

ΤΕΧΝΟΛΟΓΙΚΟ ΕΚΠΑΙΔΕΥΤΙΚΟ ΙΔΡΥΜΑ ΔΥΤΙΚΗΣ ΕΛΛΑΔΑΣ
ΣΧΟΛΗ ΤΕΧΝΟΛΟΓΙΚΩΝ ΕΦΑΡΜΟΓΩΝ
ΤΜΗΜΑ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ Τ.Ε.

ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ 1685

Απομακρυσμένος έλεγχος ψηφιακής φωτογραφικής μηχανής (DSLR) μέσω του Διαδικτύου

ΣΠΟΥΔΑΣΤΗΣ: ΜΩΚΟΣ ΣΩΤΗΡΙΟΣ (6107)

ΕΠΟΠΤΕΥΩΝ ΚΑΘΗΓΗΤΗΣ: ΤΟΠΑΛΗΣ ΕΥΑΓΓΕΛΟΣ

ΠΑΤΡΑ 2018

Ευχαριστίες

Θα ήθελα να ευχαριστήσω τον εισηγητή της πτυχιακής εργασίας μου, Δρ. Ευάγγελο Τοπάλη (έκτακτο εκπαιδευτικό προσωπικό), για την βοήθεια και καθοδήγηση σε όλη την διάρκεια της εκπόνησης της εργασίας μου. Επίσης, ευχαριστώ τον Καθηγητή κ. Λουκά Χαδέλλη που μου επέτρεψε να κάνω χρήση του χώρου του εργαστηρίου Μικροϋπολογιστικών Συστημάτων για την εκπόνηση αυτής της εργασίας και για τις υποδείξεις του. Τέλος, θα ήθελα να ευχαριστήσω την οικογένεια μου για την υποστήριξη και την υπομονή τους όλα αυτά τα χρόνια.

ΠΕΡΙΛΗΨΗ

Σκοπός αυτής της πτυχιακής εργασίας είναι ο απομακρυσμένος έλεγχος των λειτουργιών μίας επαγγελματικής ψηφιακής φωτογραφικής μηχανής μέσω του Διαδικτύου. Ο έλεγχος αυτός θα γίνει με την δημιουργία μίας δυναμικής ιστοσελίδας, η οποία θα είναι προσβάσιμη με την βοήθεια της εφαρμογής gphoto-webui που τρέχει σε ένα Raspberry Pi.

Το Raspberry Pi είναι ένας ηλεκτρονικός υπολογιστής, ο οποίος έχει ως λειτουργικό σύστημα το ελεύθερο λογισμικό Linux, το οποίο με την χρήση της ελεύθερης βιβλιοθήκης λογισμικού libgphoto2 και του πελάτη γραμμής εντολών gphoto2, θα επικοινωνεί με την μηχανή για να αποστείλει τις εντολές που επιθυμεί ο χρήστης. Η φωτογραφική μηχανή θα είναι διασυνδεδεμένη με το Raspberry Pi μέσω μιας θύρας τύπου USB. Αυτή η διάταξη επιλέχθηκε επειδή είναι φορητή, ελαφριά και μικρή σε όγκο. Τα προαναφερόμενα πακέτα λογισμικού δίνουν στον χρήστη την δυνατότητα εκτός από το να πραγματοποιήσει την λήψη φωτογραφιών και την αλλαγή διαφόρων ρυθμίσεων της μηχανής όπως την ταχύτητα του κλείστρου, το άνοιγμα του διαφράγματος, την ευαισθησία ISO, την ποιότητα και ανάλυση των φωτογραφιών, την ισορροπία του λευκού, την αντιστάθμιση έκθεσης, την λειτουργία του φλας (ανάλογα πάντα με τις δυνατότητες της μηχανής που είναι συνδεδεμένη με τον υπολογιστή).

Η ιστοσελίδα αυτή θα μπορεί να είναι προσβάσιμη από όλες ηλεκτρονικές συσκευές έχουν δυνατότητα σύνδεσης στον Παγκόσμιο Ιστό μέσω ενός προγράμματος φυλλομετρητή. Για την κατασκευή αυτής της ιστοσελίδας θα χρησιμοποιηθεί η γλώσσα σήμανσης HTML, η γλώσσα φύλλων στυλ CSS για τον έλεγχο της εμφάνισης της ιστοσελίδας και η γλώσσα προγραμματισμού PHP για την αποστολή των εντολών γραμμής για τον χειρισμό της φωτογραφικής μηχανής.

Περιεχόμενα

Κεφάλαιο 1: Εισαγωγή	5
1.1 Περίληψη	5
1.2 Ιστορική αναδρομή στις ψηφιακές φωτογραφικές μηχανές	5
1.3 Υπολογιστές μονής πλακέτας	6
1.3.1 Η σειρά υπολογιστών Raspberry Pi.....	6
1.3.2 Raspberry Pi 3 Model B.....	7
1.4 Λογισμικό	8
1.4.1 gPhoto2.....	8
1.4.2. gphoto-webui.....	8
1.4.3 Γλώσσα σήμανσης HTML (HyperText Markup Language).....	8
1.4.4 Η γλώσσα φύλλων στυλ CSS (Cascading Style Sheets)	9
1.4.5 Η γλώσσα προγραμματισμού JavaScript.....	9
1.4.6 Η γλώσσα προγραμματισμού PHP	9
Κεφάλαιο 2: Υλοποίηση του συστήματος απομακρισμένου ελέγχου	11
2.1: Εγκατάσταση του απαραίτητου υλισμικού και λογισμικού στο Raspberry Pi	11
2.1.1: Επιλογή του κατάλληλου μέσου αποθήκευσης	11
2.1.2: Επιλογή του κατάλληλου λειτουργικού συστήματος και εγκατάστασης του ...	11
2.1.3: Εγκατάσταση απαραίτητων πακέτων λογισμικού και βιβλιοθηκών στο Raspbian.....	12
2.1.4: Δοκιμή της σωστής λειτουργίας των πακέτων λογισμικού.....	12
2.2: Κατασκευή της δυναμικής ιστοσελίδας χρησιμοποιώντας ως βάση το gPhoto2 και το gphoto-webui	13
2.2.1: Δημιουργία του κουμπιού κατεύθυνσης προς την σελίδα αλλαγής των ρυθμίσεων της φωτογραφικής μηχανής.....	13
2.2.2: Δημιουργία της βασικής δομής της ιστοσελίδας αλλαγής των ρυθμίσεων	14
2.2.3: Προσθήκη των μεταδεδομένων της νέας ιστοσελίδας	15
2.2.4: Ολοκλήρωση της ανακατεύθυνσης και δημιουργία της κεφαλίδας (header) στην νέα σελίδα	16
2.2.5: Δημιουργία πίνακα πληροφοριών	18
2.2.6: Προγραμματισμός του τρόπου ανάκτησης πληροφοριών από την φωτογραφική μηχανή	21
2.2.7: Δημιουργία ενός βασικού πτυσσόμενου μενού	30
2.2.8: Προγραμματισμός των εντολών απομακρυσμένης αλλαγής ρυθμίσεων της κάμερας	31

2.2.9: Συμπλήρωση των πτυσσόμενων μενού με τις εντολές απομακρυσμένου ελέγχου της φωτογραφικής μηχανής	38
2.2.10: Δημιουργία σελίδας σύνδεσης στην ιστοσελίδα	51
Κεφάλαιο 3: Συμπεράσματα και μελλοντικές επεκτάσεις	54
Παράρτημα	55
Login.php.....	55
Logout.php	56
Index.php.....	56
Index.html.....	56
Settings.html.....	59
Service.php.....	67
Controller.js	90

Κεφάλαιο 1: Εισαγωγή

1.1 Περίληψη

Το αντικείμενο της παρούσας πτυχιακής εργασίας είναι ο απομακρυσμένος έλεγχος των λειτουργιών μιας ψηφιακής μονοοπτικής φωτογραφικής μηχανής (digital single reflex camera) μέσω του διαδικτύου. Για την επίτευξη αυτού του στόχου θα δημιουργηθεί μια δυναμική ιστοσελίδα που θα τρέχει σε ένα υπολογιστή μονής πλακέτας Raspberry Pi.

Πρωτίστως θα γίνει μία σύντομη ιστορική αναδρομή στις ψηφιακές φωτογραφικές μηχανές. Θα ακολουθήσει μια αναφορά στους υπολογιστές μονής πλακέτας με ιδιαίτερη έμφαση στην σειρά μοντέλων Raspberry Pi, καθώς το Raspberry Pi 3 Model B θα χρησιμοποιηθεί για την εκπόνηση αυτής της εργασίας. Θα γίνει μια σύντομη εισαγωγή στην γλώσσα σήμανσης HTML, στην γλώσσα φύλλων στυλ CSS και στην γλώσσα προγραμματισμού PHP. Έπειτα θα γίνει μια σύντομη παρουσίαση της εφαρμογής gphoto-webui και των βιβλιοθηκών λογισμικού που θα χρησιμοποιηθούν για την κατασκευή της ιστοσελίδας ελέγχου της φωτογραφικής μηχανής.

Θα ακολουθήσει το βασικό μέρος αυτής της πτυχιακής εργασίας όπου θα αναλυθεί ο τρόπος σύνδεσης της φωτογραφικής μηχανής με την πλακέτα Raspberry Pi, η εγκατάσταση των απαραίτητων βιβλιοθηκών για την λειτουργία της κάμερας και της εφαρμογής gphoto-webui. Έπειτα θα υπάρξει ανάλυση του τρόπου δημιουργίας και στησίματος της ιστοσελίδας ελέγχου καθώς και επεξήγηση των διαφόρων λειτουργιών της.

Τέλος, θα διατυπωθούν τα συμπεράσματα που προέκυψαν από την εκπόνηση αυτής της εργασίας καθώς επίσης και διάφορες επεκτάσεις ή βελτιώσεις που μπορούν να γίνουν.

1.2 Ιστορική αναδρομή στις ψηφιακές φωτογραφικές μηχανές

Όπως πολλές σύγχρονες τεχνολογίες που χρησιμοποιούμε στην καθημερινή μας ζωή η ψηφιακή φωτογραφία μπορεί να εντοπίσει τις αρχές στον ανταγωνισμό των ΗΠΑ και της Σοβιετικής Ένωσης για την εξερεύνηση του διαστήματος. Η ιστορία της ψηφιακής φωτογραφίας ξεκινάει το 1961 με τον Eugene F. Lally, που ενώ εργαζόνταν για το Jet Propulsion Laboratory, σκέφτηκε να χρησιμοποιήσει ένα μωσαϊκό φωτοαισθητήριο για την λήψη ψηφιακών φωτογραφιών των άστρων και των πλανητών για την συλλογή πληροφοριών για την θέση των αστροναυτών ενώ ταξιδεύουν στο διάστημα. Το 1973 η εταιρία Fairchild Semiconductor κατασκευάζει τον πρώτο αισθητήρα διάταξης συζευγμένου φορτίου (CCD). Ο αισθητήρας αυτός θα χρησιμοποιηθεί δύο χρόνια αργότερα από τον Steven Sasson, έναν μηχανικό εργαζόμενο για την εταιρία Eastman Kodak, για να κατασκευάσει την πρώτη αυτόνομη (φορητή) ψηφιακή φωτογραφική μηχανή. Το 1986 η ιαπωνική εταιρία Nikon εισάγει στην αγορά την πρώτη ψηφιακή μονοοπτική φωτογραφική μηχανή, την SVC, που απευθύνεται αποκλειστικά για χρήση από ειδησεογραφικά πρακτορεία. Το Δεκέμβριο του 1989 αρχίζει να διατίθεται στην Ιαπωνία η πρώτη ψηφιακή φορητή κάμερα για το ευρύ κοινό. Δέκα χρόνια αργότερα εισάγεται στην αγορά η Nikon D1, η πρώτη σχεδιασμένη από το μηδέν ψηφιακή μονοοπτική φωτογραφική μηχανή από μεγάλο κατασκευαστή. Από το 2003 και μετά οι πωλήσεις ψηφιακών μηχανών έχουν ξεπεράσει αυτές των αναλογικών.



Εικόνα 1: Η φωτογραφική μηχανή που θα χρησιμοποιηθεί σε αυτήν την πτυχιακή εργασία (Nikon D3)

1.3 Υπολογιστές μονής πλακέτας

Ο υπολογιστής μονής πλακέτας είναι ένας πλήρης υπολογιστής κατασκευασμένος σε μια μονή ηλεκτρονική πλακέτα, η οποία έχει όλα τα απαραίτητα μέρη, όπως επεξεργαστή, μνήμη, θήρες εισόδου και εξόδου, ενός πλήρως λειτουργικού συστήματος. Οι υπολογιστές αυτοί κατασκευάζονται για επίδειξη ή εξέλιξη συστημάτων καθώς επίσης και εκπαιδευτικούς σκοπούς και ως χρήση σαν ενσωματωμένους ελεγκτές υπολογιστών. Εξαιτίας της ευρείας γκάμας χρήσεων αυτών των υπολογιστών, χρησιμοποιούνται στην κατασκευή τους αρκετοί διαφορετικοί τύποι επεξεργαστών, μνήμης, θυρών εισόδου, εξόδου και διαφόρων άλλων περιφερειακών ανάλογα με την χρήση για την οποία προορίζεται το κάθε μοντέλο.

1.3.1 Η σειρά υπολογιστών Raspberry Pi.

Το Raspberry Pi είναι μια σειρά μικρών υπολογιστών μονής πλακέτας αναπτυγμένων στο Ηνωμένο Βασίλειο από το Raspberry Pi Foundation με σκοπό να προάγει την διδασκαλία βασικής επιστήμης των υπολογιστών και προγραμματισμού σε σχολεία και σε αναπτυσσόμενες χώρες. Για αυτόν το λόγο οι τιμές αγοράς των διαφόρων μοντέλων του κυμαίνονται από μόλις 5\$ έως 35\$. Εξαιτίας όμως αυτού του μικρού κόστους απόκτησης, του μικρού του μεγέθους, και κατανάλωσης ρεύματος το Raspberry Pi έγινε πολύ πιο δημοφιλής από όσο αναμένονταν, καθιστώντας τον Φεβρουάριο του 2015 τον πρώτο σε πωλήσεις υπολογιστή σχεδιασμένο στο Ηνωμένο Βασίλειο. Το Μάρτιο του 2018 το Raspberry Pi Foundation ανακοίνωσε ότι οι πωλήσεις έφτασαν τα 19 εκατομμύρια κομμάτια. Η μεγάλη πλειοψηφία αυτών των υπολογιστών είναι κατασκευασμένοι σε ένα

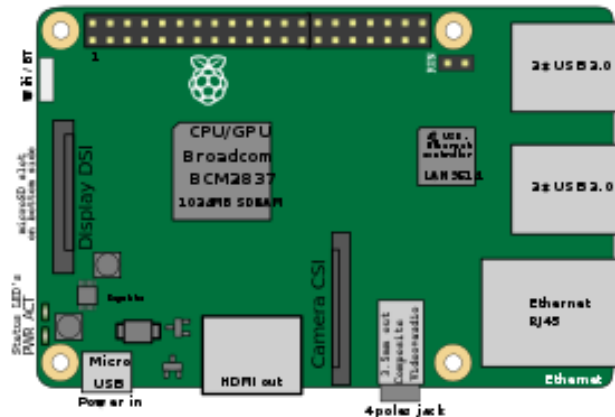
εργοστάσιο την εταιρίας Sony στο Ηνωμένο Βασίλειο με ένα μικρό αριθμό συσκευών να κατασκευάζονται σε Κίνα και Ιαπωνία.

Όλα τα μοντέλα της οικογένειας Raspberry Pi βασίζονται σε κάποιο σύστημα σε ολοκληρωμένο κύκλωμα (system on a chip) της εταιρίας Broadcom το οποίο αποτελείτε από έναν ενσωματωμένο συμβατό με την αρχιτεκτονική ARM επεξεργαστή, μία ενσωματωμένη κάρτα γραφικών και μνήμη τύπου RAM. Η ταχύτητα αυτού του επεξεργαστή κυμαίνεται από τα 700 MHz, για την πρώτη γενιά των υπολογιστών της οικογένειας έως τα 1.4 GHz, για την έκδοση 'B+' της τρίτης γενιάς. Από την δεύτερη γενιά και μετά οι επεξεργαστές που χρησιμοποιούνται σε όλα τα μοντέλα είναι τετραπύρινοι. Το μέγεθος της μνήμης RAM κυμαίνεται από τα 256 MB έως το 1 GB, ανάλογα το μοντέλο. Μνήμες τύπου SD ή MicroSD χρησιμοποιούνται για την αποθήκευση δεδομένων και του λειτουργικού συστήματος καθώς η πλακέτα δεν διαθέτει κάποιο ενσωματωμένο μέσο αποθήκευσης δεδομένων. Ο αριθμός των θυρών τύπου USB για την σύνδεση διαφόρων περιφερειακών συσκευών είναι από μία έως τέσσερις. Για την έξοδο βίντεο υπάρχουν θύρες τύπου HDMI και Composite. Η έξοδο ήχου μπορεί να γίνει είτε από τη θύρα HDMI είτε από μια θύρα 3.5 mm. Από το μοντέλο 1+ και έπειτα χρησιμοποιείτε μια τετραπολική θύρα 3.5 mm για έξοδο αναλογικού βίντεο και ήχου. Καρφάκια τύπου GPIO υπάρχουν σε όλους του υπολογιστές της σειράς για χρήση ως χαμηλού επιπέδου εξόδους. Ορισμένα μοντέλα έχουν μια 8P8C θύρα Ethernet για σύνδεση σε τοπικά δίκτυα ή στο Ίντερνετ μέσω ενός modem. Η δυνατότητα ασύρματης διασύνδεσης παρέχεται μέσω Wi-Fi 802.11n και Bluetooth στις πλακέτες που είναι εξοπλισμένες με τα αντίστοιχα ολοκληρωμένα κυκλώματα.

Όσον αφορά το λειτουργικό σύστημα του υπολογιστή, το Raspberry Pi Foundation παρέχει δωρεάν το Raspbian, μία έκδοση του λειτουργικού συστήματος Linux, για χρήση με τους υπολογιστές της σειράς. Επιπλέον στην επίσημη ιστοσελίδα υπάρχουν και άλλα λειτουργικά συστήματα, γραμμένα από τρίτους, που υποστηρίζονται επίσημα από τον υπολογιστή (όπως π.χ. Ubuntu MATE, Windows 10 IoT Core, RISC OS, FreeBSD).

1.3.2 Raspberry Pi 3 Model B

Το Raspberry Pi 3 Model B είναι το μοντέλο της σειράς Raspberry Pi το οποίο θα χρησιμοποιηθεί στην παρούσα πτυχιακή εργασία. Κυκλοφόρησε για πρώτη φορά τον Φεβρουάριο του 2016. Οι βελτιώσεις αυτής έκδοσης σε σχέση με τα προηγούμενα μοντέλα της σειράς είναι ο νέος τετρα-πύρινος 64-bit επεξεργαστής ARM Cortex-A53 στα 1.2 GHz, και τα ολοκληρωμένα κυκλώματα Wi-Fi 802.11n και Bluetooth 4.1. Ο επεξεργαστής αυτός, αν και εξαρτάται σε μεγάλο βαθμό από τις εντολές και διεργασίες που εκτελούνται, είναι κατά μέσω όρο 10 φορές πιο γρήγορος από αυτόν που χρησιμοποιούνταν στο πρώτο μοντέλο και 80% ταχύτερος από αυτόν του μοντέλου 2 στην εκτέλεση παράλληλων διεργασιών. Επιπλέον το Raspberry Pi 3 έχει την δυνατότητα να εκκινεί από αποθηκευτικά μέσα που συνδέονται με θύρα USB.



Εικόνα 1: Σχεδιάγραμμα του Raspberry Pi 3 Model B

1.4 Λογισμικό

1.4.1 gPhoto2

Το gPhoto2 είναι μία δωρεάν, αναδιανεμητέα, έτοιμη προς χρήση σειρά εφαρμογών λογισμικού για ψηφιακές κάμερες. Υποστηρίζει εκτός από την ανάκτηση εικόνων από τις συσκευές, την αποστολή εντολών για τον τηλεχειρισμό των ρυθμίσεων τους και της καταγραφής φωτογραφιών. Επιπλέον είναι συμβατό με τα πρωτόκολλα Picture Transfer Protocol (PTP) και Media Transfer Protocol (MTP) ανάλογα με της δυνατότητες της εκάστοτε συνδεδεμένης ψηφιακής φωτογραφικής μηχανής. Είναι δυνατόν να χρησιμοποιηθεί με πάνω από 2500 διαφορετικά μοντέλα καμερών και είναι συμβατό με διάφορα τύπου Unix λειτουργικά συστήματα όπως Linux, FreeBSD και NetBSD. Στο πακέτο περιλαμβάνεται και η βιβλιοθήκη λογισμικού libgphoto2 η οποία δίνει την δυνατότητα εξέλιξης άλλων πακέτων λογισμικού για την χρήση των δυνατοτήτων του gPhoto2.

1.4.2. gphoto-webui

Η εφαρμογή gphoto-webui αποτελεί την βάση αυτής της πτυχιακής εργασίας. Είναι μια διαδικτυακή εφαρμογή για το Raspberry Pi που δίνει την δυνατότητα στο χρήστη να τραβήξει και να ελέγξει τις φωτογραφίες που ήδη υπάρχουν σε αυτό μέσω μίας δυναμικής ιστοσελίδας. Η ιστοσελίδα αυτή είναι προσβάσιμη από οποιαδήποτε συσκευή έχει την δυνατότητα να συνδεθεί στο Ιντερνετ. Για την κατασκευή αυτής της εφαρμογής χρησιμοποιήθηκε η γλώσσα σήμανσης HTML και η γλώσσα προγραμματισμού PHP και JavaScript.

1.4.3 Γλώσσα σήμανσης HTML (HyperText Markup Language)

Η HTML είναι η κύρια γλώσσα σήμανσης για την κατασκευή ιστοσελίδων, και τα στοιχεία της αποτελούν τα βασικά δομικά στοιχεία όλων των ιστοσελίδων. Η HTML επιτρέπει την ενσωμάτωση κειμένου, εικόνων και άλλων αντικειμένων μέσα στη σελίδα,

και μπορεί να χρησιμοποιηθεί για να εμφανίσει τις διάφορες διαδραστικές φόρμες μίας ιστοσελίδας. Παρέχει τις μεθόδους δημιουργίας δομημένων εγγράφων, δηλαδή εγγράφων που αποτελούνται από το περιεχόμενο που μεταφέρουν και από τον κώδικα μορφοποίησης του περιεχομένου, καθορίζοντας σημαντικά δομικά στοιχεία για το κείμενο, όπως κεφαλίδες, παραγράφους, λίστες, συνδέσμους, παραθέσεις και άλλα. Μπορούν επίσης να ενσωματώνονται σενάρια εντολών σε άλλες γλώσσες όπως η JavaScript, τα οποία επηρεάζουν τη συμπεριφορά των ιστοσελίδων HTML.

1.4.4 Η γλώσσα φύλλων στυλ CSS (Cascading Style Sheets)

Η CSS είναι μια γλώσσα υπολογιστή που ανήκει στην κατηγορία των γλωσσών φύλλων στυλ που χρησιμοποιείται για τον έλεγχο της εμφάνισης ενός εγγράφου που έχει γραφτεί με μια γλώσσα σήμανσης. Χρησιμοποιείται δηλαδή για τον έλεγχο της εμφάνισης ενός εγγράφου που γράφτηκε στις γλώσσες HTML και XHTML, δηλαδή για τον έλεγχο της εμφάνισης μιας ιστοσελίδας. Η CSS είναι μια γλώσσα υπολογιστή προορισμένη για να διαμορφώνει περισσότερα χαρακτηριστικά, χρώματα, στοίχιση και να δίνει περισσότερες δυνατότητες σε σχέση με την HTML. Για μια όμορφη και καλοσχεδιασμένη ιστοσελίδα η χρήση της CSS κρίνεται ως απαραίτητη.

1.4.5 Η γλώσσα προγραμματισμού JavaScript

Η JavaScript είναι μια διερμηνευμένη γλώσσα προγραμματισμού για ηλεκτρονικούς υπολογιστές. Αρχικά αποτελούσε μέρος της υλοποίησης των φυλλομετρητών, ώστε τα σενάρια από την πλευρά του πελάτη να μπορούν να επικοινωνούν με τον χρήστη. Να ανταλλάσσουν, δηλαδή, δεδομένα ασύγχρονα και να αλλάζουν δυναμικά το περιεχόμενο του εγγράφου που εμφανίζεται. Η JavaScript είναι μια γλώσσα σεναρίων που βασίζεται στα πρωτότυπα (prototype-based), είναι δυναμική, με ασθενείς τύπους και έχει συναρτήσεις ως αντικείμενα πρώτης τάξης. Η σύνταξή της είναι επηρεασμένη από τη C. Η JavaScript αντιγράφει πολλά ονόματα και συμβάσεις ονοματοδοσίας από τη Java, αλλά γενικά οι δύο αυτές γλώσσες δε σχετίζονται και έχουν πολύ διαφορετική σημασιολογία. Οι βασικές αρχές σχεδιασμού της JavaScript προέρχονται από τις γλώσσες προγραμματισμού Self και Scheme. Είναι γλώσσα βασισμένη σε διαφορετικά προγραμματιστικά παραδείγματα (multi-paradigm), υποστηρίζοντας αντικειμενοστρεφές, προστακτικό και συναρτησιακό στυλ προγραμματισμού. Η γλώσσα αυτή χρησιμοποιείται και σε άλλες εφαρμογές εκτός ιστοσελίδων. Οι νεότερες εικονικές μηχανές και πλαίσια ανάπτυξης για JavaScript, όπως το Node.js, έχουν επίσης κάνει τη JavaScript πιο δημοφιλή για την ανάπτυξη εφαρμογών Ιστού στην πλευρά του διακομιστή (server-side).

1.4.6 Η γλώσσα προγραμματισμού PHP

Η PHP είναι μια γλώσσα προγραμματισμού για τη δημιουργία σελίδων web με δυναμικό περιεχόμενο. Μια σελίδα PHP περνά από επεξεργασία από ένα συμβατό διακομιστή του Παγκόσμιου Ιστού (π.χ. Apache), ώστε να παραχθεί σε πραγματικό χρόνο το τελικό περιεχόμενο, που είτε θα σταλεί στο πρόγραμμα περιήγησης των επισκεπτών σε μορφή κώδικα HTML είτε θα επεξεργασθεί τις εισόδους δίχως να προβάλλει την έξοδο στο χρήστη, αλλά θα τις μεταβιβάσει σε κάποιο άλλο PHP σενάριο για χρήση. Η PHP

αποτελεί μια από τις πιο διαδεδομένες τεχνολογίες στο Παγκόσμιο Ιστό, καθώς χρησιμοποιείται από πληθώρα εφαρμογών και ιστοσελίδων. Η ευρύτητα στη χρήση της είναι απόρροια της ευκολίας που παρουσιάζει ο προγραμματισμός με αυτή αλλά και στο γεγονός πως είναι μια γλώσσα η οποία βρίσκεται σχεδόν σε κάθε διακομιστή.

Κεφάλαιο 2: Υλοποίηση του συστήματος απομακρισμένου ελέγχου

2.1: Εγκατάσταση του απαραίτητου υλισμικού και λογισμικού στο Raspberry Pi

2.1.1: Επιλογή του κατάλληλου μέσου αποθήκευσης

Όπως έχει αναφερθεί και νωρίτερα το Raspberry Pi δεν διαθέτει ενσωματωμένο αποθηκευτικό χώρο για αυτό το λόγο πριν ο χρήστης αρχίσει οποιαδήποτε εργασία με τον υπολογιστή αυτό θα πρέπει να επιλέξει το σωστό αποθηκευτικό μέσο. Το Raspberry Pi Foundation προτείνει για χρήση μια κάρτας μνήμης με αποθηκευτικό χώρο τουλάχιστον 8GB κατηγορίας ταχύτητας 6 (δυνατότητα εγγραφής τουλάχιστον 6 MB/s). Η επιλογή αυτή είναι επαρκής για απλή χρήση, όμως, επειδή στο σύστημα που θα υλοποιηθεί θα γίνετε συχνά εγγραφή μεγάλων αρχείων (το μέγεθος των φωτογραφιών χωρίς συμπίεση φτάνει τα 25MB και μπορεί να ξεπεράσει τα 35MB σε περίπτωση που ο χρήστης επιλέξει να αποθηκευτούν σε αρχεία τύπου TIFF) θα προτιμηθεί μια μνήμη κατηγορίας ταχύτητας U1, με ταχύτητα εγγραφής 19.6MB/s, και χωρητικότητα 16GB, ώστε να αυξηθεί ο αριθμός των φωτογραφιών που μπορούν να είναι αποθηκευμένες στην κάρτα. Το Raspberry Pi 3 Model B έχει την δυνατότητα να χρησιμοποιήσει και συσκευές μαζικής αποθήκευσης USB ως κύριο αποθηκευτικό μέσο εκτός από κάρτες τύπου microSD, όμως η χρήση τους δεν θα προτιμηθεί για να περιοριστεί το μέγεθος της κατασκευής και ο αριθμός των θυρών USB που χρειάζονται για την υλοποίηση.

2.1.2: Επιλογή του κατάλληλου λειτουργικού συστήματος και εγκατάστασης του

Το Raspberry Pi είναι συμβατό με μια ευρεία ποικιλία λειτουργικών συστημάτων, δίνοντας στον χρήστη την δυνατότητα της επιλογής ανάλογα με την χρήση που σκοπεύει να κάνει με τον υπολογιστή αυτό. Για να περιοριστεί η πιθανότητα μη συμβατότητας μεταξύ των διαφόρων πακέτων και βιβλιοθηκών λογισμικού που χρειάζονται για την υλοποίηση αυτή θα χρησιμοποιηθεί η πιο πρόσφατη έκδοση (Ιούνιος του 2018) του Raspbian, του επίσημου λειτουργικού συστήματος όλων των μοντέλων του Raspberry Pi.

Το Raspberry Pi Foundation προτείνει το πρόγραμμα Etcher για την εγγραφή του λειτουργικού συστήματος στην κάρτα μνήμης. Το Etcher είναι ένα γραφικό εργαλείο εγγραφής κάρτας SD που είναι συμβατό με λειτουργικά συστήματα Mac OS, Linux και Windows. Έχει την δυνατότητα να εγγράψει αρχεία εικόνων κατευθείαν από αρχεία τύπου zip χωρίς να χρειάζεται κάποια αποσυμπίεση.

Τα βήματα για την εγγραφή μιας εικόνας με το πρόγραμμα Etcher είναι:

1. Κατέβασμα και εγκατάσταση του προγράμματος αυτού ακολουθώντας τις οδηγίες που εμφανίζονται στην οθόνη.
2. Τοποθέτηση μίας κατάλληλης κάρτας SD σε μια συσκευή ανάγνωσης καρτών που είναι συνδεδεμένη στον υπολογιστή.
3. Άνοιγμα του Etcher και επιλογή του αρχείου εικόνας του Raspbian (μπορεί να είναι τύπου IMG ή ZIP).
4. Επιλογή της κάρτας SD που θα γίνει εγγραφή.

5. Κλικ στο κουμπί 'Flash!' για να ξεκινήσει η εγγραφή δεδομένων στην κάρτα.
6. Με την εμφάνιση του μηνύματος 'Flash Compete!' η διεργασία ολοκληρώνεται και η κάρτα είναι έτοιμη για χρήση στο Raspberry Pi.

2.1.3: Εγκατάσταση απαραίτητων πακέτων λογισμικού και βιβλιοθηκών στο Raspbian

Μετά την επιτυχημένη εγκατάσταση του λειτουργικού συστήματος θα ακολουθήσει η εγκατάσταση των απαραίτητων πακέτων λογισμικού και βιβλιοθηκών στο λειτουργικό σύστημα του Pi.

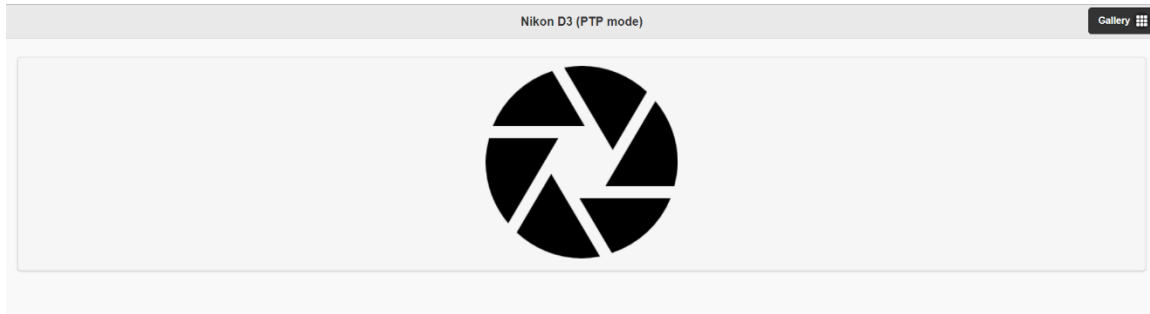
Η εγκατάσταση αυτή θα γίνει από το Τερματικό παράθυρο (Terminal) του Raspbian, όμως προέχει η εγκατάσταση τυχόν ενημερώσεων που πιθανόν υπάρχουν. Αυτό θα πραγματοποιηθεί με την εντολή `'sudo apt-get update'`. Η εντολή `'sudo'` είναι αυτή που δίνει στον χρήστη δικαιώματα διαχειριστή, απαραίτητη πριν από κάθε εντολή που εγκαθιστά κάτι στο σύστημα και η εντολή `'apt-get update'` πραγματοποιεί τον έλεγχο και την εγκατάσταση τυχόν ενημερώσεων. Όταν ολοκληρωθεί αυτή η διαδικασία θα χρησιμοποιηθεί για την εγκατάσταση του πακέτου λογισμικού gPhoto2 το σενάριο (script): `'wget https://raw.githubusercontent.com/gonzalo/gphoto2-updater/master/gphoto2-updater.sh && chmod +x gphoto2-updater.sh && sudo ./gphoto2-updater.sh'`. Ο λόγος για τον οποίο γίνεται αυτό και δεν χρησιμοποιείτε η εντολή `'sudo apt-get install gphoto2'` είναι διότι σε ορισμένες εκδόσεις του Raspbian υπάρχουν προβλήματα εγκατάστασης του gphoto2 με αυτή την μέθοδο. Έπειτα θα γίνει η εγκατάσταση των απαραίτητων βιβλιοθηκών για την σωστή λειτουργία της εφαρμογής gphoto-webui με την εντολή: `'sudo apt-get install imagemagick php php-imagick php-cli zip unzip exiv2'`. Αυτή η εντολή εγκαθιστά γλώσσα προγραμματισμού php, της επεκτάσεις της imagemagick, imagick, cli και τα πακέτα λογισμικού zip, unzip, exiv2. Τέλος για να γίνει χρήση της gphoto-webui θα χρησιμοποιηθεί το πρόγραμμα wget, που ανακτά περιεχόμενο από διακομιστές ιστού, πληκτρολογώντας: `'wget https://github.com/theonemule/gphoto-webui/archive/master.zip'`. Θα ακολουθήσει η αποσυμπίεση των δεδομένων που λήφθηκαν από το Διαδίκτυο σε μια εγγράψιμη τοποθεσία (όπως ο φάκελος home) με την εντολή `'unzip master.zip'`.

2.1.4: Δοκιμή της σωστής λειτουργίας των πακέτων λογισμικού

Προτού γίνει οποιαδήποτε τροποποίηση ή προσθήκη στα παραπάνω πακέτα λογισμικού θα ήταν φρόνιμο να γίνει μια δοκιμή για να επιβεβαιωθεί η σωστή λειτουργία τους και για να εντοπιστούν τυχόν σφάλματα που μπορεί να έγιναν κατά την εγκατάστασή τους.

Αρχικά θα συνδεθεί η κάμερα με τον υπολογιστή μέσω της θύρας USB. Έπειτα στο Τερματικό θα γίνει μετάβαση στον φάκελο όπου έγινε η αποσυμπίεση των δεδομένων της εφαρμογής gphoto-webui (στην παρούσα εργασία στο φάκελο 'gphoto-webui-master' με την εντολή: `'cd gphoto-webui-master'`). Εκεί θα ξεκινήσει η λειτουργία της ιστοσελίδας με την εντολή `'php -S 0.0.0.0:8000'`. Μετά την εκκίνηση της θα πληκτρολογηθεί η διεύθυνση IP του Pi και η πύλη 8000 (π.χ. 192.168.1.2:8000) στην γραμμή διευθύνσεων ενός προγράμματος για να αποκτηθεί πρόσβαση στην εφαρμογή gphoto-webui.

Εφόσον η παραπάνω διαδικασία ολοκληρώθηκε με επιτυχία, ο χρήστης θα βρεθεί σε μία ιστοσελίδα σαν αυτής της εικόνας 3 . Εάν έχει γίνει σωστή αναγνώριση της φωτογραφικής μηχανής από τον υπολογιστή, θα αναφέρεται το μοντέλο της (εδώ Nikon D3 (PTP mode)). Κάνοντας κλικ στο κουμπί με σήμα το διάφραγμα θα ενεργοποιείτε το κλείστρο της κάμερας και θα αποθηκεύεται μια φωτογραφία στον φάκελο images της εφαρμογής. Επιλέγοντας το κουμπί 'Gallery' ο χρήστης θα μεταβαίνει σε μία άλλη σελίδα όπου μπορεί να δει και να κατεβάσει όσες φωτογραφίες υπάρχουν στον φάκελο αυτό.



Εικόνα 3: Η βασική σελίδα της gphoto-webui

2.2: Κατασκευή της δυναμικής ιστοσελίδας χρησιμοποιώντας ως βάση το gPhoto2 και το gphoto-webui

2.2.1: Δημιουργία του κουμπιού κατεύθυνσης προς την σελίδα αλλαγής των ρυθμίσεων της φωτογραφικής μηχανής

Για να μην επηρεαστούν οι υπάρχουσες λειτουργίες της εφαρμογής gphoto-webui θα δημιουργηθεί μια νέα δυναμική ιστοσελίδα σε ένα άλλο αρχείο τύπου HTML για τις προσθήκες που θα γίνουν. Ένα κουμπί θα προστεθεί στην υπάρχουσα ιστοσελίδα για να ανακατευθύνει τον χρήστη στην νέα όπου θα μπορεί να πραγματοποιεί τις αλλαγές στις ρυθμίσεις της μηχανής που επιθυμεί.

Απαραίτητο για την δημιουργία αυτού του κουμπιού είναι ένα αρχείο εικόνας τύπου PNG για χρήση ως εικονίδιο. Το εικονίδιο αυτό, όπως και τα υπόλοιπα που τυχόν θα χρησιμοποιηθούν, θα πρέπει να είναι ευδιάκριτο και να βοηθάει τον χρήστη να καταλάβει την λειτουργία του κουμπιού που αντιπροσωπεύει. Για ευκολία χρήσης το αρχείο αυτό θα τοποθετηθεί στον ριζικό φάκελο της εφαρμογής.

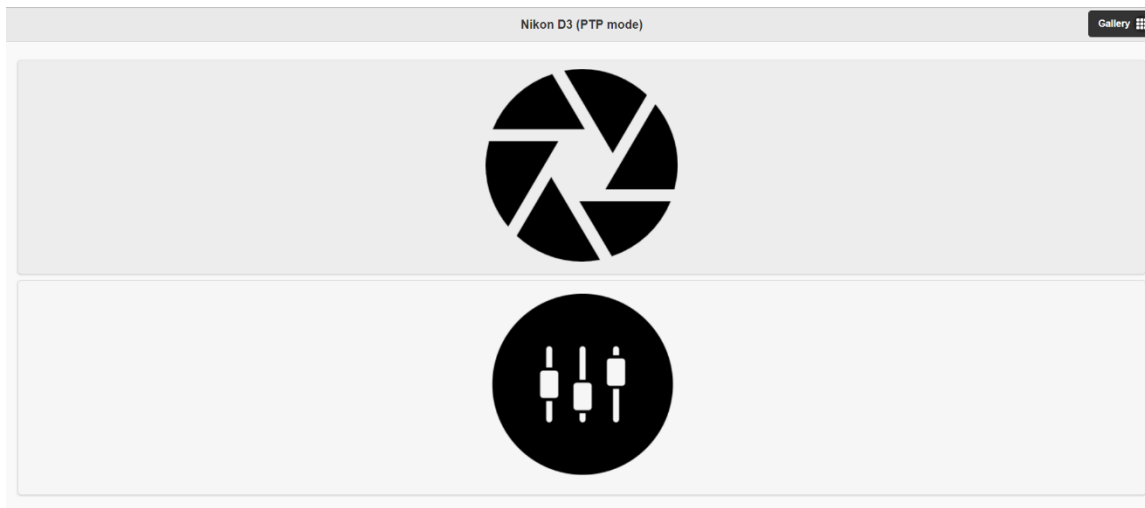
Χρησιμοποιώντας είτε ένα εξειδικευμένο πρόγραμμα συντάκτη HTML είτε έναν απλό επεξεργαστή κειμένου θα γίνει προσθήκη του νέου κώδικα στο αρχείο index.html της gphoto-webui κάτω από αυτόν του ήδη υπάρχοντος κουμπιού ώστε να εμφανίζονται το ένα κάτω από το άλλο. Ο κώδικας που θα προστεθεί είναι ο εξής:

```
<button onclick="location.href='settings.html#settings'" id="set"></button>
```

Το στοιχείο `<button>` χρησιμοποιείται για τη δημιουργία ενός κουμπιού HTML. Το `onclick` είναι ένας χειριστής συμβάντων που δίνει οδηγίες στο πρόγραμμα περιήγησης να εκτελεί ένα σενάριο όταν ο επισκέπτης κάνει κλικ σε ένα κουμπί. Το σενάριο που εκτελείτε στην συγκεκριμένη περίπτωση είναι το εξής: `location.href=`

'settings.html#settings'". Αυτό καθορίζει τον προορισμό του συνδέσμου που είναι η ετικέτα `settings` του αρχείου `settings.html` και εμφανίζει τα δεδομένα της στον περιηγητή. Το χαρακτηριστικό `id` προσδιορίζει ένα μοναδικό αναγνωριστικό για ένα στοιχείο HTML (η τιμή πρέπει να είναι μοναδική στο έγγραφο HTML). Το `id` μπορεί, επίσης, να χρησιμοποιηθεί από το CSS και τη JavaScript για την εκτέλεση ορισμένων εργασιών του στοιχείου με την καθορισμένη τιμή `id`. Στην HTML, οι εικόνες ορίζονται με την ετικέτα ``. Η ετικέτα `` είναι κενή, περιέχει μόνο χαρακτηριστικά και δεν έχει ετικέτα κλεισίματος. Το χαρακτηριστικό `src` καθορίζει τη διεύθυνση URL ή το αρχείο της εικόνας (όπως συμβαίνει εδώ). Το `</button>` είναι η ετικέτα κλεισίματος του κουμπιού.

Σε αυτό το σημείο η ιστοσελίδα θα είναι παρόμοια με αυτήν της εικόνας 4. Εξαιτίας όμως του γεγονότος ότι το αρχείο `settings.html` δεν έχει δημιουργηθεί ακόμη εάν ο χρήστης κάνει κλικ στο κουμπί που μόλις δημιουργήθηκε ο περιηγητής θα εμφανίσει σφάλμα. Για αυτό το λόγο το επόμενο βήμα θα είναι η δημιουργία μιας βασικής έκδοσης της σελίδας `settings.html` για να ανακατευθύνεται ο χρήστης.



Εικόνα 4: Η βασική σελίδα μετά την προσθήκη του κουμπιού ρυθμίσεων

2.2.2: Δημιουργία της βασικής δομής της ιστοσελίδας αλλαγής των ρυθμίσεων

Το πρώτο βήμα στην κατασκευή της νέας σελίδας είναι η δημιουργία του αρχείου `settings.html`. Ομοίως με το προηγούμενο βήμα αυτό μπορεί να επιτευχθεί είτε με ένα εξειδικευμένο πρόγραμμα συντάκτη HTML είτε με έναν απλό επεξεργαστή κειμένου. Η διαδικασία για να πραγματοποιηθεί αυτό με ένα συντάκτη HTML διαφέρει ανάλογα με το εκάστοτε πρόγραμμα. Εξαιτίας αυτού, θα γίνει περιγραφή του τρόπου δημιουργίας ενός αρχείου HTML με την χρήση ενός επεξεργαστή κειμένου που είναι παρόμοια ανεξαρτήτως του προγράμματος που θα χρησιμοποιηθεί.

Σε ένα νέο αρχείο κειμένου θα γραφτεί ο εξής απαραίτητος κώδικας για την κατασκευή μιας βασικής ιστοσελίδας:

```
<!DOCTYPE html>
<html>
<head>
<title>Page Title</title>
```

```
</head>
<body>
<h1>Heading</h1>
</body>
</html>
```

Όλα τα έγγραφα τύπου HTML πρέπει να ξεκινούν με την δήλωση `<!DOCTYPE html>` που δηλώνει τον τύπο του εγγράφου. Το ίδιο έγγραφο ξεκινάει με την ετικέτα `<html>` και τελειώνει με την `</html>`. Το στοιχείο `<head>` περιέχει μεταδεδομένα (δεδομένα σχετικά με δεδομένα) και τοποθετείται μεταξύ της ετικέτας `<html>` και της ετικέτας `<body>`. Τα μεταδεδομένα HTML είναι δεδομένα σχετικά με το έγγραφο HTML και δεν εμφανίζονται. Τυπικά ορίζουν τον τίτλο του εγγράφου, το σύνολο χαρακτήρων, τα στυλ, τους συνδέσμους, τα σενάρια και άλλες πληροφορίες μεταβάθμισης. Οι ακόλουθες ετικέτες περιγράφουν τα μεταδεδομένα: `<title>`, `<style>`, `<meta>`, `<link>`, `<script>` και `<base>`. Το στοιχείο `<title>` ορίζει τον τίτλο του εγγράφου και απαιτείται σε όλα τα έγγραφα HTML / XHTML. Επιπλέον ορίζει έναν τίτλο στην καρτέλα του προγράμματος περιήγησης, παρέχει έναν τίτλο για τη σελίδα όταν προστίθεται στα αγαπημένα και εμφανίζει έναν τίτλο για τη σελίδα στα αποτελέσματα των μηχανών αναζήτησης. Το ορατό μέρος του αρχείου βρίσκεται ανάμεσα στις ετικέτες `<body>` και `</body>`. Οι επικεφαλίδες HTML ορίζονται με τις ετικέτες `<h1>` έως `<h6>` με την `<h1>` να είναι πιο σημαντική και η `<h6>` η λιγότερο σημαντική.

Για να ολοκληρωθεί η δημιουργία της σελίδας θα πρέπει να σωθεί το έγγραφο που μόλις γράφτηκε με το σωστό τύπο αρχείου και κωδικοποίηση. Αυτό γίνεται από την επιλογή 'Αποθήκευση ως' του μενού 'Αρχείο'. Στο νέο παράθυρο που εμφανίζεται δίνεται ως όνομα αρχείου το `settings.html`, ώστε να σωθεί σαν αρχείο τύπου HTML και επιλέγεται η κωδικοποίηση UTF-8. Το αρχείο αυτό θα πρέπει να σωθεί στον ίδιο φάκελο όπου βρίσκεται το `index.html` της εφαρμογής `gphoto-webui` για να λειτουργεί η ανακατεύθυνση από την βασική ιστοσελίδα σωστά.

Εφόσον τα παραπάνω έχουν πραγματοποιηθεί με επιτυχία ανοίγοντας την σελίδα που δημιουργήθηκε νωρίτερα θα εμφανίζεται μια σελίδα με το όνομα 'Page Title' όπου θα υπάρχει μόνο μια επικεφαλίδα που θα γράφει 'Heading'.

2.2.3: Προσθήκη των μεταδεδομένων της νέας ιστοσελίδας

Απαραίτητα για την σωστή λειτουργία των ρυθμίσεων που θα προστεθούν είναι μια σειρά αρχείων σεναρίων JavaScript. Τα αρχεία αυτά θα πρέπει να δηλωθούν στα μεταδεδομένα ώστε να μπορούν να καλούνται τα σενάρια τα οποία περιέχουν από την ιστοσελίδα. Επιπλέον θα γίνει προσθήκη ορισμένων πληροφοριών για τον έλεγχο της εμφάνισης της ιστοσελίδας.

Στο αρχείο που δημιουργήθηκε στο προηγούμενο βήμα θα συμπληρωθεί ο παρακάτω κώδικας μετά από την ετικέτα `<head>`:

```
<title>Camera Control Page</title>
<meta name="viewport" content="width=device-width, initial-
scale=1">
<link rel="stylesheet" href="css/jquery.mobile-1.4.2.css" />
<script src="js/jquery.js"></script>
<script src="js/jquery.mobile-1.4.2.js"></script>
<script src="js/controller.js"></script>
</head>
<style>
.ui-grid-a img {
width : 90%;
```



```
height : auto;
border:1px #444444 solid;
box-shadow: 2px 2px 2px #888888;
}
</style>
```

Ο τίτλος της σελίδας θα αλλάξει σε 'Camera Control Page' ώστε να είναι όμοιος με αυτό του αρχείου index.html της εφαρμογής gphoto-webui. Η επόμενη σειρά απαιτείται για την ρύθμιση του παράθυρου προβολής (viewport). Το παράθυρο προβολής είναι η ορατή περιοχή του χρήστη μιας ιστοσελίδας. Διαφέρει ανάλογα με τη συσκευή π.χ. θα είναι μικρότερη σε ένα κινητό τηλέφωνο από ότι σε μία οθόνη υπολογιστή. Η HTML5 εισήγαγε μια μέθοδο ελέγχου του παραθύρου προβολής μέσω της ετικέτας `<meta>`. Το στοιχείο αυτό δίνει στο πρόγραμμα περιήγησης οδηγίες σχετικά με τον τρόπο ελέγχου των διαστάσεων και της κλίμακας της σελίδας. Το `'width=device-width'` μέρος ορίζει το πλάτος της σελίδας ώστε να ακολουθεί το εύρος οθόνης της συσκευής (το οποίο θα διαφέρει ανάλογα με τη συσκευή). Το `'initial-scale=1'` καθορίζει το αρχικό επίπεδο ζουμ όταν η σελίδα φορτώνεται για πρώτη φορά από το πρόγραμμα περιήγησης. Η ετικέτα `<link>` ορίζει μια σύνδεση μεταξύ ενός εγγράφου και ενός εξωτερικού πόρου. Το γνώρισμα (Attribute) `rel` είναι απαραίτητο για την ετικέτα αυτή και προσδιορίζει τη σχέση μεταξύ του τρέχοντος εγγράφου και του συνδεδεμένου εγγράφου δηλαδή στην συγκεκριμένη περίπτωση ότι το αρχείο jquery.mobile-1.4.2.css, που βρίσκεται στον υποφάκελο css της εφαρμογής, είναι το φύλλο στυλ (stylesheet) του παρόν εγγράφου. Η ετικέτα `<script>` χρησιμοποιείται για τον ορισμό μιας δέσμης ενεργειών πελάτη (client-side script). Μπορεί να περιέχει εντολές δέσμης ενεργειών, ή να δείχνει ένα εξωτερικό αρχείο δέσμης ενεργειών μέσω του χαρακτηριστικού `src`, όπως συμβαίνει παραπάνω όπου κατευθύνει στον υποφάκελο js όπου υπάρχουν τα 3 αρχεία δέσμης ενεργειών που χρειάζονται για την σωστή λειτουργία της σελίδας. Εκεί υπάρχουν οι βιβλιοθήκες jQuery και jQuery modile της Javascript (jquery.js, jquery.mobile-1.4.2.) και οι λειτουργίες της εφαρμογής gphoto-webui (controller.js). Στο τελευταίο αρχείο θα γίνει και η προσθήκη των εντολών ελέγχου της φωτογραφικής μηχανής. Η ετικέτα `<style>` περιέχει πληροφορίες για την εμφάνιση του περιβάλλοντος χρήστη (user interface) και είναι ακριβώς όμοιες με αυτές του index.html ώστε οι δύο σελίδες να είναι ίδιες εμφανισιακά.

2.2.4: Ολοκλήρωση της ανακατεύθυνσης και δημιουργία της κεφαλίδας (header) στην νέα σελίδα

Η ετικέτα προορισμού `'settings'` που κατευθύνει το κουμπί δεν υπάρχει, για αυτό τον λόγο, όταν ο χρήστης κάνει κλικ δεν θα πραγματοποιηθεί η ανακατεύθυνση. Θα πρέπει να δημιουργηθεί μία σελίδα στο αρχείο settings.html με ταυτότητα (id) `'settings'` ώστε να λειτουργήσει σωστά το κουμπί. Ο προσωρινός κώδικας που υπάρχει στην ετικέτα `<body>` θα σβηστεί και στην θέση του θα προστεθεί ο εξής κώδικας για την δημιουργία μιας τελείως κενής σελίδας:

```
<div data-role="page" id="settings">
</div>
```

Η ετικέτα `<div>`, με ετικέτα κλεισίματος την `</div>`, ορίζει ένα τμήμα ή μία ενότητα σε ένα έγγραφο HTML. Το στοιχείο αυτό χρησιμοποιείται συχνά και ως κοντέινερ για άλλα στοιχεία HTML για να τροποποιηθεί η εμφάνιση τους με την γλώσσα στυλ CSS ή για να εκτελέσουν συγκεκριμένες εργασίες JavaScript. Το στοιχείο `data-role="page"` καθορίζει ότι τα δεδομένα σε αυτό το τμήμα του αρχείου είναι μία σελίδα.

Μετά από αυτές τις προσθήκες στο κώδικα ο χρήστης θα κατευθύνεται σε μία κενή ιστοσελίδα. Η πρώτη προσθήκη που θα γίνει στην κενή σελίδα θα είναι μία κεφαλίδα παρόμοια με αυτή της σελίδας 'Gallery' της εφαρμογής ώστε να μπορεί ο χρήστης να επιστρέψει στην αρχή (Main Menu) και να μπορεί να ανακατευθυνθεί κατευθείαν στην σελίδα με τις υπάρχουσες φωτογραφίες (Gallery). Ο κώδικας αυτός θα προστεθεί πριν την ετικέτα κλεισίματος `</div>` και θα είναι ο εξής:

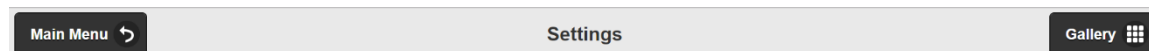
```
<div data-role="header">
  <button onclick="location.href='index.html#camera'" class="ui-btn-
left ui-btn ui-btn-b ui-btn-inline ui-mini ui-corner-all ui-btn-icon-
right ui-icon-back">Main Menu</button>
  <button onclick="location.href='index.html#gallery'" class="ui-
btn-right ui-btn ui-btn-b ui-btn-inline ui-mini ui-corner-all ui-btn-
icon-right ui-icon-grid">Gallery</button>
</a>
  <h1>Settings</h1>
</div><!-- /header -->
```

Το χαρακτηριστικό `class` καθορίζει ένα ή περισσότερα ονόματα κλάσεων για ένα στοιχείο HTML. Το όνομα κλάσης μπορεί να χρησιμοποιηθεί από το CSS και τη JavaScript για την εκτέλεση συγκεκριμένων εργασιών για στοιχεία με το καθορισμένο όνομα κλάσης. Στον παραπάνω κώδικα με την χρήση της βιβλιοθήκης jQuery της JavaScript και με την βοήθεια του `class` καθορίζεται το χρώμα, το σχήμα, η θέση, το μέγεθος και τα εικονίδια των κουμπιών 'Gallery' και 'Camera'.

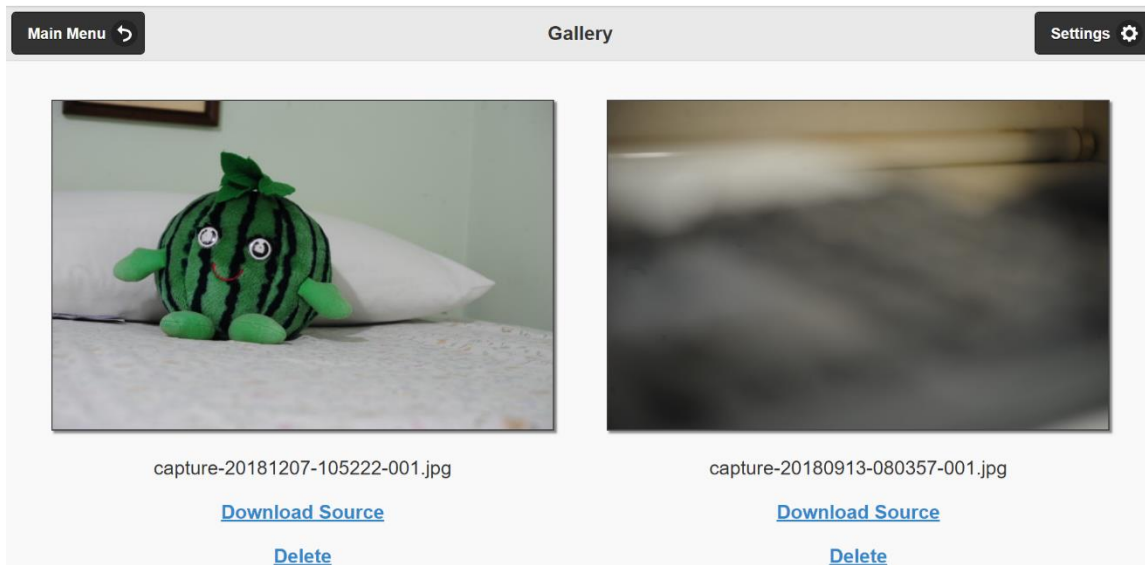
Με παρόμοιο τρόπο θα δημιουργηθεί ένα επιπλέον κουμπί στην σελίδα 'Gallery' για να μπορεί να μεταβεί απευθείας ο χρήστης στην σελίδα με τις ρυθμίσεις της φωτογραφικής μηχανής. Ο κώδικας που θα προστεθεί στην κεφαλίδα της σελίδας είναι ο εξής:

```
<button onclick="location.href='settings.html#settings'" class="ui-btn-
right ui-btn ui-btn-b ui-btn-inline ui-mini ui-corner-all ui-btn-icon-
right ui-icon-gear">Settings</button>
```

Η προσθήκη που έγινε δεν επηρεάζει τις υπόλοιπες λειτουργίες της σελίδας 'Gallery', που σημαίνει ότι ο χρήστης μπορεί ακόμα να ελέγξει, να κατεβάσει στην συσκευή του, και να σβήσει τις φωτογραφίες που έχει τραβήξει μέσω της εφαρμογής.



Εικόνα 5: Η ιστοσελίδα 'Settings' μετά την προσθήκη της κεφαλίδας



Εικόνα 6: Η ιστοσελίδα 'Gallery' μετά την προσθήκη του κουμπιού μετάβασης στην σελίδα 'Settings'

2.2.5: Δημιουργία πίνακα πληροφοριών

Στην κεντρική σελίδα της εφαρμογής (index.html) θα προστεθεί ένας πίνακας στον οποίο θα φαίνονται οι ήδη επιλεγμένες ρυθμίσεις της φωτογραφικής μηχανής. Οι ρυθμίσεις που θα εμφανίζονται είναι αυτές που θα αλλάζουν μέσω της σελίδας 'Settings' και με αυτή την προσθήκη ο χρήστης θα μπορεί να ελέγξει εύκολα και γρήγορα την κατάσταση της φωτογραφικής μηχανής δίχως να χρειάζεται να μεταβεί στην σελίδα 'Settings'.

Οι πληροφορίες που χρειάζονται για την συμπλήρωση του πίνακα αυτού θα ληφθούν απευθείας από την κάμερα μέσω της εφαρμογής gphoto2. Αυτό θα επιτευχθεί με την βοήθεια διαφόρων σεναρίων γλώσσας Javascript, που βρίσκονται στο αρχείο controller.js, τα οποία θα εκτελούν σειρές εντολών γλώσσας php, οι οποίες περιέχονται στο αρχείο service.php, και χειρίζονται την εφαρμογή gphoto2. Λεπτομερής ανάλυση της παραπάνω διαδικασίας θα γίνει στο επόμενο κεφάλαιο της εργασίας αυτής.

Στο αρχείο index.html, αμέσως μετά από το κουμπί που προστέθηκε σε προηγούμενο βήμα, θα προστεθεί ο εξής κώδικας:

```
<table>
  <tr>
    <th>Name</th>
    <th></th>
  </tr>
  <tr>
    <td>Exposure Mode</td>
    <td id="expmode"></td>
  </tr>
  <tr>
    <td>Aperture Size</td>
    <td id="apsize"></td>
  </tr>
  <tr>
    <td>Shutter Speed</td>
```

```

    <td id="shutterspeed"></td>
</tr>
<tr>
    <td>Auto ISO</td>
    <td id="autoiso"></td>
</tr>
<tr>
    <td>ISO</td>
    <td id="iso"></td>
</tr>
<tr>
    <td>Exposure Compensation</td>
    <td id="expcomp"></td>
</tr>
<tr>
    <td>Flexible Program Shift</td>
    <td id="flex"></td>
</tr>
<tr>
    <td>Image Quality</td>
    <td id="quality"></td>
</tr>
<tr>
    <td>Image Size</td>
    <td id="size"></td>
</tr>
<tr>
    <td>RAW Compression</td>
    <td id="rsize"></td>
</tr>
<tr>
    <td>White Balance</td>
    <td id="wbalance"></td>
</tr>
<tr>
    <td>Color Space</td>
    <td id="colorspace"></td>
</tr>
<tr>
    <td>High ISO Noise Reduction</td>
    <td id="isonr"></td>
</tr>
<tr>
    <td>Long Exposure Noise Reduction</td>
    <td id="lnr"></td>
</tr>
<tr>
    <td>Flash Mode</td>
    <td id="flashmode"></td>
</tr>
<tr>
    <td>Flash Sync Speed</td>
    <td id="flashsyncspeed"></td>
</tr>
<tr>
    <td>Flash Shutter Speed</td>
    <td id="flashshutterspeed"></td>

```

```
</tr>
</table>
```

Η ετικέτα `<table>` ορίζει έναν πίνακα HTML. Ένας πίνακας HTML αποτελείται από το στοιχείο `<table>` και ένα ή περισσότερα στοιχεία `<tr>`, `<th>` και `<td>`. Το στοιχείο `<tr>` ορίζει μια σειρά πίνακα, το στοιχείο `<th>` ορίζει μια κεφαλίδα πίνακα και το στοιχείο `<td>` ορίζει μια κυψέλη πίνακα. Ένας πιο σύνθετος πίνακας HTML μπορεί επίσης να περιλαμβάνει στοιχεία `<caption>`, `<col>`, `<colgroup>`, `<thead>`, `<tfoot>` και `<tbody>`. Στο αριστερό κελί της κάθε σειράς του πίνακα υπάρχει το όνομα της κάθε τρέχον επιλογής που εμφανίζεται στο δεξί κελί. Τα δεξιά κελιά περιέχουν την ταυτότητα (`id`) στην οποία αποθηκεύονται οι τιμές των ρυθμίσεων που λαμβάνονται από την κάμερα μέσω της `gphoto2`. Επειδή οι ταυτότητες δεν χρησιμοποιούνται προς το παρόν τα κελιά αυτά θα είναι κενά.

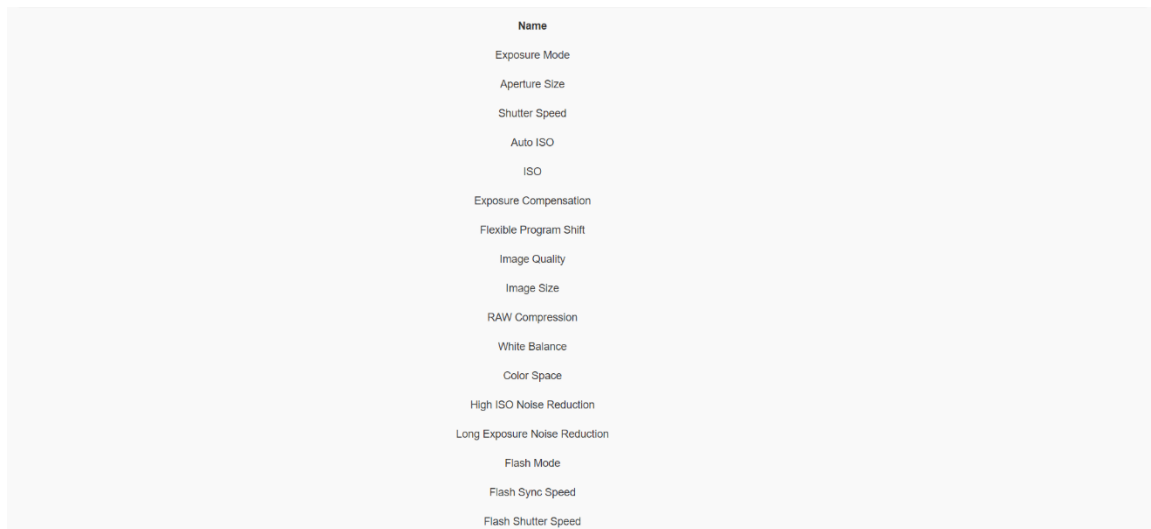
Στην συνέχεια θα γίνει προγραμματισμός της εμφάνισης του πίνακα αυτού προσθέτοντας τον παρακάτω κώδικα στην ετικέτα `<style>`:

```
table {
    border-collapse: collapse;
    width: 100%;
}

th, td {
    padding: 12px;
}
```

Η ιδιότητα της CSS `border-collapse`: καθορίζει εάν τα σύνορα των πινάκων πρέπει να συμπίπτουν σε ένα μόνο περιθώριο ή να διαχωρίζονται όπως στην τυπική HTML. Με την επιλογή `collapse` γίνεται σύμπτυξη των δύο περιθωρίων σε ένα. Η ιδιότητα `width` καθορίζει το πλάτος ενός στοιχείου. Το `100%` σημαίνει ότι το στοιχείο θα εμφανίζεται σε όλο το διαθέσιμο χώρο του εκάστοτε παράθυρου προβολής. Το `'padding'` ενός στοιχείου (των στοιχείων `th` και `td` σε αυτήν την περίπτωση) είναι ο χώρος μεταξύ του περιεχομένου του και των ορίων του. Με το `12px` ορίζεται πως το μέγεθος του θα είναι `12px` σε όλες τις τέσσερις πλευρές των στοιχείων.

Η εικόνα 6 δείχνει την τελική εμφάνιση του πίνακα μετά από τις προσθήκες που έγιναν στην ετικέτα `<style>`.



Name
Exposure Mode
Aperture Size
Shutter Speed
Auto ISO
ISO
Exposure Compensation
Flexible Program Shift
Image Quality
Image Size
RAW Compression
White Balance
Color Space
High ISO Noise Reduction
Long Exposure Noise Reduction
Flash Mode
Flash Sync Speed
Flash Shutter Speed

Εικόνα 6: Ο πίνακας πληροφοριών των ρυθμίσεων

Όνομα	Μετάφραση	Λειτουργία
Exposure Mode	Λειτουργία Έκθεσης	Καθορίζει ποιες ρυθμίσεις μπορούν να προσδιορίζουν την έκθεση
Aperture Size	Μέγεθος Διαφράγματος	Αλλάζει το μέγεθος διαφράγματος του φακού
Shutter Speed	Ταχύτητα Κλείστρου	Αλλάζει την ταχύτητα του κλείστρου της κάμερας
Auto ISO	Αυτόματος Έλεγχος Ευαισθησίας ISO	Ενεργοποιεί ή απενεργοποιεί τον αυτόματο έλεγχο της ευαισθησίας ISO
ISO	Ευαισθησία ISO	Αλλάζει την ευαισθησία της φωτογραφικής μηχανής στο φως
Exposure Compensation	Αντιστάθμιση Έκθεσης	Αλλάζει την εσκεμμένη μεταβολή της έκθεσης σε σχέση με την τιμή που συνηθίζει η κάμερα
Flexible Program Shift	Μετατόπιση του Ευέλικτου Προγράμματος	Αλλάζει τον συνδυασμό ταχύτητας κλείστρου και διαφράγματος
Image Quality	Ποιότητα Εικόνας	Αλλάζει την ποιότητα εικόνας
Image Size	Μέγεθος Εικόνας	Αλλάζει το μέγεθος των εικόνων
RAW Compression	Συμπίεση των Αρχείων Τύπου RAW	Αλλάζει το είδος της συμπίεσης των αρχείων τύπου RAW
White Balance	Ισορροπία Λευκού	Αλλάζει την επιλεγμένης ρύθμισης ισορροπίας λευκού
Color Space	Χρωματικός Χώρος	Αλλάζει τον τύπο του χρωματικού χώρου
High ISO Noise Reduction	Αποθρομβοποίηση Υψηλών Τιμών ISO	Αλλάζει τον τύπο αποθρομβοποίησης υψηλών τιμών ISO
Long Exposure Noise Reduction	Αποθρομβοποίηση Έκθεσης Μακράς Διάρκειας	Ενεργοποιεί ή απενεργοποιεί την αποθρομβοποίηση σε εκθέσεις μακράς διάρκειας
Flash Mode	Λειτουργία Φλας	Αλλάζει τον τρόπο λειτουργίας φλας
Flash Sync Speed	Ταχύτητα Συγχρονισμού Φλας	Αλλάζει την ταχύτητα συγχρονισμού φλας
Flash Shutter Speed	Ταχύτητα Κλείστρου Φλας	Αλλάζει την ταχύτητα κλείστρου φλας

2.2.6: Προγραμματισμός του τρόπου ανάκτησης πληροφοριών από την φωτογραφική μηχανή

Η εφαρμογή `gphoto2` έχει την δυνατότητα με την χρήση ορισμένων εντολών της, που πληκτρολογούνται στο Τερματικό του λειτουργικού συστήματος Linux, να εμφανίζει τις επιλεγμένες ρυθμίσεις της συνδεδεμένης φωτογραφικής μηχανής σε μορφή κειμένου στο συγκεκριμένο παράθυρο. Γνωρίζοντας τα παραπάνω, και κάνοντας χρήση της γλώσσας προγραμματισμού `php`, που μπορεί με την εντολή `exec` να στείλει στο Τερματικό μια οποιαδήποτε εντολή και να λάβει τα αποτελέσματα αυτής σε μορφή σειράς δεδομένων (`string`), είναι δυνατόν να ληφθούν απομακρυσμένα πληροφορίες για την κατάσταση της κάμερας. Για την εκτέλεση αυτών των εντολών θα προγραμματιστούν αντίστοιχες λειτουργίες JavaScript (`function`) που θα αναθέτουν ταυτόχρονα τις σειρές δεδομένων που λαμβάνονται στις σωστές ταυτότητες (`id`) για την εμφάνιση του στον πίνακα πληροφοριών και στην σελίδα `setting.html`.

Ο κώδικας που ακολουθεί δίνει την δυνατότητα να εμφανίζεται η επιλεγμένη λειτουργία έκθεσης της κάμερας και είναι ένα παράδειγμα όσων αναφέρθηκαν νωρίτερα. Αρχικά θα αναλυθούν οι προσθήκες που έγιναν στο αρχείο `service.php`, στο οποίο χρησιμοποιείται η γλώσσα προγραμματισμού `php`, και στην συνέχεια θα γίνει ανάλυση του κώδικα που θα προστεθεί στο αρχείο `controller.js`, που είναι γραμμένο σε γλώσσα JavaScript. Στο αρχείο `service.php` προστίθενται τα εξής μετά από την εντολή `switch($action){` :`

```
case "getexpmode":
```

```
    exec ("gphoto2 --get-config expprogram", $output);  
    $returnObj->camera = ($output[count($output) - 6]);  
    header('Content-Type: application/json');  
    echo json_encode($returnObj);  
    break;
```

Η εντολή **switch** που προηγείται του κώδικα που προστίθεται είναι παρόμοια λειτουργικά με μια σειρά εντολών IF. Σε πολλές περιπτώσεις, πρέπει να συγκριθεί η ίδια μεταβλητή ή έκφραση (expression) με πολλές διαφορετικές τιμές και να εκτελεστεί ένα διαφορετικό κομμάτι κώδικα ανάλογα με ποια τιμή ισούται. Αυτός ακριβώς είναι ο σκοπός της εντολής αυτής. Σε αυτό το κομμάτι γίνεται σύγκριση της τιμής `$action` με την περίπτωση (**case**) "getexpmode" και εφόσον είναι ίδιες εκτελείτε ο υπόλοιπος κώδικας. Η εντολή **exec** όπως αναφέρθηκε νωρίτερα εκτελεί μια συγκεκριμένη εντολή, εδώ την "gphoto2 --get-config expprogram" και ο πίνακας `$output` γεμίζει με κάθε γραμμή της εξόδου της εντολής. Στην επόμενη γραμμή αποθηκεύεται η έκτη από το τέλος σειρά του πίνακα `$output`, που περιέχει την επιλεγμένη λειτουργία έκθεσης της κάμερας, στο στοιχείο `$returnObj (= ($output[count($output) - 6])`). Το `$returnObj` ορίζεται ως δεδομένα του αντικειμένου (object) `camera` όταν αυτά εξαχθούν (`$returnObj->camera`). Η εντολή **header**('Content-Type: application /json') υποδηλώνει τον τρόπο που θα κωδικοποιηθούν τα παραπάνω δεδομένα (JSON σε αυτήν την περίπτωση) διότι ορισμένες φορές δεν είναι δυνατόν να εξαχθεί αυτό από τα ίδια τα δεδομένα. Με την εντολή **echo json_encode**(\$returnObj); γίνεται μετατροπή των δεδομένων `php` του `$returnObj` σε δεδομένα τύπου JSON και εξαγωγή τους στο αντικείμενο `camera`. Το αντικείμενο αυτό θα χρησιμοποιηθεί από την JavaScript για την εμφάνιση της επιλεγμένης λειτουργίας έκθεσης της κάμερας στην ιστοσελίδα (ο τρόπος θα αναλυθεί μαζί με τον κώδικα JavaScript στην επόμενη παράγραφο. Η εντολή **break** θα διακόψει την εκτέλεση της συγκεκριμένης περίπτωσης.

Στο αρχείο `controller.js` θα προτεθεί μια λειτουργίας (function) που με την χρήση του αντικειμένου `camera` θα εμφανίζει την επιλεγμένη λειτουργία έκθεσης της κάμερας στην ιστοσελίδα όπου καλείτε η ταυτότητα `expmode`. Μια λειτουργία JavaScript είναι ένα μέρος κώδικα (block of code) που έχει σχεδιαστεί για να εκτελεί μια συγκεκριμένη εργασία. Ο κώδικας αυτός εκτελείται όποτε επικαλείται το όνομα της λειτουργίας. Ορίζεται με τη λέξη κλειδί (keyword) **function** ακολουθούμενη από ένα όνομα, που ακολουθείτε από παρενθέσεις. Τα ονόματα λειτουργιών μπορούν να περιέχουν γράμματα, ψηφία, υπογράμμιση και σήμα του δολαρίου. Οι παρενθέσεις, που ακολουθούν του ονόματος, μπορεί να περιλαμβάνουν ονόματα παραμέτρων χωρισμένα με κόμμα π.χ. (parameter1, parameter2). Ο κώδικας της λειτουργίας που πρέπει να εκτελεστεί τοποθετείται μέσα σε αγκύλες. Η προσθήκη που θα γίνει στο `controller.js` θα είναι η εξής:

```
function getexpmode () {  
    $.ajax({  
        url: "service.php?action=getexpmode",  
        dataType : "json",  
        success: function(data) {  
            $("#expmode").html(data.camera);  
        }  
    });  
}
```

Η πρώτη σειρά δηλώνει το όνομα της λειτουργίας με τον τρόπο που περιγράφηκε στην προηγούμενη παράγραφο. Η εντολή που ακολουθεί `ajax` (Asynchronous JavaScript And XML) επιτρέπει στον χρήστη να διαβάσει και να στείλει δεδομένα στον διακομιστή. Σε αυτήν την περίπτωση με την εντολή `url: "service.php?action=getexpmode"` λαμβάνει τα δεδομένα από την περίπτωση που προστέθηκε στην προηγούμενη παράγραφο (`case "getexpmode"`). Τα δεδομένα αυτά (του αντικειμένου `camera`) ορίζονται ως τύπου JSON με την `dataType : "json"` και εφόσον αναγνωριστούν με επιτυχία θα εμφανίζονται όπου καλείτε η ταυτότητα `expmode`.

Παρόμοιος είναι ο κώδικας για τις υπόλοιπες εντολές ανάκτησης των ρυθμίσεων της φωτογραφικής μηχανής με εξαίρεση ορισμένες απαραίτητες τροποποιήσεις για την σωστή λειτουργία τους.

Στο αρχείο `service.php` για τις υπόλοιπες περιπτώσεις θα γίνει αλλαγή της σειράς:

```
$returnObj->camera = ($output [count($output) - 6]); με τις σειρές :  
$routput= array_reverse($output);  
$returnObj->camera = ($routput[count($routput) - 4]);
```

Ο λόγος που θα γίνει αυτό είναι ότι αναστρέφοντας τον πίνακα `$output` με την εντολή `$routput= array_reverse($output);` η γραμμή όπου αποθηκεύεται η εκάστοτε ρύθμιση παραμένει πάντοτε η ίδια (τέταρτη από το τέλος) ανεξαρτήτως του μεγέθους του κάθε πίνακα οπότε μπορούν να χρησιμοποιηθούν οι ίδιες εντολές για την ανάκτηση των πληροφοριών.

Η άλλη τροποποίηση που θα πραγματοποιηθεί αφορά τα ονόματα των καταστάσεων, λειτουργιών, ταυτοτήτων και τις εντολές που εκτελούνται σε κάθε περίπτωση.

Συγκεκριμένα ο κώδικας για την εμφάνιση του επιλεγμένου μεγέθους διαφράγματος είναι ο εξής:

Στο `service.php`:

```
case "apsize":  
exec ("gphoto2 --get-config f-number", $output);  
$routput= array_reverse($output);  
$returnObj->camera = ($routput[count($routput) - 4]);  
header('Content-Type: application/json');  
echo json_encode($returnObj);  
break;
```

Στο `controller.js`:

```
function apsize() {  
$.ajax({  
url: "service.php?action=apsize",  
dataType : "json",  
success: function(data) {  
$("#apsize").html(data.camera);  
},  
});  
}
```

Ο κώδικας για την εμφάνιση της επιλεγμένης ταχύτητας κλείστρου είναι:

Στο `service.php`:

```
case "getshutterspeed":  
exec ("gphoto2 --get-config shutterspeed2", $output);  
$routput= array_reverse($output);  
$returnObj->camera = ($routput[count($routput) - 4]);  
header('Content-Type: application/json');  
echo json_encode($returnObj);  
break;
```


Στο controller.js:

```
function getshutterspeed(){
$.ajax({
url: "service.php?action=getshutterspeed",
dataType : "json",
success: function(data){
$("#shutterspeed").html(data.camera);
},
});
}
```

Ο κώδικας για την εμφάνιση της επιλογής Auto ISO είναι:

Στο service.php:

```
case "getautoiso":
exec ("gphoto2 --get-config autoiso", $output);
$output= array_reverse($output);
$returnObj->camera = ($output[count($output) - 4]);
header('Content-Type: application/json');
echo json_encode($returnObj);
break;
```

Στο controller.js:

```
function getautoiso(){
$.ajax({
url: "service.php?action=getautoiso",
dataType : "json",
success: function(data){
$("#autoiso").html(data.camera);
},
});
}
```

Ο κώδικας για την εμφάνιση της επιλεγμένης τιμής ISO είναι:

Στο service.php:

```
case "getiso":
exec ("gphoto2 --get-config iso", $output);
$output= array_reverse($output);
$returnObj->camera = ($output[count($output) - 4]);
header('Content-Type: application/json');
echo json_encode($returnObj);
break;
```

Στο controller.js:

```
function getiso(){
$.ajax({
url: "service.php?action=getiso",
dataType : "json",
success: function(data){
$("#iso").html(data.camera);
},
});
}
```

Ο κώδικας για την εμφάνιση της επιλεγμένης τιμής αντιστάθμιση έκθεσης είναι:

Στο service.php:

```
case "getexpcomp":
exec ("gphoto2 --get-config exposurecompensation",
$output);
$output= array_reverse($output);
```

```

$returnObj->camera = ($routput[count($routput) - 4]);
header('Content-Type: application/json');
echo json_encode($returnObj);
break;

```

Στο controller.js:

```

function getexpcomp(){
$.ajax({
url: "service.php?action=getexpcomp",
dataType : "json",
success: function(data){
$("#expcomp").html(data.camera);
},
});
}

```

Ο κώδικας για την εμφάνιση της επιλεγμένης τιμής της μετατόπισης του ευέλικτου προγράμματος είναι:

Στο service.php:

```

case "getflex":
exec ("gphoto2 --get-config flexibleprogram", $output);
$routput= array_reverse($output);
$returnObj->camera = ($routput[count($routput) - 4]);
header('Content-Type: application/json');
echo json_encode($returnObj);
break;

```

Στο controller.js:

```

function getflex(){
$.ajax({
url: "service.php?action=getflex",
dataType : "json",
success: function(data){
$("#flex").html(data.camera);
},
});
}

```

Ο κώδικας για την εμφάνιση της επιλεγμένης τιμής της ποιότητας εικόνας είναι:

Στο service.php:

```

case "getquality":
exec ("gphoto2 --get-config imagequality", $output);
$routput= array_reverse($output);
$returnObj->camera = ($routput[count($routput) - 4]);
header('Content-Type: application/json');
echo json_encode($returnObj);
break;

```

Στο controller.js:

```

function getquality(){
$.ajax({
url: "service.php?action=getquality",
dataType : "json",
success: function(data){
$("#quality").html(data.camera);
},
});
}

```

Ο κώδικας για την εμφάνιση της επιλεγμένης τιμής του μεγέθους εικόνας είναι:

Στο service.php:

```
case "getsize":  
exec ("gphoto2 --get-config imagesize", $output);  
$routput= array_reverse($output);  
$returnObj->camera = ($routput[count($routput) - 4]);  
header('Content-Type: application/json');  
echo json_encode($returnObj);  
break;
```

Στο controller.js:

```
function getsize(){  
$.ajax({  
url: "service.php?action=getsize",  
dataType : "json",  
success: function(data){  
$("#size").html(data.camera);  
},  
});  
}
```

Ο κώδικας για την εμφάνιση της επιλεγμένης τιμής της συμπίεσης των αρχείων τύπου RAW είναι:

Στο service.php:

```
case "getrsize":  
exec ("gphoto2 --get-config rawcompression", $output);  
$routput= array_reverse($output);  
$returnObj->camera = ($routput[count($routput) - 4]);  
header('Content-Type: application/json');  
echo json_encode($returnObj);  
break;
```

Στο controller.js:

```
function getrsize(){  
$.ajax({  
url: "service.php?action=getrsize",  
dataType : "json",  
success: function(data){  
$("#rsize").html(data.camera);  
},  
});  
}
```

Ο κώδικας για την εμφάνιση της επιλεγμένης τιμής της ισορροπίας λευκού είναι:

Στο service.php:

```
case "getwbalance":  
exec ("gphoto2 --get-config whitebalance", $output);  
$routput= array_reverse($output);  
$returnObj->camera = ($routput[count($routput) - 4]);  
header('Content-Type: application/json');  
echo json_encode($returnObj);  
break;
```

Στο controller.js:

```
function getwbalance(){  
$.ajax({  
url: "service.php?action=getwbalance",  
dataType : "json",  
success: function(data){  
$("#wbalance").html(data.camera);  
}
```

```

    },
  });
}

```

Ο κώδικας για την εμφάνιση της επιλεγμένης τιμής του χρωματικού χώρου είναι:
Στο service.php:

```

case "getcolorspace":
exec ("gphoto2 --get-config colorpace", $output);
$output= array_reverse($output);
$returnObj->camera = ($output[count($output) - 4]);
header('Content-Type: application/json');
echo json_encode($returnObj);
break;

```

Στο controller.js:

```

function getcolorspace(){
$.ajax({
url: "service.php?action=getcolorspace",
dataType : "json",
success: function(data){
$("#colorspace").html(data.camera);
},
});
}

```

Ο κώδικας για την εμφάνιση της επιλεγμένης τιμής αποθορυβοποίησης υψηλών τιμών ISO είναι:

Στο service.php:

```

case "gethisonr":
exec ("gphoto2 --get-config highisonr", $output);
$output= array_reverse($output);
$returnObj->camera = ($output[count($output) - 4]);
header('Content-Type: application/json');
echo json_encode($returnObj);
break;

```

Στο controller.js:

```

function gethisonr(){
$.ajax({
url: "service.php?action=gethisonr",
dataType : "json",
success: function(data){
$("#isonr").html(data.camera);
},
});
}

```

Ο κώδικας για την εμφάνιση της επιλεγμένης τιμής αποθορυβοποίησης έκθεσης μακράς διάρκειας είναι:

Στο service.php:

```

case "getlnr":
exec ("gphoto2 --get-config longexpnr", $output);
$output= array_reverse($output);
$returnObj->camera = ($output[count($output) - 4]);
header('Content-Type: application/json');
echo json_encode($returnObj);
break;

```

Στο controller.js:

```

function getlnr(){

```

```

$.ajax({
  url: "service.php?action=getlnr",
  dataType : "json",
  success: function(data) {
    $("#lnr").html(data.camera);
  },
});
}

```

Ο κώδικας για την εμφάνιση της επιλεγμένης τιμής λειτουργίας φλας είναι:
Στο service.php:

```

case "getflashmode":
exec ("gphoto2 --get-config flashmode", $output);
$output= array_reverse($output);
$returnObj->camera = ($output[count($output) - 4]);
header('Content-Type: application/json');
echo json_encode($returnObj);
break;

```

Στο controller.js:

```

function getflashmode() {
$.ajax({
  url: "service.php?action=getflashmode",
  dataType : "json",
  success: function(data) {
    $("#flashmode").html(data.camera);
  },
});
}

```

Ο κώδικας για την εμφάνιση της επιλεγμένης τιμής ταχύτητα συγχρονισμού φλας είναι:

Στο service.php:

```

case "getflashsyncspeed":
exec ("gphoto2 --get-config flashsyncspeed", $output);
$output= array_reverse($output);
$returnObj->camera = ($output[count($output) - 4]);
header('Content-Type: application/json');
echo json_encode($returnObj);
break;

```

Στο controller.js:

```

function getflashsyncspeed() {
$.ajax({
  url: "service.php?action=getflashsyncspeed",
  dataType : "json",
  success: function(data) {
    $("#flashsyncspeed").html(data.camera);
  },
});
}

```

Ο κώδικας για την εμφάνιση της επιλεγμένης τιμής ταχύτητας κλείστρου φλας είναι:
Στο service.php:

```

case "getflashshutterspeed":
exec ("gphoto2 --get-config flashshutterspeed", $output);
$output= array_reverse($output);
$returnObj->camera = ($output[count($output) - 4]);
header('Content-Type: application/json');

```

```

        echo json_encode($returnObj);
        break;

```

Στο controller.js:

```

function getflashshutterspeed() {
$.ajax({
url: "service.php?action=getflashshutterspeed",
dataType : "json",
success: function(data) {
$("#flashshutterspeed").html(data.camera);
},
});
}

```

Για να εκτελούνται οι παραπάνω λειτουργίες κάθε φορά που φορτώνει η ιστοσελίδα θα προστεθεί στο controller.js το εξής:

```

$(document).on( 'pageinit',function(event) {
getCamera();
getexpmode();
apsize();
getshutterspeed();
getautoiso();
getiso();
getexpcomp();
getflex();
getquality();
getsize();
getrsize();
getwbalance();
getcolorspace();
gethisonr();
getlnr();
getflashmode();
getflashsyncspeed();
getflashshutterspeed();
});

```

Name	Current Selection
Exposure Mode	Current: P
Aperture Size	Current: f/4
Shutter Speed	Current: 1/5
Auto ISO	Current: Off
ISO	Current: 200
Exposure Compensation	Current: 0
Flexible Program Shift	Current: 0
Image Quality	Current: JPEG Fine
Image Size	Current: 4256x2832
RAW Compression	Current: Lossless
White Balance	Current: Automatic
Color Space	Current: sRGB
High ISO Noise Reduction	Current: Normal
Long Exposure Noise Reduction	Current: Off
Flash Mode	Current: Red-eye automatic
Flash Sync Speed	Current: 1/250s
Flash Shutter Speed	Current: 1/60s

Εικόνα 7: Ο πίνακας με τις τρέχων ρυθμίσεις της φωτογραφικής μηχανής

2.2.7: Δημιουργία ενός βασικού πτυσσόμενου μενού

Οι αλλαγές των ρυθμίσεων της φωτογραφικής μηχανής που θα γίνονται μέσω της ιστοσελίδας settings.html θα βρίσκονται μέσα σε πτυσσόμενα μενού ώστε να περιοριστεί το μέγεθος της σελίδας. Τα μενού αυτά θα σχηματιστούν με την χρήση γλώσσας σήμανσης HTML, την γλώσσα φύλλων στυλ CSS και την γλώσσα προγραμματισμού.

Αρχικά μετά από την ετικέτα κλεισίματος `</div>` της κεφαλίδας της σελίδας θα προστεθεί ο εξής κώδικας HTML:

```
<button onclick="drop('myDropdown1')" class="dropbtn">
Dropdown</button>
<div id="myDropdown1" class="dropdown-content">
  <input type="button" onclick=action value='Option 1'></a>
  <input type="button" onclick=action value='Option 2'></a>
  <input type="button" onclick=action value='Option 3'></a></div>
```

Ένα στοιχείο `<button>` χρησιμοποιείτε για να ανοίγει και να κλείνει το μενού. Η ετικέτα `<div>` περιέχει το ίδιο μενού, την ταυτότητα του, την κλάση του και τις επιλογές του (π.χ. `<input type="button" onclick=action value='Option 1'>`).

Στην συνέχεια θα συμπληρωθεί στην ετικέτα `<style>` του ίδιου αρχείου ο παρακάτω κώδικας:

```
.dropbtn {
  color: black;
  padding: 16px;
  font-size: 16px;
  border: none;
  cursor: pointer;
}
.dropbtn:hover, .dropbtn:focus {
  color: white;
  background-color: #2980B9;
}
.dropdown {
  position: relative;
  display: inline-block;
}
.dropdown-content {
  display: none;
  background-color: #f1f1f1;
  min-width: 160px;
  overflow: auto;
  box-shadow: 0px 8px 16px 0px rgba(0,0,0,0.2);
  z-index: 1;
}
.dropdown-content a {
  color: black;
  padding: 12px 16px;
  text-decoration: none;
  display: block;
}
.dropdown a:hover {background-color: #ddd;}
.show {display: block;}
```

Οι κλάσεις `dropbtn` (του κουμπιού του μενού), `dropbtn: hover`, `.dropbtn:focus` (του ίδιου κουμπιού όταν βρίσκεται πάνω του ο δείκτης του ποντικιού ή είναι σε εστίαση), `dropdown-content a` (του περιεχομένου των μενού) και `dropdown a:hover` (του

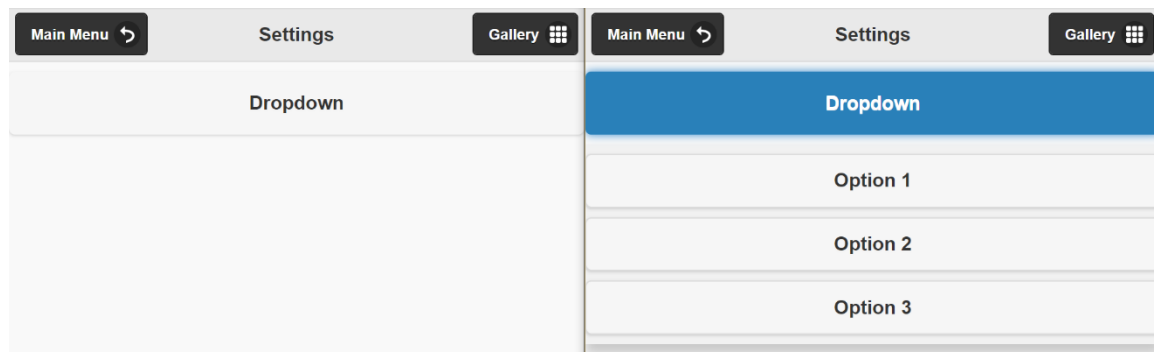
περιεχομένου των μενού όταν βρίσκεται πάνω του ο δείκτης του ποντικιού) περιέχουν τις πληροφορίες εμφάνισης του κάθε στοιχείου. Η κλάση `.dropdown` χρησιμοποιεί τη `position: relative` για να τοποθετηθεί το αναπτυσσόμενο περιεχόμενο ακριβώς κάτω από το κουμπί. Η κλάση `.dropdown-content` περιέχει την εμφάνιση του πραγματικού αναπτυσσόμενου μενού. Είναι κρυμμένο από προεπιλογή και θα εμφανιστεί στην οθόνη όταν γίνει κλικ στο κουμπί. Το ελάχιστο πλάτος έχει οριστεί σε 160px. Η εντολή `overflow: auto`; ενεργοποιεί την κύλιση σε μικρές οθόνες. Αντί περιγράμματος με την ιδιότητα `box-shadow` το αναπτυσσόμενο μενού μοιάζει σαν μια "κάρτα". Το `z-index` τοποθετεί το αναπτυσσόμενο μενού μπροστά από άλλα στοιχεία.

Στο τέλος του αρχείου θα προστεθεί ο απαραίτητος κώδικας JavaScript μέσα στην ετικέτα `<script>`:

```
function drop(idOfTheDiv) {
    document.getElementById(idOfTheDiv).classList.toggle("show");
}
window.onclick = function(event) {
    if (!event.target.matches('.dropbtn')) {

        var dropdowns = document.getElementsByClassName("dropdown-
content");
        var i;
        for (i = 0; i < dropdowns.length; i++) {
            var openDropdown = dropdowns[i];
            if (openDropdown.classList.contains('show')) {
                openDropdown.classList.remove('show');
            }
        }
    }
}
```

Η λειτουργία `drop` εμφανίζει τα περιεχόμενα του μενού όταν γίνει ένα κλικ στην περιοχή του κουμπιού (`if (!event.target.matches('.dropbtn'))`) και κλείνει όποτε γίνει κλικ εκτός του κουμπιού.



Εικόνα 8: Το πτυσσόμενο μενού κλειστό και ανοιχτό αντίστοιχα

2.2.8: Προγραμματισμός των εντολών απομακρυσμένης αλλαγής ρυθμίσεων της κάμερας

Η διαδικασία με την οποία θα σταλούν οι εντολές αλλαγής ρυθμίσεων της κάμερας είναι παρόμοια με αυτήν που λαμβάνονται οι επιλεγμένες ρυθμίσεις της φωτογραφικής

μηχανής. Δηλαδή θα προγραμματιστούν καταστάσεις σε γλώσσα php οι οποίες θα στέλνουν τις αντίστοιχες εντολές gphoto2 στο Τερματικό Linux. Οι καταστάσεις αυτές θα ενεργοποιούνται όταν καλούνται συγκεκριμένες λειτουργίες JavaScript.

Η κύρια διαφορά μεταξύ των δυο περιπτώσεων βρίσκεται στο ότι όταν στέλνονται εντολές για να αλλάζουν οι ρυθμίσεις της κάμερας δεν χρειάζεται να στέλνεται πίσω στην ιστοσελίδα τα αποτελέσματα της εκτέλεσης μιας εντολής όποτε δεν χρειάζονται οι αντίστοιχες εντολές. Παραδείγματος χάριν για να γίνει η αλλαγή της λειτουργία έκθεσης της κάμερας σε λειτουργία προγράμματος (program mode) θα γίνουν οι εξής προσθήκες στα αρχεία service.php και controller.js:

Στο service.php:

```
case "program":
    exec ("gphoto2 --set-config expprogram=P", $output);
    echo json_encode(true);
    break;
```

Στο controller.js:

```
function program() {
$.ajax({
    url: "service.php?action=program",
    dataType : "json",
    success: function(data) {
        $.mobile.loading( 'hide' );
    },
});
}
```

Οι άλλες επιλογές της λειτουργίας έκθεσης καθώς επίσης και οι υπόλοιπες ρυθμίσεις της κάμερας έχουν παραπλήσιες εντολές με το προηγούμενο παράδειγμα. Οι μόνες αλλαγές που θα γίνουν αφορούν τα ονόματα των καταστάσεων, των λειτουργιών, την εκάστοτε εντολή της gphoto2 που εκτελείτε και την θέση της αντίστοιχης της κατάστασης που καλείτε. Για αυτόν τον λόγο στην συνέχεια θα δοθεί μόνο ένα παράδειγμα μίας επιλογής από την κάθε ρύθμιση που μπορεί να γίνει (ο αναλυτικός κώδικας όλων των πιθανών ρυθμίσεων θα υπάρχει στο παράρτημα της εργασίας αυτής).

Ο κώδικας για την επιλογή f/2.8 του μεγέθους διαφράγματος είναι ο εξής:

Στο service.php:

```
case "f2.8":
    exec ("gphoto2 --set-config f-number=f/2.8", $output);
    echo json_encode(true);
    break;
```

Στο controller.js:

```
function f2_8() {
$.ajax({
    url: "service.php?action=f2.8",
    dataType : "json",
    success: function(data) {
        $.mobile.loading( 'hide' );
    },
});
}
```

Οι δυνατές τιμές του διαφράγματος εξαρτώνται από τον φακό που χρησιμοποιεί η μηχανή για αυτό το λόγο μερικές τιμές θα λειτουργούν μόνο με ορισμένους φακούς. Η τιμή του διαφράγματος είναι δυνατόν να ρυθμιστεί μόνο όταν η λειτουργία έκθεσης της κάμερας είναι σε αυτόματη προτεραιότητα διαφράγματος ή σε χειροκίνητη λειτουργία.

Ο κώδικας για την επιλογή ταχύτητας κλείστρου 1/8000s είναι:

Στο service.php:

```
case "s1_8000":
    exec ("gphoto2 --set-config shutterspeed2=1/8000", $output);
    echo json_encode(true);
    break;
```

Στο controller.js:

```
function s1_8000(){
    $.ajax({
        url: "service.php?action=s1_8000",
        dataType : "json",
        success: function(data){
            $.mobile.loading( 'hide');
        },
    });
}
```

Η τιμή της ταχύτητας του κλείστρου είναι δυνατόν να ρυθμιστεί μόνο όταν η λειτουργία έκθεσης της κάμερας είναι σε αυτόματη προτεραιότητα κλείστρου ή χειροκίνητη λειτουργία.

Ο κώδικας για την ενεργοποίηση ή όχι της αυτόματης λειτουργίας ISO είναι:

Στο service.php:

```
case "autoiso_on":
    exec ("gphoto2 --set-config autoiso=0", $output);
    echo json_encode(true);
    break;
case "autoiso_off":
    exec ("gphoto2 --set-config autoiso=1", $output);
    echo json_encode(true);
    break;
```

Στο controller.js:

```
function autoiso_on(){
    $.ajax({
        url: "service.php?action=autoiso_on",
        dataType : "json",
        success: function(data){
            $.mobile.loading( 'hide');
        },
    });
}
function autoiso_off(){
    $.ajax({
        url: "service.php?action=autoiso_off",
        dataType : "json",
        success: function(data){
            $.mobile.loading( 'hide');
        },
    });
}
```

Με την ενεργοποίηση της αυτόματης λειτουργίας η τιμή του ISO που λαμβάνει η gphoto2 παραμένει η τελευταία χειροκίνητα επιλεγμένη τιμή.

Ο κώδικας για την επιλογή τιμής ISO 100 είναι:

Στο service.php:

```
case "iso_100":
    exec ("gphoto2 --set-config iso=100", $output);
    echo json_encode(true);
```

```
break;
```

Στο controller.js:

```
function iso_100(){
$.ajax({
url: "service.php?action=iso_100",
dataType : "json",
success: function(data){
$.mobile.loading( 'hide');
},
});
}
```

Η τιμή ISO είναι δυνατόν να ρυθμιστεί μόνο όταν η λειτουργία Auto ISO είναι απενεργοποιημένη.

Ο κώδικας για την επιλογή τιμής αντιστάθμιση έκθεσης στο 5 είναι:

Στο service.php:

```
case "comp_50":
exec ("gphoto2 --set-config
exposurecompensation=5",$output);
echo json_encode(true);
break;
```

Στο controller.js:

```
function comp_50(){
$.ajax({
url: "service.php?action=comp_50",
dataType : "json",
success: function(data){
$.mobile.loading( 'hide');
},
});
}
```

Ο κώδικας για την επιλογή της τιμής μετατόπισης του ευέλικτου προγράμματος στο 30 είναι:

Στο service.php:

```
case "shift_30":
exec ("gphoto2 --set-config flexibleprogram=30",$output);
echo json_encode(true);
break;
```

Στο controller.js:

```
function shift_30(){
$.ajax({
url: "service.php?action=shift_30",
dataType : "json",
success: function(data){
$.mobile.loading( 'hide');
},
});
}
```

Η τιμή μετατόπισης του ευέλικτου προγράμματος μπορεί να ρυθμιστεί μόνο όταν η λειτουργία έκθεσης της κάμερας είναι σε αυτόματο πρόγραμμα.

Ο κώδικας για την επιλογή της ποιότητας εικόνας σε βασική είναι:

Στο service.php:

```
case "basic":
exec ("gphoto2 --set-config imagequality=0",$output);
echo json_encode(true);
```

```
break;
```

Στο controller.js:

```
function basic(){
$.ajax({
url: "service.php?action=basic",
dataType : "json",
success: function(data){
$.mobile.loading( 'hide');
},
});
}
```

Η gphoto2 δεν αναγνωρίζει την επιλογή 'TIFF' της παραπάνω ρύθμισης όταν επιλέγεται. Θα προγραμματιστεί μια λειτουργία JavaScript ώστε να εμφανίζεται σωστά. Η λειτουργία αυτή θα προστεθεί στο settings.html μετά την λειτουργία drop μέσα στην ετικέτα <script>. Ο κώδικας της είναι:

```
function ltiff(){
document.getElementById('quality').innerHTML = 'Current: TIFF';
}
```

Ο κώδικας για την αλλαγή του μεγέθους εικόνας σε μεγάλο είναι:

Στο service.php:

```
case "large":
exec ("gphoto2 --set-config imagesize=0", $output);
echo json_encode(true);
break;
```

Στο controller.js:

```
function large(){
$.ajax({
url: "service.php?action=large",
dataType : "json",
success: function(data){
$.mobile.loading( 'hide');
},
});
}
```

Ο κώδικας για την αλλαγή της συμπίεσης των αρχείων RAW σε τύπο lossless είναι:

Στο service.php:

```
case "lossless":
exec ("gphoto2 --set-config rawcompression=0", $output);
echo json_encode(true);
break;
```

Στο controller.js:

```
function lossless(){
$.ajax({
url: "service.php?action=lossless",
dataType : "json",
success: function(data){
$.mobile.loading( 'hide');
},
});
}
```

Παρομοίως με την επιλογή 'TIFF' παραπάνω η gphoto2 δεν αναγνωρίζει την επιλογή 'Uncompressed' όταν επιλέγεται. Για αυτό, μετά την λειτουργία ltiff μέσα στην ετικέτα

<script>, θα προγραμματιστεί μια λειτουργία JavaScript που θα την εμφανίζει σωστά. Ο κώδικας της είναι:

```
function sizeoff(){
    document.getElementById('rsize').innerHTML = 'Current:
Uncompressed';
}
```

Ο κώδικας για την αλλαγή της ισορροπίας λευκού σε αυτόματη είναι:

Στο service.php:

```
case "wauto":
    exec ("gphoto2 --set-config whitebalance=0", $output);
    echo json_encode(true);
    break;
```

Στο controller.js:

```
function wauto(){
$.ajax({
    url: "service.php?action=wauto",
    dataType : "json",
    success: function(data){
        $.mobile.loading( 'hide');
    },
});
}
```

Ο κώδικας για την αλλαγή τιμής του χρωματικού χώρου σε sRGB είναι:

Στο service.php:

```
case "srgb":
    exec ("gphoto2 --set-config colorspace=0", $output);
    echo json_encode(true);
    break;
```

Στο controller.js:

```
function srgb(){
$.ajax({
    url: "service.php?action=srgb",
    dataType : "json",
    success: function(data){
        $.mobile.loading( 'hide');
    },
});
}
```

Ο κώδικας για την αλλαγή της αποθρομβοποίησης υψηλών τιμών ISO σε OFF είναι:

Στο service.php:

```
case "hnroff":
    exec ("gphoto2 --set-config highisonr=0", $output);
    echo json_encode(true);
    break;
```

Στο controller.js:

```
function hnroff(){
$.ajax({
    url: "service.php?action=hnroff",
    dataType : "json",
    success: function(data){
        $.mobile.loading( 'hide');
    },
});
}
```

Ο κώδικας για την ενεργοποίηση ή όχι της αποθρομβοποίησης της έκθεσης μακράς διάρκειας είναι:

Στο service.php:

```
case "lnron":
    exec ("gphoto2 --set-config longexpnr=0", $output);
    echo json_encode(true);
    break;
case "lnroff":
    exec ("gphoto2 --set-config longexpnr=1", $output);
    echo json_encode(true);
    break;
```

Στο controller.js:

```
function lnron(){
$.ajax({
  url: "service.php?action=lnron",
  dataType : "json",
  success: function(data){
    $.mobile.loading( 'hide');
  },
});
}

function lnroff(){
$.ajax({
  url: "service.php?action=lnroff",
  dataType : "json",
  success: function(data){
    $.mobile.loading( 'hide');
  },
});
}
```

Ο κώδικας αλλαγής της επιλεγμένης λειτουργίας φλας σε αυτόματη είναι:

Στο service.php:

```
case "flauto":
    exec ("gphoto2 --set-config flashmode=1", $output);
    echo json_encode(true);
    break;
```

Στο controller.js:

```
function flauto(){
$.ajax({
  url: "service.php?action=flauto",
  dataType : "json",
  success: function(data){
    $.mobile.loading( 'hide');
  },
});
}
```

Ο κώδικας αλλαγής της επιλεγμένης ταχύτητα συγχρονισμού φλας σε 1/250s είναι:

Στο service.php:

```
case "f1_250":
    exec ("gphoto2 --set-config flashsyncspeed=1", $output);
    echo json_encode(true);
    break;
```

Στο controller.js:

```
function f1_250(){
$.ajax({
```

```

        url: "service.php?action=fs1_250",
        dataType : "json",
        success: function(data){
            $.mobile.loading( 'hide' );
        },
    });
}

```

Ο κώδικας αλλαγής της επιλεγμένης ταχύτητας κλείστρου φλας σε 1/60s είναι:

Στο service.php:

```

    case "fs1_60":
        exec ("gphoto2 --set-config flashshutterspeed=0", $output);
        echo json_encode(true);
        break;

```

Στο controller.js:

```

    function fs1_60 () {
        $.ajax({
            url: "service.php?action=fs1_60",
            dataType : "json",
            success: function(data){
                $.mobile.loading( 'hide' );
            },
        });
    }
}

```

2.2.9: Συμπλήρωση των πτυσσόμενων μενού με τις εντολές απομακρυσμένου ελέγχου της φωτογραφικής μηχανής

Με την ολοκλήρωση του προγραμματισμού των εντολών απομακρυσμένης αλλαγής ρυθμίσεων της κάμερας το μόνο βήμα που χρειάζεται για να ολοκληρωθεί ο απομακρυσμένος έλεγχος της φωτογραφικής μηχανής είναι να ενσωματωθούν οι εντολές αυτές σε πτυσσόμενα μενού παρόμοια με αυτό που περιεγράφηκε δύο κεφάλαια πριν.

Στην ετικέτα <body> ιστοσελίδα settings.html, για να γίνει πιο φιλική προς τον χρήστη, θα τοποθετηθούν επιπλέον πάνω από τα αντίστοιχα μενού δυο κεφαλίδες. Οι κεφαλίδες θα αναγράφουν την λειτουργία του μενού που ακολουθεί (κεφαλίδα <h1>) και την τρέχουσα επιλεγμένη τιμή της ρύθμισης (κεφαλίδα <h2>). Ακόμη, επειδή ορισμένες ρυθμίσεις εκτός από τον εαυτό τους επηρεάζουν και άλλες τιμές όταν αλλάζουν, για είναι όλες οι ενδείξεις της ιστοσελίδας σωστές, θα εκτελούνται μαζί τους και λειτουργίες που θα ενημερώνουν την ιστοσελίδα για υπόλοιπες αλλαγές που έγιναν.

Σύμφωνα με τα παραπάνω ο κώδικας του μενού αλλαγής της λειτουργία έκθεσης είναι:

```

<h1> Exposure Mode </h1>
<h2 id="expmode"> </h2>
<button onclick="drop('myDropdown1')" class="dropbtn"> Select
Exposure Mode</button>
<div id="myDropdown1" class="dropdown-content">
<input type="button"
onclick="program();getexpmode();apsize();getshutterspeed()";
value=Program></a>
<input type="button"
onclick="shutter();getexpmode();apsize();getshutterspeed()";
value=Shutter priority></a>

```

```

        <input type="button"
onclick="aperture () ;getexpmode () ;apsize () ;getshutterspeed () ";
value=Aperture priority></a>
        <input type="button"
onclick="manual () ;getexpmode () ;apsize () ;getshutterspeed () ";
value=Manual></a></div>

```

Με την αλλαγή της λειτουργίας έκθεσης αλλάζει επιπλέον η ταχύτητα του κλείστρου και το μέγεθος του διαφράγματος για αυτό και εκτελούνται οι λειτουργίες `apsize ()` και `getshutterspeed ()` .

Ο κώδικας του μενού αλλαγής του μεγέθους διαφράγματος είναι:

```

<h1> Aperture Size </h1>
  <h2 id="apsize"> </h2>
    <button onclick="drop ('myDropdown2') " class="dropbtn"> Select
Aperture Size</button>
    <div id="myDropdown2" class="dropdown-content">
      <input type="button"
onclick="f2_8 () ;apsize () ;getshutterspeed () ;"; value=F/2.8></a>
      <input type="button"
onclick="f3_2 () ;apsize () ;getshutterspeed () ;"; value=F/3.2></a>
      <input type="button"
onclick="f3_5 () ;apsize () ;getshutterspeed () ;"; value=F3.5></a>
      <input type="button"
onclick="f4 () ;apsize () ;getshutterspeed () ;"; value=F/4></a>
      <input type="button"
onclick="f4_5 () ;apsize () ;getshutterspeed () ;"; value=F/4.5></a>
      <input type="button"
onclick="f5 () ;apsize () ;getshutterspeed () ;"; value=F5></a>
      <input type="button"
onclick="f5_6 () ;apsize () ;getshutterspeed () ;"; value=F/5.6></a>
      <input type="button"
onclick="f6_3 () ;apsize () ;getshutterspeed () ;"; value=F/6.3></a>
      <input type="button"
onclick="f7_1 () ;apsize () ;getshutterspeed () ;"; value=F7.1></a>
      <input type="button"
onclick="f8 () ;apsize () ;getshutterspeed () ;"; value=F/8></a>
      <input type="button"
onclick="f9 () ;apsize () ;getshutterspeed () ;"; value=F/9></a>
      <input type="button"
onclick="f10 () ;apsize () ;getshutterspeed () ;"; value=F10></a>
      <input type="button" onclick="f11 () ;apsize () ;getshutterspeed () ;";
value=F/11></a>
      <input type="button"
onclick="f13 () ;apsize () ;getshutterspeed () ;"; value=F/13></a>
      <input type="button"
onclick="f14 () ;apsize () ;getshutterspeed () ;"; value=F/14></a>
      <input type="button"
onclick="f16 () ;apsize () ;getshutterspeed () ;"; value=F/16></a>
      <input type="button"
onclick="f18 () ;apsize () ;getshutterspeed () ;"; value=F/18></a>
      <input type="button"
onclick="f20 () ;apsize () ;getshutterspeed () ;"; value=F/20></a>
      <input type="button"
onclick="f22 () ;apsize () ;getshutterspeed () ;"; value=F/22></a>
      <input type="button"
onclick="f25 () ;apsize () ;getshutterspeed () ;"; value=F/25></a>

```



```

        <input type="button"
onclick="f29 ();apsize ();getshutterspeed ();"; value=F/29></a>
        <input type="button"
onclick="f32 ();apsize ();getshutterspeed ();"; value=F/32></a></div>

```

Με την αλλαγή του μεγέθους διαφράγματος αλλάζει και η ταχύτητα του κλείστρου για αυτό και εκτελείτε η λειτουργία `getshutterspeed ()` .

Ο κώδικας του μενού αλλαγής της ταχύτητας του κλείστρου είναι:

```

<h1> Shutter Speed </h1>
    <h2 id="shutterspeed"> </h2>
    <button onclick="drop ('myDropdown3') " class="dropbtn"> Select
Shutter Speed</button>
    <div id="myDropdown3" class="dropdown-content">
        <input type="button"
onclick="s1_8000 ();apsize ();getshutterspeed ();"; value=1/8000></a>
        <input type="button"
onclick="s1_6400 ();apsize ();getshutterspeed ();"; value=1/6400></a>
        <input type="button"
onclick="s1_5000 ();apsize ();getshutterspeed ();"; value=1/5000></a>
        <input type="button"
onclick="s1_4000 ();apsize ();getshutterspeed ();"; value=1/4000></a>
        <input type="button"
onclick="s1_3200 ();apsize ();getshutterspeed ();"; value=1/3200></a>
        <input type="button"
onclick="s1_2500 ();apsize ();getshutterspeed ();"; value=1/2500></a>
        <input type="button"
onclick="s1_2000 ();apsize ();getshutterspeed ();"; value=1/2000></a>
        <input type="button"
onclick="s1_1600 ();apsize ();getshutterspeed ();"; value=1/1600></a>
        <input type="button"
onclick="s1_1250 ();apsize ();getshutterspeed ();"; value=1/1250></a>
        <input type="button"
onclick="s1_1000 ();apsize ();getshutterspeed ();"; value=1/1000></a>
        <input type="button"
onclick="s1_800 ();apsize ();getshutterspeed ();"; value=1/800></a>
        <input type="button"
onclick="s1_640 ();apsize ();getshutterspeed ();"; value=1/640></a>
        <input type="button"
onclick="s1_500 ();apsize ();getshutterspeed ();"; value=1/500></a>
        <input type="button"
onclick="s1_400 ();apsize ();getshutterspeed ();"; value=1/400></a>
        <input type="button"
onclick="s1_320 ();apsize ();getshutterspeed ();"; value=1/320></a>
        <input type="button"
onclick="s1_250 ();apsize ();getshutterspeed ();"; value=1/250></a>
        <input type="button"
onclick="s1_200 ();apsize ();getshutterspeed ();"; value=1/200></a>
        <input type="button"
onclick="s1_160 ();apsize ();getshutterspeed ();"; value=1/160></a>
        <input type="button"
onclick="s1_125 ();apsize ();getshutterspeed ();"; value=1/125></a>
        <input type="button"
onclick="s1_100 ();apsize ();getshutterspeed ();"; value=1/100></a>
        <input type="button"
onclick="s1_80 ();apsize ();getshutterspeed ();"; value=1/80></a>
        <input type="button"
onclick="s1_60 ();apsize ();getshutterspeed ();"; value=1/60></a>

```

```



```

```

        <input type="button"
onclick="s13 (); apsize (); getshutterspeed ();"; value=13></a>
        <input type="button"
onclick="s15 (); apsize (); getshutterspeed ();"; value=15></a>
        <input type="button"
onclick="s20 (); apsize (); getshutterspeed ();"; value=20></a>
        <input type="button"
onclick="s25 (); apsize (); getshutterspeed ();"; value=25></a>
        <input type="button"
onclick="s30 (); apsize (); getshutterspeed ();"; value=30></a>
        <input type="button"
onclick="sx250 (); apsize (); getshutterspeed ();"; value="x 250"></a>
        <input type="button"
onclick="bulb (); apsize (); getshutterspeed ();"; value=Bulb></a></div>

```

Με την αλλαγή της ταχύτητας του κλείστρου αλλάζει επιπλέον και το μέγεθος του διαφράγματος για αυτό και εκτελείτε η λειτουργία `apsize ()`.

Ο κώδικας του μενού για την ενεργοποίηση ή όχι της αυτόματης λειτουργίας ISO είναι:

```

<h1>Auto ISO </h1>
    <h2 id="autoiso"> </h2>
        <input type="button"
onclick="autoiso_on (); getautoiso (); apsize (); getshutterspeed ();";
value=ON>
        <input type="button"
onclick="autoiso_off (); getautoiso (); apsize (); getshutterspeed ();";
value=OFF>

```

Με την αλλαγή της αυτόματης λειτουργίας ISO αλλάζει επιπλέον η ταχύτητα του κλείστρου και το μέγεθος του διαφράγματος για αυτό και εκτελούνται οι λειτουργίες `apsize ()` και `getshutterspeed ()`.

Ο κώδικας του μενού αλλαγής της τιμής ISO είναι:

```

<h1>ISO </h1>
    <h2 id="iso"> </h2>
        <button onclick="drop ('myDropdown4') " class="dropbtn"> Select
ISO</button>
        <div id="myDropdown4" class="dropdown-content">
            <input type="button"
onclick="iso_100 (); apsize (); getshutterspeed (); getiso ();";
value=100></a>
            <input type="button"
onclick="iso_125 (); apsize (); getshutterspeed (); getiso ();";
value=125></a>
            <input type="button"
onclick="iso_160 (); apsize (); getshutterspeed (); getiso ();";
value=160></a>
            <input type="button"
onclick="iso_200 (); apsize (); getshutterspeed (); getiso ();";
value=200></a>
            <input type="button"
onclick="iso_250 (); apsize (); getshutterspeed (); getiso ();";
value=250></a>
            <input type="button"
onclick="iso_320 (); apsize (); getshutterspeed (); getiso ();";
value=320></a>
            <input type="button"
onclick="iso_400 (); apsize (); getshutterspeed (); getiso ();";
value=400></a>

```

```

        <input type="button"
onclick="iso_500 (); apsize (); getshutterspeed (); getiso (); ";
value=500></a>
        <input type="button"
onclick="iso_640 (); apsize (); getshutterspeed (); getiso (); ";
value=640></a>
        <input type="button"
onclick="iso_800 (); apsize (); getshutterspeed (); getiso (); ";
value=800></a>
        <input type="button"
onclick="iso_1000 (); apsize (); getshutterspeed (); getiso (); ";
value=1000></a>
        <input type="button"
onclick="iso_1250 (); apsize (); getshutterspeed (); getiso (); ";
value=1250></a>
        <input type="button"
onclick="iso_1600 (); apsize (); getshutterspeed (); getiso (); ";
value=1600></a>
        <input type="button"
onclick="iso_2000 (); apsize (); getshutterspeed (); getiso (); ";
value=2000></a>
        <input type="button"
onclick="iso_2500 (); apsize (); getshutterspeed (); getiso (); ";
value=2500></a>
        <input type="button"
onclick="iso_3200 (); apsize (); getshutterspeed (); getiso (); ";
value=3200></a>
        <input type="button"
onclick="iso_4000 (); apsize (); getshutterspeed (); getiso (); ";
value=4000></a>
        <input type="button"
onclick="iso_5000 (); apsize (); getshutterspeed (); getiso (); ";
value=5000></a>
        <input type="button"
onclick="iso_6400 (); apsize (); getshutterspeed (); getiso (); ";
value=6400></a>
        <input type="button"
onclick="iso_8320 (); apsize (); getshutterspeed (); getiso (); ";
value=8320></a>
        <input type="button"
onclick="iso_10240 (); apsize (); getshutterspeed (); getiso (); ";
value=10240></a>
        <input type="button"
onclick="iso_12800 (); apsize (); getshutterspeed (); getiso (); ";
value=12800></a>
        <input type="button"
onclick="iso_25600 (); apsize (); getshutterspeed (); getiso (); ";
value=25600></a></div>

```

Με την αλλαγή της τιμής ISO αλλάζει επιπλέον η ταχύτητα του κλείστρου και το μέγεθος του διαφράγματος για αυτό και εκτελούνται οι λειτουργίες `apsize()` και `getshutterspeed()` ".

Ο κώδικας του μενού αλλαγής της τιμής αντιστάθμισης έκθεσης είναι:

```

<h1>Exposure Compensation </h1>
  <h2 id="expcomp"> </h2>
    <button onclick="drop ('myDropdown5') " class="dropbtn"> Select
Exposure Compensation</button>

```

```

        <div id="myDropdown5" class="dropdown-content">
            <input type="button"
onclick="comp_50 (); aresize (); getshutterspeed (); getexpcomp (); ";
value=+5.0></a>
            <input type="button"
onclick="comp_47 (); aresize (); getshutterspeed (); getexpcomp (); ";
value=+4.7></a>
            <input type="button"
onclick="comp_43 (); aresize (); getshutterspeed (); getexpcomp (); ";
value=+4.3></a>
            <input type="button"
onclick="comp_40 (); aresize (); getshutterspeed (); getexpcomp (); ";
value=+4.0></a>
            <input type="button"
onclick="comp_37 (); aresize (); getshutterspeed (); getexpcomp (); ";
value=+3.7></a>
            <input type="button"
onclick="comp_33 (); aresize (); getshutterspeed (); getexpcomp (); ";
value=+3.3></a>
            <input type="button"
onclick="comp_30 (); aresize (); getshutterspeed (); getexpcomp (); ";
value=+3.0></a>
            <input type="button"
onclick="comp_27 (); aresize (); getshutterspeed (); getexpcomp (); ";
value=+2.7></a>
            <input type="button"
onclick="comp_23 (); aresize (); getshutterspeed (); getexpcomp (); ";
value=+2.3></a>
            <input type="button"
onclick="comp_20 (); aresize (); getshutterspeed (); getexpcomp (); ";
value=+2.0></a>
            <input type="button"
onclick="comp_17 (); aresize (); getshutterspeed (); getexpcomp (); ";
value=+1.7></a>
            <input type="button"
onclick="comp_13 (); aresize (); getshutterspeed (); getexpcomp (); ";
value=+1.3></a>
            <input type="button"
onclick="comp_10 (); aresize (); getshutterspeed (); getexpcomp (); ";
value=+1.0></a>
            <input type="button"
onclick="comp_07 (); aresize (); getshutterspeed (); getexpcomp (); ";
value=+0.7></a>
            <input type="button"
onclick="comp_03 (); aresize (); getshutterspeed (); getexpcomp (); ";
value=+0.3></a>
            <input type="button"
onclick="comp_00 (); aresize (); getshutterspeed (); getexpcomp (); ";
value=0></a>
            <input type="button"
onclick="comp_m03 (); aresize (); getshutterspeed (); getexpcomp (); "; value=-
0.3></a>
            <input type="button"
onclick="comp_m07 (); aresize (); getshutterspeed (); getexpcomp (); "; value=-
0.7></a>

```

```

        <input type="button"
onclick="comp_m10 (); aresize (); getshutterspeed (); getexpcomp ();"; value=-
1.0></a>
        <input type="button"
onclick="comp_m13 (); aresize (); getshutterspeed (); getexpcomp ();"; value=-
1.3></a>
        <input type="button"
onclick="comp_m17 (); aresize (); getshutterspeed (); getexpcomp ();"; value=-
1.7></a>
        <input type="button"
onclick="comp_m20 (); aresize (); getshutterspeed (); getexpcomp ();"; value=-
2.0></a>
        <input type="button"
onclick="comp_m23 (); aresize (); getshutterspeed (); getexpcomp ();"; value=-
2.3></a>
        <input type="button"
onclick="comp_m27 (); aresize (); getshutterspeed (); getexpcomp ();"; value=-
2.7></a>
        <input type="button"
onclick="comp_m30 (); aresize (); getshutterspeed (); getexpcomp ();"; value=-
3.0></a>
        <input type="button"
onclick="comp_m33 (); aresize (); getshutterspeed (); getexpcomp ();"; value=-
3.3></a>
        <input type="button"
onclick="comp_m37 (); aresize (); getshutterspeed (); getexpcomp ();"; value=-
3.7></a>
        <input type="button"
onclick="comp_m40 (); aresize (); getshutterspeed (); getexpcomp ();"; value=-
4.0></a>
        <input type="button"
onclick="comp_m43 (); aresize (); getshutterspeed (); getexpcomp ();"; value=-
4.3></a>
        <input type="button"
onclick="comp_m47 (); aresize (); getshutterspeed (); getexpcomp ();"; value=-
4.7></a>
        <input type="button"
onclick="comp_m50 (); aresize (); getshutterspeed (); getexpcomp ();"; value=-
5.0></a></div>

```

Με την αλλαγή της τιμής αντιστάθμισης έκθεσης αλλάζει επιπλέον η ταχύτητα του κλείστρου και το μέγεθος του διαφράγματος για αυτό και εκτελούνται οι λειτουργίες **arsize ()** και **getshutterspeed ()** .

Ο κώδικας του μενού αλλαγής της τιμής μετατόπισης του ευέλικτου προγράμματος είναι:

```

<h1>Flexible Program Shift </h1>
    <h2 id="flex"> </h2>
        <button onclick="drop ('myDropdown6') " class="dropbtn"> Select
Program Shift</button>
        <div id="myDropdown6" class="dropdown-content">
            <input type="button"
onclick="shift_30 (); aresize (); getshutterspeed (); getflex ();";
value=+30></a>
            <input type="button"
onclick="shift_28 (); aresize (); getshutterspeed (); getflex ();";
value=+28></a>

```

```

        <input type="button"
onclick="shift_26();apsize();getshutterspeed();getflex();"
value=+26></a>
        <input type="button"
onclick="shift_24();apsize();getshutterspeed();getflex();"
value=+24></a>
        <input type="button"
onclick="shift_22();apsize();getshutterspeed();getflex();"
value=+22></a>
        <input type="button"
onclick="shift_20();apsize();getshutterspeed();getflex();"
value=+20></a>
        <input type="button"
onclick="shift_18();apsize();getshutterspeed();getflex();"
value=+18></a>
        <input type="button"
onclick="shift_16();apsize();getshutterspeed();getflex();"
value=+16></a>
        <input type="button"
onclick="shift_14();apsize();getshutterspeed();getflex();"
value=+14></a>
        <input type="button"
onclick="shift_12();apsize();getshutterspeed();getflex();"
value=+12></a>
        <input type="button"
onclick="shift_10();apsize();getshutterspeed();getflex();"
value=+10></a>
        <input type="button"
onclick="shift_8();apsize();getshutterspeed();getflex();"
value=+8></a>
        <input type="button"
onclick="shift_6();apsize();getshutterspeed();getflex();"
value=+6></a>
        <input type="button"
onclick="shift_4();apsize();getshutterspeed();getflex();"
value=+4></a>
        <input type="button"
onclick="shift_2();apsize();getshutterspeed();getflex();"
value=+2></a>
        <input type="button"
onclick="shift_0();apsize();getshutterspeed();getflex();" value=0></a>
        <input type="button"
onclick="shift_m2();apsize();getshutterspeed();getflex();" value=-
2></a>
        <input type="button"
onclick="shift_m4();apsize();getshutterspeed();getflex();" value=-
4></a>
        <input type="button"
onclick="shift_m6();apsize();getshutterspeed();getflex();" value=-
6></a>
        <input type="button"
onclick="shift_m8();apsize();getshutterspeed();getflex();" value=-
8></a>
        <input type="button"
onclick="shift_m10();apsize();getshutterspeed();getflex();" value=-
10></a>

```

```

        <input type="button"
onclick="shift_m12 ();apsize ();getshutterspeed ();getflex ();"; value=--
12></a>
        <input type="button"
onclick="shift_m14 ();apsize ();getshutterspeed ();getflex ();"; value=--
14></a>
        <input type="button"
onclick="shift_m16 ();apsize ();getshutterspeed ();getflex ();"; value=--
16></a>
        <input type="button"
onclick="shift_m18 ();apsize ();getshutterspeed ();getflex ();"; value=--
18></a>
        <input type="button"
onclick="shift_m20 ();apsize ();getshutterspeed ();getflex ();"; value=--
20></a>
        <input type="button"
onclick="shift_m22 ();apsize ();getshutterspeed ();getflex ();"; value=--
22></a>
        <input type="button"
onclick="shift_m24 ();apsize ();getshutterspeed ();getflex ();"; value=--
24></a>
        <input type="button"
onclick="shift_m26 ();apsize ();getshutterspeed ();getflex ();"; value=--
26></a>
        <input type="button"
onclick="shift_m28 ();apsize ();getshutterspeed ();getflex ();"; value=--
28></a>
        <input type="button"
onclick="shift_m30 ();apsize ();getshutterspeed ();getflex ();"; value=--
30></a></div>

```

Με την αλλαγή της τιμής μετατόπισης του ευέλικτου προγράμματος αλλάζει επιπλέον η ταχύτητα του κλείστρου και το μέγεθος του διαφράγματος για αυτό και εκτελούνται οι λειτουργίες `apsize()` και `getshutterspeed()`.

Ο κώδικας του μενού αλλαγής της ποιότητας εικόνας είναι:

```

<h1>Image Quality </h1>
    <h2 id="quality"> </h2>
        <button onclick="drop ('myDropdown7') " class="dropbtn"> Select
Image Quality</button>
        <div id="myDropdown7" class="dropdown-content">
            <input type="button" onclick="basic ();getquality ();";
value="JPEG Basic"></a>
            <input type="button" onclick="normal ();getquality ();";
value="JPEG Normal"></a>
            <input type="button" onclick="fine ();getquality ();";
value="JPEG Fine"></a>
            <input type="button" onclick="tiff ();ltiff ();"; value=TIFF></a>
            <input type="button" onclick="raw ();getquality ();"; value=NEF
(Raw)></a>
            <input type="button" onclick="rbasic ();getquality ();";
value=NEF+Basic></a>
            <input type="button" onclick="rnormal ();getquality ();";
value=NEF+Normal></a>
            <input type="button" onclick="rfine ();getquality ();";
value=NEF+Fine></a></div>

```

Ο κώδικας του μενού αλλαγής του μεγέθους εικόνας είναι:

```

h1>Image Size </h1>

```



```

    <h2 id="size"> </h2>
    <button onclick="drop ('myDropdown8') " class="dropbtn"> Select
Image Size</button>
    <div id="myDropdown8" class="dropdown-content">
    <input type="button" onclick="large ();getsize ();";
value="4256x2832"></a>
    <input type="button" onclick="med ();getsize ();";
value="3184x2120"></a>
    <input type="button" onclick="sml ();getsize ();";
value="2128x1416"></a></div>

```

Ο κώδικας του μενού αλλαγής της συμπίεσης των αρχείων τύπου RAW είναι:

```

<h1>RAW Compression </h1>
    <h2 id="rsize"> </h2>
    <button onclick="drop ('myDropdown9') " class="dropbtn"> Select
RAW Compression</button>
    <div id="myDropdown9" class="dropdown-content">
    <input type="button" onclick="lossless ();getsize ();";
value="Lossless compressed"></a>
    <input type="button" onclick="lossy ();getsize ();";
value="Compressed"></a>
    <input type="button" onclick="comp_off ();sizeof ();";
value="Uncompressed"></a></div>

```

Ο κώδικας του μενού αλλαγής της ισορροπία λευκού είναι:

```

<h1>White Balance </h1>
    <h2 id="wbalance"> </h2>
    <button onclick="drop ('myDropdown10') " class="dropbtn"> Select
White Balance</button>
    <div id="myDropdown10" class="dropdown-content">
    <input type="button" onclick="wauto ();getwbalance ();";
value="Automatic"></a>
    <input type="button" onclick="dayl ();getwbalance ();";
value="Daylight"></a>
    <input type="button" onclick="fluo ();getwbalance ();";
value="Fluorescent"></a>
    <input type="button" onclick="tung ();getwbalance ();";
value="Tungsten"></a>
    <input type="button" onclick="wflash ();getwbalance ();";
value="Flash"></a>
    <input type="button" onclick="cloudy ();getwbalance ();";
value="Cloudy"></a>
    <input type="button" onclick="shade ();getwbalance ();";
value="Shade"></a>
    <input type="button" onclick="color_temp ();getwbalance ();";
value="Color Temperature"></a>
    <input type="button" onclick="wpreset ();getwbalance ();";
value="Preset"></a></div>

```

Ο κώδικας του μενού αλλαγής του χρωματικού χώρου είναι:

```

<h1>Color Space </h1>
    <h2 id="colorspace"> </h2>
    <button onclick="drop ('myDropdown11') " class="dropbtn"> Select
Color Space</button>
    <div id="myDropdown11" class="dropdown-content">
    <input type="button" onclick="srgb ();getcolorspace ();";
value="sRGB"></a>
    <input type="button" onclick="adobe ();getcolorspace ();";
value="Adobe"></a></div>

```

Ο κώδικας του μενού αλλαγής της αποθρομβοποίησης υψηλών τιμών ISO είναι:

```
<h1>High ISO Noise Reduction </h1>
  <h2 id="isonr"> </h2>
  <button onclick="drop ('myDropdown12') " class="dropbtn">Select
High ISO Noise Reduction</button>
  <div id="myDropdown12" class="dropdown-content">
  <input type="button" onclick="hnroff ();gethisonr ();";
value="Off"></a>
  <input type="button" onclick="hnr low ();gethisonr ();";
value="Low"></a>
  <input type="button" onclick="hnr norm ();gethisonr ();";
value="Normal"></a>
  <input type="button" onclick="hnr high ();gethisonr ();";
value="High"></a></div>
```

Ο κώδικας του μενού ενεργοποίηση ή όχι της αποθρομβοποίησης της έκθεσης μακράς διάρκειας είναι:

```
<h1>Long Exposure Noise Reduction </h1>
  <h2 id="lnr"> </h2>
  <button onclick="drop ('myDropdown13') " class="dropbtn"> Long
Exposure Noise Reduction</button>
  <div id="myDropdown13" class="dropdown-content">
  <input type="button" onclick="lnron ();getlnr ();";
value="On"></a>
  <input type="button" onclick="lnroff ();getlnr ();";
value="Off"></a></div>
```

Ο κώδικας του μενού αλλαγής της λειτουργίας φλας είναι:

```
<h1>Flash Mode </h1>
  <h2 id="flashmode"> </h2>
  <button onclick="drop ('myDropdown14') " class="dropbtn"> Select
Flash Mode</button>
  <div id="myDropdown14" class="dropdown-content">
  <input type="button" onclick="flredauto ();getflashmode ();";
value="Red-eye Automatic"></a>
  <input type="button" onclick="flauto ();getflashmode ();";
value="Auto"></a>
  <input type="button" onclick="flautoslow ();getflashmode ();";
value="Auto Slow Sync"></a>
  <input type="button"
onclick="flrearcertainslow ();getflashmode ();"; value="Rear Curtain Sync
+ Slow Sync"></a>
  <input type="button" onclick="flredslow ();getflashmode ();";
value="Red-eye Reduction + Slow Sync"></a></div>
```

Ο κώδικας του μενού αλλαγής της ταχύτητας συγχρονισμού φλας είναι:

```
<h1>Flash Sync Speed </h1>
  <h2 id="flashsyncspeed"> </h2>
  <button onclick="drop ('myDropdown15') " class="dropbtn">Select
Flash Sync Speed</button>
  <div id="myDropdown15" class="dropdown-content">
  <input type="button" onclick="autofp ();getflashsyncspeed ();";
value="1/250s (Auto FP)"></a>
  <input type="button" onclick="f1_250 ();getflashsyncspeed ();";
value="1/250s"></a>
  <input type="button" onclick="f1_200 ();getflashsyncspeed ();";
value="1/200s"></a>
  <input type="button" onclick="f1_160 ();getflashsyncspeed ();";
value="1/160s"></a>
```

```



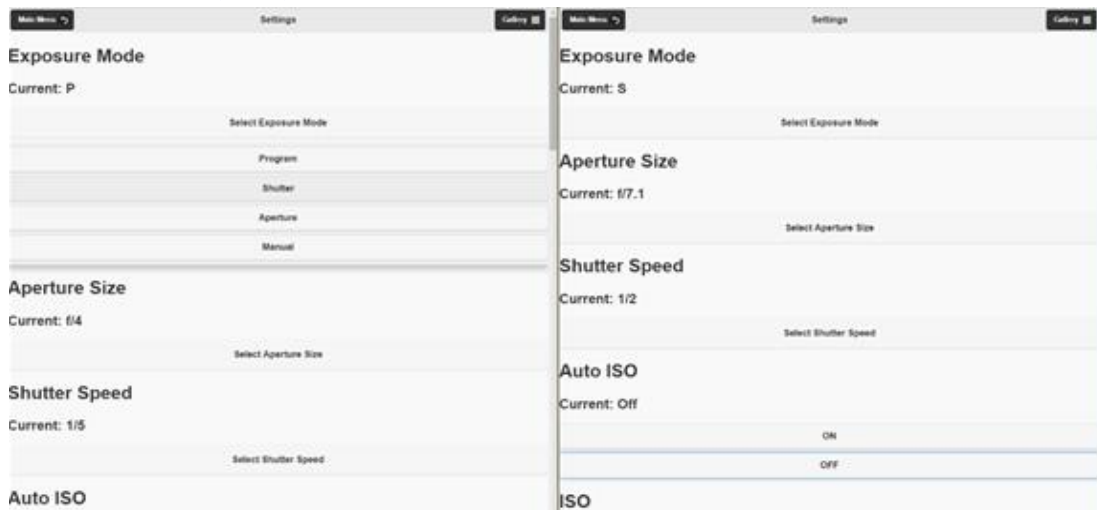
```

Ο κώδικας του μενού αλλαγής της ταχύτητας κλείστρου φλας είναι:

```

<h1>Flash Shutter Speed </h1>
<h2 id="flashshutterspeed"> </h2>
<button onclick="drop ('myDropdown16')" class="dropbtn">Select
Flash Sync Speed</button>
<div id="myDropdown16" class="dropdown-content">
<input type="button"
onclick="fs1_60 ();getflashshutterspeed ();"; value="1/60s"></a>
<input type="button"
onclick="fs1_30 ();getflashshutterspeed ();"; value="1/30s"></a>
<input type="button"
onclick="fs1_15 ();getflashshutterspeed ();"; value="1/15s"></a>
<input type="button" onclick="fs1_8 ();getflashshutterspeed ();";
value="1/8s"></a>
<input type="button" onclick="fs1_4 ();getflashshutterspeed ();";
value="1/4s"></a>
<input type="button" onclick="fs1_2 ();getflashshutterspeed ();";
value="1/2s"></a>
<input type="button" onclick="fs1 ();getflashshutterspeed ();";
value="1s"></a>
<input type="button" onclick="fs2 ();getflashshutterspeed ();";
value="2s"></a>
<input type="button" onclick="fs4 ();getflashshutterspeed ();";
value="4s"></a>
<input type="button" onclick="fs8 ();getflashshutterspeed ();";
value="8s"></a>
<input type="button" onclick="fs15 ();getflashshutterspeed ();";
value="15s"></a>
<input type="button" onclick="fs30 ();getflashshutterspeed ();";
value="30s"></a></div>

```



Εικόνα 9:Αλλαγή του προγράμματος έκθεσης από αυτοματοποιημένο πρόγραμμα σε αυτόματη προτεραιότητα κλείστρου.

2.2.10: Δημιουργία σελίδας σύνδεσης στην ιστοσελίδα

Η ιστοσελίδα μετά την ολοκλήρωση των πτυσσόμενων μενού ρυθμίσεων είναι πλήρως λειτουργική. Όταν λειτουργεί με τον υπάρχον κώδικα είναι δυνατόν να είναι προσβάσιμη μέσω του Ίντερνετ από οποιοδήποτε υπολογιστή είναι συνδεδεμένος στο Διαδίκτυο. Αυτό είναι ένα κενό ασφαλείας διότι δεν υπάρχει κάποια δικλείδα ασφαλείας που να αποτρέπει την μη εξουσιοδοτημένη, και πιθανός κακόβουλη, χρήση της ιστοσελίδας. Για αυτόν τον λόγο θα δημιουργηθεί ένα σύστημα σύνδεσης στην σελίδα που θα επιτρέπει την πρόσβαση μόνο σε εξουσιοδοτημένους χρήστες. Το σύστημα αυτό θα γραφτεί σε γλώσσα php και για να συνδεθεί κάποιος θα πρέπει να γνωρίζει ένα έγκυρο όνομα χρήστη (username) και κωδικό (password). Για την επίτευξη αυτού του σκοπού θα δημιουργηθούν τρία νέα αρχεία (login.php, logout.php, index.php).

Το αρχείο login.php είναι η πρώτη σελίδα που θα βλέπει ο χρήστης και εκεί θα υπάρχει η φόρμα HTML όπου θα πληκτρολογεί τα διαπιστευτήριά του. Ο κώδικας HTML της φόρμας αυτής είναι:

```
<!doctype html>
<html>
<head>
<meta charset="utf-8">
<title>Login</title>
<link href="./css/style.css" rel="stylesheet">
</head>
<body>
<br>
<form action="" method="post" name="Login_Form">
  <table width="400" border="0" align="center" cellpadding="5"
  cellspacing="1" class="Table">
    <?php if(isset($msg)) {?>
    <tr>
      <td colspan="2" align="center" valign="top"><?php echo
$msg;?></td>
    </tr>
    <?php } ?>
    <tr>
      <td colspan="2" align="left" valign="top"><h3>Login</h3></td>
    </tr>
    <tr>
      <td align="right" valign="top">Username</td>
      <td><input name="Username" type="text" class="Input"></td>
    </tr>
    <tr>
      <td align="right">Password</td>
      <td><input name="Password" type="password" class="Input"></td>
    </tr>
    <tr>
      <td>&nbsp;</td>
      <td><input name="Submit" type="submit" value="Login"
class="Button3"></td>
    </tr>
  </table>
</form>
</body>
</html>
```

Η κώδικας της ιστοσελίδας αυτής έχει εξηγηθεί σε μεγάλο του μέρος σε προηγούμενα κεφάλαια (Κεφάλαιο 2.2.2: Δημιουργία της βασικής δομής της ιστοσελίδας αλλαγής των ρυθμίσεων, Κεφάλαιο 2.2.4: Ολοκλήρωση της ανακατεύθυνσης και δημιουργία της κεφαλίδας (header) στην νέα σελίδα, Κεφάλαιο 2.2.5: Δημιουργία πίνακα πληροφοριών για αυτό και δεν θα ξαναγίνει ανάλυση όσων έχουν ήδη αναφερθεί.

Η ετικέτα `
` εισάγει ένα σπάσιμο μιας γραμμής. Είναι μια κενή ετικέτα που σημαίνει ότι δεν έχει τελική ετικέτα. Η `form action="" method="post"` καθορίζει τον τρόπο αποστολής δεδομένων φόρμας. Τα δεδομένα φόρμας μπορούν να σταλούν ως μεταβλητές URL, με τη μέθοδο `"get"`, ή ως μετά-συναλλαγή HTTP (HTTP post transaction), με τη μέθοδο `"post"`.

Στην συνέχεια θα γίνει προσθήκη πάνω από τον κώδικα HTML. του εξής απαραίτητου κώδικα php:

```
<?php session_start();

    if(isset($_POST['Submit'])) {

        $logins = array('User' => '123456', 'username1' => 'password1');

        $Username = isset($_POST['Username']) ? $_POST['Username'] :
'';
        $Password = isset($_POST['Password']) ? $_POST['Password'] :
'';

        if (isset($logins[$Username]) && $logins[$Username] ==
$Password) {

$_SESSION['UserData']['Username']=$logins[$Username];
        header("location:index.php");
        exit;
        } else {
            $msg="<span style='color:red'>Invalid Login
Details</span>";
        }
    }
?>
```

Αρχικά δηλώνεται η έναρξη της συνεδρίας με την εντολή `session_start();` και ακολουθεί ο έλεγχος της απεσταλμένης φόρμας Log in (`if(isset($_POST['Submit']))`). Έπειτα γίνεται ο ορισμός του ονόματος χρήστη (username) και κωδικού (password) σε μία διάταξη (`$logins = array('User' => '123456', 'username1' => 'password1');`) και ελέγχεται και αποδίδεται το υποβληθέν όνομα χρήστη και κωδικό πρόσβασης στη νέα μεταβλητή (`$Username = isset($_POST['Username']) ? $_POST['Username'] : ''; $Password = isset($_POST['Password']) ? $_POST['Password'] : '';`). Τέλος γίνεται έλεγχος της ύπαρξης του ονόματος χρήστη και κωδικού πρόσβασης σε καθορισμένο πίνακα (`if (isset($logins[$Username]) && $logins[$Username] == $Password)`) και εφόσον αναγνωρισθεί με επιτυχία έχουμε την μεταφορά του χρήστη στην `index.php` όπου θα μπορεί να αποσυνδεθεί μέχρι να μεταφερθεί στην αρχική σελίδα της εφαρμογής (`$_SESSION['UserData']['Username']=$logins[$Username]; header("location:index.php"); exit;`). Εάν δεν γίνει επιτυχημένη αναγνώριση θα εμφανιστεί ένα μήνυμα σφάλματος (`$msg="Invalid Login Details;`).

| Login | |
|--------------------------------------|--------------------------|
| Username | <input type="text"/> |
| Password | <input type="password"/> |
| <input type="button" value="Login"/> | |

Εικόνα 10: Η σελίδα σύνδεσης χρήστη

Η σελίδα index.php όπου θα ανακατευθύνεται ο χρήστη, όπως αναφέρθηκε νωρίτερα, θα δίνει την δυνατότητα να αποσυνδεθεί μέχρι να μεταφερθεί στην αρχική σελίδα της εφαρμογής. Ο κώδικας της είναι ο εξής:

```
<?php session_start();
```

```
if(!isset($_SESSION['UserData']['Username'])){
    header("location:login.php");
    exit;
}
?>
```

Logging in..... [Click here](logout.php) to Logout.

```
<!DOCTYPE html>
<html>
<head>
<meta http-equiv="Refresh" content="5;index.html">
</head>
```

Η εντολή `http-equiv="Refresh" content="5;index.html"` μεταφέρει τον χρήστη στην αρχική σελίδα της εφαρμογής μετά από 5 δευτερόλεπτα.

Logging in..... [Click here](#) to Logout.

Εικόνα 11: Επιτυχημένη σύνδεση στην σελίδα

Η σελίδα logout.php αποσυνδέει τον χρήστη από την εφαρμογή. Αυτό γίνεται καταστρέφοντας την υπάρχων συνεδρία. Ο κώδικας της είναι ο εξής:

```
<?php session_start();
```

```
session_destroy();
header("location:login.php");
exit;
?>
```

Κεφάλαιο 3: Συμπεράσματα και μελλοντικές επεκτάσεις

Με την υλοποίηση της πτυχιακής αποκτήθηκαν σημαντικές γνώσεις πάνω στην οικογένεια υπολογιστών Raspberry Pi και γενικά του τρόπου λειτουργίας συσκευών με επεξεργαστή τύπου ARM. Επιπλέον η εργασία αυτή βοήθησε στην εισαγωγή στον τρόπο κατασκευής ιστοσελίδων, διαδικτυακών εφαρμογών και στους τρόπους αποστολής και λήψης δεδομένων μέσω σελίδων διαδικτύου. Για την δημιουργία της χρειάστηκε να χρησιμοποιηθούν οι γλώσσες προγραμματισμού HTML, PHP, Javascript και αυτό αποτέλεσε αφορμή για την εκμάθηση των γλώσσων αυτών. Τέλος αποκτήθηκαν χρήσιμες γνώσεις για την ασφάλεια ιστοσελίδων καθώς και για τα λειτουργικά τύπου Linux.

Το σύστημα που υλοποιήθηκε είναι μικρού κόστους και μεγέθους. Είναι εύκολα δυνατόν να γίνει φορητό με την χρήση μιας κατάλληλης μπαταρίας για την τροφοδοσία του υπολογιστή χάρις την μικρή κατανάλωση ενέργειας που έχει. Ένα μεγάλο επίσης πλεονέκτημα που διαθέτει είναι ότι μπορούν εύκολα να γίνουν βελτιώσεις και επεκτάσεις εξαιτίας του γεγονότος ότι βασίζεται σε λογισμικό ανοιχτού κώδικα και η πλακέτα Raspberry Pi διαθέτει αρκετές εισόδους και εξόδους διαφόρων ειδών.

Ορισμένες βελτιώσεις και επεκτάσεις που μπορούν να γίνουν είναι:

- Επέκταση της συμβατότητας της ιστοσελίδας με άλλα μοντέλα φωτογραφικών μηχανών εκτός της οικογένειας μονοψήφιων μοντέλων της εταιρίας Nikon (D1,D2,D3 κοκ).
- Προσθήκη λειτουργιών βίντεο και άλλων λειτουργιών που υπάρχουν σε σώματα μηχανών νεότερα από αυτό που χρησιμοποιήθηκε στην πτυχιακή αυτή.
- Βελτίωση της ασφάλειας και της εμφάνισης της ιστοσελίδας.
- Με την χρήση διαφόρων ειδών αισθητήρων συνδεδεμένων στο Pi, προσθήκη της δυνατότητας αυτόματης ενεργοποίησης της φωτογραφικής μηχανής βάση κάποιων προγραμματιζόμενων σεναρίων (π.χ. ενεργοποίηση σε περίπτωση εντοπισμού κίνησης)
- Προσθήκη δυνατότητας απομακρυσμένης κίνησης της κάμερας μέσω ενός συστήματος pan-tilt που ελέγχεται από το Raspberry Pi.

Παράρτημα

Login.php

```
<?php session_start();

    if(isset($_POST['Submit'])){

        $logins = array('User' => '123456', 'username1' => 'password1');

        $Username = isset($_POST['Username']) ? $_POST['Username'] : '';
        $Password = isset($_POST['Password']) ? $_POST['Password'] : '';

        if (isset($logins[$Username]) && $logins[$Username] == $Password){

            $_SESSION['UserData']['Username']=$logins[$Username];
            header("location:index.php");
            exit;
        } else {

            $msg="<span style='color:red'>Invalid Login Details</span>";
        }
    }
?>
<!doctype html>
<html>
<head>
<meta charset="utf-8">
<title>Login</title>
<link href="./css/style.css" rel="stylesheet">
</head>
<body>
<br>
<form action="" method="post" name="Login_Form">
    <table width="400" border="0" align="center" cellpadding="5" cellspacing="1"
class="Table">
        <?php if(isset($msg)) {?>
        <tr>
            <td colspan="2" align="center" valign="top"><?php echo $msg;?></td>
        </tr>
        <?php } ?>
        <tr>
            <td colspan="2" align="left" valign="top"><h3>Login</h3></td>
        </tr>
        <tr>
            <td align="right" valign="top">Username</td>
            <td><input name="Username" type="text" class="Input"></td>
        </tr>
        <tr>
            <td align="right">Password</td>
            <td><input name="Password" type="password" class="Input"></td>
        </tr>
        <tr>
            <td>&nbsp;</td>
            <td><input name="Submit" type="submit" value="Login" class="Button3"></td>
        </tr>
    </table>
</form>
</body>
</html>
```


Logout.php

```
<?php session_start();
session_destroy();
header("location:login.php");
exit;
?>
```

Index.php

```
<?php session_start();

if(!isset($_SESSION['UserData']['Username'])){
    header("location:login.php");
    exit;
}
?>
```

Logging in..... [Click here](logout.php) to Logout.

```
<!DOCTYPE html>
<html>
<head>
<meta http-equiv="Refresh" content="5;index.html">
</head>
```

Index.html

```
<!DOCTYPE html>
<html>
<head>
    <title>Camera Control Page</title>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet" href="css/jquery.mobile-1.4.2.css" />
    <script src="js/jquery.js"></script>
    <script src="js/jquery.mobile-1.4.2.js"></script>
    <script src="js/controller.js"></script>
</head>

<style>
.ui-grid-a img {
    width : 90%;
    height : auto;
    border:1px #444444 solid;
    box-shadow: 2px 2px 2px #888888;
}

table {
    border-collapse: collapse;
    width: 100%;
}

th, td {
    padding: 12px;
}
</style>

<body>

<div data-role="page" id="camera">

    <div data-role="header">
        <h1 id="cameraName"></h1>
```

```

        <a href="#gallery" class="ui-btn-right ui-btn ui-btn-b ui-btn-inline ui-mini
ui-corner-all ui-btn-icon-right ui-icon-grid">Gallery
    </a>
</div><!-- /header -->

<div role="main" class="ui-content" style="text-align:center">

    <button onclick="takePicture()" id="snap"></button>
    <button onclick="location.href='settings.html#settings'" id="set"></button>
    <table>
<tr>
    <th>Name</th>
</tr>
<tr>
    <td>Exposure Mode</td>
    <td id="expmode"></td>
</tr>
<tr>
    <td>Aperture Size</td>
    <td id="apsize"></td>
</tr>
<tr>
    <td>Shutter Speed</td>
    <td id="shutterspeed"></td>
</tr>
<tr>
    <td>Auto ISO</td>
    <td id="autoiso"></td>
</tr>
<tr>
    <td>ISO</td>
    <td id="iso"></td>
</tr>
<tr>
    <td>Exposure Compensation
</td>
    <td id="expcomp"></td>
</tr>
<tr>
    <td>Flexible Program Shift</td>
    <td id="flex"></td>
</tr>
<tr>
    <td>Image Quality</td>
    <td id="quality"></td>
</tr>
<tr>
    <td>Image Size</td>
    <td id="size"></td>
</tr>
<tr>
    <td>RAW Compression</td>
    <td id="rsize"></td>
</tr>
<tr>
    <td>White Balance</td>
    <td id="wbalance"></td>
</tr>
<tr>
    <td>Color Space</td>
    <td id="colorspace"></td>
</tr>
<tr>
    <td>High ISO Noise Reduction</td>
    <td id="isonr"></td>
</tr>
<tr>
    <td>Long Exposure Noise Reduction</td>
    <td id="lnr"></td>

```

```

</tr>
<tr>
  <td>Flash Mode</td>
  <td id="flashmode"></td>
</tr>
<tr>
  <td>Flash Sync Speed</td>
  <td id="flashsyncspeed"></td>
</tr>
<tr>
  <td>Flash Shutter Speed</td>
  <td id="flashshutterspeed"></td>
</tr>
</table>
</div><!-- /content -->

</div>

<div data-role="page" id="gallery">

  <div data-role="header">
    <button onclick="location.href='#camera'" class="ui-btn-left ui-btn ui-btn-b ui-
    btn-inline ui-mini ui-corner-all ui-btn-icon-right ui-icon-back">Main Menu</button>
    <button onclick="location.href='settings.html#settings'" class="ui-btn-right
    ui-btn ui-btn-b ui-btn-inline ui-mini ui-corner-all ui-btn-icon-right ui-icon-
    gear">Settings</button>
  </a>
  <h1 id="cameraName">Gallery</h1>
</div><!-- /header -->

  <div role="main" class="ui-content">

    <div class="ui-grid-a my-breakpoint" id="galleryGrid"></div>

    <script type="text/html" id="galleryTemplate">
      <div class="ui-block-@char" id="@id" style="text-align:center">
        <p></p>
        <p>@imageName</p>
        <p><a target="_blank"
href="service.php?action=getImage&file=@imageName">Download Source</a></p>
        <p><a href="javascript:deleteFile('@imageName');">Delete</a></p>
      </div>
    </script>

    </div><!-- /content -->

    <script>

function drop(idOfTheDiv) {
  document.getElementById(idOfTheDiv).classList.toggle("show");
}

// Close the dropdown menu if the user clicks outside of it
window.onclick = function(event) {
  if (!event.target.matches('.dropbtn')) {

    var dropdowns = document.getElementsByClassName("dropdown-content");
    var i;
    for (i = 0; i < dropdowns.length; i++) {
      var openDropdown = dropdowns[i];
      if (openDropdown.classList.contains('show')) {
        openDropdown.classList.remove('show');
      }
    }
  }
}

function ltiff(){
  document.getElementById('quality').innerHTML = 'Current: TIFF';
}

```

```

    }
    function sizeoff() {
        document.getElementById('rsize').innerHTML = 'Current: Uncompressed';
    }
</script>
</div><!-- /content -->

</div><!-- /page -->

</body>
</html>

```

Settings.html

```

<!DOCTYPE html>
<html>
<head>
    <title>Camera Control Page</title>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet" href="css/jquery.mobile-1.4.2.css" />
    <script src="js/jquery.js"></script>
    <script src="js/jquery.mobile-1.4.2.js"></script>
    <script src="js/controller.js"></script>
</head>

<style>
.ui-grid-a img {
    width : 90%;
    height : auto;
    border:1px #444444 solid;
    box-shadow: 2px 2px 2px #888888;
}
.dropbtn {
    color: black;
    padding: 16px;
    font-size: 16px;
    border: none;
    cursor: pointer;
}

.dropbtn:hover, .dropbtn:focus {
    color: white;
    background-color: #2980B9;
}

.dropdown {
    position: relative;
    display: inline-block;
}

.dropdown-content {
    display: none;
    background-color: #f1f1f1;
    min-width: 160px;
    overflow: auto;
    box-shadow: 0px 8px 16px 0px rgba(0,0,0,0.2);
    z-index: 1;
}

.dropdown-content a {
    color: black;
    padding: 12px 16px;
    text-decoration: none;
    display: block;
}

.dropdown a:hover {background-color: #ddd;}

```

```

.show {display: block;}
</style>

<body>

<!-- Start of second page -->
<div data-role="page" id="settings">

    <div data-role="header">
        <button onclick="location.href='index.html#camera'" class="ui-btn-left ui-btn ui-btn-
b ui-btn-inline ui-mini ui-corner-all ui-btn-icon-right ui-icon-back">Main Menu</button>
        <button onclick="location.href='index.html#gallery'" class="ui-btn-right ui-btn
ui-btn-b ui-btn-inline ui-mini ui-corner-all ui-btn-icon-right ui-icon-
grid">Gallery</button>
    </a>
        <h1>Settings</h1>
    </div><!-- /header -->
    <h1> Exposure Mode </h1>
    <h2 id="expmode"> </h2>
        <button onclick="drop ('myDropdown1') " class="dropbtn"> Select Exposure
Mode</button>
        <div id="myDropdown1" class="dropdown-content">
            <input type="button" onclick="program();getexpmode();apsize();getshutterspeed();"
value=Program></a>
            <input type="button" onclick="shutter();getexpmode();apsize();getshutterspeed();"
value=Shutter priority></a>
            <input type="button"
onclick="aperture();getexpmode();apsize();getshutterspeed();" value=Aperture
priority></a>
            <input type="button" onclick="manual();getexpmode();apsize();getshutterspeed();"
value=Manual></a></div>
        <h1> Aperture Size </h1>
        <h2 id="apsize"> </h2>
            <button onclick="drop ('myDropdown2') " class="dropbtn"> Select Aperture
Size</button>
            <div id="myDropdown2" class="dropdown-content">
                <input type="button" onclick="f2_8();apsize();getshutterspeed();"
value=F/2.8></a>
                <input type="button" onclick="f3_2();apsize();getshutterspeed();"
value=F/3.2></a>
                <input type="button" onclick="f3_5();apsize();getshutterspeed();"
value=F3.5></a>
                <input type="button" onclick="f4();apsize();getshutterspeed();" value=F/4></a>
                <input type="button" onclick="f4_5();apsize();getshutterspeed();"
value=F/4.5></a>
                <input type="button" onclick="f5();apsize();getshutterspeed();" value=F5></a>
                <input type="button" onclick="f5_6();apsize();getshutterspeed();"
value=F/5.6></a>
                <input type="button" onclick="f6_3();apsize();getshutterspeed();"
value=F/6.3></a>
                <input type="button" onclick="f7_1();apsize();getshutterspeed();"
value=F7.1></a>
                <input type="button" onclick="f8();apsize();getshutterspeed();" value=F/8></a>
                <input type="button" onclick="f9();apsize();getshutterspeed();" value=F/9></a>
                <input type="button" onclick="f10();apsize();getshutterspeed();" value=F10></a>
                <input type="button" onclick="f11();apsize();getshutterspeed();" value=F/11></a>
                <input type="button" onclick="f13();apsize();getshutterspeed();" value=F/13></a>
                <input type="button" onclick="f14();apsize();getshutterspeed();"
value=F/14></a>
                <input type="button" onclick="f16();apsize();getshutterspeed();" value=F/16></a>
                <input type="button" onclick="f18();apsize();getshutterspeed();" value=F/18></a>
                <input type="button" onclick="f20();apsize();getshutterspeed();" value=F/20></a>
                <input type="button" onclick="f22();apsize();getshutterspeed();" value=F/22></a>
                <input type="button" onclick="f25();apsize();getshutterspeed();" value=F/25></a>
                <input type="button" onclick="f29();apsize();getshutterspeed();" value=F/29></a>
                <input type="button" onclick="f32();apsize();getshutterspeed();"
value=F/32></a></div>
            <h1> Shutter Speed </h1>
            <h2 id="shutterspeed"> </h2>

```

```

    <button onclick="drop('myDropdown3')" class="dropbtn"> Select Shutter
Speed</button>
    <div id="myDropdown3" class="dropdown-content">
    <input type="button" onclick="s1_8000();apsize();getshutterspeed();"
value=1/8000></a>
    <input type="button" onclick="s1_6400();apsize();getshutterspeed();"
value=1/6400></a>
    <input type="button" onclick="s1_5000();apsize();getshutterspeed();"
value=1/5000></a>
    <input type="button" onclick="s1_4000();apsize();getshutterspeed();"
value=1/4000></a>
    <input type="button" onclick="s1_3200();apsize();getshutterspeed();"
value=1/3200></a>
    <input type="button" onclick="s1_2500();apsize();getshutterspeed();"
value=1/2500></a>
    <input type="button" onclick="s1_2000();apsize();getshutterspeed();"
value=1/2000></a>
    <input type="button" onclick="s1_1600();apsize();getshutterspeed();"
value=1/1600></a>
    <input type="button" onclick="s1_1250();apsize();getshutterspeed();"
value=1/1250></a>
    <input type="button" onclick="s1_1000();apsize();getshutterspeed();"
value=1/1000></a>
    <input type="button" onclick="s1_800();apsize();getshutterspeed();"
value=1/800></a>
    <input type="button" onclick="s1_640();apsize();getshutterspeed();"
value=1/640></a>
    <input type="button" onclick="s1_500();apsize();getshutterspeed();"
value=1/500></a>
    <input type="button" onclick="s1_400();apsize();getshutterspeed();"
value=1/400></a>
    <input type="button" onclick="s1_320();apsize();getshutterspeed();"
value=1/320></a>
    <input type="button" onclick="s1_250();apsize();getshutterspeed();"
value=1/250></a>
    <input type="button" onclick="s1_200();apsize();getshutterspeed();"
value=1/200></a>
    <input type="button" onclick="s1_160();apsize();getshutterspeed();"
value=1/160></a>
    <input type="button" onclick="s1_125();apsize();getshutterspeed();"
value=1/125></a>
    <input type="button" onclick="s1_100();apsize();getshutterspeed();"
value=1/100></a>
    <input type="button" onclick="s1_80();apsize();getshutterspeed();"
value=1/80></a>
    <input type="button" onclick="s1_60();apsize();getshutterspeed();"
value=1/60></a>
    <input type="button" onclick="s1_50();apsize();getshutterspeed();"
value=1/50></a>
    <input type="button" onclick="s1_40();apsize();getshutterspeed();"
value=1/40></a>
    <input type="button" onclick="s1_30();apsize();getshutterspeed();"
value=1/30></a>
    <input type="button" onclick="s1_25();apsize();getshutterspeed();"
value=1/25></a>
    <input type="button" onclick="s1_20();apsize();getshutterspeed();"
value=1/20></a>
    <input type="button" onclick="s1_15();apsize();getshutterspeed();"
value=1/15></a>
    <input type="button" onclick="s1_13();apsize();getshutterspeed();"
value=1/13></a>
    <input type="button" onclick="s1_10();apsize();getshutterspeed();"
value=1/10></a>
    <input type="button" onclick="s1_8();apsize();getshutterspeed();" value=1/8></a>
    <input type="button" onclick="s1_6();apsize();getshutterspeed();" value=1/6></a>
    <input type="button" onclick="s1_5();apsize();getshutterspeed();" value=1/5></a>
    <input type="button" onclick="s1_4();apsize();getshutterspeed();" value=1/4></a>
    <input type="button" onclick="s1_3();apsize();getshutterspeed();" value=1/3></a>
    <input type="button" onclick="s10_25();apsize();getshutterspeed();"
value=1/2.5></a>
    <input type="button" onclick="s1_2();apsize();getshutterspeed();" value=1/2></a>

```

```



```

```



```



```

        <input type="button"
onclick="comp_m40();apsize();getshutterspeed();getexpcomp();" value=-4.0></a>
        <input type="button"
onclick="comp_m43();apsize();getshutterspeed();getexpcomp();" value=-4.3></a>
        <input type="button"
onclick="comp_m47();apsize();getshutterspeed();getexpcomp();" value=-4.7></a>
        <input type="button"
onclick="comp_m50();apsize();getshutterspeed();getexpcomp();" value=-5.0></a></div>
        <h1>Flexible Program Shift </h1>
        <h2 id="flex"> </h2>
        <button onclick="drop('myDropdown6')" class="dropbtn"> Select Program
Shift</button>
        <div id="myDropdown6" class="dropdown-content">
        <input type="button" onclick="shift_30();apsize();getshutterspeed();getflex();"
value=+30></a>
        <input type="button" onclick="shift_28();apsize();getshutterspeed();getflex();"
value=+28></a>
        <input type="button" onclick="shift_26();apsize();getshutterspeed();getflex();"
value=+26></a>
        <input type="button" onclick="shift_24();apsize();getshutterspeed();getflex();"
value=+24></a>
        <input type="button" onclick="shift_22();apsize();getshutterspeed();getflex();"
value=+22></a>
        <input type="button" onclick="shift_20();apsize();getshutterspeed();getflex();"
value=+20></a>
        <input type="button" onclick="shift_18();apsize();getshutterspeed();getflex();"
value=+18></a>
        <input type="button" onclick="shift_16();apsize();getshutterspeed();getflex();"
value=+16></a>
        <input type="button" onclick="shift_14();apsize();getshutterspeed();getflex();"
value=+14></a>
        <input type="button" onclick="shift_12();apsize();getshutterspeed();getflex();"
value=+12></a>
        <input type="button" onclick="shift_10();apsize();getshutterspeed();getflex();"
value=+10></a>
        <input type="button" onclick="shift_8();apsize();getshutterspeed();getflex();"
value=+8></a>
        <input type="button" onclick="shift_6();apsize();getshutterspeed();getflex();"
value=+6></a>
        <input type="button" onclick="shift_4();apsize();getshutterspeed();getflex();"
value=+4></a>
        <input type="button" onclick="shift_2();apsize();getshutterspeed();getflex();"
value=+2></a>
        <input type="button" onclick="shift_0();apsize();getshutterspeed();getflex();"
value=0></a>
        <input type="button" onclick="shift_m2();apsize();getshutterspeed();getflex();"
value=-2></a>
        <input type="button" onclick="shift_m4();apsize();getshutterspeed();getflex();"
value=-4></a>
        <input type="button" onclick="shift_m6();apsize();getshutterspeed();getflex();"
value=-6></a>
        <input type="button" onclick="shift_m8();apsize();getshutterspeed();getflex();"
value=-8></a>
        <input type="button" onclick="shift_m10();apsize();getshutterspeed();getflex();"
value=-10></a>
        <input type="button" onclick="shift_m12();apsize();getshutterspeed();getflex();"
value=-12></a>
        <input type="button" onclick="shift_m14();apsize();getshutterspeed();getflex();"
value=-14></a>
        <input type="button" onclick="shift_m16();apsize();getshutterspeed();getflex();"
value=-16></a>
        <input type="button" onclick="shift_m18();apsize();getshutterspeed();getflex();"
value=-18></a>
        <input type="button" onclick="shift_m20();apsize();getshutterspeed();getflex();"
value=-20></a>
        <input type="button" onclick="shift_m22();apsize();getshutterspeed();getflex();"
value=-22></a>
        <input type="button" onclick="shift_m24();apsize();getshutterspeed();getflex();"
value=-24></a>
        <input type="button" onclick="shift_m26();apsize();getshutterspeed();getflex();"
value=-26></a>

```

```



```

```



```

```

function ltiff(){
    document.getElementById('quality').innerHTML = 'Current: TIFF';
}
function sizeoff(){
    document.getElementById('rsize').innerHTML = 'Current: Uncompressed';
}
</script>
</div><!-- /content -->

</div><!-- /page -->

</body>
</html>

```

Service.php

```

<?php
require_once("CameraRaw.php");

//time gphoto2 --quiet --capture-image-and-download --filename
"./images/capture-%Y%m%d-%H%M%S-%03n.%C"
//exec ("gphoto2 --set-config uilock=1",$output);
//echo join("\n",$output);
//exec ("gphoto2 --capture-image",$output);
//echo join("\n",$output);
//exec ("gphoto2 --set-config uilock=1",$output);
//echo join("\n",$output);

$action = '';

if (isset($_GET['action'])){
    $action = $_GET['action'];
}

$returnObj;

try{
    switch($action){

        case "takePicture":
            exec ("gphoto2 --capture-image-and-download --filename
\"./images/capture-%Y%m%d-%H%M%S-%03n.%C\"", $output);
            echo json_encode(true);
            break;

        case "deleteFile":
            $file = $_GET['file'];
            $path_parts = pathinfo('images/'.$file);
            unlink('images/'.$file);
            unlink('images/thumbs/'.$path_parts['basename'].'.jpg');
            header('Content-Type: application/json');
            echo json_encode(true);
            break;

        case "getImage":
            $file = $_GET['file'];
            header('Content-Type: application/octet-stream');
            header('Content-Disposition: attachment; filename="'.$file.'");
            header('Content-Length: '.filesize('images/'.$file));
            $fp = fopen('images/'.$file, 'rb');
            fpassthru($fp);
            exit;
            break;

        case "program":

```

```

    exec ("gphoto2 --set-config expprogram=P", $output);
    echo json_encode(true);
    break;

case "shutter":

    exec ("gphoto2 --set-config expprogram=S", $output);
    echo json_encode(true);
    break;

case "aperture":

    exec ("gphoto2 --set-config expprogram=A", $output);
    echo json_encode(true);
    break;

case "manual":

    exec ("gphoto2 --set-config expprogram=M", $output);
    echo json_encode(true);
    break;

case "getexpmode":

    exec ("gphoto2 --get-config expprogram", $output);
    $returnObj->camera = ($output[count($output) - 6]);
    header('Content-Type: application/json');
    echo json_encode($returnObj);
    break;

case "apsize":

    exec ("gphoto2 --get-config f-number", $output);
    $routput= array_reverse($output);
    $returnObj->camera = ($routput[count($routput) - 4]);
    header('Content-Type: application/json');
    echo json_encode($returnObj);
    break;

case "f2.8":

    exec ("gphoto2 --set-config f-number=f/2.8", $output);
    echo json_encode(true);
    break;

case "f3.2":

    exec ("gphoto2 --set-config f-number=f/3.2", $output);
    echo json_encode(true);
    break;

case "f3.5":

    exec ("gphoto2 --set-config f-number=f/3.5", $output);
    echo json_encode(true);
    break;

case "f4":

    exec ("gphoto2 --set-config f-number=f/4", $output);
    echo json_encode(true);
    break;

case "f4.5":

    exec ("gphoto2 --set-config f-number=f/4.5", $output);
    echo json_encode(true);
    break;

case "f5":

```

```
    exec ("gphoto2 --set-config f-number=f/5", $output);  
    echo json_encode(true);  
    break;
```

```
case "f5.6":
```

```
    exec ("gphoto2 --set-config f-number=f/5.6", $output);  
    echo json_encode(true);  
    break;
```

```
case "f6.3":
```

```
    exec ("gphoto2 --set-config f-number=f/6.3", $output);  
    echo json_encode(true);  
    break;
```

```
case "f7.1":
```

```
    exec ("gphoto2 --set-config f-number=f/7.1", $output);  
    echo json_encode(true);  
    break;
```

```
case "f8":
```

```
    exec ("gphoto2 --set-config f-number=f/8", $output);  
    echo json_encode(true);  
    break;
```

```
case "f9":
```

```
    exec ("gphoto2 --set-config f-number=f/9", $output);  
    echo json_encode(true);  
    break;
```

```
case "f10":
```

```
    exec ("gphoto2 --set-config f-number=f/10", $output);  
    echo json_encode(true);  
    break;
```

```
case "f11":
```

```
    exec ("gphoto2 --set-config f-number=f/11", $output);  
    echo json_encode(true);  
    break;
```

```
case "f13":
```

```
    exec ("gphoto2 --set-config f-number=f/13", $output);  
    echo json_encode(true);  
    break;
```

```
case "f14":
```

```
    exec ("gphoto2 --set-config f-number=f/14", $output);  
    echo json_encode(true);  
    break;
```

```
case "f16":
```

```
    exec ("gphoto2 --set-config f-number=f/16", $output);  
    echo json_encode(true);  
    break;
```

```
case "f18":
```

```
    exec ("gphoto2 --set-config f-number=f/18", $output);  
    echo json_encode(true);  
    break;
```

```

    case "f20":

        exec ("gphoto2 --set-config f-number=f/20", $output);
        echo json_encode(true);
        break;

    case "f22":

        exec ("gphoto2 --set-config f-number=f/22", $output);
        echo json_encode(true);
        break;

    case "f25":

        exec ("gphoto2 --set-config f-number=f/25", $output);
        echo json_encode(true);
        break;

    case "f29":

        exec ("gphoto2 --set-config f-number=f/29", $output);
        echo json_encode(true);
        break;

    case "f32":

        exec ("gphoto2 --set-config f-number=f/32", $output);
        echo json_encode(true);
        break;

    case "getshutterspeed":

        exec ("gphoto2 --get-config shutterspeed2", $output);
        $routput= array_reverse($output);
        $returnObj->camera = ($routput[count($routput) - 4]);
        header('Content-Type: application/json');
        echo json_encode($returnObj);
        break;

    case "s1_8000":

        exec ("gphoto2 --set-config shutterspeed2=1/8000", $output);
        echo json_encode(true);
        break;

    case "s1_6400":

        exec ("gphoto2 --set-config shutterspeed2=1/6400", $output);
        echo json_encode(true);
        break;

    case "s1_5000":

        exec ("gphoto2 --set-config shutterspeed2=1/5000", $output);
        echo json_encode(true);
        break;

    case "s1_4000":

        exec ("gphoto2 --set-config shutterspeed2=1/4000", $output);
        echo json_encode(true);
        break;

    case "s1_3200":

        exec ("gphoto2 --set-config shutterspeed2=1/3200", $output);
        echo json_encode(true);
        break;

    case "s1_2500":

```

```

        exec ("gphoto2 --set-config shutterspeed2=1/2500", $output);
        echo json_encode(true);
        break;

    case "s1_2000":

        exec ("gphoto2 --set-config shutterspeed2=1/2000", $output);
        echo json_encode(true);
        break;

    case "s1_1600":

        exec ("gphoto2 --set-config shutterspeed2=1/1600", $output);
        echo json_encode(true);
        break;

    case "s1_1250":

        exec ("gphoto2 --set-config shutterspeed2=1/1250", $output);
        echo json_encode(true);
        break;

    case "s1_1000":

        exec ("gphoto2 --set-config shutterspeed2=1/1000", $output);
        echo json_encode(true);
        break;

    case "s1_800":

        exec ("gphoto2 --set-config shutterspeed2=1/800", $output);
        echo json_encode(true);
        break;

    case "s1_640":

        exec ("gphoto2 --set-config shutterspeed2=1/640", $output);
        echo json_encode(true);
        break;

    case "s1_500":

        exec ("gphoto2 --set-config shutterspeed2=1/500", $output);
        echo json_encode(true);
        break;

    case "s1_400":

        exec ("gphoto2 --set-config shutterspeed2=1/400", $output);
        echo json_encode(true);
        break;

    case "s1_320":

        exec ("gphoto2 --set-config shutterspeed2=1/320", $output);
        echo json_encode(true);
        break;

    case "s1_250":

        exec ("gphoto2 --set-config shutterspeed2=1/250", $output);
        echo json_encode(true);
        break;

    case "s1_200":

        exec ("gphoto2 --set-config shutterspeed2=1/200", $output);
        echo json_encode(true);
        break;

    case "s1_160":

```



```

    exec ("gphoto2 --set-config shutterspeed2=1/160", $output);
    echo json_encode(true);
    break;

case "s1_125":

    exec ("gphoto2 --set-config shutterspeed2=1/125", $output);
    echo json_encode(true);
    break;

case "s1_100":

    exec ("gphoto2 --set-config shutterspeed2=1/100", $output);
    echo json_encode(true);
    break;

case "s1_80":

    exec ("gphoto2 --set-config shutterspeed2=1/80", $output);
    echo json_encode(true);
    break;

case "s1_60":

    exec ("gphoto2 --set-config shutterspeed2=1/60", $output);
    echo json_encode(true);
    break;

case "s1_50":

    exec ("gphoto2 --set-config shutterspeed2=1/50", $output);
    echo json_encode(true);
    break;

case "s1_40":

    exec ("gphoto2 --set-config shutterspeed2=1/40", $output);
    echo json_encode(true);
    break;

case "s1_30":

    exec ("gphoto2 --set-config shutterspeed2=1/30", $output);
    echo json_encode(true);
    break;

case "s1_25":

    exec ("gphoto2 --set-config shutterspeed2=1/25", $output);
    echo json_encode(true);
    break;

case "s1_20":

    exec ("gphoto2 --set-config shutterspeed2=1/20", $output);
    echo json_encode(true);
    break;

case "s1_15":

    exec ("gphoto2 --set-config shutterspeed2=1/15", $output);
    echo json_encode(true);
    break;

case "s1_13":

    exec ("gphoto2 --set-config shutterspeed2=1/13", $output);
    echo json_encode(true);
    break;

```

```

    case "s1_10":
        exec ("gphoto2 --set-config shutterspeed2=1/10", $output);
        echo json_encode(true);
        break;

    case "s1_8":
        exec ("gphoto2 --set-config shutterspeed2=1/8", $output);
        echo json_encode(true);
        break;

    case "s1_6":
        exec ("gphoto2 --set-config shutterspeed2=1/6", $output);
        echo json_encode(true);
        break;

    case "s1_5":
        exec ("gphoto2 --set-config shutterspeed2=1/5", $output);
        echo json_encode(true);
        break;

    case "s1_4":
        exec ("gphoto2 --set-config shutterspeed2=1/4", $output);
        echo json_encode(true);
        break;

    case "s1_3":
        exec ("gphoto2 --set-config shutterspeed2=1/3", $output);
        echo json_encode(true);
        break;

    case "s10_25":
        exec ("gphoto2 --set-config shutterspeed2=10/25", $output);
        echo json_encode(true);
        break;

    case "s1_2":
        exec ("gphoto2 --set-config shutterspeed2=1/2", $output);
        echo json_encode(true);
        break;

    case "s10_16":
        exec ("gphoto2 --set-config shutterspeed2=10/16", $output);
        echo json_encode(true);
        break;

    case "s10_13":
        exec ("gphoto2 --set-config shutterspeed2=10/13", $output);
        echo json_encode(true);
        break;

    case "s1":
        exec ("gphoto2 --set-config shutterspeed2=1", $output);
        echo json_encode(true);
        break;

    case "s13_10":
        exec ("gphoto2 --set-config shutterspeed2=13/10", $output);
        echo json_encode(true);
        break;

```

```

case "s16_10":
    exec ("gphoto2 --set-config shutter speed=16/10", $output);
    echo json_encode(true);
    break;

case "s2":
    exec ("gphoto2 --set-config shutter speed=2", $output);
    echo json_encode(true);
    break;

case "s25_10":
    exec ("gphoto2 --set-config shutter speed=25/10", $output);
    echo json_encode(true);
    break;

case "s3":
    exec ("gphoto2 --set-config shutter speed=3", $output);
    echo json_encode(true);
    break;

case "s4":
    exec ("gphoto2 --set-config shutter speed=4", $output);
    echo json_encode(true);
    break;

case "s5":
    exec ("gphoto2 --set-config shutter speed=5", $output);
    echo json_encode(true);
    break;

case "s6":
    exec ("gphoto2 --set-config shutter speed=6", $output);
    echo json_encode(true);
    break;

case "s8":
    exec ("gphoto2 --set-config shutter speed=8", $output);
    echo json_encode(true);
    break;

case "s10":
    exec ("gphoto2 --set-config shutter speed=10", $output);
    echo json_encode(true);
    break;

case "s13":
    exec ("gphoto2 --set-config shutter speed=13", $output);
    echo json_encode(true);
    break;

case "s15":
    exec ("gphoto2 --set-config shutter speed=15", $output);
    echo json_encode(true);
    break;

case "s20":
    exec ("gphoto2 --set-config shutter speed=20", $output);
    echo json_encode(true);

```

```

        break;

    case "s25":

        exec ("gphoto2 --set-config shutterspeed2=25", $output);
        echo json_encode(true);
        break;

    case "s30":

        exec ("gphoto2 --set-config shutterspeed2=30", $output);
        echo json_encode(true);
        break;

    case "sx250":

        exec ("gphoto2 --set-config shutterspeed2=55", $output);
        echo json_encode(true);
        break;

    case "bulb":

        exec ("gphoto2 --set-config shutterspeed2=56", $output);
        echo json_encode(true);
        break;

    case "getautoiso":

        exec ("gphoto2 --get-config autoiso", $output);
        $routput= array_reverse($output);
        $returnObj->camera = ($routput[count($routput) - 4]);
        header('Content-Type: application/json');
        echo json_encode($returnObj);
        break;

    case "autoiso_on":

        exec ("gphoto2 --set-config autoiso=0", $output);
        echo json_encode(true);
        break;

    case "autoiso_off":

        exec ("gphoto2 --set-config autoiso=1", $output);
        echo json_encode(true);
        break;

    case "getiso":

        exec ("gphoto2 --get-config iso", $output);
        $routput= array_reverse($output);
        $returnObj->camera = ($routput[count($routput) - 4]);
        header('Content-Type: application/json');
        echo json_encode($returnObj);
        break;

    case "iso_100":

        exec ("gphoto2 --set-config iso=100", $output);
        echo json_encode(true);
        break;

    case "iso_125":

        exec ("gphoto2 --set-config iso=125", $output);
        echo json_encode(true);
        break;

    case "iso_160":

        exec ("gphoto2 --set-config iso=160", $output);

```

```

        echo json_encode(true);
        break;

    case "iso_200":

        exec ("gphoto2 --set-config iso=200", $output);
        echo json_encode(true);
        break;

    case "iso_250":

        exec ("gphoto2 --set-config iso=250", $output);
        echo json_encode(true);
        break;

    case "iso_320":

        exec ("gphoto2 --set-config iso=320", $output);
        echo json_encode(true);
        break;

    case "iso_400":

        exec ("gphoto2 --set-config iso=400", $output);
        echo json_encode(true);
        break;

    case "iso_500":

        exec ("gphoto2 --set-config iso=500", $output);
        echo json_encode(true);
        break;

    case "iso_640":

        exec ("gphoto2 --set-config iso=640", $output);
        echo json_encode(true);
        break;

    case "iso_800":

        exec ("gphoto2 --set-config iso=800", $output);
        echo json_encode(true);
        break;

    case "iso_1000":

        exec ("gphoto2 --set-config iso=1000", $output);
        echo json_encode(true);
        break;

    case "iso_1250":

        exec ("gphoto2 --set-config iso=1250", $output);
        echo json_encode(true);
        break;

    case "iso_1600":

        exec ("gphoto2 --set-config iso=1600", $output);
        echo json_encode(true);
        break;

    case "iso_2000":

        exec ("gphoto2 --set-config iso=2000", $output);
        echo json_encode(true);
        break;

    case "iso_2500":

```

```

        exec ("gphoto2 --set-config iso=2500", $output);
        echo json_encode(true);
        break;

    case "iso_3200":

        exec ("gphoto2 --set-config iso=3200", $output);
        echo json_encode(true);
        break;

    case "iso_4000":

        exec ("gphoto2 --set-config iso=4000", $output);
        echo json_encode(true);
        break;

    case "iso_5000":

        exec ("gphoto2 --set-config iso=5000", $output);
        echo json_encode(true);
        break;

    case "iso_6400":

        exec ("gphoto2 --set-config iso=6400", $output);
        echo json_encode(true);
        break;

    case "iso_8320":

        exec ("gphoto2 --set-config iso=8320", $output);
        echo json_encode(true);
        break;

    case "iso_10240":

        exec ("gphoto2 --set-config iso=10240", $output);
        echo json_encode(true);
        break;

    case "iso_12800":

        exec ("gphoto2 --set-config iso=12800", $output);
        echo json_encode(true);
        break;

    case "iso_25600":

        exec ("gphoto2 --set-config iso=25600", $output);
        echo json_encode(true);
        break;

    case "getexpcomp":

        exec ("gphoto2 --get-config exposurecompensation", $output);
        $routput= array_reverse($output);
        $returnObj->camera = ($routput[count($routput) - 4]);
        header('Content-Type: application/json');
        echo json_encode($returnObj);
        break;

    case "comp_50":

        exec ("gphoto2 --set-config exposurecompensation=5", $output);
        echo json_encode(true);
        break;

    case "comp_47":

        exec ("gphoto2 --set-config exposurecompensation=4.666", $output);
        echo json_encode(true);

```

```

        break;

    case "comp_43":

        exec ("gphoto2 --set-config exposurecompensation=4.333", $output);
        echo json_encode(true);
        break;

    case "comp_40":

        exec ("gphoto2 --set-config exposurecompensation=4", $output);
        echo json_encode(true);
        break;

    case "comp_37":

        exec ("gphoto2 --set-config exposurecompensation=3.666", $output);
        echo json_encode(true);
        break;

    case "comp_33":

        exec ("gphoto2 --set-config exposurecompensation=3.333", $output);
        echo json_encode(true);
        break;

    case "comp_30":

        exec ("gphoto2 --set-config exposurecompensation=3", $output);
        echo json_encode(true);
        break;

    case "comp_27":

        exec ("gphoto2 --set-config exposurecompensation=2.666", $output);
        echo json_encode(true);
        break;

    case "comp_23":

        exec ("gphoto2 --set-config exposurecompensation=2.333", $output);
        echo json_encode(true);
        break;

    case "comp_20":

        exec ("gphoto2 --set-config exposurecompensation=2", $output);
        echo json_encode(true);
        break;

    case "comp_17":

        exec ("gphoto2 --set-config exposurecompensation=1.666", $output);
        echo json_encode(true);
        break;

    case "comp_13":

        exec ("gphoto2 --set-config exposurecompensation=1.333", $output);
        echo json_encode(true);
        break;

    case "comp_10":

        exec ("gphoto2 --set-config exposurecompensation=1", $output);
        echo json_encode(true);
        break;

    case "comp_07":

        exec ("gphoto2 --set-config exposurecompensation=0.666", $output);

```

```

        echo json_encode(true);
        break;

    case "comp_03":

        exec ("gphoto2 --set-config exposurecompensation=0.333", $output);
        echo json_encode(true);
        break;

    case "comp_00":

        exec ("gphoto2 --set-config exposurecompensation=0", $output);
        echo json_encode(true);
        break;

    case "comp_m03":

        exec ("gphoto2 --set-config exposurecompensation=-0.333", $output);
        echo json_encode(true);
        break;

    case "comp_m07":

        exec ("gphoto2 --set-config exposurecompensation=-0.666", $output);
        echo json_encode(true);
        break;

    case "comp_m10":

        exec ("gphoto2 --set-config exposurecompensation=-1", $output);
        echo json_encode(true);
        break;

    case "comp_m13":

        exec ("gphoto2 --set-config exposurecompensation=-1.333", $output);
        echo json_encode(true);
        break;

    case "comp_m17":

        exec ("gphoto2 --set-config exposurecompensation=-1.666", $output);
        echo json_encode(true);
        break;

    case "comp_m20":

        exec ("gphoto2 --set-config exposurecompensation=-2", $output);
        echo json_encode(true);
        break;

    case "comp_m23":

        exec ("gphoto2 --set-config exposurecompensation=-2.333", $output);
        echo json_encode(true);
        break;

    case "comp_m27":

        exec ("gphoto2 --set-config exposurecompensation=-2.666", $output);
        echo json_encode(true);
        break;

    case "comp_m30":

        exec ("gphoto2 --set-config exposurecompensation=-3", $output);
        echo json_encode(true);
        break;

    case "comp_m33":

```



```

        exec ("gphoto2 --set-config exposurecompensation=-3.333", $output);
        echo json_encode(true);
        break;

    case "comp_m37":

        exec ("gphoto2 --set-config exposurecompensation=-3.666", $output);
        echo json_encode(true);
        break;

    case "comp_m40":

        exec ("gphoto2 --set-config exposurecompensation=-4", $output);
        echo json_encode(true);
        break;

    case "comp_m43":

        exec ("gphoto2 --set-config exposurecompensation=-4.333", $output);
        echo json_encode(true);
        break;

    case "comp_m47":

        exec ("gphoto2 --set-config exposurecompensation=-4.666", $output);
        echo json_encode(true);
        break;

    case "comp_m50":

        exec ("gphoto2 --set-config exposurecompensation=-5", $output);
        echo json_encode(true);
        break;

    case "getflex":

        exec ("gphoto2 --get-config flexibleprogram", $output);
        $routput= array_reverse($output);
        $returnObj->camera = ($routput[count($routput) - 4]);
        header('Content-Type: application/json');
        echo json_encode($returnObj);
        break;

    case "shift_30":

        exec ("gphoto2 --set-config flexibleprogram=30", $output);
        echo json_encode(true);
        break;

    case "shift_28":

        exec ("gphoto2 --set-config flexibleprogram=28", $output);
        echo json_encode(true);
        break;

    case "shift_26":

        exec ("gphoto2 --set-config flexibleprogram=26", $output);
        echo json_encode(true);
        break;

    case "shift_24":

        exec ("gphoto2 --set-config flexibleprogram=24", $output);
        echo json_encode(true);
        break;

    case "shift_22":

        exec ("gphoto2 --set-config flexibleprogram=22", $output);
        echo json_encode(true);

```

```

        break;

    case "shift_20":

        exec ("gphoto2 --set-config flexibleprogram=20", $output);
        echo json_encode(true);
        break;

    case "shift_18":

        exec ("gphoto2 --set-config flexibleprogram=18", $output);
        echo json_encode(true);
        break;

    case "shift_16":

        exec ("gphoto2 --set-config flexibleprogram=16", $output);
        echo json_encode(true);
        break;

    case "shift_14":

        exec ("gphoto2 --set-config flexibleprogram=14", $output);
        echo json_encode(true);
        break;

    case "shift_12":

        exec ("gphoto2 --set-config flexibleprogram=12", $output);
        echo json_encode(true);
        break;

    case "shift_10":

        exec ("gphoto2 --set-config flexibleprogram=10", $output);
        echo json_encode(true);
        break;

    case "shift_8":

        exec ("gphoto2 --set-config flexibleprogram=8", $output);
        echo json_encode(true);
        break;

    case "shift_6":

        exec ("gphoto2 --set-config flexibleprogram=6", $output);
        echo json_encode(true);
        break;

    case "shift_4":

        exec ("gphoto2 --set-config flexibleprogram=4", $output);
        echo json_encode(true);
        break;

    case "shift_2":

        exec ("gphoto2 --set-config flexibleprogram=2", $output);
        echo json_encode(true);
        break;

    case "shift_0":

        exec ("gphoto2 --set-config flexibleprogram=0", $output);
        echo json_encode(true);
        break;

    case "shift_m2":

        exec ("gphoto2 --set-config flexibleprogram=-2", $output);

```

```

        echo json_encode(true);
        break;

    case "shift_m4":

        exec ("gphoto2 --set-config flexibleprogram=-4", $output);
        echo json_encode(true);
        break;

    case "shift_m6":

        exec ("gphoto2 --set-config flexibleprogram=-6", $output);
        echo json_encode(true);
        break;

    case "shift_m8":

        exec ("gphoto2 --set-config flexibleprogram=-8", $output);
        echo json_encode(true);
        break;

    case "shift_m10":

        exec ("gphoto2 --set-config flexibleprogram=-10", $output);
        echo json_encode(true);
        break;

    case "shift_m12":

        exec ("gphoto2 --set-config flexibleprogram=-12", $output);
        echo json_encode(true);
        break;

    case "shift_m14":

        exec ("gphoto2 --set-config flexibleprogram=-14", $output);
        echo json_encode(true);
        break;

    case "shift_m16":

        exec ("gphoto2 --set-config flexibleprogram=-16", $output);
        echo json_encode(true);
        break;

    case "shift_m18":

        exec ("gphoto2 --set-config flexibleprogram=-18", $output);
        echo json_encode(true);
        break;

    case "shift_m20":

        exec ("gphoto2 --set-config flexibleprogram=-20", $output);
        echo json_encode(true);
        break;

    case "shift_m22":

        exec ("gphoto2 --set-config flexibleprogram=-22", $output);
        echo json_encode(true);
        break;

    case "shift_m24":

        exec ("gphoto2 --set-config flexibleprogram=-24", $output);
        echo json_encode(true);
        break;

    case "shift_m26":

```

```

        exec ("gphoto2 --set-config flexibleprogram=-26", $output);
        echo json_encode(true);
        break;

    case "shift_m28":

        exec ("gphoto2 --set-config flexibleprogram=-28", $output);
        echo json_encode(true);
        break;

    case "shift_m30":

        exec ("gphoto2 --set-config flexibleprogram=-30", $output);
        echo json_encode(true);
        break;

    case "getquality":

        exec ("gphoto2 --get-config imagequality", $output);
        $routput= array_reverse($output);
        $returnObj->camera = ($routput[count($routput) - 4]);
        header('Content-Type: application/json');
        echo json_encode($returnObj);
        break;

    case "basic":

        exec ("gphoto2 --set-config imagequality=0", $output);
        echo json_encode(true);
        break;

    case "normal":

        exec ("gphoto2 --set-config imagequality=1", $output);
        echo json_encode(true);
        break;

    case "fine":

        exec ("gphoto2 --set-config imagequality=2", $output);
        echo json_encode(true);
        break;

    case "tiff":

        exec ("gphoto2 --set-config imagequality=3", $output);
        echo json_encode(true);
        break;

    case "raw":

        exec ("gphoto2 --set-config imagequality=4", $output);
        echo json_encode(true);
        break;

    case "rbasic":

        exec ("gphoto2 --set-config imagequality=5", $output);
        echo json_encode(true);
        break;

    case "rnormal":

        exec ("gphoto2 --set-config imagequality=6", $output);
        echo json_encode(true);
        break;

    case "rfine":

        exec ("gphoto2 --set-config imagequality=7", $output);
        echo json_encode(true);

```

```

        break;

    case "getsize":

        exec ("gphoto2 --get-config imagesize", $output);
        $routput= array_reverse($output);
        $returnObj->camera = ($routput[count($routput) - 4]);
        header('Content-Type: application/json');
        echo json_encode($returnObj);
        break;

    case "large":

        exec ("gphoto2 --set-config imagesize=0", $output);
        echo json_encode(true);
        break;

    case "med":

        exec ("gphoto2 --set-config imagesize=1", $output);
        echo json_encode(true);
        break;

    case "small":

        exec ("gphoto2 --set-config imagesize=2", $output);
        echo json_encode(true);
        break;

    case "getrsize":

        exec ("gphoto2 --get-config rawcompression", $output);
        $routput= array_reverse($output);
        $returnObj->camera = ($routput[count($routput) - 4]);
        header('Content-Type: application/json');
        echo json_encode($returnObj);
        break;

    case "lossless":

        exec ("gphoto2 --set-config rawcompression=0", $output);
        echo json_encode(true);
        break;

    case "lossy":

        exec ("gphoto2 --set-config rawcompression=1", $output);
        echo json_encode(true);
        break;

    case "comp_off":

        exec ("gphoto2 --set-config rawcompression=2", $output);
        echo json_encode(true);
        break;

    case "getwbalance":

        exec ("gphoto2 --get-config whitebalance", $output);
        $routput= array_reverse($output);
        $returnObj->camera = ($routput[count($routput) - 4]);
        header('Content-Type: application/json');
        echo json_encode($returnObj);
        break;

    case "wauto":

        exec ("gphoto2 --set-config whitebalance=0", $output);
        echo json_encode(true);
        break;

```

```

case "day1":
    exec ("gphoto2 --set-config whitebalance=1", $output);
    echo json_encode(true);
    break;

case "fluo":
    exec ("gphoto2 --set-config whitebalance=2", $output);
    echo json_encode(true);
    break;

case "tung":
    exec ("gphoto2 --set-config whitebalance=3", $output);
    echo json_encode(true);
    break;

case "wflash":
    exec ("gphoto2 --set-config whitebalance=4", $output);
    echo json_encode(true);
    break;

case "cloudy":
    exec ("gphoto2 --set-config whitebalance=5", $output);
    echo json_encode(true);
    break;

case "shade":
    exec ("gphoto2 --set-config whitebalance=6", $output);
    echo json_encode(true);
    break;

case "color_temp":
    exec ("gphoto2 --set-config whitebalance=7", $output);
    echo json_encode(true);
    break;

case "wpreset":
    exec ("gphoto2 --set-config whitebalance=8", $output);
    echo json_encode(true);
    break;

case "getcolorspace":
    exec ("gphoto2 --get-config colorspace", $output);
    $routput= array_reverse($output);
    $returnObj->camera = ($routput[count($routput) - 4]);
    header('Content-Type: application/json');
    echo json_encode($returnObj);
    break;

case "srgb":
    exec ("gphoto2 --set-config colorspace=0", $output);
    echo json_encode(true);
    break;

case "adobe":
    exec ("gphoto2 --set-config colorspace=1", $output);
    echo json_encode(true);
    break;

case "gethisonr":

```

```
exec ("gphoto2 --get-config highisonr", $output);
$routput= array_reverse($output);
$returnObj->camera = ($routput[count($routput) - 4]);
header('Content-Type: application/json');
echo json_encode($returnObj);
break;
```

```
case "hnroff":
```

```
exec ("gphoto2 --set-config highisonr=0", $output);
echo json_encode(true);
break;
```

```
case "hnr1ow":
```

```
exec ("gphoto2 --set-config highisonr=1", $output);
echo json_encode(true);
break;
```

```
case "hnrnorm":
```

```
exec ("gphoto2 --set-config highisonr=2", $output);
echo json_encode(true);
break;
```

```
case "hnrhigh":
```

```
exec ("gphoto2 --set-config highisonr=3", $output);
echo json_encode(true);
break;
```

```
case "getlnr":
```

```
exec ("gphoto2 --get-config longexpnr", $output);
$routput= array_reverse($output);
$returnObj->camera = ($routput[count($routput) - 4]);
header('Content-Type: application/json');
echo json_encode($returnObj);
break;
```

```
case "lnron":
```

```
exec ("gphoto2 --set-config longexpnr=0", $output);
echo json_encode(true);
break;
```

```
case "lnroff":
```

```
exec ("gphoto2 --set-config longexpnr=1", $output);
echo json_encode(true);
break;
```

```
case "getflashmode":
```

```
exec ("gphoto2 --get-config flashmode", $output);
$routput= array_reverse($output);
$returnObj->camera = ($routput[count($routput) - 4]);
header('Content-Type: application/json');
echo json_encode($returnObj);
break;
```

```
case "flredauto":
```

```
exec ("gphoto2 --set-config flashmode=0", $output);
echo json_encode(true);
break;
```

```
case "flauto":
```

```
exec ("gphoto2 --set-config flashmode=1", $output);
echo json_encode(true);
```

```

        break;

    case "flautoslow":

        exec ("gphoto2 --set-config flashmode=2", $output);
        echo json_encode(true);
        break;

    case "flrearcertainslow":

        exec ("gphoto2 --set-config flashmode=3", $output);
        echo json_encode(true);
        break;

    case "flredslow":

        exec ("gphoto2 --set-config flashmode=4", $output);
        echo json_encode(true);
        break;

    case "getflashesyncspeed":

        exec ("gphoto2 --get-config flashesyncspeed", $output);
        $routput= array_reverse($output);
        $returnObj->camera = ($routput[count($routput) - 4]);
        header('Content-Type: application/json');
        echo json_encode($returnObj);
        break;

    case "autofp":

        exec ("gphoto2 --set-config flashesyncspeed=0", $output);
        echo json_encode(true);
        break;

    case "f1_250":

        exec ("gphoto2 --set-config flashesyncspeed=1", $output);
        echo json_encode(true);
        break;

    case "f1_200":

        exec ("gphoto2 --set-config flashesyncspeed=2", $output);
        echo json_encode(true);
        break;

    case "f1_160":

        exec ("gphoto2 --set-config flashesyncspeed=3", $output);
        echo json_encode(true);
        break;

    case "f1_125":

        exec ("gphoto2 --set-config flashesyncspeed=4", $output);
        echo json_encode(true);
        break;

    case "f1_100":

        exec ("gphoto2 --set-config flashesyncspeed=5", $output);
        echo json_encode(true);
        break;

    case "f1_80":

        exec ("gphoto2 --set-config flashesyncspeed=6", $output);
        echo json_encode(true);
        break;

```



```

case "f1_60":

    exec ("gphoto2 --set-config flashsyncspeed=7", $output);
    echo json_encode(true);
    break;

case "getflashshutterspeed":

    exec ("gphoto2 --get-config flashshutterspeed", $output);
    $routput= array_reverse($output);
    $returnObj->camera = ($routput[count($routput) - 4]);
    header('Content-Type: application/json');
    echo json_encode($returnObj);
    break;

case "fs1_60":

    exec ("gphoto2 --set-config flashshutterspeed=0", $output);
    echo json_encode(true);
    break;

case "fs1_30":

    exec ("gphoto2 --set-config flashshutterspeed=1", $output);
    echo json_encode(true);
    break;

case "fs1_15":

    exec ("gphoto2 --set-config flashshutterspeed=2", $output);
    echo json_encode(true);
    break;

case "fs1_8":

    exec ("gphoto2 --set-config flashshutterspeed=3", $output);
    echo json_encode(true);
    break;

case "fs1_4":

    exec ("gphoto2 --set-config flashshutterspeed=4", $output);
    echo json_encode(true);
    break;

case "fs1_2":

    exec ("gphoto2 --set-config flashshutterspeed=5", $output);
    echo json_encode(true);
    break;

case "fs1":

    exec ("gphoto2 --set-config flashshutterspeed=6", $output);
    echo json_encode(true);
    break;

case "fs2":

    exec ("gphoto2 --set-config flashshutterspeed=7", $output);
    echo json_encode(true);
    break;

case "fs4":

    exec ("gphoto2 --set-config flashshutterspeed=8", $output);
    echo json_encode(true);
    break;

case "fs8":

```

```

        exec ("gphoto2 --set-config flashshutterspeed=9",$output);
        echo json_encode(true);
        break;

    case "fs15":

        exec ("gphoto2 --set-config flashshutterspeed=10",$output);
        echo json_encode(true);
        break;

    case "fs30":

        exec ("gphoto2 --set-config flashshutterspeed=11",$output);
        echo json_encode(true);
        break;

    case "getCamera":

        exec ("gphoto2 --auto-detect", $output);
        $returnObj->camera = trim(explode("usb", $output[count($output) - 1])[0]);
        header('Content-Type: application/json');
        echo json_encode($returnObj);
        break;

    case "getImages":

        $files = array();
        $imageDir = opendir('images');
        while (($file = readdir($imageDir)) !== false) {
            if(!is_dir('images/'.$file)){
                $path_parts = pathinfo('images/'.$file);
                if (!file_exists('images/thumbs/'.$path_parts['basename'].'.jpg')){
                    try { //try to extract the preview image from the RAW
                        CameraRaw::extractPreview('images/'.$file,
'images/thumbs/'.$path_parts['basename'].'.jpg');
                    } catch (Exception $e) { //else resize the image...
                        $sim = new Imagick('images/'.$file);
                        $sim->setImageFormat('jpg');
                        $sim->scaleImage(1024,0);
                        $sim->writeImage('images/thumbs/'.$path_parts['basename'].'.jpg');
                        $sim->clear();
                        $sim->destroy();
                    }
                }
                $returnFile;
                $returnFile->name = $path_parts['basename'];
                $returnFile->sourcePath = 'images/'.$file;
                $returnFile->thumbPath =
'images/thumbs/'.$path_parts['basename'].'.jpg';

                array_push($files,$returnFile);

                unset($returnFile);
            }
        }
        closedir($imageDir);
        $returnObj = $files;
        header('Content-Type: application/json');
        echo json_encode($returnObj);
        break;
    default:
        break;
}
} catch (Exception $e) { //else resize the image...
}
?>

```

Controller.js

```
//var galleryImages;

$(document).on( 'pageinit',function(event) {
    getCamera();
    getexpmode();
    apsize();
    getshutterspeed();
    getautoiso();
    getiso();
    getexpcomp();
    getflex();
    getquality();
    getsize();
    getrsize();
    getwbalance();
    getcolorspace();
    gethisonr();
    getlnr();
    getflashmode();
    getflashsyncspeed();
    getflashshutterspeed();
});

function takePicture(){

    $.mobile.loading( 'show', {
        text: 'Taking Image...',
        textVisible: true,
        theme: 'a'
    });

    $.ajax({
        url: "service.php?action=takePicture",
        dataType : "json",
        success: function(data){
            $.mobile.loading( 'hide');
        },
    });
}

$(document).on( "pageshow", "#gallery", function( event ) {
    $.ajax({
        url: "service.php?action=getImages",
        dataType : "json",
        success: function(data){
            updateGalleryGrid(data);
        },
    });
})

function updateGalleryGrid(data){
    //$("#galleryGrid").html("");

    var galleryHTML = "";

    for(var i = 0; i < data.length; i++){

        var uiClass = "a";

        if (i % 2 == 1){
            uiClass = "b";
        }
    }
}
```

```

    }

    var image = data[i];

    var id = image.name.replace(/[-\.]/g, '');

    if ($('#' + id).length > 0){
        $('#' + id).removeClass("ui-block-a");
        $('#' + id).removeClass("ui-block-b");
        $('#' + id).addClass("ui-block-" + uiClass);
    }else{
        var galleryTemplate = $("#galleryTemplate").text();
        galleryTemplate = galleryTemplate.replace(/@imageThumb/g, image.thumbPath);
        galleryTemplate = galleryTemplate.replace(/@char/g, uiClass);
        galleryTemplate = galleryTemplate.replace(/@sourceURL/g, image.sourcePath);
        galleryTemplate = galleryTemplate.replace(/@imageName/g, image.name);
        galleryTemplate = galleryTemplate.replace(/@id/g, id);
        $("#galleryGrid").append(galleryTemplate);
    }
}

$(document).on( 'pageinit', function(event) {
    getCamera();
});

function deleteFile(file){

    var id = file.replace(/[-\.]/g, '');
    $('#' + id).remove();

    $.ajax({
        url: "service.php?action=deleteFile&file=" + file,
        dataType: "json",
        success: function(data) {
            $.ajax({
                url: "service.php?action=getImages",
                dataType: "json",
                success: function(data) {
                    updateGalleryGrid(data);
                },
            });
        },
    });
}

function program(){
    $.ajax({
        url: "service.php?action=program",
        dataType: "json",
        success: function(data) {
            $.mobile.loading( 'hide' );
        },
    });
}

function shutter(){
    $.ajax({
        url: "service.php?action=shutter",
        dataType: "json",
        success: function(data) {
            $.mobile.loading( 'hide' );
        },
    });
}

function aperture(){
    $.ajax({
        url: "service.php?action=aperture",
        dataType: "json",

```

```

        success: function(data){
            $.mobile.loading( 'hide');
        },
    });
}

function manual(){
    $.ajax({
        url: "service.php?action=manual",
        dataType : "json",
        success: function(data){
            $.mobile.loading( 'hide');
        },
    });
}

function getexpmode(){
    $.ajax({
        url: "service.php?action=getexpmode",
        dataType : "json",
        success: function(data){
            $("#expmode").html(data.camera);
        },
    });
}

function apsize(){
    $.ajax({
        url: "service.php?action=apsize",
        dataType : "json",
        success: function(data){
            $("#apsize").html(data.camera);
        },
    });
}

function f2_8(){
    $.ajax({
        url: "service.php?action=f2.8",
        dataType : "json",
        success: function(data){
            $.mobile.loading( 'hide');
        },
    });
}

function f3_2(){
    $.ajax({
        url: "service.php?action=f3.2",
        dataType : "json",
        success: function(data){
            $.mobile.loading( 'hide');
        },
    });
}

function f3_5(){
    $.ajax({
        url: "service.php?action=f3.5",
        dataType : "json",
        success: function(data){
            $.mobile.loading( 'hide');
        },
    });
}

function f4(){
    $.ajax({
        url: "service.php?action=f4",
        dataType : "json",
        success: function(data){

```

```

        $.mobile.loading( 'hide');
    },
    });
}

function f4_5(){
$.ajax({
    url: "service.php?action=f4.5",
    dataType : "json",
    success: function(data){
        $.mobile.loading( 'hide');
    },
    });
}

function f5(){
$.ajax({
    url: "service.php?action=f5",
    dataType : "json",
    success: function(data){
        $.mobile.loading( 'hide');
    },
    });
}

function f5_6(){
$.ajax({
    url: "service.php?action=f5.6",
    dataType : "json",
    success: function(data){
        $.mobile.loading( 'hide');
    },
    });
}

function f6_3(){
$.ajax({
    url: "service.php?action=f6.3",
    dataType : "json",
    success: function(data){
        $.mobile.loading( 'hide');
    },
    });
}

function f7_1(){
$.ajax({
    url: "service.php?action=f7.1",
    dataType : "json",
    success: function(data){
        $.mobile.loading( 'hide');
    },
    });
}

function f8(){
$.ajax({
    url: "service.php?action=f8",
    dataType : "json",
    success: function(data){
        $.mobile.loading( 'hide');
    },
    });
}

function f9(){
$.ajax({
    url: "service.php?action=f9",
    dataType : "json",
    success: function(data){
        $.mobile.loading( 'hide');
    }

```

```

    },
  });
}

function f10(){
$.ajax({
  url: "service.php?action=f10",
  dataType : "json",
  success: function(data){
    $.mobile.loading( 'hide');
  },
});
}

function f11(){
$.ajax({
  url: "service.php?action=f11",
  dataType : "json",
  success: function(data){
    $.mobile.loading( 'hide');
  },
});
}

function f13(){
$.ajax({
  url: "service.php?action=f13",
  dataType : "json",
  success: function(data){
    $.mobile.loading( 'hide');
  },
});
}

function f14(){
$.ajax({
  url: "service.php?action=f14",
  dataType : "json",
  success: function(data){
    $.mobile.loading( 'hide');
  },
});
}

function f16(){
$.ajax({
  url: "service.php?action=f16",
  dataType : "json",
  success: function(data){
    $.mobile.loading( 'hide');
  },
});
}

function f18(){
$.ajax({
  url: "service.php?action=f18",
  dataType : "json",
  success: function(data){
    $.mobile.loading( 'hide');
  },
});
}

function f20(){
$.ajax({
  url: "service.php?action=f20",
  dataType : "json",
  success: function(data){
    $.mobile.loading( 'hide');
  },
});
}

```

```

    });
}

function f22(){
$.ajax({
    url: "service.php?action=f22",
    dataType : "json",
    success: function(data){
        $.mobile.loading( 'hide');
    },
});
}

function f25(){
$.ajax({
    url: "service.php?action=f25",
    dataType : "json",
    success: function(data){
        $.mobile.loading( 'hide');
    },
});
}

function f29(){
$.ajax({
    url: "service.php?action=f29",
    dataType : "json",
    success: function(data){
        $.mobile.loading( 'hide');
    },
});
}

function f32(){
$.ajax({
    url: "service.php?action=f32",
    dataType : "json",
    success: function(data){
        $.mobile.loading( 'hide');
    },
});
}

function getshutterspeed(){
$.ajax({
    url: "service.php?action=getshutterspeed",
    dataType : "json",
    success: function(data){
        $("#shutterspeed").html(data.camera);
    },
});
}

function s1_8000(){
$.ajax({
    url: "service.php?action=s1_8000",
    dataType : "json",
    success: function(data){
        $.mobile.loading( 'hide');
    },
});
}

function s1_6400(){
$.ajax({
    url: "service.php?action=s1_6400",
    dataType : "json",
    success: function(data){
        $.mobile.loading( 'hide');
    },
});
}

```



```

}

function s1_5000(){
$.ajax({
url: "service.php?action=s1_5000",
dataType : "json",
success: function(data){
$.mobile.loading( 'hide');
},
});
}

function s1_4000(){
$.ajax({
url: "service.php?action=s1_4000",
dataType : "json",
success: function(data){
$.mobile.loading( 'hide');
},
});
}

function s1_3200(){
$.ajax({
url: "service.php?action=s1_3200",
dataType : "json",
success: function(data){
$.mobile.loading( 'hide');
},
});
}

function s1_2500(){
$.ajax({
url: "service.php?action=s1_2500",
dataType : "json",
success: function(data){
$.mobile.loading( 'hide');
},
});
}

function s1_2000(){
$.ajax({
url: "service.php?action=s1_2000",
dataType : "json",
success: function(data){
$.mobile.loading( 'hide');
},
});
}

function s1_1600(){
$.ajax({
url: "service.php?action=s1_1600",
dataType : "json",
success: function(data){
$.mobile.loading( 'hide');
},
});
}

function s1_1250(){
$.ajax({
url: "service.php?action=s1_1250",
dataType : "json",
success: function(data){
$.mobile.loading( 'hide');
},
});
}

```

```

function s1_1000(){
$.ajax({
url: "service.php?action=s1_1000",
dataType : "json",
success: function(data){
$.mobile.loading( 'hide');
},
});
}

function s1_800(){
$.ajax({
url: "service.php?action=s1_800",
dataType : "json",
success: function(data){
$.mobile.loading( 'hide');
},
});
}

function s1_640(){
$.ajax({
url: "service.php?action=s1_640",
dataType : "json",
success: function(data){
$.mobile.loading( 'hide');
},
});
}

function s1_500(){
$.ajax({
url: "service.php?action=s1_500",
dataType : "json",
success: function(data){
$.mobile.loading( 'hide');
},
});
}

function s1_400(){
$.ajax({
url: "service.php?action=s1_400",
dataType : "json",
success: function(data){
$.mobile.loading( 'hide');
},
});
}

function s1_320(){
$.ajax({
url: "service.php?action=s1_320",
dataType : "json",
success: function(data){
$.mobile.loading( 'hide');
},
});
}

function s1_250(){
$.ajax({
url: "service.php?action=s1_250",
dataType : "json",
success: function(data){
$.mobile.loading( 'hide');
},
});
}

```

```

function s1_200(){
    $.ajax({
        url: "service.php?action=s1_200",
        dataType : "json",
        success: function(data){
            $.mobile.loading( 'hide');
        },
    });
}

function s1_160(){
    $.ajax({
        url: "service.php?action=s1_160",
        dataType : "json",
        success: function(data){
            $.mobile.loading( 'hide');
        },
    });
}

function s1_125(){
    $.ajax({
        url: "service.php?action=s1_125",
        dataType : "json",
        success: function(data){
            $.mobile.loading( 'hide');
        },
    });
}

function s1_100(){
    $.ajax({
        url: "service.php?action=s1_100",
        dataType : "json",
        success: function(data){
            $.mobile.loading( 'hide');
        },
    });
}

function s1_80(){
    $.ajax({
        url: "service.php?action=s1_80",
        dataType : "json",
        success: function(data){
            $.mobile.loading( 'hide');
        },
    });
}

function s1_60(){
    $.ajax({
        url: "service.php?action=s1_60",
        dataType : "json",
        success: function(data){
            $.mobile.loading( 'hide');
        },
    });
}

function s1_50(){
    $.ajax({
        url: "service.php?action=s1_50",
        dataType : "json",
        success: function(data){
            $.mobile.loading( 'hide');
        },
    });
}

function s1_40(){

```

```

$.ajax({
  url: "service.php?action=s1_40",
  dataType : "json",
  success: function(data){
    $.mobile.loading( 'hide');
  },
});
}

function s1_30(){
$.ajax({
  url: "service.php?action=s1_30",
  dataType : "json",
  success: function(data){
    $.mobile.loading( 'hide');
  },
});
}

function s1_25(){
$.ajax({
  url: "service.php?action=s1_25",
  dataType : "json",
  success: function(data){
    $.mobile.loading( 'hide');
  },
});
}

function s1_20(){
$.ajax({
  url: "service.php?action=s1_20",
  dataType : "json",
  success: function(data){
    $.mobile.loading( 'hide');
  },
});
}

function s1_15(){
$.ajax({
  url: "service.php?action=s1_15",
  dataType : "json",
  success: function(data){
    $.mobile.loading( 'hide');
  },
});
}

function s1_13(){
$.ajax({
  url: "service.php?action=s1_13",
  dataType : "json",
  success: function(data){
    $.mobile.loading( 'hide');
  },
});
}

function s1_10(){
$.ajax({
  url: "service.php?action=s1_10",
  dataType : "json",
  success: function(data){
    $.mobile.loading( 'hide');
  },
});
}

function s1_8(){
$.ajax({

```

```

        url: "service.php?action=s1_8",
        dataType : "json",
        success: function(data){
            $.mobile.loading( 'hide' );
        },
    });
}

function s1_6(){
    $.ajax({
        url: "service.php?action=s1_6",
        dataType : "json",
        success: function(data){
            $.mobile.loading( 'hide' );
        },
    });
}

function s1_5(){
    $.ajax({
        url: "service.php?action=s1_5",
        dataType : "json",
        success: function(data){
            $.mobile.loading( 'hide' );
        },
    });
}

function s1_4(){
    $.ajax({
        url: "service.php?action=s1_4",
        dataType : "json",
        success: function(data){
            $.mobile.loading( 'hide' );
        },
    });
}

function s1_3(){
    $.ajax({
        url: "service.php?action=s1_3",
        dataType : "json",
        success: function(data){
            $.mobile.loading( 'hide' );
        },
    });
}

function s10_25(){
    $.ajax({
        url: "service.php?action=s10_25",
        dataType : "json",
        success: function(data){
            $.mobile.loading( 'hide' );
        },
    });
}

function s1_2(){
    $.ajax({
        url: "service.php?action=s1_2",
        dataType : "json",
        success: function(data){
            $.mobile.loading( 'hide' );
        },
    });
}

function s10_16(){
    $.ajax({
        url: "service.php?action=s10_16",

```

```

        dataType : "json",
        success: function(data){
            $.mobile.loading( 'hide');
        },
    });
}

function s10_13(){
$.ajax({
    url: "service.php?action=s10_13",
    dataType : "json",
    success: function(data){
        $.mobile.loading( 'hide');
    },
});
}

function s1(){
$.ajax({
    url: "service.php?action=s1",
    dataType : "json",
    success: function(data){
        $.mobile.loading( 'hide');
    },
});
}

function s13_10(){
$.ajax({
    url: "service.php?action=s13_10",
    dataType : "json",
    success: function(data){
        $.mobile.loading( 'hide');
    },
});
}

function s16_10(){
$.ajax({
    url: "service.php?action=s16_10",
    dataType : "json",
    success: function(data){
        $.mobile.loading( 'hide');
    },
});
}

function s2(){
$.ajax({
    url: "service.php?action=s2",
    dataType : "json",
    success: function(data){
        $.mobile.loading( 'hide');
    },
});
}

function s25_10(){
$.ajax({
    url: "service.php?action=s25_10",
    dataType : "json",
    success: function(data){
        $.mobile.loading( 'hide');
    },
});
}

function s3(){
$.ajax({
    url: "service.php?action=s3",
    dataType : "json",

```

```

        success: function(data){
            $.mobile.loading( 'hide');
        },
    });
}

function s4(){
    $.ajax({
        url: "service.php?action=s4",
        dataType : "json",
        success: function(data){
            $.mobile.loading( 'hide');
        },
    });
}

function s5(){
    $.ajax({
        url: "service.php?action=s5",
        dataType : "json",
        success: function(data){
            $.mobile.loading( 'hide');
        },
    });
}

function s6(){
    $.ajax({
        url: "service.php?action=s6",
        dataType : "json",
        success: function(data){
            $.mobile.loading( 'hide');
        },
    });
}

function s8(){
    $.ajax({
        url: "service.php?action=s8",
        dataType : "json",
        success: function(data){
            $.mobile.loading( 'hide');
        },
    });
}

function s10(){
    $.ajax({
        url: "service.php?action=s10",
        dataType : "json",
        success: function(data){
            $.mobile.loading( 'hide');
        },
    });
}

function s13(){
    $.ajax({
        url: "service.php?action=s13",
        dataType : "json",
        success: function(data){
            $.mobile.loading( 'hide');
        },
    });
}

function s15(){
    $.ajax({
        url: "service.php?action=s15",
        dataType : "json",
        success: function(data){

```

```

        $.mobile.loading( 'hide');
    },
    });
}

function s20(){
$.ajax({
    url: "service.php?action=s20",
    dataType : "json",
    success: function(data){
        $.mobile.loading( 'hide');
    },
    });
}

function s25(){
$.ajax({
    url: "service.php?action=s25",
    dataType : "json",
    success: function(data){
        $.mobile.loading( 'hide');
    },
    });
}

function s30(){
$.ajax({
    url: "service.php?action=s30",
    dataType : "json",
    success: function(data){
        $.mobile.loading( 'hide');
    },
    });
}

function sx250(){
$.ajax({
    url: "service.php?action=sx250",
    dataType : "json",
    success: function(data){
        $.mobile.loading( 'hide');
    },
    });
}

function bulb(){
$.ajax({
    url: "service.php?action=bulb",
    dataType : "json",
    success: function(data){
        $.mobile.loading( 'hide');
    },
    });
}

function getautoiso(){
$.ajax({
    url: "service.php?action=getautoiso",
    dataType : "json",
    success: function(data){
        $("#autoiso").html(data.camera);
    },
    });
}

function autoiso_on(){
$.ajax({
    url: "service.php?action=autoiso_on",
    dataType : "json",
    success: function(data){
        $.mobile.loading( 'hide');
    }
});
}

```



```

    },
  });
}

function autoiso_off(){
$.ajax({
  url: "service.php?action=autoiso_off",
  dataType : "json",
  success: function(data){
    $.mobile.loading( 'hide');
  },
});
}

function getiso(){
$.ajax({
  url: "service.php?action=getiso",
  dataType : "json",
  success: function(data){
    $("#iso").html(data.camera);
  },
});
}

function iso_100(){
$.ajax({
  url: "service.php?action=iso_100",
  dataType : "json",
  success: function(data){
    $.mobile.loading( 'hide');
  },
});
}

function iso_125(){
$.ajax({
  url: "service.php?action=iso_125",
  dataType : "json",
  success: function(data){
    $.mobile.loading( 'hide');
  },
});
}

function iso_160(){
$.ajax({
  url: "service.php?action=iso_160",
  dataType : "json",
  success: function(data){
    $.mobile.loading( 'hide');
  },
});
}

function iso_200(){
$.ajax({
  url: "service.php?action=iso_200",
  dataType : "json",
  success: function(data){
    $.mobile.loading( 'hide');
  },
});
}

function iso_250(){
$.ajax({
  url: "service.php?action=iso_250",
  dataType : "json",
  success: function(data){
    $.mobile.loading( 'hide');
  },
});
}

```

```

    });
}

function iso_320(){
$.ajax({
url: "service.php?action=iso_320",
dataType : "json",
success: function(data){
$.mobile.loading( 'hide');
},
});
}

function iso_400(){
$.ajax({
url: "service.php?action=iso_400",
dataType : "json",
success: function(data){
$.mobile.loading( 'hide');
},
});
}

function iso_500(){
$.ajax({
url: "service.php?action=iso_500",
dataType : "json",
success: function(data){
$.mobile.loading( 'hide');
},
});
}

function iso_640(){
$.ajax({
url: "service.php?action=iso_640",
dataType : "json",
success: function(data){
$.mobile.loading( 'hide');
},
});
}

function iso_800(){
$.ajax({
url: "service.php?action=iso_800",
dataType : "json",
success: function(data){
$.mobile.loading( 'hide');
},
});
}

function iso_1000(){
$.ajax({
url: "service.php?action=iso_1000",
dataType : "json",
success: function(data){
$.mobile.loading( 'hide');
},
});
}

function iso_1250(){
$.ajax({
url: "service.php?action=iso_1250",
dataType : "json",
success: function(data){
$.mobile.loading( 'hide');
},
});
}

```

```

}

function iso_1600(){
$.ajax({
url: "service.php?action=iso_1600",
dataType : "json",
success: function(data){
$.mobile.loading( 'hide');
},
});
}

function iso_2000(){
$.ajax({
url: "service.php?action=iso_2000",
dataType : "json",
success: function(data){
$.mobile.loading( 'hide');
},
});
}

function iso_2500(){
$.ajax({
url: "service.php?action=iso_2500",
dataType : "json",
success: function(data){
$.mobile.loading( 'hide');
},
});
}

function iso_3200(){
$.ajax({
url: "service.php?action=iso_3200",
dataType : "json",
success: function(data){
$.mobile.loading( 'hide');
},
});
}

function iso_4000(){
$.ajax({
url: "service.php?action=iso_4000",
dataType : "json",
success: function(data){
$.mobile.loading( 'hide');
},
});
}

function iso_5000(){
$.ajax({
url: "service.php?action=iso_5000",
dataType : "json",
success: function(data){
$.mobile.loading( 'hide');
},
});
}

function iso_6400(){
$.ajax({
url: "service.php?action=iso_6400",
dataType : "json",
success: function(data){
$.mobile.loading( 'hide');
},
});
}

```

```

function iso_8320(){
$.ajax({
url: "service.php?action=iso_8320",
dataType : "json",
success: function(data){
$.mobile.loading( 'hide');
},
});
}

function iso_10240(){
$.ajax({
url: "service.php?action=iso_10240",
dataType : "json",
success: function(data){
$.mobile.loading( 'hide');
},
});
}

function iso_12800(){
$.ajax({
url: "service.php?action=iso_12800",
dataType : "json",
success: function(data){
$.mobile.loading( 'hide');
},
});
}

function iso_25600(){
$.ajax({
url: "service.php?action=iso_25600",
dataType : "json",
success: function(data){
$.mobile.loading( 'hide');
},
});
}

function getexpcomp(){
$.ajax({
url: "service.php?action=getexpcomp",
dataType : "json",
success: function(data){
$("#expcomp").html(data.camera);
},
});
}

function comp_50(){
$.ajax({
url: "service.php?action=comp_50",
dataType : "json",
success: function(data){
$.mobile.loading( 'hide');
},
});
}

function comp_47(){
$.ajax({
url: "service.php?action=comp_47",
dataType : "json",
success: function(data){
$.mobile.loading( 'hide');
},
});
}

```

```

function comp_43(){
$.ajax({
url: "service.php?action=comp_43",
dataType : "json",
success: function(data){
$.mobile.loading( 'hide');
},
});
}

function comp_40(){
$.ajax({
url: "service.php?action=comp_40",
dataType : "json",
success: function(data){
$.mobile.loading( 'hide');
},
});
}

function comp_37(){
$.ajax({
url: "service.php?action=comp_37",
dataType : "json",
success: function(data){
$.mobile.loading( 'hide');
},
});
}

function comp_33(){
$.ajax({
url: "service.php?action=comp_33",
dataType : "json",
success: function(data){
$.mobile.loading( 'hide');
},
});
}

function comp_30(){
$.ajax({
url: "service.php?action=comp_30",
dataType : "json",
success: function(data){
$.mobile.loading( 'hide');
},
});
}

function comp_27(){
$.ajax({
url: "service.php?action=comp_27",
dataType : "json",
success: function(data){
$.mobile.loading( 'hide');
},
});
}

function comp_23(){
$.ajax({
url: "service.php?action=comp_23",
dataType : "json",
success: function(data){
$.mobile.loading( 'hide');
},
});
}

function comp_20(){

```

```

$.ajax({
  url: "service.php?action=comp_20",
  dataType : "json",
  success: function(data){
    $.mobile.loading( 'hide' );
  },
});
}

function comp_17(){
$.ajax({
  url: "service.php?action=comp_17",
  dataType : "json",
  success: function(data){
    $.mobile.loading( 'hide' );
  },
});
}

function comp_13(){
$.ajax({
  url: "service.php?action=comp_13",
  dataType : "json",
  success: function(data){
    $.mobile.loading( 'hide' );
  },
});
}

function comp_10(){
$.ajax({
  url: "service.php?action=comp_10",
  dataType : "json",
  success: function(data){
    $.mobile.loading( 'hide' );
  },
});
}

function comp_07(){
$.ajax({
  url: "service.php?action=comp_07",
  dataType : "json",
  success: function(data){
    $.mobile.loading( 'hide' );
  },
});
}

function comp_03(){
$.ajax({
  url: "service.php?action=comp_03",
  dataType : "json",
  success: function(data){
    $.mobile.loading( 'hide' );
  },
});
}

function comp_00(){
$.ajax({
  url: "service.php?action=comp_00",
  dataType : "json",
  success: function(data){
    $.mobile.loading( 'hide' );
  },
});
}

function comp_m03(){
$.ajax({

```

```

        url: "service.php?action=comp_m03",
        dataType : "json",
        success: function(data){
            $.mobile.loading( 'hide' );
        },
    });
}

function comp_m07 () {
    $.ajax({
        url: "service.php?action=comp_m07",
        dataType : "json",
        success: function(data){
            $.mobile.loading( 'hide' );
        },
    });
}

function comp_m10 () {
    $.ajax({
        url: "service.php?action=comp_m10",
        dataType : "json",
        success: function(data){
            $.mobile.loading( 'hide' );
        },
    });
}

function comp_m13 () {
    $.ajax({
        url: "service.php?action=comp_m13",
        dataType : "json",
        success: function(data){
            $.mobile.loading( 'hide' );
        },
    });
}

function comp_m17 () {
    $.ajax({
        url: "service.php?action=comp_m17",
        dataType : "json",
        success: function(data){
            $.mobile.loading( 'hide' );
        },
    });
}

function comp_m20 () {
    $.ajax({
        url: "service.php?action=comp_m20",
        dataType : "json",
        success: function(data){
            $.mobile.loading( 'hide' );
        },
    });
}

function comp_m23 () {
    $.ajax({
        url: "service.php?action=comp_m23",
        dataType : "json",
        success: function(data){
            $.mobile.loading( 'hide' );
        },
    });
}

function comp_m27 () {
    $.ajax({
        url: "service.php?action=comp_m27",

```

```

        dataType : "json",
        success: function(data){
            $.mobile.loading( 'hide');
        },
    });
}

function comp_m30(){
$.ajax({
    url: "service.php?action=comp_m30",
    dataType : "json",
    success: function(data){
        $.mobile.loading( 'hide');
    },
});
}

function comp_m33(){
$.ajax({
    url: "service.php?action=comp_m33",
    dataType : "json",
    success: function(data){
        $.mobile.loading( 'hide');
    },
});
}

function comp_m37(){
$.ajax({
    url: "service.php?action=comp_m37",
    dataType : "json",
    success: function(data){
        $.mobile.loading( 'hide');
    },
});
}

function comp_m40(){
$.ajax({
    url: "service.php?action=comp_m40",
    dataType : "json",
    success: function(data){
        $.mobile.loading( 'hide');
    },
});
}

function comp_m43(){
$.ajax({
    url: "service.php?action=comp_m43",
    dataType : "json",
    success: function(data){
        $.mobile.loading( 'hide');
    },
});
}

function comp_m47(){
$.ajax({
    url: "service.php?action=comp_m47",
    dataType : "json",
    success: function(data){
        $.mobile.loading( 'hide');
    },
});
}

function comp_m50(){
$.ajax({
    url: "service.php?action=comp_m50",
    dataType : "json",

```



```

        success: function(data){
            $.mobile.loading( 'hide');
        },
    });
}

function getflex(){
    $.ajax({
        url: "service.php?action=getflex",
        dataType : "json",
        success: function(data){
            $("#flex").html(data.camera);
        },
    });
}

function shift_30(){
    $.ajax({
        url: "service.php?action=shift_30",
        dataType : "json",
        success: function(data){
            $.mobile.loading( 'hide');
        },
    });
}

function shift_28(){
    $.ajax({
        url: "service.php?action=shift_28",
        dataType : "json",
        success: function(data){
            $.mobile.loading( 'hide');
        },
    });
}

function shift_26(){
    $.ajax({
        url: "service.php?action=shift_26",
        dataType : "json",
        success: function(data){
            $.mobile.loading( 'hide');
        },
    });
}

function shift_24(){
    $.ajax({
        url: "service.php?action=shift_24",
        dataType : "json",
        success: function(data){
            $.mobile.loading( 'hide');
        },
    });
}

function shift_22(){
    $.ajax({
        url: "service.php?action=shift_22",
        dataType : "json",
        success: function(data){
            $.mobile.loading( 'hide');
        },
    });
}

function shift_20(){
    $.ajax({
        url: "service.php?action=shift_20",
        dataType : "json",
        success: function(data){

```

```

        $.mobile.loading( 'hide');
    },
    });
}

function shift_18(){
$.ajax({
    url: "service.php?action=shift_18",
    dataType : "json",
    success: function(data){
        $.mobile.loading( 'hide');
    },
    });
}

function shift_16(){
$.ajax({
    url: "service.php?action=shift_16",
    dataType : "json",
    success: function(data){
        $.mobile.loading( 'hide');
    },
    });
}

function shift_14(){
$.ajax({
    url: "service.php?action=shift_14",
    dataType : "json",
    success: function(data){
        $.mobile.loading( 'hide');
    },
    });
}

function shift_12(){
$.ajax({
    url: "service.php?action=shift_12",
    dataType : "json",
    success: function(data){
        $.mobile.loading( 'hide');
    },
    });
}

function shift_10(){
$.ajax({
    url: "service.php?action=shift_10",
    dataType : "json",
    success: function(data){
        $.mobile.loading( 'hide');
    },
    });
}

function shift_8(){
$.ajax({
    url: "service.php?action=shift_8",
    dataType : "json",
    success: function(data){
        $.mobile.loading( 'hide');
    },
    });
}

function shift_6(){
$.ajax({
    url: "service.php?action=shift_6",
    dataType : "json",
    success: function(data){
        $.mobile.loading( 'hide');
    }

```

```

    },
  });
}

function shift_4(){
$.ajax({
  url: "service.php?action=shift_4",
  dataType : "json",
  success: function(data){
    $.mobile.loading( 'hide' );
  },
});
}

function shift_2(){
$.ajax({
  url: "service.php?action=shift_2",
  dataType : "json",
  success: function(data){
    $.mobile.loading( 'hide' );
  },
});
}

function shift_0(){
$.ajax({
  url: "service.php?action=shift_0",
  dataType : "json",
  success: function(data){
    $.mobile.loading( 'hide' );
  },
});
}

function shift_m30(){
$.ajax({
  url: "service.php?action=shift_m30",
  dataType : "json",
  success: function(data){
    $.mobile.loading( 'hide' );
  },
});
}

function shift_m28(){
$.ajax({
  url: "service.php?action=shift_m28",
  dataType : "json",
  success: function(data){
    $.mobile.loading( 'hide' );
  },
});
}

function shift_m26(){
$.ajax({
  url: "service.php?action=shift_m26",
  dataType : "json",
  success: function(data){
    $.mobile.loading( 'hide' );
  },
});
}

function shift_m24(){
$.ajax({
  url: "service.php?action=shift_m24",
  dataType : "json",
  success: function(data){
    $.mobile.loading( 'hide' );
  },
});
}

```

```

    },
  });
}

function shift_m22(){
$.ajax({
  url: "service.php?action=shift_m22",
  dataType : "json",
  success: function(data){
    $.mobile.loading( 'hide');
  },
});
}

function shift_m20(){
$.ajax({
  url: "service.php?action=shift_m20",
  dataType : "json",
  success: function(data){
    $.mobile.loading( 'hide');
  },
});
}

function shift_m18(){
$.ajax({
  url: "service.php?action=shift_m18",
  dataType : "json",
  success: function(data){
    $.mobile.loading( 'hide');
  },
});
}

function shift_m16(){
$.ajax({
  url: "service.php?action=shift_m16",
  dataType : "json",
  success: function(data){
    $.mobile.loading( 'hide');
  },
});
}

function shift_m14(){
$.ajax({
  url: "service.php?action=shift_m14",
  dataType : "json",
  success: function(data){
    $.mobile.loading( 'hide');
  },
});
}

function shift_m12(){
$.ajax({
  url: "service.php?action=shift_m12",
  dataType : "json",
  success: function(data){
    $.mobile.loading( 'hide');
  },
});
}

function shift_m10(){
$.ajax({
  url: "service.php?action=shift_m10",
  dataType : "json",
  success: function(data){
    $.mobile.loading( 'hide');
  },
});
}

```

```

    });
}

function shift_m8(){
$.ajax({
url: "service.php?action=shift_m8",
dataType : "json",
success: function(data){
$.mobile.loading( 'hide');
},
});
}

function shift_m6(){
$.ajax({
url: "service.php?action=shift_m6",
dataType : "json",
success: function(data){
$.mobile.loading( 'hide');
},
});
}

function shift_m4(){
$.ajax({
url: "service.php?action=shift_m4",
dataType : "json",
success: function(data){
$.mobile.loading( 'hide');
},
});
}

function shift_m2(){
$.ajax({
url: "service.php?action=shift_m2",
dataType : "json",
success: function(data){
$.mobile.loading( 'hide');
},
});
}

function getquality(){
$.ajax({
url: "service.php?action=getquality",
dataType : "json",
success: function(data){
$("#quality").html(data.camera);
},
});
}

function basic(){
$.ajax({
url: "service.php?action=basic",
dataType : "json",
success: function(data){
$.mobile.loading( 'hide');
},
});
}

function normal(){
$.ajax({
url: "service.php?action=normal",
dataType : "json",
success: function(data){
$.mobile.loading( 'hide');
},
});
}

```

```

}

function fine(){
$.ajax({
url: "service.php?action=fine",
dataType: "json",
success: function(data){
$.mobile.loading( 'hide');
},
});
}

function tiff(){
$.ajax({
url: "service.php?action=tiff",
dataType: "json",
success: function(data){
$.mobile.loading( 'hide');
},
});
}

function raw(){
$.ajax({
url: "service.php?action=raw",
dataType: "json",
success: function(data){
$.mobile.loading( 'hide');
},
});
}

function rbasic(){
$.ajax({
url: "service.php?action=rbasic",
dataType: "json",
success: function(data){
$.mobile.loading( 'hide');
},
});
}

function rnormal(){
$.ajax({
url: "service.php?action=rnormal",
dataType: "json",
success: function(data){
$.mobile.loading( 'hide');
},
});
}

function rfine(){
$.ajax({
url: "service.php?action=rfine",
dataType: "json",
success: function(data){
$.mobile.loading( 'hide');
},
});
}

function getsize(){
$.ajax({
url: "service.php?action=getsize",
dataType: "json",
success: function(data){
$("#size").html( data.camera);
},
});
}

```

```

function large(){
    $.ajax({
        url: "service.php?action=large",
        dataType : "json",
        success: function(data){
            $.mobile.loading( 'hide');
        },
    });
}

function med(){
    $.ajax({
        url: "service.php?action=med",
        dataType : "json",
        success: function(data){
            $.mobile.loading( 'hide');
        },
    });
}

function sml(){
    $.ajax({
        url: "service.php?action=small",
        dataType : "json",
        success: function(data){
            $.mobile.loading( 'hide');
        },
    });
}

function getrsz(){
    $.ajax({
        url: "service.php?action=getrsz",
        dataType : "json",
        success: function(data){
            $("#rsz").html(data.camera);
        },
    });
}

function lossless(){
    $.ajax({
        url: "service.php?action=lossless",
        dataType : "json",
        success: function(data){
            $.mobile.loading( 'hide');
        },
    });
}

function lossy(){
    $.ajax({
        url: "service.php?action=lossy",
        dataType : "json",
        success: function(data){
            $.mobile.loading( 'hide');
        },
    });
}

function comp_off(){
    $.ajax({
        url: "service.php?action=comp_off",
        dataType : "json",
        success: function(data){
            $.mobile.loading( 'hide');
        },
    });
}

```

```

function getwbalance(){
    $.ajax({
        url: "service.php?action=getwbalance",
        dataType : "json",
        success: function(data){
            $("#wbalance").html(data.camera);
        },
    });
}

function wauto(){
    $.ajax({
        url: "service.php?action=wauto",
        dataType : "json",
        success: function(data){
            $.mobile.loading( 'hide' );
        },
    });
}

function dayl(){
    $.ajax({
        url: "service.php?action=dayl",
        dataType : "json",
        success: function(data){
            $.mobile.loading( 'hide' );
        },
    });
}

function fluo(){
    $.ajax({
        url: "service.php?action=fluo",
        dataType : "json",
        success: function(data){
            $.mobile.loading( 'hide' );
        },
    });
}

function tung(){
    $.ajax({
        url: "service.php?action=tung",
        dataType : "json",
        success: function(data){
            $.mobile.loading( 'hide' );
        },
    });
}

function wflash(){
    $.ajax({
        url: "service.php?action=wflash",
        dataType : "json",
        success: function(data){
            $.mobile.loading( 'hide' );
        },
    });
}

function cloudy(){
    $.ajax({
        url: "service.php?action=cloudy",
        dataType : "json",
        success: function(data){
            $.mobile.loading( 'hide' );
        },
    });
}

function shade(){

```



```

$.ajax({
  url: "service.php?action=shade",
  dataType : "json",
  success: function(data){
    $.mobile.loading( 'hide');
  },
});
}

function color_temp(){
$.ajax({
  url: "service.php?action=color_temp",
  dataType : "json",
  success: function(data){
    $.mobile.loading( 'hide');
  },
});
}

function wpreset(){
$.ajax({
  url: "service.php?action=wpreset",
  dataType : "json",
  success: function(data){
    $.mobile.loading( 'hide');
  },
});
}

function getcolorspace(){
$.ajax({
  url: "service.php?action=getcolorspace",
  dataType : "json",
  success: function(data){
    $("#colorspace").html(data.camera);
  },
});
}

function srgb(){
$.ajax({
  url: "service.php?action=srgb",
  dataType : "json",
  success: function(data){
    $.mobile.loading( 'hide');
  },
});
}

function adobe(){
$.ajax({
  url: "service.php?action=adobe",
  dataType : "json",
  success: function(data){
    $.mobile.loading( 'hide');
  },
});
}

function gethisonr(){
$.ajax({
  url: "service.php?action=gethisonr",
  dataType : "json",
  success: function(data){
    $("#isonr").html(data.camera);
  },
});
}

function hnroff(){
$.ajax({

```

```

        url: "service.php?action=hnrroff",
        dataType : "json",
        success: function(data){
            $.mobile.loading( 'hide');
        },
    });
}

function hnrllow(){
    $.ajax({
        url: "service.php?action=hnrllow",
        dataType : "json",
        success: function(data){
            $.mobile.loading( 'hide');
        },
    });
}

function hnrnorm(){
    $.ajax({
        url: "service.php?action=hnrnorm",
        dataType : "json",
        success: function(data){
            $.mobile.loading( 'hide');
        },
    });
}

function hnrhigh(){
    $.ajax({
        url: "service.php?action=hnrhigh",
        dataType : "json",
        success: function(data){
            $.mobile.loading( 'hide');
        },
    });
}

function getlnr(){
    $.ajax({
        url: "service.php?action=getlnr",
        dataType : "json",
        success: function(data){
            $("#lnr").html(data.camera);
        },
    });
}

function lnron(){
    $.ajax({
        url: "service.php?action=lnron",
        dataType : "json",
        success: function(data){
            $.mobile.loading( 'hide');
        },
    });
}

function lnroff(){
    $.ajax({
        url: "service.php?action=lnroff",
        dataType : "json",
        success: function(data){
            $.mobile.loading( 'hide');
        },
    });
}

function getflashmode(){
    $.ajax({
        url: "service.php?action=getflashmode",

```

```

        dataType : "json",
        success: function(data) {
            $("#flashmode").html (data.camera);
        },
    });
}

function flredauto(){
$.ajax({
    url: "service.php?action=flredauto",
    dataType : "json",
    success: function(data) {
        $.mobile.loading( 'hide');
    },
});
}

function flauto(){
$.ajax({
    url: "service.php?action=flauto",
    dataType : "json",
    success: function(data) {
        $.mobile.loading( 'hide');
    },
});
}

function flautoslow(){
$.ajax({
    url: "service.php?action=flautoslow",
    dataType : "json",
    success: function(data) {
        $.mobile.loading( 'hide');
    },
});
}

function flrearcertainslow(){
$.ajax({
    url: "service.php?action=flrearcertainslow",
    dataType : "json",
    success: function(data) {
        $.mobile.loading( 'hide');
    },
});
}

function flredslow(){
$.ajax({
    url: "service.php?action=flredslow",
    dataType : "json",
    success: function(data) {
        $.mobile.loading( 'hide');
    },
});
}

function getflashsyncspeed(){
$.ajax({
    url: "service.php?action=getflashsyncspeed",
    dataType : "json",
    success: function(data) {
        $("#flashsyncspeed").html (data.camera);
    },
});
}

function autofp(){
$.ajax({
    url: "service.php?action=autofp",
    dataType : "json",

```

```

        success: function(data){
            $.mobile.loading( 'hide');
        },
    });
}

function f1_250(){
    $.ajax({
        url: "service.php?action=f1_250",
        dataType : "json",
        success: function(data){
            $.mobile.loading( 'hide');
        },
    });
}

function f1_200(){
    $.ajax({
        url: "service.php?action=f1_200",
        dataType : "json",
        success: function(data){
            $.mobile.loading( 'hide');
        },
    });
}

function f1_160(){
    $.ajax({
        url: "service.php?action=f1_160",
        dataType : "json",
        success: function(data){
            $.mobile.loading( 'hide');
        },
    });
}

function f1_125(){
    $.ajax({
        url: "service.php?action=f1_125",
        dataType : "json",
        success: function(data){
            $.mobile.loading( 'hide');
        },
    });
}

function f1_100(){
    $.ajax({
        url: "service.php?action=f1_100",
        dataType : "json",
        success: function(data){
            $.mobile.loading( 'hide');
        },
    });
}

function f1_80(){
    $.ajax({
        url: "service.php?action=f1_80",
        dataType : "json",
        success: function(data){
            $.mobile.loading( 'hide');
        },
    });
}

function f1_60(){
    $.ajax({
        url: "service.php?action=f1_60",
        dataType : "json",
        success: function(data){

```

```

        $.mobile.loading( 'hide');
    },
    });
}

function getflashshutterspeed(){
$.ajax({
    url: "service.php?action=getflashshutterspeed",
    dataType : "json",
    success: function(data){
        $("#flashshutterspeed").html(data.camera);
    },
});
}

function fs1_60(){
$.ajax({
    url: "service.php?action=fs1_60",
    dataType : "json",
    success: function(data){
        $.mobile.loading( 'hide');
    },
});
}

function fs1_30(){
$.ajax({
    url: "service.php?action=fs1_30",
    dataType : "json",
    success: function(data){
        $.mobile.loading( 'hide');
    },
});
}

function fs1_15(){
$.ajax({
    url: "service.php?action=fs1_15",
    dataType : "json",
    success: function(data){
        $.mobile.loading( 'hide');
    },
});
}

function fs1_8(){
$.ajax({
    url: "service.php?action=fs1_8",
    dataType : "json",
    success: function(data){
        $.mobile.loading( 'hide');
    },
});
}

function fs1_4(){
$.ajax({
    url: "service.php?action=fs1_4",
    dataType : "json",
    success: function(data){
        $.mobile.loading( 'hide');
    },
});
}

function fs1_2(){
$.ajax({
    url: "service.php?action=fs1_2",
    dataType : "json",
    success: function(data){
        $.mobile.loading( 'hide');
    },
});
}

```

```

    },
  });
}

function fs1(){
$.ajax({
  url: "service.php?action=fs1",
  dataType : "json",
  success: function(data){
    $.mobile.loading( 'hide');
  },
});
}

function fs2(){
$.ajax({
  url: "service.php?action=fs2",
  dataType : "json",
  success: function(data){
    $.mobile.loading( 'hide');
  },
});
}

function fs4(){
$.ajax({
  url: "service.php?action=fs4",
  dataType : "json",
  success: function(data){
    $.mobile.loading( 'hide');
  },
});
}

function fs8(){
$.ajax({
  url: "service.php?action=fs8",
  dataType : "json",
  success: function(data){
    $.mobile.loading( 'hide');
  },
});
}

function fs15(){
$.ajax({
  url: "service.php?action=fs15",
  dataType : "json",
  success: function(data){
    $.mobile.loading( 'hide');
  },
});
}

function fs30(){
$.ajax({
  url: "service.php?action=fs30",
  dataType : "json",
  success: function(data){
    $.mobile.loading( 'hide');
  },
});
}

function getCamera(){
$.ajax({
  url: "service.php?action=getCamera",
  dataType : "json",
  success: function(data){
    $("#cameraName").html(data.camera);
  },
});
}

```

1 10;

Βιβλιογραφία

Βιβλία:

Εγχειρίδιο χρήσης ψηφιακής φωτογραφικής μηχανής Nikon D3

Michael R. Peres (2007). *The Focal Encyclopedia of Photography* (4th ed.). Focal Press. ISBN 0-240-80740-5.

Prakel, David (2009). *The Visual Dictionary of Photography*. AVA Publishing. p. 91. ISBN 978-2-940411-04-7

Διαδίκτυο:

<http://apphotnum.free.fr/N2BE2.html>

<http://www.eeei.gr/odhgos/htmlfaq.htm>

https://elinux.org/RPi_SD_cards

<http://www.wlearn.gr/index.php/home-css-83>

<https://www.macworld.com/article/1039455/digitalfilm.html>

<https://www.w3schools.com/html>

"Foundation Strategy 2016 - 2018" (PDF). *Raspberry Pi*. Raspberry Pi Foundation.

Upton, Eben (29 February 2016). "Raspberry Pi 3 on sale now at \$35 – Raspberry Pi"

<http://gphoto.sourceforge.net/>

<https://freestuff.gr/forums/viewtopic.php?t=21175>

<https://www.blaize.net/2014/04/gphoto-webui-a-php-web-ui-for-gphoto2/>

<https://www.raspberrypi.org/documentation/installation/sd-cards.md>

<http://php.net/manual/en/function.exec.php>

<http://php.net/manual/en/function.session-start.php>

<http://php.net/manual/en/control-structures.break.php>

<http://php.net/manual/en/function.echo.php>

<http://php.net/manual/en/function.json-encode.php>

<http://php.net/manual/en/sdo.sample.getset.php>

https://www.w3schools.com/js/js_json_php.asp

https://www.w3schools.com/php/func_http_header.asp

<http://php.net/manual/en/internals2.opcodes.catch.php>

<http://php.net/manual/en/control-structures.switch.php>

<http://php.net/manual/en/language.exceptions.php>

<http://php.net/manual/de/control-structures.switch.php>

<http://php.net/manual/en/function.substr.php>

https://www.w3schools.com/php/func_string_explode.asp

<https://www.w3schools.in/php-script/php-login-without-using-database/>