπει να σχεδιάζουμε ευκολότερα προσβάσιμες σε όλους σελίδες. Αυτό το επιτυγχάνει κυρίως μέσω του Accessible Rich Internet Applications (ARIA).

Το ARIA είναι μια προδιαγραφή της W3C που καθορίζει τον τρόπο με τον οποίο μπορούμε να κάνουμε το web περιεχόμενο και τις διαδικτυακές εφαρμογές πιο εύκολα προσβάσιμες για τα άτομα με αναπηρία. Σχετίζεται κυρίως με δυναμικό περιεχόμενο και προηγμένα controls που αναπτύσσονται με Ajax, HTML και Javascript.

Αυτή τη στιγμή υπάρχουν λειτουργίες σε διάφορες ιστοσελίδες που δεν είναι διαθέσιμες για κάποιους χρήστες με αναπηρία και κυρίως για άτομα που χρησιμοποιούν screen readers και για άτομα που δεν μπορούν να χρησιμοποιήσουν ποντίκι. Το ARIA σχετίζεται με αυτές τις προκλήσεις προσδιορίζοντας, για παράδειγμα, νέους τρόπους με τους οποίους μπορούμε να προσθέσουμε λειτουργικότητα στις βοηθητικές τεχνολογίες.

Οι σύγχρονες ιστοσελίδες χρησιμοποιούνται όλο και περισσότερο προηγμένα και σύνθετα controls, όπως για παράδειγμα δένδρικά controls για την πλοήγηση μέσα στην σελίδα. Για να είναι αυτά προσβάσιμα στα άτομα με αναπηρία, θα πρέπει να μπορούν οι βοηθητικές τεχνολογίες να αλληλεπιδρούν με αυτά τα controls. Παρόλ' αυτά, οι πληροφορίες που χρειάζονται οι βοηθητικές τεχνολογίες δεν είναι παρέχονται από τις περισσότερες σύγχρονες τεχνολογίες.

Ένα ακόμα τέτοιο παράδειγμα είναι η λειτουργία drag-and-drop που δεν είναι διαθέσιμη για χρήστες που χρησιμοποιούν μόνο πληκτρολόγιο επειδή δεν μπορούν να χρησιμοποιήσουν ποντίκι. Ακόμα και σχετικά απλές σελίδες μπορεί να γίνουν δύσκολες στην πλοήγηση αν απαιτούν ένα μεγάλο πλήθος πληκτρολογήσεων.

Επίσης, όταν οι ενέργειες του χρήστη προκαλούν αλλαγή στο περιεχόμενο της σελίδας, το νέο περιεχόμενο δεν είναι διαθέσιμο σε μερικούς ανθρώπους, όπως για παράδειγμα τυφλούς χρήστες ή άτομα με μαθησιακές δυσκολίες που χρησιμοποιούν screen readers.

Το ARIA απευθύνεται σ' αυτές τις προκλήσεις προσβασιμότητας προσδιορίζοντας πώς μπορούν να δοθούν οι κατάλληλες πληροφορίες στις βοηθητικές τεχνολογίες ώστε να παρέχουν αυτή τη λειτουργικότητα. Μέσω του ARIA μπορούμε να κάνουμε τις web εφαρμογές προσβάσιμες στα άτομα με αναπηρία.

Για να προσθέσουμε προσβασιμότητα στα widgets του GWT θα πρέπει να προσθέσουμε τις αντίστοιχες ιδιότητες στα DOM στοιχεία. Οι ιδιότητες αυτές των στοιχείων χρησιμοποιούνται από τον φυλλομετρητή ώστε να μπορεί συλλάβει το γεγονός κατά την αλληλεπίδραση του χρήστη. Έτσι τα screen readers με την σειρά τους μπορούν να αντιδράσουν στα γεγονότα αυτά και να “ανακοινώσουν” έτσι την λειτουργία του GWT widget. Οι DOM ιδιότητες που καθορίζονται από το ARIA χωρίζονται σε roles (ρόλους) και states (καταστάσεις).

Η ιδιότητα role του ARIA δείχνει τι τύπος είναι το widget, περιγράφει δηλαδή τον τρόπο με τον οποίο συμπεριφέρεται το widget. Το role είναι στατικό που σημαίνει ότι δεν αλλάζει κατά την διάρκεια ζωής του widget. Μερικά παραδείγματα ARIA roles είναι τα εξής: tree, menubar, menuitem, tab. Το ARIA role ενός widget εκφράζεται με το DOM χαρακτηριστικό role, η τιμή του οποίου θα είναι ένα από τα ARIA role strings.

Υπάρχουν επίσης ARIA ιδιότητες που περιγράφουν την κατάσταση ενός widget. Για παράδειγμα ένα checkbox widget θα μπορούσε να είναι στην κατάσταση “checked” ή “unchecked”. Μια κατάσταση είναι δυναμική και θα πρέπει να ενημερώνεται κατά την αλληλεπίδραση με τον χρήστη. Μερικά παραδείγματα από

ARIA καταστάσεις είναι τα εξής: aria-disabled, aria-pressed, aria-expanded, aria-haspopup.

Το role ενός widget καθορίζει ένα σύνολο από δυνατά states τα οποία υποστηρίζει. Ένα widget με τον ρόλο μιας λίστας, για παράδειγμα, δεν θα είχε state aria-pressed.

Για να είναι ένα widget προσβάσιμο, θα πρέπει να υποστηρίζει την χρήση πληκτρολογίου. Τα screen readers “ανακοινώνουν” το στοιχείο που έχει πάρει focus από το πληκτρολόγιο. Έτσι η πρόσβαση μπορεί να επιτευχθεί μετακινώντας το focus στα διάφορα στοιχεία της σελίδας μέσω του πληκτρολογίου.

Όταν τα ARIA roles έχουν προστεθεί στο DOM ενός GWT widget, οι φυλλομετρητές θα “ανακοινώσουν” τα κατάλληλα event στον screen reader. Το ARIA βρίσκεται ακόμα υπό ανάπτυξη οπότε οι φυλλομετρητές μπορεί να μην ανακοινώσουν όλα τα events για κάθε ARIA ιδιότητα, με αποτέλεσμα οι screen readers να μην αναγνωρίσουν όλα τα events που “ανακοινώνονται”.

Πολλά GWT widgets είναι προσβάσιμα από το πληκτρολόγιο και διαθέτουν ARIA ιδιότητες. Σε αυτά συμπεριλαμβάνονται τα CustomButton, Tree, TreeItem, MenuBar, MenuItem, TabBar και TabPanel. Επίσης όλα τα widgets που κληρονομούν το FocusWidget έχουν πλέον το tabIndex εξορισμού, διευκολύνοντας έτσι την πλοήγηση μέσω πληκτρολογίου.

Στην συνέχεια του κεφαλαίου θ’ ασχοληθούμε με ακόμα δύο frameworks, το Pyjamas και το Morfik BX. Θα εξηγήσουμε πώς δουλεύουν και θα τα συγκρίνουμε μεταξύ τους.

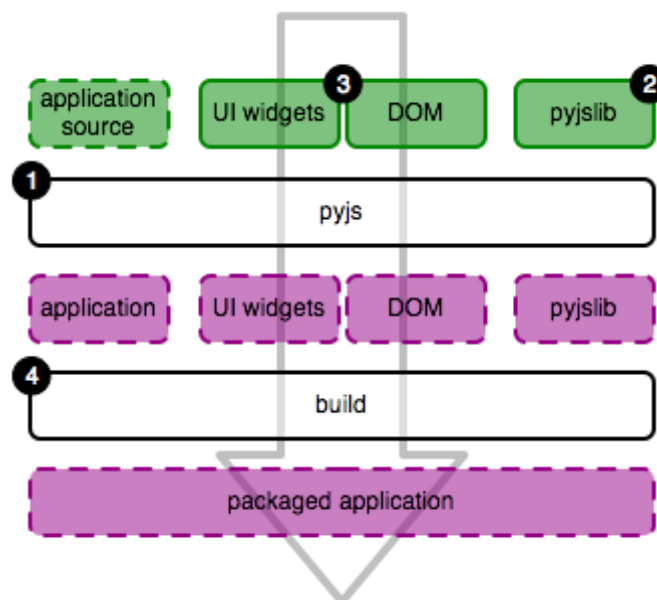
3.2. Pyjamas

Το Pyjamas είναι μία πλατφόρμα ανάπτυξης που χρησιμοποιείται τόσο για web όσο και για desktop εφαρμογές. Περιλαμβάνει έναν μεταγλωττιστή από Python σε Javascript, ένα AJAX framework και ένα σύνολο από widgets. Το Pyjamas φτιάχτηκε σαν μία Python εκδοχή του GWT.

Το βασικότερο πλεονέκτημα και ταυτόχρονα κυριότερος στόχος του Pyjamas, όπως και του GWT, είναι η δυνατότητα γραφής web εφαρμογών από άτομα που δεν γνωρίζουν HTML και Javascript. Μέσω του Pyjamas μπορούμε να ενθυλακώσουμε έννοιες υψηλού επιπέδου μέσα σε κλάσεις ενώ μπορούμε παράλληλα να επαναχρησιμοποιήσουμε και να εισάγουμε έτοιμα κομμάτια κώδικα και κλάσεις.

Το Pyjamas μας παρέχει ένα API σύνολο από widgets. Η λίστα των διαθέσιμων widgets και panels είναι αρκετά μεγάλη και θυμίζει αυτή του GWT. Μέσω των διαθέσιμων panels μπορούμε να καθορίσουμε την διάταξη των widgets στην σελίδα. Ένα panel μπορεί να περιέχει μέσα του άλλα panels αλλά όλ' αυτά πρέπει να περιέχονται σε ένα RootPanel. Για μια πλήρη αναφορά όλων των διαθέσιμων widgets και panels μπορούμε να δούμε τη σελίδα: <http://pyjs.org/api/>

Πώς όμως το Pyjamas κάνει την μετατροπή από python σε Javascript; Όπως ακριβώς το GWT, έτσι και το Pyjamas μεταφράζει μια εφαρμογή Python μαζί με τις βιβλιοθήκες της (συμπεριλαμβανομένων των UI Widgets και των κλάσεων DOM) σε μια εφαρμογή μαζί με τις βιβλιοθήκες της σε Javascript.



Σχήμα 3 – Μετατροπή από python σε Javascript

1 Το pyjs μεταφράζει τον python κώδικα σε javascript ακολουθώντας το abstract syntax tree της Python.

2 Το `pyjslib` αναλαμβάνει τους τύπους της Python που απαιτούν μια ειδική υλοποίηση javascript ώστε να δουλεύει το `pyjs`. Για παράδειγμα παρότι οι λίστες της Python είναι όμοιες με τους πίνακες της javascript, οι λίστες της Python μετατρέπονται σε ειδικά αντικείμενα που εκτελούν μεθόδους όπως η `append`.

3 Τα UI widgets και η βιβλιοθήκη DOM προσφέρονται για ευκολία. Είναι γραμμένα σε Python και όπως όλα τα υπόλοιπα μεταφράζονται σε Javascript για την ανάπτυξη. Είναι βασισμένα σε αυτά του GWT.

4 Το Build χειρίζεται την καθολική μετάφραση των ατομικών στοιχείων και δημιουργεί ένα σύνολο από `.html` και `.js` αρχεία που μπορούν χρησιμοποιηθούν από κάθε Web Server.

Ο `python-σε-javascript` μεταγλωττιστής του `pyjamas` είναι ουσιαστικά ένας μεταφραστής γλωσσών. Έχει δύο κύρια mode το `-O` και το `--strict`. Το `--strict mode` αποσκοπεί σε πλήρης διαλειτουργικότητα της python ακόμα και εις βάρος της απόδοσης. Αντίθετα το `-O mode` είναι βελτιστοποιημένο για ταχύτητα, ακόμα και αν έτσι χαθεί η υποστήριξη σε μερικές λειτουργίες της γλώσσας python.

Στην συνέχεια ακολουθεί το παράδειγμα Hello World σε python.

```
from pyjamas import Window
from pyjamas.ui import RootPanel, Button
def greet(sender):
    Window.alert("Hello, World!")
b = Button("Click me", greet)
RootPanel().add(b)
```

Αυτό που πρέπει να παρατηρήσουμε είναι ότι και εδώ, όπως και στο GWT, όλα προστίθενται στο `RootPanel`. Το `RootPanel()` μας δίνει πρόσβαση στο μοντέλο DOM του φυλλομετρητή (ξεκινώντας από το `body`).

3.2.1. Σύγκριση του Pyjamas με το GWT

Όπως ήδη έχουμε αναφέρει το GWT και το Pyjamas έχουν αρκετά κοινά μεταξύ τους. Πρώτ' απ' όλα και με τα δυο μπορούμε να δημιουργήσουμε AJAX

εφαρμογές. Επίσης, και τα δυο διαθέτουν μεταγλωττιστές που μεταγλωττίζουν από μια γλώσσα σε Javascript. Αυτή είναι παράλληλα και η κυριότερη διαφορά τους. Στο GWT ο προγραμματιστής γράφει κώδικα σε Java και αυτός μετά μεταγλωττίζεται σε Javascript, ενώ στο Pyjamas ο προγραμματιστής γράφει κώδικα σε python και αυτός μεταγλωττίζεται σε Javascript. Επιπλέον, αν και και τα δύο χρησιμοποιούνται κυρίως για web εφαρμογές, μπορούμε μέσω αυτών να δημιουργήσουμε και desktop εφαρμογές.

Όμως αν και αυτά τα δύο frameworks μοιάζουν αρκετά παρόμοια σε πρώτη φάση, έχουν και αρκετές διαφορές μεταξύ τους. Το GWT φτιάχτηκε από μια μεγάλη εταιρία πληροφορικής, την Google, ενώ το Pyjamas είναι αποτέλεσμα συνεργασίας 50 συντελεστών. Το GWT αποτελείται από εκατοντάδες χιλιάδες γραμμές κώδικα ενώ το Pyjamas από μόλις μερικές δεκάδες χιλιάδες. Τέλος, παρουσιάζουν μεγάλη διαφορά στην υποστήριξη της διεθνοποίησης, το GWT περιέχει δεκάδες χιλιάδες γραμμές κώδικα και εκατοντάδες αρχεία, υποστηρίζοντας πάνω από εκατό διαφορετικές γλώσσες. Αντίθετα το Pyjamas, όσον αφορά την διεθνοποίηση περιέχει περίπου πενήντα γραμμές κώδικα και δεν υποστηρίζει άλλες γλώσσες.

3.3. Morfik BX

Η Morfik Technology Pty Ltd. είναι μια εταιρία από Αυστραλία που αναπτύσει το Morfik, ένα σύνολο από εργαλεία οπτικής σχεδίασης, μεταγλωττιστές και ένα framework που συνδιάζονται σε ένα IDE και έχουν σαν σκοπό την ανάπτυξη AJAX εφαρμογών σε γλώσσες υψηλού επιπέδου όπως η C#, η BASIC ή η Object Pascal. Το Morfik περιλαμβάνει εργαλεία οπτικής σχεδίασης για web διεπιφάνειες, βάσεις δεδομένων και ερωτήματα. Υποστηρίζει το κλασσικό client-server μοντέλο αν και, όπως σε όλες τις Ajax εφαρμογές ο client-side κώδικας τρέχει μέσα στον φυλλομετρητή. Αυτή η εταιρία έχει σχεδιάσει τρία προϊόντα, το Morfik FX, το Morfik CX και το Morfik BX. Το Morfik FX σχετίζεται με την Pascal, το Morfik CX με την C# και το Morfik BX με την BASIC. Εμείς σ' αυτήν την πτυχιακή θ' ασχοληθούμε με το Morfik BX. Ο κώδικας που προορίζεται για την μεριά του φυλλομετρητή μεταγλωττίζεται σε ένα συνδυασμό από HTML, XML, CSS και Javascript, ενώ ο

κώδικας που προορίζεται για τον έλεγχο της εφαρμογής (server) μεταγλωτίζεται σε ένα δυαδικό εκτελέσιμο αρχείο ή σε μια δυναμικά συνδεδεμένη βιβλιοθήκη.

Το Morfik παρέχει ένα σύνολο από εργαλεία για το σχεδιασμό της εφαρμογής μέσω ενός WhatYouSeelsWhatYouGet (WYSIWYG) τρόπου προσέγγισης. Επίσης το Morfik επιτρέπει το χτίσιμο εφαρμογών για πρόσβαση μέσω iPhone, iPod Touch και iPad. Οι σχεδιαστές μπορούν να σχεδιάσουν διαφορετικές διεπιφάνειες για κάθε συσκευή μέσα στο ίδιο Morfik project και το Morfik framework αποφασίζει ποια σελίδα να ανοίξει βασισμένο στο είδος της συσκευής που έχει πρόσβαση στην εφαρμογή. Τέλος, όπως και τα προηγούμενα framework, το Morfik είναι κατάλληλο και για desktop εφαρμογές.

Στο Morfik BX υπάρχει η έννοια του module. Ένα module στο Morfik μπορεί να έχει τρεις μορφές:

- Browser side module
- Server side module
- Dual module

Και στις τρεις περιπτώσεις το module αναπαριστάται από ένα .mmd αρχείο. Στο Morfik BX μια επίσημη δομή κώδικα προσδιορίζει το module για κάθε πλευρά: του φυλλομετρητή και του εξυπηρετητή. Αυτή η δομή ονομάζεται namespace. Το πιο απλό module που μπορεί να δημιουργηθεί με το Morfik BX είναι το παρακάτω:

```
Namespace Globals
  ` This is an empty module
End Namespace
```

Όπως είπαμε, στις Morfik εφαρμογές τα modules μπορεί να είναι browser side, server side ή dual. Τα dual modules είναι και browser και server ταυτόχρονα. Το IDE του Morfik μας επιτρέπει να συντάξουμε το module για την μεριά που θέλουμε: browser ή server.

Τα modules στο Morfik έχουν τμήμα αρχικοποίησης και οριστικοποίησης. Ο κώδικας που περιλαμβάνεται στο τμήμα αρχικοποίησης καλείται όταν η εφαρμογή φορτώνεται για πρώτη φορά ενώ ο κώδικας στο τμήμα οριστικοποίησης καλείται όταν η εφαρμογή τερματίζεται. Το τμήμα της οριστικοποίησης είναι αποτελεσματικό μόνο στον server, όπου μπορούμε να διαβεβαιώσουμε ότι η

εφαρμογή όντως θα τερματιστεί. Ουσιαστικά το τμήμα οριστικοποίησης στην μεριά του φυλλομετρητή δεν θα κληθεί ποτέ. Όμως στην μεριά του server το κομμάτι της οριστικοποίησης θα κληθεί κατά το κλείσιμο της εφαρμογής.

Στο Morfik BX μπορούμε να έχουμε σχόλια, μεταβλητές και σταθερές, συναρτήσεις, τελεστές, δομές επιλογής, βρόγχους επανάληψης και όλα αυτά που αφορούν τον object oriented προγραμματισμό όπως κληρονομικότητα, ενθυλάκωση και πολυμορφισμό. Όλα αυτά όμως ακολουθούν την λογική της BASIC και γι' αυτό δεν θα τα αναλύσουμε περαιτέρω.

3.3.1. Σύγκριση του Morfik BX με το GWT

Το Morfik έχει αρκετές ομοιότητες με το GWT. Και τα δύο δημιουργήθηκαν για τον σχεδιασμό web εφαρμογών από άτομα που δεν γνωρίζουν Javascript. Επίσης και τα δυο υποστηρίζουν και desktop εφαρμογές. Και στα δύο ο προγραμματιστής γράφει σε κάποια άλλη γλώσσα προγραμματισμού η οποία μεταγλωττίζεται στην συνέχεια σε Javascript. Επιπλέον και τα δύο υποστηρίζουν την δημιουργία εφαρμογών για πρόσβαση μέσω iPhone. Τέλος και τα δυο frameworks είναι open source.

Η κυριότερη τους διαφορά είναι ότι με το Morfik μπορούμε να επιλέξουμε ανάμεσα σε ένα μεγάλο φάσμα γλωσσών προγραμματισμού όπως η BASIC, η Pascal και η C#, ενώ το GWT υποστηρίζει την γραφή κώδικα μόνο σε Java. Μια επιπλέον διαφορά τους είναι ότι το Morfik μεταγλωττίζει σε HTML, Javascript και XML ενώ το GWT μεταγλωττίζει μόνο σε Javascript.

3.4. Σύγκριση των Javascript και HTML generator frameworks με τα υπόλοιπα

Σ' αυτό το κεφάλαιο αναφερθήκαμε σε frameworks που δημιουργούν Javascript και HTML κώδικα: στο GWT, το Pyjamas και το Morfik BX. Και τα τρία μας δίνουν την δυνατότητα δημιουργίας AJAX εφαρμογών χωρίς να χρειάζεται να γνωρίζουμε

Javascript και HTML. Αυτή είναι και η ιδιαιτερότητά τους, που τα κάνει να διαφέρουν από τα frameworks που αναφέραμε στο προηγούμενο κεφάλαιο.

Τα frameworks του προηγούμενου κεφαλαίου μας παρείχαν μια πιο εύκολη πρόσβαση στο DOM, δίνοντάς μας την δυνατότητα να συνθέσουμε έτοιμα κομμάτια κώδικα ώστε να έχουμε καλά αποτελέσματα σε λιγότερο χρόνο, αλλά πάντα μέσω της χρήσης της Javascript και της HTML. Για να χρησιμοποιήσει κανείς τα frameworks που αναφέραμε στο προηγούμενο κεφάλαιο πρέπει να έχει γνώση και της HTML και της Javascript έστω σε κάποιον βαθμό.

Η δύναμη των Javascript και HTML generator frameworks βρίσκεται ακριβώς σ' αυτό το σημείο. Μπορεί δηλαδή κανείς να δημιουργήσει web εφαρμογές χωρίς να έχει καθόλου γνώση HTML και Javascript, χρησιμοποιώντας απλά κάποια γλώσσα υψηλού επιπέδου που ήδη γνωρίζει.

4. Προσβασιμότητα στον παγκόσμιο ιστό

Σ' αυτό το κεφάλαιο θ' αναφερθούμε στην προσβασιμότητα. Θα αναφέρουμε τί είναι η προσβασιμότητα, με τί σχετίζεται, προβλήματα που προκύπτουν λόγω της έλλειψης της προσβασιμότητας και πώς αντιμετωπίζονται. Παράλληλα θα κάνουμε μια εκτενέστερη αναφορά στους τυφλούς χρήστες. Θ' αναφερθούμε στην βοηθητική τεχνολογία που χρησιμοποιούν και στα προβλήματα που αντιμετωπίζουν.

Η προσβασιμότητα στον παγκόσμιο ιστό αναφέρεται στην δημιουργία websites που να μπορούν να χρησιμοποιηθούν από άτομα όλων των δυνατοτήτων αλλά και από άτομα με αναπηρία. Όταν μια σελίδα έχει σχεδιαστεί, αναπτυχθεί και εκδοθεί σωστά, όλοι οι χρήστες μπορούν να έχουν ισάξια πρόσβαση στην πληροφορία και στην λειτουργικότητά της.

Όταν για παράδειγμα μια σελίδα έχει γραφεί με σημασιολογική HTML, με την παροχή ισοδύναμου κειμένου για τις εικόνες και με υπερσυνδέσμους που έχουν ονομαστεί κατάλληλα, αυτό βοηθάει τους τυφλούς χρήστες που χρησιμοποιούν text-to-speech λογισμικό και/ή text-to-Braille υλικό. Όταν το κείμενο και οι εικόνες είναι μεγάλα σε μέγεθος και/ή μπορούν να μεγαλώσουν, είναι πολύ πιο εύκολο για τους χρήστες με χαμηλή όραση να διαβάσουν και να καταλάβουν το περιεχόμενο. Όταν οι υπερσύνδεσμοι είναι υπογραμμισμένοι (ή διαφοροποιημένοι) καθώς και χρωματισμένοι, αυτό εξασφαλίζει ότι οι χρήστες που έχουν αχρωματοψία θα είναι ικανοί να τους ξεχωρίσουν. Όταν οι υπερσύνδεσμοι και οι περιοχές στις οποίες μπορεί να κάνει κλικ ο χρήστης είναι μεγάλες, αυτό βοηθάει τους χρήστες που δεν μπορούν να ελέγξουν το ποντίκι με ακρίβεια. Όταν οι σελίδες είναι γραμμένες με τέτοιο κώδικα ώστε να μπορούν οι χρήστες να πλοηγηθούν μέσω του πληκτρολογίου ή μιας απλής βοηθητικής μηχανής μονάχα, αυτό βοηθάει τους χρήστες που δεν μπορούν να χρησιμοποιήσουν ποντίκι ή ακόμα και ένα κλασσικό πληκτρολόγιο. Όταν τα βίντεο έχουν ενσωματωμένους υπότιτλους ή είναι διαθέσιμη μια έκδοση σε νοηματική γλώσσα, οι κωφοί και βαρήκοοι χρήστες μπορούν να καταλάβουν το βίντεο. Όταν εφέ που αναβοσβήνουν και είναι έντονα

αποφεύγονται ή γίνονται προαιρετικά, οι χρήστες που είναι επιρρεπείς σε επιληπτικές κρίσεις λόγω τέτοιων εφέ δεν βρίσκονται σε κίνδυνο. Και όταν το περιεχόμενο μιας σελίδας είναι γραμμένο και εικονογραφημένο με διδακτικά διαγράμματα και κινούμενα σχέδια, για τους χρήστες που έχουν δυσλεξία και μαθησιακές δυσκολίες είναι πολύ πιο εύκολο να καταλάβουν το περιεχόμενο. Όταν οι σελίδες είναι σωστά χτισμένες και συντηρημένες όλοι αυτοί οι χρήστες μπορούν να έχουν πρόσβαση σ' αυτές, χωρίς αυτό να έχει αντίκτυπο στην χρησιμότητα της σελίδας, τόσο για τους χρήστες που δεν έχουν αναπηρία όσο και για αυτούς που έχουν.

Τα άτομα με αναπηρία χρησιμοποιούν βοηθητικές τεχνολογίες όπως οι ακόλουθες για να έχουν πρόσβαση στον παγκόσμιο ιστό:

- Screen readers: Είναι λογισμικό που χρησιμοποιείται από άτομα που είναι τυφλά ή που έχουν δυσλεξία και ερμηνεύει ότι υπάρχει στην οθόνη είτε σε σύνθετη ψηφιακή ομιλία για ηχητική έξοδο, είτε σε refreshable braille για απτή έξοδο. Μερικοί screen readers χρησιμοποιούν τη δένδρική δομή του εγγράφου σαν είσοδο. Παλιότεροι screens readers κάνουνε χρήση της έκδοσης που απεικονίζεται στην οθόνη και έτσι μπορεί να χαθεί η δομή (για παράδειγμα όταν οι πίνακες χρησιμοποιούντε για δομή) και η έξοδός τους μπορεί να προκαλέσει σύγχυση στον χρήστη.
- Τερματικά Braille και refreshable Braille: Το Braille είναι ένα σύστημα που χρησιμοποιεί από έξι εως οχτώ ακίδες που ανεβαίνουν σχηματίζοντας διαφορετικούς συνδιασμούς για να αναπαραστήσουν γράμματα και αριθμούς που μπορούν να διαβαστούν με τα ακροδάχτυλα. Υπάρχουν ποικίλα συστήματα Braille ανά τον κόσμο. Μερικές εκδόσεις του braille περιλαμβάνουν επιπλέον κωδικοποιήσεις πέρα από τους συνηθισμένους αλφαριθμητικούς χαρακτήρες για να αναπαραστήσουν συνηθισμένες ομάδες γραμμάτων (για παράδειγμα “th” και “ble” στην δεύτερη έκδοση του American English Braille) με σκοπό να κάνουν την braille πιο συμπαγή. Υπάρχει μια έκδοση της braille με 8 ακίδες που επιτρέπει την αναπαράσταση όλων των ASCII χαρακτήρων. Η refreshable ή δυναμική braille αποτελείται από μια οθόνη braille όπου οι ακίδες μπορούν να ανεβαίνουν και να κατεβαίνουν δυναμικά για να επιτρέπουν την

αναπαράσταση κάθε braille χαρακτήρα. Οι refreshable braille οθόνες μπορούν να ενσωματωθούν σε φορητές braille συσκευές με τις δυνατότητες ενός μικρού υπολογιστή, ενώ παράλληλα μπορούν να χρησιμοποιηθούν και σαν διεπαφές σε συσκευές όπως τα information kiosks. Μια τέτοια braille συσκευή φαίνεται στην εικόνα που ακολουθεί:



Εικόνα 4 – Braille συσκευή

- Λογισμικό μεγένθυσης οθόνης: Είναι λογισμικό που χρησιμοποιείται κυρίως από άτομα με χαμηλή όραση και μεγενθύνει ένα κομμάτι της οθόνης για καλύτερη προβολή. Αυτό το λογισμικό κάνει το περιεχόμενο της σελίδας να φαίνεται πιο μεγάλο, ενώ παράλληλα μειώνει την περιοχή του αρχείου που είναι ορατή, απομακρύνοντας το τριγύρω περιεχόμενο. Μερικά τέτοια προγράμματα διαθέτουν δυο δυνατούς τρόπους προβολής: έναν μεγενθυμένο και έναν με το προκαθορισμένο μέγεθος για πλοήγηση.
- Λογισμικό αναγνώρισης φωνής: Είναι λογισμικό που χρησιμοποιείται από άτομα με σωματικές αναπηρίες ή με προσωρινούς τραυματισμούς στα χέρια σαν μέθοδος εισαγωγής σε κάποιους φυλλομετρητές φωνής. Μπορεί να δεχτεί λεκτικές εντολές για τον υπολογιστή ή να μετατρέπει υπαγόρευση σε γραμματικά ορθό κείμενο, που είναι χρήσιμο για άτομα που έχουν δυσκολία στην χρήση του ποντικιού ή του πληκτρολόγιου.
- Ειδικά πληκτρολόγια: Είναι υλικό ή συσκευές λογισμικού που χρησιμοποιούνται από άτομα με σωματικές αναπηρίες. Κάνουν την πληκτρολόγηση πιο εύκολη και πιο ακριβή για άτομα που αντιμετωπίζουν

δυσκολίες στον έλεγχο των κινήσεων. Παρέχουν ένα εναλλακτικό τρόπο δημιουργίας σύνθετων πληκτρολογήσεων (όπου πολλά πλήκτρα μαζί έχουν σαν αποτέλεσμα την εμφάνιση ενός χαρακτήρα) που φαίνεται σαν να προήλθαν από ένα κλασικό πληκτρολόγιο. Παραδείγματα τέτοιων πληκτρολογίων περιλαμβάνουν πληκτρολόγια με μικρές ή μεγάλες αποστάσεις πλήκτρων, keyguards που επιτρέπουν το πάτημα μόνο ενός κουμπιού κάθε φορά, eyegaze πληκτρολόγια και sip-and-puff διακόπτες. Web-based εφαρμογές που μπορούν να χειριστούν αποκλειστικά από το πληκτρολόγιο, χωρίς να απαιτείται ποντίκι, υποστηρίζουν μεγάλη ποικιλία εναλλακτικών τρόπων εισαγωγής. Δυο περιπτώσεις τέτοιων πληκτρολογίων φαίνονται στην εικόνα που ακολουθεί:



Εικόνα 5 – Ειδικά πληκτρολόγια

- Λογισμικό σάρωσης (Scanning software): Είναι προσαρμοστικό λογισμικό που χρησιμοποιείται από άτομα που έχουν σωματικές ή γνωστικές αναπηρίες και χρησιμοποιείται για να επισημάνει ή να δηλώσει ένα σύνολο από επιλογές (για παράδειγμα στοιχεία μενού, υπερσυνδέσμους) επισημαίνοντας ή δηλώνοντας ένα κάθε φορά. Ο χρήστης επιλέγει το στοιχείο που επιθυμεί πατώντας έναν διακόπτη όταν το αντικείμενο που επιθυμεί επισημανθεί ή δηλωθεί.
- Σύνθεση φωνής (έξοδος φωνής): Η σύνθεση φωνής ή έξοδος φωνής μπορεί να παραχθεί από screen readers ή voice browsers και περιλαμβάνει

την παραγωγή ψηφιακής φωνής από το κείμενο. Τα άτομα που είναι συνηθισμένα στο να χρησιμοποιούν έξοδο φωνής μερικές φορές την ακούνε σε πολύ μεγάλες ταχύτητες.

- Tabbing ανάμεσα σε δομικά στοιχεία: Μερικές λύσεις προσβασιμότητας είναι περισσότερο προσαρμοστικές στρατηγικές παρά συγκεκριμένες βοηθητικές τεχνολογίες όπως υλικό ή λογισμικό. Για παράδειγμα, για τα άτομα που δεν μπορούν να χρησιμοποιήσουν ποντίκι, μια στρατηγική γρήγορης σάρωσης ανάμεσα στα δομικά στοιχεία της web σελίδας είναι η χρήση του πλήκτρου tab για να μεταφέρονται σ' αυτά τα στοιχεία με την σειρά. Αυτά τα δομικά στοιχεία μπορεί να περιλαμβάνουν τους υπερσυνδέσμους, τις κεφαλίδες, τα στοιχεία μιας λίστας και διάφορα άλλα δομικά στοιχεία. Τα άτομα που χρησιμοποιούν screen readers, ανεξάρτητα απ' το αν είναι τυφλά ή δυσλεκτικά, μπορούν να κινηθούν ανάμεσα στα στοιχεία της σελίδας χρησιμοποιώντας το πλήκτρο tab. Το ίδιο ισχύει και για τα άτομα που χρησιμοποιούν λογισμικό αναγνώρισης φωνής.
- Φυλλομετρητές κειμένων: Οι φυλλομετρητές κειμένου όπως ο Lynx είναι εναλλακτικοί των φυλλομετρητών που διαθέτουν γραφικές διεπιφάνειες. Μπορούν να χρησιμοποιηθούν σε συνδυασμό με προγράμματα screen readers για τα άτομα που είναι τυφλά. Χρησιμοποιούνται επίσης από πολλούς ανθρώπους που έχουν συνδέσεις χαμηλού εύρους ζώνης και δεν θέλουν να περιμένουν να κατέβουν οι εικόνες.
- Οπτική ειδοποίηση: Η οπτική ειδοποίηση είναι ένα εναλλακτικό χαρακτηριστικό μερικών λειτουργικών συστημάτων που επιτρέπει σε κωφά ή βαρήκοα άτομα να λαμβάνουν μια οπτική ειδοποίηση για ένα μήνυμα πρεοειδοποίησης ή λάθους που διαφορετικά θα παρουσιαζόταν σαν ηχητική ειδοποίηση.
- Φυλλομετρητές φωνής: Οι φυλλομετρητές φωνής είναι συστήματα που επιτρέπουν πλοήγηση καθοδηγούμενη από φωνή. Μερικοί περιλαμβάνουν και είσοδο και έξοδο φωνής και μερικοί επιτρέπουν πρόσβαση στον παγκόσμιο ιστό μέσω τηλεφώνου.

4.1. Διάφορα είδη αναπηρίας

Στην συνέχεια θ' αναφερθούμε σε μερικά είδη αναπηρίας που σχετίζονται με την προσβασιμότητα στον παγκόσμιο ιστό. Θα εξηγήσουμε τί περιλαμβάνει το καθένα απ' αυτά και τί μπορούμε να κάνουμε για να αυξήσουμε την προσβασιμότητα.

4.1.1. Προβλήματα όρασης

Σ' αυτήν την κατηγορία ανήκει η τύφλωση, η χαμηλή όραση και η αχρωματοψία.

Η τύφλωση σχετίζεται με μια ουσιαστική, ανεπανόρθωτη απώλεια της όρασης και στα δυο μάτια.

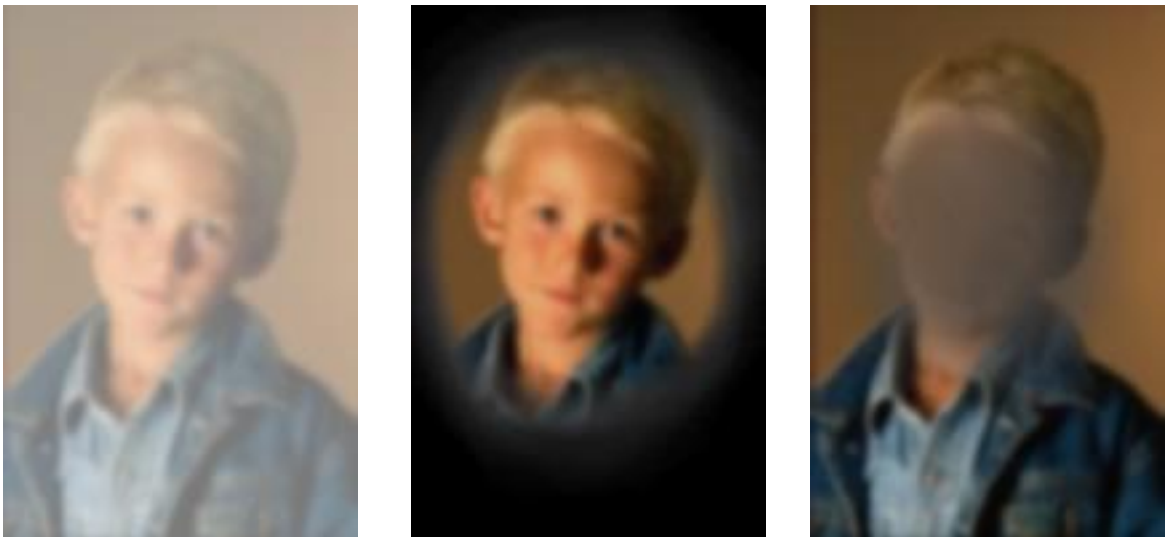
Για να έχουν πρόσβαση στον παγκόσμιο ιστό πολλά άτομα που είναι τυφλά βασίζονται σε screen readers. Επίσης, μερικά άτομα που είναι τυφλά χρησιμοποιούν text-based φυλλομετρητές, όπως ο Lynx, ή φυλλομετρητές φωνής αντί για φυλλομετρητές που διαθέτουν γραφική διεπιφάνεια χρήστη σε συνδιασμό με screen readers. Μπορεί να διαθέτουν στρατηγικές γρήγορης πλοήγησης, όπως η μετακίνηση ανάμεσα στις κεφαλίδες και τους υπερσυνδέσμους μέσω του tab, αντί να διαβάζονται οι λέξεις της σελίδας με την σειρά.

Παραδείγματα προβλημάτων που μπορεί να παρουσιαστούν σε έναν τυφλό χρήστη στο διαδίκτυο είναι τα εξής:

- Εικόνες που δεν έχουν εναλλακτικό κείμενο.
- Πολύπλοκες εικόνες (όπως γραφήματα ή πίνακες) που δεν έχουν επαρκή σχολιασμό.
- Βίντεο το οποίο δεν περιγράφεται με κείμενο ή ηχητικά.
- Πίνακες που δεν βγάζουν νόημα όταν διαβάζονται σειριακά (κελί κελί ή γραμμικά).
- Frames που δεν έχουν "NOFRAME" εναλλακτική, ή που έχουν άσχετα ονόματα.

- Φόρμες που δεν μπορούν να προσπελασθούν με το πλήκτρο tab με λογική σειρά ή που έχουν κακή ή ελλιπή ονομασία.
- Φυλλομετρητές και εργαλεία συγγραφής που δεν έχουν υποστήριξη όλων των εντολών μέσω πληκτρολογίου.
- Φυλλομετρητές και εργαλεία συγγραφής που δεν χρησιμοποιούν τις συνηθισμένες διεπιφάνειες εφαρμογών του λειτουργικού συστήματος στο οποίο βασίζονται.
- Μη συνηθισμένες μορφές αρχείων που δυσκολεύουν την ερμηνεία από το screen reader.

Στην συνέχεια θ' αναφερθούμε σε παθήσεις που σχετίζονται με την μειωμένη όραση. Υπάρχουν πολλές μορφές μειωμένης όρασης, όπως χαμηλή οξύτητα (όραση που δεν είναι έντονη), όραση τούνελ (όπου κανείς βλέπει μόνο το κέντρο του οπτικού πεδίου), απώλεια κεντρικού οράματος (όπου κανείς βλέπει μόνο την άκρη του οπτικού πεδίου) και θολή όραση. Στην παρακάτω εικόνα φαίνονται αυτές οι παθήσεις.



Εικόνα 6 – Χαμηλή οξύτητα, όραση τούνελ και απώλεια κεντρικού οράματος

Για να χρησιμοποιήσουν το διαδίκτυο μερικοί άνθρωποι με μειωμένη όραση χρησιμοποιούν οθόνες πολύ μεγάλου μεγέθους και μεγαλώνουν τις εικόνες και την γραμματοσειρά του συστήματος. Άλλοι χρησιμοποιούν μεγενθητές οθόνης ή

λογισμικό ενίσχυσης οθόνης. Μερικοί κάνουν χρήση ειδικού συνδυασμού χρωμάτων κειμένου και φόντου, όπως 24άρα κίτρινη γραμματοσειρά και μαύρο φόντο, ή διαλέγουν μια γραμματοσειρά που είναι ιδιαίτερα ευανάγνωστη για την δικιά τους οπτική πάθηση.

Παραδείγματα προβλημάτων που μπορεί να παρουσιαστούν σε έναν χρήστη με μειωμένη όραση στο διαδίκτυο είναι τα εξής:

- Σελίδες με απόλυτο μέγεθος γραμμάτων το οποίο δεν αλλάζει (μεγαλώνει ή μικραίνει) εύκολα.
- Σελίδες οι οποίες, λόγω ακατάλληλης διάταξης, είναι δύσκολες στην πλοήγηση όταν μεγεθυνθούν, λόγω της απώλειας προσανατολισμού στο περιεχόμενο της σελίδας.
- Σελίδες ή εικόνες σε σελίδες που έχουν χαμηλή αντίθεση, η οποία δεν αλλάζει εύκολα μέσω της επικάλυψης των style sheets του συγγραφέα με αυτά του χρήστη.
- Κείμενο το οποίο παρουσιάζεται σε μορφή εικόνας, το οποίο όταν μεγεθυνθεί αλλάζει γραμμή αυτόματα.
- Πολλά από τα προβλήματα που αναφέρθηκαν στους τυφλούς χρήστες ισχύουν και για τους χρήστες με μειωμένη όραση, ανάλογα με το βαθμό της ασθένειάς τους.

Τέλος θα αναφερθούμε στην αχρωματοψία. Η αχρωματοψία είναι η έλλειψη ευαισθησίας σε συγκεκριμένα χρώματα. Συχνές μορφές αχρωματοψίας περιλαμβάνουν την δυσκολία διάκρισης ανάμεσα στο κόκκινο και το πράσινο ή στο κίτρινο με το μπλε. Μερικές φορές η αχρωματοψία μπορεί να υπάρξει σε τόσο έντονο βαθμό που καθιστά ανέφικτη τη διάκριση κάθε είδους χρώματος.

Για να χρησιμοποιήσουν το διαδίκτυο μερικοί χρήστες με αχρωματοψία χρησιμοποιούν δικά τους style sheets για να υπερβούν τις χρωματικές επιλογές της γραμματοσειράς και του φόντου του προγραμματιστή.

Τα προβλήματα που αντιμετωπίζουν οι χρήστες με αχρωματοψία στο διαδίκτυο περιλαμβάνουν τα εξής:

- Χρώματα που χρησιμοποιούνται για να δώσουν έμφαση σε κείμενο.

- Κείμενο το οποίο δεν έχει επαρκή αντίθεση με το φόντο.
- Φυλλομετρητές που δεν υποστηρίζουν την επικάλυψη των style sheets του προγραμματιστή με αυτά του χρήστη.

4.1.2. Προβλήματα Ακοής

Σ' αυτήν την κατηγορία ανήκει η κώφωση και η μειωμένη ακοή.

Η κώφωση περιλαμβάνει ανεπανόρθωτη βλάβη της ακοής και στα δυο αυτιά. Μερικοί κουφοί χρήστες έχουν σαν κύρια γλώσσα τους την νοηματική, και ίσως να μην διαβάζουν ή μιλούν άπταιστα την κανονική γλώσσα.

Για να χρησιμοποιήσουν το διαδίκτυο πολλοί κουφοί χρήστες βασίζονται σε επεξηγηματικό κείμενο για ηχητικό περιεχόμενο. Ίσως χρειαστεί να ενεργοποιήσουν τους υπότιτλους σε ηχητικά αρχεία καθώς προσπελαίνουν μια σελίδα, για να μπορέσουν να συγκεντρωθούν καλύτερα στην ανάγνωση της σελίδας ή να βασιστούν σε συμπληρωματικές εικόνες που επισημαίνουν το περιεχόμενο.

Τα προβλήματα που αντιμετωπίζουν οι κουφοί χρήστες στο διαδίκτυο είναι τα εξής:

- Έλλειψη επεξηγηματικού κειμένου για το αντίστοιχο περιεχόμενο ήχου.
- Έλλειψη άμεσης σχέσης εικόνων με το περιεχόμενο της σελίδας, σε σελίδες που περιέχουν πολύ κείμενο, καθώς η κατανόηση μπορεί να είναι δύσκολη για άτομα που έχουν σαν κύρια γλώσσα την νοηματική αντί της γραπτής/προφορικής γλώσσας.
- Έλλειψη καθαρής και απλής γλώσσας.
- Απαίτηση ηχητικής εισαγωγής πληροφοριών στις διάφορες σελίδες.

Στην συνέχεια θ' αναφερθούμε στην μειωμένη ακοή. Άτομα με ήπια εως μέτρια βλάβη στην ακοή μπορούν να θεωρηθούν άτομα με μειωμένη ακοή.

Για να χρησιμοποιήσουν τον παγκόσμιο ιστό οι βαρήκοοι χρήστες βασίζονται σε επεξηγηματικό κείμενο για περιεχόμενο ήχου και/ή την ενίσχυση του ήχου. Ίσως χρειαστεί να ενεργοποιήσουν τους υπότιτλους για τα ηχητικά αρχεία ή να ρυθμίσουν την ένταση του ηχητικού αρχείου.

Το κυριότερο πρόβλημα που αντιμετωπίζουν οι χρήστες με μειωμένη ακοή είναι το εξής:

- Έλλειψη επεξηγηματικού κειμένου για το αντίστοιχο ηχητικό περιεχόμενο.

4.1.3. Σωματικές αναπηρίες

Σ' αυτήν την κατηγορία ανήκουν οι κινητικές αναπηρίες. Στις κινητικές αναπηρίες συμπεριλαμβάνεται η αδυναμία, οι περιορισμοί του μυϊκού ελέγχου (όπως οι ακούσιες κινήσεις, η έλλειψη συντονισμού ή η παράλυση), οι περιορισμοί της αίσθησης, τα προβλήματα που σχετίζονται με τις αρθρώσεις και η έλλειψη των άκρων. Μερικές σωματικές αναπηρίες μπορεί να περιλαμβάνουν πόνο που εμποδίζει την κίνηση. Αυτές οι καταστάσεις μπορεί να επηρεάσουν τόσο τα χέρια και τα μπράτσα όσο και άλλα μέρη του σώματος.

Για να χρησιμοποιήσουν τον παγκόσμιο ιστό, τα άτομα με κινητικά προβλήματα που σχετίζονται με τα χέρια και τα μπράτσα, μπορούν να χρησιμοποιήσουν ένα ειδικό ποντίκι, ένα πληκτρολόγιο με διάταξη πλήκτρων που ταιριάζει στο δικό τους εύρος κίνησης του χεριού, μια συσκευή κατάδειξης όπως ένα ποντίκι κεφαλής, ένα ραβδί στόματος, λογισμικό αναγνώρισης φωνής ή άλλες βοηθητικές τεχνολογίες για να έχουν πρόσβαση και να αλληλεπιδρούν με τις σελίδες του παγκόσμιου ιστού. Μπορεί να ενεργοποιήσουν εντολές πληκτρολογώντας ένα κουμπί κάθε φορά αλλά σε σειρά με ένα δείκτη κεφαλής αντί να πατάνε πολλά κουμπιά ταυτόχρονα για να ενεργοποιήσουν κάποια εντολή. Μπορεί να χρειάζονται παραπάνω χρόνο όταν συμπληρώνουν διαδραστικές φόρμες σε σελίδες στον παγκόσμιο ιστό αν πρέπει να συγκεντρώνονται ή να ελίσσονται προσεκτικά για να επιλέξουν κάθε πλήκτρο.

Τα προβλήματα που μπορεί ν' αντιμετωπίζουν οι άνθρωποι που έχουν κινητικά προβλήματα που σχετίζονται με τα χέρια και τα μπράτσα είναι τα εξής:

- Περιορισμένο χρόνο για την επιλογή απάντησης στις σελίδες του παγκόσμιου ιστού.

- Φυλλομετρητές και εργαλεία συγγραφής που δεν υποστηρίζουν εναλλακτικές εντολές πληκτρολογίου για τις εντολές που γίνονται με το ποντίκι.
- Φόρμες που δεν μπορούμε να προσπελάσουμε χρησιμοποιώντας το πλήκτρο tab με μια λογική σειρά.

4.1.4. Λεκτικά προβλήματα

Τα προβλήματα στην ομιλία μπορεί να περιλαμβάνουν δυσκολία στην παραγωγή λόγου που να αναγνωρίζεται από κάποιο λογισμικό αναγνώρισης φωνής, είτε από άποψη καθαρότητας είτε από άποψη σαφήνειας.

Για να χρησιμοποιήσουν σελίδες στις οποίες απαιτείται αναγνώριση φωνής, τα άτομα που έχουν προβλήματα άρθρωσης θα πρέπει να μπορούν να χρησιμοποιήσουν ένα είδος εναλλακτικής εισόδου όπως κείμενο που εισάγεται μέσω πληκτρολογίου.

Το κυριότερο πρόβλημα που αντιμετωπίζουν αυτά τα άτομα χρησιμοποιώντας τον παγκόσμιο ιστό είναι το εξής:

- Σελίδες του παγκόσμιου ιστού που απαιτούν αλληλεπίδραση που βασίζεται στην φωνή και δεν έχουν εναλλακτική μέθοδο εισαγωγής.

4.1.5. Γνωστική και νευρολογική αναπηρία

Σ' αυτήν την κατηγορία ανήκει η δυσλειτουργία της οπτικής και ακουστικής αντίληψης, το σύνδρομο ελλειμματικής προσοχής, τα προβλήματα νοημοσύνης, τα προβλήματα μνήμης, οι διαταραχές που σχετίζονται με την ψυχική υγεία και οι επιληπτικές διαταραχές.

Τα άτομα με προβλήματα στην οπτική και ακουστική αντίληψη, που περιλαμβάνουν την δυσλεξία και την δυσαριθμησία, παρουσιάζουν δυσκολία στην επεξεργασία της γλώσσας και των αριθμών. Μπορεί να έχουν δυσκολία στην

επεξεργασία της προφορικής γλώσσας όταν την ακούν (προβλήματα που σχετίζονται με την ακουστική αντίληψη). Μπορεί επίσης να έχουν προβλήματα με τον προσανατολισμό.

Για να χρησιμοποιήσουν τον παγκόσμιο ιστό τα άτομα που παρουσιάζουν προβλήματα στη οπτική και ακουστική αντίληψη μπορεί να βασίζονται στην λήψη πληροφορίας μέσω διαφορετικών τρόπων αναπαράστασης της πληροφορίας ταυτόχρονα. Για παράδειγμα, κάποιος που έχει δυσκολία στην ανάγνωση μπορεί να χρησιμοποιεί screen reader για καλύτερη κατανόηση, ενώ κάποιος που παρουσιάζει προβλήματα στην ακουστική επεξεργασία μπορεί να χρησιμοποιεί υπότιτλους για να καταλάβει ένα ηχητικό κομμάτι.

Το κυριότερο πρόβλημα που αντιμετωπίζουν τα άτομα με οπτική και ακουστική ανικανότητα, χρησιμοποιώντας τον παγκόσμιο ιστό είναι το εξής:

- Έλλειψη εναλλακτικών στοιχείων για τις πληροφορίες που περιλαμβάνονται στις σελίδες του παγκόσμιου ιστού, όπως για παράδειγμα η έλλειψη εναλλακτικού κειμένου που μπορεί να μετατραπεί σε ήχο γι' αυτούς που παρουσιάζουν οπτικά προβλήματα ή η έλλειψη υπότιτλων.

Τα άτομα που πάσχουν από το σύνδρομο ελλειματικής προσοχής μπορεί να παρουσιάζουν δυσκολία στην συγκέντρωση.

Για να χρησιμοποιήσει τον παγκόσμιο ιστό, ένα άτομο με σύνδρομο ελλειματικής προσοχής μπορεί να χρειαστεί να απενεργοποιήσει τα κινούμενα σχέδια της ιστοσελίδας για να μπορέσει να συγκεντρωθεί καλύτερα στο περιεχόμενό της.

Τα προβλήματα που μπορεί να αντιμετωπίσουν τα άτομα που πάσχουν από σύνδρομο ελλειματικής προσοχής στον παγκόσμιο ιστό περιλαμβάνουν τα εξής:

- Απόσπαση της προσοχής από οπτικά και ηχητικά στοιχεία που δεν μπορούν εύκολα να απενεργοποιηθούν.
- Έλλειψη καθαρής και σταθερής οργάνωσης των ιστοσελίδων.

Τα άτομα με προβλήματα νοημοσύνης (μαθησιακές δυσκολίες) μπορεί να μαθαίνουν με πιο αργό ρυθμό ή να αντιμετωπίζουν δυσκολία στην κατανόηση

πολύπλοκων εννοιών. Το σύνδρομο Down είναι μια από τις αιτίες που προκαλούν τέτοιου είδους προβλήματα.

Για να χρησιμοποιήσουν το διαδίκτυο, τα άτομα με πνευματική αναπηρία μπορεί να χρειάζονται περισσότερο χρόνο σε μια ιστοσελίδα, μπορεί να βασίζονται περισσότερο στα γραφικά για να κατανοήσουν το περιεχόμενο της σελίδας και ίσως επωφεληθούν από την χρήση απλής γλώσσας στο κείμενο της ιστοσελίδας.

Τα προβλήματα που μπορεί να συναντήσουν είναι τα εξής:

- Χρήση υπερβολικά πολύπλοκης γλώσσας στο κείμενο της σελίδας.
- Ελλιπής χρήση γραφικών στη σελίδα.
- Ελλιπής οργάνωση του περιεχομένου της σελίδας.

Τα άτομα με προβλήματα μνήμης μπορεί να έχουν πρόβλημα με την βραχυπρόθεσμη μνήμη, με την μακροπρόθεσμη μνήμη ή μπορεί να έχουν κάποια έλλειψη ικανότητας στην ανάκληση της γνώσης σχετικά με την γλώσσα.

Για να χρησιμοποιήσουν το διαδίκτυο τα άτομα με προβλήματα μνήμης μπορεί να βασίζονται στην οργάνωση προσπέλασης μέσα στη σελίδα.

Το κυριότερο πρόβλημα που μπορεί να συναντήσουν τα άτομα αυτά στον παγκόσμιο ιστό είναι το εξής:

- Ελλιπής οργάνωση του περιεχομένου της σελίδας.

Τα άτομα με προβλήματα ψυχικής υγείας μπορεί να έχουν δυσκολία να συγκεντρωθούν στο περιεχόμενο της ιστοσελίδας, είτε λόγω θολής όρασης, είτε λόγω τρεμάμενων χεριών, που είναι αποτελέσματα των φαρμάκων που λαμβάνουν.

Για να χρησιμοποιήσουν το διαδίκτυο, τα άτομα με προβλήματα ψυχικής υγείας ίσως χρειαστεί να απενεργοποιήσουν τα οπτικά ή ηχητικά στοιχεία που μπορεί να τους αποσπάσουν την προσοχή ή να χρησιμοποιήσουν μεγενθυντές οθόνης.

Τα προβλήματα που μπορεί να συναντήσουν κατά την πρόσβαση στον παγκόσμιο ιστό είναι τα εξής:

- Οπτικό ή ηχητικό περιεχόμενο που μπορεί να αποσπάσει την προσοχή και δεν μπορεί να απενεργοποιηθεί εύκολα.
- Σελίδες με απόλυτο μέγεθος γραμματοσειράς που δεν μεγενθύνεται εύκολα.

Μερικά άτομα που πάσχουν από διαταραχές που μπορεί να προκαλέσουν κρίσεις, συμπεριλαμβανομένων μερικών τύπων επιληψίας, όπως η φωτοευαίσθητη επιληψία, επηρεάζονται από οπτικό τρεμόπαιγμα ή από ηχητικά σήματα συγκεκριμένων συχνοτήτων.

Για να χρησιμοποιήσουν το διαδίκτυο, τα άτομα με διαταραχές που προκαλούν κρίσεις, ίσως χρειαστεί να απενεργοποιήσουν τα κινούμενα σχέδια, κείμενο που αναβοσβήνει ή κάποιες συγκεκριμένες ηχητικές συχνότητες. Η αποφυγή των κινουμένων σχεδίων, του κείμενο που αναβοσβήνει και κάποιων ηχητικών συχνοτήτων μπορεί να αποτρέψει μια τέτοια κρίση.

Το κυριότερο πρόβλημα που μπορεί να συναντήσουν είναι το εξής:

- Σελίδες με οπτικοακουστικό περιεχόμενο σε συγκεκριμένες συχνότητες που μπορεί να προκαλέσει κρίσεις.

4.1.6. Πολλαπλές διαταραχές

Ο συνδυασμός περισσότερων της μιας διαταραχής μπορεί να μειώσει την ικανότητα ενός χρήστη στην χρήση προσβάσιμης πληροφορίας.

Για παράδειγμα, ενώ ένας τυφλός χρήστης μπορεί να επωφεληθεί από την ηχητική περιγραφή ενός βίντεο και κάποιος κουφός χρήστης μπορεί να επωφεληθεί από τους υπότιτλους που συνοδεύουν κάποιο ηχητικό μήνυμα, κάποιος που είναι και τυφλός και κουφός χρειάζεται ένα αντίγραφο του περιεχομένου του ηχητικού μηνύματος και του βίντεο, το οποίο θα μπορεί να προσπελάσει με ειδική συσκευή Braille.

Παρομοίως, κάποιος που είναι κουφός και παράλληλα έχει μειωμένη όραση μπορεί να επωφεληθεί από τους υπότιτλους που συνοδεύουν ένα ηχητικό μήνυμα

μόνο εαν οι υπότιτλοι μπορούν να μεγιστοποιηθούν και η αντίθεση χρώματος να ρυθμιστεί.

Κάποιος που δεν μπορεί να κουνήσει τα χέρια του και ταυτόχρονα δεν μπορεί να δει την οθόνη καλά, ίσως χρησιμοποιήσει έναν συνδυασμό εισαγωγής δεδομένων μέσω κάποιου προγράμματος αναγνώρισης φωνής και εξαγωγής πληροφοριών μέσω ήχου. Θα πρέπει όμως να βασιστεί σε ακριβείς δείκτες που θα παρέχουν πληροφορίες για το που βρίσκεται μέσα στη σελίδα και για τις επιλογές πλοήγησης που υπάρχουν μέσα στη σελίδα.

4.1.7. Συμπτώματα που σχετίζονται με την γήρανση

Οι αλλαγές στις ικανότητες ενός ατόμου λόγω γήρανσης μπορεί να περιλαμβάνουν αλλαγές στην δυνατότητα χρήσης μίας ικανότητας ή των συνδυασμό περισσότερων ικανοτήτων όπως όραση, ακοή, αντοχή και μνήμη. Τα προβλήματα που μπορεί να αντιμετωπίσουν κατά την χρήση του διαδικτύου μπορεί να περιλαμβάνουν όλα τα προαναφερόμενα. Κάθε ένα από αυτά τα προβλήματα μπορεί να επηρεάσει την ικανότητα του ατόμου να κάνει χρήση του περιεχομένου του παγκόσμιου ιστού. Ο συνδυασμός περισσότερων του ενός προβλημάτων καθιστά πιο δύσκολο τον τρόπο με τον οποίο παρακάμπτεται το κάθε πρόβλημα.

Για παράδειγμα, ένας χρήστης με μειωμένη όραση ίσως χρειαστεί να μεγεθύνει το περιεχόμενο μιας σελίδας, χάνοντας όμως έτσι οπτική πρόσβαση στις τριγύρω σχετικές πληροφορίες. Έτσι αυξάνεται η δυσκολία για έναν χρήστη, που έχει πρόβλημα βραχυπρόθεσμης μνήμης, στην πλοήγηση και χρήση της σελίδας.

4.2. Τυφλοί χρήστες

Προφανώς οι άνθρωποι που είναι τυφλοί δεν είναι δυνατόν να δούνε αυτά που βλέπουν οι υπόλοιποι άνθρωποι. Παρόλο που είναι αλήθεια ότι οι περισσότεροι τυφλοί χρήστες έχουν ένα βαθμό όρασης, μπορούμε να πούμε ότι δεν χρησιμοποιούν τα μάτια τους για να έχουν πρόσβαση στον παγκόσμιο ιστό καθώς

η όραση που έχουν δεν επαρκεί γι' αυτό το είδος εργασίας. Αυτό σημαίνει ότι η οθόνη του υπολογιστή και το ποντίκι είναι πολύ λιγότερο χρήσιμα για ένα άτομο που είναι τυφλό. Δεν φταίει ότι οι τυφλοί χρήστες είναι ανήκανοι να κουνήσουν ή να κάνουν κλικ με ένα ποντίκι αλλά ότι δεν ξέρουν πού να το κουνήσουν ή πού να κάνουν κλικ καθώς δεν μπορούν να δουν ότι υπάρχει στην οθόνη.

Οι βασικές αρχές προσβασιμότητας για τους τυφλούς χρήστες είναι οι εξής:

- Αντίληψη: καθώς δεν μπορούν ν' αντιληφθούν (να δουν) οπτικές πληροφορίες όπως γραφικά, διάταξη ή χρώματα.
- Λειτουργικότητα: καθώς συνήθως χρησιμοποιούν πληκτρολόγιο για να χειριστούν (πλοηγηθούν) το περιεχόμενο του παγκόσμιου ιστού.
- Κατανόηση: γιατί δεν μπορούν να καταλάβουν το περιεχόμενο που παρουσιάζεται σε μια μη-λογική γραμμική σειρά ή που περιέχει ειδικό κείμενο το οποίο δεν μπορεί να διαβαστεί λέξη προς λέξη ή χαρακτήρα προς χαρακτήρα (όπως μεγάλες διευθύνσεις διαδικτίου κτλ)
- Ευρωστία: επειδή η βοηθητική τεχνολογία που χρησιμοποιείται από τους τυφλούς χρήστες δεν είναι πάντα ικανή να έχει πρόσβαση σε ένα ευρύ φάσμα τεχνολογιών, ειδικά όταν αυτές οι τεχνολογίες είναι καινούργιες.

4.2.1. Screen readers

Οι τυφλοί χρήστες χρησιμοποιούν προγράμματα λογισμικού που μετατρέπουν το κείμενο σε ομιλία και έτσι είναι σε θέση να ακούσουν το περιεχόμενο του παγκόσμιου ιστού. Αυτά τα προγράμματα συνήθως ονομάζονται screen readers παρόλο που κάνουν περισσότερα πράγματα απ' το να διαβάζουν μια οθόνη. Οι screen readers επιτρέπουν στους χρήστες να πλοηγούνται μέσα στο περιεχόμενο του παγκόσμιου ιστού με πολλούς τρόπους. Ο χρήστης μπορεί να αφήσει απλά το screen reader να διαβάσει τα πάντα απ' την αρχή μέχρι το τέλος, διαβάζοντας μια γραμμή κάθε φορά, ή μπορεί να χρησιμοποιήσει το κουμπί tab για να πλοηγηθεί από σύνδεσμο σε σύνδεσμο. Οι χρήστες μπορούν επίσης να πλοηγηθούν από την μια επικεφαλίδα στην επόμενη (αν το περιεχόμενο του παγκόσμιου ιστού περιλαμβάνει επικεφαλίδες), από το ένα frame στο επόμενο (αν υπάρχουν frames)

ή με άλλους τρόπους. Οι πιο γνωστοί screen readers είναι οι JAWS, Window Eyes και Home Page Reader.

Οι screen readers μπορούν επίσης να χρησιμοποιηθούν από ανθρώπους που είναι και τυφλοί και κωφοί, αλλά αντί να μετατρέπουν το κείμενο σε φωνή, οι screen readers που είναι για κωφάλαλα άτομα μετατρέπουν το κείμενο σε braille χαρακτήρες μέσω refreshable braille συσκευών. Αυτές οι συσκευές έχουν μικρές ακίδες, οι οποίες μπορούν να ανεβοκατεβαίνουν ώστε να σχηματίζουν braille χαρακτήρες που τα κωφάλαλα άτομα μπορούν να αισθανθούν μέσω της αφής.

Είναι εκπληκτικό να σκεφτούμε το πόσες δυνατότητες δημιουργούνται μέσω αυτών των τεχνολογιών για τους τυφλούς χρήστες. Τα τυφλά άτομα δεν χρειάζεται να βασίζονται σε άλλα άτομα για να έχουν πρόσβαση σε κοινές πληροφορίες όπως εφημερίδες, περιοδικά, τραπεζικούς λογαριασμούς, σχολικές εγγραφές κ.α. Για παράδειγμα, παλιότερα, όταν οι τυφλοί χρήστες ήθελαν να γραφτούν σε κάποιο μάθημα στο πανεπιστήμιο έπρεπε να βασιστούν στην βοήθεια άλλων, που πραγματοποιούσαν αυτή τη δουλειά για χάρη τους. Έπρεπε να το κάνουν σύμφωνα με το πρόγραμμα των άλλων και να θυσιάσουν ως ένα σημείο την προστασία της ιδιωτικής τους ζωής. Τώρα, με την βοήθεια των screen readers, οι τυφλοί μαθητές μπορούν να χρησιμοποιήσουν το σύστημα εγγραφής στον παγκόσμιο ιστό που υπάρχει σε πολλά πανεπιστήμια. Αυτή η νέα δυνατότητα τους παρέχει ένα μεγαλύτερο επίπεδο ανεξαρτησίας που παλιότερα ήταν αδύνατο να υπάρξει.

Φυσικά, για να ισχύσει αυτό πρέπει όλο το περιεχόμενο των ιστοσελίδων να είναι προσβάσιμο από τις τεχνολογίες που οι τυφλοί χρήστες χρησιμοποιούν.

4.2.2. Περιορισμοί των screen readers

Τα προγράμματα screen readers έχουν αρκετές δυνατότητες, έχουν όμως και περιορισμούς. Δεν μπορούν να αντικατασταθούν πλήρως από την οπτική εμπειρία. Ευτυχώς, υπάρχουν τρόποι να αντισταθμιστεί η αδυναμία των screen readers.

Εικόνες: Τα προγράμματα screen readers δεν μπορούν να περιγράψουν τις εικόνες. Ο μοναδικός τρόπος για να μεταβιβαστεί το νόημα μιας εικόνας μέσω screen reader είναι διαβάζοντας κείμενο από το αρχείο που παρουσιάζεται σαν υποκατάστατο ή εναλλακτικό γι' αυτήν την εικόνα. Αν δεν υπάρχει εναλλακτικό κείμενο το πρόγραμμα screen reader δεν μπορεί να μεταδώσει το νόημα της εικόνας με ακρίβεια.

Οπτική διάταξη: Επίσης, οι screen readers δεν μπορούν να μεταδώσουν όλη τη σελίδα με τον τρόπο που την αντιλαμβάνεται ένας οπτικός χρήστης. Ένας οπτικός χρήστης μπορεί να κοιτάξει την σελίδα και να αντιληφθεί γρήγορα πώς είναι οργανωμένη και μετά να εστιάσει στο πιο σημαντικό περιεχόμενο. Ένα πρόγραμμα screen reader δεν είναι ικανό να το κάνει αυτό. Διαβάζει με γραμμικό τρόπο μια λέξη κάθε φορά. Δεν μπορεί πάντα να παραλείπει αποτελεσματικά εξωτερικό περιεχόμενο, όπως για παράδειγμα διαφημίσεις.

Πίνακες δεδομένων: Παρομοίως, ακριβώς επειδή οι screen readers πρέπει να διαβάζουν γραμμικά, οι πίνακες δεδομένων μπορεί να είναι αρκετά δυσνόητοι. Ας φανταστούμε πώς θα ήταν να προσπαθούμε να ακούσουμε και να καταλάβουμε ένα μεγάλο πίνακα περιεχομένων με δεκατέσσερις στήλες και εικοσιοχτώ γραμμές. Όταν θα διαβάζεται η εικοστή γραμμή, ο χρήστης θα έχει ξεχάσει προ πολλού ποιος ήταν ο τίτλος της όγδοης στήλης. Είναι πολύ δύσκολο αν όχι αδύνατο να ερμηνεύσουμε αυτήν την πληροφορία.

4.2.3. Προσβασιμότητα πληκτρολογίου

Το γεγονός ότι οι χρήστες που χρησιμοποιούν screens readers χρησιμοποιούν το πληκτρολόγιο ως το κύριο μέσο πλοήγησής τους στον παγκόσμιο ιστό είναι ένα σημείο στο οποίο οι προγραμματιστές πρέπει να δίνουν προσοχή. Χωρίς καν να συνειδητοποιούν τις συνέπειες, μερικοί προγραμματιστές διαδικτυακών εφαρμογών αναπτύσσουν τις σελίδες τους με τέτοιο τρόπο που είναι προσβάσιμες μόνο με χρήση ποντικιού. Τις περισσότερες φορές η έλλειψη προσβασιμότητας από το πληκτρολόγιο είναι αποτέλεσμα Javascript γεγονότων που σχετίζονται είτε με το κλικ είτε με την κίνηση του ποντικιού. Για παράδειγμα, οποιοδήποτε γεγονός

προκαλείται από το onmouseover γεγονός θα λειτουργήσει μόνο όταν ο χρήστης μετακινήσει το ποντίκι του πάνω από το συγκεκριμένο αντικείμενο. Δεν υπάρχει τρόπος να έχουμε πρόσβαση σ' αυτό το είδος περιεχομένου χρησιμοποιώντας μόνο πληκτρολόγιο χωρίς να παρέχουμε κάποια άλλη μέθοδο, είτε επιπρόσθετη, είτε αντί του onmouseover γεγονότος.

Σε μερικές περιπτώσεις δεν υπάρχει λόγος να παρέχουμε εναλλακτική λύση, όπως λόγου χάριν όταν το γεγονός onmouseover προκαλεί μια αισθητική αλλαγή σε μια εικόνα (για παράδειγμα όταν αλλάζουν απλά τα χρώματά της). Κάποιος που είναι τυφλός πιθανόν να μην ενδιαφέρεται για μια τέτοια αλλαγή. Αυτή η πληροφορία είναι άσχετη.

Παρόλ' αυτά, αν νέες πληροφορίες εμφανίζονται με το onmouseover γεγονός, όπως για παράδειγμα η ξαφνική εμφάνιση ενός μενού συστήματος, τότε είναι σημαντικό να βρούμε μια μέθοδο που να κάνει το περιεχόμενο προσβάσιμο από το πληκτρολόγιο.

4.2.4. Βασικές έννοιες

Ας έχουμε στο μυαλό ότι οι τυφλοί έχουν πρόσβαση στον παγκόσμιο ιστό χρησιμοποιώντας screen readers και πληκτρολόγια, και ότι υπάρχουν μερικά μεγάλα εμπόδια που αντιμετωπίζουν όταν προσπαθούν να έχουν πρόσβαση σε περιεχόμενο του παγκόσμιου ιστού. Μια προσβάσιμη ιστοσελίδα σημαίνει ελευθερία και ανεξαρτησία για κάποιον που είναι τυφλός. Όταν οι προγραμματιστές δεν παρέχουν προσβάσιμο περιεχόμενο, ο μοναδικός τρόπος με τον οποίο ένας τυφλός μπορεί να πάρει πληροφορίες από τον παγκόσμιο ιστό είναι μέσω κάποιου άλλου.

Προκλήσεις	Λύσεις
Οι τυφλοί χρήστες δεν χρησιμοποιούν ποντίκι	Δεν πρέπει να γράφουμε scripts που απαιτούν την χρήση ποντικιού. Πρέπει να

	παρέχουμε εναλλακτικές μέσω πληκτρολογίου.
Οι εικόνες, οι φωτογραφίες και τα γραφικά είναι άχρηστα	Πρέπει να παρέχουμε περιγραφές μέσω κειμένου, είτε μέσω του alt, είτε, αν είναι απαραίτητο, μέσω μεγαλύτερων περιγραφών (μέσα στην ίδια σελίδα ή μέσα σε άλλη σελίδα).
Οι χρήστες συνήθως ακούνε το περιεχόμενο των σελίδων χρησιμοποιώντας screen readers	Πρέπει να αφήνουμε τους χρήστες να παραλείπουν μενού πλοήγησης, μεγάλες λίστες και άλλα στοιχεία που μπορεί να είναι δύσκολα ή κουραστικά στην ακρόαση.
Οι χρήστες συχνά μεταβαίνουν από υπερσύνδεσμο σε υπερσύνδεσμο μέσω του πλήκτρου tab	Πρέπει να είμαστε σίγουροι ότι οι υπερσύνδεσμοι έχουν όνομα σχετικό με το περιεχόμενό τους (Το “κάνε κλικ εδώ” είναι προβληματικό).
Οι τυφλοί χρήστες δεν μπορούν να “δουν” όλα τα frames ταυτόχρονα. Μπορούν να επισκεφτούν μόνο ένα κάθε φορά, πράγμα το οποίο μπορεί να οδηγήσει σε αποπροσανατολισμό	Δεν πρέπει να χρησιμοποιούμε frames με ευκολία εκτός και αν πρέπει. Όταν τα χρησιμοποιούμε πρέπει να έχουμε τίτλους που να σχετίζονται με το σκοπό τους (για παράδειγμα “frame πλοήγησης”, “κυρίως περιεχόμενο” κτλ).
Είναι δύσκολο γι’ αυτούς τους χρήστες να ξέρουν που βρίσκονται όταν ακούνε τα περιεχόμενα ενός πίνακα	Πρέπει να παρέχουμε κεφαλίδες στις γραμμές και στις στήλες (<th>). Πρέπει να ξέρουμε ότι οι πίνακες, ειδικά αυτοί που περιέχουν συγχωνευμένα κελιά, έχουν νόημα όταν διαβάζονται γραμμή γραμμή από τ’ αριστερά προς τα δεξιά.
Σύνθετοι πίνακες και γραφήματα που συνήθως ερμηνεύονται οπτικά είναι άχρηστοι	Πρέπει να παρέχουμε περιλήψεις και/ή περιγραφές κειμένου.
Δεν υποστηρίζουν όλοι οι screen	Πρέπει να παρέχουμε επιπλέον

readers image maps	υπερσυνδέσμους κειμένου για τα σημεία που μας ενδιαφέρουν να εξηγήσουμε στα image maps
Τα χρώματα είναι άχρηστα	Δεν πρέπει να βασιζόμαστε αποκλειστικά στο χρώμα για να εξάγουμε νόημα
Οι χρήστες περιμένουν οι υπερσύνδεσμοι να τους μεταφέρουν κάπου	Δεν πρέπει να γράφουμε scripts σε υπερσυνδέσμους που δεν έχουν κάποιον πραγματικό προορισμό που να συσχετίζεται μ' αυτά (για παράδειγμα href="javascript: function(this)").

Πίνακας 3 - Βασικές έννοιες που αφορούν την τύφλωση

5. Εφαρμογή

Στο κεφάλαιο αυτό θ' αναφερθούμε σε μια εφαρμογή που έγινε στα πλαίσια αυτής της πτυχιακής σαν ένα παράδειγμα όσων αναφέραμε στα προηγούμενα κεφάλαια. Αυτή η εφαρμογή είναι σχεδιασμένη με το GWT και αφορά ένα σενάριο δημιουργίας λογαριασμού mail από έναν τυφλό χρήστη.

Για την δημιουργία της εφαρμογής βασίστηκα σε ένα video της NFB (National Federation of the Blind), δηλαδή στα ελληνικά εθνική ομοσπονδία για τους τυφλούς. Αυτή η ομοσπονδία είναι η μεγαλύτερη και παράλληλα αυτή που έχει την μεγαλύτερη επιρροή σε θέματα που αφορούν τους τυφλούς στις Ηνωμένες Πολιτείες της Αμερικής. Η NFB έχει σαν στόχο την βελτίωση της ζωής των τυφλών ατόμων μέσω εκπαίδευσης, ερευνών, τεχνολογιών και προγραμμάτων που σαν απώτερο σκοπό έχουν να τονώσουν την αυτοπεποίθηση και την ανεξαρτησία αυτών των ατόμων. Είναι πρωτοπόρος ομοσπονδία στον τομέα των τυφλών σήμερα και θεωρείται εκπρόσωπος των τυφλών στις Ηνωμένες Πολιτείες της Αμερικής. Τον Ιανουάριο του 2004 η NFB άνοιξε το Federation of the Blind Jernigan Institute που είναι το πρώτο ερευνητικό και εκπαιδευτικό κέντρο των Ηνωμένων Πολιτειών για τους τυφλούς το οποίο ηγούνται τυφλοί.

Στο βίντεο στο οποίο βασίστηκα αναφέρονται κάποια προβλήματα που καθιστούν δύσκολη την δημιουργία λογαριασμού mail στο gmail από έναν τυφλό χρήστη, ενώ παράλληλα προτείνονται και κάποιες λύσεις και βελτιώσεις σ' αυτά τα προβλήματα. Αυτός ακριβώς είναι και ο στόχος αυτής της εφαρμογής, δηλαδή η δημιουργία λύσεων στα προβλήματα που αντιμετωπίζουν οι τυφλοί χρήστες κατά την δημιουργία ενός λογαριασμού mail, χρησιμοποιώντας τεχνολογίες που αναφέραμε στα προηγούμενα κεφάλαια και συγκεκριμένα χρησιμοποιώντας το GWT. Το link για το video αυτό είναι το ακόλουθο:

<http://www.nfb.org/images/nfb/video/google/gmailsignup.html>

5.1. Τα προβλήματα που παρουσιάζονται

Η NFB έχει δημιουργήσει μια σειρά από video που σχεδιάστηκαν σαν ένα παράδειγμα έλλειψης προσβασιμότητας κατά την χρήση των Google apps και του gmail. Το συγκεκριμένο βίντεο εστιάζει στην διαδικασία δημιουργίας λογαριασμού στο gmail και στα όρια προσβασιμότητας που θέτονται για τους τυφλούς χρήστες χρησιμοποιώντας τεχνολογίες πρόσβασης που δεν είναι οπτικές, όπως για παράδειγμα χρησιμοποιώντας screen readers.

Για την καλύτερη κατανόηση αυτού του σεναρίου ακολουθεί μια εικόνα όπου φαίνονται τα στοιχεία που πρέπει να εισάγει ο χρήστης.

Get started with Gmail

First name:

Last name:

Desired Login Name: @gmail.com
Examples: JSmith, John.Smith

Εικόνα 7 Κουμπί check availability

Αρχικά ο χρήστης πρέπει να εισάγει το όνομα, το επίθετο και το επιθυμητό όνομα σύνδεσης στα αντίστοιχα πεδία και μετά να πατήσει το κουμπί check availability. Αν ο χρήστης πληκτρολογήσει σαν επιθυμητό όνομα σύνδεσης κάτι το οποίο να περιέχει κάποιον ειδικό χαρακτήρα, όπως για παράδειγμα η κάτω παύλα (_) και μετά πατήσει το κουμπί “check availability”, τότε εμφανίζεται ένα μήνυμα λάθους που ενημερώνει τον χρήστη ότι αυτός ο χαρακτήρας δεν επιτρέπεται (Εικόνα 8). Το μήνυμα αυτό είναι ορατό αλλά δεν διαβάζεται αυτόματα από το πρόγραμμα ανάγνωσης οθόνης, οπότε ο τυφλός χρήστης δεν μπορεί να το αντιληφθεί. Αυτό θα συμβεί μόνο αν ο χρήστης μετακινηθεί πάνω του χρησιμοποιώντας τα αντίστοιχα πλήκτρα.

Desired Login Name: @gmail.com
Examples: JSmith, John.Smith

Sorry, only letters (a-z), numbers (0-9), and periods (.) are allowed.

Εικόνα 8 Κουμπί check availability και μήνυμα λάθους

Κάτι αντίστοιχο θα συμβεί και αν εισάγουμε ένα σωστό επιθυμητό όνομα σύνδεσης, όπως για παράδειγμα αν αντικαταστήσουμε την κάτω παύλα με τελεία, και στην συνέχεια πατήσουμε το κουμπί “check availability”. Σ’ αυτή τη περίπτωση εμφανίζεται ένα μήνυμα που ενημερώνει τον χρήστη ότι αυτό το όνομα σύνδεσης είναι διαθέσιμο (Εικόνα 9). Το μήνυμα είναι ορατό αλλά δεν διαβάζεται αυτόματα από το πρόγραμμα screen reader. Επίσης, σ’ αυτήν την περίπτωση ακόμα και ο χρήστης να μετακινηθεί προς τα κάτω, το μήνυμα αυτό δεν θα διαβαστεί από το screen reader.

Desired Login Name: @gmail.com
Examples: JSmith, John.Smith

paschalia.nivorli is available

Εικόνα 9 Κουμπί check availability και is available μήνυμα

Όπως ήδη αναφέραμε στο προηγούμενο κεφάλαιο, υπερσύνδεσμοι με όνομα “κάνε κλικ εδώ” είναι προβληματικοί. Οι υπερσύνδεσμοι θα πρέπει να έχουν όνομα σχετικό με το περιεχόμενό τους. Κατά την διαδικασία εγγραφής στο gmail υπάρχουν αρκετοί υπερσύνδεσμοι με όνομα “Learn more” και έτσι ο τυφλός χρήστης δεν γνωρίζει που θα μετακινηθεί αν πατήσει πάνω τους. Η λύση σ’ αυτό το πρόβλημα είναι στο όνομά τους να περιέχουν στοιχεία για το που πρόκειται να μεταφερθεί ο χρήστης αν πατήσει σε αυτούς.

Ένα ακόμα πρόβλημα που αναφέρεται στο βίντεο είναι η δυσκολία ακουστικής αντίληψης για τους τυφλούς χρήστες στο captcha πεδίο. Το captcha είναι ένας μηχανισμός που χρησιμοποιείται για να διαχωρίσει τους ανθρώπους από τα bots.

Αυτό συνήθως γίνεται με την εισαγωγή κάποιων χαρακτήρων, που βρίσκονται σε μια εικόνα, σε ένα πεδίο κειμένου και αντίστοιχο server side έλεγχο. Εναλλακτικά αυτό μπορεί να γίνει και με την πληκτρολόγηση των χαρακτήρων ή αριθμών από ηχητικό μήνυμα. Κατά την αναπαραγωγή του ήχου ακούγεται πολύς θόρυβος και παρεμβολές δυσκολεύοντας έτσι την σαφή κατανόηση του μηνύματος. Η google στην οποία ανήκει στο reCaptcha έχει βελτιώσει πολύ αυτό το μηχανισμό.

Επίσης, αν η σελίδα περιέχει κάποιο λάθος, αφού ο χρήστης συμπληρώσει όλα τα στοιχεία και πατήσει το κουμπί “I accept. Create my account.” το πρόγραμμα screen reader ξεκινάει να διαβάσει σειριακά ότι υπάρχει στην οθόνη χωρίς να μεταφέρει ή να διαβάζει απ’ ευθείας το σημείο στο οποίο υπάρχει το λάθος. Αν ο χρήστης υποθέσει ότι υπάρχει κάποιο λάθος και προσπαθήσει να το βρει μέσω γρήγορης πλοήγησης, όπως για παράδειγμα πλοήγηση μεταξύ κεφαλίδων, θα αντιληφθεί ότι στην σελίδα υπάρχει μόνο μια κεφαλίδα με όνομα “Create an account”. Σ’ αυτή την περίπτωση θα έπρεπε να υπάρχει κάποιος τρόπος ώστε ο χρήστης να μεταφέρεται απ’ ευθείας ή μέσω γρήγορης πλοήγησης στα σημεία που παρουσιάζεται το εκάστοτε πρόβλημα.

5.2. Οι δοθείσες λύσεις

Στόχος της εφαρμογής ήταν να δώσει λύσεις και βελτιώσεις στα προβλήματα που μόλις περιγράψαμε. Όσον αφορά τα μηνύματα που εμφανίζονται στον χρήστη όταν επιλέγει το κουμπί “check availability” η πρώτη μου σκέψη ήταν να εμφανίζεται το αντίστοιχο μήνυμα και στην συνέχεια να μεταφέρεται το focus πάνω του ώστε να διαβάζεται απ’ το screen reader. Αρχικά είχα τα μηνύματα αυτά σε label αλλά στην πορεία είδα ότι τα labels δεν έχουν μέθοδο focus. Έτσι στην συνέχεια έψαξα και βρήκα ποιά widgets έχουν focus μέθοδο και έτσι κατέλειξα να εμφανίζω αυτά τα μηνύματα σε RichTextArea, που διαθέτει μέθοδο focus.

Στην συνέχεια έπρεπε να βρω ένα δυναμικό τρόπο εμφάνισης αυτών των μνημάτων. Η λύση σ’ αυτό το πρόβλημα δόθηκε μέσω του FlexTable που είναι ένα widget του GWT. Το συγκεκριμένο widget είναι ένας πίνακας που επιτρέπει την δυναμική προσθήκη και αφαίρεση γραμμών και κελιών μέσω των μεθόδων

`insertRow(int beforeRow), insertCell(int beforeRow,int beforeColumn) ,
removeRow(int row) και removeCell(int row, int col) αντίστοιχα.`

Στην συνέχεια φαίνεται ένα απόσπασμα του κώδικα για το κουμπί “check availability”.

```
checkAvailabilityButton = new Button("New button");
checkAvailabilityButton.addClickListener(new ClickHandler() {
    RichTextArea availableRichTextArea = new RichTextArea();
    public void onClick(ClickEvent checkAvailabilityEvent) {
        if(!desiredLoginNameTextBox.getText().matches("^([a-zA-Z0-9.]+)$")){
            availableRichTextArea.setText("Sorry, only letters  
(a-z), numbers (0-9) and periods (.) are allowed");
            flexTable.insertRow(5);
            flexTable.setWidget(5, 1, availableRichTextArea);
            availableRichTextArea.setFocus(true);
        }
        else{
            if(!desiredLoginNameTextBox.getText().equals("lina")){
                availableRichTextArea.setText(desiredLoginNameTextBox.getText  
()+" is available" );
                flexTable.insertRow(5);
                flexTable.setWidget(5, 1, availableRichTextArea);
                availableRichTextArea.setFocus(true);
            }
            else {
                availableRichTextArea.setText(desiredLoginNameTextBox.ge  
tText()+" is not available" );
                flexTable.insertRow(5);
                flexTable.setWidget(5, 1, availableRichTextArea);
                availableRichTextArea.setFocus(true);
            }
        }
    }
}
```



```
});
```

Στο πρώτο if ελέγχουμε αν οι χαρακτήρες του textbox ανήκουν στο επιτρεπτό σύνολο χαρακτήρων. Αυτό γίνεται μέσω της μεθόδου matches των String και της κανονικής έκφρασης `^[a-zA-Z0-9.]+$`. Ακολουθεί το else που αποτελείται από δυο σκέλη, το if και το else. Στο if ελέγχουμε αν το δοθέν επιθυμητό όνομα σύνδεσης είναι διαθέσιμο και στο else μένουν οι περιπτώσεις που τα επιθυμητά ονόματα δεν είναι διαθέσιμα. Σε όλες τις περιπτώσεις αλλάζει το περιεχόμενο του RichTextArea, προστείνεται μια νέα σειρά στον πίνακα, είτε πάνω είτε κάτω απ' το κουμπί, μπαίνει εκεί το RichTextArea και στην συνέχεια μεταφέρεται το focus εκεί, ώστε να μπορεί ο screen reader να διαβάσει το αντίστοιχο μήνυμα.

Όσον αφορά το πρόβλημα με τους υπερσυνδέσμους απλά τους δόθηκε ονομασία που να σχετίζεται με την τοποθεσία στην οποία οδηγούν και το περιεχόμενο της.

Τέλος, στην περίπτωση του κουμπιού "I accept. Create my account." δόθηκε μια λύση παρόμοια με αυτή που δόθηκε στο κουμπί "check availability". Αν δεν υπάρχει σωστή αντιστοιχία ανάμεσα στα δύο πεδία επιθυμητού κωδικού εμφανίζεται ένα μήνυμα που λέει "re-enter password" και στην συνέχεια μεταφέρεται το focus εκεί ώστε να διαβαστεί το μήνυμα απ' το screen reader. Στην συνέχεια ακολουθεί ένα απόσπασμα κώδικα που αναφέρεται σ' αυτό το κουμπί.

```
Button createAccountButton = new Button("Create account!");
createAccountButton.setOnClickListener(new ClickHandler() {
    public void onClick(ClickEvent createAccountEvent) {
        RichTextArea passwordMismatch = new RichTextArea();
        if (!(choosePasswordTextBox.getText().equals(reEnterTextB
ox.getText()))){
            passwordMismatch.setText("Re-enter password");
            flexTable.insertRow(8);
            flexTable.setWidget(8, 1, passwordMismatch);
            choosePasswordTextBox.setFocus(true);
        }
        else
            Window.alert("Your account has been created!");
    }
});
```

}) ;

6. Συμπεράσματα

Όπως έχουμε ήδη αναφέρει είναι πολύ σημαντικό ο παγκόσμιος ιστός να είναι προσβάσιμος από όλους. Δηλαδή όλα τα είδη χρηστών πρέπει να μπορούν να πλοηγηθούν στις σελίδες του, να καταλάβουν το περιεχόμενό τους και να αλληλεπιδράσουν μ' αυτές, ανεξάρτητα από το αν διαθέτουν κάποιο είδος αναπηρίας ή όχι. Πρέπει δηλαδή να σχεδιάζουμε φιλικές προς τον χρήστη σελίδες, το οποίο επιτυγχάνεται μέσω φιλικών user interface. Για να πετύχουμε φιλικά προς τον χρήστη user interface πρέπει να έχουμε καλή γνώση των βοηθητικών τεχνολογιών και συσκευών που χρησιμοποιούν οι χρήστες με αναπηρίες για να έχουν πρόσβαση στον παγκόσμιο ιστό, ενώ παράλληλα πρέπει να χρησιμοποιήσουμε front end τεχνολογίες για τον σχεδιασμό και την ανάπτυξη τέτοιων εφαρμογών.

Αυτός ακριβώς ήταν και ο στόχος αυτής της πτυχιακής, δηλαδή οι τρόποι με τους οποίους μπορούμε να δημιουργήσουμε φιλικά προς τον χρήστη user interfaces. Αναφερθήκαμε στους τρόπους με τους οποίους μπορούμε να βελτιώσουμε την προσβασιμότητα μέσω HTML και Javascript, ενώ παράλληλα αναφερθήκαμε και σε συνήθη λάθη που κάνουν οι προγραμματιστές ώστε αυτά τα αποφεύγονται.

Αναφερθήκαμε σε δύο μεγάλες κατηγορίες frameworks. Στην πρώτη κατηγορία ανήκουν αυτά τα οποία απαιτούν έστω και βασική γνώση HTML και Javascript και μας παρέχουν κάποιες διευκολύνσεις στην γραφή κώδικα είτε μέσω βιβλιοθηκών και έτοιμων κομματιών κώδικα είτε μέσω εύκολης πρόσβασης και τροποποίησης των HTML στοιχείων. Στην δεύτερη κατηγορία ανήκουν αυτά τα οποία δεν απαιτούν την γνώση HTML και Javascript. Η γραφή κώδικα μέσω αυτών των frameworks γίνεται σε κάποια γλώσσα υψηλού επιπέδου και υπάρχουν μεταγλωτιστές που μεταγλωττίζουν από αυτή τη γλώσσα σε HTML και Javascript.

Τέλος αναφερθήκαμε σε κατηγορίες αναπηρίας και στις τεχνολογίες που χρησιμοποιούν τα άτομα αυτά για να έχουν πρόσβαση στον παγκόσμιο ιστό, καθώς και στα προβλήματα που μπορεί να αντιμετωπίσουν κατά την πλοήγηση σε μια σελίδα που δεν είναι προσβάσιμη, ενώ δόθηκε ιδιαίτερη έμφαση στους τυφλούς χρήστες.

Σαν μία υλοποίηση όσων αναφέραμε πιο μπροστά, δημιουργήθηκε μια εφαρμογή όπου δόθηκαν λύσεις και βελτιώσεις στα προβλήματα που αντιμετωπίζουν οι τυφλοί χρήστες κατά την εγγραφή τους στο gmail. Ο σχεδιασμός αυτής της εφαρμογής έγινε με το GWT. Ένα από τα προβλήματα που αντιμετώπισα κατά την δημιουργία της εφαρμογής ήταν η εισαγωγή του captcha πεδίου καθότι αυτό δεν είναι δυνατό μέσω Java. Γι'αυτό το λόγω ένα μέρος της εφαρμογής έχει γραφεί χρησιμοποιώντας στατική HTML.

Σαν συμπέρασμα αυτού του προβλήματος που προέκυψε μπορούμε να πούμε ότι η επιλογή του framework για τον σχεδιασμό και την ανάπτυξη μιας εφαρμογής είναι επιλογή του προγραμματιστή σύμφωνα με τις γνώσεις του αλλά πρέπει να έχουμε στο μυαλό μας ότι μπορεί να υπάρξουν περιορισμοί.

Βιβλιογραφία

Websites

<http://en.wikipedia.org/wiki/Internet>

<http://en.wikipedia.org/wiki/Www>

<http://www.w3.org/WAI/intro/accessibility.php>

<http://www.w3.org/WAI/EO/Drafts/PWD-Use-Web/>

<http://www.guardian.co.uk/help/insideguardian/2009/sep/28/blogpost>

<http://www.w3.org/WAI/WCAG20/quickref/Overview.php>

http://www.w3schools.com/html/html_xhtml.asp

<http://reference.sitepoint.com/html/accessibility>

<http://webaim.org/techniques/javascript/>

<http://webaim.org/techniques/javascript/summary>

http://www.w3.org/wiki/CSS/Training/What_is_CSS

http://www.w3schools.com/css/css_howto.asp

<http://www.w3.org/TR/CSS-access>

http://www.w3schools.com/jquery/jquery_intro.asp

http://www.w3schools.com/jquery/jquery_syntax.asp

http://www.w3schools.com/jquery/jquery_selectors.asp

http://www.authenticociety.com/blog/jQueryPluginTutorial_Beginner

<http://jqueryui.com/demos/>

<http://easyframework.com/documentation.php>

Πτυχιακή εργασία της φοιτήτριας Νιβορλή Πασχαλιάς

<http://code.google.com/intl/el-GR/webtoolkit/overview.html>

<http://code.google.com/intl/el-GR/webtoolkit/doc/latest/tutorial/create.html>

http://en.wikipedia.org/wiki/Quirks_mode

<http://code.google.com/intl/el-GR/webtoolkit/doc/latest/RefWidgetGallery.html>

<http://code.google.com/intl/el-GR/webtoolkit/doc/latest/tutorial/buildui.html>

<http://code.google.com/intl/el-GR/webtoolkit/doc/latest/tutorial/style.html>

<http://www.w3.org/WAI/intro/aria>

<http://code.google.com/intl/el-GR/webtoolkit/doc/latest/DevGuideA11y.html>

<http://pyjs.org/>

<http://pyjs.org/#Overview>

<http://pyjs.org/#FAQ>

<http://www.morfikwiki.com/wiki3/Introduction>

http://www.morfikwiki.com/wiki3/Morfik_Basic_Overview

<http://www.w3.org/WAI/EO/Drafts/PWD-Use-Web/>

<http://webaim.org/articles/visual/blind>

Βιβλία

Duckett , J. (2010), Beginning HTML, XHTML, CSS, and JavaScript

Paciello, M. G. (2000), Web accessibility for people with disabilities