

ΙΟΝΙΟ ΠΑΝΕΠΙΣΤΗΜΙΟ

ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ



- Πτυχιακή Εργασία -

**Σχεδιασμός και υλοποίηση μοντελοποίησης
γενικευμένου σεναρίου και διαδικτυακού API & βάσης
δεδομένων για παιχνίδια escape room**

Καλερίδης Ηλίας Π2018091

Επιβλέπων: Ιωάννης Καρύδης

18 Μαρτίου 2024

Επιβλέπων

Ιωάννης Καρύδης, Επίκουρος Καθηγητής, Τμ. Πληροφορικής, Ιόνιο Πανεπιστήμιο

Τριμελής Επιτροπή

Ιωάννης Καρύδης, Επίκουρος Καθηγητής, Τμ. Πληροφορικής, Ιόνιο Πανεπιστήμιο

Αγγελική Τσώχου, Αναπληρώτρια Καθηγήτρια, Τμ. Πληροφορικής, Ιόνιο Πανεπιστήμιο

Μάρκος Αυλωνίτης, Καθηγητής, Τμ. Πληροφορικής, Ιόνιο Πανεπιστήμιο

Περίληψη

Τα τελευταία χρόνια η δημοτικότητα των εργαλείων no/low-code έχει αυξηθεί. Η αύξηση αυτή προκύπτει επειδή τα συγκεκριμένα εργαλεία διευκολύνουν τους χρήστες με περιορισμένες τεχνικές γνώσεις να συμβάλλουν στην ανάπτυξη ενός ψηφιακού προϊόντος, αυξάνοντας έτσι την προσαρμοστικότητα του προϊόντος στις ανάγκες του πελάτη. Από την άλλη πλευρά, τα παραδοσιακά εργαλεία ανάπτυξης με κώδικα προσφέρουν διαφορετικές δυνατότητες. Η παρούσα έρευνα εξετάζει, λοιπόν, ποια εργαλεία είναι περισσότερο κατάλληλα για διάφορους τύπους και μεγέθη επιχειρήσεων, βάσει των πλεονεκτημάτων και μειονεκτημάτων τους. Επιπλέον, όπως αναφέρεται στη σχετική βιβλιογραφία, ο τομέας των no/low-code εργαλείων μπορεί να επωφεληθεί από περισσότερες μελέτες χρησιμότητας. Η ιστοσελίδα που αναπτύχθηκε για τις ανάγκες της εργασίας βοηθά τον τελικό χρήστη να δημιουργήσει και να οργανώσει πληροφορία που αφορά τα escape rooms, που επιλέχθηκαν λόγω της δημοτικότητας και του εκπαιδευτικού τους ρόλου. Η ιστοσελίδα αναπτύχθηκε χρησιμοποιώντας ένα no/low-code εργαλείο, το Bubble, και ένα παραδοσιακό εργαλείο ανάπτυξης, το Django, συμπεριλαμβανομένων της διεπαφής χρήστη, μιας βάσης δεδομένων και ενός API. Σκοπός ήταν να συγκριθούν οι δύο μέθοδοι ανάπτυξης με τη συλλογή τόσο ποσοτικών όσο και ποιοτικών δεδομένων. Για τα ποιοτικά δεδομένα πραγματοποιήθηκαν συνεντεύξεις μικρής κλίμακας με άτομα που είχαν χρησιμοποιήσει και τα δύο εργαλεία σε μια άσκηση. Συμπεραίνεται ότι τα no/low-code εργαλεία είναι ιδανικά για εταιρείες που επιδιώκουν τη μεγιστοποίηση των πόρων τους με μη υπερβολικά απαιτητικές ανάγκες για το προϊόν τους, όπως συμβαίνει συνήθως με τις εταιρείες start-up. Αντιθέτως, τα εργαλεία με κώδικα ταιριάζουν καλύτερα σε πιο εδραιωμένες εταιρείες με πιο εξειδικευμένες ανάγκες, όπου τα προϊόντα απαιτούν περισσότερη βελτιστοποίηση και πολυπλοκότητα στις λειτουργίες τους. Το συμπέρασμα που προκύπτει από την εργασία αυτή υπογραμμίζει την ανάγκη για μια ισορροπημένη προσέγγιση στην επιλογή εργαλείων ανάπτυξης, που να βασίζεται στις συγκεκριμένες ανάγκες και στόχους κάθε εταιρείας.

Abstract

In the past few years the popularity of no/low-code tools has risen. This is because these tools make it easier for users with limited technical skills to contribute in the production of a digital product, leading to a more adaptable product in regards to client needs. On the other hand, there are the traditional development tools that use code, which offer different possibilities. This research paper looks into which tool is better suited for different types and sizes of companies based on each tool's advantages and disadvantages. What's more, as discussed in related literature, the no/low-code sector could benefit from more usability studies. The web application developed for this paper, allows the end user to create and organize information about escape rooms. These were chosen based on their popularity and their role as educational tools. The application was developed using one no/low code tool, Bubble, and a traditional development tool, Django, including a graphical interface, a database and an API. The goal was to compare the two development methods with the collection of both quantitative and qualitative data. For the qualitative data, a small number of interviews were taken from individuals who had completed an exercise with each development tool. From the collected data, it is concluded that no/low-code tools are ideal for companies that look to maximize their resources while at the same time developing a product with moderate requirements. These companies are usually start-ups. On the contrary, code development tools are better suited for companies with more specialized needs, whose products demand more optimization and more complex features. These are usually more established companies. As a result, this paper underlines the importance of a balanced approach when selecting a development tool, that is based on the specific needs and goals of each company.

Ευχαριστίες

Η εκπόνηση αυτής της πτυχιακής εργασίας δε θα ήταν δυνατή χωρίς τους ανθρώπους που με στήριξαν κατά τη δημιουργία της. Αυτό το σημείο, αλλά και η εργασία, είναι για εκείνους. Η διαδικασία, δεν ήταν εύκολη και η ολοκλήρωση της γεννά τόσο θετικά αλλά και αρνητικά συναισθήματα. Από τη μία σηματοδοτεί τη λήξη των προπτυχιακών σπουδών και των δυσκολιών που αυτή συνεπάγεται αλλά από την άλλη σημαίνει και την λήξη των φοιτητικών ετών. Γενικά, θα ήθελα να ευχαριστήσω όλους τους φίλους, συνεργάτες, καθηγητές και συναδέλφους που γνώρισα καθόλη τη διάρκεια των σπουδών μου, αλλά φυσικά και την οικογένεια μου που μου έδωσαν τη δυνατότητα γι' αυτές. Ειδικότερα, για τη συνεργασία σε αυτή την εργασία θα ήθελα να ευχαριστήσω τον επιβλέποντα καθηγητή μου κ. Ιωάννη Καρύδη και τη στενότερη μου συνεργάτη κ. Ελένη Βλάχου. Οι συνεργασία μας ήταν άσογη και η βοήθεια τους ανεκτίμητη. Επίσης ευχαριστώ το Σπύρο Τσουκαλά που μου έδωσε μια εξαιρετική ιδέα που συμπεριλήφθηκε στο πείραμα της εργασίας. Ευχαριστώ την κοπέλα μου, που με στήριξε και ήταν πάντα εκεί για να με βοηθάει και να μου δίνει κίνητρο. Οι φίλοι μου είναι ένα αναπόσπαστο κομμάτι της ζωής μου και η εργασία αυτή δε θα μπορούσε να είναι πραγματικότητα χωρίς εκείνους. Ευχαριστώ για το έργο τους όλους όσους με βοήθησαν στην δημιουργία και εκπόνηση αυτής της εργασίας. Ευχαριστώ επιπλέον όσους ήταν μέρος αυτής της εργασίας δωρίζοντας το χρόνο τους με καλή διάθεση και θέληση, παρόλες τις προσωπικές τους συνθήκες, για τη συλλογή απαραίτητων δεδομένων. Το μεγαλύτερο ευχαριστώ όλων πηγαίνει στην οικογένεια μου. Χωρίς αυτούς και τη στήριξή τους τίποτα από αυτά δε θα ήταν δυνατό.

Περιεχόμενα

A' Εισαγωγή.....	7
A'.1 Ορισμός του escape room.....	7
A'.2 Κίνητρα και συνεισφορά.....	8
B' Βιβλιογραφική Έρευνα.....	10
B'.1 Ιστορική Αναδρομή escape room.....	10
B'.2 Υπάρχουσα βιβλιογραφία για τα escape rooms.....	12
B'.3 No-code εργαλεία.....	13
Γ' Μεθοδολογία.....	14
Γ'.1 Μεθοδολογία.....	14
Γ'.2 Σκοπός.....	14
Γ'.3 Απαιτήσεις χρήστη.....	15
Γ'.4 Δομή.....	17
Γ'.5 Βάση δεδομένων.....	19
Γ'.6 Παρουσίαση σελίδων.....	24
Γ'.6.1 Εργαλεία κώδικα.....	24
Γ'.6.1.1 Βάση δεδομένων.....	25
Γ'.6.1.2 Αρχική σελίδα.....	27
Γ'.6.1.3 Εγγραφή.....	28
Γ'.6.1.4 Σύνδεση.....	31
Γ'.6.1.5 Σελίδα προβολής λίστας δωματίων.....	33
Γ'.6.1.6 Δημιουργία δωματίου.....	34
Γ'.6.1.7 Διαγραφή δωματίου.....	36
Γ'.6.1.8 Προβολή δωματίου.....	37
Γ'.6.1.9 Επεξεργασία δωματίου.....	38
Γ'.6.1.10 Υπόλοιπες σελίδες.....	40
Γ'.6.2 Εργαλεία “no/low-code”.....	42
Γ'.6.2.1 Βάση δεδομένων.....	43
Γ'.6.2.2 Αρχική σελίδα.....	45
Γ'.6.2.3 Εγγραφή.....	47
Γ'.6.2.4 Σύνδεση.....	49
Γ'.6.2.5 Σελίδα προβολής λίστας δωματίων.....	51
Γ'.6.2.6 Δημιουργία δωματίου.....	54
Γ'.6.2.7 Διαγραφή δωματίου.....	57
Γ'.6.2.8 Προβολή δωματίου.....	59
Γ'.6.2.9 Επεξεργασία δωματίου.....	61
Γ'.6.2.10 Υπόλοιπες σελίδες.....	64
Δ' Πείραμα.....	67
Δ'.1 Περιγραφή Πειράματος.....	67
Δ'.2 Δεδομένα.....	69
Δ'.2.1 Χρόνος ανάπτυξης της εφαρμογής από το δημιουργό.....	69
Δ'.2.2 Απόκλιση αποτελέσματος από το επιθυμητό.....	71

Δ'.2.3 Ποιοτικά στοιχεία υποκειμενικού χαρακτήρα.....	72
Δ'.2.4 Υποστήριξη λειτουργιών από τα εργαλεία.....	79
Δ'.2.5 Κόστος ανάπτυξης.....	79
Δ'.3 Ανάλυση.....	80
Δ'.3.1 Περιορισμοί.....	80
Δ'.3.2 Αρνητικά αποτελέσματα.....	81
Ε' Συζήτηση.....	81
Ε'.1 Συμπεράσματα.....	82
Ε'.2 Μελλοντικές επεκτάσεις.....	83
Βιβλιογραφία.....	84
Γλωσσάρι Ξενικών Όρων.....	86

Κεφάλαιο Α'

Α' Εισαγωγή

Τα escape room είναι μια κατηγορία παιχνιδιών στα οποία συχνά, αλλά όχι πάντα, μια ομάδα συμμετεχόντων προσπαθεί να ολοκληρώσει. Ο στόχος που πρέπει συχνότερα να επιτευχθεί είναι η απόδραση από ένα φυσικό δωμάτιο αλλά δεν περιορίζεται εκεί. Μπορεί επιπλέον να είναι η συγκέντρωση πόντων μέσω διαφόρων δραστηριοτήτων ή ακόμα και το κυνήγι θησαυρού. Συχνά αυτός ο στόχος πρέπει να επιτευχθεί εντός συγκεκριμένου χρονικού ορίου ώστε οι παίκτες να είναι νικητές.

Τα escape room είναι μια σχετικά νέα δραστηριότητα. Η εμφάνιση και η αύξηση της δημοτικότητας τους ανά τον κόσμο έγινε κυρίως τη δεκαετία του 2010. Από τότε εξυπηρετούν δύο κύριου σκοπούς την ψυχαγωγία και την εκπαίδευση. Τα ψυχαγωγικά escape room έχουν εξαπλωθεί σε όλο τον κόσμο και αποτελούν πλέον μια επιλογή διασκέδασης για όλους. Τα εκπαιδευτικά escape room είναι λιγότερο διαδεδομένα, αλλά η ακαδημαϊκή κοινότητα υποστηρίζει πως αποτελούν ιδανικό εργαλείο για την εκπαίδευση ομάδων ατόμων. Μία επίσης σχετικά νέα συζήτηση που απασχολεί την επιστημονική κοινότητα είναι η υιοθέτηση των εργαλείων no/low-code. Αυτά μπορούν ενδεχομένως να προσφέρουν δυνατότητες σε ανθρώπους και κλάδους που δεν μπορούν τα παραδοσιακά εργαλεία ανάπτυξης λογισμικού. Με την ευκολία που προσφέρουν στο χρήστη, επιτρέπουν τη χρήση τους από άτομα με μικρότερο τεχνικό υπόβαθρο. Επιπλέον είναι δυνατή η χρήση τους σε κλάδους εκτός από αυτόν της πληροφορικής.

Σε αυτή την εργασία συνδυάζονται τα δύο αυτά θέματα. Έτσι έχει δημιουργηθεί μια διαδικτυακή εφαρμογή που επιτρέπει στους τελικούς χρήστες να σχεδιάσουν ένα escape room. Το πετυχαίνει αυτό δημιουργώντας ένα πλαίσιο στο οποίο οι χρήστες είναι ελεύθεροι να δημιουργήσουν το δικό τους αποτέλεσμα. Η εφαρμογή αυτή αναπτύχθηκε με δύο τρόπους. Με ένα πιο κλασικό εργαλείο ανάπτυξης διαδικτυακών εφαρμογών με κώδικα το Django και ένα εργαλείο no/low-code το Bubble. Στη συνέχεια διεξάγεται πείραμα από το οποίο εξάγονται συμπεράσματα για τις εφαρμογές του κάθε εργαλείου στον πραγματικό κόσμο. Το σενάριο στη συγκεκριμένη περίπτωση αφορά τη χρήση αυτών των εργαλείων από εταιρίες διαφορετικού μεγέθους και απαιτήσεων. Αναλύεται δηλαδή, αν το καθένα από αυτά τα εργαλεία είναι καταλληλότερο για χρήση από μια startup ή μια μεγαλύτερη εταιρία.

Α'.1 Ορισμός του escape room

Τα escape rooms είναι ατομικά ή ομαδικά παιχνίδια παζλ και γρίφων, όπου ο στόχος είναι η απόδραση από ένα δωμάτιο στο οποίο οι παίκτες είναι κλειδωμένοι. Στο παρόν, με τον όρο “escape room” εννοούμε συνήθως ένα φυσικό δωμάτιο από το οποίο οι παίκτες, που βρίσκονται με φυσική παρουσία εντός του δωματίου, πρέπει να αποδράσουν. Επίσης, αυτού του είδους τα escape rooms είναι κατά κανόνα ομαδικά. Σήμερα, τα escape rooms ελκύουν κατά 37% παίκτες άνω των 21, κατά 14% οικογένειες, 19%

είναι κάτω των 21, ενώ οι εταιρικοί πελάτες είναι στο 19%. Τέλος το 11% είναι ζευγάρια που το επέλεξαν για το ραντεβού τους. Όσον αφορά στο φύλο των παικτών, τα escape rooms έχουν εξίσου παίκτες του αρσενικού και του θηλυκού φύλου. Η πλειοψηφία των ομάδων απαρτίζονται από παίκτες και των δύο φύλων, ενώ ομάδες με αποκλειστικά αρσενικά ή θηλυκά μέλη είναι εξίσου κοινές (Nicholson, 2015). Δύο σημαντικοί όροι που πρέπει να εντάξουμε είναι αυτοί της θεματικής ενότητας και του σεναρίου. Ένα δωμάτιο μπορεί να μην έχει τίποτα από τα δύο, μπορεί να έχει μόνο ένα από τα δύο, αλλά μπορεί να έχει και τα δύο. Εδώ δεν υπάρχει κάποια προσέγγιση που να είναι καλύτερη από τις άλλες. Το είδος του δωματίου, που θα επιλέξει ο κάθε με βάση την θεματική ενότητα και το σενάριο, είναι κάτι που βασίζεται στο υποκειμενικό του γούστο. (Nicholson, 2015)

Α'.2 Κίνητρα και συνεισφορά

Ένα καλά σχεδιασμένο escape room μπορεί να προσφέρει αξία με διάφορους τρόπους. Όπως αναλύεται παρακάτω, ένα τέτοιο δωμάτιο, μπορεί να προσφέρει εκπαίδευση, ψυχαγωγία αλλά και ανάπτυξη διαφόρων δεξιοτήτων. Με μια πρώτη ματιά, θα έλεγε κανείς πως ένα escape room μπορεί να προσφέρει ψυχαγωγία ή εκπαίδευση και ανάπτυξη δεξιοτήτων. Κάτι τέτοιο όμως δεν ισχύει. Ακόμα και ένα escape room σχεδιασμένο για να προσφέρει ψυχαγωγία, αναπτύσει ταυτόχρονα κάποιες δεξιότητες του παίκτη. Είναι ενδιαφέρον πως πολλές φορές η εξάσκηση αυτών γίνεται χωρίς να το αντιληφθεί ο παίκτης. Το γεγονός πως ψυχαγωγία και ανάπτυξη δεξιοτήτων λαμβάνουν χώρα ταυτόχρονα, καθιστά τα escape room έναν ευχάριστο τρόπο να καλλιεργήσει κανείς κάποια μέρη του χαρακτήρα του. Με την ίδια λογική, τα escape room είναι και ένας εξαιρετικός τρόπος διδασκαλίας.

Αναλυτικότερα, ένα τέτοιο δωμάτιο περιέχει τόσο νοητικούς γρίφους όσο και γρίφους που απαιτούν τη φυσική ενέργεια. Οι γρίφοι που θέλουν τους παίκτες να κάνουν κάποιες φυσικές ενέργειες, σπάνια θα αποτελέσουν εμπόδιο για κάποιο άτομο με βάση τις φυσικές του ικανότητες. Δεν απαιτούν δηλαδή εξαιρετική φυσική κατάσταση ή κάποια συγκεκριμένα φυσικά χαρακτηριστικά όπως, για παράδειγμα, το ύψος. Έχουν, όμως, ως στόχο την εμπύθιση του χρήστη στο περιβάλλον που προσπαθούν να προσομοιώσουν. Αναπόσπαστο χαρακτηριστικό των escape room είναι η ομαδική δουλειά. Όταν λοιπόν αυτή συνδυάζεται με παζλ που αποσκοπούν στην σκέψη, δημιουργείται μια δοκιμασία που μπορεί να λυθεί μόνο από μια ομάδα που έχει την ικανότητα να επικοινωνήσει αποδοτικά και να δουλέψει συνεργατικά και αποτελεσματικά. Λόγω αυτού, οι παίκτες “αναγκάζονται” μέσω της διαδικασίας να εξελίσουν κομμάτια του χαρακτήρα τους που θα τους βοηθήσουν σε αυτό το σενάριο. Μερικά από αυτά παρουσιάζονται στον παρακάτω πίνακα Α'.1.

Δεξιότητα	Περιγραφή
Αναζήτηση	Είναι συχνό φαινόμενο οι παίκτες να πρέπει να αναζητήσουν στοιχεία.
Μνήμη	Οι παίκτες καλούνται να θυμούνται στοιχεία που έχουν

	συναντήσει, καθώς αυτά μπορεί να τους χρειαστούν στη συνέχεια.
Αναγνώριση μοτίβων	Πολλά παζλ/γρίφοι απαιτούν την παρατήρηση, επεξεργασία και αναγνώριση μοτίβων για να λυθούν.
Κατηγοριοποίηση	Συχνά υπάρχει η ανάγκη της διάσπασης και κατηγοριοποίησης ενός δωματίου σε μικρότερα κομμάτια.

Πίνακας Α'.1: Αναπτυσσόμενες δεξιότητες των παικτών escape room

Τα escape rooms άρχισαν να απασχολούν την επιστημονική κοινότητα από το 2015. Οι περισσότερες μελέτες πάνω σε αυτά αφορούν τους τρόπους που μπορούν να συνδυαστούν με την εκπαίδευση. Πολλές από αυτές υποστηρίζουν πως τα escape rooms είναι ένας τρόπος κονστρουκτιβιστικής εκπαίδευσης. Αυτός ο τύπος εκπαίδευσης βασίζεται στο γεγονός “ότι η μάθηση είναι μια διαδικασία προσαρμογής στην προσπάθεια του ατόμου να δομήσει καινούργια γνώση με τρόπο ενεργητικό” (Mayer, 1992)(Καραγιώργη & Συμεού, 2003). “Ο κονστρουκτιβισμός θεωρείται η πιο σύγχρονη θεωρία εκμάθησης και οι εκπαιδευτικές παρεμβάσεις που βασίζονται σε αυτή είναι η πιο αποτελεσματικές” (Olusegun & Olusegun, 2015)(Papaioannou et al., 2022). Η αξία των escape rooms, λοιπόν, είναι εμφανής, αν λάβει κανείς υπόψη την απότομη αύξηση στον αριθμό τους τα τελευταία χρόνια, από 2,800 το 2015 σε περισσότερα από 7,200 ανά τον κόσμο το 2018 (Fotaris & Mastoras, 2019). Αλλά και το αυξανόμενο ενδιαφέρον της επιστημονικής κοινότητας, το οποίο φαίνεται από τον αριθμό των επιστημονικών εργασιών όπου, από 3 το 2016, αυξήθηκαν σε 72 περίπου, μόνο την χρονιά του 2019 (Fotaris & Mastoras, 2019). Με αυτά τα δεδομένα, καταλαβαίνουμε πως υπάρχει και αυξανόμενη ενασχόληση με τη δημιουργία escape rooms, είτε αυτά είναι για ψυχαγωγία, είτε για εκπαίδευση.

Η δεύτερη πτυχή αυτής της εργασίας είναι η ανάπτυξη μιας διαδικτυακής εφαρμογής με τη χρήση δύο εργαλείων. Τα εργαλεία αυτά είναι, ένα περισσότερο παραδοσιακό εργαλείο κώδικα και ένα εργαλείο ανάπτυξης διαδικτυακών εφαρμογών no/low-code. Τα no/low-code εργαλεία δίνουν τη δυνατότητα σε άτομα με λιγότερες, ή και καθόλου, τεχνικές γνώσεις συμμετέχουν ενεργά στη διαδικασία ανάπτυξης. Έτσι μπορεί, γρηγορότερα, να επιτευχθεί ένα αποτέλεσμα που είναι πολύ περισσότερο κοντά στο αποτέλεσμα που θέλει ο πελάτης. Επιπλέον, αυτά τα εργαλεία, επιτρέπουν τη χρήση εφαρμογών πληροφορικής σε κλάδους που παραδοσιακά δεν έχουν άμεση τεχνική σχέση με αυτή. Οι (Pinho et al., 2023) υποστηρίζουν πως περισσότερες μελέτες χρησιμότητας μπορούν να βοηθήσουν τον κλάδο των no/low-code εργαλείων.

Έτσι το αντικείμενο αυτής της εργασίας είναι η δημιουργία μιας διαδικτυακής εφαρμογής η οποία θα έχει ως σκοπό να βοηθά το χρήστη να οργανώσει την πληροφορία που του χρειάζεται για την δημιουργία ενός escape room. Η εφαρμογή αυτή αποτελείται από γραφική διεπαφή, βάση δεδομένων αλλά και ένα API που μπορεί να χρησιμοποιήσει ο τελικός χρήστης. Ταυτόχρονα, σε αυτή την εργασία θα υπάρχει και πειραματικό κομμάτι στο οποίο θα συγκρίνεται η διαδικασία ανάπτυξης, της εν λόγω εφαρμογής, με εργαλεία που απαιτούν την χρήση κώδικα και με άλλα που την απαιτούν λίγο έως καθόλου. Αυτό έχει ως σκοπό να απαντήσει σε μια αρχική υπόθεση. Η υπόθεση αυτή αναφέρει πως τα no/low-code μπορούν να

επιτρέψουν τη γρηγορότερη ανάπτυξη ενός ελάχιστου βιώσιμου προϊόντος και δίνουν πρόσβαση σε μεγαλύτερο σύνολο εργαζομένων καθώς απαιτούν λιγότερη εξειδίκευση. Τα κλασικότερα εργαλεία με κώδικα, από τη άλλη, προσφέρουν περισσότερες δυνατότητες αλλά και επιτρέπουν την κλιμάκωση του προϊόντος ανάλογα με τις απαιτήσεις. Το προσωπικό, όμως, που απαιτούν αυτά τα εργαλεία πρέπει να έχει περισσότερη εξειδίκευση. Τα παραπάνω γεγονότα κάνουν τα no/low-code καταλληλότερα για νεοσύστατες εταιρείες τύπου start-up ενώ τα πιο παραδοσιακά εργαλεία για μεγαλύτερες και πιο εδραιωμένες εταιρείες.

Κεφάλαιο Β'

Β' Βιβλιογραφική Έρευνα

Β'.1 Ιστορική Αναδρομή escape room

Η βιβλιογραφική έρευνα γι' αυτή την εργασία αφορά δύο διαφορετικά θέματα. Τα escape rooms αλλά και τη χρήση code και no-code εργαλείων. Στην ενότητα αυτή υπάρχει βιβλιογραφική έρευνα για τα δύο αυτά θέματα.

Ο Nicholson (2015), μας περιγράφει μια σύντομη ιστορία των escape rooms. Εξηγεί πως, ενώ μπορεί κανείς να βρει στο διαδίκτυο πηγές που υποστηρίζουν πως τα πρώτα escape room εμφανίστηκαν το 2006, τα δεδομένα αυτά βασίζονται σε πληροφορίες που δεν είναι πλέον διαθέσιμες. Η πρώτη δραστηριότητα, που αποκάλεσε τον εαυτό της escape game και είναι καλά καταγεγραμμένη, ήταν από την εταιρεία SCRAP. Η δραστηριότητα αυτή λάμβανε χώρα στο Κιότο της Ιαπωνίας το 2007 και είχε τη μορφή ενός δωματίου, από το οποίο ομάδες των 5-6 ατόμων έπρεπε να αποδράσουν. Από την στιγμή εκείνη, η ανάπτυξη των escape rooms ήταν ραγδαία, στην αρχή στην Ασία (2012-2013) και στη συνέχεια στην Ευρώπη, την Αυστραλία, τις Ηνωμένες Πολιτείες και τον Καναδά.

Στη συνέχεια, ο Nicholson (2015), αναφέρει μερικούς από τους προγόνους που οδήγησαν, τελικά, στην δημιουργία των escape rooms. Αναφορικά αυτοί είναι:

1. Δια ζώσης παιχνίδια ρόλων
2. Παιχνίδια περιπέτειας με τον κέρσορα & Ψηφιακά παιχνίδια απόδρασης από δωμάτια
3. Κυνήγι θησαυρού
4. Διαδραστικό θέατρο & Στοιχειωμένα σπίτια
5. Τηλεοπτικά παιχνίδια και ταινίες περιπέτειας
6. Θεματική βιομηχανία ψυχαγωγίας

Αναλυτικότερα, τα δια ζώσης παιχνίδια ρόλων γεννήθηκαν τη δεκαετία του 1980 όταν η δημοτικότητα του παιχνιδιού Dungeons and Dragons έκανα τους παίκτες να θέλουν να ζήσουν το παιχνίδι τους με έναν πιο διαδραστικό τρόπο. Μερικά από τα σενάρια, που άρχισαν να δημιουργούνται από οργανωμένες προσπάθειες νεοσύστατων οργανισμών, έβαζαν στους παίκτες το εμπόδιο των γρίφων που θα έπρεπε να λύσουν για να προχωρήσουν στην πλοκή. Οι γρίφοι αυτοί είχαν τη μορφή αναζήτησης στοιχείων στο περιβάλλον ή επίλυση παζλ με σκοπό την έξοδο από κλειστά δωμάτια.

Τα παιχνίδια περιπέτειας με τον κέρσορα & ψηφιακά παιχνίδια απόδρασης από δωμάτια είναι ένας ακόμη πρόγονος των σημερινών escape rooms. Αυτά που ήταν βασισμένα στο κείμενο, είτε σε γραφική διεπαφή, ανάγκαζαν τους παίκτες να εξερευνήσουν τοποθεσίες, να βρουν και να συνδυάσουν αντικείμενα αλλά και να λύσουν γρίφους.

Μια ακόμη κατηγορία παιχνιδιών που προϋπήρχε και συνέβαλε στην δημιουργία των escape rooms, είναι τα παιχνίδια κυνηγιού θησαυρού. Από αυτά, τα escape rooms, έχουν κληρονομήσει το χαρακτηριστικό της ομαδικότητας και της συνεργασίας. Επίσης, έχουν παρόμοια δομή καθώς και στα δύο οι παίκτες λύνουν γρίφους που συνήθως είναι αλληλένδετοι και οδηγούν στην επίτευξη του τελικού στόχου. Η μεγαλύτερη διαφορά που έχουν τα escape rooms με τα παιχνίδια κυνηγιού θησαυρού, είναι πως τα πρώτα, συνήθως, λαμβάνουν χώρα σε εσωτερικό χώρο. Επίσης, τα παιχνίδια κυνηγιού θησαυρού είναι λιγότερο περιορισμένα ως προς τον τύπο των γρίφων που, κατά κανόνα, τα απαρτίζουν. Τα escape rooms συνηθίζουν να εστιάζουν σε γρίφους που βάζουν τους παίκτες να αλληλεπιδρούν με φυσικό τρόπο με τα αντικείμενα που βρίσκονται μέσα στο δωμάτιο, ενώ τα παιχνίδια κυνηγιού θησαυρού, λόγω της ελευθερίας που τους προσφέρει το περιβάλλον, μπορούν να περιέχουν μεγαλύτερη ποικιλία γρίφων.

Το διαδραστικό θέατρο, αν και όχι πρόγονος, θεωρείται μια πηγή έμπνευσης για κάποιους δημιουργούς escape room. Τα escape rooms και το διαδραστικό θέατρο μοιράζονται την προσέγγιση τους ως προς τον τρόπο ψυχαγωγίας του παίκτη. Αυτά θέλουν τον παίκτη να είναι ενεργός και ουσιαστικά να δημιουργεί ο ίδιος την ψυχαγωγία του μέσω της συμμετοχής του στα δρώμενα. Στην ίδια κατηγορία, ο Nicholson (2015), αναφέρει και τις εμπειρίες που προσφέρουν τα στοιχειωμένα σπίτια. Παρομοιάζει το τρόπο που, οι συμμετέχοντες σε ένα στοιχειωμένο σπίτι, κινούνται από δωμάτιο σε δωμάτιο ενώ αλληλεπιδρούν με φυσικό τρόπο με τα διάφορα στοιχεία του χώρου. Οι διαφορές τους με τα escape room είναι οι γρίφοι που οι παίκτες καλούνται να λύσουν σε μια εμπειρία escape room, ενώ παράλληλα πρέπει να περιηγηθούν στο χώρο αυτού.

Πολλά από τα χαρακτηριστικά των escape rooms έχουν έρθει από τη βιομηχανία του θεάματος και πιο συγκεκριμένα από τηλεοπτικές σειρές και ταινίες. Τηλεοπτικές σειρές που βγήκαν στον αέρα τη δεκαετία του 1980, είχαν ως κύριο θέμα ψυχαγωγίας, ομάδες παικτών που προσπαθούσαν να λύσουν μια πληθώρα από διαφορετικούς γρίφους ώστε να πετύχουν το στόχο τους. Ο στόχος αυτός ήταν άλλες φορές η απλή απόδραση από ένα δωμάτιο ή απλώς το δικαίωμα συνέχειας στον επόμενο γρίφο, μέχρι την προαγωγή στον τελικό γρίφο του παιχνιδιού όπου καθορίζεται το έπαθλο. Ένα από τα παιχνίδια τέτοιου περιεχομένου ήταν και το “Fort Boyard”, το οποίο προβλήθηκε και στην ελληνική τηλεόραση. Τηλεοπτικές σειρές όπως αυτές, αλλά και μεταγενέστερες όπως το “Survivor”, οδήγησαν στην αλλαγή της κουλτούρας δημιουργώντας τις κατάλληλες συνθήκες ώστε να αυξηθεί η δημοφιλία των escape rooms. Ιδιοκτήτες escape room αναφέρουν τις κινηματογραφικές ταινίες ως πηγή έμπνευσης κυρίως για τις θεματικές των περιπετειών των δωματίων τους. Ταινίες όπως το “Indiana Jones” και το “Saw” έχουν τη μεγαλύτερη ομοιότητα με τα escape rooms.

Σαν τελευταία πηγή έμπνευσης για τα escape room, ο Nicholson (2015), αναφέρει τη βιομηχανία της θεματικής ψυχαγωγίας. Αυτός ο όρος είναι, εσκεμμένα, αρκετά γενικός καθώς εδώ μας παρουσιάζει

μεμονωμένα παραδείγματα δραστηριοτήτων που αποτέλεσαν έμπνευση για κάποιους δημιουργούς escape room. Μία από αυτές τις δραστηριότητες ήταν το Entros, ένα εστιατόριο του Σιάτλ στο οποίο οι επισκέπτες συμμετείχαν σε παιχνίδια μυστηρίου τα οποία περιείχαν παζλ. Τα παζλ αυτά είχαν παρόμοια δομή με τα παζλ που συναντώνται σήμερα σε σύγχρονα escape room. Το Wits είναι μια Αμερικανική αλυσίδα που ιδρύθηκε το 2003. Σε αυτά οι παίκτες συμμετείχαν σε κάτι που μοιάζει γενικότερα πολύ με σύγχρονο escape room. Καλούνταν δηλαδή να λύσουν γρίφους φυσικού χαρακτήρα με περιορισμένο χρονικό περιθώριο και σκοπό την απόδραση προς το επόμενο δωμάτιο.

B'.2 Υπάρχουσα βιβλιογραφία για τα escape rooms

Οι βιβλιογραφία που μελετήθηκε για τα escape rooms και τη συγκεκριμένη εργασία αφορά κυρίως εφαρμογές των escape rooms σε διάφορα σενάρια εκπαιδευτικού χαρακτήρα. Οι (Brown et al., 2019) δημιούργησαν, με τη χρήση του (Breakout EDU), μία πρόκληση για μαθητές της νοσηλευτικής. Αντί για κλειδωμένα δωμάτια χρησιμοποιήθηκε το Breakout EDU. Αυτό είναι ένα εκπαιδευτικό κουτί το οποίο διαμορφώθηκε ώστε οι μαθητές να πρέπει να εφαρμόσουν τις γνώσεις τους ώστε να καταφέρουν να ξεκλειδώσουν τέσσερα λουκέτα στο διάστημα της μίας ώρας. Οι μαθητές που συμμετείχαν σε αυτό το παιχνίδι απόδρασης δήλωσαν πως η διαδικασία της παιχνιδοποίησης τους βοήθησε να μάθουν καλύτερα. Σύμφωνα με τον (Nicholson, 2018) υποστηρίζει πως τα παιχνίδια απόδρασης είναι ιδανικά για μια τάξη. Λέει πως αυτά επιτρέπουν στους μαθητές να βυθιστούν και να συγκεντρωθούν περισσότερο στο παιχνίδι. Επίσης προάγουν τη συνεργασία. Ένας ακόμη σημαντικός παράγοντας είναι πως αυτού του είδους τα παιχνίδια δε βασίζονται στο συντονισμό χεριού-ματιού αλλά στη συνεργασία και την ικανότητα επίλυσης παζλ.

Το φαινόμενο της βύθισης των παικτών σε ένα παιχνίδι απόδρασης (Escape Game) υποστηρίζουν και οι (Fotaris & Mastoras, 2019). Στην έρευνα τους που αποτελείται από ανασκόπηση 68 άλλων εργασιών αναφέρουν πως τα escape rooms πετυχαίνουν μία σειρά από αποτελέσματα. Εκπαιδεύουν τις κοινωνικές δεξιότητες (soft skills) των παικτών όπως η ομαδικότητα, η δημιουργικότητα, η λήψη αποφάσεων και άλλα, ενώ ταυτόχρονα είναι μία ευχάριστη δραστηριότητα. Τα escape room είναι ένα μέσο που, λόγω της φύσεως τους, μπορούν να χρησιμοποιηθούν για να βοηθήσουν στην εκμάθηση οποιουδήποτε θέματος (Papaioannou et al., 2022). Πέραν της ευελιξίας των escape room, αυτά καταφέρνουν να είναι και πολύ καλά εργαλεία. Αυτό συμβαίνει λόγω του τρόπου με τον οποίο είναι, κατά γενικό κανόνα, σχεδιασμένα. Ο τρόπος αυτός τα κάνει ιδανικά εργαλεία σύμφωνα με τη θεωρία του κονστρουκτιβισμού (Papaioannou et al., 2022). Ο κονστρουκτιβισμός υποστηρίζει πως η μάθηση αποτελείται από την προσπάθεια του ατόμου να κατασκευάσει γνώση γνώση (Καραγιώργη & Συμεού, 2003). Αυτό συνεπάγεται πως το άτομο συμμετέχει ενεργά στην απόκτηση της γνώσης αυτής και δεν είναι απλός παθητικός αποδέκτης (Καραγιώργη & Συμεού, 2003).

Για τη σχεδίαση του μοντέλου ενός escape room χρησιμοποιήθηκαν κυρίως δύο εργασίες. Οι (Nicholson, 2015) και (Wiemker et al., 2015). Η εργασία του (Nicholson, 2015) είναι μια επισκόπηση εγκαταστάσεων escape room. 175 εγκαταστάσεις ανά τον κόσμο συμμετείχαν. Από αυτές προέκυψαν κάποια στατιστικά στοιχεία για διάφορους από του παράγοντες ενός escape room. Έτσι, επιλέχθηκαν για το μοντέλο

οι παράγοντες που φαίνεται να είναι οι δημοφιλέστεροι. Στην εργασία τους, οι (Wiemker et al., 2015) περιγράφουν αναλυτικά διάφορους από τους παράγοντες ενός escape room. Περιγράφουν για παράδειγμα τη δομή ενός παζλ. Σύμφωνα με αυτούς το παζλ αποτελείται από μία πρόκληση, μία λύση και μία επιβράβευση. Αυτό είναι και το μοντέλο που υιοθετήθηκε για το παζλ στη συγκεκριμένη εφαρμογή. Περιγράφουν επίσης τους διαφορετικούς τρόπους που μπορεί τα παζλ να σχετίζονται μεταξύ τους και κατ' επέκταση, να είναι σχεδιασμένο το escape room.

B'.3 No-code εργαλεία

Στην εργασία τους οι (Pinho et al., 2023) τονίζουν μερικά χαρακτηριστικά που έχει η παρούσα κατάσταση των no/low-code εργαλείων. Παρουσιάζουν πως δεν υπάρχει επίσημος ορισμός για τα no/low-code εργαλεία, αλλά πως οι χρήστες είναι αρκετά εξοικειωμένοι με τον τρόπο χρήσης και λειτουργίας τους. Υποστηρίζουν επίσης πως περισσότερες έρευνες χρησιμότητας για τα no/low-code εργαλεία μπορούν να ωφελήσουν τον κλάδο. Μια τέτοια έρευνα χρησιμότητας είναι και η συγκεκριμένη εργασία. Οι (Al Alamin et al., 2021) ορίζουν ένα no/low-code εργαλείο σαν κάτι που συνδυάζει ελάχιστο πηγαίο κώδικα με μία γραφική διεπαφή χρήστη που σκοπός του είναι η ταχύτατη ανάπτυξη εφαρμογών. Επιγραμματικά βρίσκουν, μετά από έρευνα στο StackOverflow, πως οι 4 μεγάλες κατηγορίες που απασχολούν περισσότερο τους υιοθέτες της νέας αυτής τεχνολογίας είναι:

- Προσαρμογή (customization)
- Υιοθέτηση πλατφόρμας (Platform Adoption)
- Βάση δεδομένων (Database)
- Ενσωμάτωση (Integration)

Στην εργασία τους οι (Martinez & Pfister, 2023) υποστηρίζουν πως τα low-code εργαλεία είναι μια τεχνολογία που επιτρέπει την ανάπτυξη εφαρμογών σε άτομα με λίγη τεχνική εμπειρία στη συγγραφή κώδικα. Σύμφωνα με την έρευνά τους, τα low-code εργαλεία έχουν τη δυνατότητα να προωθήσουν την καινοτομία σε συγκεκριμένες διαδικασίες του κατασκευαστικού τομέα. Για το γεγονός πως τα no/low-code εργαλεία βοηθούν άτομα με μικρότερη τεχνική εμπειρία στον κώδικα, να συμμετέχουν ενεργά στην ανάπτυξη μια ψηφιακής εφαρμογής συμφωνούν και οι (Rokis & Kirikova, 2023). Επίσης, προσθέτουν πως κάτι τέτοιο οδηγεί σε καλύτερο αποτέλεσμα σε ό,τι αφορά τις απαιτήσεις του πελάτη και το τελικό προϊόν.

Κεφάλαιο Γ'

Γ' Μεθοδολογία

Γ'.1 Μεθοδολογία

Στις επόμενες ενότητες, αναλύονται οι δύο εφαρμογές που δημιουργήθηκαν, μια με κάθε μέθοδο ανάπτυξης στο κεφάλαιο Γ'. Εκεί αναλύεται ο σκοπός ή, με άλλα λόγια, η ανάγκη που προσπαθεί να καλύψει

η εφαρμογή αυτή. Επιπλέον, συζητείται ο τρόπος με τον οποίο προσπαθεί να πετύχει το στόχο αυτό μέσω μιας ανάλυσης των απαιτήσεων του χρήστη, μαζί με τα χαρακτηριστικά που προσφέρει η εφαρμογή για να ικανοποιήσει τις απαιτήσεις αυτές. Επίσης υπάρχει μια, υψηλού επιπέδου, παρουσίαση της δομής και της λειτουργίας της εφαρμογής. Τέλος, υπάρχουν δύο υποκεφάλαια, ένα για κάθε έναν από τους τρόπους ανάπτυξης, όπου παρουσιάζονται αναλυτικά όλες οι σελίδες που περιέχουν οι δύο υλοποιήσεις, με εικόνες αλλά και γραπτή ανάλυση του κάθε στοιχείου.

Ακολουθεί το κεφάλαιο Δ', το οποίο πραγματεύεται το πείραμα. Εκεί, υπάρχει περιγραφή του πειράματος, η οποία θα βοηθήσει στην κατανόηση του και των μηχανισμών που το απαρτίζουν. Στη συνέχεια, υπάρχουν υποκεφάλαια στα οποία θα αναλύονται, με περισσότερη λεπτομέρεια, κάθε μια από τις πτυχές του πειράματος. Αυτά τα υποκεφάλαια περιέχουν λεπτομέρειες για τα δεδομένα του πειράματος, ποια είναι, πως αποφασίστηκαν, πως συλλέχθηκαν και πως χρησιμοποιούνται για τις ανάγκες του πειράματος. Επιπροσθέτως, υπάρχει υποκεφάλαιο που παρουσιάζει τους περιορισμούς που μπορεί να υπάρχουν σε οποιαδήποτε από τις πλευρές του πειράματος αλλά και τυχόν αρνητικά αποτελέσματα. Αποτελέσματα δηλαδή, που δεν υποστηρίζουν την αρχική υπόθεση. Ο σκοπός αυτού του υποκεφαλαίου είναι να δώσει προοπτική στο πείραμα, τονίζοντας τις αδυναμίες του αλλά και μέσω της διαδικασίας αυτής να παρουσιάσει πως θα μπορούσε να βελτιωθεί.

Το επόμενο κεφάλαιο, Ε', αφορά τα συμπεράσματα τα οποία προέκυψαν από το πείραμα. Πιο συγκεκριμένα, υπάρχει παρουσίαση της αρχικής υπόθεσης και γίνεται ξεκάθαρο αν αυτή επιβεβαιώνεται ή καταρρίπτεται, με βάση τα συμπεράσματα που έχουν αντληθεί από τα αποτελέσματα του πειράματος. Υπάρχει επίσης συζήτηση που χαρακτηρίζεται περισσότερο από υποκειμενικότητα. Εκεί, υπάρχει ερμηνεία των ευρημάτων. Άλλα θέματα τα οποία παρουσιάζονται μέσα από υποκειμενικό φακό, αφορούν τις επιπτώσεις, θετικές ή αρνητικές, που είναι πιθανό να προκύπτουν από τα συμπεράσματα που θα βγουν από το πείραμα. Τέλος μελλοντικές επεκτάσεις ή/και βελτιώσεις που θα έκαναν την εργασία αυτή, στο σύνολο της, περισσότερο ολοκληρωμένη. Το κεφάλαιο Ε' σηματοδοτεί και το τέλος του κυρίως περιεχομένου, μετά από αυτό ακολουθούν το παράρτημα, η βιβλιογραφία και το γλωσσάρι ξενικών όρων.

Γ'.2 Σκοπός

Με την ξαφνική αύξηση της δημοτικότητας των escape rooms, τόσο στον χώρο της ψυχαγωγίας όσο και στον χώρο της επιστημονικής έρευνας, υπάρχει η νέα ανάγκη οργάνωσης αυτής της νέας πληροφορίας που αποτελεί ένα escape room. Έτσι λοιπόν, σκοπός αυτής της διαδικτυακής εφαρμογής είναι η διευκόλυνση του χρήστη να οργανώνει την πληροφορία που εκείνος θέλει για οποιαδήποτε περίπτωση escape room. Επίσης, η πληροφορία αυτή θα πρέπει να είναι διαθέσιμη από οποιονδήποτε τόπο και σε οποιαδήποτε στιγμή θελήσει ο χρήστης να τη δει, να την επεξεργαστεί, να την διαγράψει αλλά ακόμη και να δημιουργήσει νέα πληροφορία. Με τα παραπάνω η παρούσα εφαρμογή προσπαθεί να κάνει τη διαδικασία υλοποίησης ενός escape room ευκολότερη και γρηγορότερη, έτσι ώστε ο δημιουργός να μπορεί να επικεντρωθεί στην σχεδίαση αυτού και να καταφέρνει το escape room να πετυχαίνει καλύτερα τον στόχο του.

Στην συνέχεια αυτού του κεφαλαίου, παρουσιάζονται οι απαιτήσεις του χρήστη που θα πρέπει να καλύπτονται από το σύστημα, η δομή της εφαρμογής και, με τη βοήθεια των εικόνων, η φυσική μορφή που έχει η εφαρμογή σε αυτό το στάδιο.

Γ'.3 Απαιτήσεις χρήστη

Για να πετύχει η εφαρμογή τους στόχους που αναφέρθηκαν, θα πρέπει να καλύπτει συγκεκριμένες ανάγκες του χρήστη η οποίες λειτουργούν αλληλένδετα ώστε να βοηθούν το χρήστη να πάρει την αξία που θέλει να του προσφέρει η συγκεκριμένη εφαρμογή. Οι ανάγκες αυτές έχουν οριστεί πειραματικά, από το δημιουργό της εφαρμογής. Για να ελεγχθεί η κανονική λειτουργία των υπηρεσιών αυτών προς το χρήστη, έχουν οριστεί μια σειρά δοκιμών. Αναλυτικότερα, στον πίνακα Γ'.1. μπορούμε να δούμε την υπηρεσία που θα παρέχεται από την εφαρμογή (αριστερή στήλη) και την ανάγκη του χρήστη που σκοπεύει να ικανοποιήσει (δεξιά στήλη).

Υπηρεσία	Ανάγκη
Υπηρεσία δημιουργίας λογαριασμού.	Σαν μη εγγεγραμμένος χρήστης, θέλω να μπορώ να εγγραφώ (sign up), ώστε να μπορώ να σώσω την δουλειά μου.
Υπηρεσία σύνδεσης με υπάρχων λογαριασμό.	Σαν εγγεγραμμένος και αποσυνδεδεμένος (logged out) χρήστης, θέλω να μπορώ να συνδεθώ (log in), ώστε να μπορώ να δω τη δουλειά που έχω αποθηκεύσει.
Υπηρεσία δημιουργίας λογαριασμού.	Σαν εγγεγραμμένος και αποσυνδεδεμένος (logged out) χρήστης, θέλω να μπορώ να εγγραφώ, ώστε να έχω πολλούς λογαριασμούς.
Λειτουργία προφίλ όπου παρουσιάζεται, οργανωμένη η δουλειά του χρήστη, με δυνατότητα επεξεργασίας.	Σαν συνδεδεμένος χρήστης, θέλω να μπορώ να δω προηγούμενη μου δουλειά, ώστε να μπορώ να την επεξεργαστώ.
Λειτουργία προβολής.	Σαν συνδεδεμένος χρήστης, θέλω να έχω πρόσβαση σε προηγούμενη μου δουλειά, ώστε να μπορώ να την ξαναδώ.
Λειτουργία διαγραφής.	Σαν συνδεδεμένος χρήστης, θέλω να μπορώ να δω προηγούμενη μου δουλειά, ώστε να μπορώ να την διαγράψω.
Λειτουργία δημιουργίας και αποθήκευσης πληροφορίας.	Σαν συνδεδεμένος χρήστης, θέλω να μπορώ να δημιουργήσω καινούργιο escape room.

Υπηρεσία	Ανάγκη
Υπηρεσία δημιουργίας λογαριασμού.	Σαν μη εγγεγραμμένος χρήστης, θέλω να μπορώ να εγγραφώ (sign up), ώστε να μπορώ να σώζω την δουλειά μου.
Υπηρεσία σύνδεσης με υπάρχων λογαριασμό.	Σαν εγγεγραμμένος και αποσυνδεδεμένος (logged out) χρήστης, θέλω να μπορώ να συνδεθώ (log in), ώστε να μπορώ να δω τη δουλειά που έχω αποθηκεύσει.
Υπηρεσία δημιουργίας λογαριασμού.	Σαν εγγεγραμμένος και αποσυνδεδεμένος (logged out) χρήστης, θέλω να μπορώ να εγγραφώ, ώστε να έχω πολλούς λογαριασμούς.
Λειτουργία προφίλ όπου παρουσιάζεται, οργανωμένη η δουλειά του χρήστη, με δυνατότητα επεξεργασίας.	Σαν συνδεδεμένος χρήστης, θέλω να μπορώ να δω προηγούμενη μου δουλειά, ώστε να μπορώ να την επεξεργαστώ.
Λειτουργία προβολής.	Σαν συνδεδεμένος χρήστης, θέλω να έχω πρόσβαση σε προηγούμενη μου δουλειά, ώστε να μπορώ να την ξαναδώ.
Λειτουργία διαγραφής.	Σαν συνδεδεμένος χρήστης, θέλω να μπορώ να δω προηγούμενη μου δουλειά, ώστε να μπορώ να την διαγράψω.
Λειτουργία εξαγωγής του αποθετηρίου σε JSON μέσω API.	Σαν συνδεδεμένος χρήστης, θέλω να μπορώ να πάρω σαν έξοδο σε μορφή JSON λεπτομέρειες για όσα έχω δημιουργήσει.

Πίνακας Γ'.1: Απαιτήσεις χρήστη

Ακολουθεί η λίστα λίστα Γ'.1 όπου αναγράφονται οι δοκιμές που έχουν οριστεί από το δημιουργό της εφαρμογής, για τον έλεγχο της σωστής λειτουργίας της κάθε υπηρεσία με σκοπό την σωστή εξυπηρέτηση του χρήστη.

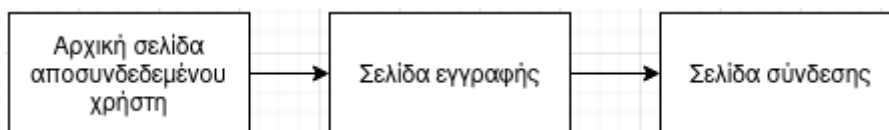
- Σε έναν μη εγγεγραμμένο (ή μη συνδεδεμένο) χρήστης, στην αρχική σελίδα, το σύστημα θα πρέπει να παρέχει δυνατότητα εγγραφής (sign up).
- Σε έναν μη εγγεγραμμένο (ή μη συνδεδεμένο) χρήστης, στην αρχική σελίδα, το σύστημα θα πρέπει να παρέχει δυνατότητα σύνδεσης (log in).
- Σε έναν μη εγγεγραμμένο χρήστη που βρίσκεται στη σελίδα εγγραφής, όταν συμπληρώνει τα στοιχεία του, τότε, το σύστημα θα πρέπει να του κάνει εγγραφή.

- Σε έναν μη εγγεγραμμένο χρήστη που βρίσκεται στη σελίδα εγγραφής, όταν συμπληρώνει τα στοιχεία του, τότε, το σύστημα θα πρέπει να τον ενημερώνει για την επιτυχία/αποτυχία της εγγραφής του.
- Σε έναν μη εγγεγραμμένο χρήστη που βρίσκεται στη σελίδα εγγραφής, όταν συμπληρώνει τα στοιχεία του, τότε, το σύστημα θα πρέπει να τον μεταβιβάζει στη σελίδα σύνδεσης.
- Σε ένας μη συνδεδεμένος χρήστης που βρίσκεται στη σελίδα σύνδεσης, όταν συμπληρώνει τα στοιχεία του το σύστημα θα πρέπει να τον ενημερώνει για την επιτυχία/αποτυχία της σύνδεσής του.
- Σε έναν μη συνδεδεμένο χρήστη που βρίσκεται στη σελίδα σύνδεσης, όταν συμπληρώνει τα στοιχεία του και η σύνδεση είναι επιτυχής, το σύστημα θα πρέπει να κάνει ξεκάθαρο ότι είναι συνδεδεμένος.
- Σε έναν συνδεδεμένο χρήστη που βρίσκεται στην αρχική σελίδα, όταν το επιλέξει, το σύστημα θα πρέπει να τον μεταβιβάζει στην σελίδα όπου μπορεί να δει τα escape room του.
- Σε έναν συνδεδεμένο χρήστη που βρίσκεται στην κατάλληλη σελίδα, όταν το επιλέξει, το σύστημα θα πρέπει να τον μεταβιβάζει στην σελίδα δημιουργίας.
- Σε έναν συνδεδεμένο χρήστη που βρίσκεται στην κατάλληλη σελίδα, όταν το επιλέξει, το σύστημα θα πρέπει να τον μεταβιβάζει στην σελίδα επεξεργασίας.
- Σε έναν συνδεδεμένο χρήστη που βρίσκεται στην κατάλληλη σελίδα, όταν το επιλέξει, το σύστημα θα πρέπει να διαγράφει μια οντότητα, αφού τον ρωτήσει για επιβεβαίωση

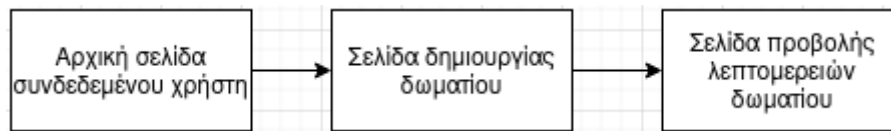
Λίστα Γ.1: Δοκιμές για τον έλεγχο σωστής λειτουργίας

Γ'.4 Δομή

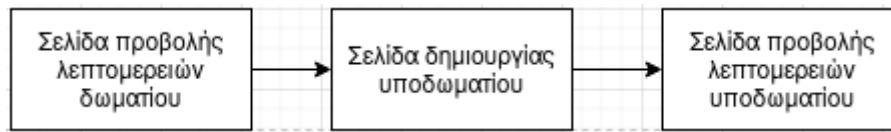
Η εφαρμογή που έχει αναπτυχθεί στα πλαίσια αυτής της πτυχιακής εργασίας, είναι ουσιαστικά, μια γραφική διεπαφή, στο διαδίκτυο, μέσω της οποίας ένας χρήστης μπορεί να αλληλεπιδράσει με μια βάση δεδομένων στην οποία υπάρχει όλη η πληροφορία που ενδεχομένως τον ενδιαφέρει. Στο πρωτότυπο της εφαρμογής, περιέχονται τρόποι με τους οποίους ο χρήστης μπορεί να κάνει χρήση των υπηρεσιών που έχουν προαναφερθεί. Στην εφαρμογή που έχει αναπτυχθεί δεν περιέχεται ωμή πληροφορία, καθώς ο σχεδιασμός αυτής είναι τέτοιος, όπου η πληροφορία προέρχεται από τους χρήστες και ο καθένας από αυτούς έχει πρόσβαση μόνο στην δική του. Η διάδραση του χρήστη με τη εφαρμογή, γίνεται κυρίως με την χρήση της γραφικής διεπαφής που παρέχεται. Υπάρχει όμως και η δυνατότητα χρήσης API (Application Programming Interface), μέσω του οποίου ο χρήστης μπορεί να πάρει σαν έξοδο πληροφορίες για όσα έχει δημιουργήσει. Όταν ένας χρήστης επισκέπτεται την ιστοσελίδα έχει τη δυνατότητα να περιηγηθεί σε αυτή μέσω κουμπιών τα οποία οδηγούν στις διάφορες σελίδες. Ακολουθούν παραδείγματα διαφόρων “μονοπατιών” που μπορεί ο χρήστης να ακολουθήσει για να πετύχει το σκοπό του.



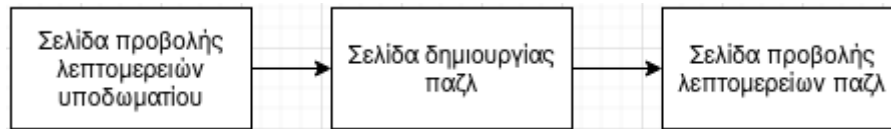
Σχήμα Γ.1.α: Διαδρομή του χρήστη για να κάνει εγγραφή.



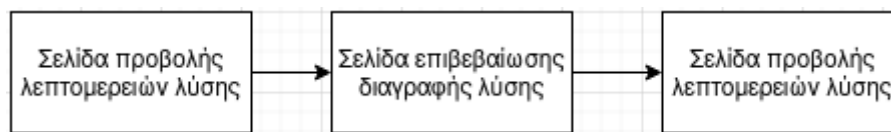
Σχήμα Γ'.1.β: Διαδρομή του χρήστη για τη δημιουργία δωματίου από την αρχική σελίδα συνδεδεμένου χρήστη.



Σχήμα Γ'.1.γ: Διαδρομή του χρήστη για δημιουργία υποδωματίου.



Σχήμα Γ'.1.δ: Διαδρομή του χρήστη για δημιουργία παζλ.

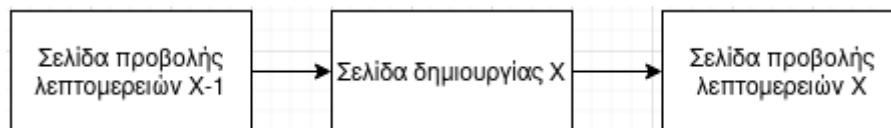


Σχήμα Γ'.1.ε: Διαδρομή του χρήστη για διαγραφή λύσης, με ακύρωση στη σελίδα επιβεβαίωσης διαγραφής.

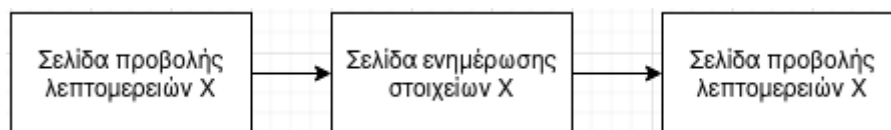
Σχήμα Γ'.1: Διάφορες διαδρομές που ακολουθεί ο χρήστης σύμφωνα με τα δεδομένα και τον στόχο του.

Οι διαδρομές που παρουσιάζονται παραπάνω, είναι απλός παραδείγματα των βημάτων που πρέπει να ακολουθήσει ο χρήστης, ανάλογα με τα δεδομένα και τον στόχο του. Αυτό σημαίνει πως, ανά πάσα στιγμή, ο χρήστης είναι αυτός που αποφασίζει τι θέλει να πετύχει. Έτσι δημιουργούνται πολλά διαφορετικά σενάρια για το χρήστη, πράγμα που σημαίνει πως η χρήση της εφαρμογής δεν ακολουθεί μια γραμμική σειρά βημάτων, αλλά δίνει στο χρήστη την ελευθερία να αποφασίσει εκείνος τι θέλει να κάνει και πως θα πετύχει. Για χάρη συντομίας, παραπάνω, δεν παρουσιάζονται όλα τα δυνατά σενάρια διαδρομών αλλά ένα μικρό μέρος αυτών. Στο παράρτημα υπάρχει σχεδιάγραμμα στο οποίο φαίνονται όλες οι πιθανές διαδρομές.

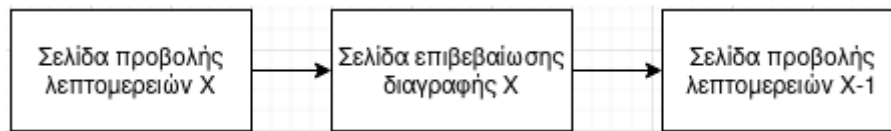
Η δομή που οδηγεί στη δημιουργία των μονοπατιών που ακολουθεί ο χρήστης περιέχει μέσα της επανάληψη. Αυτό θα φαίνεται ξεκάθαρα στο παράρτημα όπου θα μπορεί κανείς να δει όλα τα μονοπάτια. Ακολουθεί περιγραφή αυτής της επανάληψης για όσα μέρη της δομής την έχουν. Για τη δημιουργία νέας οντότητας X, με εξαίρεση της οντότητας “δωμάτιο”, ακολουθείται η παρακάτω γενική διαδρομή:



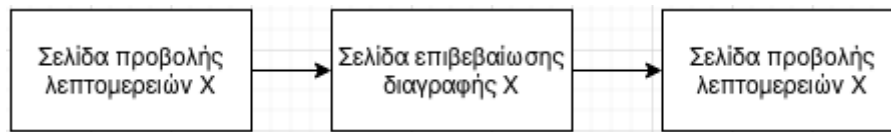
Σχήμα Γ'2.α: Γενική διαδρομή δημιουργίας αντικειμένου.



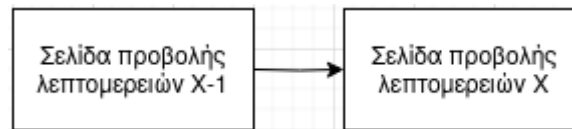
Σχήμα Γ'2.β: Γενική διαδρομή ενημέρωσης στοιχείων.



Σχήμα Γ'2.γ: Γενική διαδρομή διαγραφής οντότητας



Σχήμα Γ'2.δ: Γενική διαδρομή διαγραφής οντότητας με ακύρωση.



Σχήμα Γ'2.ε: Γενική διαδρομή προβολής λεπτομερειών οντότητας.

Σχήμα Γ'.2: Γενικές διαδρομές χρήστη.

Στο σχήμα Γ'2.α, βλέπουμε τη γενική διαδρομή που ακολουθεί ο χρήστης ώστε να δημιουργήσει μια νέα οντότητα. Έτσι το X παρουσιάζει την οντότητα που ο χρήστης σκοπεύει να δημιουργήσει και το $X-1$ παρουσιάζει την οντότητα που σχετίζεται με τη X με σχέση 1 ($X-1$) προς πολλά (X). Η συσχετίσεις παρουσιάζονται αναλυτικότερα παρακάτω. Η οντότητα “δωμάτιο” αποτελεί εξαίρεση καθώς δεν υπάρχει κάποια άλλη οντότητα στη θέση $X-1$. Στο σχήμα Γ'2.β, φαίνεται η διαδρομή που ακολουθεί ο χρήστης για να ενημερώσει τα στοιχεία μιας οντότητας. Σε κάθε σελίδα προβολής λεπτομερειών, υπάρχει σχετικό κουμπί που οδηγεί το χρήστη στη σελίδα ενημέρωσης στοιχείων. Από εκεί είτε ο χρήστης αλλάξει στοιχεία, είτε όχι, επιστρέφει στη σελίδα προβολή λεπτομερειών της αντίστοιχης οντότητας. Το Γ'3.γ δείχνει τη διαδρομή διαγραφής μια οντότητας. Παρόμοια με την ενημέρωση στοιχείων, κάθε σελίδα προβολής λεπτομερειών έχει κουμπί που οδηγεί στη σελίδα επιβεβαίωσης διαγραφής. Από εκεί ο χρήστης οδηγείται στη σελίδα προβολής λεπτομερειών της $X-1$ οντότητας εάν επιβεβαιώσει τη διαγραφή. Από εδώ εξαιρείται ξανά η οντότητα “δωμάτιο” όπου ο χρήστης καταλήγει στη σελίδα προβολής λίστας δωματίων. Εάν τελικά ακυρώσει τη διαγραφή, επιστρέφει στη σελίδα προβολής λεπτομερειών της οντότητας X όπως στο σχήμα Γ'.2.δ. Τέλος, στο σχήμα Γ'.2.ε φαίνεται η γενική διαδρομή που ακολουθεί ο χρήστης για να δει τις λεπτομέρειες κάθε οντότητας, με εξαίρεση για άλλη μια φορά, την οντότητα “δωμάτιο”. Σε αυτή ο χρήστης ξεκινά από την αρχική σελίδα συνδεδεμένου χρήστη.

Γ'.5 Βάση δεδομένων

Όπως ήδη αναφέρεται παραπάνω, η εφαρμογή που αναπτύχθηκε δεν είναι παρά μια διεπαφή (γραφική και API) που επιτρέπει στο χρήστη να αλληλεπιδρά με μία βάση δεδομένων. Η δομή, λοιπόν, αυτής της βάσης για τις ανάγκες της συγκεκριμένης εφαρμογής περιγράφεται σε αυτό το κεφάλαιο. Αποτελείται από 7 αλληλένδετες οντότητες, που σχετίζονται μεταξύ τους έτσι ώστε να δίνουν στο χρήστη μια βασική δομή, αλλά ταυτόχρονα και ελευθερία. Αυτό επιτυγχάνεται με τη χρήση συσχετίσεων “1-προς-πολλά”. Αυτές αν και περιορίζουν το χρήστη ως προς το ποιές οντότητες μπορεί να συσχετίσει, έτσι

δημιουργείται η δομή, του αφήνουν την ελευθερία να συσχετίσει όσες θέλει, έτσι δίνεται η ελευθερία.

Συγκεκριμένα, οι επτά οντότητες είναι:

- Χρήστης
- Δωμάτιο
- Υποδωμάτιο
- Παζλ
- Λύση
- Επιβράβευση
- Βοήθεια

Λίστα Γ'.2

Στην εικόνα Γ'.1 φαίνεται το διάγραμμα οντοτήτων συσχετίσεων της βάσης. Εκεί φαίνονται όλες οι οντότητες με τα χαρακτηριστικά που τις απαρτίζουν.

- Χρήστης
 - Όνομα Χρήστη (Username)
 - Κωδικός (Password)
- Δωμάτιο
 - Μοναδικός χαρακτηριστικός αριθμός (ID)
 - Σενάριο
 - Τίτλος
 - Επιλογή σχεδιασμού συσχετίσεων παζλ (Puzzle Path Design)
 - Ελάχιστος αριθμός παικτών
 - Μέγιστος αριθμός παικτών
 - Παρουσία Ηθοποιού
 - Στόχος
 - Δυσκολία
 - Χρονικό Περιθώριο
 - Θεματική (Theme)
 - Περίληψη του δωματίου (Brief)
 - Απολογισμός (Debrief)
- Υποδωμάτιο
 - Μοναδικός χαρακτηριστικός αριθμός (ID)
 - Τίτλος
- Παζλ
 - Τίτλος
 - Περιγραφή
 - Μοναδικός χαρακτηριστικός αριθμός (ID)
- Λύση
 - Μοναδικός χαρακτηριστικός αριθμός (ID)

- Περιγραφή
- Επιβράβευση
 - Μοναδικός χαρακτηριστικός αριθμός (ID)
 - Περιγραφή
- Βοήθεια
 - Μοναδικός χαρακτηριστικός αριθμός (ID)
 - Συνθήκη
 - Περιγραφή βοήθειας

Λίστα Γ'.3

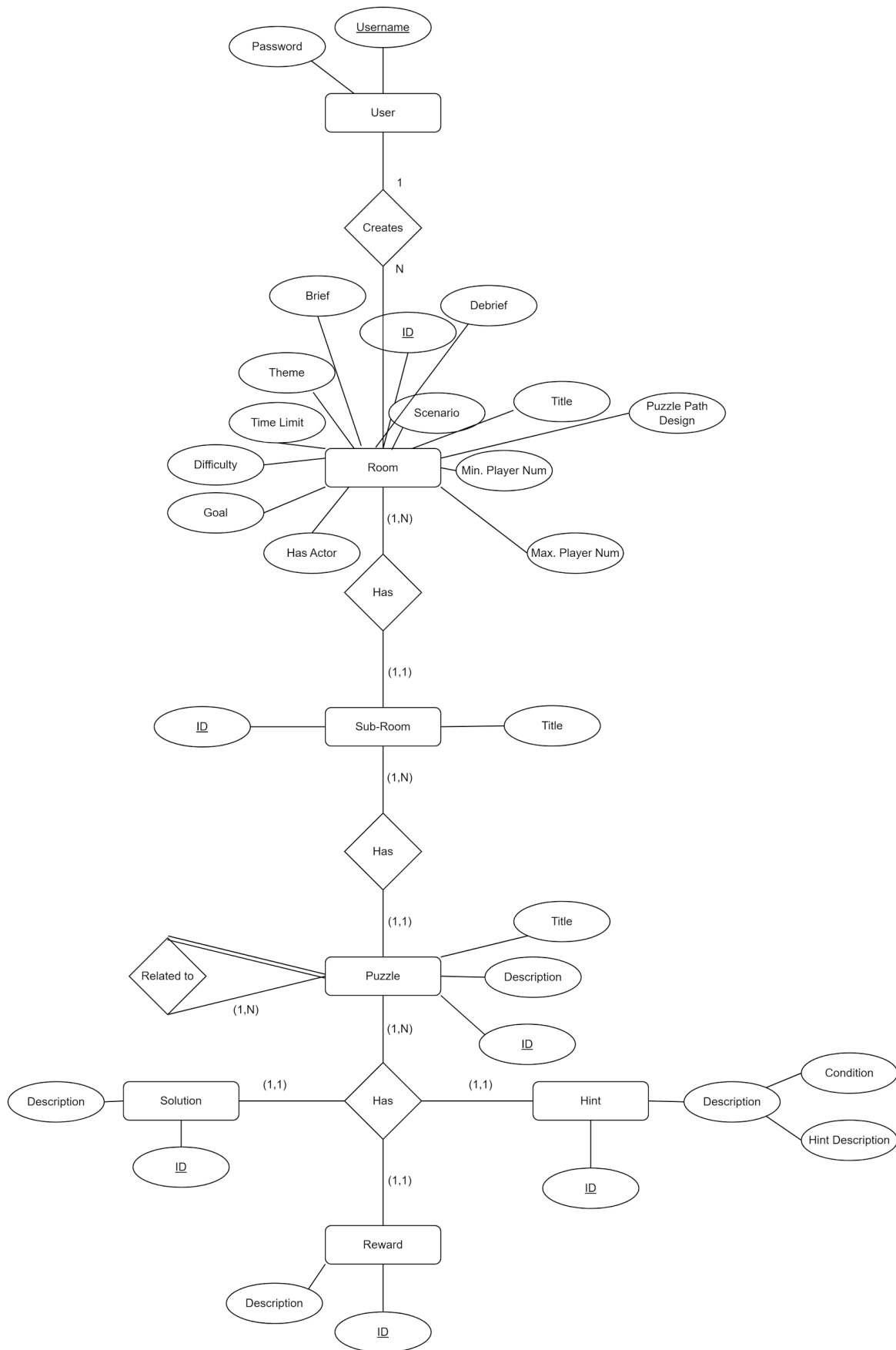
Στη λίστα Γ'.3, παρουσιάζονται αναλυτικά τα χαρακτηριστικά της κάθε οντότητας. Η οντότητα “χρήστης” έχει τα χαρακτηριστικά “όνομα χρήστη” και “κωδικός”. Η συγκεκριμένη οντότητα χρειάζεται μόνο για να εξυπηρετεί το σκοπό της σύνδεσης του χρήστη. Με αυτόν τον τρόπο, είναι δυνατό να αποθηκεύεται η δουλειά του σε περίπτωση που αποσυνδεθεί και κατ’ επέκταση να ανακληθεί από τη βάση όταν συνδέεται ξανά.

Η οντότητα “δωμάτιο” έχει τα περισσότερα χαρακτηριστικά και αντιπροσωπεύει ένα ολόκληρο escape room. Το χαρακτηριστικό “Μοναδικός χαρακτηριστικός αριθμός (ID)”, όπου υπάρχει, δίνει τη δυνατότητα διαχωρισμού των παραδειγμάτων της κάθε οντότητας. Στο χαρακτηριστικό “τίτλος” ο χρήστης μπορεί να ορίσει τον τίτλο που επιθυμεί να έχει το escape room του. Το χαρακτηριστικό αυτό επιτρέπει στο χρήστη τη χρήση αλφαριθμητικών χαρακτήρων. Το “Επιλογή σχεδιασμού συσχετίσεων παζλ (Puzzle Path Design)” δίνει στο χρήστη τη δυνατότητα να επιλέξει ένα από τρεις πιθανούς σχεδιασμούς συσχετίσεων παζλ. Αυτοί είναι ο γραμμικός σχεδιασμός, όπου κάθε παζλ σχετίζεται μόνο με ένα το οποίο είναι ένα βήμα πιο κοντά στο τελικό. Ο ανοιχτός σχεδιασμός, όπου το κάθε παζλ είναι αυτόνομο και η λύση ενός δεν εξαρτάται από κάποιο άλλο. Εξαίρεση αποτελεί το τελικό παζλ του δωματίου, το οποίο δεν μπορεί να λυθεί ωστόσο λυθούν όλα τα προηγούμενα. Τέλος, ο πολυγραμμικός σχεδιασμός. Αυτός ο σχεδιασμός προσφέρει τη μεγαλύτερη ελευθερία στη σειρά των παζλ. Αυτά μπορεί να είναι ουσιαστικά μια σειρά από γραμμικούς σχεδιασμούς. Η διαφορά εδώ είναι πως, αν και ο κάθε γραμμικός σχεδιασμός πρέπει να λυθεί σαν γραμμικός, αυτός δεν επηρεάζει τους άλλους. Έτσι οι παίκτες μπορούν να λύνουν ταυτόχρονα πολλούς από τους γραμμικούς σχεδιασμούς. Μπορεί επίσης τα μονοπάτια να διασταυρώνονται. Επιπλέον, ο σχεδιαστής μπορεί να κάνει φανερά όλα τα μονοπάτια από την αρχή ή να τα σχεδιάσει έτσι ώστε να εμφανίζονται όσο οι παίκτες προοδεύουν. Καθένας από αυτούς τους σχεδιασμούς έχει συγκεκριμένα χαρακτηριστικά. Ο γραμμικός σχεδιασμός είναι ευκολότερος να τον αντιληφθούν οι παίκτες. Επιπλέον είναι ευκολότερος να σχεδιαστεί. Υπάρχει όμως το ενδεχόμενο να μην υπάρχει δραστηριότητα για όλους τους παίκτες καθώς κάποιο παζλ, το οποίο είναι απαραίτητο να λυθεί, μπορεί να μην προσφέρει απασχόληση σε όλους τους παίκτες. Ο ανοιχτός σχεδιασμός λύνει αυτό το πρόβλημα καθώς επιτρέπει στους παίκτες να λύνουν πολλά παζλ ταυτόχρονα. Η έλλειψη ξεκάθαρα μονοπατιού όμως αυξάνει τη δυσκολία (Wiemker et al., 2015). Ο πολυγραμμικός σχεδιασμός φαίνεται να έχει τα χαρακτηριστικά του ανοιχτού σχεδιασμού σε μεγαλύτερο βαθμό καθώς προσφέρει πολλά μονοπάτια που μπορούν να λύνονται ταυτόχρονα αλλά φαίνεται ο πιο περίπλοκος ως προς τη σχεδίαση. Επιπλέον, μιας που υπάρχει το ενδεχόμενο τα μονοπάτια να περιπλέκονται

αλλά και να μην είναι φανερά από την αρχή, δίνει την εντύπωση πως είναι ο πιο δύσκολος ως προς την κατανόηση από τους παίκτες. Τα χαρακτηριστικά "Ελάχιστος αριθμός παικτών" και "Μέγιστος αριθμός παικτών" επιτρέπουν τη χρήση αριθμητικών χαρακτήρων. Ο χρήστης δεν μπορεί να επιλέξει τον αριθμό 0 σε αυτά τα χαρακτηριστικά. Για την "Παρουσία Ηθοποιού" υπάρχει ένα πλαίσιο το οποίο αν είναι επιλεγμένο επισημαίνει την παρουσία ηθοποιού και αν δεν είναι επιλεγμένο, την απουσία ηθοποιού. Παρόλο που ο "Στόχος" στα escape room είναι συνήθως η απόδραση από ένα δωμάτιο, είτε αυτό έχει κάποια "Θεματική" (π.χ. μπουντρούμι) είτε όχι, δεν είναι απαραίτητο να είναι αυτός. Ένα escape room μπορεί να έχει σαν στόχο τη λύση ενός μυστηρίου φώνου, την εξουδετέρωση μιας βόμβας, τη διερεύνηση ενός εγκλήματος και άλλα. (Nicholson, 2015). Έτσι τα δύο αυτά χαρακτηριστικά επιτρέπουν στο χρήστη τη χρήση αλφαριθμητικών χαρακτήρων ώστε αυτός να έχει την κατάλληλη ελευθερία. Σε μία έρευνα στην οποία συμμετείχαν 175 escape room απ' όλο τον κόσμο, η δυσκολία χωριζόταν σε τρία διακριτά επίπεδα: Εύκολο, μέτριο και δύσκολο (Nicholson, 2015). Η "Δυσκολία" λοιπόν, δίνει στο χρήστη τη δυνατότητα επιλογής μεταξύ αυτών των τριών επιλογών. Από την έρευνα, προκύπτει πως τα περισσότερα escape room έχουν σαν συνθήκη αποτυχίας ένα "Χρονικό Περιθώριο". Όταν αυτό παρέλθει και αν οι παίκτες δεν έχουν καταφέρει να φέρουν εις πέρας όλες τις δραστηριότητες, χάνουν το παιχνίδι. Το χαρακτηριστικό αυτό δέχεται αριθμητικές τιμές. Τέλος τα χαρακτηριστικά "Περίληψη του δωματίου (Brief)" που λαμβάνει χώρα πριν ξεκινήσει το παιχνίδι και "Απολογισμός (Debrief)" που λαμβάνει χώρα μετά το τέλος του παιχνιδιού. Η περίληψη του δωματίου προκύπτει από προσωπική εμπειρία του συγγραφέα. Σε αυτή ο gamemaster, παρουσιάζει στους παίκτες το δωμάτιο. Του αναφέρει τυχόν κανόνες, πράγματα που αναμένεται να συναντήσουν και πράγματα που δεν αναμένεται (π.χ. σκοτάδι), αν οι παίκτες θα πρέπει να έχουν κάποια συγκεκριμένη συμπεριφορά (π.χ. να μην αντισταθούν στις πράξεις του ηθοποιού) μεταξύ άλλων. Αν και η περίληψη πρέπει να περιέχει κάποιες συγκεκριμένες πληροφορίες, δύναται να έχει περισσότερο τη μορφή σεναρίου. Στον απολογισμό δίνεται στους παίκτες η ευκαιρία να συζητήσουν για το παιχνίδι που μόλις τελείωσε. Είναι επίσης, μια πολύ καλή ευκαιρία για τον gamemaster ή τους δημιουργούς να ακούσουν σκέψεις που μπορεί να έχουν οι παίκτες για το δωμάτιο που μόλις τελείωσαν. Η φύση αυτής της κουβέντας είναι περισσότερο ελεύθερη, έτσι ο απολογισμός δεν μπορεί να έχει τη μορφή σεναρίου. Μπορεί όμως να αποτελείται από ερωτήσεις που μπορεί να κάνει ο gamemaster προς τους παίκτες ή κάποια βασικά σημεία τα οποία θεωρούν οι δημιουργοί σημαντικό να συζητηθούν.

Οι υπόλοιπες οντότητες είναι απλούστερες και εισάγουν μόνο 2 νέα χαρακτηριστικά. Την "περιγραφή" και τη "συνθήκη". Η "περιγραφή" υπάρχει σε όλα τα υπόλοιπα χαρακτηριστικά εκτός του "υποδωμάτιο". Εκεί ο χρήστης μπορεί να περιγράψει τα απαραίτητα για την κάθε οντότητα. Το πεδίο δέχεται αλφαριθμητικούς χαρακτήρες. Η "συνθήκη" παρουσιάζεται μόνο στη "βοήθεια". Περιέχει περιγραφή της συνθήκης κατά την οποία ο gamemaster θα δώσει μια βοήθεια στους παίκτες. Αυτή μπορεί να είναι χρονική ή μπορεί να βασίζεται στον αριθμό αποτυχημένων προσπαθειών που έχουν κάνει οι παίκτες για να λύσουν ένα παζλ. Το "παζλ" έχει επίσης ένα μοναδικό χαρακτηριστικό. Αυτό είναι η αναδρομική σχέση που έχει. Αυτή επιτρέπει στο δημιουργό να κατασκευάσει έναν από τους σχεδιασμούς συσχετίσεων παζλ που αναφέρθηκαν παραπάνω. Η συσχέτιση αυτή έχει λόγο πληθικότητας ένα-προς-πολλά (1:N) καθώς κάθε παζλ

μπορεί να σχετίζεται μόνο με ακριβώς ένα που βρίσκεται πιο κοντά στη λύση από αυτό, αλλά το τελικό παζλ πρέπει να μπορεί να δέχεται συσχετίσεις από πολλά παζλ που βρίσκονται πριν από αυτό.



Καθώς η εφαρμογή έχει αναπτυχθεί με δύο διαφορετικά εργαλεία, το παραπάνω διάγραμμα είναι αρκετά γενικό. Στις αντίστοιχες ενότητες του κάθε εργαλείου θα δούμε αναλυτικότερα διαγράμματα με τυχόν διαφορές και διαμορφωμένα ονόματα μεταβλητών και τύπων μεταβλητών σωστά για το κάθε ένα από τα εργαλεία που χρησιμοποιήθηκαν για την ανάπτυξη της εφαρμογής. Από το διάγραμμα αυτό, μπορούμε να καταλάβουμε ποιες οντότητες, χαρακτηριστικά και συσχετίσεις περιέχει η βάση δεδομένων της εφαρμογής.

Γ'.6 Παρουσίαση σελίδων

Σε αυτό το κεφάλαιο, παρουσιάζονται αναλυτικότερα οι σελίδες που αναπτύχθηκαν με τα εργαλεία κώδικα και τα εργαλεία με λίγο ή καθόλου κώδικα. Η παρουσίαση θα γίνει μέσω εικόνων των σελίδων και περιγραφή του περιεχομένου της εικόνας αλλά και της λειτουργίας των αντικειμένων που παρουσιάζονται στην εικόνα. Οι σελίδες που αναπτύχθηκαν με διαφορετικά εργαλεία, ακόμα και αν είναι αντίστοιχες (π.χ. η αρχική σελίδα), ενδέχεται να διαφέρουν οπτικά και στην λειτουργία τους. Όπως θα δούμε και στο κεφάλαιο του πειράματος, το γεγονός αυτό είναι ένα μέρος στο οποίο τα εργαλεία θα συγκριθούν μεταξύ τους.

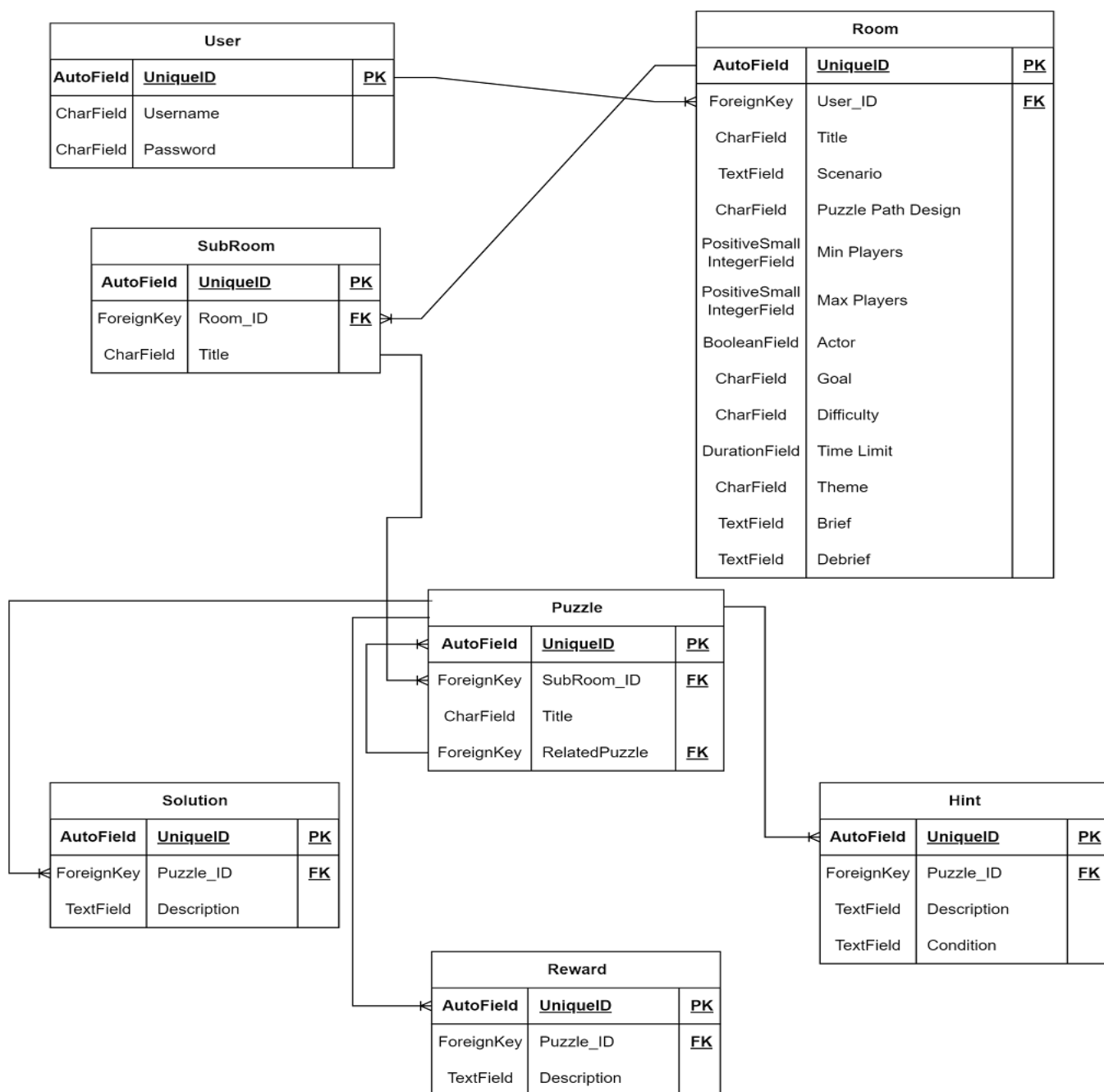
Γ'.6.1 Εργαλεία κώδικα

Το εργαλείο που επιλέχθηκε για την ανάπτυξη της εφαρμογής με κώδικα, είναι το Django. Το Django είναι ένα εργαλείο τύπου “framework”, όπως συνηθίζει να τα αποκαλεί η κοινότητα. Τα κριτήρια για την επιλογή του εργαλείου ήταν, η γλώσσα προγραμματισμού που χρησιμοποιείται με το εργαλείο, το πόσο φιλικό είναι το εργαλείο προς έναν, άπειρο στην ανάπτυξη διαδικτυακών εφαρμογών, χρήστη και το πόσο γρήγορα μπορεί το εργαλείο να επιτρέψει και να βοηθήσει το χρήστη να παράξει ένα “minimum viable product” (MVP). Σχετικά με την γλώσσα προγραμματισμού, το Django χρησιμοποιεί την Python. Η Python καλύπτει τα παραπάνω κριτήρια όντας μία από τις ευκολότερες γλώσσες για αρχάριους λόγω του απλού συντακτικού της και του υψηλού επιπέδου προγραμματισμού που προσφέρει. Είναι επίσης, η δεύτερη δημοφιλέστερη γλώσσα προγραμματισμού, πίσω μόνο από την JavaScript (*Stack Overflow Developer Survey 2022*, n.d.), με αποτέλεσμα την ύπαρξη άφθονου υλικού σε forum και μιας μεγάλης κοινότητας, γεγονότα που οδηγούν στην εύκολη εύρεση λύσης ή βοήθειας στο διαδίκτυο. Σχετικά με το Django, καλύπτει τα παραπάνω κριτήρια προσφέροντας αρκετές λειτουργίες μέσω κλάσεων οι οποίες είναι διαθέσιμες στον προγραμματιστή. Έτσι δημιουργείται μια αφαίρεση, από τον προγραμματιστή, των χαμηλού επιπέδου λειτουργιών οδηγώντας σε γρηγορότερους χρόνους ανάπτυξης, κώδικα που δεν επαναλαμβάνεται και είναι απλός στην κατανόηση. Την ίδια αφαίρεση προσφέρει το Django και για την βάση δεδομένων, επιτρέποντας στο χρήστη να γράφει Python με συγκεκριμένη σύνταξη ώστε να κάνει οποιεσδήποτε ενέργειες θέλει στην βάση δεδομένων, έτσι δεν υπάρχει πλέον η απαίτηση από τον προγραμματιστή να γνωρίζει κάποιο πρόγραμμα διαχείρισης βάσεων δεδομένων (RDBMS) ή ακόμα και SQL. Στο ίδιο πνεύμα το Django, με χρήση ειδικών αρχείων και συντακτικού, προσφέρει και ομαλή σύνδεση της βάσης δεδομένων με τη γραφική διεπαφή (frontend) κερδίζοντας έτσι στον προγραμματιστή, χρόνο, αλλά και κάνοντας την εφαρμογή περισσότερη σταθερή, με λιγότερα λογικά σφάλματα στον κώδικα. Ταυτόχρονα, όμως, το Django,

προσφέρει και την δυνατότητα παραμετροποίησης των λειτουργιών που ήδη παρέχει δίνοντας έτσι στο σχεδιαστή και δημιουργό της εφαρμογής πραγματική ελευθερία στον τρόπο που θα υλοποιήσει την εφαρμογή του. Ένα παράδειγμα της, προαιρετικής λοιπόν, αφαίρεσης που προσφέρει το Django είναι η μηχανή (engine) που θα χρησιμοποιηθεί στην εφαρμογή. Η προεπιλεγμένη μηχανή που χρησιμοποιεί το Django είναι η SQLite, αν ο δημιουργός το επιλέξει, το μόνο που έχει να κάνει, είναι να γράψει τον ειδικό κώδικα σε Python για την δημιουργία και επεξεργασία της βάσης. Αν όμως θέλει να επιλέξει κάποια άλλη μηχανή, για οποιονδήποτε λόγο, το μόνο που έχει να κάνει είναι να αλλάξει μερικές γραμμές κώδικα που λένε στο Django ποια μηχανή να χρησιμοποιήσει. Τέλος, αξίζει να αναφερθεί πως το Django παρέχει αναλυτικό περιεχόμενο τεκμηρίωσης των λειτουργιών που παρέχει και είναι εργαλείο ανοιχτού κώδικα.

Γ'6.1.1 Βάση δεδομένων

Στο κεφάλαιο Γ'.5 παρουσιάζεται η γενική σχεδίαση της βάσης δεδομένων. Σε αυτό το υποκεφάλαιο παρουσιάζεται συγκεκριμένα η βάση δεδομένων που σχεδιάστηκε και αναπτύχθηκε με το εργαλείο κώδικα. Αυτό είναι απαραίτητο καθώς στο κεφάλαιο Γ'.5 δεν υπάρχουν συγκεκριμένες λεπτομέρειες για τη βάση όπως παραδείγματος χάριν, τύποι δεδομένων.



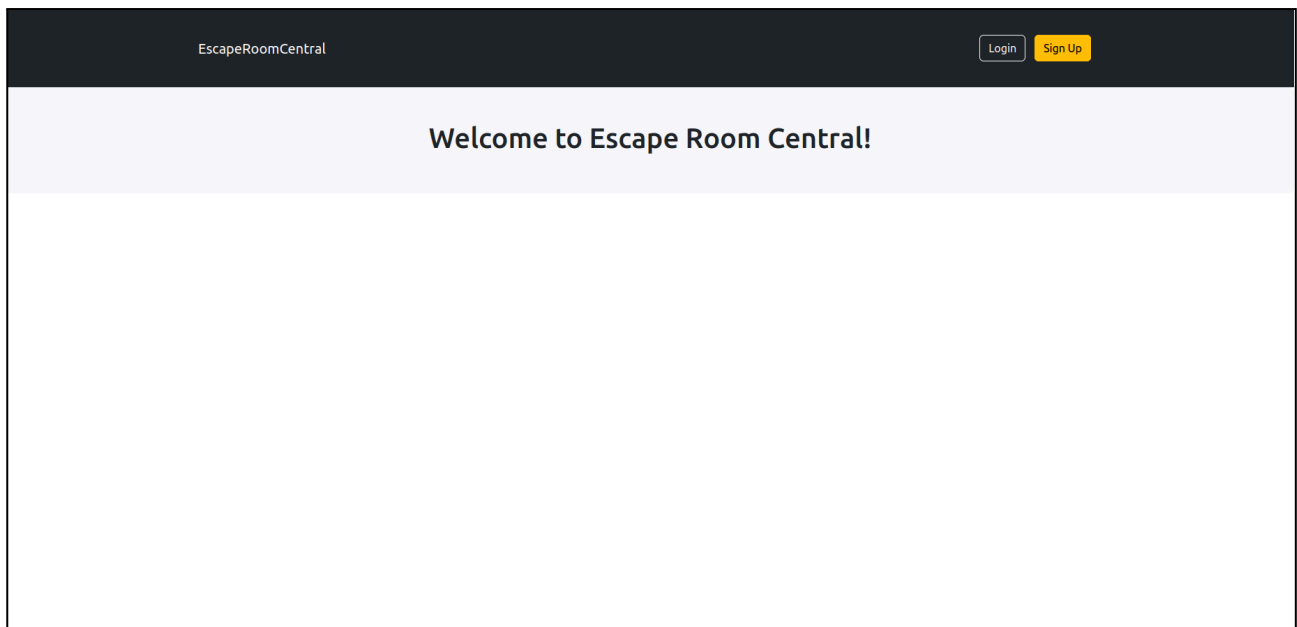
Εικόνα Γ'.2: Αναλυτικό διάγραμμα οντοτήτων συσχετίσεων

Η εικόνα Γ'.2 είναι η αναλυτική αναπαράσταση της βάσης δεδομένων που σχεδιάστηκε και αναπτύχθηκε για τη σελίδα με το εργαλείο κώδικα. Έτσι, οι τύποι των μεταβλητών που βλέπουμε είναι τύποι μεταβλητών του Django. Αυτοί χρησιμοποιούνται κατά τη συγγραφή του κώδικα που είναι υπεύθυνος για τη δημιουργία της βάσης δεδομένων. Ένα κοινό χαρακτηριστικό όλων των οντοτήτων είναι το UniqueID με τύπο AutoField. Αυτό είναι και το πρωτεύον κλειδί των οντοτήτων. Είναι η προκαθορισμένη λύση του Django για τη διαχείριση των κλειδιών. Σε περίπτωση που ο δημιουργός θέλει να ορίσει κάποιο διαφορετικό χαρακτηριστικό σαν κλειδί, τότε μπορεί να το κάνει δηλώνοντάς το ρητά. Αντίστοιχα η επιλογή αυτού του χαρακτηριστικού να είναι τύπου AutoField είναι η προκαθορισμένη του Django. Σύμφωνα με το documentation του Django το AutoField είναι ίδιο με το IntegerField αλλά αυξάνεται αυτόματα ανάλογα με τα διαθέσιμα ID. Άλλο ένα γνώρισμα που είναι απαραίτητο για τη λειτουργία της βάσης είναι το ξένο κλειδί ή όπως το λέει το Django, το ForeignKey. Με αυτό ο δημιουργός μπορεί να συσχετίσει δύο οντότητες. Ο

συγκεκριμένος τύπος έχει δύο παραμέτρους που πρέπει υποχρεωτικά να δηλωθούν. Αυτές είναι η κλάση η οποία αντιπροσωπεύει την οντότητα με την οποία υπάρχει η συσχέτιση και η διαχείριση των πλειάδων σε περίπτωση διαγραφής της συσχετιζόμενης τους. Η επιλογή των τύπων για την κάθε οντότητα έγινε με σκοπό τη δέσμευση του λιγότερου δυνατού χώρου αλλά ταυτόχρονα και την εξυπηρέτηση του σκοπού των χαρακτηριστικών. Αξίζει επίσης να αναφερθεί η επιλογή του DurationField για το χαρακτηριστικό Time Limit. Αρχικά, είχε επιλεγθεί ο τύπος TimeField ο οποίος όμως δεν εξυπηρετούσε το σκοπό του χαρακτηριστικού που είναι να επιτρέψει στον τελικό χρήστη να ορίσει τη μέγιστη διάρκεια που θέλει να έχει το δωμάτιό του. Έτσι έγινε αλλαγή στον DurationField όπου δίνει αυτή τη δυνατότητα.

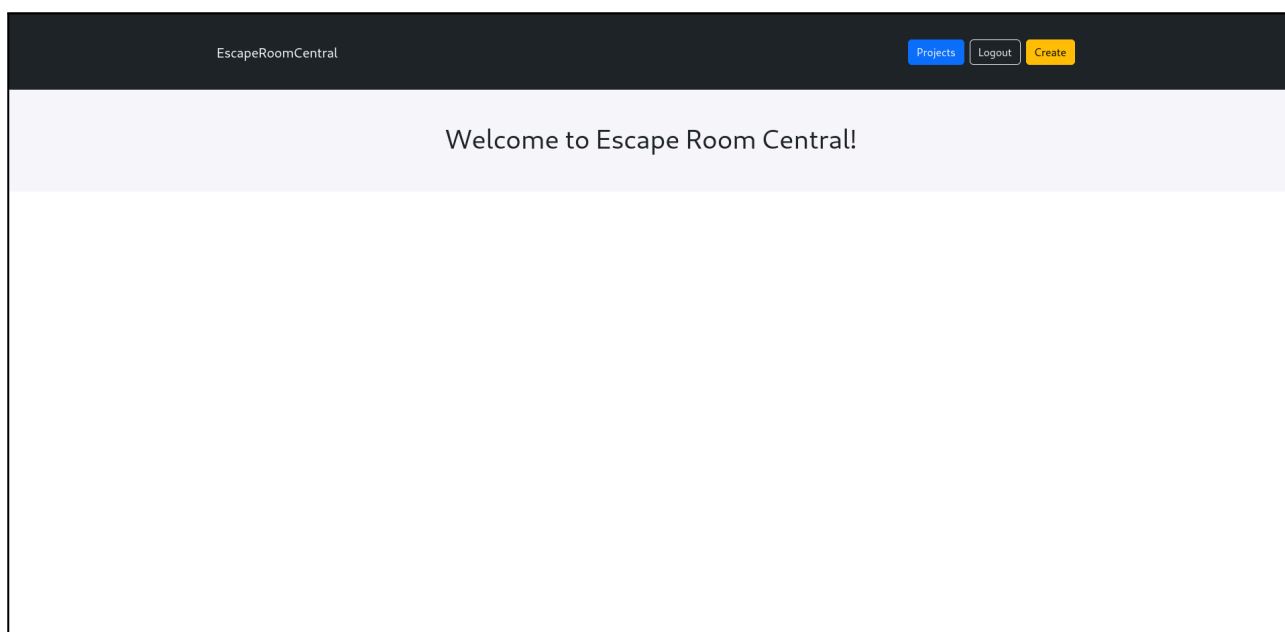
Γ'.6.1.2 Αρχική σελίδα

Η αρχική σελίδα είναι η πρώτη που συναντά ο χρήστης, συνδεδεμένος ή μη, όταν επισκέπτεται την εφαρμογή. Από εκεί ο χρήστης μπορεί να επιλέξει να συνδεθεί αν είναι εγγεγραμμένος αλλά όχι συνδεδεμένος, να εγγραφεί αν δεν είναι εγγεγραμμένος, να αποσυνδεθεί αν είναι συνδεδεμένος, να μεταβεί στην σελίδα του προφίλ του αν είναι συνδεδεμένος ή και να δημιουργήσει ένα δωμάτιο.



Εικόνα Γ'.3: Αρχική σελίδα αποσυνδεδεμένου χρήστη

Όπως φαίνεται στην εικόνα Γ'.3 από την αρχική σελίδα, ένας αποσυνδεδεμένος χρήστης, μπορεί είτε να εγγραφεί, πατώντας το κίτρινο κουμπί με επιγραφή “Sign Up”, είτε να συνδεθεί, αν είναι ήδη εγγεγραμμένος, πατώντας το διαφανές κουμπί με επιγραφή “Login”. Το κουμπί της σύνδεσης αποκτά λευκό φόντο και μαύρα γράμματα όταν ο χρήστης περνά τον κέρσορα από πάνω του.



Εικόνα Γ'.4: Αρχική σελίδα συνδεδεμένου χρήστη

Στην εικόνα Γ'.4 απεικονίζεται η αρχική σελίδα ενός συνδεδεμένου χρήστη. Οι διαφορές σε σχέση με την αρχική σελίδα του αποσυνδεδεμένου χρήστη υπάρχουν στο πάνω και δεξιά μέρος της σελίδας, όπου πλέον τα κουμπιά αντιστοιχούν σε ενέργειες που έχει την δυνατότητα να πράξει ένας συνδεδεμένος χρήστης. Με το μπλε κουμπί με επιγραφή “Projects” ο χρήστης μπορεί να μεταβεί στην σελίδα προφίλ του, όπου βρίσκονται τα προσωπικά του δωμάτια. Αυτό το κουμπί σκουραίνει το φόντο του όταν ο χρήστης περνά τον κέρσορα από πάνω του. Με το διαφανές κουμπί με επιγραφή “Logout” ο χρήστης μπορεί να αποσυνδεθεί οποτεδήποτε εκείνος επιθυμεί. Αυτό το κουμπί αλλάζει το φόντο του σε λευκό και τα γράμματα σε μαύρο χρώμα όταν ο χρήστης περνά τον κέρσορα από πάνω του. Με το κίτρινο κουμπί με επιγραφή “Create” ο χρήστης μπορεί να μεταβεί στην σελίδα δημιουργίας δωματίου αν επιθυμεί να δημιουργήσει νέο δωμάτιο. Τέλος, αξίζει να σημειωθεί πως οποιαδήποτε στιγμή ο χρήστης πατήσει την επωνυμία “EscapeRoomCentral” μεταβαίνει στην αρχική σελίδα όπου και αν βρίσκεται.

Γ'.6.1.3 Εγγραφή

Στην σελίδα εγγραφής υπάρχει μια φόρμα, την οποία θα πρέπει να συμπληρώσει κάποιος χρήστης σε περίπτωση που θέλει να εγγραφεί και να δημιουργήσει λογαριασμό, ώστε να μπορούν να αποθηκευτούν τα δωμάτια τα οποία δημιουργεί.

Εικόνα Γ'.5: Σελίδα εγγραφής

Σε αυτή την φόρμα λοιπόν, ο χρήστης θα πρέπει να συμπληρώσει, το όνομα με το οποίο θα συνδέεται (Display Name) και τον κωδικό (Password) που επιθυμεί να χρησιμοποιεί για να συνδέεται. Τον κωδικό ο χρήστης θα πρέπει να τον συμπληρώσει δύο φορές για λόγους ασφαλείας και αποφυγή εισαγωγής μη επιθυμητού κωδικού.

Εικόνα Γ'.6: Σελίδα εγγραφής με λάθη

Σε περίπτωση που κάποια από τα στοιχεία που εισήγαγε ο χρήστης δεν είναι αποδεκτά, η φόρμα το κάνει ξεκάθαρο εμφανίζοντας μηνύματα λάθους με κόκκινα γράμματα στο αντίστοιχο πεδίο, όπως φαίνεται στην εικόνα Γ'.5. Για τη μείωση των λαθών στα στοιχεία που εισάγει ο χρήστης, η φόρμα έχει εμφανείς οδηγίες για το πως πρέπει να συμπληρωθεί το κάθε πεδίο σύμφωνα με τυχόν περιορισμούς που μπορεί να έχει. Είναι σημαντικό να αναφερθεί πως, η λειτουργία ελέγχου και εμφάνισης των πιθανών λαθών στα στοιχεία που

εισήγαγε ο χρήστης, προσφέρεται από το Django σε υψηλό επίπεδο. Ο δημιουργός δεν έχει παρά να φτιάξει μια ειδική φόρμα Django. Το ίδιο ισχύει και για της οδηγίες που εμφανίζει η φόρμα για το κάθε πεδίο και κατ' επέκταση, ο έλεγχος τήρησης αυτών των οδηγιών. Το γεγονός αυτό προσφέρει, πολύ εύκολα, ένα κατώφλι ασφάλειας σε ό,τι αφορά τους λογαριασμούς των χρηστών αλλά και γλυτώνει χρόνο στο δημιουργό.

```
4 class UserRegisterForm(UserCreationForm):
5     class Meta:
6         model = get_user_model()
7         fields = ['username', 'password1', 'password2']
8
9     def __init__(self, *args, **kwargs):
10         super().__init__(*args, **kwargs)
11         self.fields['username'].label='Display name'
```

Εικόνα Γ'.7: Κώδικας φόρμας δημιουργίας χρήστη

Στην εικόνα Β'.6 βλέπουμε τον κώδικα που πρέπει να γράψει ο δημιουργός για να φτιάξει τη λειτουργία της φόρμας σύνδεσης. Βλέπουμε πως στη γραμμή 7 ορίζονται τα πεδία της φόρμας τα οποία θα πρέπει να συμπληρώσει ο χρήστης. Στη γραμμή 9 ξεκινάει μια συνάρτηση που, στην προκειμένη περίπτωση, επιτρέπει στο δημιουργό να παραμετροποιήσει τον τίτλο κάθε πεδίου της φόρμας.

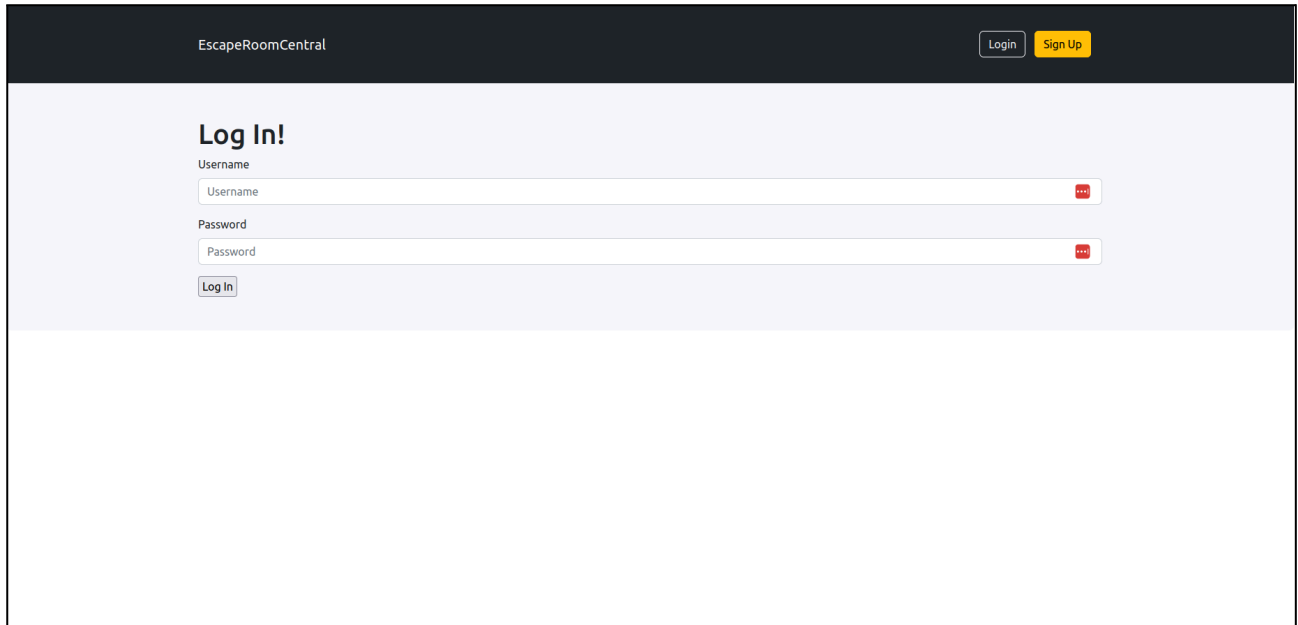
```
1 {% extends 'base.html' %}
2 {% load django_bootstrap5 %}
3
4 {% block content %}
5 <div class="container">
6     <h1>Sign Up!</h1>
7     <form method='POST'>
8         {% csrf_token %}
9         {% bootstrap_form form %}
10        <input type="submit" value="Sign Up">
11    </form>
12 </div>
13 {% endblock content %}
```

Εικόνα Γ'.8: Κώδικας για τη δημιουργία της σελίδας της φόρμας εγγραφής

Η εικόνα Γ'.7 μας δείχνει τον κώδικα που χρειάζεται για να δημιουργηθεί η σελίδα που φαίνεται στις εικόνες Β'4 και Β'5. Τα κομμάτια που βρίσκονται ανάμεσα στα σύμβολα {% και %} είναι κομμάτια του Django template language. Μέσω αυτού μπορεί κανείς να δημιουργήσει templates που ισοδυναμούν περίπου με προγραμματισμό front end. Έτσι η πρώτη γραμμή συμπεριλαμβάνει σε αυτό το template τα περιεχόμενα του base.html. Στο base.html υπάρχει ο κώδικας για τη δημιουργία του header. Επειδή ο header υπάρχει σε όλες τις σελίδες, η συγκεκριμένη λειτουργία είναι πολύ χρήσιμη για να εξοικονομείται κώδικας. Η γραμμή 8 υπάρχει καθώς προσφέρει ασφάλεια στη φόρμα.

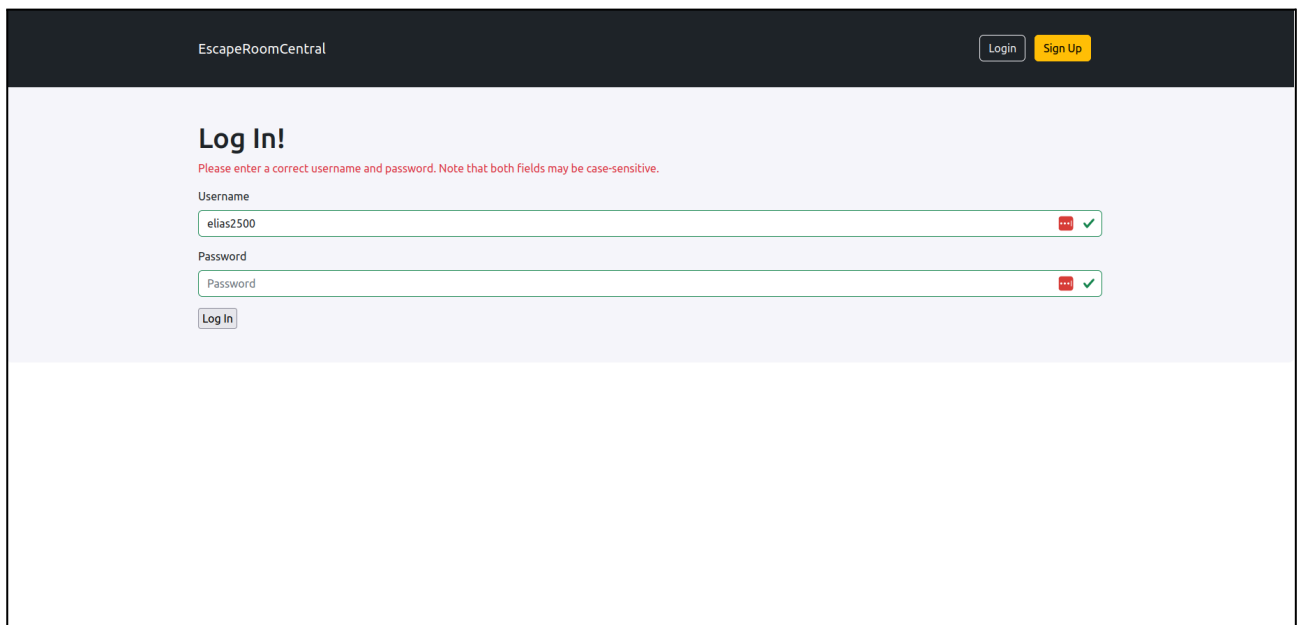
Γ'.6.1.4 Σύνδεση

Από την σελίδα σύνδεσης και συμπληρώνοντας την αντίστοιχη φόρμα, ο χρήστης μπορεί να συνδεθεί σε λογαριασμό που έχει δημιουργήσει στο παρελθόν, με σκοπό να μπορεί να ανατρέξει σε παλαιότερη πληροφορία του.



Εικόνα Γ'.9: Σελίδα σύνδεσης

Σε αυτή την περίπτωση η φόρμα που θα πρέπει να συμπληρώσει ο χρήστης, είναι πιο απλή. Το μόνο που απαιτείται είναι το όνομα χρήστη και ο κωδικός που επέλεξε κατά την διαδικασία της εγγραφής.



Εικόνα Γ'.10: Σελίδα σύνδεσης με λάθη

Σε περίπτωση που ο χρήστης δεν δώσει τα σωστά στοιχεία σύνδεσης, η φόρμα τον ενημερώνει με μήνυμα λάθους όπως στην εικόνα Γ'.10. Η λειτουργία επαλήθευσης των στοιχείων με την βάση δεδομένων προσφέρεται και πάλι από το Django με αφαιρετικό, προαιρετικό, προς το δημιουργό, τρόπο.

Για τη φόρμα σύνδεσης δεν υπάρχει ο αντίστοιχος κώδικας με τη φόρμα εγγραφής που καθορίζει τη λειτουργία της φόρμας. Αυτό συμβαίνει γιατί το Django προσφέρει αυτή τη λειτουργία έτοιμη. Έτσι, όπως φαίνεται στην εικόνα Γ'.11 το μόνο που έχει να κάνει ο δημιουργός είναι να συμπεριλάβει αυτή τη λειτουργία στον ορισμό του URL.

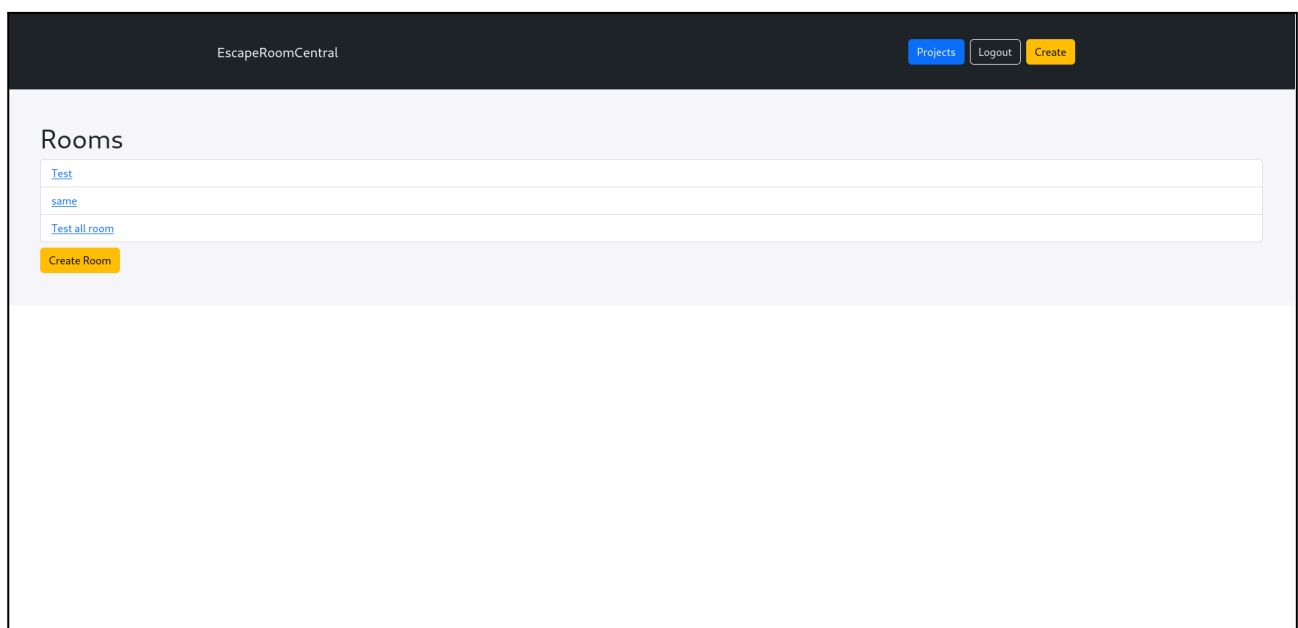
```
1 from django.urls import path
2 from . import views
3 from django.contrib.auth import views as auth_views
4
5 app_name = 'users'
6
7 urlpatterns = [
8     path('login/', auth_views.LoginView.as_view(template_name='users/login.html', next_page='/'), name='login'),
9     path('logout/', auth_views.LogoutView.as_view(next_page='/'), name='logout'),
10    path('register/', views.RegisterView.as_view(), name='register')
11 ]
```

Εικόνα Γ'.11: Αρχείο με ορισμούς URL.

Αυτό που ουσιαστικά υλοποιείται εδώ είναι η χαρτογράφηση των URL. Ο δημιουργός δηλαδή ορίζει ποια σελίδα θα φαίνεται κάτω από κάθε συγκεκριμένο URL. Στην περίπτωση της σελίδας με τη φόρμα σύνδεσης υπάρχει το κομμάτι `auth_views.LoginView` ή `auth_views.LogoutView`. Το `django.contrib.auth.views` προσφέρει, μεταξύ άλλων, ένα σύνολο κλάσεων που έχουν να κάνουν με την ταυτοποίηση. Μπορούμε στο ίδιο αρχείο στη γραμμή 10 να δούμε ένα παράδειγμα όπου ο δημιουργός χρησιμοποιεί ένα view το οποίο έχει φτιάξει ο ίδιος. Σε αυτή τη γραμμή το `RegisterView` έρχεται από το αρχείο `views.py` όπου μπορεί ο δημιουργός να ορίσει τα δικά του views. Τα views στο Django είναι κάποια κλητά στοιχεία τα οποία είναι υπεύθυνα για τη διαχείριση των αιτημάτων. Με ειδικό συντακτικό λοιπόν, προγραμματίζονται να επιστρέφουν μια απάντηση όταν δεχτούν ένα συγκεκριμένο αίτημα.

Γ'.6.1.5 Σελίδα προβολής λίστας δωματίων

Η σελίδα προβολής λίστας του χρήστη περιέχει τη λίστα από τα δωμάτια που έχει δημιουργήσει. Από αυτή, μπορεί να επιλέξει οποιοδήποτε επιθυμεί ώστε να κάνει τις δυνατές ενέργειες σε αυτό, ή να δημιουργήσει ένα καινούργιο.



Στη σελίδα αυτή, ο χρήστης βλέπει την λίστα των δωματίων του, όπως απεικονίζεται στην εικόνα Γ'.12. Πατώντας ένα από τα ονόματα των δωματίων, μπορεί να μεταβεί στη σελίδα που του δείχνει με λεπτομέρεια τις παραμέτρους του δωματίου και πώς τις έχει ορίσει. Πατώντας το κίτρινο κουμπί με επιγραφή “Create Room” ο χρήστης, μπορεί να μεταβεί στην σελίδα που του επιτρέπει να δημιουργήσει νέο δωμάτιο, συμπληρώνοντας την κατάλληλη φόρμα.

Για να δημιουργηθεί η παραπάνω σελίδα χρησιμοποιείται ο κώδικας που φαίνεται στην εικόνα Γ'.13. Σε αυτή βλέπουμε ένα Django view, συγκεκριμένα ένα view που κληρονομεί από το `generic.ListView`. Αυτό είναι που προσφέρει τη βασική λειτουργία της δημιουργίας της λίστας, μερικές παραμετροποιήσεις είναι όμως απαραίτητες. Αυτές γίνονται από μια σειρά συναρτήσεων όπου η κάθε μια είναι υπεύθυνη για να παραμετροποιεί ένα συγκεκριμένο κομμάτι της λίστας. Η πρώτη στη γραμμή 47 είναι υπεύθυνη για να ορίσει το `queryset`. Δηλαδή δημιουργήσει ένα ερώτημα προς τη βάση με συγκεκριμένες παραμέτρους με σκοπό να πάρει μια συγκεκριμένη απάντηση. Σε αυτή την περίπτωση ορίζει ποιά δωμάτια θα παρουσιάζονται στη λίστα. Αυτό είναι απαραίτητο καθώς κάθε χρήστης πρέπει να μπορεί να δει μόνο τα δικά του δωμάτια. Στη γραμμή 57 ξεκινά μια συνάρτηση που επιτρέπει τη χρήση μεταβλητών στο αντίστοιχο αρχείο template.

```

44 class RoomListView(LoginRequiredMixin, UserPassesTestMixin, generic.ListView):
45     model = models.Room
46
47     def get_queryset(self):
48         try:
49             self.room_user = User.objects.prefetch_related('rooms').get(
50                 username__iexact=self.kwargs.get('username')
51             )
52         except User.DoesNotExist:
53             raise Http404
54         else:
55             return self.room_user.rooms.all()
56
57     def get_context_data(self, **kwargs):
58         context = super().get_context_data(**kwargs)
59         context["room_user"] = self.room_user
60         return context
61
62     def test_func(self):
63         return self.request.user.username == self.kwargs.get('username')

```

Εικόνα Γ'.13: Django View για τη λίστα των δωματίων

Γ'.6.1.6 Δημιουργία δωματίου

Από αυτή τη σελίδα ο χρήστης μπορεί να δημιουργήσει ένα νέο δωμάτιο μέσω της συμπλήρωσης της αντίστοιχης φόρμας.

EscapeRoomCentral

Projects Logout Create

Create a new Room!

Title

Title

Scenario

Scenario

PuzzlePathDesign

MinPlayers

MinPlayers

MaxPlayers

MaxPlayers

HasActor

No

Εικόνα Γ'.14.α: Σελίδα δημιουργίας δωματίου

Goal

Goal

Difficulty

TimeLimit

TimeLimit

Theme

Theme

Brief

Brief

Debrief

Debrief

Εικόνα Γ'14.β: Σελίδα δημιουργίας δωματίου συνέχεια

Εικόνα Γ'.14: Σελίδα δημιουργίας δωματίου

Οι παράμετροι που έχει την δυνατότητα να μεταβάλει ο χρήστης είναι, ο τίτλος του δωματίου, το σενάριο, το puzzle path design, ο μέγιστος και ελάχιστος αριθμός παικτών, η δυσκολία του δωματίου, αν το δωμάτιο θα έχει ηθοποιό ή όχι, το σενάριο, ο στόχος, το χρονικό περιθώριο και τέλος τα brief και debrief. Στο τέλος της σελίδα υπάρχει το κουμπί με επιγραφή "Create Room" το οποίο μπορεί να πατήσει ο χρήστης για να αποθηκευτεί το δωμάτιό του. Εναλλακτικά ο χρήστης, μπορεί οποιαδήποτε στιγμή να πατήσει το κουμπί "Enter" στο πληκτρολόγιο του ή το κουμπί "Create Room" για να αποθηκεύσει το επίσης το δωμάτιό του χωρίς να έχει συμπληρώσει όλα τα πεδία. Σε περίπτωση που ο χρήστης τελικά δεν επιθυμεί να

δημιουργήσει δωμάτιο δεν έχει παρά μόνο να μεταβεί στην προηγούμενη σελίδα χρησιμοποιώντας την αντίστοιχη λειτουργία του browser του.

Το δωμάτιο είναι η μεγαλύτερη και πιο πολύπλοκη οντότητα και αποτελείται από τα χαρακτηριστικά που αναφέρονται παραπάνω. Αναλυτικότερα, στην εικόνα Γ'.14 φαίνεται η κλάση που δημιουργεί αυτή την οντότητα στη βάση, στο αρχείο models.py του Django που εξυπηρετεί αυτό το σκοπό.

```
25 class Room(models.Model):
26     userId = models.ForeignKey(User, on_delete=models.CASCADE, related_name='rooms')
27     title = models.CharField(max_length=50, unique=True, null=True)
28     scenario = models.TextField(blank=True, null=True)
29     puzzlePathDesign = models.CharField(max_length=50, blank=True, choices=PUZZLE_CHOICES_LIST, null=True)
30     minPlayers = models.PositiveSmallIntegerField(blank=True, validators = [MinValueValidator(1)], null=True)
31     maxPlayers = models.PositiveSmallIntegerField(blank=True, null=True) #, validators = [MinValueValidator(int(minPlayers))]
32     hasActor = models.BooleanField(default=False, null=True)
33     goal = models.CharField(max_length=50, blank=True, null=True)
34     difficulty = models.CharField(blank=True, max_length=50, choices=DIFFICULTY_CHOICES_LIST, null=True)
35     #Changed timeLimit from a TimeField to a DurationField
36     timeLimit = models.DurationField(blank=True, null=True)
37     theme = models.CharField(blank=True, max_length=50, null=True)
38     brief = models.TextField(blank=True, null=True)
39     debrief = models.TextField(blank=True, null=True)
```

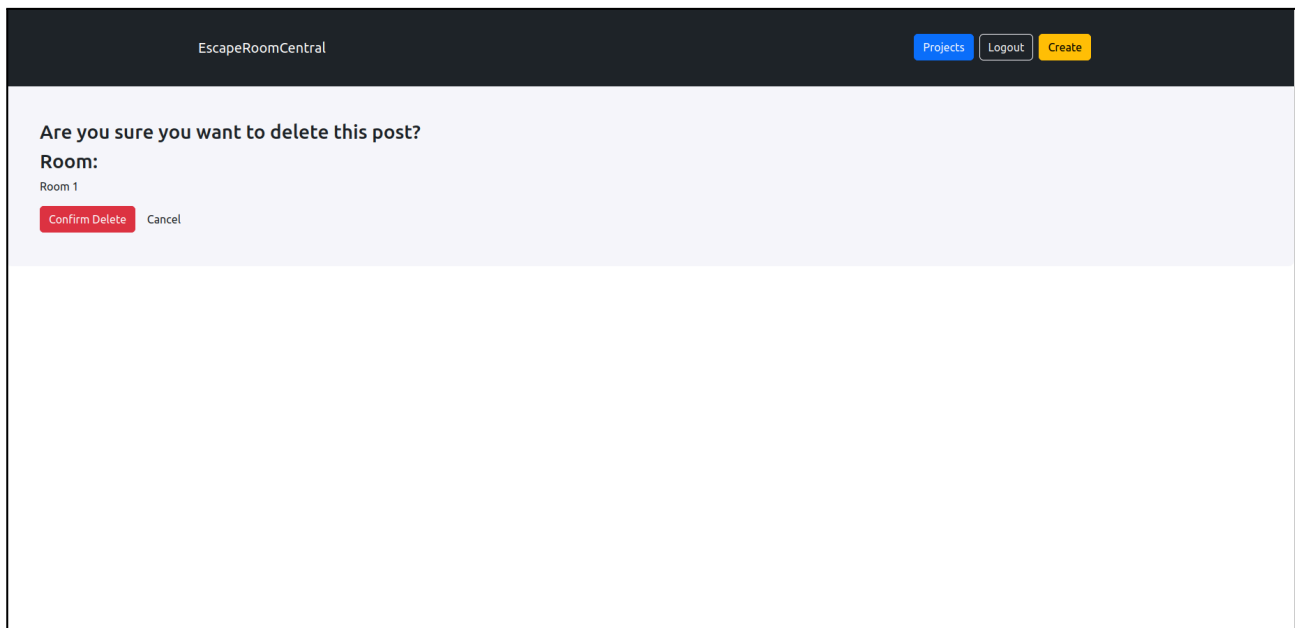
Εικόνα Γ'.15: Κώδικας οντότητας δωματίου

Στο αρχείο models.py του Django ορίζονται η οντότητες που πρέπει να δημιουργηθούν χρησιμοποιώντας python. Το Django αναλαμβάνει την επικοινωνία με τη βάση δεδομένων και τη μετάφραση της python σε μια γλώσσα διαχείρισης βάσεων δεδομένων, όπως η SQL. Η κλάση αντιπροσωπεύει μια οντότητα της βάσης και οι μεταβλητές της κλάσεις αντιπροσωπεύουν τα χαρακτηριστικά της οντότητας. Για να οριστεί ένα χαρακτηριστικό, ορίζεται μια μεταβλητή συγκεκριμένου τύπου. Οι τύποι αυτοί προσφέρονται από το Django και είναι παραμετροποιήσιμοι. Έτσι το χαρακτηριστικό title έχει τον τύπο CharField. Ένα πεδίο, δηλαδή, που δέχεται μια ακολουθία αλφαριθμητικών χαρακτήρων. Μερικές δυνατές παραμετροποιήσεις είναι το μέγιστο επιτρεπτό μέγεθος της ακολουθίας αλφαριθμητικών, το αν επιτρέπονται διπλότυπα ή όχι και το αν οι κενές τιμές θα αποθηκεύονται στη βάση ως NULL αντί για κενή ακολουθία. Στη γραμμή 29 φαίνεται ο τρόπος με το οποίο δημιουργήθηκε το μενού επιλογών για το puzzle path design. Το PUZZLE_CHOICES_LIST είναι μια λίστα python που τα περιεχόμενα της είναι python tuples και είναι ουσιαστικά οι επιλογές που δίνονται στη δημιουργό του δωματίου. Στο χαρακτηριστικό minPlayers υπάρχει ένας validator που ελέγχει την εγκυρότητα της τιμής που εισήγαγε ο δημιουργός.

Γ'.6.1.7 Διαγραφή δωματίου

Η σελίδα διαγραφής είναι μια σημαντική δικλείδα ασφάλειας που αποτρέπει στο χρήστη το ενδεχόμενο να διαγράψει κάποιο δωμάτιο ενώ δεν το επιθυμεί. Έτσι η σελίδα αυτή είναι η ερώτηση για την

επιβεβαίωση της διαγραφής ενός δωματίου.



Εικόνα Γ'.16: Σελίδα διαγραφής δωματίου

Στο χρήστη παρουσιάζεται η ερώτηση επιβεβαίωσης διαγραφής του δωματίου, συμπεριλαμβανομένου του τίτλου του δωματίου που πρόκειται να διαγράψει. Το κόκκινο κουμπί με επιγραφή "Confirm Delete" επιβεβαιώνει την διαγραφή και διαγράφει το δωμάτιο οριστικά. Μετά την διαγραφή του δωματίου ο χρήστης μεταφέρεται πίσω στη σελίδα προβολής δωματίων του, όπου μπορεί να δει τη νέα λίστα δωματίων. Αν ο χρήστης επιλέξει την επιγραφή "Cancel" τότε το δωμάτιο δεν διαγράφεται και ο χρήστης επιστρέφει στη σελίδα προβολής του συγκεκριμένου δωματίου. Η οντότητα δωμάτιο είναι ένα υπερσύνολο των υπόλοιπων οντοτήτων. Έτσι όταν ένα δωμάτιο διαγράφεται, πρέπει να διαγραφούν και όλες οι οντότητες που σχετίζονται με αυτό. Για να επιτευχθεί αυτό, πρέπει να οριστεί η αντίστοιχη παράμετρος στην κάθε οντότητα.

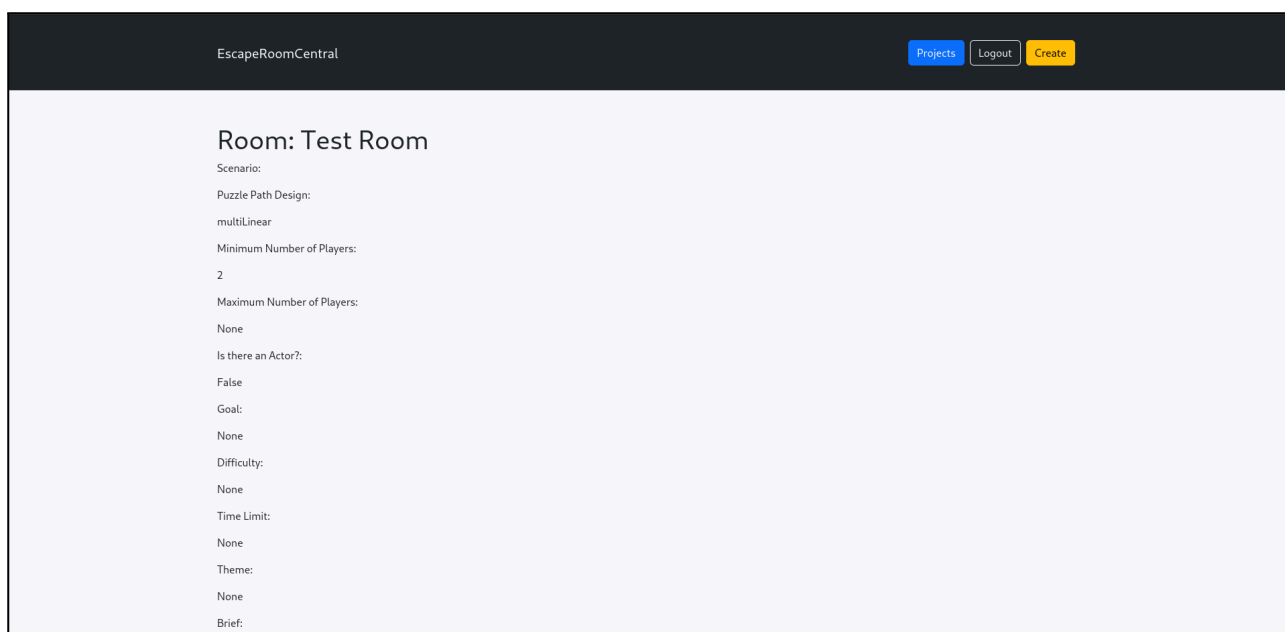
```
roomId = models.ForeignKey(Room, on_delete=models.CASCADE, related_name='subRooms')
```

Εικόνα Γ'.17: Ορισμός ξένου κλειδιού

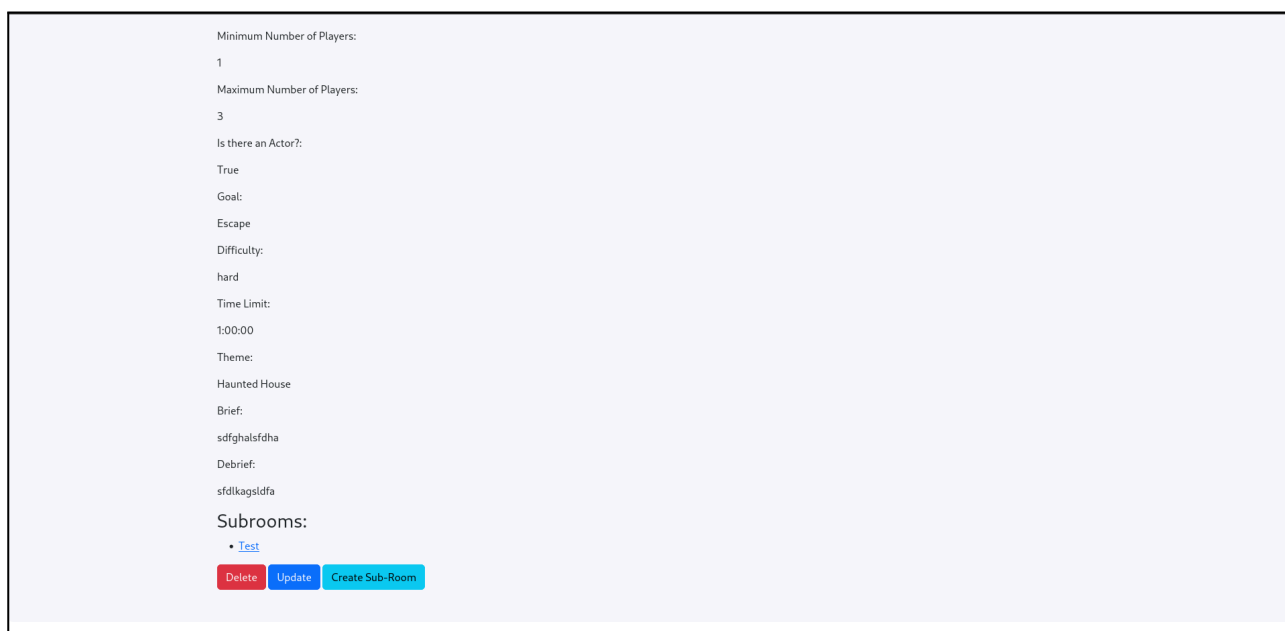
Στους ορισμούς ξένων κλειδιών έχει οριστεί η μεταβλητή `on_delete` με την τιμή `models.CASCADE`. Έτσι όταν διαγραφεί η οντότητα που συνδέεται με την παρούσα μέσω του ξένου κλειδιού, θα διαγραφεί και η ίδια.

Γ'.6.1.8 Προβολή δωματίου

Σε αυτή τη σελίδα ο χρήστης μπορεί να δει λεπτομερώς τις παραμέτρους ενός δωματίου και τις τιμές που έχει δώσει σε αυτές αν υπάρχουν.



Εικόνα Γ'.18.α: Σελίδα προβολής δωματίου



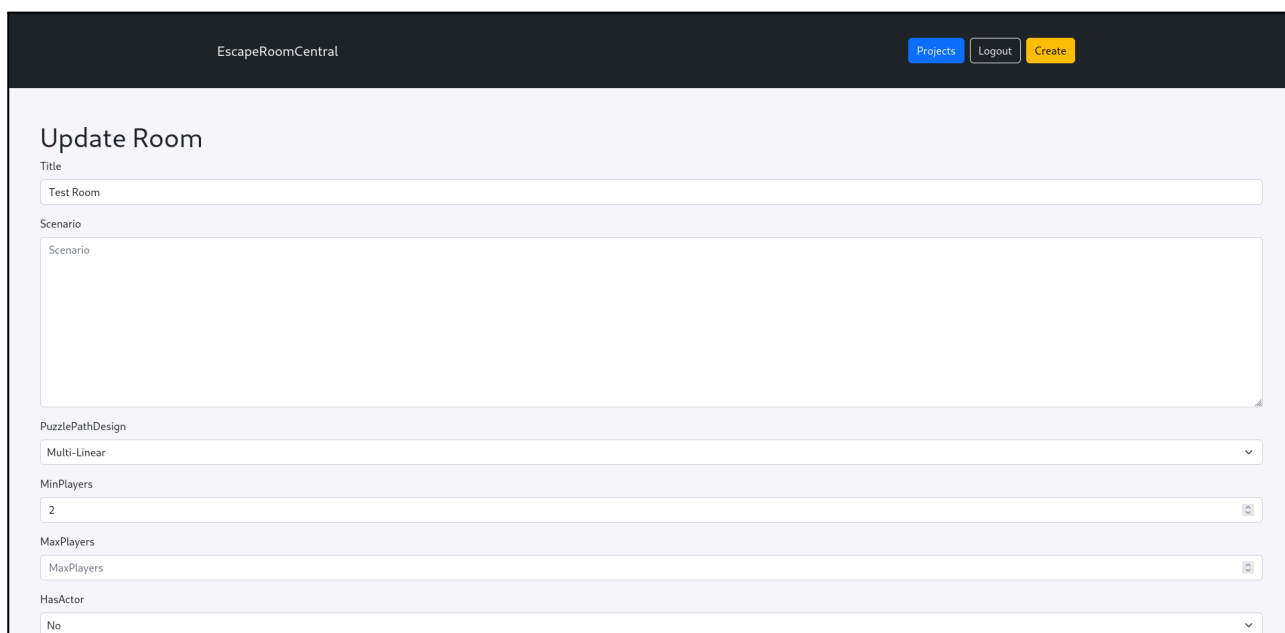
Εικόνα Γ'.18.β: Σελίδα προβολής δωματίου συνέχεια

Εικόνα Γ'.18: Σελίδα προβολής δωματίου

Οι παράμετροι παρουσιάζονται με το όνομά τους και στην συνέχεια την τιμή που τους έχει δώσει ο χρήστης. Στο κάτω μέρος της σελίδας βρίσκονται τρία κουμπιά, το κόκκινο με επιγραφή "Delete" οδηγεί στη σελίδα διαγραφής, το μπλε με επιγραφή "Update" οδηγεί στη σελίδα επεξεργασίας δωματίου όπου ο χρήστης μπορεί να αλλάξει τις τιμές των παραμέτρων και το ανοιχτό μπλε με επιγραφή "Create Sub-Room" που οδηγεί στη σελίδα δημιουργίας υποδωματίου. Σε περίπτωση που ο χρήστης δεν έχει συμπληρώσει κάποια από τα πεδία, αυτά παρουσιάζονται με κενό κάτω από τον τίτλο του πεδίου ή με τη λέξει "None".

Γ'.6.1.9 Επεξεργασία δωματίου

Η σελίδα επεξεργασίας δωματίου είναι αυτή που επιτρέπει στο χρήστη να αλλάξει τιμές των παραμέτρων που έχει δώσει στο παρελθόν ή να προσθέσει τιμές σε παραμέτρους που δεν είχαν.



The screenshot shows the 'Update Room' form in the EscapeRoomCentral application. The form is titled 'Update Room' and contains several input fields and dropdown menus. The fields are: Title (with 'Test Room' entered), Scenario (a large text area), PuzzlePathDesign (a dropdown menu with 'Multi-Linear' selected), MinPlayers (a text input with '2' entered), MaxPlayers (a text input with 'MaxPlayers' entered), and HasActor (a dropdown menu with 'No' selected). The form is styled with a light blue background and dark blue text. The top navigation bar is dark blue with the text 'EscapeRoomCentral' and three buttons: 'Projects' (blue), 'Logout' (grey), and 'Create' (yellow).

Εικόνα Γ'.19.α: Σελίδα επεξεργασίας δωματίου



The screenshot shows the bottom section of the 'Update Room' form. It includes a dropdown menu for 'TimeLimit' (with '-----' selected), a text input for 'TimeLimit' (with 'TimeLimit' entered), a text input for 'Theme' (with 'Theme' entered), a text area for 'Brief' (with 'Brief' entered), and a text area for 'Debrief' (with 'Debrief' entered). At the bottom left, there is a button labeled 'Update Room'.

Εικόνα Γ'.19.β: Σελίδα επεξεργασίας δωματίου συνέχεια

Εικόνα Γ'.19: Σελίδα επεξεργασίας δωματίου

Σε αυτή τη σελίδα υπάρχει η ίδια φόρμα με τη σελίδα δημιουργίας δωματίου. Η διαφορά είναι πως σε αυτή τη σελίδα οι παράμετροι που είχαν λάβει τιμή στο παρελθόν, την παρουσιάζουν μέσα στα πεδία της φόρμας. Έτσι είναι δυνατή η επεξεργασία αυτών, είτε αυτή θα είναι πρόσθεση νέας πληροφορίας είτε

αφαίρεση είτε συνδυασμός και των δύο. Επιπλέον υπάρχει αλλαγή στην επιγραφή της επικεφαλίδας της σελίδας και του κουμπιού επιβεβαίωσης όπου τώρα αντί για "Create Room" γράφουν "Update Room".

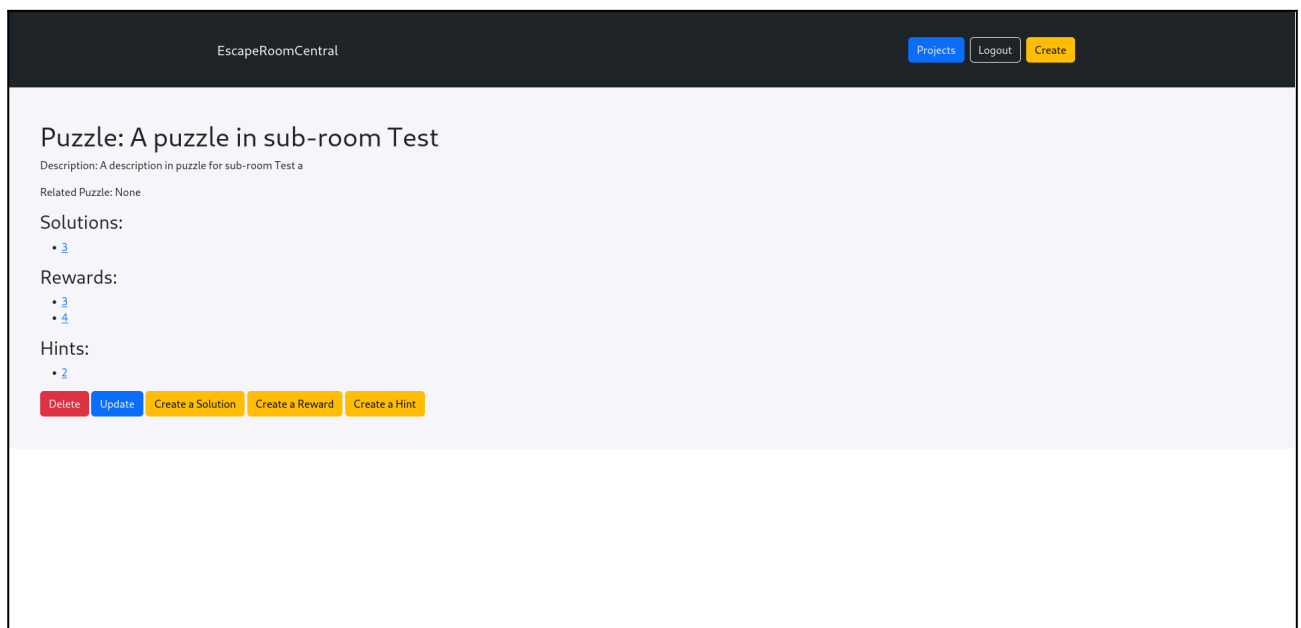
Για να δημιουργηθούν και να παραμετροποιηθούν οι φόρμες, το Django χρησιμοποιεί το αρχείο forms.py. Στην εικόνα Γ'.19 φαίνεται μια κλάση φόρμας. Σε αυτή ορίζονται τα πεδία στα οποία θα έχει πρόσβαση ο τελικός χρήστης. Για να οριστούν αυτά, χρησιμοποιούνται τα ονόματα των μεταβλητών που υπάρχουν στο αντίστοιχο models.py αρχείο για να αρχικοποιήσουν τις αντίστοιχες μεταβλητές. Παρατηρούμε λοιπόν, πως δεν υπάρχει το χαρακτηριστικό userId το οποίο υπάρχει στον ορισμό της οντότητας δωμάτιο. Αυτό συμβαίνει καθώς δεν είναι επιθυμητή η επεξεργασία αυτού του χαρακτηριστικού από τον τελικό χρήστη.

```
4 class CreateRoomForm(forms.ModelForm):
5     class Meta:
6         fields = ('title', 'scenario', 'puzzlePathDesign', 'minPlayers', 'maxPlayers', 'hasActor', 'goal', 'difficulty', 'timeLimit', 'theme', 'briefing')
7         model = models.Room
```

Εικόνα Γ'.20: Κλάση φόρμας

Γ'.6.1.10 Υπόλοιπες σελίδες

Οι σελίδες που απομένουν αφορούν τις υπόλοιπες οντότητες. Αυτές είναι το υποδωμάτιο, το παζλ, η λύση, η επιβράβευση και η βοήθεια. Οι σελίδες αυτών των οντοτήτων ακολουθούν το ίδιο μοτίβο με τις σελίδες του δωματίου. Κάθε οντότητα έχει δηλαδή μια σελίδα προβολής, μια σελίδα δημιουργίας, μια σελίδα επεξεργασίας και μια σελίδα διαγραφής. Από τη σελίδα προβολής της κάθε οντότητας ο τελικός χρήστης μπορεί να μεταβεί στη σελίδα προβολής οποιασδήποτε άλλης συσχετιζόμενης οντότητας. Σε αυτό το υποκεφάλαιο παρουσιάζονται όσες σελίδες από αυτές παρουσιάζουν κάποια ιδιαιτερότητα σε σχέση με αυτές που έχουν παρουσιαστεί ως τώρα.



Εικόνα Γ'.21: Σελίδα προβολής παζλ

Το παζλ είναι η μοναδική οντότητα που έχει συσχέτιση ένα-προς-πολλά με τρεις άλλες οντότητες επιπλέον ιδιαιτερότητα είναι η αναδρομική συσχέτιση που έχει. Αυτή εξυπηρετεί την ανάγκη συσχέτισης

παζλ μεταξύ τους, έτσι ώστε να δημιουργηθεί το puzzle path design. Η σελίδα προβολής του παζλ λοιπόν, διαφέρει από τις άλλες καθώς παρουσιάζει τρεις λίστες οντοτήτων αλλά και τη συσχέτιση που ενδεχομένως έχει με κάποιο άλλο παζλ.

The screenshot shows the 'Create a Puzzle' form in the EscapeRoomCentral application. The form is located on a light gray background. At the top, there is a dark blue header with the text 'EscapeRoomCentral' and three buttons: 'Projects' (blue), 'Logout' (gray), and 'Create' (yellow). The form itself has a title 'Create a Puzzle' and three main sections: 'Title' with a text input field containing 'A puzzle', 'Description' with a large text area containing 'A description', and 'RelatedPuzzle' with a dropdown menu. The dropdown menu is open, showing options: 'None', 'A puzzle in same', and 'A puzzle in same 2'. The 'None' option is currently selected.

Εικόνα Γ'.22: Σελίδα δημιουργίας παζλ

The screenshot shows the 'Update Puzzle' form in the EscapeRoomCentral application. The form is located on a light gray background. At the top, there is a dark blue header with the text 'EscapeRoomCentral' and three buttons: 'Projects' (blue), 'Logout' (gray), and 'Create' (yellow). The form itself has a title 'Update Puzzle' and three main sections: 'Title' with a text input field containing 'A puzzle in same 3', 'Description' with a large text area containing 'A description', and 'RelatedPuzzle' with a dropdown menu. The dropdown menu is open, showing options: 'None', 'A puzzle in same', 'A puzzle in same 2', and 'A puzzle in same 3'. The 'None' option is currently selected.

Εικόνα Γ'.23: Σελίδα επεξεργασίας παζλ

Κατ' επέκταση ιδιαίτερη είναι και η σελίδα δημιουργίας παζλ όπου υπάρχει η δυνατότητα συσχέτισης με άλλα παζλ. Όπως φαίνεται στην εικόνα Γ'.21, το παζλ μπορεί να συσχετιστεί με όλα τα άλλα παζλ πέραν του εαυτού του.

```

18 class CreatePuzzleForm(forms.ModelForm):
19     class Meta:
20         fields = ('title', 'description', 'relatedPuzzle')
21         model = models.Puzzle
22
23     def __init__(self, *args, **kwargs):
24         self.subroom = kwargs.pop('subroom', None)
25         super().__init__(*args, **kwargs)
26         self.fields['relatedPuzzle'].queryset = models.Puzzle.objects.filter(subRoomId=self.subroom)
27         self.fields['relatedPuzzle'].label_from_instance = lambda obj: obj.title
28         self.fields['relatedPuzzle'].required = False
29         self.fields['relatedPuzzle'].empty_label = "None"
30
31     def clean_relatedPuzzle(self):
32         relatedPuzzle = self.cleaned_data.get('relatedPuzzle')
33         if self.instance and relatedPuzzle == self.instance:
34             raise forms.ValidationError("A puzzle can't be related to itself.")
35         return relatedPuzzle

```

Εικόνα Γ'.24: Κώδικας φόρμας παζλ

Για να επιτευχθεί η αναδρομική συσχέτιση, είναι απαραίτητος ο κώδικας της εικόνας Γ'.22. Σε αυτή βλέπουμε τον κώδικα της φόρμας για τα παζλ στο forms.py. Στη γραμμή 23 ξεκινά η συνάρτηση `__init__`. Σε αυτή υπάρχει κώδικας που παραμετροποιεί τον τρόπο που παρουσιάζονται τα παζλ στο πεδίο επιλογής. Χωρίς αυτόν, παρουσιάζεται ο τρόπος με τον οποίο τα παζλ είναι αποθηκευμένα στη βάση και όχι με τον τίτλο τους. Ο τελικός χρήστης θα έβλεπε Puzzle Object (X). Αυτή η πληροφορία δεν είναι χρήσιμη καθώς δεν μπορεί κανείς να καταλάβει ποιο παζλ είναι το Puzzle Object (3). Στη γραμμή 31 υπάρχει ο κώδικας που ελέγχει αν ο χρήστης αποπειραθεί να συσχετίσει το παζλ με τον εαυτό του και του παρουσιάζει μήνυμα σφάλματος αν το κάνει. Αυτό είναι μόνο πιθανό όταν ο χρήστης κάνει επεξεργασία ενός παζλ καθώς κατά τη δημιουργία το εκάστοτε παζλ δεν υπάρχει ακόμη για να εμφανιστεί στη λίστα.

Γ'.6.2 Εργαλεία “no/low-code”

Τα κριτήρια για την επιλογή του no/low-code εργαλείου ήταν διαφορετικά από αυτά για την επιλογή του εργαλείου που χρησιμοποιεί κώδικα. Μιας και τα no-code εργαλεία είναι ευκολότερα στην χρήση και λιγότερο τεχνικά, τα κριτήρια ήταν σχετικότερα με τις δυνατότητες του κάθε εργαλείου. Έτσι, τα κριτήρια ήταν, ευκολία χρήσης, δυνατότητα του εργαλείου να καλύψει τις ανάγκες της εφαρμογής και κοινότητα. Το κριτήριο της κοινότητας υπάρχει καθώς καθορίζει την πιθανότητα να βρεθεί λύση ή/και βοήθεια για κάποιο ενδεχόμενο πρόβλημα στο διαδίκτυο. Το no/low-code εργαλείο που επιλέχθηκε είναι το Bubble. Αυτό μπορεί να καλύψει τις ανάγκες της εφαρμογής καθώς προσφέρει δημιουργία και διαχείριση βάσης δεδομένων, σχεδίαση της γραφικής διεπαφής και τρόπο προγραμματισμού της επικοινωνίας των δύο. Επίσης, βοηθά στην γρηγορότερη ανάπτυξη εφαρμογών δίνοντας την δυνατότητα δημιουργίας στοιχείων που προορίζονται για πολλαπλές χρήσεις, με σκοπό να εξοικονομεί χρόνο που ο δημιουργός θα έχανε επαναλαμβάνοντας την ίδια δουλειά. Επιπλέον, προσφέρει έτοιμες λειτουργίες, που χρησιμοποιούνται συχνά από τους δημιουργούς εφαρμογών όπως σύστημα σύνδεσης και εγγραφής χρήστη. Τέλος υπάρχει αναλυτική τεκμηρίωση των λειτουργιών του, με συνδέσμους στην γραφική διεπαφή του δημιουργού που οδηγούν άμεσα στο κατάλληλο σημείο της τεκμηρίωσης. Οι δύο εφαρμογές έχουν αναπτυχθεί με σκοπό η λειτουργία τους να είναι ίδια. Οποιοσδήποτε διαφορές υπάρχουν μεταξύ των δύο είναι αποτέλεσμα της επίτευξης του ίδιου στόχου αλλά με διαφορετικό, ενδεχομένως, τρόπο ανάλογα με την “φιλοσοφία” κάθε εργαλείου.

Γ'.6.2.1 Βάση δεδομένων

Σε αυτή την υποενότητα παρουσιάζεται αναλυτικά η βάση δεδομένων που αναπτύχθηκε για τις ανάγκες της εφαρμογής με το "no-code" εργαλείο Bubble. Αναλύεται με λεπτομέρεια η δομή της βάσης, ο τρόπος ανάπτυξης και τυχόν ιδιαιτερότητες.

User			Room		
Alphanumeric	<u>UniqueID</u>	PK	Alphanumeric	<u>UniqueID</u>	PK
Date	Date created		Date	Date created	
Date	Date modified		Date	Date modified	
Text	Slug		Text	Slug	
Text	email		User	Creator	
Alphanumeric	Password		text	Title	
			number	TimeLimit	
			text	Theme	
			text	Scenario	
			text	PuzzlePathDesign	
			number	MinNumPlayers	
			number	MaxNumPlayers	
			yes / no	HasActor	
			text	Goal	
			text	Difficulty	
			text	Brief	
			text	Debrief	
SubRoom					
Alphanumeric	<u>UniqueID</u>	PK			
date	Date created				
date	Date modified				
text	Slug				
User	Creator				
Room	room_id				
text	title				
Solution					
Alphanumeric	<u>UniqueID</u>	PK			
date	Date created				
date	Date modified				
text	Slug				
User	Creator				
Puzzle	Puzzle_ida				
text	Description				
Hint			Puzzle		
Alphanumeric	<u>UniqueID</u>	PK	Alphanumeric	<u>UniqueID</u>	PK
date	Date created		date	Date created	
date	Date modified		date	Date modified	
text	Slug		text	Slug	
User	Creator		User	Creator	
Puzzle	Puzzle_Id		text	title	
text	Description		Sub-Room	sub_room_id	
text	Condition		Puzzle	RelatedPuzzle	
			text	Description	
			Reward		
Alphanumeric	<u>UniqueID</u>	PK	Alphanumeric	<u>UniqueID</u>	PK
date	Date created		date	Date created	
date	Date modified		date	Date modified	
text	Slug		text	Slug	
User	Creator		User	Creator	
Puzzle	Puzzle_Id		Puzzle	Puzzle_Id	
text	Description		text	Description	

Εικόνα Γ'25: Αναλυτικό διάγραμμα οντοτήτων συσχετίσεων

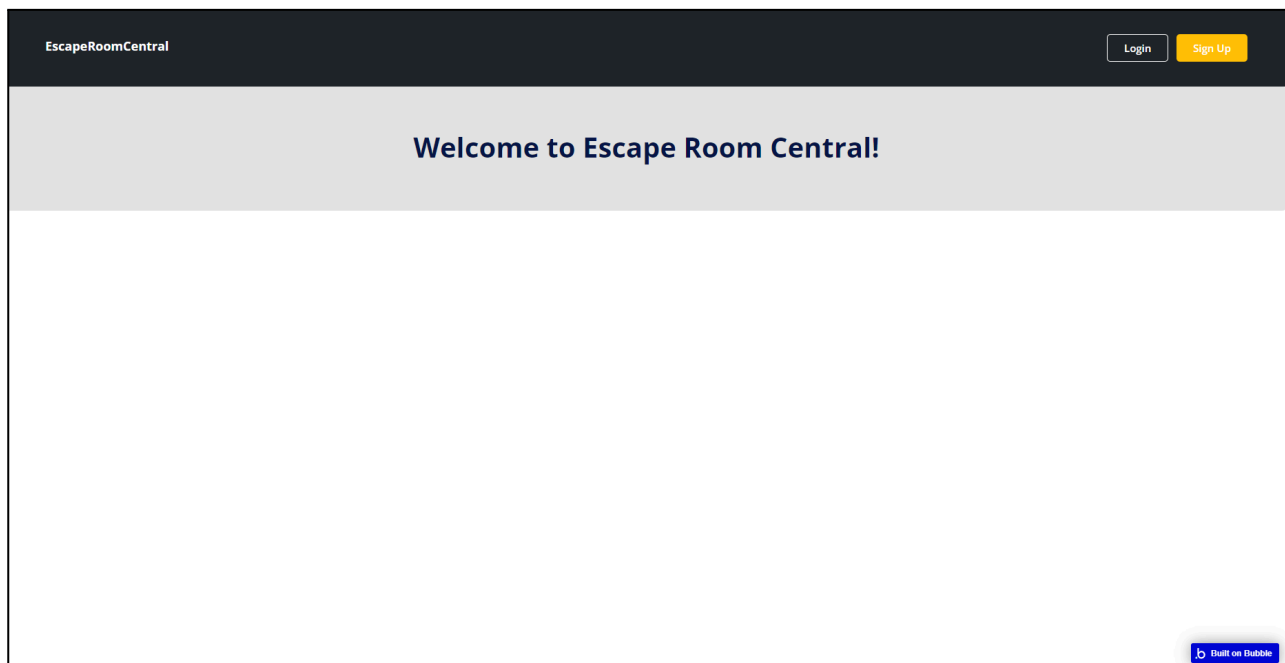
Στην εικόνα Γ'25 απεικονίζεται το αναλυτικό διάγραμμα οντοτήτων συσχετίσεων της βάσης δεδομένων που αναπτύχθηκε για την "no-code" εφαρμογή. Σε σχέση με το αντίστοιχο διάγραμμα βάσης

δεδομένων που αναπτύχθηκε για την εφαρμογή με τον κώδικα, εδώ απουσιάζουν τα βέλη που συμβολίζουν την "ένα-προς-πολλά" συσχέτιση μεταξύ των οντοτήτων. Αυτό δεν είναι τυχαίο. Αν και το Bubble χρησιμοποιεί πρωτεύοντα και ξένα κλειδιά, αυτό δεν είναι ορατό στο χρήστη. Για να δημιουργηθεί μια συσχέτιση, ο χρήστης θα πρέπει να συμπεριλάβει ένα χαρακτηριστικό τύπου οντότητας A στην οντότητα B. Έστω, για παράδειγμα οι οντότητες ΤΑΞΗ και ΜΑΘΗΤΗΣ. Για να συσχετίσουμε την ΜΑΘΗΤΗΣ με την ΤΑΞΗ θα πρέπει να της προσθέσουμε ένα χαρακτηριστικό τύπου ΤΑΞΗ. Έτσι επιτυγχάνεται συσχέτιση μεταξύ των δύο αυτών οντοτήτων. Με αυτόν το τρόπο όμως, δεν είναι δυνατόν να προσδιοριστεί ο τύπος της συσχέτισης. Δεν μπορεί, δηλαδή, ο χρήστης να ορίσει συσχέτιση "ένα-προς-ένα" ή "ένα-προς-πολλά" ή "πολλά-προς-πολλά" άμεσα από τη σχεδίαση της βάσης δεδομένων. Αυτό μπορεί μόνο να οριστεί έμμεσα μέσω της σχεδίασης της εφαρμογής και συγκεκριμένα, τον τρόπο δημιουργίας της κάθε οντότητας. Εκεί ο χρήστης μπορεί να ορίσει περιορισμούς ώστε να προσομοιάσει έναν από τους παραπάνω τύπους συσχετίσεων. Επιπλέον αξίζει να σημειωθεί, πως το Bubble δεν προσφέρει δυνατότητα διαχείρισης της συμπεριφοράς της βάσης δεδομένων σε περίπτωση διαγραφής μιας πλειάδας. Αν για παράδειγμα ο δημιουργός θελήσει να διαγράψει όλες τις πλειάδες της οντότητας B όταν διαγράφεται μια συσχετιζόμενη πλειάδα της οντότητας A, θα πρέπει να υλοποιήσει κάτι τέτοιο ο ίδιος. Από την εικόνα φαίνεται επίσης και η ύπαρξη μερικών κοινών χαρακτηριστικών σε όλες τις οντότητες. Συγκεκριμένα αυτά είναι, "Date Created", "Date Modified" και "slug". Το χαρακτηριστικό "Creator" είναι επίσης κοινό μεταξύ των οντοτήτων εκτός της "User". Αυτά τα χαρακτηριστικά παρέχονται από το Bubble και δεν μπορούν να αφαιρεθούν. Παρομοίως, η οντότητα "User" παρέχεται από το Bubble και δεν μπορεί να διαγραφεί. Το χαρακτηριστικό "UniqueID" έχει πάντα μοναδική τιμή και είναι ο τρόπος του Bubble να ξεχωρίζει τις πλειάδες. Από την τεκμηρίωση του Bubble προκύπτει πως ο τύπος μεταβλητής που αποθηκεύει το "UniqueID" είναι ένα αλφαριθμητικό 32 χαρακτήρων. Τα χαρακτηριστικά που έχουν αριθμητικό χαρακτήρα ορίζονται με τον τύπο δεδομένων "number" αποκλειστικά. Αυτό σημαίνει πως είτε ο προγραμματιστής θέλει να αποθηκεύσει έναν ακέραιο, είτε έναν αριθμό με υποδιαστολή, επιλέγει τον "number". Από την τεκμηρίωση δεν είναι προφανές αν ο αριθμός αποθηκεύεται στη βάση με τον πιο ταιριαστό τύπο δεδομένων ή αν όλοι οι αριθμοί αποθηκεύονται με τον ίδιο.

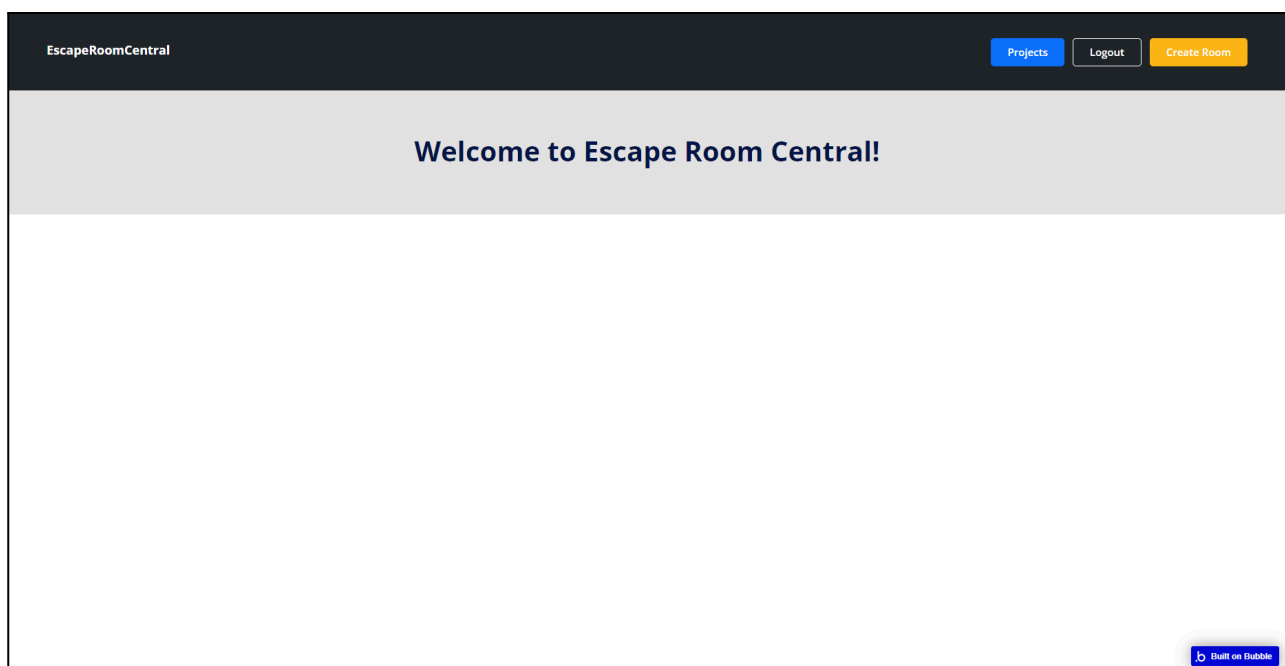
Γ'.6.2.2 Αρχική σελίδα

Η αρχική σελίδα είναι αυτή που συναντά πρώτη ο χρήστης, είτε είναι αποσυνδεδεμένος είτε είναι συνδεδεμένος. Αν είναι αποσυνδεδεμένος, από εκεί μπορεί να μεταβεί στην σελίδα σύνδεσης ή στην σελίδα εγγραφής. Αν είναι συνδεδεμένος, υπάρχουν περισσότερες δυνατότητες, πλέον μπορεί να επιλέξει να

αποσυνδεθεί, να μεταφερθεί στη σελίδα προβολής δωματίων ή να δημιουργήσει απευθείας ένα νέο δωμάτιο.



Εικόνα Γ'.25: Αρχική σελίδα αποσυνδεδεμένου χρήστη



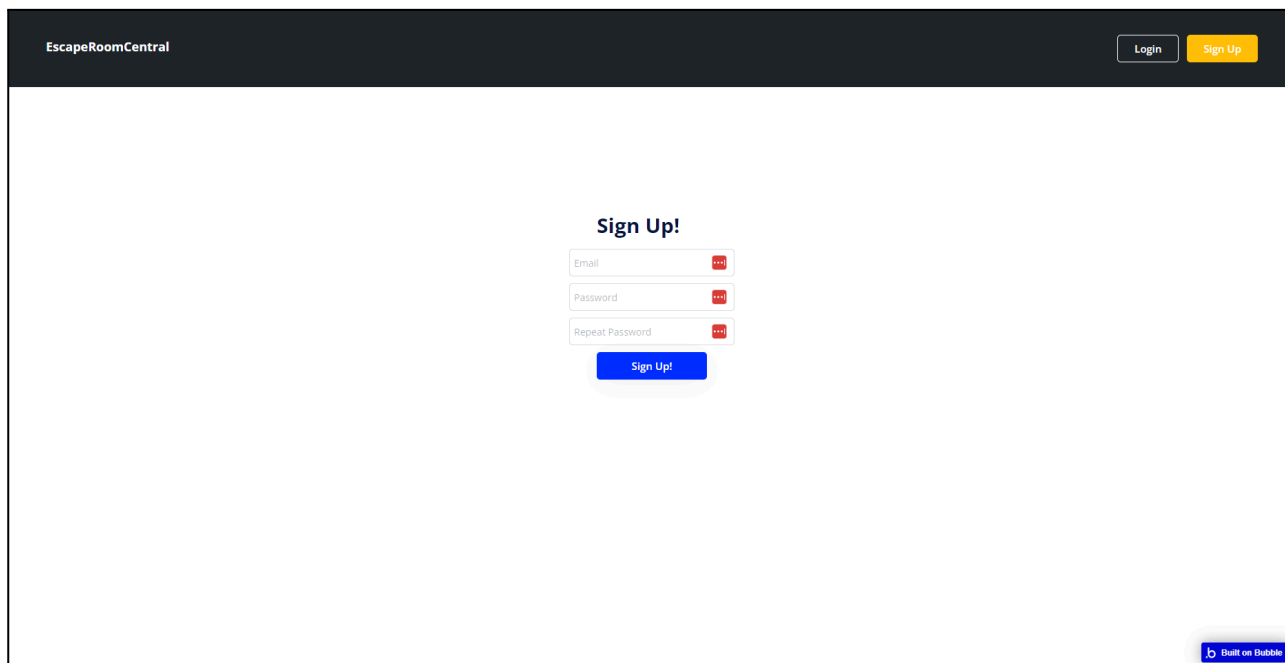
Εικόνα Γ'.26: Αρχική σελίδα συνδεδεμένου χρήστη

Όπως φαίνεται στην εικόνα Γ'.25, ένας αποσυνδεδεμένος χρήστης μπορεί να συνδεθεί πατώντας το σκούρο κουμπί με επιγραφή "Login", έτσι θα μεταβεί στη σελίδα σύνδεσης. Μπορεί επίσης να εγγραφεί, χρησιμοποιώντας το κίτρινο κουμπί με επιγραφή "Sign Up" που τον μεταφέρει στη σελίδα εγγραφής. Στην εικόνα Γ'.26 φαίνεται η αρχική σελίδα ενός συνδεδεμένου χρήστη. Από εκεί υπάρχει η δυνατότητα αποσύνδεσης από το σκούρο κουμπί με επιγραφή "Logout" το οποίο τον αποσυνδέει και τον μεταφέρει στην αρχική σελίδα ως αποσυνδεδεμένο. Με το μπλε κουμπί με επιγραφή "Projects" μεταφέρεται στη σελίδα προβολής δωματίων όπου μπορεί να δει τα δωμάτια που έχει δημιουργήσει. Πατώντας το κίτρινο κουμπί με

επιγραφή “Create Room” πηγαίνει στη σελίδα δημιουργίας νέου δωματίου. Τέλος, πατώντας την επιγραφή “EscapeRoomCentral” μπορεί να επιστρέψει στην αρχική σελίδα.

Γ'.6.2.3 Εγγραφή

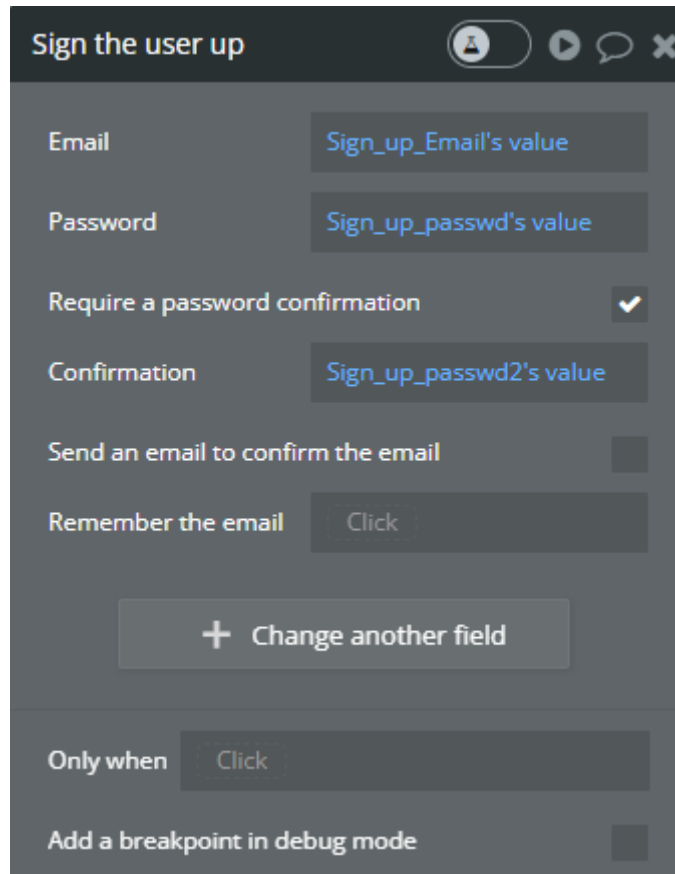
Στη σελίδα εγγραφής ο χρήστης μπορεί να δημιουργήσει ένα νέο λογαριασμό. Εκεί θα πρέπει να συμπληρώσει κατάλληλα τη φόρμα που υπάρχει με τα στοιχεία που επιθυμεί σε κάθε ένα από τα πεδία.



Εικόνα Γ'.27: Σελίδα εγγραφής

Στην εικόνα Γ'.27 φαίνεται πως ο χρήστης θα πρέπει να συμπληρώσει τη διεύθυνση του ηλεκτρονικού του ταχυδρομείου στο πρώτο πεδίο με επιγραφή "Email". Αυτό δέχεται σαν είσοδο μια διεύθυνση ηλεκτρονικού ταχυδρομείου και είναι υποχρεωτικό να συμπληρωθεί. Αυτό σημαίνει πως η είσοδος του χρήστη θα πρέπει να είναι της μορφής "ΑΛΦΑΡΙΘΜΗΤΙΚΟ@ΑΛΦΑΡΙΘΜΗΤΙΚΟ.ΑΛΦΑΡΙΘΜΗΤΙΚΟ". Το αλφαριθμητικό πριν το "@" μπορεί να περιέχει λατινικούς χαρακτήρες, αριθμούς και σύμβολα. Αυτό που βρίσκεται μετά το "@" αλλά πριν την τελεία, μπορεί να περιέχει λατινικούς χαρακτήρες και νούμερα. Τέλος, το τελευταίο πρέπει να αποτελείται αποκλειστικά από λατινικούς χαρακτήρες. Τα δύο τελευταία πεδία έχουν να κάνουν με τον κωδικό που επιθυμεί να χρησιμοποιήσει ο χρήστης. Τα πεδία αυτά θα πρέπει να περιέχουν ακριβώς την ίδια είσοδο η οποία είναι αλφαριθμητική. Αυτό συμβαίνει για να εξασφαλιστεί ότι η είσοδος του χρήστη είναι η επιθυμητή. Επίσης τα πεδία αυτά θα πρέπει υποχρεωτικά να συμπληρωθούν. Τέλος, πατώντας το μπλε κουμπί στο τέλος της σελίδας με επιγραφή "Sign Up!" και δεδομένου ότι τα πεδία έχουν συμπληρωθεί σωστά, δημιουργείται ο λογαριασμός του χρήστη και αυτός μεταφέρεται στην αρχική σελίδα ως συνδεδεμένος. Σε αυτή τη σελίδα υπάρχει ουσιαστική διαφορά μεταξύ των δύο υλοποιήσεων. Καθώς το Bubble απαιτεί κάθε χρήστης που εγγράφεται να έχει μοναδική διεύθυνση ηλεκτρονικού ταχυδρομείου δεν υπάρχει, στην προκειμένη περίπτωση, λόγος για την προσθήκη πεδίου ονόματος χρήστη. Η διαδικασία σύνδεσης, που είναι υλοποιημένη και προσφέρεται έτοιμη από το Bubble, κάνει χρήση της διεύθυνσης

ηλεκτρονικού ταχυδρομείου για να πετύχει τη λειτουργικότητα. Επιπλέον το Bubble δεν προσφέρει αυτόματα οδηγίες για τη σωστή συμπλήρωση της φόρμας εγγραφής αλλά ούτε και κάποια απαραίτητα ελάχιστα χαρακτηριστικά που θα πρέπει να τηρεί ο κωδικός χρήστη ώστε να είναι επαρκώς ασφαλής. Μερικά από αυτά θα μπορούσαν να είναι το ελάχιστο όριο ψηφίων, η υποχρεωτική συμπερίληψη διαφορετικών ειδών ψηφίων (χαρακτήρες, αριθμούς, σύμβολα, κ.α.), ή αποφυγή χρήσης επαναλαμβανόμενων ψηφίων κ.α.

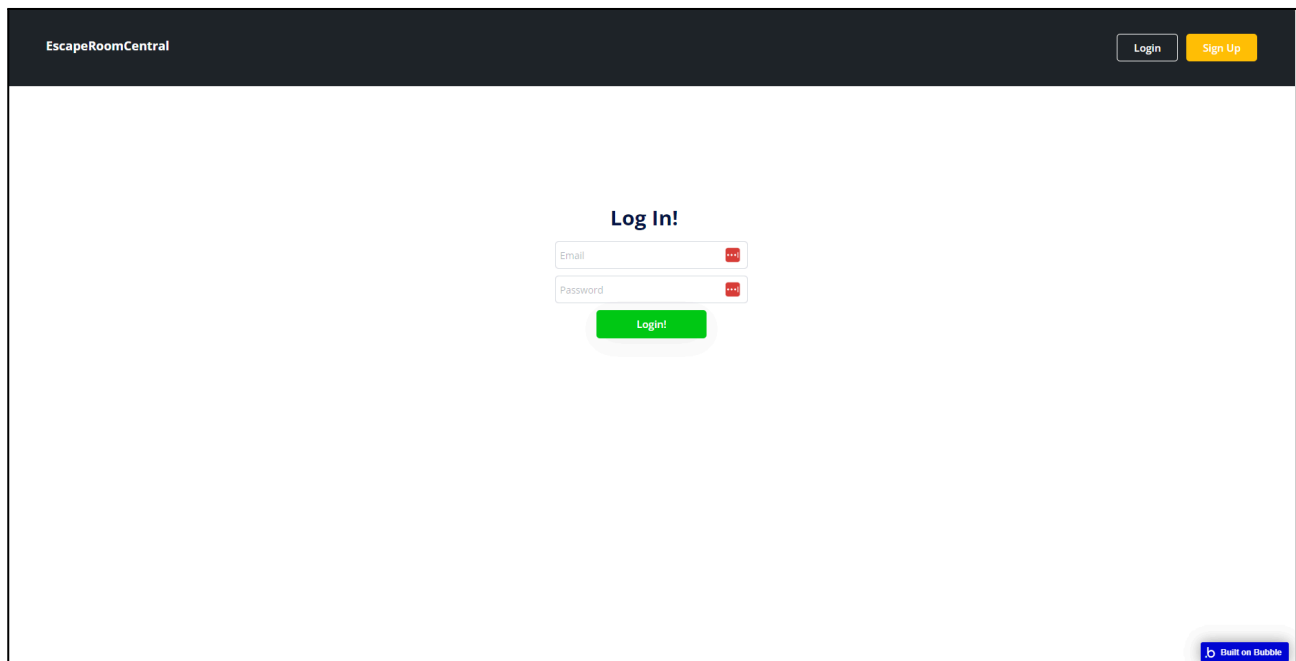


Εικόνα Γ'.28: Ορισμός παραμέτρων ενέργειας "Sign the user up"

Όπως αναφέρθηκε παραπάνω και γίνεται εμφανές με την εικόνα Γ'.28, ο προγραμματιστής είναι υποχρεωμένος να χρησιμοποιήσω τη διεύθυνση ηλεκτρονικού ταχυδρομείου για να δημιουργηθεί ένας λογαριασμός χρήστη. Προσφέρονται επιπροσθέτως λειτουργίες όπως η επιβεβαίωση του κωδικού χρήστη και η αποστολή μηνύματος ηλεκτρονικού ταχυδρομείου ώστε να επιβεβαιωθεί και η σωστή εισαγωγή της διεύθυνσης ηλεκτρονικού ταχυδρομείου. Το μόνο που έχει να κάνει ο δημιουργός, είναι να ορίσει τα πεδία από τα οποία η συγκεκριμένη ενέργεια θα λάβει τις κατάλληλες πληροφορίες για να μπορεί να ολοκληρωθεί.

Γ'.6.2.4 Σύνδεση

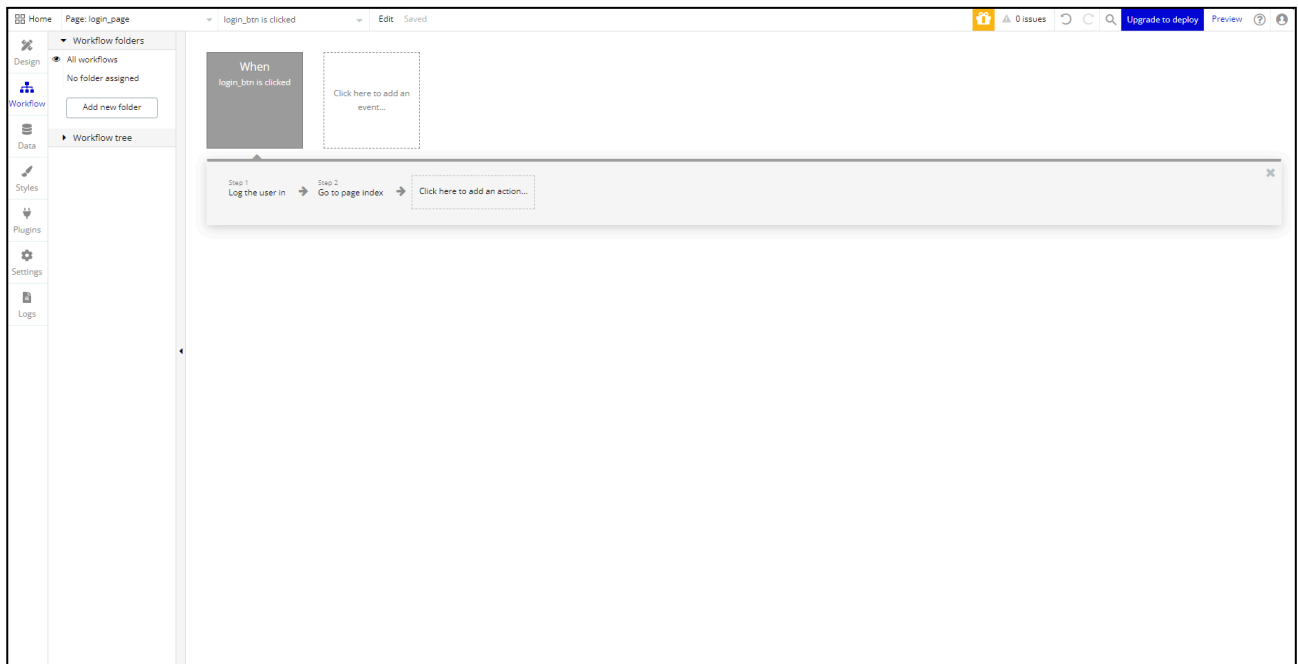
Αυτή η σελίδα δίνει την δυνατότητα στον χρήστη να συνδεθεί σε ένα λογαριασμό που έχει δημιουργήσει στο παρελθόν με σκοπό να έχει πρόσβαση σε αποθηκευμένα δεδομένα.



Εικόνα Γ'.29: Σελίδα σύνδεσης

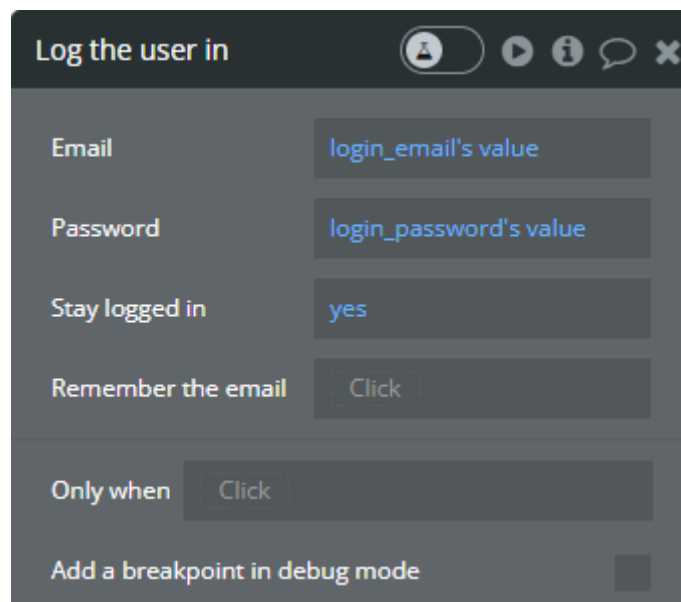
Εδώ ο χρήστης θα πρέπει να συμπληρώσει τη φόρμα με στοιχεία ενός λογαριασμού που υπάρχει ήδη. Όταν ενεργοποιήσει το πράσινο κουμπί με επιγραφή "Login!" και εφόσον τα στοιχεία που έχει συμπληρώσει επαληθευθούν στη βάση δεδομένων, ο χρήστης θα μεταφερθεί στην αρχική σελίδα ως συνδεδεμένος. Σε περίπτωση που τα στοιχεία που εισάγει ο χρήστης δεν επαληθευτούν, εμφανίζεται μήνυμα σφάλματος. Από την μεριά του ο δημιουργός έχει παρά μόνο να προσδιορίσει στο Bubble τη λειτουργία "Σύνδεσε το χρήστη" ή "Log user in" προσδιορίζοντας τα σημεία όπου ο χρήστης προσφέρει τα αντίστοιχα δεδομένα. Το Bubble αναλαμβάνει να επιβεβαιώσει αυτά τα στοιχεία στην βάση και να κάνει τις απαραίτητες αλλαγές που απαιτούνται για να θεωρείται ο χρήστης συνδεδεμένος, όπως αλλαγές στην γραφική διεπαφή και αλλαγή των δικαιωμάτων του χρήστη. Για να προγραμματιστεί η λογική, το Bubble χρησιμοποιεί τα "Workflows". Μέσω γραφικής διεπαφής αυτά δίνουν τη δυνατότητα στον προγραμματιστή

να ορίσει τις διάφορες λειτουργίες που επιθυμεί.



Εικόνα Γ'.30: Παράδειγμα "Workflow"

Στη συγκεκριμένη περίπτωση πρέπει να επιλεγεί η "Log the user in" ενέργεια. Αυτή αναλαμβάνει να συνδέσει το χρήστη. Ο προγραμματιστής πρέπει να ορίσει μερικές παραμέτρους που χρειάζεται η συγκεκριμένη ενέργεια. Το δεύτερο βήμα είναι η μεταφορά του χρήστη σε διαφορετική σελίδα. Εδώ, ο χρήστης μεταφέρεται στην αρχική σελίδα της εφαρμογής όντας πλέον συνδεδεμένος.



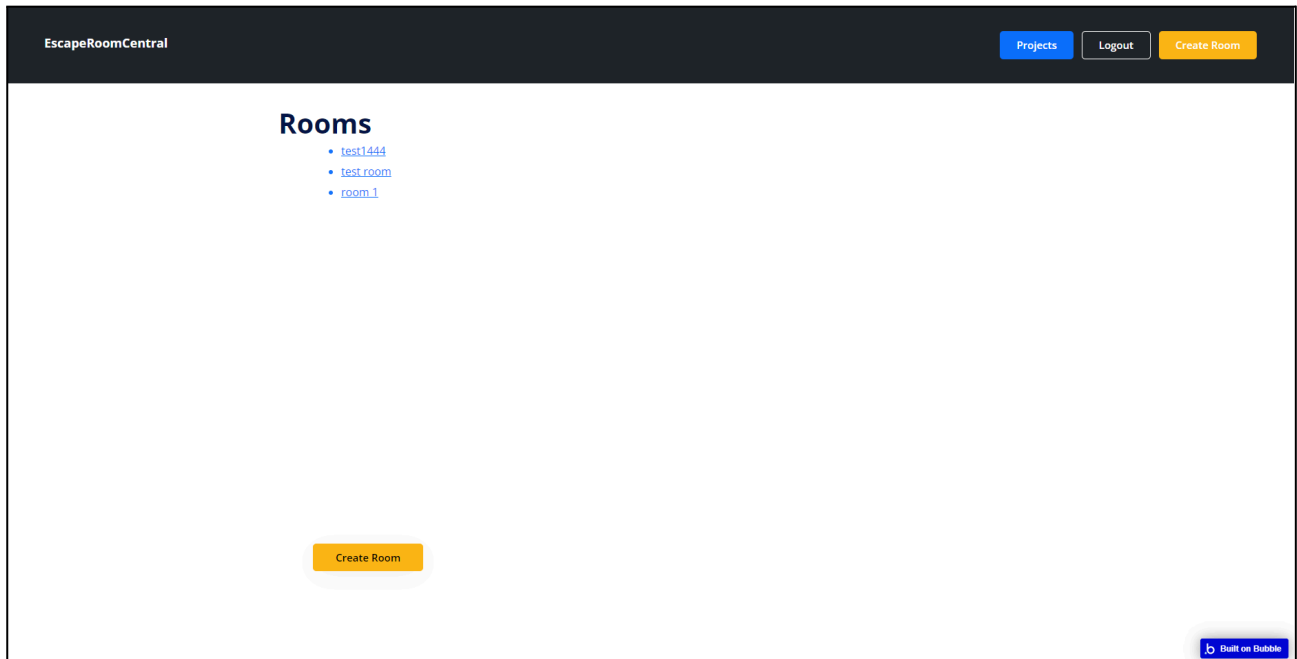
Εικόνα Γ'31: Ορισμός παραμέτρων ενέργειας "Log the user in"

Οι παράμετροι αυτοί είναι η διεύθυνση ηλεκτρονικού ταχυδρομείου και ο κωδικός. Ο προγραμματιστής πρέπει ουσιαστικά να πει στο Bubble από ποιο πεδίο της φόρμας θα πάρει την τιμή για κάθε μια από τις παραμέτρους. Η συγκεκριμένη ενέργεια, έχει συγκεκριμένες παραμέτρους που πρέπει να

λάβουν τιμή. Στη συνέχεια αυτής της ενότητας παρουσιάζονται και παράμετροι που έχουν μεταβαλλόμενο αριθμό και τύπο παραμέτρων.

Γ'.6.2.5 Σελίδα προβολής λίστας δωματίων

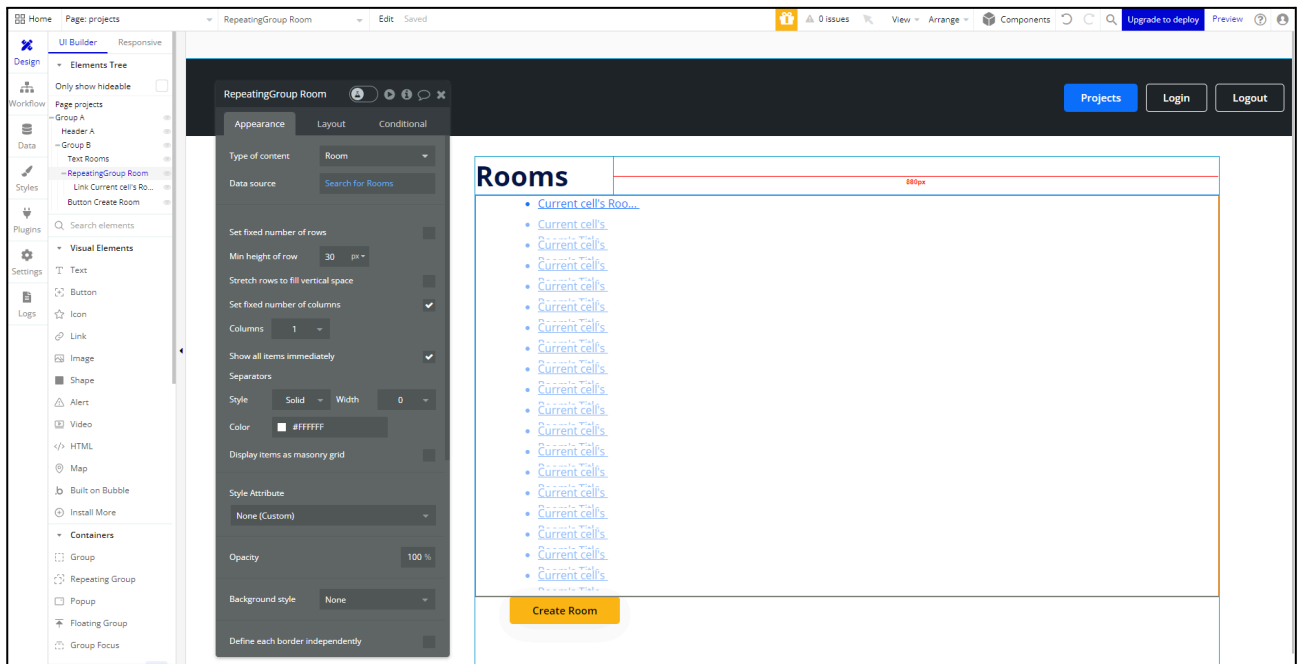
Σε αυτή την σελίδα ο χρήστης μπορεί να δει τη λίστα με τα δωμάτια που έχει δημιουργήσει. Καθένα από τα αντικείμενα της λίστας είναι σύνδεσμος που, αν τον επιλέξει ο χρήστης, τον οδηγεί στην σελίδα προβολής λεπτομερειών του αντίστοιχου δωματίου.



Εικόνα Γ'.32: Σελίδα προβολής λίστας δωματίων

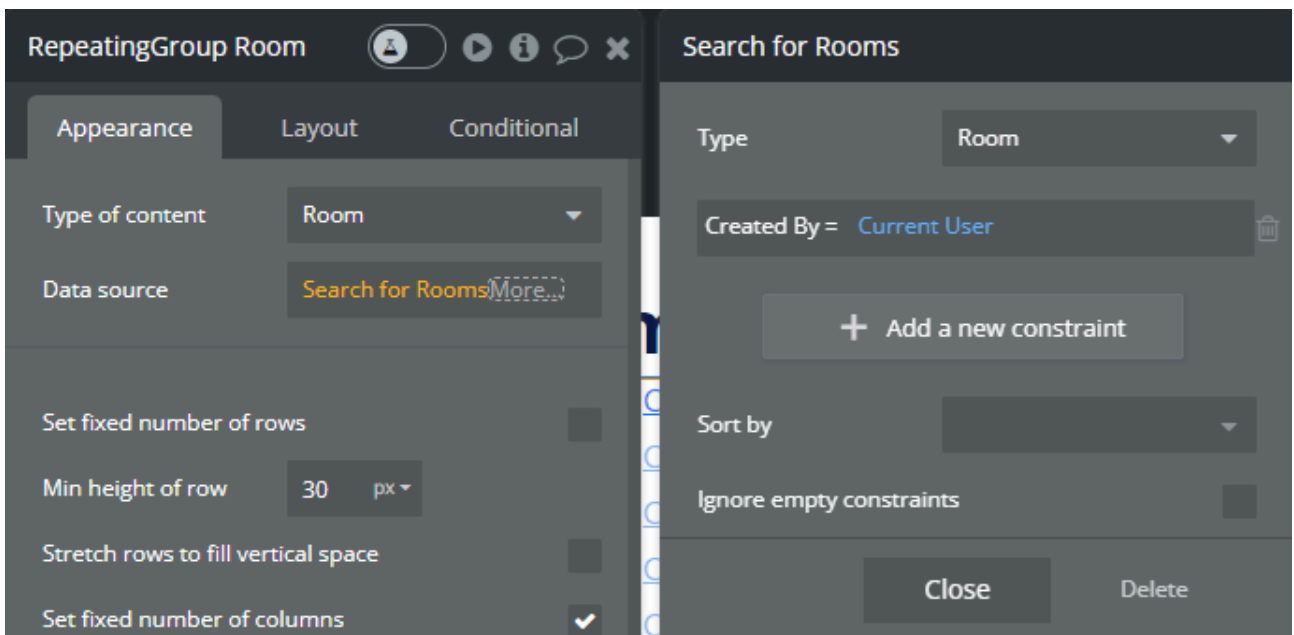
Όπως φαίνεται στην εικόνα Γ'.32, ο χρήστης βλέπει τη λίστα με τα δωμάτια που έχει δημιουργήσει στο επάνω και αριστερό σημείο της σελίδας. Στο τέλος αυτής υπάρχει και ένα κίτρινο κουμπί με επιγραφή "Create Room". Αν ο χρήστης το επιλέξει, θα μεταφερθεί στην σελίδα δημιουργίας δωματίου. Αν ο χρήστης επιλέξει έναν από τους τίτλους των δωματίων της λίστας, μεταφέρεται στη σελίδα που παρουσιάζει τις

λεπτομέρειες του αντίστοιχου δωματίου.

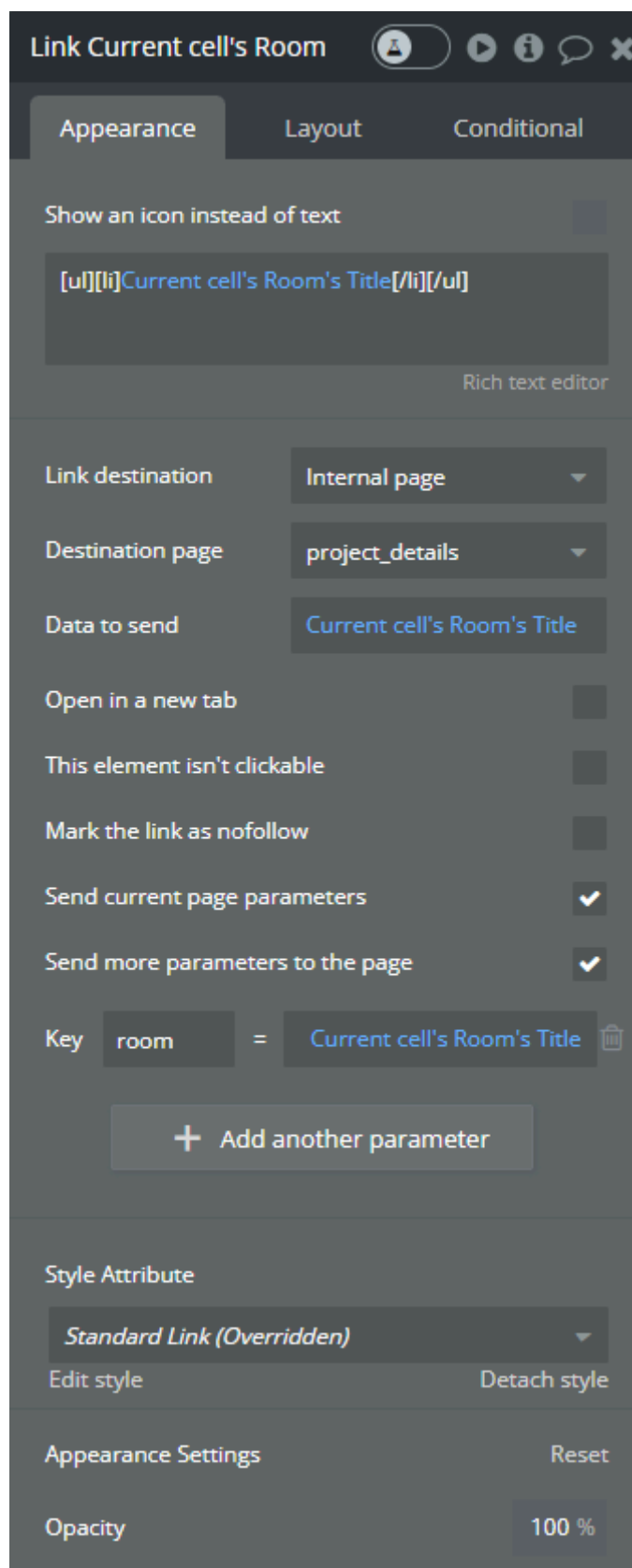


Εικόνα Γ'.33: Τρόπος δημιουργίας της λίστας δωματίων

Για να δημιουργηθεί η λίστα των δωματίων, πρέπει ο προγραμματιστής να κάνει χρήση ενός δοχείο με τίτλο "RepeatingGroup". Αυτό, δίνει τη δυνατότητα να επαναλαμβάνεται ένα στοιχείο το οποίο τοποθετείται μέσα σε αυτό. Το στοιχείο που χρειάζεται στη συγκεκριμένη περίπτωση είναι το "Link" ή υπερσύνδεσμος. Ο υπερσύνδεσμος πρέπει να προγραμματιστεί ώστε να εμφανίζει δυναμικά το κατάλληλο κάθε φορά κείμενο που είναι ο τίτλος κάθε δωματίου που έχει δημιουργηθεί. Αυτό απαιτεί την παραμετροποίηση και των δύο στοιχείων "RepeatingGroup" και "Link".



Εικόνα Γ'.34: Προγραμματισμός "RepeatingGroup"



Εικόνα Γ'.35: Προγραμματισμός "Link"

Στις παραπάνω εικόνες, φαίνεται η κατάλληλη παραμετροποίηση των δύο στοιχείων ώστε να επιτυγχάνεται το επιθυμητό αποτέλεσμα που είναι η παρουσίαση, ως σύνδεσμο, του τίτλου του κάθε δωματίου που έχει δημιουργηθεί από των εκάστοτε χρήστη. Αναλυτικότερα, στην εικόνα Γ'.34 φαίνεται πως

το "RepeatingGroup" έχει παραμετροποιηθεί ώστε να πραγματοποιεί μια αναζήτηση για οντότητες τύπου "δωμάτιο". Η λίστα των αποτελεσμάτων περιορίζεται στα δωμάτια που έχει δημιουργήσει ο τωρινός χρήστης, ο χρήστης δηλαδή που είναι αυτή τη στιγμή συνδεδεμένος. Η αναζήτηση αυτή μπορεί να στοχευεί περαιτέρω προσθέτοντας επιπλέον περιορισμούς πατώντας το κουμπί "Add a new constrained". Το στοιχείο "Link" έχει παραμετροποιηθεί ώστε να παίρνει το δωμάτιο του κάθε κελιού του "RepeatingGroup" και να εμφανίζει το χαρακτηριστικό "τίτλος" του. Επιπλέον έχει παραμετροποιηθεί να κάνει αυτό το, δυναμικά φτιαγμένο κείμενο, ένα σύνδεσμο προς μια εσωτερική σελίδα με τίτλο "project_details". Η σελίδα αυτή παρουσιάζει λεπτομερώς τα στοιχεία του κάθε δωματίου. Τέλος η μεταφορά στη νέα σελίδα γίνεται με την προσθήκη στο URL των παραμέτρων της σελίδας προβολής λίστας δωματίων και μιας επιπλέον παραμέτρου με τίτλο "room" και τιμή τον τίτλο του δωματίου που επιλέχθηκε. Αυτό είναι απαραίτητο για να μπορούν να παρουσιάζονται τα σωστά στοιχεία στη σελίδα προβολής λεπτομερειών δωματίου.

Γ'.6.2.6 Δημιουργία δωματίου

Συμπληρώνοντας τη φόρμα στη σελίδα δημιουργίας δωματίου, ο χρήστης μπορεί να δημιουργήσει ένα δωμάτιο ώστε να αποθηκευτεί η πληροφορία του για πιθανή μελλοντική χρήση. Δεν απαιτείται από τον χρήστη να συμπληρώσει όλα τα πεδία της φόρμας, παρά μόνο τον τίτλο. Η λειτουργία αυτή είναι σχεδιασμένη έτσι από επιλογή, με σκοπό να δώσει στο χρήστη την ελευθερία να μη δημιουργήσει ολόκληρο το δωμάτιο μονομιάς, αλλά να μπορεί να επιστρέψει σε αυτό και να συμπληρώσει επιπρόσθετη πληροφορία στο μέλλον. Ο στόχος αυτός επιτυγχάνεται σε συνεργασία και με την λειτουργία επεξεργασίας ενός δωματίου που ήδη υπάρχει.

The screenshot shows the 'Create a New Room!' form in the EscapeRoomCentral application. The form is located in the center of the page, with a dark-themed header and footer. The header contains the 'EscapeRoomCentral' logo on the left and 'Projects', 'Logout', and 'Create Room' buttons on the right. The form itself has a white background and includes the following fields:

- Title**: A text input field.
- Scenario**: A large text area for a detailed description.
- Puzzle Path Design**: A dropdown menu.
- Minimum Number Of Players**: A text input field with the value '1'.
- Maximum Number Of Players**: A text input field with the placeholder 'Max players number'.
- Has Actor?**: A checkbox.
- Goal**: A text input field.
- Difficulty**: A dropdown menu.
- Time Limit**: A text input field.

In the bottom right corner of the form, there is a small button that says 'Built on Bubble'.

Εικόνα Γ'.36.α: Σελίδα δημιουργίας δωματίου

Minimum Number Of Players

1

Maximum Number Of Players

Max players number

☐ Has Actor?

Goal

Goal

Difficulty

Difficulty

Time Limit

Time Limit

Theme

Theme

Brief

Brief

Debrief

Debrief

Create Room!

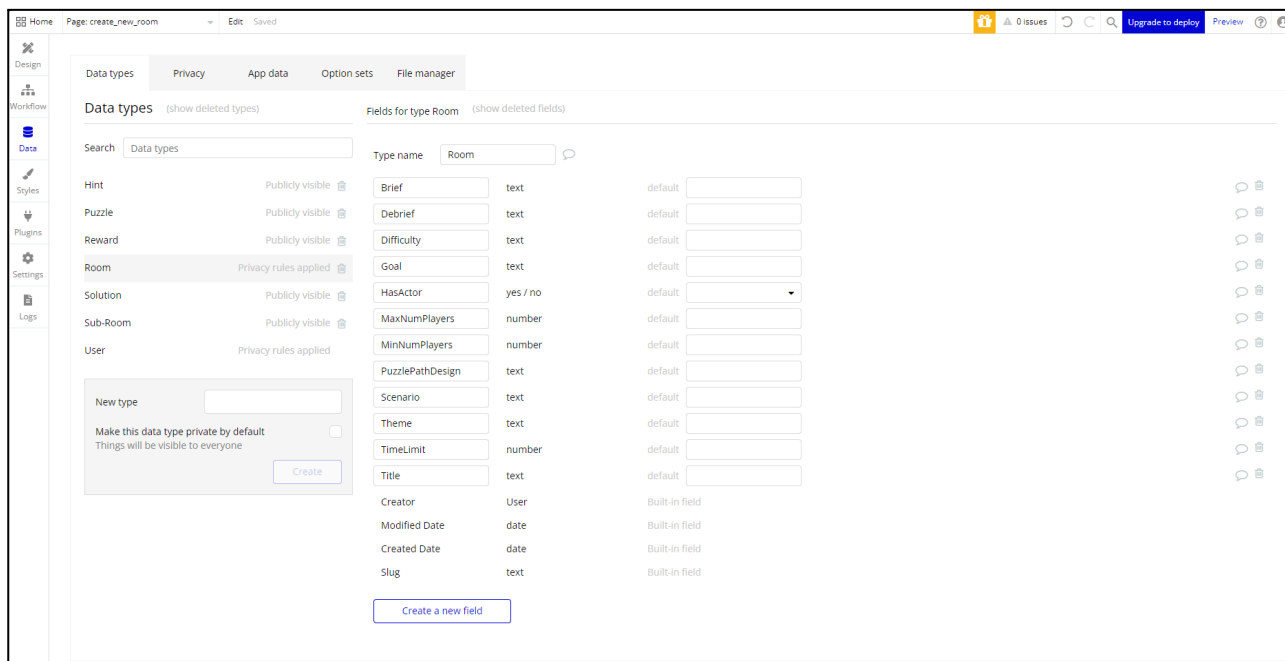
Built on Bubble

Εικόνα Γ'.36.β: Σελίδα δημιουργίας δωματίου συνέχεια

Εικόνα Γ'.36: Σελίδα δημιουργίας δωματίου

Αναλυτικότερα λοιπόν ο χρήστης δύναται να συμπληρώσει 12 πεδία. Το πεδίο του τίτλου είναι το μοναδικό που είναι απαραίτητο να συμπληρωθεί, δέχεται τιμές τύπου "text" μέγιστου μεγέθους 256 χαρακτήρων και είναι τοποθετημένο στην κορυφή της φόρμας με επιγραφή "Title". Τα υπόλοιπα πεδία δίνουν τη δυνατότητα παραμετροποίησης του σεναρίου, του σχεδιασμού της ακολουθίας των παζλ, του ελάχιστου και μέγιστου αριθμού παικτών, την ύπαρξη ή μη ηθοποιού, τον στόχο του δωματίου, τη δυσκολία όπου δίνονται 3 επιλογές με τη μορφή "Drop down menu", το χρονικού ορίου, της θεματικής και τέλος του "Brief" και "Debrief". Αξίζει να σημειωθεί πως για το πεδίο του χρονικού περιθωρίου δεν προσφέρεται κάποιος τύπος δεδομένων που να εξυπηρετεί ικανοποιητικά την είσοδο που δέχεται το συγκεκριμένο πεδίο.

Έτσι, δέχεται δεδομένα τύπου text.



Εικόνα Γ'.37: Οντότητα "δωμάτιο" στη βάση δεδομένων

Στην εικόνα Γ'.37 παρουσιάζεται ταυτόχρονα η οντότητα "δωμάτιο" αλλά και ο τρόπος που παρέχει το Bubble για τη διαχείριση της βάσης δεδομένων. Με το κουμπί "Create a new field" μπορεί να προσθέσει ο προγραμματιστής ένα χαρακτηριστικό σε μία οντότητα. Παραπάνω του κουμπιού αυτού φαίνονται τα χαρακτηριστικά που έχει η συγκεκριμένη οντότητα μαζί με τον τύπο δεδομένων που τους έχει δοθεί. Στη στήλη αριστερά, φαίνονται όλες οι υπάρχουσες οντότητες. Φαίνονται επίσης και τα τέσσερα προϋπάρχοντα χαρακτηριστικά που αναφέρθηκαν στην υποενότητα Γ'.6.2.1 τα οποία δεν μπορεί ο προγραμματιστής να διαγράψει.

The image shows a modal dialog box titled 'Create a new field'. It contains three input fields: 'Field name' with a text input, 'Field type' with a dropdown menu, and 'This field is a list (multiple entries)' with a checkbox. At the bottom of the dialog, there are two buttons: 'CREATE' (highlighted in blue) and 'Cancel'.

Εικόνα Γ'.38: Δημιουργία χαρακτηριστικού για μία οντότητα

Για τη δημιουργία και προσθήκη ενός χαρακτηριστικού σε μία οντότητα, πρέπει να συμπληρωθεί η μικρή φόρμα της εικόνας Γ'.38. Σε αυτή ο προγραμματιστής επιλέγει το όνομα του χαρακτηριστικού και τον τύπο δεδομένων που το απαρτίζουν.

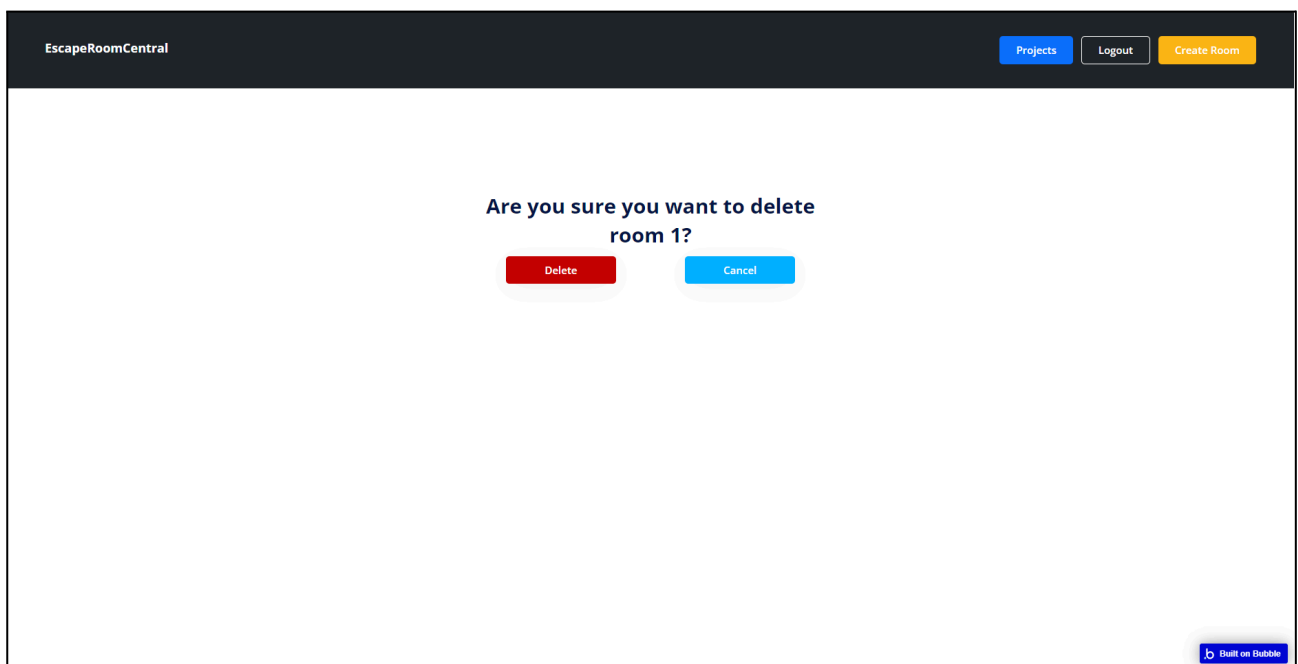


Εικόνα Γ'.39: Δημιουργία οντότητας

Τέλος, για να δημιουργηθεί μια νέα οντότητα, ο προγραμματιστής, δεν έχει παρά μόνο να συμπληρώσει τον τίτλο της οντότητας αυτής και να επιλέξει αν θα είναι ιδιωτική.

Γ'.6.2.7 Διαγραφή δωματίου

Η σελίδα αυτή ακολουθεί την επιλογή διαγραφής ενός δωματίου από το χρήστη. Η επιλογή αυτή υπάρχει στη σελίδα προβολής λεπτομερειών του κάθε δωματίου. Ο σκοπός της είναι η αποφυγή μιας ανεπιθύμητης διαγραφής δωματίου από το χρήστη σε περίπτωση κατά λάθος ενεργοποίησης του αντίστοιχου κουμπιού στη σελίδα προβολής λεπτομερειών δωματίου.



Εικόνα Γ'.40: Σελίδα διαγραφής δωματίου

Στην εικόνα Γ'.40 φαίνεται πως ο χρήστης βλέπει ένα μήνυμα επιβεβαίωσης διαγραφής του αντίστοιχου δωματίου του οποίου το κουμπί διαγραφής πατήθηκε. Επιπλέον βλέπει δύο κουμπιά, ένα κόκκινο με επιγραφή "Delete" και ένα μπλε με επιγραφή "Cancel". Αν επιλέξει το κόκκινο, τότε το δωμάτιο θα διαγραφεί και ο χρήστης θα μεταφερθεί στη σελίδα προβολής λίστας δωματίων του, αν επιλεγθεί το μπλε, τότε η διαδικασία διαγραφής δε γίνεται και ο χρήστης επιστρέφει στη σελίδα προβολής του κατάλληλου δωματίου. Για να εμφανιστεί ο κατάλληλος τίτλος δωματίου στο τέλος της πρότασης που βλέπει ο χρήστης, γίνεται χρήση της παραμέτρου "room" του URL που είδαμε αναφέραμε στην υποενότητα της σελίδας προβολής λίστας δωματίων.

?room=room%201

Εικόνα Γ'.41: Παράμετρος "room" στο URL

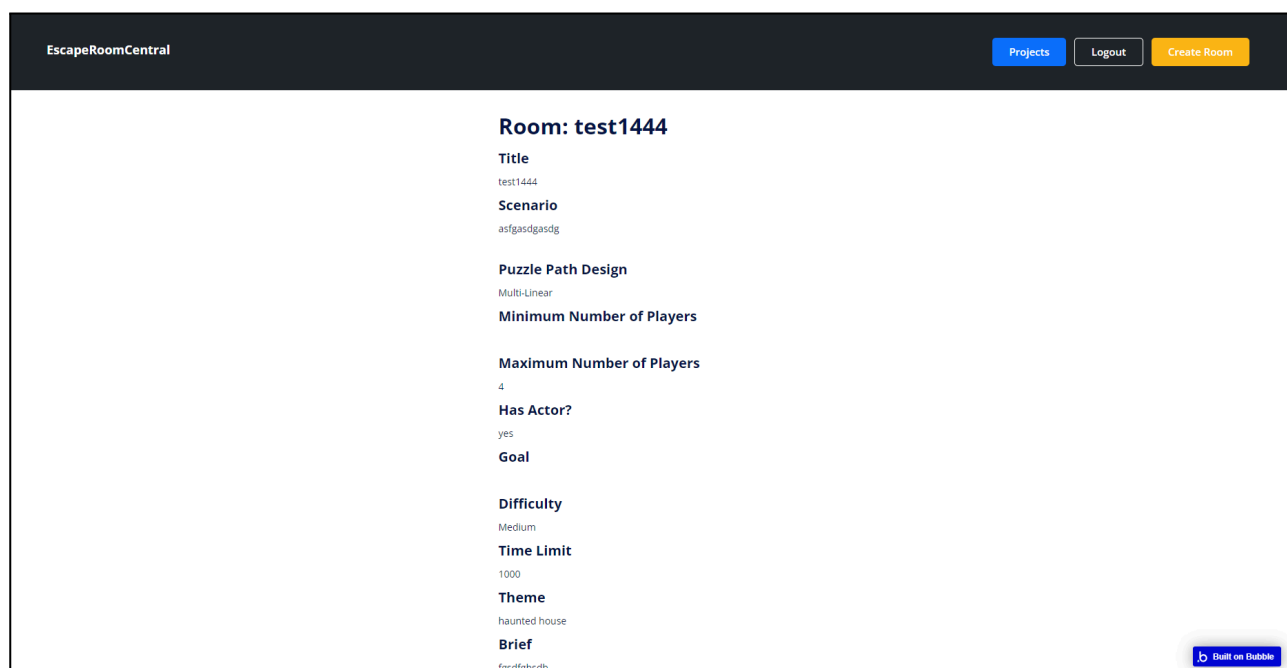
Εικόνα Γ'.42.α: Παραμετροποίηση της ενέργειας "Get data from page URL"

Εικόνα Γ'.42.β: Παραμετροποίηση του στοιχείου κειμένου

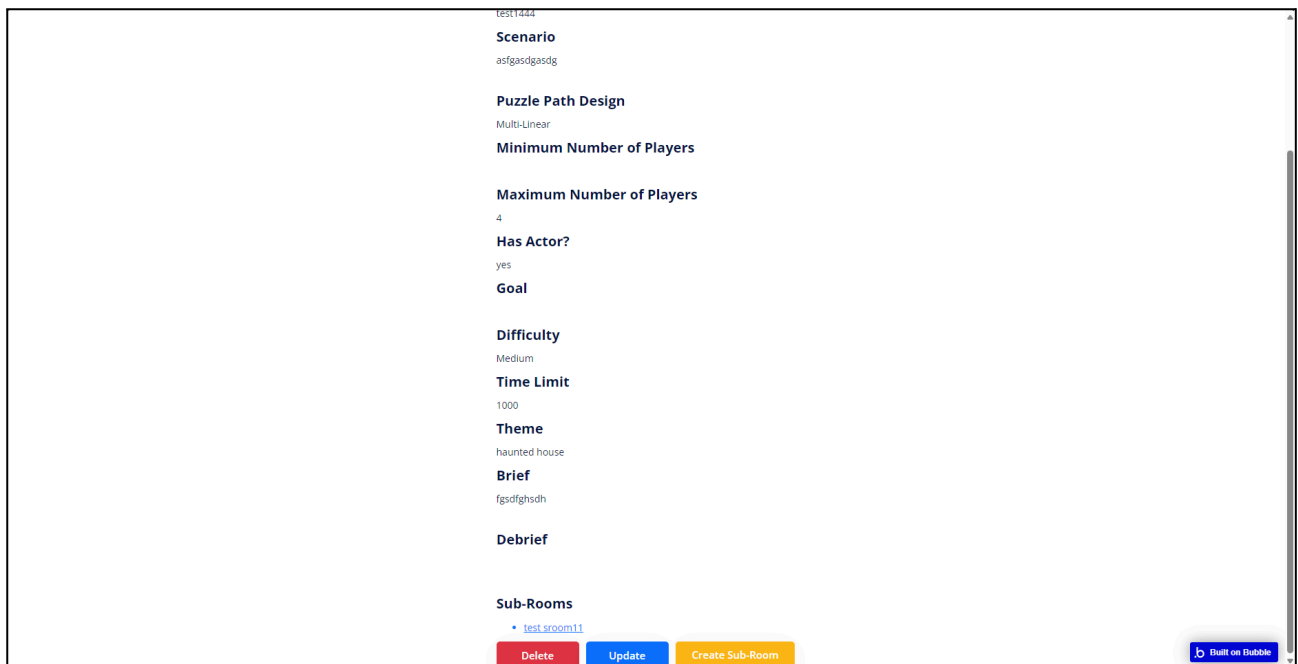
Στις εικόνες Γ'.41 και Γ'.42 είναι όλα τα απαραίτητα στοιχεία για να λειτουργήσει ο μηχανισμός που τοποθετεί δυναμικά, στο τέλος της πρότασης της σελίδας διαγραφής δωματίου, τον τίτλο του σχετικού δωματίου. Για να επιτευχθεί αυτό, πρέπει να προστεθεί κομμάτι με δυναμικά δεδομένα στο τέλος του στοιχείου κειμένου, μετά από το κομμάτι με τα στατικά δεδομένα. Πρέπει επίσης, να γίνει η εξόρυξη των σωστών δεδομένων από τη βάση. Αυτό επιτυγχάνεται με τις παραμετροποιήσεις που φαίνονται στην εικόνα Γ'.42. Γίνεται αναζήτηση για δωμάτια η οποία περιορίζεται πρώτα σε δωμάτια που έχει δημιουργήσει ο τωρινός χρήστης και στη συνέχεια σε δωμάτια όπου ο τίτλος τους είναι ίδιος με αυτόν που βρίσκεται στην παράμετρο "room" του URL. Μιας που ο τίτλος του κάθε δωματίου είναι μοναδικός, στην πραγματικότητα, η παραπάνω αναζήτηση επιστρέφει ένα δωμάτιο του οποίου ο τίτλος παρουσιάζεται και στο τέλος της πρότασης.

Γ'.6.2.8 Προβολή δωματίου

Η σελίδα αυτή παρουσιάζει στο χρήστη τις λεπτομέρειες του κάθε δωματίου. Όταν ο χρήστης επιλέγει ένα δωμάτιο από τη σελίδα προβολής λίστας δωματίων του, καταλήγει εδώ. Μπορεί λοιπόν, να δει τα χαρακτηριστικά ενός δωματίου και τα στοιχεία που έχει συμπληρώσει στο καθένα από αυτά. Αν ο χρήστης δεν έχει συμπληρώσει κάποια πληροφορία σε κάποιο από τα πεδία που αυτό είναι δυνατό, τότε θα δει κενό κάτω από τον αντίστοιχο τίτλο πεδίου.



Εικόνα Γ'.43.α: Σελίδα προβολής λεπτομερειών δωματίου



Εικόνα Γ'.43.β: Σελίδα προβολής λεπτομερειών δωματίου συνέχεια

Εικόνα Γ'.43: Σελίδα προβολής λεπτομερειών δωματίου

Ο δημιουργός του δωματίου "test1444", που φαίνεται στην εικόνα Γ'.43, δεν έχει συμπληρώσει πληροφορίες για τα πεδία "Minimum Number of Players", "Goal" και "Debrief" και ως αποτέλεσμα, κάτω από τον τίτλο αυτών των πεδίων δεν υπάρχει κάποια πληροφορία. Στο κάτω μέρος της σελίδας το μπλε κουμπί με επιγραφή "Update" οδηγεί στη σελίδα επεξεργασίας δωματίου, το κόκκινο κουμπί με επιγραφή "Delete" οδηγεί στην σελίδα διαγραφής, ενώ το κίτρινο κουμπί με επιγραφή "Create Sub-Room" οδηγεί στη σελίδα δημιουργίας υποδωματίου.

Γ'.6.2.9 Επεξεργασία δωματίου

Η σελίδα επεξεργασίας δωματίου επιτρέπει στο χρήστη να αλλάξει πληροφορίες ενός δωματίου τις οποίες έχει ορίσει στο παρελθόν ή και να προσθέσει πληροφορίες σε χαρακτηριστικά που δεν έχουν ακόμη.

EscapeRoomCentral

Projects Logout Create Room

Update Room

test1444

Title
test1444

Scenario
asfgasdgasdg

Puzzle Path Design
Multi-Linear

Minimum Number Of Players
Minimum Number Of Players

Maximum Number Of Players
4

☒ Has Actor?

Goal
Goal

Difficulty
Medium

Built on Bubble

Εικόνα Γ'.44.α: Σελίδα επεξεργασίας δωματίου

Minimum Number Of Players
Minimum Number Of Players

Maximum Number Of Players
4

☒ Has Actor?

Goal
Goal

Difficulty
Medium

Time Limit
1000

Theme
haunted house

Brief
fgsdfghsdh

Debrief
Debrief

Update Room!

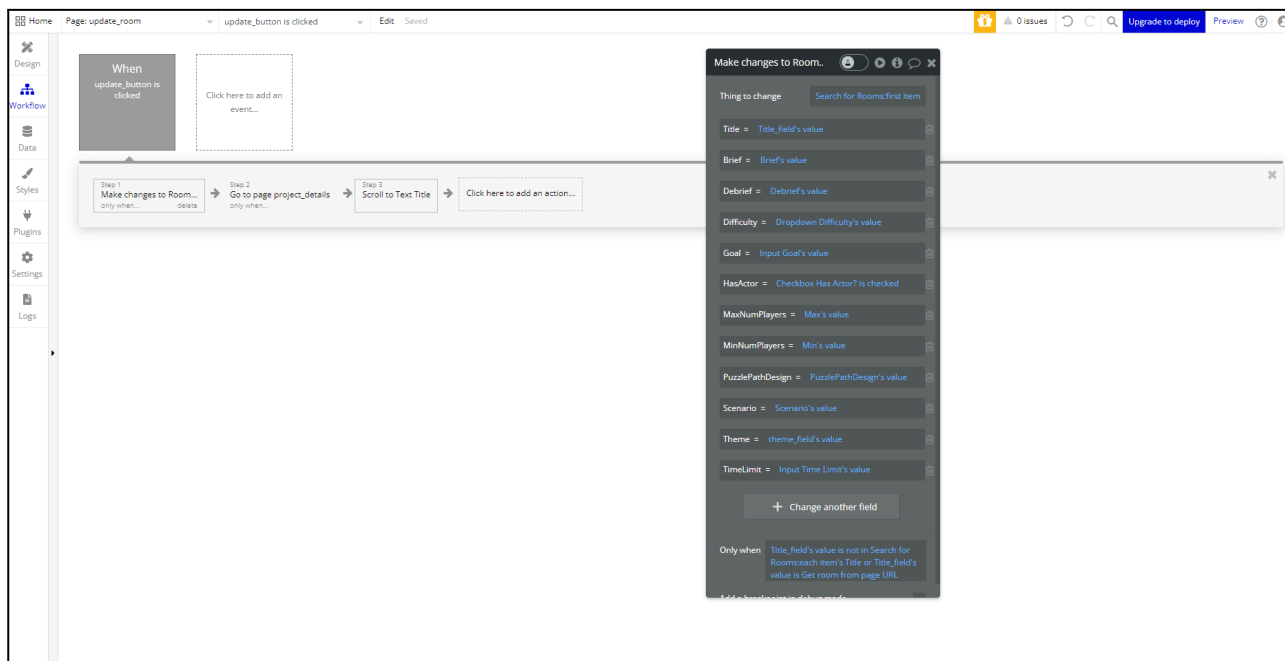
Built on Bubble

Εικόνα Γ'.44.β: Σελίδα επεξεργασίας δωματίου

Εικόνα Γ'.44.: Σελίδα επεξεργασίας δωματίου

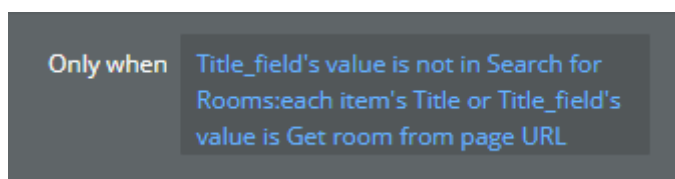
Στη σελίδα επεξεργασίας δωματίου, παρουσιάζεται ουσιαστικά στο χρήστη η φόρμα δημιουργίας δωματίου, αλλά αυτή τη φορά με ήδη συμπληρωμένα τα στοιχεία που έχει βάλει στο παρελθόν. Με το πάτημα του γαλάζιου κουμπιού με επιγραφή “Update Room!” τα στοιχεία που υπάρχουν στα πεδία αποθηκεύονται ενώ τα παλιά σβήνονται, ενημερώνοντας έτσι το δωμάτιο. Εδώ το Bubble αναλαμβάνει και πάλι την υλοποίηση αυτών των ενεργειών αφήνοντας στο δημιουργό, απλώς να δώσει την οδηγία στο Bubble να κάνει τις αλλαγές. Αναλυτικότερα, πρέπει να οριστεί μια ακολουθία ενεργειών οι οποίες προσφέρονται

από το Bubble αλλά χρειάζονται παραμετροποίηση όπως φαίνεται στην εικόνα Γ'.45.



Εικόνα Γ'.45: Σειρά ενεργειών για την επεξεργασία δωματίου

Η ενέργειες που θα πρέπει να ακολουθηθούν είναι η επεξεργασία των στοιχείων του δωματίου, η μεταφορά στη σελίδα προβολής λεπτομερειών του σχετικού δωματίου και τέλος η κύλιση στο πεδίο του τίτλου. Οι ενέργειες αυτές πραγματοποιούνται μόνο αν ικανοποιούνται συγκεκριμένες συνθήκες που μπορεί να ορίσει ο προγραμματιστής. Στη συγκεκριμένη περίπτωση, η συνθήκη αυτή είναι η μοναδικότητα του τίτλου του δωματίου. Στη συνέχεια, η συνθήκη αυτή και η υλοποίησή της, παρουσιάζονται αναλυτικότερα. Η ενέργεια επεξεργασίας του δωματίου είναι αυτή που πρέπει να παραμετροποιηθεί περισσότερο. Πρέπει να οριστούν τα πεδία που θα αλλάξουν και από που θα πάρει το καθένα τα σωστά δεδομένα. Έτσι εδώ υπάρχουν όλα τα χαρακτηριστικά του δωματίου και η κατάλληλη παραμετροποίηση για να δεχθούν τα δεδομένα από το αντίστοιχο πεδίο της φόρμας.



Εικόνα Γ'.46.α: Ολόκληρη η έκφραση της συνθήκης

Εικόνα Γ'.46.β: Παραμετροποίηση του "Search for Rooms"

Εικόνα Γ'.46.γ: Παραμετροποίηση του "Get value from page URL"

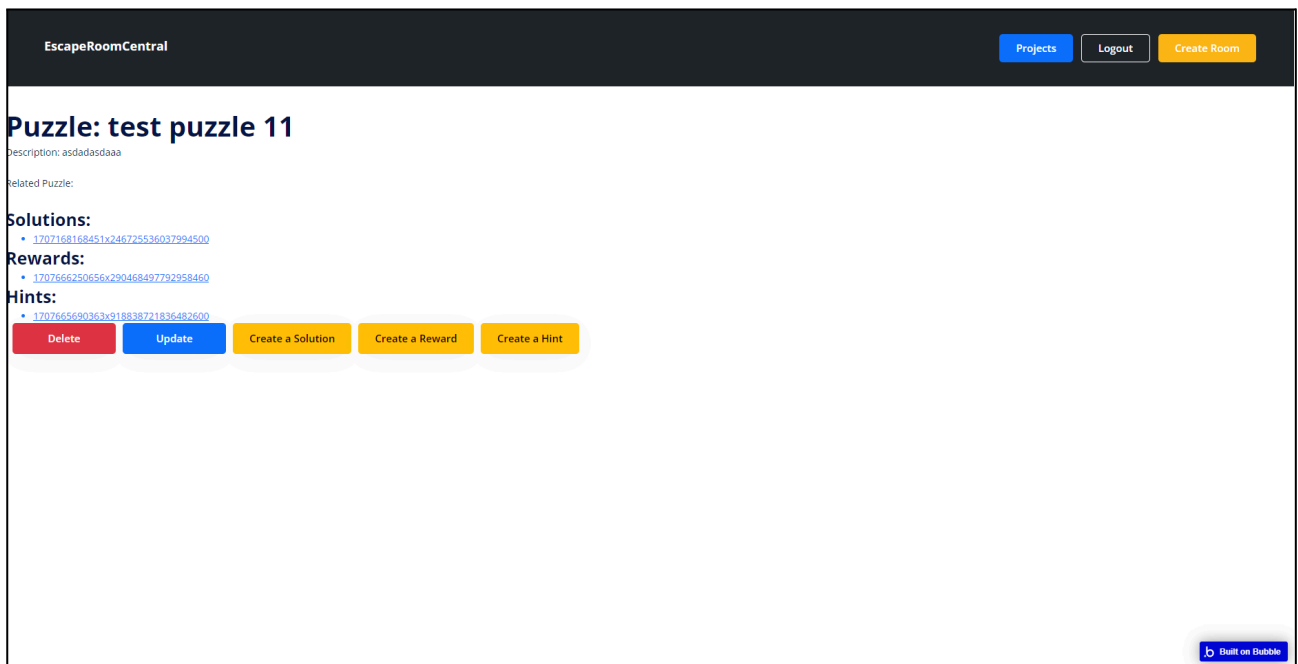
Εικόνα Γ'.46: Προγραμματισμός συνθήκης της ενέργειας επεξεργασίας δωματίου

Συγκεκριμένα, η συνθήκη επιτρέπει την επεξεργασία ενός δωματίου μόνο αν ο τίτλος αυτού είναι μοναδικός ή είναι ίδιος με αυτόν στην παράμετρο "room" του URL. Έτσι εξασφαλίζεται η μοναδικότητα του τίτλου σε περίπτωση που αυτός αλλάξει, αλλά και επιτρέπεται η διατήρηση του ίδιου τίτλου. Σε περίπτωση που το δεύτερο μέρος της συνθήκης δεν υπήρχε, τότε ο τελικός χρήστης δε θα μπορούσε να διατηρήσει τον ίδιο τίτλο καθώς το πρώτο μέρος της συνθήκης είναι αληθές.

Γ'.6.2.10 Υπόλοιπες σελίδες

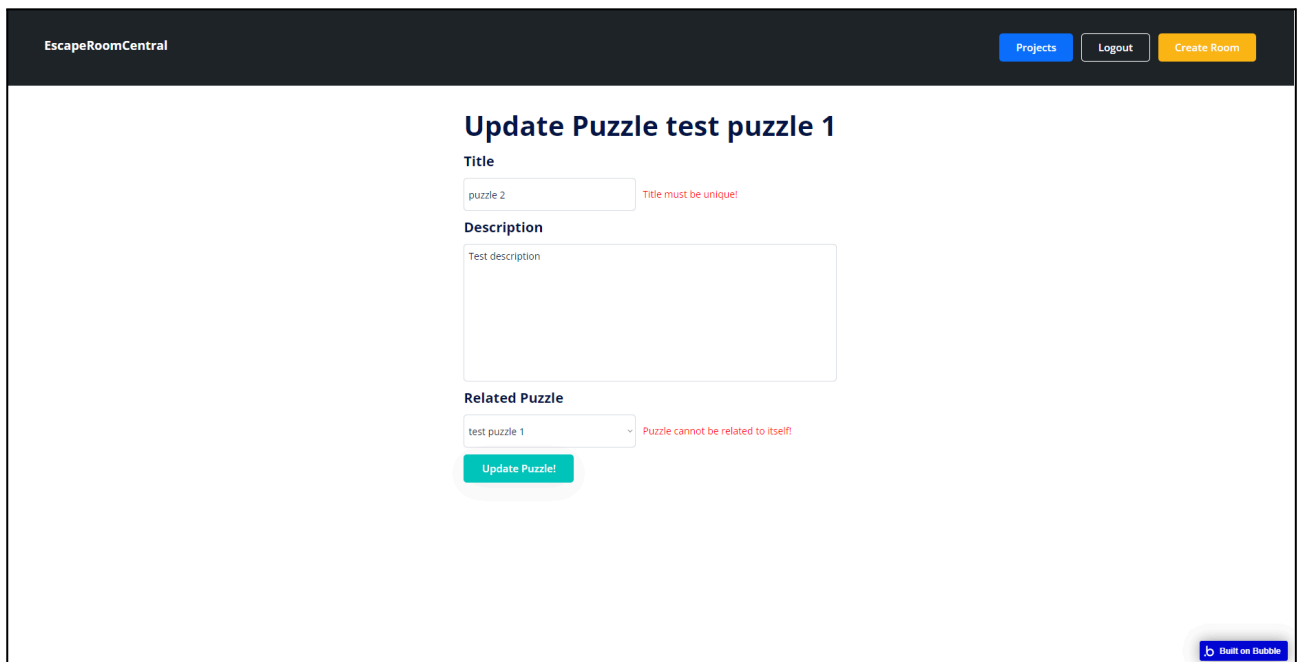
Όπως αναφέρεται και στην υποενότητα Γ'.6.1.10, οι σελίδες των υπολοίπων οντοτήτων που ακολουθούν το ίδιο μοτίβο με τις σελίδες του δωματίου. Έχουν δηλαδή σελίδα δημιουργίας, επεξεργασίας, προβολής και διαγραφής. Έτσι και εδώ θα παρουσιαστούν επιλεκτικά κάποιες από αυτές που έχουν αξιοσημείωτες ιδιαιτερότητες. Μιας και οι δύο σελίδες προσπαθούν να είναι κλώνοι με τη μόνη τους

διαφορά να είναι το εργαλείο που χρησιμοποιήθηκε για τη δημιουργία τους και σε αυτή την υποενότητα θα παρουσιαστούν σελίδες που είναι σχετικές με την οντότητα "παζλ".



Εικόνα Γ'.47: Σελίδα προβολής λεπτομερειών παζλ

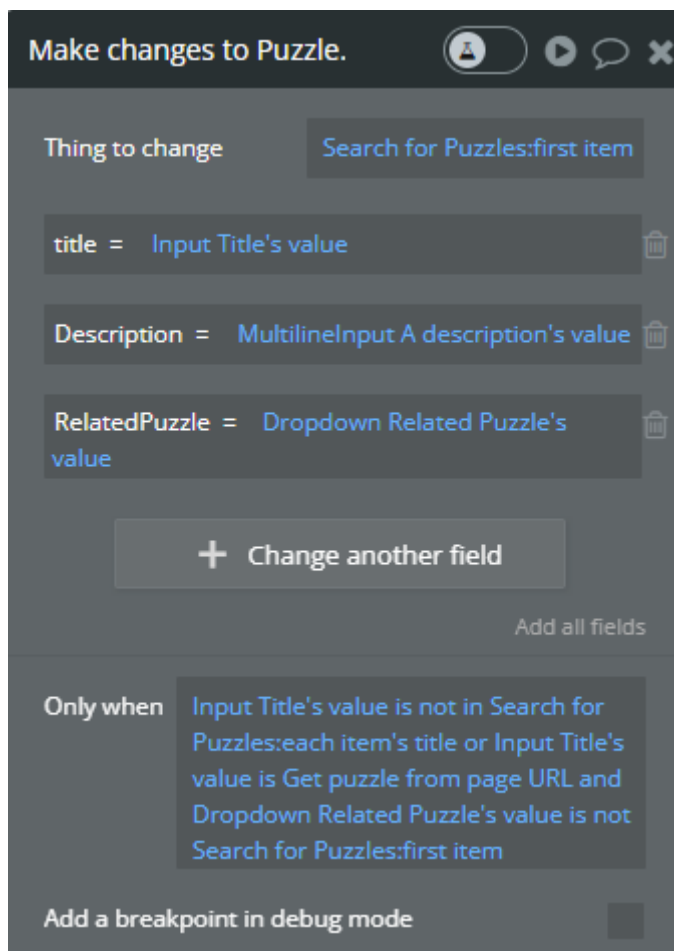
Αντίστοιχα και με τη σελίδα προβολής λεπτομερειών παζλ που αναπτύχθηκε με το Django, η ιδιαιτερότητα που παρουσιάζει αυτή η σελίδα είναι στο γεγονός πως έχει τρεις λίστες με οντότητες-παιδιά και πως έχει το πεδίο "Related Puzzle" που εξυπηρετεί το σκοπό της δημιουργίας σχεδίασης ακολουθίας παζλ μέσω μιας αναδρομικής συσχέτισης στη βάση δεδομένων. Στη σελίδα αυτή φαίνεται και η μορφή που έχει το "UniqueID" που χρησιμοποιεί το Bubble για τη διάκριση των στοιχείων της βάσης δεδομένων.



Εικόνα Γ'.48: Επεξεργασία παζλ

Η σελίδα επεξεργασίας του παζλ παρουσιάζει επίσης ιδιαιτερότητα αλλά και δυσκολία. Έχει δύο συνθήκες που θα πρέπει να ικανοποιούνται για να είναι επιτρεπτή η επεξεργασία. Στην εικόνα Γ'.48 οι δ'θο

αυτές συνθήκες έχουν εσκεμμένα παραβιαστεί για να αναδειχτούν. Είναι λοιπόν η μοναδικότητα του τίτλου που πρέπει να εξασφαλιστεί αλλά και η αποφυγή συσχέτισης ενός παζλ με τον εαυτό του. Αυτό είναι απαραίτητο καθώς κάτι τέτοιο θα οδηγούσε σε βρόχο στη σχεδίαση ακολουθίας παζλ.



Εικόνα Γ'.49: Παραμετροποίηση ενέργειας "Make changes to Puzzle"

Στην εικόνα Γ'.49 φαίνεται η παραμετροποίηση της ενέργειας "Make changes to Puzzle" ώστε αυτή να λαμβάνει χώρα μόνο όταν ο τίτλος του παζλ είναι μοναδικός και το παζλ δεν σχετίζεται με τον εαυτό του. Για να δημιουργηθεί η συγκεκριμένη έκφραση, κομμάτια της χρειάζονται περαιτέρω παραμετροποίηση όπως έχει παρουσιαστεί σε προηγούμενα παραδείγματα. Εδώ βρίσκεται άλλη μια διαφορά στην υλοποίηση της ίδιας λειτουργίας με διαφορετικό τρόπο με το κάθε εργαλείο. Στο Bubble, η λειτουργία βασίζεται στο URL που περιέχει τον τίτλο του παζλ σαν παράμετρο. Έτσι, συγκρίνει την πληροφορία αυτή, με την είσοδο του "RelatedPuzzle". Αν οι δύο τιμές είναι ίδιες, δεν επιτρέπεται η καταχώρηση της φόρμας. Το Django αντίστοιχα περιέχει το πρωτεύον κλειδί στο URL, άρα η ίδια λειτουργία δεν μπορεί να υλοποιηθεί με τον ίδιο τρόπο. Σε εκείνη την περίπτωση γίνεται χρήση του "self.instance" για να συγκριθεί με την επιλογή στο πεδίο "RelatedPuzzle". Αν τα δύο αυτά είναι ίδια, δεν επιτρέπεται η καταχώρηση της φόρμας.

Κεφάλαιο Δ'

Δ' Πείραμα

Στο κεφάλαιο του Δ', αρχικά γίνεται μια αναλυτική περιγραφή του πειράματος, της οποίας σκοπός είναι να επιτρέψει στον αναγνώστη να επαναλάβει το πείραμα χωρίς να χρειαστεί έρευνα για περαιτέρω εξωτερική πληροφορία. Επιπλέον, ο αναγνώστης θα μπορεί να είναι σε θέση να καταλήξει στα ίδια αποτελέσματα με αυτά που θα παρουσιαστούν σε αυτό το κεφάλαιο, χωρίς να αντιμετωπίσει κάποιο εμπόδιο. Στη συνέχεια, παρουσιάζονται τα δεδομένα που παρήχθησαν από την διεξαγωγή αυτού του πειράματος. Τα δεδομένα αυτά αφορούν την διαδικασία ανάπτυξης αλλά και το αποτέλεσμα που προέκυψε, από τα δύο διαφορετικά εργαλεία ανάπτυξης. Ακολουθεί η σύγκριση των δεδομένων που προήλθαν από τους δύο διαφορετικούς τρόπους ανάπτυξης, ώστε να εμφανιστούν οποιαδήποτε σημεία διαφοροποίησης, από τα οποία μπορούν να εξαχθούν συμπεράσματα για την χρησιμότητα και την αξία του καθενός. Τέλος, παρουσιάζονται τυχόν περιορισμοί που μπορεί να υπήρχαν σε οποιοδήποτε σημείο του πειράματος. Αυτό σημαίνει πως, σε αυτή την υποενότητα, θα παρουσιαστούν σημεία του πειράματος τα οποία θα μπορούσαν να βελτιωθούν.

Δ'.1 Περιγραφή Πειράματος

Το πείραμα της συγκεκριμένης εργασίας είναι η σύγκριση των δύο τρόπων ανάπτυξης που χρησιμοποιήθηκαν για την δημιουργία μιας συγκεκριμένης και ίδιας εφαρμογής. Σημαντικό είναι να αναφερθεί και η αρχική υπόθεση, την οποία προσπαθεί να επιβεβαιώσει ή να απορρίψει το παρόν πείραμα. Η υπόθεση αυτή υποστηρίζει πως τα no/low code εργαλεία είναι καταλληλότερα για μια νεοσύστατη εταιρεία, επειδή της επιτρέπουν να δημιουργήσει ένα ελάχιστο βιώσιμο προϊόν γρηγορότερα και της δίνουν πρόσβαση σε μεγαλύτερο σύνολο υποψηφίων εργαζομένων, καθώς η εξειδίκευση που απαιτούν είναι λιγότερη. Από την άλλη, τα εργαλεία κώδικα προσφέρουν περισσότερες δυνατότητες αλλά και επιτρέπουν στο παραγόμενο προϊόν να κλιμακωθεί ανάλογα με τις απαιτήσεις της αναπτυσσόμενης εταιρείας. Μειονέκτημα αυτής της μεθόδου ανάπτυξης είναι ότι η εταιρία θα χρειαστεί να προσλάβει προσωπικό που είναι περισσότερο εξειδικευμένο.

Η ανάπτυξη της εφαρμογής με τα εργαλεία του κώδικα και το εργαλείο "no/low code" θα συγκριθούν σε έναν αριθμό ποιοτικών και ποσοτικών χαρακτηριστικών που σχετίζονται με την ανάπτυξη της εφαρμογής, με τον εκάστοτε τρόπο, αλλά και με το τελικό αποτέλεσμα. Η ανάπτυξη των δύο εφαρμογών έγινε από ένα δημιουργό που δεν είχε προϋπάρχουσα γνώση των εργαλείων. Τα εργαλεία που χρησιμοποιήθηκαν για την ανάπτυξη της εφαρμογής ήταν, για αυτήν που αναπτύχθηκε με κώδικα, HTML και Bootstrap για το κομμάτι της γραφικής διεπαφής του χρήστη στο διαδίκτυο, δηλαδή το σημείο αλληλεπίδρασης του με την εφαρμογή. Για τον προγραμματισμό της λογικής που έπρεπε να ακολουθεί η εφαρμογή, δηλαδή για την δημιουργία των διαφόρων λειτουργιών, χρησιμοποιήθηκε το Django. Πιο αναλυτικά, HTML χρησιμοποιήθηκε για την δημιουργία του βασικού περιεχομένου της σελίδας, όπως το κείμενο, τους υπερσυνδέσμους και άλλα. Ενώ το Bootstrap, για τον σχεδιασμό της σελίδας σε ότι αφορά τα χρώματα, την δομή και τα σχήματα. Το Django, αναλαμβάνει τη δημιουργία και διαχείριση της βάσης

δεδομένων, την οργάνωση των αρχείων και φακέλων του κώδικα και όλα τα υπόλοιπα αναγκαία στοιχεία μιας διαδικτυακής εφαρμογής όπως, τη δημιουργία URL, τον προγραμματισμό της λογικής και λειτουργικότητας, τη δημιουργία φορμών, τη διαχείριση των λογαριασμών χρηστών και άλλα. Για την εφαρμογή που αναπτύχθηκε χωρίς κώδικα το εργαλείο που επιλέχθηκε ήταν το Bubble, το οποίο χρησιμοποιήθηκε για την ανάπτυξη της γραφικής διεπαφής αλλά και της λογικής της εφαρμογής. Οι δύο εφαρμογές αναπτύχθηκαν με σκοπό να είναι ίδιες τόσο στο κομμάτι της γραφικής διεπαφής του χρήστη όσο και στην λογική τους. Οποιοσδήποτε διαφορές προέκυψαν μεταξύ των δύο εφαρμογών ήταν αποτέλεσμα της διαφοράς φιλοσοφίας των δύο εργαλείων και τρόπων ανάπτυξης.

Τα χαρακτηριστικά στα οποία θα συγκριθούν οι δύο τρόποι ανάπτυξης φαίνονται στη λίστα Δ'1 που ακολουθεί.

- Χρόνος ανάπτυξης της εφαρμογής.
- Απόκλιση αποτελέσματος από το επιθυμητό.
- Ποιοτικά δεδομένα.
 - Εμπειρία ανάπτυξης της εφαρμογής.
 - Υποστήριξη από την κοινότητα/Υπαρξη διαδικτυακών forum.
- Υποστήριξη λειτουργιών από τα εργαλεία.
- Κόστος ανάπτυξης.

Λίστα Δ'1

Για τα παραπάνω χαρακτηριστικά θα παρουσιαστούν δεδομένα για τους δύο τρόπους ανάπτυξης. Στο χρόνο ανάπτυξης περιλαμβάνεται και ο χρόνος που ενδεχομένως χρειάστηκε για την εκμάθηση των εργαλείων που χρησιμοποιήθηκαν στον κάθε τρόπο ανάπτυξης. Το χαρακτηριστικό "απόκλιση αποτελέσματος από το επιθυμητό" αναφέρεται σε τυχόν διαφορές που είχε η εφαρμογή, που τελικά αναπτύχθηκε, σε σχέση με την αρχική πρόθεση του δημιουργού. Στα στοιχεία υποκειμενικού χαρακτήρα θα παρουσιαστούν οι σχετικές διαφορές των εργαλείων, ποιοτικά χαρακτηριστικά που μετρήθηκαν με τη χρήση ερωτηματολογίου και η προσωπική άποψη του δημιουργού επάνω σε αυτές. Στην υποστήριξη των λειτουργιών θα συγκριθούν ενδεχόμενες ελλείψεις λειτουργιών που ήταν αναγκαίες ή επιθυμητές από το δημιουργό. Τέλος, στο κόστος ανάπτυξης θα παρουσιαστούν τα κόστη που υπήρχαν κατά την διάρκεια ανάπτυξης των εφαρμογών αλλά και τα κόστη που είναι πιθανό να υπάρχουν υπό διαφορετικές συνθήκες ανάπτυξης.

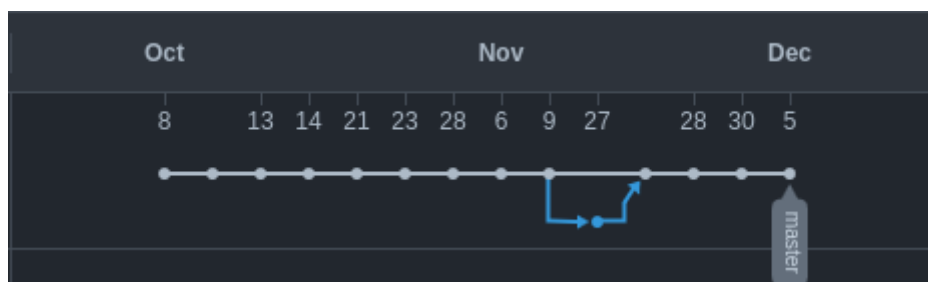
Δ'2 Δεδομένα

Σε αυτή την υποενότητα, παρουσιάζονται τα δεδομένα που συγκεντρώθηκαν για κάθε έναν από τους άξονες σύγκρισης των δύο εργαλείων.

Δ'.2.1 Χρόνος ανάπτυξης της εφαρμογής από το δημιουργό

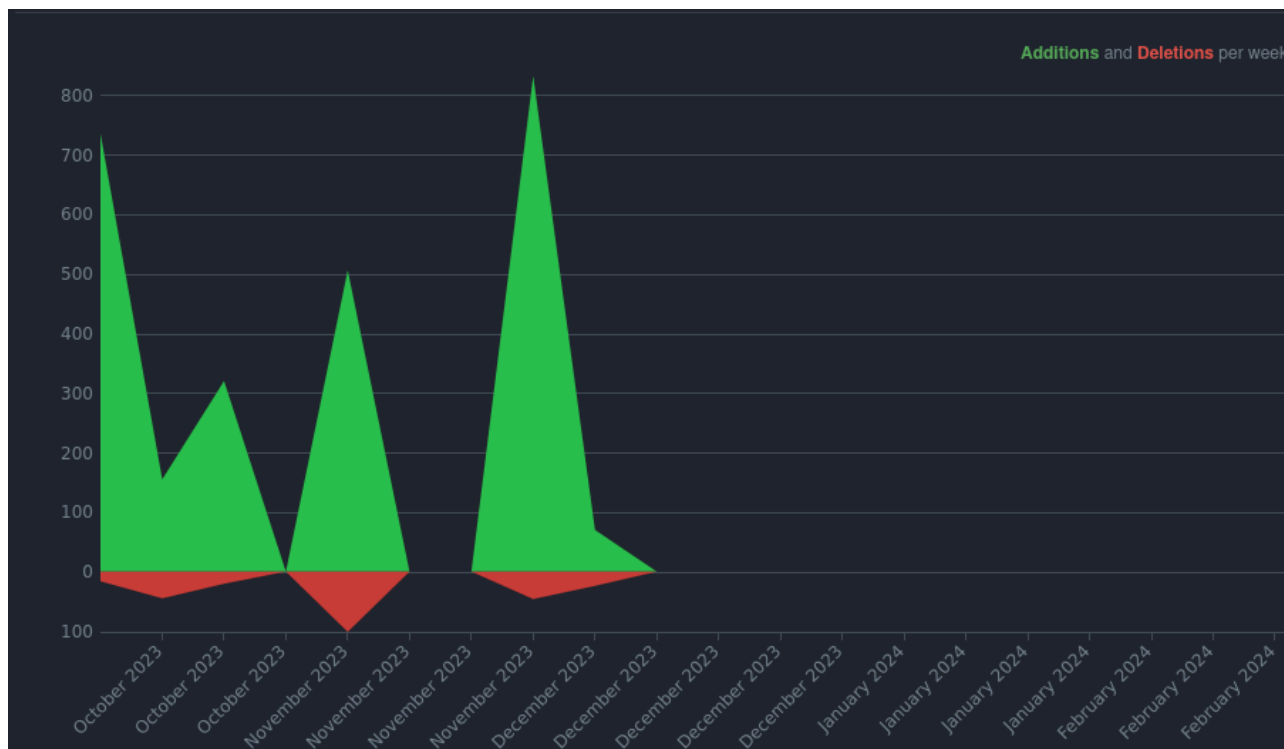
Τα δεδομένα για το χρόνο ανάπτυξης παρουσιάζονται σαν διάστημα από ημερομηνίες της πρώτης και της τελευταίας συνεισφοράς που έγινε με το κάθε εργαλείο. Για το Django, τις ημερομηνίες αυτές θα δώσει το GitHub. Το Git και GitHub αξιοποιήθηκαν από το δημιουργό κατά τη διαδικασία ανάπτυξης της σελίδας με το Django σαν εργαλεία διαχείρισης του αποθετηρίου του κώδικα, αλλά και σαν ένα ασφαλές μέρος φύλαξής του. Για το Bubble η ημερομηνία έναρξης έχει κρατηθεί χειροκίνητα, ενώ η ημερομηνία της τελευταίας συνεισφοράς δίνεται στη σελίδα των εφαρμογών του Bubble.

Για την ιστοσελίδα που αναπτύχθηκε με το Django, η πρώτη συνεισφορά έγινε στις 08/10/2023 και η τελευταία στις 5/12/2023. Ένα διάστημα 59 ημερών. Το GitHub προσφέρει και διαγράμματα που βοηθούν στην οπτικοποίηση των συνεισφορών.



Εικόνα Δ'.1: Διάγραμμα με ημερομηνίες συνεισφορών.

Επειδή η πληροφορία αυτή δεν είναι πολύ χρήσιμη από μόνη της, στην εικόνα Δ'.2 παρουσιάζονται αναλυτικότερα η συνεισφορές με βάση τις προσθήσεις και αφαιρέσεις γραμμών κώδικα.



Εικόνα Δ'.2: Διάγραμμα με προσθήσεις και αφαιρέσεις γραμμών κώδικα ανά εβδομάδα.

Από το παραπάνω διάγραμμα και τα αναλυτικότερα δεδομένα της κάθε συνεισφοράς, προκύπτει πως έγιναν 2.871 αλλαγές. Είτε αυτές ήταν προσθήσεις είτε αφαιρέσεις γραμμών. Κατα μέσο όρο λοιπόν, έγιναν 48,66

αλλαγές τη μέρα. Αυτές αναλύονται σε 2.619 προσθήσεις και 252 αφαιρέσεις. Κατά μέσο όρο 44,38 προσθήσεις και 4,27 αφαιρέσεις τη μέρα. Αν συνδυαστούν οι παραπάνω πληροφορίες, μπορεί να φτιαχτεί μια καλύτερη εικόνα για το χρόνο που χρειάστηκε για την κατασκευή της σελίδας. Αν οι συνεισφορές ανά ημέρα ήταν περισσότερες, το τελικό αποτέλεσμα θα κατασκευαζόταν νωρίτερα. Αντίστοιχα, αν υπήρχαν λιγότερες συνεισφορές ανά ημέρα, θα χρειαζόταν περισσότερος χρόνος για το ίδιο τελικό αποτέλεσμα.

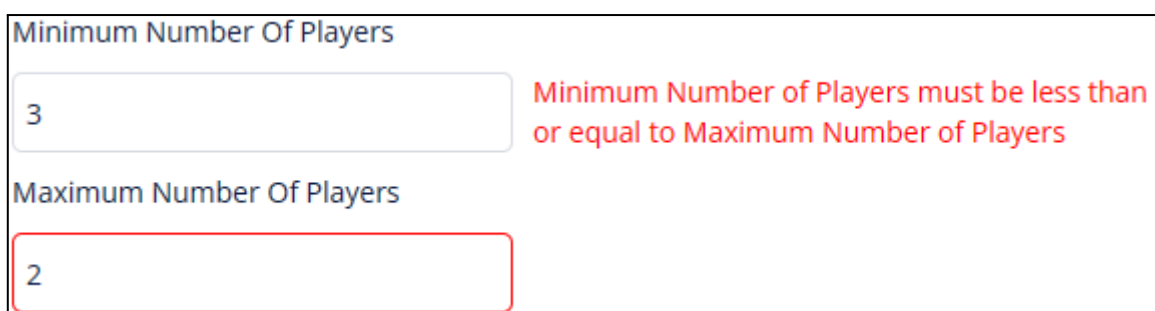
Για την ιστοσελίδα που αναπτύχθηκε με το Bubble το διάστημα είναι 25/01/2024 με 15/02/2024. Το διάστημα είναι 22 ημέρες. Για αυτή την ιστοσελίδα, δεν υπάρχουν τα παραπάνω διαγράμματα για τις συνεισφορές ανά ημέρα. Δεν προσφέρονται από το Bubble, ενώ δεν υπάρχει και κάποιος τρόπος να συνδεθεί κάποια εξωτερική υπηρεσία που να τα προσφέρει.

Τέλος, πρέπει να ληφθεί υπόψη και ο χρόνος που χρειάστηκε για την εκμάθηση του κάθε εργαλείου. Για το Django αυτή έγινε μέσω μαθήματος που προσφέρεται σε μορφή βίντεο από την υπηρεσία Udemy. Για το Bubble δεν υπήρξε τέτοια διαδικασία. Για το Django η διάρκεια του μαθήματος αυτού είναι 32 ώρες.

Δ'.2.2 Απόκλιση αποτελέσματος από το επιθυμητό

Σε αυτή την υποενότητα παρουσιάζονται παραδείγματα από τις δύο σελίδες όπου δεν ήταν δυνατή η ακριβής υλοποίηση του επιθυμητού. Αυτό μπορεί να συνέβει λόγω έλλειψης κάποιας λειτουργίας από ένα ή και τα δύο από τα εργαλεία ή αδυναμία υλοποίησης της από το δημιουργό.

Μία λειτουργία που έχει υλοποιηθεί στο Bubble αλλά όχι στο Django, είναι ο έλεγχος του ελάχιστου αριθμού παικτών που θα πρέπει να είναι μικρότερος ή ίσως από αυτό του μέγιστου αριθμού παικτών. Στο Bubble η δημιουργία ή επεξεργασία ενός δωματίου έχει προγραμματιστεί να λαμβάνει χώρα μόνο αν το πεδίο του ελάχιστου αριθμού παικτών περιέχει τιμή μικρότερη ή ίση με αυτή από το πεδίο του μέγιστου αριθμού παικτών. Αυτό υλοποιείται στην καστέλα "Workflow" όπου η αντίστοιχη λειτουργία ρυθμίζεται να πραγματοποιείται μόνο όταν πληρούνται η συγκεκριμένη προϋπόθεση. Παρομοίως, με αντίστοιχη ρύθμιση, εμφανίζεται κατάλληλο μήνυμα στη φόρμα για να ενημερώσει το χρήστη πως η είσοδός του είναι λανθασμένη όπως φαίνεται στην εικόνα Δ'.3.



Minimum Number Of Players

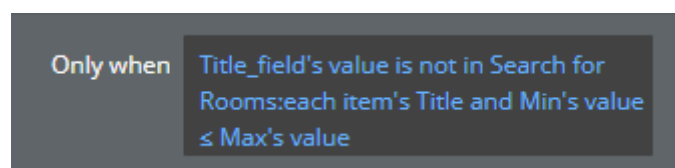
3

Maximum Number Of Players

2

Minimum Number of Players must be less than or equal to Maximum Number of Players

Εικόνα Δ'.3: Μήνυμα σφάλματος στη φόρμα.



Only when Title_field's value is not in Search for Rooms:each item's Title and Min's value ≤ Max's value

Εικόνα Δ'.4: Περιορισμός εκτέλεσης ενέργειας δημιουργίας/επεξεργασίας δωματίου.

Η ίδια λειτουργία δεν έχει υλοποιηθεί στο Django. Αυτό συνέβει λόγω του υψηλού βαθμού δυσκολίας της υλοποίησης της συγκεκριμένης λειτουργίας στο Django. Σε αυτό θα πρέπει να δημιουργηθεί μια νέα συνάρτηση στην κατάλληλη κλάση του views.py που κάνει τον παραπάνω έλεγχο. Αν και η λογική παραμένει ίδια, η υλοποίηση της στο Django γίνεται με κώδικα. Έτσι προκύπτουν ερωτήματα για το πως θα ανακτηθούν τα απαραίτητα δεδομένα για να γίνει η σύγκριση.

Μία λειτουργία που έχει υλοποιηθεί στο Django αλλά όχι, όπως ήταν επιθυμητό, στο Bubble είναι το χαρακτηριστικό του χρονικού ορίου. Αυτό βρίσκεται στην οντότητα του δωματίου και καθορίζει το μέγιστο χρονικό όριο που έχουν οι παίκτες του escape room για να φτάσουν στο στόχο τους. Αν υπερβούν αυτό το όριο χωρίς να έχουν φτάσει στο στόχο τους, χάνουν. Έτσι, για τις ανάγκες αυτού του χαρακτηριστικού, στο Django, έχει χρησιμοποιηθεί ο τύπος δεδομένων "DurationField". Το Bubble προσφέρει μικρότερη ποικιλία τύπων δεδομένων. Επομένως, δεν υπάρχει κάποιος τύπος που να καλύπτει τις απαιτήσεις του συγκεκριμένου χαρακτηριστικού. Ως αποτέλεσμα έχει γίνει χρήση του τύπου δεδομένων "number" για το συγκεκριμένο πεδίο. Ο τύπος δεδομένων αυτός δέχεται αριθμούς, οπότε δεν έχει κάποια αυτόματη μορφοποίηση ώρας (π.χ: 22:13:56). Από την άλλη, ο τύπος του Django προσφέρει αυτόματη μορφοποίηση αλλά και μετατροπή της εισόδου του χρήστη με μορφοποίηση. Όταν λοιπόν κάποιος χρήστης εισάγει τον αριθμό 600 στην εφαρμογή που αναπτύχθηκε με το Django, θα βλέπει 00:10:00. Το Django δηλαδή έχει καταλάβει πως το 600 είναι δευτερόλεπτα και το έχει μορφοποιήσει έτσι ώστε να φαίνεται σαν 10 λεπτά. Στο αντίστοιχο παράδειγμα στο Bubble, ο χρήστης θα έβλεπε απλά τον αριθμό 600.

Δ'.2.3 Ποιοτικά στοιχεία υποκειμενικού χαρακτήρα.

Σε αυτή την υποενότητα παρουσιάζονται μερικά ποιοτικά χαρακτηριστικά τα οποία δεν είναι μετρήσιμα με τη χρήση εργαλείων. Για τη μέτρηση τους έχουν γίνει μια σειρά από συνεντεύξεις. Στις συνεντεύξεις αυτές, οι συμμετέχοντες, κλήθηκαν να ολοκληρώσουν μια άσκηση με την χρήση των δύο εργαλείων και στη συνέχεια να απαντήσουν μερικές ερωτήσεις με βάση την εμπειρία που είχαν κατά την επίλυση της άσκησης.

Η άσκηση που έπρεπε να λύσουν οι συμμετέχοντες ήταν η υλοποίηση μιας από τις σελίδες που παρουσιάστηκαν στην ενότητα Γ'. Συγκεκριμένα, η σελίδα αυτή ήταν η σελίδα επεξεργασίας παζλ. Η σελίδα αυτή επιλέχθηκε καθώς απαιτεί την ενασχόληση με όλα τα βασικά αρχεία του Django και αντίστοιχα, τις βασικές καρτέλες του Bubble. Είναι μια σελίδα που χρειάζεται προγραμματισμό της βάσης δεδομένων, καθώς πρέπει να δημιουργηθεί η οντότητα "παζλ". Προγραμματισμό της γραφικής διεπαφής, καθώς πρέπει να δημιουργηθεί η σελίδα αλλά και η φόρμα που θα συμπληρώσει ο τελικός χρήστης. Τέλος, απαιτείται προγραμματισμός της λογικής της διαχείρισης των αιτημάτων HTTP. Όλες αυτές οι απαιτήσεις σε συνδυασμό με μερικούς επιπλέον περιορισμούς που χρειάζεται αυτή η σελίδα, κάνουν την υλοποίηση της μια καλή άσκηση ώστε να δει κανείς πολλές από τις πτυχές και των δύο εργαλείων. Ταυτόχρονα, ο όγκος κώδικα και ο χρόνος που χρειάζεται για τη δημιουργία αυτής της σελίδας την κατατάσσουν στο μέσο όρο των σελίδων.

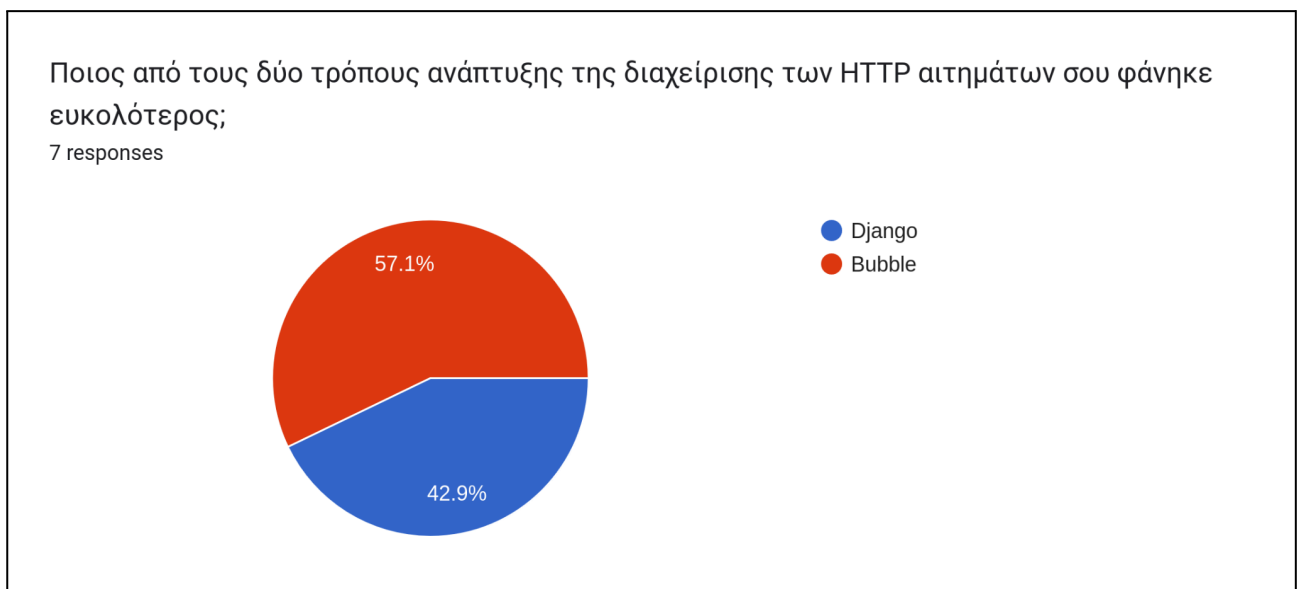
Στο σύνολο, πραγματοποιήθηκαν 7 συνεντεύξεις διάρκειας περίπου 2 ωρών η κάθε μια. Συμπληρώθηκαν ισάριθμα ερωτηματολόγια των οποίων τα δεδομένα παρουσιάζονται παρακάτω. Επιπλέον οι απαντήσεις στις ερωτήσεις δεν περιορίζονται σε αυτές από τα ερωτηματολόγια. Για την κάθε απάντηση

έχουν δοθεί παραπάνω πληροφορίες σε μορφή ηχογράφησης. Μερικές από τις ηχογραφημένες απαντήσεις, που περιέχουν περισσότερα συμφραζόμενα, θα παρατεθούν επίσης συμπληρωματικά των απαντήσεων από το ερωτηματολόγιο.

Στην ερώτηση "Ποιος από τους δύο τρόπους ανάπτυξης της γραφικής διεπαφής σου φάνηκε ευκολότερος;" το 85.7% απάντησε πως πιο εύκολος τρόπος ανάπτυξης της γραφικής διεπαφής τους φάνηκε αυτός που προσφέρει το Bubble. Ενώ το 14.3% προτίμησε τον τρόπο ανάπτυξης γραφικής διεπαφής που προσφέρει το Django.

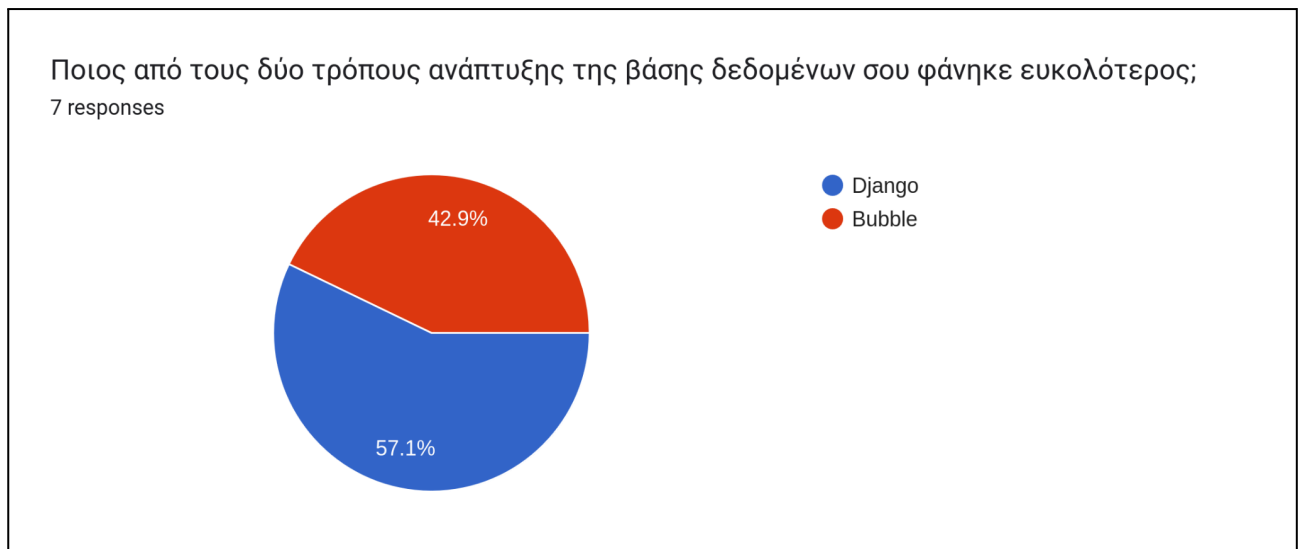


Εικόνα Δ'.5: Πίτα για την ερώτηση "Ποιος από τους δύο τρόπους ανάπτυξης της γραφικής διεπαφής σου φάνηκε ευκολότερος;" Η επόμενη ερώτηση είναι "Ποιος από τους δύο τρόπους ανάπτυξης της διαχείρισης των HTTP αιτημάτων σου φάνηκε ευκολότερος;". Σε αυτή, το 57.1% επέλεξε το Bubble, ενώ το 42.9% επέλεξε το Django. Εδώ παρατηρείται μια αύξηση στο ποσοστό των απαντήσεων που προτίμησαν το Django.



Εικόνα Δ'.6: Πίτα για την ερώτηση "Ποιος από τους δύο τρόπους ανάπτυξης της διαχείρισης των HTTP αιτημάτων σου φάνηκε ευκολότερος;"

Τέλος, στην ερώτηση "Ποιος από τους δύο τρόπους ανάπτυξης της βάσης δεδομένων σου φάνηκε ευκολότερος;" παρατηρείται αντιστροφή των αποτελεσμάτων της προηγούμενης ερωτησεις με το 42.9% τώρα να διαλέγει σαν πιο εύκολο τον τρόπο του Bubble και το 57.1% τον τρόπο του Django.

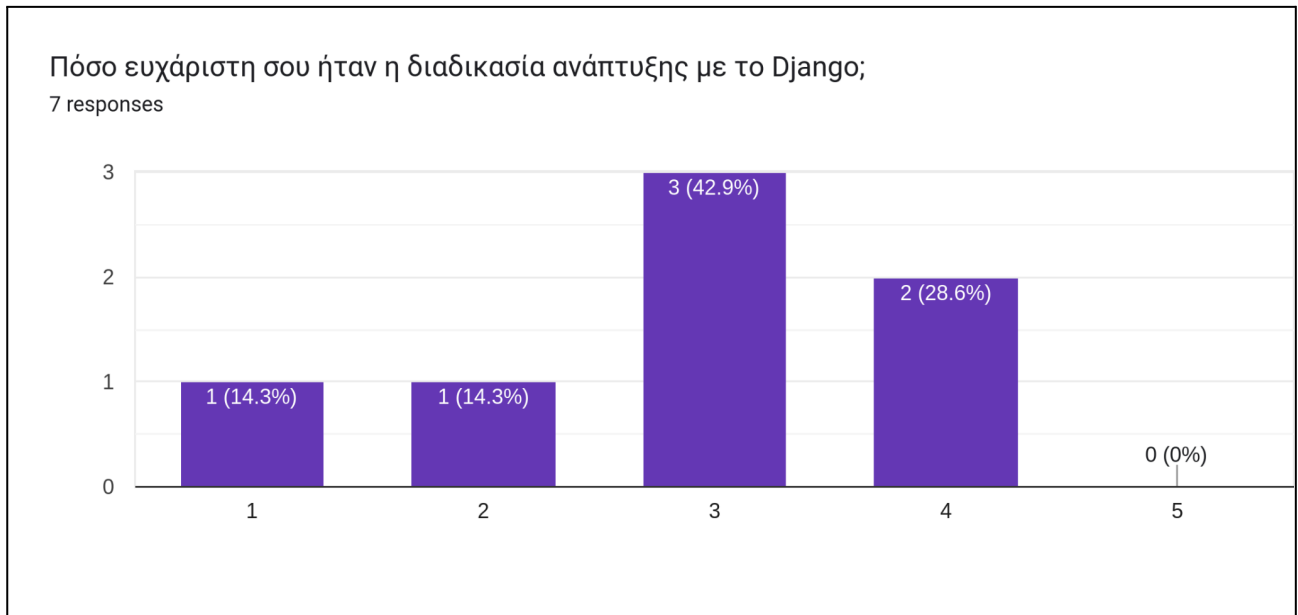


Εικόνα Δ'.7: "Ποιος από τους δύο τρόπους ανάπτυξης της βάσης δεδομένων σου φάνηκε ευκολότερος;"

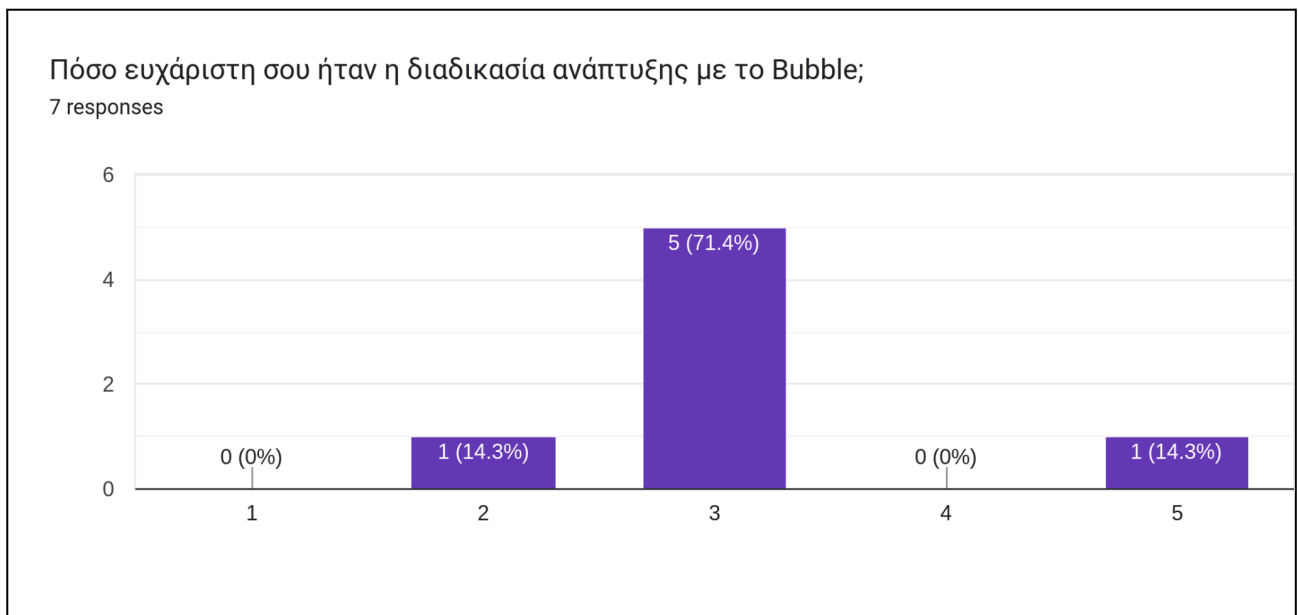
Οι τρεις αυτές ερωτήσεις αφορούν τα τρία μεγάλα πράγματα που διαχειρίζονται τα δύο εργαλεία στη διαδικασία ανάπτυξης μιας διαδικτυακής εφαρμογής. Αν και δεν είναι τα μόνα, είναι αυτά που κυρίως χρησιμοποιήθηκαν στην προκειμένη περίπτωση. Μιας που οι ερωτώμενοι ξεχώρισαν σαν ευκολότερο το Bubble σε δύο από τα τρία βασικά μέρη, μπορούμε να πούμε πως το Bubble θεωρήθηκε το πιο εύκολο από τα δύο εργαλεία γενικότερα. Αξίζει να σημειωθούν και μερικά από τα σχόλια των ερωτώμενων για τις απαντήσεις τους. Μερικοί από τους ερωτώμενους, παρόλο που εν τέλει διάλεξαν το Bubble σαν τον καλύτερο τρόπο ανάπτυξης γραφικής διεπαφής αλλά και προγραμματισμού της διαχείρισης των HTTP αιτημάτων, παραπονέθηκαν για το γεγονός πως το Bubble χρησιμοποιεί πολλαπλά παράθυρα για την παραμετροποίηση των λειτουργιών. Για τη διαχείριση των αιτημάτων HTTP αλλά και τον προγραμματισμό της βάσης δεδομένων, τονίστηκε σε σχόλια η έλλειψη τύπων δεδομένων του Bubble σε σχέση με το Django.

Οι δύο επόμενες ερωτήσεις αφορούν τη διαδικασία ανάπτυξης με τα δύο εργαλεία συνολικά. Πλέον οι ερωτήσεις δεν αφορούν επιμέρους στοιχεία του κάθε εργαλείου. Επίσης οι δύο ερωτήσεις που ακολουθούν δε συγκρίνουν τα εργαλεία μεταξύ τους. Αντιπροσωπεύουν τη βαθμολογία που έδωσαν οι ερωτώμενοι στο κάθε εργαλείο ξεχωριστά. Η ερώτηση "Πόσο ευχάριστη σου ήταν η διαδικασία ανάπτυξης με το Django;" είναι σε κλίμακα από το ένα έως το πέντε. Το δηλώνει πως η διαδικασία δεν ήταν καθόλου ευχάριστη, ενώ το πέντε πως η διαδικασία ήταν πολύ ευχάριστη. Στην ερώτηση αυτή, οι περισσότεροι βαθμολόγησαν το Django με τρία στα πέντε. Ενώ κατά μέσο όρο, η βαθμολογία του Django είναι 2.85 στα

πέντε. Στον κάθετο άξονα βλέπουμε το πλήθος των απαντήσεων και στον οριζόντιο τη βαθμολογία.

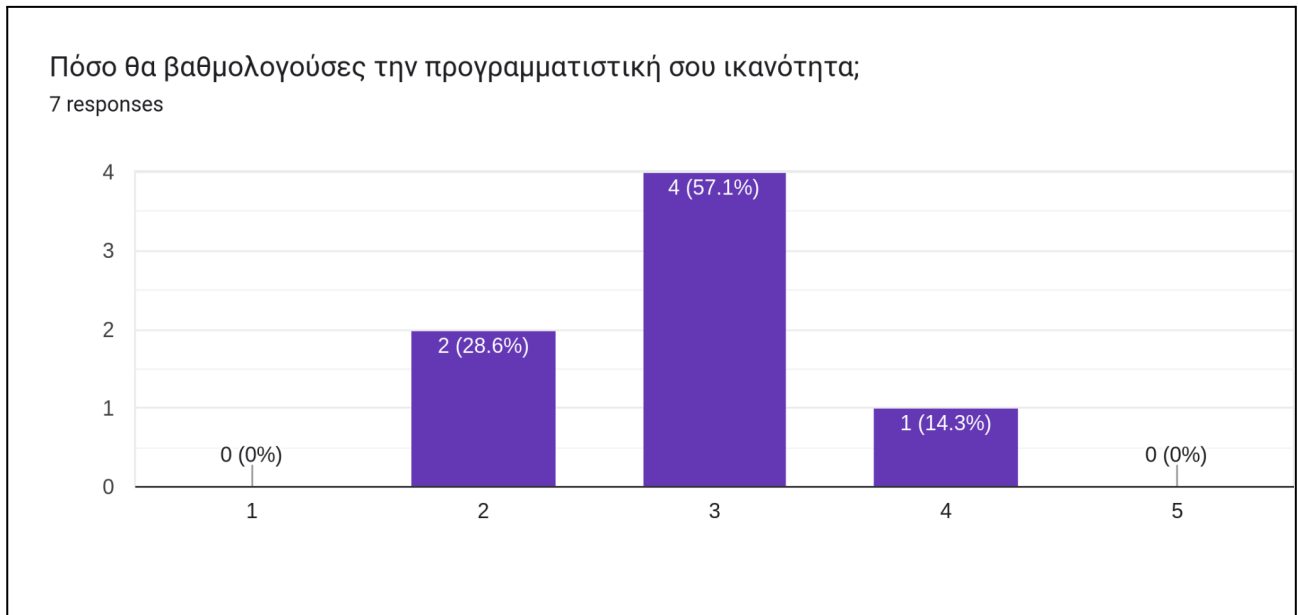


Εικόνα Δ'.8: Ιστόγραμμα για την ερώτηση "Πόσο ευχάριστη σου ήταν η διαδικασία ανάπτυξης με το Django;" Αντίστοιχα, για την ερώτηση "Πόσο ευχάριστη σου ήταν η διαδικασία ανάπτυξης με το Bubble;" οι περισσότεροι βαθμολόγησαν το Bubble επίσης με τρία στα πέντε. Ο μέσος όρος βαθμολογίας είναι 3.14 στα πέντε. Είναι σαφώς μεγαλύτερος καθώς όπως φαίνεται και στην εικόνα Δ'.9, περισσότεροι έβαλαν βαθμολογία τρία στα πέντε σε σχέση με το Django. Επιπλέον δεν υπάρχει βαθμολογία ένα στα πέντε και υπάρχει και βαθμολογία πέντε στα πέντε.



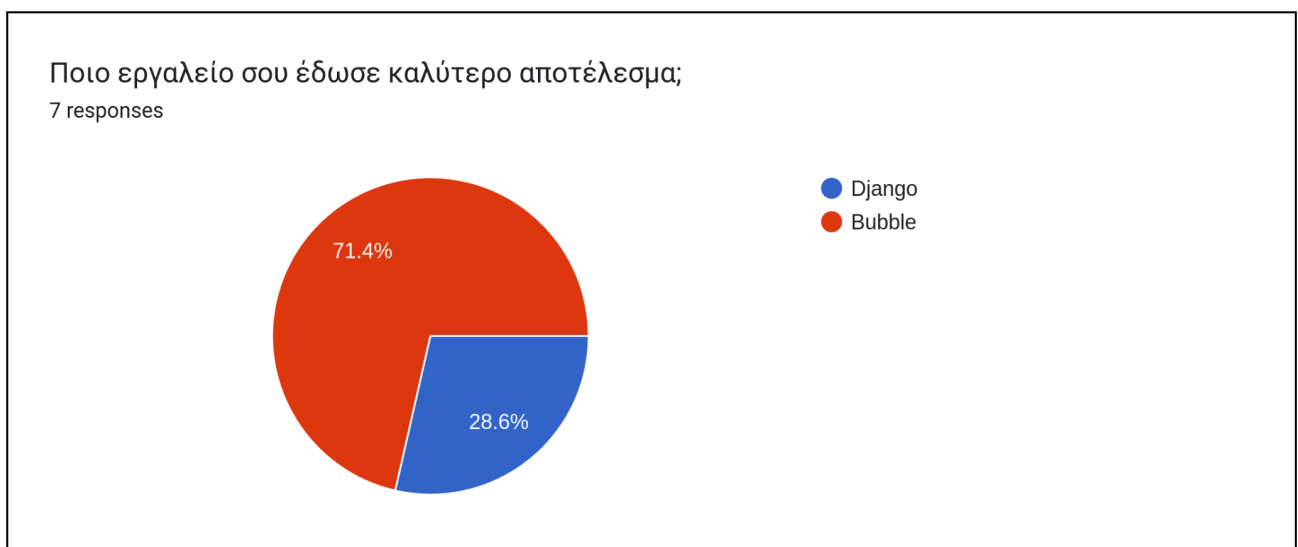
Εικόνα Δ'.9: Ιστόγραμμα για την ερώτηση "Πόσο ευχάριστη σου ήταν η διαδικασία ανάπτυξης με το Bubble;" Οι δύο παραπάνω ερωτήσεις συνοδεύονται από την ερώτηση "Πόσο θα βαθμολογούσες την προγραμματιστική σου ικανότητα;". Το ιστόγραμμα των απαντήσεων σε αυτή την ερώτηση παρουσιάζεται στην εικόνα Δ'.10. Αν και για τη συγκεκριμένη ερώτηση, το ιστόγραμμα δεν είναι αυτό που θα μας δώσει χρήσιμη πληροφορία. Εδώ θα πρέπει να δούμε τις απαντήσεις μεμονωμένα σε συνδυασμό με τις μεμονωμένες απαντήσεις για τις δύο παραπάνω ερωτήσεις. Με αυτόν τον τρόπο εξάγονται συμπεράσματα

σε επόμενη ενότητα.



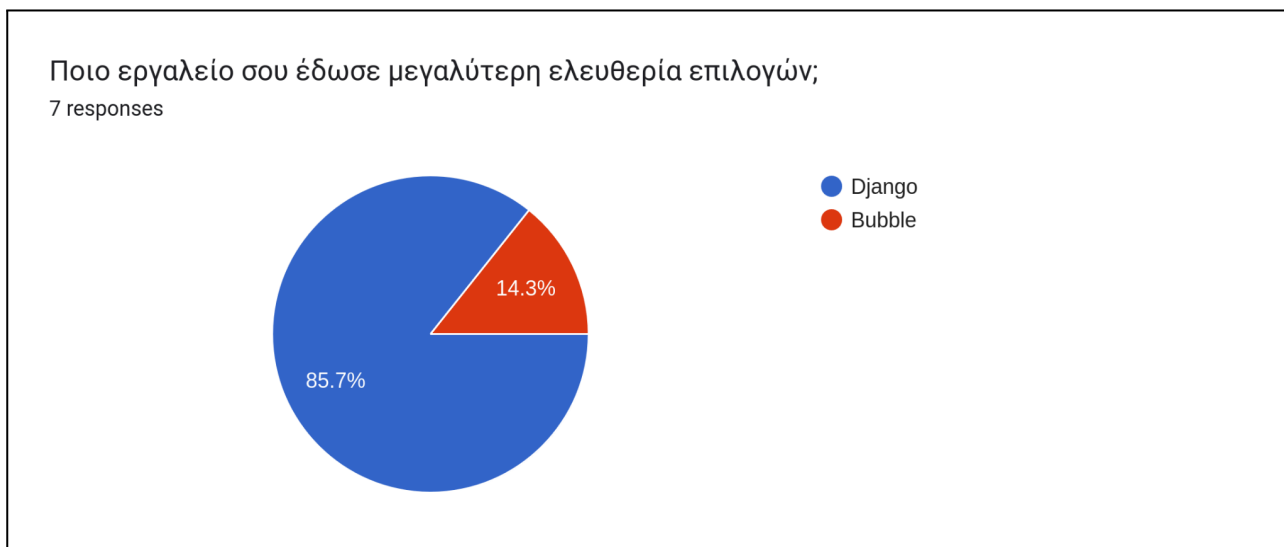
Εικόνα Δ'.10: Ιστόγραμμα για την ερώτηση "Πόσο θα βαθμολογούσες την προγραμματιστική σου ικανότητα;"

Οι δύο ερωτήσεις που ακολουθούν αφορούν την ποιότητα του τελικού αποτελέσματος και την ελευθερία επιλογών που προσφέρει το κάθε εργαλείο. Η ερώτηση "Ποιο εργαλείο σου έδωσε καλύτερο αποτέλεσμα;" είχε το 71.4% να βρίσκει το αποτέλεσμα που παράχθηκε με το Bubble καλύτερο και το 28.6% να βρίσκει καλύτερο το αποτέλεσμα του Django.



Εικόνα Δ'.11: Πίτα για την ερώτηση "Ποιο εργαλείο σου έδωσε καλύτερο αποτέλεσμα;"

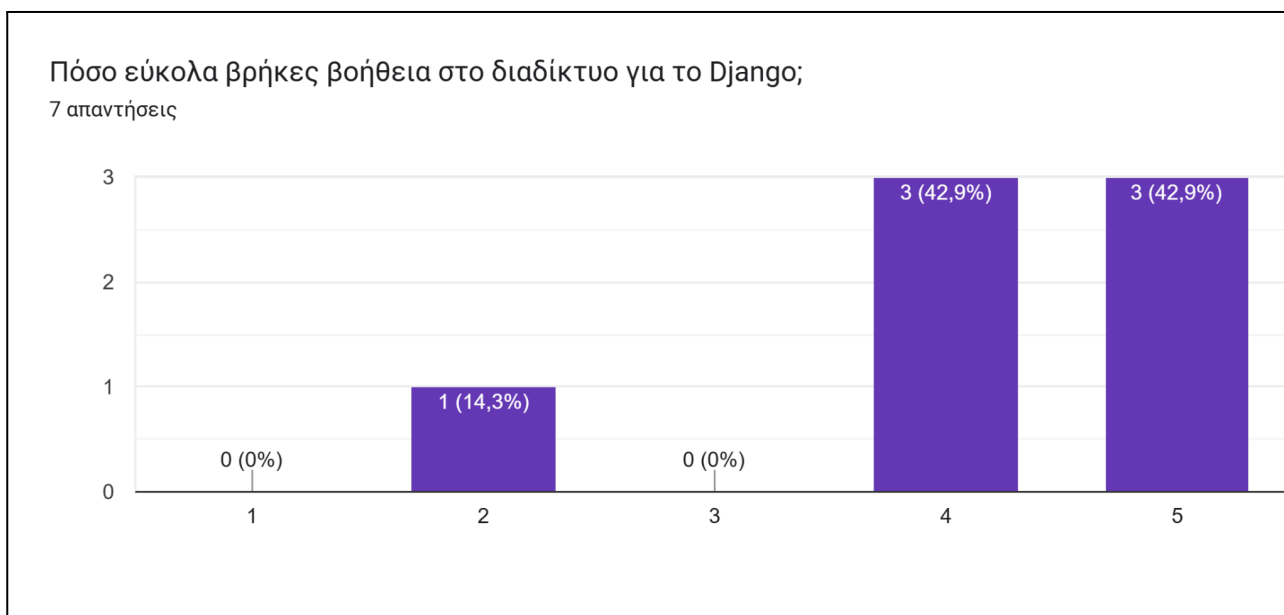
Για την ερώτηση "Ποιο εργαλείο σου έδωσε μεγαλύτερη ελευθερία επιλογών;" το 85.7% απάντησε το Django, ενώ το 14.3% απάντησε το Bubble.



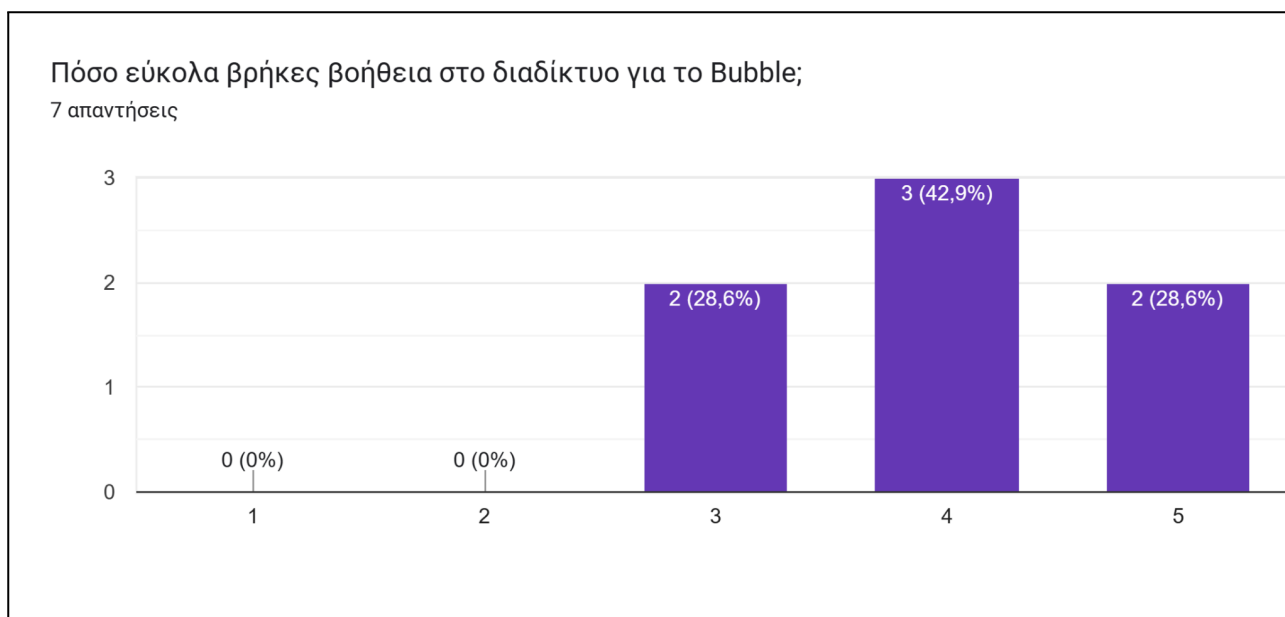
Εικόνα Δ'.12: Πίτα για την ερώτηση "Ποιο εργαλείο σου έδωσε μεγαλύτερη ελευθερία επιλογών;"

Τα σχόλια που έδωσαν οι ερωτώμενοι για τις απαντήσεις τους στις δύο παραπάνω ερωτήσεις προσφέρουν σημαντική επιπλέον πληροφορία. Οι περισσότεροι από αυτούς είπαν πως, δεδομένου του περιορισμένου χρόνου, το Bubble τους έδωσε καλύτερο αποτέλεσμα. Όσον αφορά την ελευθερία επιλογών τονίστηκε για ξανά η μεγαλύτερη ποικιλία σε επιλογές που δίνει το Django για τύπους δεδομένων αλλά και η υποχρεωτική οντότητα "User" και τα υποχρεωτικά χαρακτηριστικά σε κάθε οντότητα που επιβάλλει το Bubble.

Ένα σημαντικός παράγοντας που πρέπει να λάβει κανείς υπόψη όταν διαλέγει ένα εργαλείο για να αναπτύξει το λογισμικό που θέλει είναι η κοινότητα. Αν ένα εργαλείο έχει μεγάλη και ενεργή κοινότητα, τότε θα είναι πιο εύκολο να βρεθεί βοήθεια για ενδεχόμενα προβλήματα στο διαδίκτυο. Έτσι οι συμμετέχοντες του πειράματος ερωτήθηκαν: "Πόσο εύκολα βρήκες βοήθεια στο διαδίκτυο για το Django;" αλλά και "Πόσο εύκολα βρήκες βοήθεια στο διαδίκτυο για το Bubble;". Οι απαντήσεις φαίνονται στα παρακάτω ιστογράμματα.



Εικόνα Δ'.13: Ιστόγραμμα για την ερώτηση "Πόσο εύκολα βρήκες βοήθεια στο διαδίκτυο για το Django;"



Εικόνα Δ'.14: Ιστόγραμμα για την ερώτηση "Πόσο εύκολα βρήκες βοήθεια στο διαδίκτυο για το Bubble;"

Στην εικόνα Δ'.13 φαίνεται πως ένας χρήστης απάντησε δύο στα πέντε και από τρεις απάντησαν τέσσερα και πέντε στα πέντε αντίστοιχα για το Django. Αυτό του δίνει μέσο όρο βαθμολογίας 4,14. Για το Bubble από δύο συμμετέχοντες απάντησαν τρία στα πέντε και πέντε στα πέντε αντίστοιχα, ενώ τρεις επέλεξαν τέσσερα στα πέντε. Ο μέσος όρος βαθμολογίας του Bubble είναι 4. Μια διαφορά που σχολιάστηκε από συμμετέχοντα ήταν η δυνατότητα χρήσης μέσων τεχνητής νοημοσύνης για βοήθεια με των κώδικα του Django. Αν και η χρήση αυτών των μέσων μπορεί να γίνει και για το Bubble, η αποδοτικότητα τους δεν είναι η ίδια. Στην περίπτωση του κώδικα υπάρχουν εξειδικευμένα, τέτοιου τύπου εργαλεία. Αλλά ακόμα και εργαλεία που δεν έχουν σαν κύριο σκοπό την παραγωγή κώδικα μπορούν να χρησιμοποιηθούν, για να παράξουν κώδικα έτοιμο προς χρήση. Στην περίπτωση του Bubble, αυτά τα εργαλεία, μπορούν να δώσουν οδηγίες για το πως να επιτευχθεί ένα ερώτημα. Δεν μπορούν όμως να εφαρμόσουν αυτές τις οδηγίες.

Δ'.2.4 Υποστήριξη λειτουργιών από τα εργαλεία

Στην υποενότητα αυτή παρουσιάζονται οι δυνατότητες ή αδυναμίες των δύο εργαλείων να υποστηρίξουν λειτουργίες που ήταν αναγκαίες στο δημιουργό για να έρθει εις πέρας το τελικό αποτέλεσμα. Σε γενικές γραμμές και τα δύο εργαλεία πρόσφεραν όλες τις απαραίτητες λειτουργίες για την υλοποίηση του συγκεκριμένου πρότζεκτ χωρίς μεγάλους συμβιβασμούς. Αν αναλύσουμε την απόδοση των εργαλείων λίγο αναλυτικότερα όμως θα παρατηρήσουμε πως το Bubble προσφέρει συγκεκριμένες λειτουργίες με πολύ συγκεκριμένο τρόπο. Για παράδειγμα, όπως αναφέρεται και παραπάνω, το Bubble απαιτεί τη χρήση διεύθυνσης ηλεκτρονικού ταχυδρομείου για τη διαχείριση των λογαριασμών των χρηστών. Ένα ακόμα παρόμοιο παράδειγμα, είναι η παρουσία οντοτήτων αλλά και χαρακτηριστικών σε κάθε οντότητα, που δεν μπορούν να σβηστούν ασχέτως από το αν τελικά χρησιμοποιούνται. Επιπλέον το Bubble δεν προσφέρει κάποιον τύπο οντοτήτων που καλύπτει την ανάγκη του συγκεκριμένου πρότζεκτ για την αποθήκευση δεδομένων διάρκειας για το χαρακτηριστικό "Time Limit". Μια ακόμα έλλειψη του Bubble σε σχέση με το Django είναι η λειτουργία διαχείρισης των συσχετισμένων οντοτήτων σε περίπτωση διαγραφής. Αν για παράδειγμα, διαγραφεί ένα παζλ στην περίπτωση του Django δύναται να οριστεί η συμπεριφορά των

"βοήθεια", "επιβράβευση" και "λύση". Ορίζοντας στο αρχείο `models.py` τη μεταβλητή `"on_delete"` που είναι μάλιστα υποχρεωτική. Οι επιλογές που δίνονται είναι `"CASCADE"`, `"PROTECT"`, `"RESTRICT"`, `"SET_NULL"`, `"SET_DEFAULT"`, `"SET()"` και `"DO_NOTHING"`. Αυτές δίνουν τη δυνατότητα παραμετροποίησης της συμπεριφοράς των οντοτήτων "παιδιών" όταν διαγράφεται η οντότητα "γονιός". Στο Bubble, λόγω εν μέρη και του τρόπου διαχείρισης της βάσεις δεδομένων, κάτι τέτοιο δεν προσφέρεται αυτόματα. Θα πρέπει λοιπόν ο δημιουργός να προγραμματίσει τη συμπεριφορά αυτή χειροκίνητα στη λειτουργία διαγραφής της οντότητας. Η πολυπλοκότητα αυτής της λειτουργίας αυξάνεται πολύ γρήγορα στα σενάρια όπου υπάρχει αλυσίδα από οντότητες "γονείς" και οντότητες "παιδιά". Στην προκειμένη περίπτωση, αν διαγραφεί ένα παράδειγμα της οντότητας "γρίφος" θα πρέπει να διαγραφούν και τα κατάλληλα παραδείγματα των οντοτήτων "παζλ", "βοήθεια", "επιβράβευση" και "λύση". Αν κάποια τέτοια συμπεριφορά δεν προγραμματιστεί ρητά από το δημιουργό, η προκαθορισμένη συμπεριφορά είναι να μη γίνει τίποτα. Αυτό οδηγεί σε υπερβολική χρήση πόρων.

Δ'.2.5 Κόστος ανάπτυξης

Εδώ αναλύονται πιθανά κόστη που υπήρξαν κατά τη διάρκεια της ανάπτυξης της ιστοσελίδας με καθένα από τα εργαλεία. Δεν αναφέρονται τιμές, καθώς κάτι τέτοιο δεν είναι αναγκαίο σε αυτή την περίπτωση. Αναφέρονται συγκεκριμένες λειτουργίες που πιθανόν να βρίσκονται κλειδωμένες πίσω από κάποια συνδρομή ή κάποια πληρωμή.

Είναι σημαντικό να αναφερθεί πως η ανάπτυξη των ιστοσελίδων για τις ανάγκες αυτής της εργασίας έγινε χωρίς να πληρωθεί κάποιο ποσό για το ξεκλείδωμα κάποιον από αυτών των λειτουργιών. Αν κάποια λειτουργία που ήταν απαραίτητη για την υλοποίηση μιας εκ των δύο σελίδων και χρειαζόταν πληρωμή, παραλήφθηκε και θα σημειωθεί σε αυτή την υποενότητα.

Αν και όπως αναφέρεται, δεν πληρώθηκε κάποιο ποσό για το ξεκλείδωμα λειτουργιών, για το Django, έγινε αγορά μαθήματος 36 ωρών από την υπηρεσία Udemy. Αντίστοιχα μαθήματα για το Django μπορούν να βρεθούν με σχετική ευκολία και σε δωρεάν μορφή σε υπηρεσίες όπως το YouTube. Δύο εργαλεία που χρησιμοποιήθηκαν για το Django αλλά ήταν κλειδωμένα πίσω από συνδρομή για το Bubble είναι: Εργαλείο για διαχείριση εκδόσεων (Version Control). Για το Django έγινε χρήση του Git και GitHub που είναι δωρεάν, ενώ το Bubble προσφέρει το δικό του εργαλείο αν επιλεγθεί συγκεκριμένη επί πληρωμή συνδρομή. Εργαλείο για API. Για το Django έγινε χρήση του εργαλείου "django REST framework". Αυτό δίνει τη δυνατότητα δημιουργίας API endpoints μέσω παρόμοιου συντακτικού και φιλοσοφίας με το Django. Το Bubble δίνει επίσης τη δυνατότητα ανάπτυξης API μέσω εσωτερικών εργαλείων, που είναι όμως κλειδωμένα πίσω από συνδρομή επί πληρωμή. Τέλος αξίζει να συζητηθεί το ενδεχόμενο κόστος φιλοξενίας της σελίδας σε κάποιο διακομιστή με σκοπό να είναι διαθέσιμη μέσω του διαδικτύου. Σε αυτή την περίπτωση και οι δύο ιστοσελίδες θα απαιτούσαν την καταβολή κάποιου ποσού στη μορφή συνδρομής σε κάποια υπηρεσία φιλοξενίας ιστοσελίδων. Η διαφορά εδώ είναι στη δυνατότητα και τον αριθμό επιλογών. Μιας που το Django βασίζεται στην Python, υπάρχουν διαφορετικές επιλογές για υπηρεσίες φιλοξενίας ιστοσελίδων. Μερικών εκ των οποίων μπορεί να είναι και δωρεάν. Επίσης υπάρχει η δυνατότητα, δεδομένων των απαραίτητων γνώσεων και ικανοτήτων, η ιστοσελίδα να φιλοξενηθεί από τον ίδιο το δημιουργό. Λόγω

της φύσεώς του Bubble, που είναι μια εφαρμογή με κλειστό και ιδιωτικό κώδικα, οι επιλογές φιλοξενίας περιορίζονται στις λύσεις που προσφέρει το ίδιο το Bubble και μόνο.

Δ'.3 Ανάλυση

Στις υποενότητες της Δ'.3 αναλύονται τυχόν περιορισμοί που υπάρχουν στον τρόπο διεξαγωγής του πειράματος αλλά και οποιαδήποτε αρνητικά αποτελέσματα. Περιορισμοί του πειράματος μπορεί να είναι είτε μέρη αυτού που θα μπορούσαν να έχουν μετρηθεί καλύτερα είτε συγκεκριμένες παραδοχές που χαρακτηρίζουν το πείραμα. Σαν αρνητικά αποτελέσματα ορίζονται οποιαδήποτε αποτελέσματα από τα οποία προκύπτουν συμπεράσματα που δεν υποστηρίζουν την αρχική υπόθεση που αναφέρθηκε στο Δ'.1.

Δ'.3.1 Περιορισμοί

Το συγκεκριμένο πείραμα έχει ως σκοπό την επιβεβαίωση ή κατάρριψη μιας αρχικής υπόθεσης. Για το συγκεκριμένο πείραμα συγκεντρώθηκαν μια σειρά από δεδομένα κάποια από τα οποία είναι ποσοτικά ενώ κάποια είναι ποιοτικά. Οι περιορισμοί και παραδοχές αυτού του πειράματος αφορούν τους διάφορους τρόπους με τους οποίους συλλέχθηκαν τα δεδομένα αυτά. Η πρώτη παραδοχή έχει να κάνει με τα δεδομένα του χρόνου ανάπτυξης. Η παραδοχή είναι πως δεν υπάρχει ιδιαίτερος λόγος για ακριβή δεδομένα μιας και η παρατηρούμενη διαφορά είναι τέτοια ώστε να μπορούμε με ασφάλεια να εξάγουμε συμπεράσματα. Έτσι, ο περιορισμός για το συγκεκριμένο άξονα σύγκρισης είναι πως τα δεδομένα δεν είναι ακριβή. Μια καλή ακρίβεια για την εξυπηρέτηση του σκοπού αυτού του άξονα σύγκρισης, θα ήταν οι ώρες. Για τον άξονα "Υποστήριξη λειτουργιών από τα εργαλεία." μια σημαντική παραδοχή έχει να κάνει με τον τρόπο με τον οποίο αναζητήθηκε η πληροφορία για την υποστήριξη ή μη μιας λειτουργίας. Αυτή έγινε με δύο τρόπους. Κατά τη στιγμή της ανάπτυξης, αν δηλαδή ήταν προφανής η μη υποστήριξη κάποιας λειτουργίας για οποιονδήποτε λόγο. Μετά από διαδικτυακή έρευνα. Αν ο δημιουργός βρήκε οριστική απάντηση στο διαδίκτυο για τη μη υποστήριξη μιας λειτουργίας ή δε βρήκε τον τρόπο υλοποίησης της μετά από αυτή την έρευνα. Τέλος, για τα ποιοτικά στοιχεία τα οποία συλλέχθηκαν με τη χρήση συνεντεύξεων, ο περιορισμός βρίσκεται στον όγκο των απαντήσεων. Αυτός δημιουργεί και άλλους περιορισμούς όπως τον "τύπο" των συμμετεχόντων. Αυτοί ήταν κατά κύριο λόγο φοιτητές ή απόφοιτοι της πληροφορικής. Αν και η συμμετοχή στο πείραμα απαιτεί βασικές γνώσεις πληροφορικής, η έλλειψη της ποικιλίας αναφέρεται στην εμπειρία που είχαν οι συμμετέχοντες πάνω στα συγκεκριμένα εργαλεία. Οι περισσότεροι από αυτούς δεν είχαν καμία προηγούμενη επαφή με αυτά. Ένα μεγαλύτερο και πιο ποικίλο δείγμα συμμετεχόντων θα οδηγούσε σε αποτελέσματα που ανταποκρίνονται περισσότερο στην πραγματικότητα.

Δ'.3.2 Αρνητικά αποτελέσματα

Γενικότερα, δεν προκύπτουν αρκετά στοιχεία που αντικρούουν την αρχική υπόθεση. Ένα από τα βασικά επιχειρήματα της υπόθεσης που φαίνεται να μην επιβεβαιώνεται, είναι η διαφορά στη δυνατότητα κλιμάκωσης των δύο εφαρμογών. Κάτι τέτοιο φαίνεται να μην ισχύει, καθώς και τα δύο εργαλεία διαθέτουν μεθόδους που επιτρέπουν στους προγραμματιστές να επιταχύνουν το ρυθμό ανάπτυξης.

Κεφάλαιο Ε'

Ε' Συζήτηση

Στην ενότητα αυτή παρουσιάζεται μια συζήτηση με πιο υποκειμενικό χαρακτήρα και ερμηνεία των αποτελεσμάτων. Το πείραμα που διενεργήθηκε στο πλαίσιο αυτής της εργασίας πραγματοποιήθηκε με σκοπό την εξέταση μιας αρχικής υπόθεσης. Η υπόθεση αυτή σχετίζεται με την εφαρμογή εργαλείων προγραμματισμού σε ένα επιχειρησιακό περιβάλλον. Τα βασικά στοιχεία αυτής της υπόθεσης αφορούν το πιο προσβάσιμο επίπεδο εισόδου που προσφέρουν τα no-code εργαλεία σε αντίθεση με αυτά που χρησιμοποιούν κώδικα. Για να επιτευχθεί αυτό, τα no-code εργαλεία προσφέρουν λιγότερες λειτουργίες, δυνατότητες και ελευθερίες στους χρήστες τους. Λόγω της μεγαλύτερης πολυπλοκότητας τους τα code εργαλεία οδηγούν σε ένα ελάχιστο βιώσιμο προϊόν πιο αργά από τα no-code. Τέλος, λόγω των περισσότερων δυνατοτήτων, τα code εργαλεία προσφέρουν μεγαλύτερη δυνατότητα κλιμάκωσης στο ρυθμό ανάπτυξης.

Ε'.1 Συμπεράσματα

Με τα δεδομένα που συλλέχθηκαν και παρουσιάστηκαν μπορούν να εξαχθούν χρήσιμα συμπεράσματα ώστε να απαντηθούν καταφατικά ή αρνητικά οι προτάσεις της αρχικής υπόθεσης. Όσον αφορά το χαμηλότερο κατώφλι εισόδου που προσφέρουν τα no-code εργαλεία μπορούμε, με βάση τα δεδομένα, να εξάγουμε με ασφάλεια πως κάτι τέτοιο ισχύει. Όπως παρουσιάστηκε, οι συμμετέχοντες του ερωτηματολογίου, απάντησαν στην πλειοψηφία τους πως σε δεδομένο χρονικό διάστημα το Bubble φέρνει καλύτερο αποτέλεσμα. Ένα ακόμα γεγονός που στηρίζει την πρόταση αυτή είναι η εμπειρία που δεν είχαν οι συμμετέχοντες σε κανένα από τα δύο εργαλεία. Επιπλέον στοιχείο είναι και ο χρόνος που χρειάστηκε ο δημιουργός της διαδικτυακής εφαρμογής για να τη δημιουργήσει με το κάθε εργαλείο. Η επόμενη θέση της υπόθεσης αφορά τις λειτουργίες, δυνατότητες και ελευθερίες που προσφέρουν τα δύο εργαλεία. Τα δεδομένα δείχνουν πως τα code εργαλεία προσφέρουν σαφώς περισσότερες λειτουργίες. Είτε αυτές δεν υποστηρίζονται καθόλου από τα no-code εργαλεία, είτε είναι κλειδωμένες πίσω από κάποια συνδρομή ή πληρωμή. Επιπροσθέτως, το no-code εργαλείο στην προσπάθειά του να επισπεύσει τη διαδικασία ανάπτυξης και να διευκολύνει τους χρήστες του, παίρνει κάποιες αποφάσεις γι' αυτούς οι οποίες ενδέχεται να μην είναι πάντοτε θεμιτές. Το παράδειγμα της οντότητας του χρήστη, που δε δύναται να σβηστεί, αλλά και των υποχρεωτικών χαρακτηριστικών που έχουν όλες οι οντότητες είναι αρκετό για να αναδείξει τα παραπάνω και πως τα no-code εργαλεία θυσιάζουν τις ελευθερίες για την ευκολία χρήσης. Για του λόγους που μόλις αναφέρθηκαν είναι ξεκάθαρο πως το code εργαλείο προσφέρει και περισσότερες δυνατότητες. Κάτι τέτοιο γίνεται εμφανές όταν δούμε την δυνατότητα παραμετροποίησης της συμπεριφοράς της βάσης δεδομένων σε περίπτωση διαγραφής μιας πλειάδας. Εκεί που το Django προσφέρει επτά διαφορετικές επιλογές για τη διαχείριση αυτή, το Bubble απαιτεί χειροκίνητη υλοποίηση. Οι διαφορές αυτές έγιναν αισθητές και από τους συμμετέχοντες στις συνεντεύξεις. Από την εμπειρία του δημιουργού της συγκεκριμένης εφαρμογής, αλλά και τις απαντήσεις των ερωτηματολογίων, προκύπτει πως τα no-code εργαλεία οδηγούν όντως γρηγορότερα

σε ένα ελάχιστο βιώσιμο προϊόν. Ένα τέτοιο είναι και η εφαρμογή που αναπτύχθηκε για τη συγκεκριμένη εργασία. Η οποία αναπτύχθηκε γρηγορότερα με το Bubble. Όπως φαίνεται και από τα δεδομένα των ερωτηματολογίων, το Bubble έδωσε επίσης καλύτερο αποτέλεσμα σε δεδομένο χρόνο. Έτσι όταν είναι δεδομένο και ίδιο το αποτέλεσμα, αυτό μπορεί να παραχθεί πιο γρήγορα με το Bubble. Όταν αντίθετα είναι δεδομένος ο χρόνος, τότε το αποτέλεσμα που παράχθηκε με το Bubble θεωρήθηκε καλύτερο. Τέλος, τα δεδομένα δε φαίνεται να συνιστούν πως ένα από τα δύο εργαλεία έχει το πλεονέκτημα στην κλιμάκωση του ρυθμού ανάπτυξης. Τα δύο εργαλεία προσφέρουν λειτουργίες που επιτρέπουν στους χρήστες τους να δημιουργούν επαναχρησιμοποιούμενες μονάδες με σκοπό να μη χρειάζεται να τις δημιουργούν ξεχωριστά κάθε φορά που επιθυμούν να τις χρησιμοποιήσουν. Προσφέρουν επίσης τη δυνατότητα συνδυασμού πολλών τέτοιων μονάδων με σκοπό την αύξηση, με την πάροδο του χρόνου, της λειτουργικότητας που προσφέρει η κάθε μια. Ένα τέτοιο παράδειγμα από την εφαρμογή που αναπτύχθηκε για αυτή την εργασία είναι η κεφαλίδα (“Header”) που υπάρχει σε κάθε σελίδα. Αυτή δημιουργήθηκε μία φορά και χρησιμοποιήθηκε επανειλημμένα σε όλες τις σελίδες.

Ολιστικά λοιπόν, φαίνεται πως τα no-code εργαλεία, δεδομένου ότι είναι ίδια φύσεως με το Bubble, μπορούν να προσφέρουν τη δυνατότητα σε μια επιχείρηση να βγάλει το προϊόν της γρηγορότερα στην αγορά σε σχέση με ένα εργαλείο με κώδικα. Όσον αφορά το κόστος, το συμπέρασμα που μπορεί να εξαχθεί δεν είναι τόσο ξεκάθαρο. Τα δεδομένα δείχνουν πως το no-code εργαλείο έχει περισσότερες λειτουργίες πίσω από συνδρομές ή πληρωμές ενώ ταυτόχρονα είναι δεσμευτικό ως προς την επιλογή παρόχου φιλοξενίας. Από την άλλη επιτρέπει την επιλογή εργαζομένων με χαμηλότερη εμπειρία, γεγονός που μπορεί να οδηγήσει στη μείωση των εξόδων της μισθοδοσίας. Λαμβάνοντας τα πάντα υπόψη, η νεοσύστατες εταιρείες τύπου startup φαίνεται να είναι το καλύτερο περιβάλλον εφαρμογής για τα no-code εργαλεία. Θα τους επιτρέψουν να βγάλουν το προϊόν τους στην αγορά γρηγορότερα και αποδοτικότερα. Παράλληλα, η πρώτη εμφάνιση ενός προϊόντος δεν έχει, συνήθως, απαιτήσεις για περίπλοκες λειτουργίες και δεν είναι ανάγκη να είναι τέλεια βελτιστοποιημένο. Αντίθετα, εταιρείες των οποίων τα προϊόντα που απαιτούν την υλοποίηση περισσότερο ανεπτυγμένων λειτουργιών ή και την μεγιστοποίηση της βελτιστοποίησής τους, είναι τα καταλληλότερα περιβάλλοντα για τη χρήση code εργαλείων. Μια τέτοια εταιρεία μπορεί να είναι και μια startup της οποίας το προϊόν έχει πλέον εδραιωθεί στην αγορά και το κύριο της μέλημα είναι η ανάπτυξη του ως προς τις λειτουργίες που προσφέρει. Μπορεί επίσης να είναι μια εταιρεία με περισσότερα χρόνια παρουσίας στον κλάδο που, με την εμπειρία της, αναλαμβάνει περισσότερο πολύπλοκα έργα.

Ε'.2 Μελλοντικές επεκτάσεις

Ένας σημαντικός και εύκολος τρόπος να επεκταθεί η συγκεκριμένη εργασία μελλοντικά είναι η προσθήκη επιπλέον εργαλείων και των δύο τύπων. Έτσι θα σχηματιστεί μια πιο ολοκληρωμένη και γενικότερη εικόνα για τα εργαλεία των δύο αυτών τύπων ανάπτυξης. Τα γεγονότα πως ένα εργαλείο παρουσίασε τα συγκεκριμένα δεδομένα, δεν εγγυάται πως αυτά θα παρουσιαστούν και από τα υπόλοιπα. Η διεξαγωγή επιπλέον συνεντεύξεων είναι ένα ακόμα τρόπος. Με μεγαλύτερο όγκο απαντήσεων και από συμμετέχοντες με μεγαλύτερη διαφορά στην εμπειρία, τα στατιστικά αποτελέσματα των ερωτηματολογίων θα είναι περισσότερο αξιόπιστα. Ένα τέτοιο δείγμα συμμετεχόντων αντιπροσωπεύει καλύτερα την

πραγματικότητα. Τα escape room αποτελούν, συνήθως, επιχειρηματική δραστηριότητα. Μιας που η εφαρμογή στοχεύει στη δημιουργία escape room άρα και στον ιδιοκτήτη της επιχείρησης και όχι στον παίκτη, θα μπορούσαν μελλοντικά να προστεθούν επιχειρηματικά στοιχεία στο μοντέλο που αναπτύχθηκε για την εφαρμογή. Για παράδειγμα θα μπορούσε να προστεθεί το χαρακτηριστικό της τιμής ανα δωμάτιο.

Βιβλιογραφία

- Bubble Docs: Introduction. Retrieved June 8, 2023, from <https://manual.bubble.io/>
- Django documentation | Django documentation. Retrieved June 8, 2023, from <https://docs.djangoproject.com/en/4.1/>
- Καραγιώργη, Γ., & Συμεού, Α. (2003). *Κονστρουκτιβισμός και Εκπαιδευτικός Σχεδιασμός*.
- Al Alamin, M. A., Malakar, S., Uddin, G., Afroz, S., Bin Haider, T., & Iqbal, A. (2021, June 28). *An Empirical Study of Developer Discussions on Low-Code Software Development Challenges*. IEEE Xplore. Retrieved March 11, 2024, from <https://ieeexplore.ieee.org/abstract/document/9463132/authors>
- Breakout EDU*. (n.d.). Breakout EDU - Educational Games. Retrieved March 10, 2024, from <https://www.breakoutedu.com/>
- Brown, N., Darby, W., & Coronel, H. (2019, May). *An Escape Room as a Simulation Teaching Strategy*. Retrieved March 5, 2023, from <https://www.sciencedirect.com/science/article/pii/S1876139918302615>
- Erlam, G. (2014). *Simulation and 'millenials' - a great fit*. <https://www.proquest.com/docview/1510299432?pq-origsite=gscholar&fromopenview=true>
- Fotaris, P., & Mastoras, T. (2019, October). Escape rooms for learning: A systematic review. *Proceedings of the European Conference on Games Based Learning*, 235-243. 10.34190/GBL.19.179
- López-Pernas, S., Gordillo, A., & Barra, E. (2021). Technology-Enhanced Educational Escape Rooms: A Road Map. *IT Professional*, 23(02), 26-32. 10.1109/MITP.2021.3062749
- Martinez, E., & Pfister, L. (2023, May 11). Benefits and limitations of using low-code development to support digitalization in the construction industry. *Automation in Construction*, 152. <https://www.sciencedirect.com/science/article/pii/S0926580523001693>
- Mayer, R. (1992). Cognition and instruction: Their historic meeting within educational psychology. *Journal of educational Psychology*, 84(4), 405-412. 10.1037/0022-0663.84.4.405
- Mystakidis, S., Cachafeiro, E., & Hatzilygeroudis, I. (2019). *Enter the Serious E-scape Room: A Cost-Effective Serious Game Model for Deep and Meaningful E-learning*.

https://ieeexplore.ieee.org/abstract/document/8900673?casa_token=9bm9OF2Esa8AAAAA:ZwFS2qIfb3GKVQXovc84QCXPmFxwh5jlh65Dqq-XGMFpSX0Zzr5uPFtRizIOVuRkSM8BctYdY-XD

Nicholson, S. (2015, May 24). *Peeking Behind the Locked Door: A Survey of Escape Room Facilities*.

Retrieved March 5, 2023, from <https://scottnicholson.com/pubs/erfacwhite.pdf>

Nicholson, S. (2016). *Ask Why: Creating a Better Player Experience Through Environmental Storytelling and Consistency in Escape Room Design*.

Nicholson, S. (2016). *Preprint of Nicholson, Scott. (2016). The State of Escape: Escape Room Design and Facilities*. Retrieved March 5, 2023, from <https://scottnicholson.com/pubs/stateofescape.pdf>

Nicholson, S. (2018). Creating Engaging Escape Rooms for the Classroom. *Childhood Education*, 94(1), 44-49. 10.1080/00094056.2018.1420363

Olusegun, B. S., & Olusegun, S. (2015). Constructivism learning theory: A paradigm for teaching and learning. *Journal of Research & Method in Education*, 5(6), 66-70. 10.9790/7388-05616670

Papaioannou, T., Tsohou, A., Bounias, G., & Karagiannis, S. (2022, October 06). *A Constructive Approach for Raising Information Privacy Competences: The Case of Escape Room Games*. Springer Link. https://link.springer.com/chapter/10.1007/978-3-031-17926-6_3

Pinho, D., Aguiar, A., & Amaral, V. (2023, January). What about the usability in low-code platforms? A systematic literature review. *Journal of Computer Languages*, 74, 101–185. <https://www.sciencedirect.com/science/article/pii/S259011842200082X>

Rokis, K., & Kirikova, M. (2023). Exploring Low-Code Development: A Comprehensive Literature Review. *Complex Systems Informatics and Modeling Quarterly*. <https://csimq-journals.rtu.lv/article/view/csimq.2023-36.04>

Shay, R., Komanduri, S., Durity, A. L., Huh, P. S., Mazurek, M. L., Segreti, S. M., Ur, B., Bauer, L., Christin, N., & Cranor, L. F. (2016, May). Designing Password Policies for Strength and Usability. *ACM Transactions on Privacy and Security*, 18(4), 34. 10.1145/2891411

Stack Overflow Developer Survey 2022. (n.d.). Stack Overflow Annual Developer Survey. Retrieved June 8, 2023, from <https://survey.stackoverflow.co/2022/>

Wiemker, M., Elumir, E., & Clare, A. (2015, November). *Escape Room Games: "Can you transform an unpleasant situation into a pleasant one?"* the Codex. Retrieved March 5, 2023, from

[https://thecodex.ca/wp-content/uploads/2016/08/00511Wiemker-et-al-Paper-Escape-Room-Games.p
df](https://thecodex.ca/wp-content/uploads/2016/08/00511Wiemker-et-al-Paper-Escape-Room-Games.pdf)