

# **Εγχειρίδιο Ανάλυσης και Σχεδιασμού**

**Η ανάπτυξη της εφαρμογής Έξυπνο Σπίτι από την  
πλευρά του προγραμματιστή**

## Περιεχόμενα

Πρόλογος .....	3
Σαλόνι.....	4
Υπνοδωμάτιο .....	4
Κουζίνα.....	4
Κινητό.....	5
Control Panel.....	5
Έξυπνο Ψυγείο .....	5
Έξυπνη Ντουλάπα.....	6
Επίλογος.....	7

## Πρόλογος

Η εφαρμογή Έξυπνο Σπίτι πρόκειται για μια παραθυρική εφαρμογή με γραφικά φτιαγμένη εξολοκλήρου στη γλώσσα προγραμματισμού C Sharp. Το εργαλείο που χρησιμοποιήθηκε για την ανάπτυξη της είναι το Blend for Visual Studio 2015. Στο σύνολο της η εφαρμογή αποτελείται από αρχεία τύπου xaml όπου είναι περιγραφικού κώδικα και περιέχουν τις διάφορες μορφοποιήσεις πχ φόντο, τα control του παραθύρου αλλά και χαρακτηριστικά όπως τις διαστάσεις των παραθύρων τα οποία συνοδεύονται με το αντίστοιχο cs αρχείο το οποίο περιέχει τον κώδικα σε C Sharp για το κάθε xaml. Η εφαρμογή περιέχει τα εξής αρχεία xaml: bedroom, living\_room, kitchen, control\_panel, smartphone, smart\_fridge, smart\_wardrobe αλλά και τα ομώνυμα συνοδευτικά αρχεία cs. Κάθε αρχείο περιγράφει ένα διαφορετικό κομμάτι του Έξυπνου Σπιτιού το οποίο κομμάτι είναι ενδεικτικό του ονόματος που φέρει (πχ living\_room για το χώρο του σαλονιού, bedroom για το υπνοδωμάτιο κλπ). Μέσω των αρχείων τύπου cs δημιουργούνται οι συνδέσεις και οι αλληλεπιδράσεις μεταξύ των κομματιών αυτών όπως για παράδειγμα κώδικας που χρειάζεται ώστε ο χρήστης να μεταβεί από σαλόνι στην κουζίνα. Η C Sharp είναι μια αντικειμεστραφής γλώσσα έτσι λοιπόν στην περίπτωση μας κάθε ένα από αυτά τα 7 κομμάτια αναπαριστάται με μια διαφορετική κλάση που έχει τα controls τις μεθόδους και τα χαρακτηριστικά της που θα αναλυθούν παρακάτω.

## Σαλόνι

Το σαλόνι αποτελείται από τα αρχεία `living_room.xaml` και `living_room.xaml.cs`. Όπως και σε κάθε άλλο παράθυρο βασικό στοιχείο είναι το `control image` με το ανάλογο `source` το οποίο υπάρχει ως `background` επίσης το παράθυρο είναι ρυθμισμένο μέσα στον `xaml` κώδικα να μην είναι `resizable`. Ακόμη το παράθυρο εμφανίζεται στο κέντρο της οθόνης αυτό είναι ένα χαρακτηριστικό που το μοιράζονται μόνο τα 3 δωμάτια όπως και ότι στο πάνω δεξιά υπάρχει πλήκτρο βοήθειας το οποίο είναι μια εικόνα και με το πάτημα του κουμπιού δημιουργείται το ανάλογο `event` που τυπώνει σε νέο παράθυρο το βοηθητικό κείμενο. Στο σαλόνι βρίσκονται τα 2 κουμπιά για τα άλλα 2 δωμάτια τα οποία δημιουργούν ένα νέο `instance` των αντίστοιχων κλάσεων που ο χρήστης θέλει να μεταβεί, το ίδιο ισχύει και για το κινητό. Πέρα από αυτά στο σαλόνι βρίσκεται ένα αόρατο πλήκτρο πάνω από το πληκτρολόγιο της εικόνας το οποίο ανοίγει το `control panel` της εφαρμογής. Ακόμη μέσα στο σαλόνι βρίσκονται 2 κρυφά `textblocks` και 1 κρυφή εικόνα στην οθόνη της τηλεόρασης τα οποία υπάρχουν εκεί για να δείξουν αλλαγές όπως όταν για παράδειγμα ο χρήστης ενεργοποιήσει την τηλεόραση από το `control panel` και ως συνέπεια η κατάλληλη εικόνα θα γίνει ορατή στο σαλόνι.

## Υπνοδωμάτιο

Το υπνοδωμάτιο αποτελείται από τα αρχεία `bedroom.xaml` και `bedroom.xaml.cs`. Για το υπνοδωμάτιο ισχύουν τα βασικά χαρακτηριστικά που είναι κοινά μεταξύ των παραθύρων όπως προαναφέρθηκαν. Ακόμη όμως φέρει ένα κρυφό `textblock` για τις ενδείξεις θερμοκρασίας που θα εμφανιστεί με αλλαγές από το πάνελ μια κρυφή εικόνα πιο φωτεινού φόντου και ένα μεγάλο κοκκινωπό πλήκτρο πάνω από την ντουλάπα το οποίο με το `click event` θα δημιουργήσει ένα νέο αντικείμενο για το `smart_wardrobe` και θα το εμφανίσει.

## Κουζίνα

Η κουζίνα αποτελείται από τα αρχεία `kitchen.xaml` και `kitchen.xaml.cs`. Και σε αυτή την περίπτωση ισχύουν όσα προαναφέρθηκαν όπως επίσης και στην κουζίνα συναντάται το `hidden textblock` για τη θερμοκρασία δωματίου που εμφανίζει και παίρνει χαρακτηριστικά ανάλογα με την ενεργοποίηση από το `control panel`. Στην κουζίνα επίσης βρίσκεται το έξυπνο ψυγείο στο οποίο ο χρήστης μπορεί να αποκτήσει πρόσβαση πατώντας στο μπλε πλαίσιο που καλύπτει μέρος της μεγάλης του οθόνης. Με αυτή την ενέργεια(event) ένα νέο αντικείμενο θα δημιουργηθεί για το έξυπνο ψυγείο και θα εμφανιστεί με μορφή παραθύρου. Γενικά η μετάβαση σε νέα παράθυρα κάθε φορά στο σύνολο της εφαρμογής γίνεται με τη δημιουργία ενός αντικειμένου εκεί που θέλουμε να μεταβούμε και μετά εμφάνισής του με τη συνάρτηση `Show`. Όταν όμως πρόκειται για μετάβαση μεταξύ δωματίων κάνουμε και `hide` το αντικείμενο στο οποίο βρισκόμασταν πριν μεταφερθούμε στο επόμενο.

## Κινητό

Το κινητό αποτελείται από τα αρχεία `smartphone.xaml` και `smartphone.xaml.cs`. Για να αποκτήσει ο χρήστης πρόσβαση σε αυτό πρέπει να πατήσει στο `control image` που υπάρχει σε κάθε δωμάτιο στο κάτω δεξιά μέρος και έχει μια μικρογραφία του κινητού. Πατώντας την εικόνα δημιουργείται ένα νέο instance από την κλάση `smartphone` και εμφανίζεται. Το κινητό εμφανίζεται σε ένα νέο παράθυρο και βρίσκεται ήδη στην εφαρμογή σύνδεσης του με το έξυπνο ψυγείο. Υπάρχουν 3 πλήκτρα διαφορετικών λειτουργιών που προσφέρει και 2 κρυφά πλήκτρα όσων αφορά το `home` και το `back button`. Όλα όσα γίνονται μέσα στο κινητό είναι αποτελέσματα `controls` που εμφανίζονται και κρύβονται αλλάζοντας κάθε φορά το `property` του `control` από `hidden` σε `visible` και ανάποδα. Το `home` και το `back button` είναι αόρατα στις θέσεις των πραγματικών κουμπιών του κινητού και κρύβουν όλα τα `controls` εντός του παραθύρου του κινητού εκτός από τα 3 κουμπιά των λειτουργιών που τα εμφανίζουν για ευνόητους λόγους.

## Control Panel

Το `control panel` αποτελείται από τα αρχεία `control_panel.xaml` και `control_panel.xaml.cs`. Πρόκειται για ένα παράθυρο που αποτελείται από ένα σύνολο `checkboxes` τα οποία δίπλα τους έχουν και μια ονομασία ενδεικτική για αυτό που κάνουν. Στο `cs` αρχείο του `control panel` δημιουργούνται 3 αντικείμενα ένα για κάθε δωμάτιο στην αρχή του προγράμματος άρατά για κάθε μέθοδο. Κάθε `checkbox` έχει 2 δικά του `event` ένα για όταν τσεκάρετε και ένα για όταν ξετσεκάρεται. Τα `checkboxes` που αφορούν τα φώτα εμφανίζουν το νέο αντικείμενο που έχει δημιουργηθεί νωρίτερα εμφανίζοντας το κρυφό `control` τύπου `image lights` το οποίο έχει το φόντο για κάθε δωμάτιο σε πιο φωτεινό. Τα `checkboxes` σχετικά με τα `καλοριφέρ` και τα `air condition` κάνουν `visible` το `property` των `textblocks` με την θερμοκρασία που υπάρχουν κρυφά σε κάθε δωμάτιο, δίνουν τιμή και αλλάζουν το χρώμα της γραμματοσειράς (`foreground property` ίσο με `Brushes.Red`) κόκκινο για το `καλοριφέρ` μπλε για το `air condition`. Ακόμη για τα δωμάτια που έχουν και `καλοριφέρ` και `air condition` (σαλόνι, υπνοδωμάτιο) με το που γίνει `checked` κάποιο από τα 2 αυτόματα μέσω του `property IsChecked` ξετσεκάρεται το άλλο. Με αυτό τον τρόπο ποτέ δεν μπορεί σε ένα δωμάτιο να είναι ανοιχτό και το `καλοριφέρ` και το `air condition`. Για το ηχοσύστημα το οποίο βρίσκεται στο σαλόνι με το τσεκάρισμα εμφανίζεται ένα `textblock` που γράφει ποιο τραγούδι παίζει εκείνη τη στιγμή, ενώ για την τηλεόραση γίνεται μια εικόνα `visible` στη θέση της οθόνης για να υποδηλώσει ότι άνοιξε. Και για τα 2 αυτά `control` όταν ξετσεκάρονται τα `properties visibility` γίνονται ξανά `hidden`.

## Έξυπνο Ψυγείο

Το Έξυπνο Ψυγείο αποτελείται από τα αρχεία `smart_fridge.xaml` και `smart_fridge.xaml.cs`. Το παράθυρο αυτό αποτελεί το πάνελ αλληλεπίδρασης του έξυπνου ψυγείου. Όταν ανοίγει ο χρήστης βρίσκεται στην επιφάνεια εργασίας του και στο πάνω μέρος της γραμμής εντολών μπορεί να δει όλες τις εφαρμογές που μπορεί να τρέξει, επίσης στο πάνω δεξιά μέρος της οθόνης βρίσκονται τα δύο πλήκτρα του πάνελ `home` και `button`. Με αυτά τα δύο πλήκτρα ο χρήστης μπορεί να μεταβεί στην αρχική οθόνη (κάνοντας `hidden` το `visibility property` του συνόλου των `controls` που εμπεριέχει. Κάθε εικονίδιο εφαρμογής εντός του πάνελ εμπεριέχεται

σε ένα σχεδόν αόρατο button πάνω στο οποίο click event είναι κάθε φορά γραμμένα τα controls που πρέπει να εμφανιστούν και να εξαφανιστούν. Επειδή τα controls ιεραρχούνται όσον αφορά ποιο εμφανίζεται πάνω από ποιο κάθε νέα εφαρμογή που ανοίγει κάνει αυτόματα και ένα κρυφό πάτημα του κουμπιού home ώστε να σπάσει αυτή η ιεράρχηση και να μην φαίνονται controls κάτω από άλλα. Ένα χαρακτηριστικό που δεν έχει εμφανιστεί ως τώρα είναι τα slider υπάρχουν δύο μέσα στο πάνελ ένα για το ραδιόφωνο και ένα για την εσωτερική θερμοκρασία του ψυγείου, και στις δύο περιπτώσεις το event πάνω στο slider είναι το value change στο οποίο το όρισμα e που φέρει την τιμή του value αποδίδεται σε μία μεταβλητή double και αυτή τυπώνεται πίσω σε κάποιο κατάλληλο textblock με ακρίβεια ενός δεκαδικού ψηφίου.

## **Έξυπνη Ντουλάπα**

Η Έξυπνη Ντουλάπα αποτελείται από τα αρχεία smart\_wardrobe.xaml και smart\_wardrobe.xaml.cs. Το παράθυρο της στο μεγαλύτερο του κομμάτι αποτελείται από ένα control τύπου image που αποτελεί το εσωτερικό της ντουλάπας ενώ στο δεξιά μέρος του παραθύρου υπάρχουν μερικά κουμπιά. Υπάρχουν 2 κουμπιά ένα μαύρο και ένα λευκό για το φύλο του χρήστη και τέσσερα κουμπιά τα οποία αναφέρονται στο είδος ντυσίματος (καθημερινό, επίσημο, αθλητικό, εργασίας) τα οποία με το πάτημα τους ελέγχουν με ένα if την Boolean μεταβλητή gender\_clicked που υποδηλώνει το φύλλο που έχει επιλεγεί. Αν η μεταβλητή έχει τιμή true τότε επιλέγεται να εμφανιστεί το control image το οποίο περιέχει την ανάλογη κατάσταση ντυσίματος με το κατάλληλο φύλο. Η διαδικασία αυτή γίνεται με αλλαγή του property του κατάλληλου από τα 8 image σε visible, αυτή η ανορθόδοξη διαδικασία γίνεται διότι δεν γίνεται σε εφαρμογές τύπου WPF να αλλάξει δυναμικά το source του image. Κάθε φορά το φύλο που επιλέγεται αποκτά μαύρο χρώματος background ενώ το άλλο γίνεται λευκό και το αντίθετο για το foreground color που είναι το χρώμα του text.

## Επίλογος

Η εφαρμογή Έξυπνο Σπίτι έχει σκοπό να προσωμοιώσει πολλές τεχνολογικές καινοτομίες και αυτό τελικά κάνει. Δεν είναι αυτοσκοπός της να είναι πλήρως λειτουργικά τα controls αλλά να δείξει σαν προτότυπο τι κινήσεις πρέπει να γίνουν. Το σκοπό της το πετυχαίνει πολύ καλά και είναι πολύ φιλική προς το χρήστη και εύκολη στην κατανόηση. Από προγραμματιστικής απόψης δεν έχει και τις πλέον βέλτιστες λύσεις και υστερεί στο ότι πολλά controls χρησιμοποιούνται μόνο για να εμφανίζονται και να κρύβονται δηλαδή προφοτώνονται στην μνήμη μεγαλύτερος όγκος δεδομένων από όσα τελικά αξιοποιούνται κάθε στιγμή. Ένα τελευταίο αρνητικό είναι το γεγονός πως κάθε δωμάτιο αποτελεί διαφορετική κλάση και τα controls που φέρει δεν είναι τύπου static ώστε να μπορούν να τροποποιηθούν στο ίδιο αντικείμενο και από άλλες κλάσεις με αποτέλεσμα να πρέπει να δημιουργηθούν νέα αντικείμενα όταν επιφέρουμε αλλαγές.