
Πανεπιστήμιο Πειραιά

Τμήμα Ψηφιακών Συστημάτων

Δικτυοκεντρικά Πληροφοριακά Συστήματα 2020-2021

Εξαμηνιαία Εργασία

Παναγιώτης Καραμολέγκος

E17065

Περιεχόμενα

Λίστα Εικόνων	iii
1. Θεματολογία Συστήματος	1
1.1. Πρόβλημα ως προς επίλυση	1
1.2. Διαδικασία Υλοποίησης Συστήματος	1
2. Υλοποίηση Συστήματος.....	2
2.1. Αρχιτεκτονική	2
2.2. Υλοποιημένες Υπηρεσίες Ιστού.....	2
2.2.1. Υπηρεσίες του Συστήματος	3
2.2.2. Υπηρεσίες των Χρηστών του Συστήματος.....	4
2.2.3. Υπηρεσίες των Λιστών Παιχνιδιών	8
2.2.4. Υπηρεσίες των Δημοσιεύσεων	10
2.2.5. Υπηρεσίες των Σχολίων	12
2.2.6. Υπηρεσίες των Upvote (Likes)	13
2.2.7. Υπηρεσίες των Μηνυμάτων.....	15
2.3. Εξωτερικές Υπηρεσίες Ιστού/ Μικροϋπηρεσίες.....	16
3. Εγχειρίδιο Χρήσης Συστήματος.....	17
3.1. Τεχνολογίες.....	17
3.1.1. Γλώσσες	17
3.1.2. Βιβλιοθήκες.....	17
3.1.3. Προγράμματα.....	18
3.2. Βάση Δεδομένων – MySQL.....	19
3.2.1. User	20
3.2.2. Game	20
3.2.3. Message	20
3.2.4. Post	20
3.2.5. Comment	21
3.2.6. Upvote	21
3.3. Εγχειρίδιο Χρήστη	22
3.3.1. Εγκατάσταση, Εκτέλεση και Άνοιγμα.....	22
3.3.2. Η Μεριά του Χρήστη.....	23
3.3.3. Η Μεριά του Διαχειριστή.....	34
3.4. Τεχνικό Εγχειρίδιο	36
3.4.1. welcome.....	37
3.4.2. HomePage.....	40
3.4.3. Admin.....	42
3.4.4. MyGames	42

3.4.5.	Post	44
3.4.6.	Profile.....	47
3.4.7.	Search	52
3.4.8.	ShowFreeGames	53

Λίστα Εικόνων

Εικόνα 1: Ένα Sprint στην Μεθοδολογία Agile.....	1
Εικόνα 2: Game Society Database (gamesocietydb).....	19
Εικόνα 3: Σελίδα Log in	22
Εικόνα 4: Σελίδα Εγγραφής	24
Εικόνα 5: Homepage.....	24
Εικόνα 6: Search Users.....	25
Εικόνα 7: Αποτέλεσμα Αναζήτησης Nickname με το αγγλικό γράμμα «n»	26
Εικόνα 8: Free Games.....	27
Εικόνα 9: Game I Play.....	27
Εικόνα 10: Το δικό μας Profile	28
Εικόνα 11: Το Profile ενός άλλου Χρήστη	29
Εικόνα 12: Τα παιχνίδια που παίζει ένας άλλος Χρήστης	29
Εικόνα 13: Edit Profile	30
Εικόνα 14: Το «Delete My Account» στο κάτω μέρος του «Edit Profile»	30
Εικόνα 15: Chat.....	31
Εικόνα 16: Η φόρμα για την δημιουργία και αποστολή μηνύματος	31
Εικόνα 17: Μία Δημοσίευση	31
Εικόνα 18: Μία Δημοσίευση του Χρήστη που είναι συνδεδεμένος στο Game Society .	32
Εικόνα 19: Η σελίδα μίας Δημοσίευσης	32
Εικόνα 20: Comment Section	33
Εικόνα 21: See who liked the post.....	33
Εικόνα 22: Show All Users tab	34
Εικόνα 23: Show All Users	35
Εικόνα 24: Profile Viewing – By Admins	35
Εικόνα 25: Το μονοπάτι για τις σελίδες του Συστήματος.....	36
Εικόνα 26: welcome	37
Εικόνα 27: Άνοιγμα της Βάσεις.....	37
Εικόνα 28: Χτίσιμο της Βάσης	37
Εικόνα 29: Έλεγχος ήδη συνδεδεμένου Χρήστη	38
Εικόνα 30: Έλεγχος ταυτοποίησης	38
Εικόνα 31: Οι κλείσεις του register.jsp	39
Εικόνα 32: Εισαγωγή νέου Χρήστη στην Βάση.....	39
Εικόνα 33: HomePage.....	40
Εικόνα 34: Απόκτηση στοιχείων του συνδεδεμένου Χρήστη	41
Εικόνα 35: Web Services του HomePage.jsp	41
Εικόνα 36: Βάση Δεδομένων πριν και μετά την διαγραφή Δημοσίευσης	42
Εικόνα 37: Κλήση του getAllUsers.....	42
Εικόνα 38: Τα Services της σελίδας Games I Play	43
Εικόνα 39: Εισαγωγή και Διαγραφή ενός παιχνιδιού.....	43
Εικόνα 40: Διαγραφή Δημοσίευσης από την σελίδα της.....	44
Εικόνα 41: Διαγραφή ενός Σχολίου.....	44
Εικόνα 42: Προσθήκη και Διαγραφή Like.....	45
Εικόνα 43: Προσθήκη Σχολίου	45
Εικόνα 44: Προσθήκη Σχολίου στην Βάση Δεδομένων	45
Εικόνα 45: Εύρεση της ως προς προβολή Δημοσίευσης	46
Εικόνα 46: Απόκτηση Σχολίων και ανίχνευση Like του Χρήστη	46
Εικόνα 47: Απόκτηση πλήθους Like μίας Δημοσίευσης	46
Εικόνα 48: Κλήση όλων των Χρηστών που έχουν κάνει Like σε μία Δημοσίευση	47
Εικόνα 49: Σελίδες των Προφίλ	47

Εικόνα 50: Διαγραφή μίας Δημοσίευσης μέσω ενός προφίλ	48
Εικόνα 51: Διαγραφή ενός Χρήστη μέσω του Προφίλ του.....	48
Εικόνα 52: Διαγραφή ενός Χρήστη από την Βάση Δεδομένων	48
Εικόνα 53: Κώδικας για αλλαγή δικαιωμάτων	49
Εικόνα 54: Αλλαγή δικαιωμάτων Χρήστη στην Βάση.....	49
Εικόνα 55: Κώδικας εισαγωγής Δημοσίευσης.....	50
Εικόνα 56: Εισαγωγή Δημοσίευσης στην Βάση Δεδομένων	50
Εικόνα 57: Κώδικας αλλαγής στοιχείων Χρήστη	50
Εικόνα 58: Κώδικας απόκτησης μηνυμάτων	51
Εικόνα 59: Web Service για την αποστολή μηνυμάτων	51
Εικόνα 60: Αποστολή μηνύματος ανάμεσα σε δύο Χρήστες	51
Εικόνα 61: Web Service για την απόκτηση των παιχνιδιών ενός Χρήστη	52
Εικόνα 62: Απόκτηση Χρηστών Ψευδώνυμο παρόμοιο με την είσοδο.....	52
Εικόνα 63: Απόκτηση Χρηστών που παίζουν κάποιο δοθέν παιχνίδι.....	52
Εικόνα 64: Third Party Web Services	53
Εικόνα 65: Σωστή Χρήση του Free To Game	54
Εικόνα 66: Σωστή Χρήση του Free To Game For Platforms	54
Εικόνα 67: Λάθος Χρήση του Free To Game For Platforms	54

1. Θεματολογία Συστήματος

1.1. Πρόβλημα ως προς επίλυση

Επάνω στα πλαίσια της απαλλακτικής εργασίας του μαθήματος Δικτυοκεντρικά Πληροφοριακά Συστήματα 2020-2021, έλαβα την απόφαση να κατασκευάσω ένα Πληροφοριακό Σύστημα τύπου Social Media. Το Social Media αυτό αναφέρεται στους ανθρώπους οι οποίοι δυσκολεύονται να κοινωνικοποιηθούν και ως αποτέλεσμα δεν μπορούν να χαρούν στιγμές εντός των ηλεκτρονικών παιχνιδιών με μία παρέα.

Το Πληροφοριακό Σύστημα ονομάστηκε Game Society και έχει ως σκοπό να φέρει κοντά αυτούς τους ανθρώπους, ως μεσολαβητής. Λόγο αυτού, δεν θα έχουν ανάγκη την αλλαγή του χαρακτήρα τους και θα είναι ικανοί να κοινωνικοποιηθούν εύκολα και γρήγορα, ώστε να ζήσουν πολλές στιγμές χαράς μέσα από τους χρήστες του Game Society.

Το Game Society έχει χτιστεί με την χρήση RESTful Web Services και την Βάση Δεδομένων MySQL.

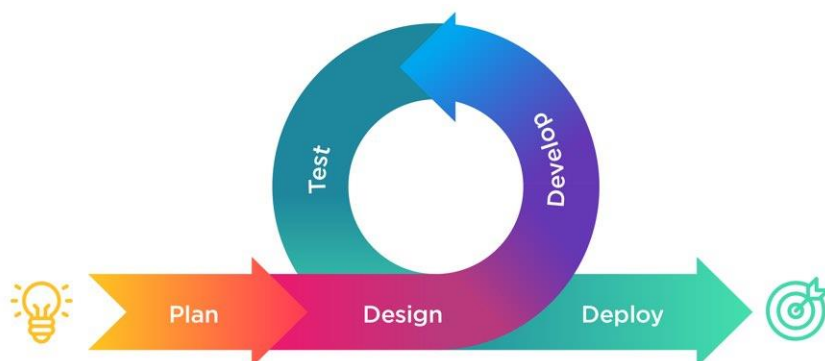
1.2. Διαδικασία Υλοποίησης Συστήματος

Για την υλοποίηση του Συστήματος ακολουθήθηκε η παρακάτω διαδικασία:

- Απόφαση του προβλήματος ως προς επίλυση
- Απόφαση των λειτουργιών που θα πρέπει να πληροί
- Σχεδίαση του σχήματος της Βάσης Δεδομένων
- Δημιουργία των RESTful Web Services με την χρήση του εργαλείου Postman
- Κατασκευή του UI για το Πληροφοριακό Σύστημα

Η μεθοδολογία που ακολουθήθηκε ήταν τύπου Agile καθώς σε κάθε ένα από τα παραπάνω βήματα (Sprints) της διαδικασίας υλοποίησης γινόταν:

- Planning
- Designing
- Building
- Testing
- Reviewing + Deploying



Εικόνα 1: Ένα Sprint στην Μεθοδολογία Agile

2. Υλοποίηση Συστήματος

2.1. Αρχιτεκτονική

Το Σύστημα έχει δημιουργηθεί χρησιμοποιώντας Service-Oriented Architecture (SOA) με σκοπό την επαναχρησιμοποίηση υπηρεσιών, την καλύτερη συντήρηση και την παράλληλη ανάπτυξη των Υπηρεσιών του. Με την χρήση του SOA οι πολύπλοκες υπηρεσίες σπάνε σε μικρότερες και απομονωμένες – ανεξάρτητες υπηρεσίες. Συνεπώς η δημιουργία τους καταλήγει ταχύτερη και καλά οργανωμένη.

Η κατασκευή του Πληροφοριακού Συστήματος Βασίστηκε REST-based Υπηρεσίες Ιστού και συγκεκριμένα σε RESTful Web Services για λόγους ταχύτητας υλοποίησης καθώς και εξάσκησης επάνω σε αυτό το είδος των υπηρεσιών ιστού. Λόγο της απαίτησης ασφαλείας του εν λόγω Πληροφοριακού Συστήματος, θα ήταν χρήσιμο μελλοντικά να αναδημιουργηθεί χρησιμοποιώντας SOAP-based Υπηρεσίες Ιστού.

2.2. Υλοποιημένες Υπηρεσίες Ιστού

Όλα τα Web Services που έχουν κατασκευαστεί είναι RESTful και υπάρχουν εντός του src του project. Βρίσκονται μέσα στο πακέτο my.restful.web.services, στην κλάση GameSociety.

Στην συνέχεια κάθε Web Service θα παρουσιαστεί εντός ενός πίνακα με τα εξής πεδία:

- Είσοδος: Στο οποίο θα υπάρχουν όλα τα ορίσματα που χρειάζεται το κάθε Web Service.
- Επαναχρησιμοποίηση: **(NAI | OXI)** Στο οποίο θα αναφέρετε εάν αυτό το Web Service είναι από αυτά τα οποία επαναχρησιμοποιούνται εντός του Συστήματος.
- Χρησιμοποιείτε από: **(System | Users | Admins | Users/Admins)** Σε αυτό το πεδίο θα εμφανίζεται:
 - Εάν το Web Services αυτό προορίζεται για την χρήση του αυτόματα από το σύστημα στο οποίο καλείτε **(System)**.
 - Εάν προορίζεται για χρήση από όλους τους Χρήστες του Συστήματος **(Users)**.
 - Εάν προορίζεται για χρήση από τους Διαχειριστές του Συστήματος μόνο **(Admins)**.
 - Εάν προορίζεται για όλους τους Χρήστες αλλά προσθέτει μία επιπλέον λειτουργία στους Διαχειριστές **(Users/Admins)**.
- HTTP METHOD: Το πεδίο αυτό αναφέρετε στο είδος της HTTP μεθόδου του Web Service.
- Επιστρέφει: Το είδος της πληροφορίας που επιστρέφει αυτό το Web Service.
- Περιγραφή: Η λειτουργία του Web Service.
- Endpoint: Το endpoint στο οποίο μπορεί να κληθεί το Web Service.
- Χρήση στο Σύστημα: Τα endpoints που χρησιμοποιούν το Web Service.
- Σχόλιο: Επιπλέον χρήσιμες πληροφορίες.

2.2.1. Υπηρεσίες του Συστήματος

testForDB

Είσοδος	Επαναχρησιμοποίηση	Χρησιμοποιείται από	HTTP METHOD
-	NAI	System	GET
Επιστρέφει			
-			
Περιγραφή			
Χτίζει το Database στην περίπτωση που δεν υπάρχει εξαρχής. Εισάγει ως default Διαχειριστή έναν Χρήστη με nickname «admin» στην περίπτωση που δεν υπάρχει ήδη ένας Διαχειριστής μέσα.			
Endpoint			
http://localhost:8080/GameSociety/rest/GameSociety/testForDB			
Χρήση στο Σύστημα			
Σε όλα τα endpoints			
Σχόλιο			
Αυτό το Web Service πρέπει να κληθεί ώστε να γίνει η σύνδεση με την βάση δεδομένων και τελικά να μπορούν να λειτουργήσουν όλα τα υπόλοιπα Web Services του Συστήματος.			

testForUser

Είσοδος	Επαναχρησιμοποίηση	Χρησιμοποιείται από	HTTP METHOD
User's nickname	OXI	System	GET
Επιστρέφει			
JSON Object			
Περιγραφή			
Επιστρέφει τον Χρήστη που κατέχει το εισαγόμενο nickname. Χρησιμοποιείται ως έλεγχος ώστε να μην εισάγονται Χρήστες με ίδιο ψευδώνυμο.			
Endpoint			
http://localhost:8080/GameSociety/rest/GameSociety/testForUser/{nickName}			
Χρήση στο Σύστημα			
http://localhost:8080/GameSociety/Pages/welcome/register.jsp			
Σχόλιο			
Το συγκεκριμένο Web Service κάνει κλείσει του testForDB και συνεπώς χτίζει (στην περίπτωση που δεν υπάρχει) και συνδέει την Βάση Δεδομένων. Όπως ακόμα αρχικοποιεί τον Διαχειριστή «admin» στην περίπτωση που δεν υπάρχει άλλος Διαχειριστής εντός του Πληροφοριακού Συστήματος. Χρησιμοποιείτε για την εισαγωγή νέων Χρηστών εντός του Πληροφοριακού Συστήματος.			

testForEmail

Είσοδος	Επαναχρησιμοποίηση	Χρησιμοποιείται από	HTTP METHOD
User's email	OXI	System	GET
Επιστρέφει			
JSON Object			
Περιγραφή			
Επιστρέφει τον Χρήστη που κατέχει το εισαγόμενο email. Χρησιμοποιείται ως έλεγχος ώστε να μην εισάγονται Χρήστες με ίδιο email.			
Endpoint			
http://localhost:8080/GameSociety/rest/GameSociety/testForEmail/{email}			
Χρήση στο Σύστημα			
http://localhost:8080/GameSociety/Pages/welcome/register.jsp			
Σχόλιο			
Το συγκεκριμένο Web Service κάνει κλείσει του testForDB και συνεπώς χτίζει (στην περίπτωση που δεν υπάρχει) και συνδέει την Βάση Δεδομένων. Όπως ακόμα αρχικοποιεί τον Διαχειριστή «admin» στην περίπτωση που δεν υπάρχει άλλος Διαχειριστής εντός του Πληροφοριακού Συστήματος. Χρησιμοποιείτε για την εισαγωγή νέων Χρηστών εντός του Πληροφοριακού Συστήματος.			

2.2.2. Υπηρεσίες των Χρηστών του Συστήματος

addAUser

Είσοδος	Επαναχρησιμοποίηση	Χρησιμοποιείται από	HTTP METHOD
User's name	OXI	System	POST
User's surname			
User's nickname			
User's privileges			
User's password			
User's email			
Επιστρέφει			
-			
Περιγραφή			
Προσθήκη ενός νέου χρήστη εντός του Πληροφοριακού Συστήματος.			
Endpoint			
http://localhost:8080/GameSociety/rest/GameSociety/addAUser/{name}/{surname}/{nickName}/{isAdmin}/{password}/{email}			
Χρήση στο Σύστημα			
http://localhost:8080/GameSociety/Pages/welcome/register.jsp			
Σχόλιο			
-			

deleteAUser

Είσοδος	Επαναχρησιμοποίηση	Χρησιμοποιείται από	HTTP METHOD
User's ID	NAI	Users/Admins	DELETE
Επιστρέφει			
-			
Περιγραφή			
Διαγράφει έναν χρήστη από το Πληροφοριακό Σύστημα. Στην περίπτωση που έχει διαγραφτεί ο τελευταίος Διαχειριστής, τότε εισάγει ως default Διαχειριστή έναν Χρήστη με nickname «admin».			
Endpoint			
http://localhost:8080/GameSociety/rest/GameSociety/deleteAUser/{userID}			
Χρήση στο Σύστημα			
http://localhost:8080/GameSociety/Pages/HomePage/Profile/UserProfile.jsp			
http://localhost:8080/GameSociety/Pages/HomePage/Profile/EditProfile.jsp			
Σχόλιο			
Χρησιμοποιείτε στις ρυθμίσεις του προφίλ του κάθε Χρήστη στην περίπτωση που θέλει να διαγράψει τον εαυτό του. Χρησιμοποιείτε επίσης ως μία επιπλέον λειτουργία για τους Διαχειριστές, ώστε να μπορούν από το προφίλ ενός απλού Χρήστη να τον διαγράψουν από το Πληροφοριακό Σύστημα.			

getUsersByNickName

Είσοδος	Επαναχρησιμοποίηση	Χρησιμοποιείται από	HTTP METHOD
User's nickname	OXI	Users	GET
Επιστρέφει			
JSON Array			
Περιγραφή			
Επιστρέφει όλους τους χρήστες που το ψευδώνυμο τους περιέχει το εισαγόμενο αλφαριθμητικό.			
Endpoint			
http://localhost:8080/GameSociety/rest/GameSociety/getUsersByNickName/{nickName}			
Χρήση στο Σύστημα			
http://localhost:8080/GameSociety/Pages/HomePage/Search/SearchSelection.jsp			
Σχόλιο			
Αν το ψευδώνυμο κάποιου χρήστη είναι Παναγιώτης, και δώσουμε ως είσοδο το «v», τότε θα επιστρέφει και τον Παναγιώτη, λόγω του -v- εντός του αλφαριθμητικού αυτού.			

getUser

Είσοδος	Επαναχρησιμοποίηση	Χρησιμοποιείται από	HTTP METHOD
User's ID	NAI	System	GET
Επιστρέφει			
JSON Object			
Περιγραφή			
Επιστρέφει τα στοιχεία του Χρήστη που κατέχει το εισαγόμενο ID.			
Endpoint			
http://localhost:8080/GameSociety/rest/GameSociety/getUser/{userID}			
Χρήση στο Σύστημα - σε όλα εκτός από			
http://localhost:8080/GameSociety/Pages/HomePage/LogOut.jsp			
http://localhost:8080/GameSociety/Pages/welcome/login.jsp			
http://localhost:8080/GameSociety/Pages/welcome/register.jsp			
Σχόλιο			
Αυτό το Web Service υπάρχει κυρίως για την παρουσίαση του Χρήστη που είναι Logged in στο Social Media. Χρησιμοποιείται όμως επίσης, για την απόκτηση πληροφοριών για χρήστες οι οποίοι δεν είναι logged in στον browser που χρησιμοποιεί ο Χρήστης.			

changePrivileges

Είσοδος	Επαναχρησιμοποίηση	Χρησιμοποιείται από	HTTP METHOD
User's ID	OXI	Admins	PUT
Επιστρέφει			
-			
Περιγραφή			
Αλλάζει τα δικαιώματα ενός Χρήστη, από Διαχειριστή σε απλό Χρήστη και το Αντίθετο. Ακόμα, στην περίπτωση που δεν υπάρχει πλέον Διαχειριστής μετά την χρήση του, τότε κατασκευάζει τον default Διαχειριστή με ψευδώνυμο «admin».			
Endpoint			
http://localhost:8080/GameSociety/rest/GameSociety/changePrivileges/{userID}			
Χρήση στο Σύστημα			
http://localhost:8080/GameSociety/Pages/HomePage/Profile/UserProfile.jsp			
Σχόλιο			
Αυτήν την λειτουργία μπορούν να την χρησιμοποιήσουν μόνο οι Διαχειριστές του Συστήματος, με σκοπό την δημιουργία νέων Διαχειριστών, ή τον υποβιβασμό μη αναγκαίων πλέον διαχειριστών. Το Πληροφοριακό Σύστημα, έχει κατασκευαστεί ώστε ένας Διαχειριστής να μην μπορεί να υποβιβάσει τον εαυτό του.			

getAllUsers

Είσοδος	Επαναχρησιμοποίηση	Χρησιμοποιείται από	HTTP METHOD
-	OXI	Admins	GET
Επιστρέφει			
JSON Array			
Περιγραφή			
Επιστρέφει όλους τους χρήστες του Πληροφοριακού Συστήματος.			
Endpoint			
http://localhost:8080/GameSociety/rest/GameSociety/getAllUsers			
Χρήση στο Σύστημα			
http://localhost:8080/GameSociety/Pages/HomePage/Admin/ShowAllUsers.jsp			
Σχόλιο			
Αυτήν την λειτουργία μπορούν να την χρησιμοποιήσουν μόνο οι Διαχειριστές του Συστήματος, με σκοπό την εύκολη εύρεση των υπόλοιπων Χρηστών ώστε να μπορούν να πειράξουν τα δικαιώματα τους ή πιθανώς να διαγράψουν προβληματικό υλικό από αυτούς.			

changeProfileInfo

Είσοδος	Επαναχρησιμοποίηση	Χρησιμοποιείται από	HTTP METHOD
User's ID	OXI	Users	PUT
User's name			
User's surname			
User's profile Picture Path			
User's nickname			
User's password			
User's email			
Επιστρέφει			
-			
Περιγραφή			
Αλλάζει τον κωδικό, το όνομα και το επίθετο ενός χρήστη με τα εισαγόμενα στοιχεία του Service.			
Endpoint			
http://localhost:8080/GameSociety/rest/GameSociety/changeProfileInfo/{userID}/{name}/{surname}/{profilePicturePath}/{nickName}/{password}/{email}			
Χρήση στο Σύστημα			
http://localhost:8080/GameSociety/Pages/HomePage/Profile/EditProfile.jsp			
Σχόλιο			
Τα υπόλοιπα στοιχεία δεν δίνονται μέσω του UI ως επιλογή για update καθώς θα παρέμβαιναν στην ασφάλεια του Συστήματος αλλά και θα έφερναν σε αντίφαση κανονισμούς, όπως ο κανονισμός ότι δύο Χρήστες δεν μπορούν να έχουν το ίδιο ψευδώνυμο.			

2.2.3. Υπηρεσίες των Λιστών Παιχνιδιών

addAUserGame

Είσοδος	Επαναχρησιμοποίηση	Χρησιμοποιείται από	HTTP METHOD
User's ID	OXI	Users	POST
Game's name			
Επιστρέφει			
-			
Περιγραφή			
Προσθέτει στην Λίστα έναν τίτλο παιχνιδιού στην λίστα κάποιου Χρήστη.			
Endpoint			
http://localhost:8080/GameSociety/rest/GameSociety/addAUserGame/{userID}/{name}			
Χρήση στο Σύστημα			
http://localhost:8080/GameSociety/Pages/HomePage/MyGames/MyGames.jsp			
Σχόλιο			
Τα παιχνίδια αυτά χρησιμοποιούνται για να μπορούν οι Χρήστες να βρουν άτομα με κοινά ενδιαφέροντα εντός του Social Media.			

deleteAUserGame

Είσοδος	Επαναχρησιμοποίηση	Χρησιμοποιείται από	HTTP METHOD
User's ID	OXI	Users	DELETE
Game's ID			
Επιστρέφει			
-			
Περιγραφή			
Διαγράφει ένα παιχνίδι από την Λίστα ενός Χρήστη.			
Endpoint			
http://localhost:8080/GameSociety/rest/GameSociety/deleteAUserGame/{userID}/{gameID}			
Χρήση στο Σύστημα			
http://localhost:8080/GameSociety/Pages/HomePage/MyGames/MyGames.jsp			
Σχόλιο			
Κάθε Χρήστης θα μπορεί να διαγράφει από την Λίστα παιχνίδια τα οποία δεν παίζει πιθανώς πλέον.			

getAllUserGames

Είσοδος	Επαναχρησιμοποίηση	Χρησιμοποιείται από	HTTP METHOD
User's ID	NAI	Users	GET
Επιστρέφει			
JSON Array			
Περιγραφή			
Επιστρέφει όλη την λίστα με τα παιχνίδια ενός χρήστη.			
Endpoint			
http://localhost:8080/GameSociety/rest/GameSociety/getAllUserGames/{userID}			
Χρήση στο Σύστημα			
http://localhost:8080/GameSociety/Pages/HomePage/MyGames/MyGames.jsp			
http://localhost:8080/GameSociety/Pages/HomePage/Profile/OtherUser/SeeGames.jsp			
Σχόλιο			
Η λειτουργία αυτή, εκτός για την εμφάνιση των παιχνιδιών που έχει προσθέσει ο συνδεδεμένος Χρήστης στην λίστα του, χρησιμοποιείται και για την εμφάνιση των παιχνιδιών των λιστών των άλλων Χρηστών μέσα στο Σύστημα.			

getUsersPlayingTheGame

Είσοδος	Επαναχρησιμοποίηση	Χρησιμοποιείται από	HTTP METHOD
Game's name	OXI	Users	GET
Επιστρέφει			
JSON Array			
Περιγραφή			
Επιστρέφει όλους τους Χρήστες που έχουν βάλει στην λίστα τους τον τίτλο παιχνιδιού που δίνεται ως εισαγωγή.			
Endpoint			
http://localhost:8080/GameSociety/rest/GameSociety/getUsersPlayingTheGame/{name}			
Χρήση στο Σύστημα			
http://localhost:8080/GameSociety/Pages/HomePage/Search/SearchSelection.jsp			
Σχόλιο			
Το Web Service αυτό, χρησιμοποιείτε για την εύρεση ατόμων με κοινά ενδιαφέροντα, συνεπώς φέρνει κοντά τους Χρήστες με αποτέλεσμα την ευκολότερη εύρεση παρέας για το παιχνίδι που αρέσει στον κάθε Χρήστη.			

2.2.4. Υπηρεσίες των Δημοσιεύσεων

addAPost

Είσοδος	Επαναχρησιμοποίηση	Χρησιμοποιείται από	HTTP METHOD
User's ID	OXI	Users	Post
Post's content			
Post's date and time			
Επιστρέφει			
-			
Περιγραφή			
Προσθέτει μία νέα Δημοσίευση ενός χρήστη στο Πληροφοριακό Σύστημα.			
Endpoint			
http://localhost:8080/GameSociety/rest/GameSociety/addAPost/{userID}/{content}/{dateTime}			
Χρήση στο Σύστημα			
http://localhost:8080/GameSociety/Pages/HomePage/Profile/UserProfile.jsp			
Σχόλιο			
Κάθε Χρήστης θα μέσα από το προφίλ του να ανεβάζει Δημοσιεύσεις, τις οποίες θα μπορούν να δουν όλοι οι Χρήστες του Game Society.			

getAllPosts

Είσοδος	Επαναχρησιμοποίηση	Χρησιμοποιείται από	HTTP METHOD
User's ID	OXI	Users	GET
Επιστρέφει			
JSON Array			
Περιγραφή			
Επιστρέφει όλες τις Δημοσιεύσεις ενός Χρήστη ταξινομημένες σε φθίνουσα σειρά σε σχέση με την στιγμή που ανέβηκαν..			
Endpoint			
http://localhost:8080/GameSociety/rest/GameSociety/getAllPosts/{userID}			
Χρήση στο Σύστημα			
http://localhost:8080/GameSociety/Pages/HomePage/Profile/UserProfile.jsp			
Σχόλιο			
Ο κάθε Χρήστης θα έχει όλες του τις Δημοσιεύσεις μαζεμένες στο προφίλ του.			

getHomeScreenPosts

Είσοδος	Επαναχρησιμοποίηση	Χρησιμοποιείται από	HTTP METHOD
-	NAI	Users	GET
Επιστρέφει			
JSON Array			
Περιγραφή			
Επιστρέφει όλες τις Δημοσιεύσεις όλες τις Δημοσιεύσεις όλων των Χρηστών εντός του Πληροφοριακού Συστήματος ταξινομημένες σε φθίνουσα σειρά σε σχέση με την στιγμή που ανέβηκαν.			
Endpoint			
http://localhost:8080/GameSociety/rest/GameSociety/getHomeScreenPosts			
Χρήση στο Σύστημα			
http://localhost:8080/GameSociety/Pages/HomePage/HomePage.jsp			
http://localhost:8080/GameSociety/Pages/HomePage/Post/Post.jsp			
Σχόλιο			
Αυτή η λειτουργία θα χρησιμοποιείται ως Home Page για το Game Society. Όστε όλοι οι Χρήστες να έχουν την ευκαιρία να φανούν στο υπόλοιπο κοινό και να κοινωνικοποιηθούν. Επίσης θα χρησιμοποιείται για την εύρεση μίας Δημοσίευσης που θέλει να δει κάποιο Χρήστης συγκεκριμένα.			

deletePost

Είσοδος	Επαναχρησιμοποίηση	Χρησιμοποιείται από	HTTP METHOD
Post's ID	NAI	Users/Admins	DELETE
Επιστρέφει			
-			
Περιγραφή			
Διαγράφει την Δημοσίευση η οποία έχει το ID το οποίο δίνεται ως είσοδος.			
Endpoint			
http://localhost:8080/GameSociety/rest/GameSociety/deletePost/{postID}			
Χρήση στο Σύστημα			
http://localhost:8080/GameSociety/Pages/HomePage/HomePage.jsp			
http://localhost:8080/GameSociety/Pages/HomePage/Profile/UserProfile.jsp			
http://localhost:8080/GameSociety/Pages/HomePage/Post/Post.jsp			
Σχόλιο			
Η λειτουργία της διαγραφής μίας Δημοσίευσης επιτρέπεται στους απλούς Χρήστες οπουδήποτε συναντάνε δικιά τους Δημοσίευση. Όμως οι Διαχειριστές έχουν αυτήν την επιλογή, σε όλες τις Δημοσιεύσεις του Πληροφοριακού Συστήματος.			

2.2.5. Υπηρεσίες των Σχολίων

addAComment

Είσοδος	Επαναχρησιμοποίηση	Χρησιμοποιείται από	HTTP METHOD
User's ID (commenting)	OXI	Users	Post
Post's ID			
Comment's text			
Comment's date and time			
Επιστρέφει			
-			
Περιγραφή			
Προσθέτει ένα comment, ενός χρήστη, σε μία Δημοσίευση.			
Endpoint			
http://localhost:8080/GameSociety/rest/GameSociety/addAComment/{userID}/{postIdID}/{text}/{dateTime}			
Χρήση στο Σύστημα			
http://localhost:8080/GameSociety/Pages/HomePage/Post/Post.jsp			
Σχόλιο			
Κάθε Χρήστης μπορεί να κάνει σχόλια σε οποιαδήποτε Δημοσίευση υπάρχει στο Game Society.			

deleteComment

Είσοδος	Επαναχρησιμοποίηση	Χρησιμοποιείται από	HTTP METHOD
Comment's ID	OXI	Users/Admins	DELETE
Επιστρέφει			
-			
Περιγραφή			
Διαγράφει ένα σχόλιο ενός χρήστη από μία Δημοσίευση.			
Endpoint			
http://localhost:8080/GameSociety/rest/GameSociety/deleteComment/{commentID}			
Χρήση στο Σύστημα			
http://localhost:8080/GameSociety/Pages/HomePage/Post/Post.jsp			
Σχόλιο			
Κάθε Χρήστης μπορεί να διαγράψει οποιοδήποτε δικό του Σχόλιο. Όμως ένας Διαχειριστής, μπορεί να διαγράψει οποιοδήποτε σχόλιο, οποιουδήποτε Χρήστη.			

showComments

Είσοδος	Επαναχρησιμοποίηση	Χρησιμοποιείται από	HTTP METHOD
Post's ID	OXI	Users	GET
Επιστρέφει			
JSON Array			
Περιγραφή			
Επιστρέφει όλα τα σχόλια μία Δημοσίευσης.			
Endpoint			
http://localhost:8080/GameSociety/rest/GameSociety/showComments/{postId}			
Χρήση στο Σύστημα			
http://localhost:8080/GameSociety/Pages/HomePage/Post/Post.jsp			
Σχόλιο			
Αυτή η λειτουργία χρησιμοποιείται ώστε οι Χρήστες να μπορούν να δουν τα σχόλια που υπάρχουν στις δημοσιεύσεις του Πληροφοριακού Συστήματος.			

2.2.6. Υπηρεσίες των Upvote (Likes)

likeOrDislike

Είσοδος	Επαναχρησιμοποίηση	Χρησιμοποιείται από	HTTP METHOD
User's ID (liking)	OXI	Users	POST
Post's ID			
Επιστρέφει			
-			
Περιγραφή			
Προσθέτει (εάν δε υπάρχει) ή αφαιρεί (εάν υπάρχει) το Like ενός Χρήστη σε μία Δημοσίευση.			
Endpoint			
http://localhost:8080/GameSociety/rest/GameSociety/likeOrDislike/{userID}/{postID}			
}			
Χρήση στο Σύστημα			
http://localhost:8080/GameSociety/Pages/HomePage/Post/Post.jsp			
Σχόλιο			
Κάθε χρήστης μπορεί να κάνει σε κάθε δημοσίευση Like, αλλά μπορεί και να το παίρνει πίσω.			

getAllUsersLiked

Είσοδος	Επαναχρησιμοποίηση	Χρησιμοποιείται από	HTTP METHOD
Post's ID	NAI	Users	GET
Επιστρέφει			
JSON Array			
Περιγραφή			
Επιστρέφει όλους τους Χρήστες που έχουν κάνει Like σε μία Δημοσίευση.			
Endpoint			
http://localhost:8080/GameSociety/rest/GameSociety/getAllUsersLiked/{postId}			
Χρήση στο Σύστημα			
http://localhost:8080/GameSociety/Pages/HomePage/Post/Post.jsp			
http://localhost:8080/GameSociety/Pages/HomePage/Post/Likes.jsp			
Σχόλιο			
Η λειτουργία αυτή, χρησιμοποιείται από το σύστημα στο Post.jsp ώστε να βρεθεί το αν ο χρήστης που είναι Logged in, έχει κάνει Like στην Δημοσίευση που εμφανίζεται ή όχι. Όμως χρησιμοποιείται και στο Likes.jsp ώστε να εμφανίσει όλους τους χρήστες που έχουν κάνει Like στην Δημοσίευση που κοίταζε μαζεμένους.			

getAmountOfLikes

Είσοδος	Επαναχρησιμοποίηση	Χρησιμοποιείται από	HTTP METHOD
Post's ID	OXI	Users	GET
Επιστρέφει			
JSON Object			
Περιγραφή			
Επιστρέφει το σύνολο των likes για μία Δημοσίευση.			
Endpoint			
http://localhost:8080/GameSociety/rest/GameSociety/getAmountOfLikes/{postId}			
Χρήση στο Σύστημα			
http://localhost:8080/GameSociety/Pages/HomePage/Post/Post.jsp			
Σχόλιο			
Αυτή η λειτουργία, χρησιμοποιείται ώστε ο Χρήστης να μην έχει ανάγκη να μετρήσει το σύνολο των ανθρώπων που τους άρεσε κάποια Δημοσίευση.			

2.2.7. Υπηρεσίες των Μηνυμάτων

sendMessage

Είσοδος	Επαναχρησιμοποίηση	Χρησιμοποιείται από	HTTP METHOD
User's ID (logged in)	OXI	Users	Post
User's ID (friend user)			
Message's content			
Message's date and time			
Επιστρέφει			
-			
Περιγραφή			
Στέλνει ένα μήνυμα από έναν Χρήστη σε Έναν Άλλον Χρήστη του Πληροφοριακού Συστήματος.			
Endpoint			
http://localhost:8080/GameSociety/rest/GameSociety/sendAMessage/{theUserID}/{friendUserID}/{content}/{dateTime}			
Χρήση στο Σύστημα			
http://localhost:8080/GameSociety/Pages/HomePage/Profile/OtherUser/Chat.jsp			
Σχόλιο			
Κάθε Χρήστης μπορεί να στείλει μηνύματα σε άλλους Χρήστες του Πληροφοριακού Συστήματος, μέσω του προφίλ των άλλων Χρηστών, με σκοπό την συνεννόηση τους σχετικά με τα σχέδια τους για τα κοινά τους ενδιαφέροντα ή ακόμα και για δικές του πιθανές ελεύθερες συζητήσεις.			

showMessages

Είσοδος	Επαναχρησιμοποίηση	Χρησιμοποιείται από	HTTP METHOD
User's ID (logged in)	OXI	Users	GET
User's ID (friend user)			
Επιστρέφει			
JSON Array			
Περιγραφή			
Επιστρέφει τα 10 τελευταία μηνύματα δύο Χρηστών του Πληροφοριακού Συστήματος, ταξινομημένα κατά φθίνουσα σειρά.			
Endpoint			
http://localhost:8080/GameSociety/rest/GameSociety/showMessages/{theUserID}/{friendUserID}			
Χρήση στο Σύστημα			
http://localhost:8080/GameSociety/Pages/HomePage/Profile/OtherUser/Chat.jsp			
Σχόλιο			
Κάθε Χρήστης θα μπορεί να βλέπει τα τελευταία του μηνύματα με άλλους Χρήστες του Game Society ώστε να μπορεί να θυμηθεί τι είχε υποθεί στην συζήτησή του.			

2.3. Εξωτερικές Υπηρεσίες Ιστού/ Μικροϋπηρεσίες

Η πηγή των εξωτερικών Υπηρεσιών που Χρησιμοποιούνται στο Σύστημα, είναι η παρακάτω:

<https://www.freetogame.com/api-doc>

Free To Game

Είσοδος	Επαναχρησιμοποίηση	Χρησιμοποιείται από	HTTP METHOD
-	OXI	Users	GET
Επιστρέφει			
JSON Array			
Περιγραφή			
Επιστρέφει όλα τα δωρεάν βίντεο-παιχνίδια			
Endpoint			
https://www.freetogame.com/api/games			
Χρήση στο Σύστημα			
http://localhost:8080/GameSociety/Pages/HomePage/ShowFreeGames/FreeGamesSelection.jsp			
Σχόλιο			
Αυτή η λειτουργία, υπάρχει ώστε οι Χρήστες να μπορούν να βρουν πιθανώς νέα και εύκολα προσβάσιμα παιχνίδια, καθώς αυτό είναι ένα από τα hobby τους, αφού συμμετέχουν στο Game Society.			

Free To Game For Platforms

Είσοδος	Επαναχρησιμοποίηση	Χρησιμοποιείται από	HTTP METHOD
Game's platform	OXI	Users	GET
Επιστρέφει			
JSON Array			
Περιγραφή			
Επιστρέφει όλα τα δωρεάν βίντεο-παιχνίδια για την πλατφόρμα η οποία δίνεται ως εισαγωγή			
Endpoint			
https://www.freetogame.com/api/games?platform={platform}			
Χρήση στο Σύστημα			
http://localhost:8080/GameSociety/Pages/HomePage/ShowFreeGames/FreeGamesSelection.jsp			
Σχόλιο			
Αυτή η λειτουργία, υπάρχει ώστε οι Χρήστες να μπορούν να βρουν πιθανώς νέα και εύκολα προσβάσιμα παιχνίδια, καθώς αυτό είναι ένα από τα hobby τους, αφού συμμετέχουν στο Game Society. Η διαφορά με την προηγούμενη λειτουργία είναι ότι πλέον μπορούν να ψάξουν πιο γρήγορα παιχνίδια τα οποία θα είναι επάνω στις προτιμήσεις τους. Στην περίπτωση που δεν βρεθεί κανένα παιχνίδι για την εισαγόμενη πλατφόρμα, τότε αυτό το Web Service θα επιστρέψει JSON Object.			

3. Εγχειρίδιο Χρήσης Συστήματος

3.1. Τεχνολογίες

3.1.1. Γλώσσες

Για την υλοποίηση του Πληροφοριακού Συστήματος χρησιμοποιήθηκαν οι παρακάτω τεχνολογίες:

Java Server Pages (JSP)

Χρησιμοποιήθηκαν σελίδες .jsp ώστε να παραχθεί το front-end κομμάτι του πληροφοριακού συστήματος. Αυτές οι σελίδες περιέχουν κώδικα html και java, καθώς κάνουν κλείσεις σε αρχεία .css.

Java

Χρησιμοποιήθηκε η γλώσσα Java με σκοπό την σύνδεση της Βάσης Δεδομένων και την δημιουργία αλλά και την κλήση των διαφόρων Web Services που υπάρχουν εντός του Πληροφοριακού Συστήματος.

Hypertext Markup Language (HTML)

Έγινε χρήσης της γλώσσας html με σκοπό την εμφάνιση πληροφορίας στους Users του Συστήματος.

Cascading Style Sheets (CSS)

Η γλώσσα αυτή αποτελεί το στιλιστικό κομμάτι του πληροφοριακού συστήματος.

3.1.2. Βιβλιοθήκες

Στο Game Society χρησιμοποιήθηκαν διάφορες βιβλιοθήκες ώστε να ολοκληρωθεί το Πληροφοριακό Σύστημα.

*java.sql.**

Η βιβλιοθήκη αυτή χρησιμοποιήθηκε με σκοπό την δημιουργία SQL Queries, όπως και SQL Statements και την εκτέλεση τους στα υλοποιημένα Web Services.

*java.ws.rs.**

Αυτή η βιβλιοθήκη χρησιμοποιήθηκε για την κατασκευή των endpoints των υλοποιημένων Web Services του Πληροφοριακού Συστήματος. Επίσης χρησιμοποιήθηκε για την δημιουργία java annotations για την παροχή metadata.

*org.json.**

Έγινε χρήση αυτής της βιβλιοθήκης, ώστε να μπορούν να παρέχονται JSON αντικείμενα και πίνακες μέσω των Web Services για την καλύτερη οργάνωση και διαχείριση της πληροφορίας που μεταφέρεται μέσω αυτών.

java.time.LocalDateTime

Αυτή η βιβλιοθήκη χρησιμοποιήθηκε ώστε να βρίσκεται η πραγματική ώρα του κόσμου, ώστε να προστίθεται σαν δεδομένο σε σχόλια και δημοσιεύσεις των Χρηστών.

java.time.format.DateTimeFormatter

Η συγκεκριμένη βιβλιοθήκη υπάρχει για την διαχείριση της μορφής της ώρας και ημερομηνίας που λαμβάνεται από την βιβλιοθήκη «java.time.LocalDateTime».

3.1.3. Προγράμματα

Eclipse IDE for Enterprise Java Developers

Χρησιμοποιήθηκε το Eclipse για την Δημιουργία Dynamic Web Project

Postman

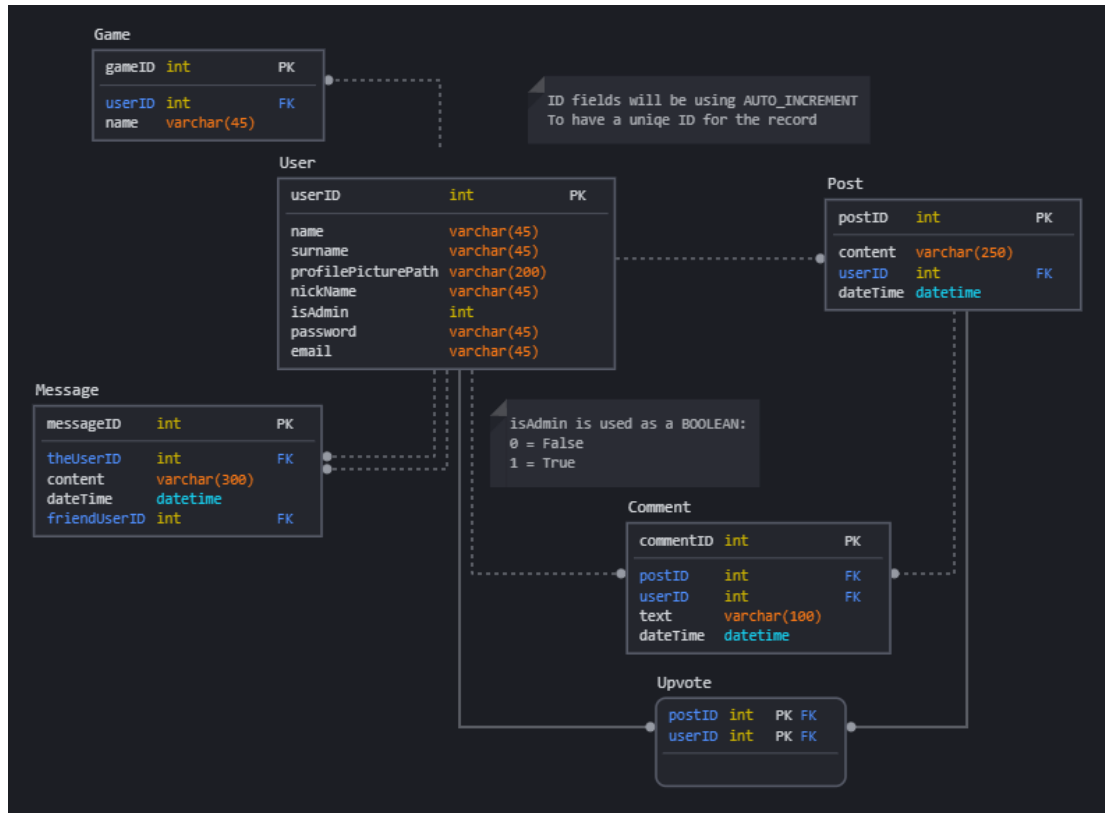
Χρησιμοποιήθηκε το Postman για τον έλεγχο της σωστής λειτουργίας των υλοποιημένων Web Services αλλά και των Third Party Web Services.

MySQL Workbench

Χρησιμοποιήθηκε το Workbench της MySQL για λόγους ταχύτητας και ευκολίας στον έλεγχο της σωστής λειτουργίας της Βάσης σε συμφωνία με τα υλοποιημένα Web Services.

3.2. Βάση Δεδομένων – MySQL

Ως Βάση Δεδομένων για το Game Society χρησιμοποιήθηκε η MySQL. Επιλέχθηκε η MySQL σε σχέση με την MongoDB καθώς με την χρήση της υπάρχει SCHEMA και είναι δυνατή η πρόβλεψη των πεδίων των records που δίνονται ως απάντηση στα Queries. Η Βάση Δεδομένων του Πληροφοριακού Συστήματος φαίνεται στην Εικόνα 2.



Εικόνα 2: Game Society Database (gamesocietydb)

Παρακάτω γίνεται μία εξήγηση ως προς το τι αναπαριστάτε στην Εικόνα 2

Σε ορθογώνιο παραλληλόγραμμα φαίνονται τα Tables της Βάσης και πάνω αριστερά τους εμφανίζεται το όνομα του αντίστοιχου Table (Τα παραλληλόγραμμα που δεν έχουν όνομα είναι notes). Κάθε table χωρίζεται σε δύο κομμάτια, τα primary keys και τα υπόλοιπα παιδιά του. Στα αριστερά των Tables φαίνονται τα ονόματα των πεδίων και στα δεξιά τους οι τύποι τους αντίστοιχα. Οι γραμμές υποδηλώνουν σχέση 1:N – όπου N είναι η μεριά που υπάρχει άσπρη κουκίδα. Οι συνεχόμενες γραμμές σημαίνουν ότι το εξωτερικό κλειδί συμμετέχει στο μοναδικό κλειδί του πίνακα, ενώ οι διακεκομμένες σημαίνουν ότι το μοναδικό κλειδί δεν περιέχει το εξωτερικό κλειδί.

3.2.1. User

Οι Χρήστες του Πληροφοριακού Συστήματος θα έχουν τα εξής στοιχεία:

- userID: Ένα μοναδικό αναγνωριστικό για κάθε Χρήστη (Ακέραιος)
- name: Το όνομα του κάθε Χρήστη (Αλφαριθμητικό – μεγέθους 45)
- surname: Το επώνυμο του κάθε Χρήστη (Αλφαριθμητικό – μεγέθους 45)
- profilePicturePath: Το path της φωτογραφίας προφίλ του κάθε Χρήστη (Αλφαριθμητικό – μεγέθους 200)
- nickName: Το ψευδώνυμο του κάθε Χρήστη (Αλφαριθμητικό – μεγέθους 45)
- isAdmin: Μία ένδειξη που υποδεικνύει αν ο Χρήστης είναι Διαχειριστής ή όχι (Ακέραιος 0/1)
- password: Ο κωδικός του κάθε Χρήστη για το Σύστημα (Αλφαριθμητικό – μεγέθους 45)
- email: Το email του κάθε Χρήστη (Αλφαριθμητικό – μεγέθους 45)

3.2.2. Game

Κάθε Χρήστης μπορεί να έχει πολλά παιχνίδια. Κάθε παιχνίδι απαρτίζεται από:

- gameId: Ένα μοναδικό αναγνωριστικό για κάθε Παιχνίδι (Ακέραιος)
- userID: Το μοναδικό αναγνωριστικό του Χρήστη που έχει εισάγει αυτό το παιχνίδι – **foreign key** (Ακέραιος)
- name: Ο τίτλος του εκάστοτε παιχνιδιού (Αλφαριθμητικό – μεγέθους 45)

3.2.3. Message

Κάθε Χρήστης μπορεί να στέλνει πολλά μηνύματα. Κάθε Χρήστης μπορεί να δέχεται πολλά μηνύματα. Το κάθε μήνυμα περιέχει τις εξής πληροφορίες:

- messageId: Ένα μοναδικό αναγνωριστικό για κάθε Μήνυμα (Ακέραιος)
- content: Το περιεχόμενο του Μηνύματος (Αλφαριθμητικό – μεγέθους 300)
- dateTime: Η ώρα και η μέρα που στάλθηκε το μήνυμα από τον αποστολέα (Datetime)
- theUserID: Το μοναδικό αναγνωριστικό του Χρήστη που έχει στείλει αυτό το Μήνυμα – **foreign key** (Ακέραιος)
- friendUserID: Το μοναδικό αναγνωριστικό του Χρήστη που λαμβάνει αυτό το Μήνυμα – **foreign key** (Ακέραιος)

3.2.4. Post

Κάθε Χρήστης μπορεί να ανεβάζει πολλές Δημοσιεύσεις. Οπότε οι Δημοσιεύσεις έχουν τα παρακάτω πεδία:

- postID: Ένα μοναδικό αναγνωριστικό για κάθε Δημοσίευση (Ακέραιος)
- content: Το περιεχόμενο της Δημοσίευσης (Αλφαριθμητικό – μεγέθους 250)
- dateTime: Η ώρα και η μέρα που ανέβηκε η εκάστοτε Δημοσίευση (Datetime)
- userID: Το μοναδικό αναγνωριστικό του Χρήστη που ανέβασε την Δημοσίευση – **foreign key** (Ακέραιος)

3.2.5. Comment

Κάθε Χρήστης μπορεί να κάνει πολλά Σχόλια. Κάθε Δημοσίευση μπορεί να περιέχει πολλά Σχόλια. Συνεπώς τα Σχόλια του Πληροφοριακού Συστήματος θα περιέχουν τις παρακάτω πληροφορίες:

- commentID: Ένα μοναδικό αναγνωριστικό για κάθε Σχόλιο (Ακέραιος)
- postID: Το μοναδικό αναγνωριστικό της Δημοσίευσης που υπάρχει αυτό το Σχόλιο – **foreign key** (Ακέραιος)
- text: Το περιεχόμενο του Σχόλιου (Αλφαριθμητικό – μεγέθους 100)
- dateTime: Η ώρα και η μέρα που ανέβηκε το Σχόλιο (Datetime)
- userID: Το μοναδικό αναγνωριστικό του Χρήστη που ανέβασε την Δημοσίευση – **foreign key** (Ακέραιος)

3.2.6. Upvote

Κάθε Χρήστης μπορεί να κάνει πολλά likes. Κάθε Δημοσίευση μπορεί να πάρει likes από πολλούς Χρήστες. Για αυτό η πληροφορία των Upvotes θα περιέχει ως κλειδί τα παρακάτω εξωτερικά κλειδιά μόνο:

- postID: Το μοναδικό αναγνωριστικό της Δημοσίευσης που υπάρχει αυτό Upvote – **foreign key** (Ακέραιος)
- userID: Το μοναδικό αναγνωριστικό του Χρήστη που έκανε το Upvote – **foreign key** (Ακέραιος)

3.3. Εγχειρίδιο Χρήστη

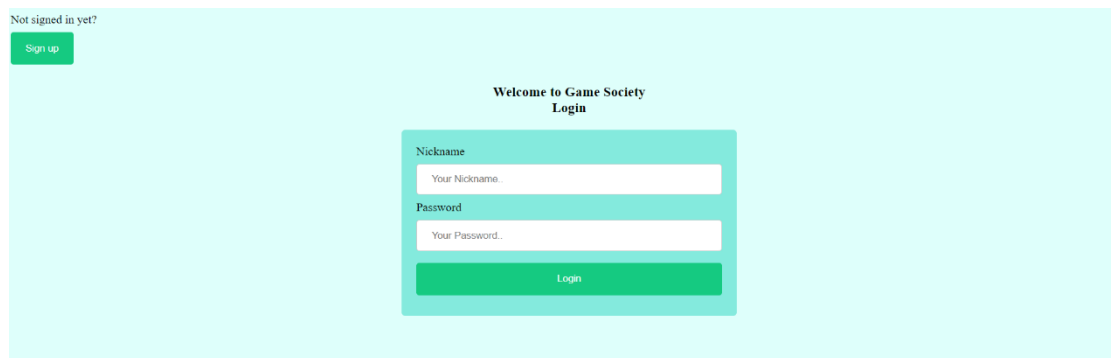
Το παρακάτω Εγχειρίδιο Χρήστη περιέχει εικόνες από τον browser Google Chrome.

3.3.1. Εγκατάσταση, Εκτέλεση και Άνοιγμα

Η εγκατάσταση του Game Society γίνεται ακολουθώντας τα παρακάτω βήματα:

1. Εγκατάσταση της δοσμένης Βάσης Δεδομένων (Δεν είναι αναγκαστικό βήμα στην περίπτωση που θέλουμε να ξεκινήσει μόνο με τον default Διαχειριστή το Σύστημα)
2. Εισαγωγή του φακέλου του Game Society στο Eclipse
3. Άνοιγμα του παρακάτω path:
Game Society/WebContent/Pages/welcome/login.jsp
4. Εκτέλεση του login.jsp
5. Στην συνέχεια προτείνεται να ανοιχτεί κάποιο browser τύπου Google Chrome ή Mozilla Firefox ώστε να φαίνονται τα style των σελίδων του συστήματος.
6. Πάμε στο παρακάτω URL:
<http://localhost:8080/GameSociety/Pages/welcome/login.jsp>

Πλέον αφού βλέπουμε την παρακάτω Εικόνα 3 στον browser, τότε έχουμε ανοίξει το Σύστημα κανονικά.



Εικόνα 3: Σελίδα Log in

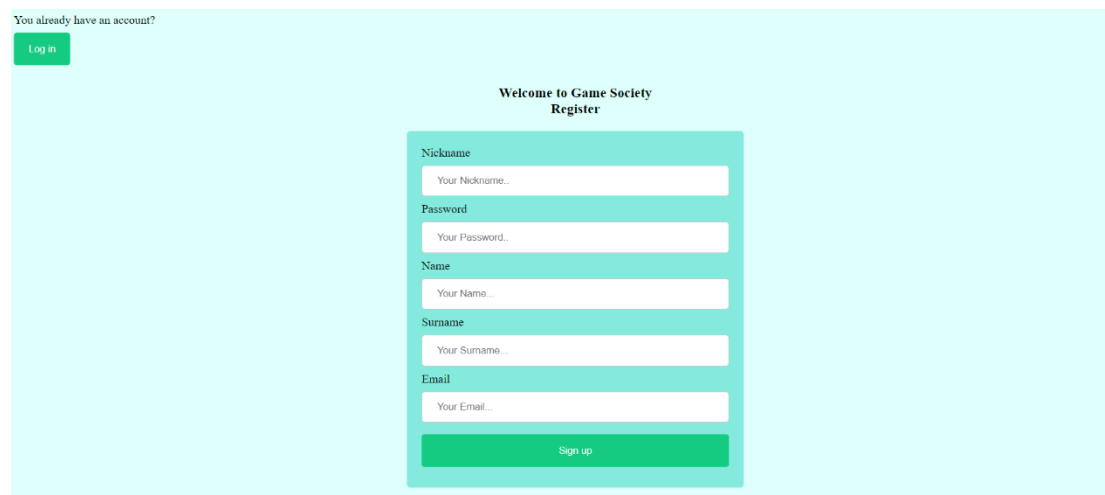
3.3.2. Η Μεριά του Χρήστη

Οι Χρήστες του Game Society μπορούν να εκτελέσουν τις παρακάτω λειτουργίες:

- Πραγματοποίηση της εγγραφής τους στο Σύστημα.
- Εισαγωγή στο σύστημα με τα στοιχεία τους.
- Έξοδος από το Σύστημα
- Μπορούν να δουν όλες τις Δημοσιεύσεις όλων των Χρηστών στο Σύστημα.
- Μπορούν να αναζητήσουν Χρήστες βάση:
 - ο του ψευδωνύμου τους.
 - ο των παιχνιδιών που παίζουν.
- Μπορούν να δουν όλα τα γνωστά δωρεάν παιχνίδια που υπάρχουν στο διαδίκτυο.
- Μπορούν να αναζητήσουν όλα τα γνωστά δωρεάν παιχνίδια που υπάρχουν στο διαδίκτυο βάση της πλατφόρμας των παιχνιδιών.
- Προσθήκη των τίτλων παιχνιδιών που παίζει στο Σύστημα.
- Μπορεί να δει τους τίτλους παιχνιδιών που έχει προσθέσει στο Σύστημα.
- Διαγραφή τίτλων παιχνιδιών που έχει προσθέσει στο Σύστημα.
- Μπορούν να δουν το προφίλ τους.
- Μπορούν να δουν το προφίλ άλλων Χρηστών.
- Αλλαγή στοιχείων τους στο Σύστημα.
- Διαγραφή του Λογαριασμού τους από το Σύστημα.
- Δημιουργία Δημοσίευσης εντός του Συστήματος.
- Διαγραφή κάποιας Δημοσίευσής τους από το Σύστημα.
- Μπορούν να δουν όλες τους τις Δημοσιεύσεις στο προφίλ τους μαζεμένες.
- Μπορούν να δουν όλες τις δημοσιεύσεις κάποιου άλλου Χρήστη μαζεμένες στο προφίλ του.
- Άνοιγμα Δημοσίευσης που θέλουν να δουν παραπάνω στοιχεία για αυτήν
- Δημιουργία Σχολίων σε Δημοσιεύσεις.
- Διαγραφή των δικών τους Σχολίων από Δημοσιεύσεις.
- Μπορούν να δουν όλα τα Σχόλια σε Δημοσιεύσεις.
- Μπορούν να κάνουν Like σε Δημοσιεύσεις.
- Μπορούν να βγάλουν το Like τους από Δημοσιεύσεις που έχουν κάνει ήδη Like.
- Μπορούν να δουν πόσοι έχουν κάνει Like σε μία Δημοσίευση.
- Μπορούν να δουν ποιοι έχουν κάνει Like σε μία Δημοσίευση.
- Μπορούν να στείλουν μήνυμα σε κάποιον άλλων Χρήστη.
- Μπορούν να δουν τα μηνύματά τους με άλλους Χρήστες.
- Μπορούν να δουν ποιους τίτλους παιχνιδιών παίζει ένας άλλος Χρήστης, τους οποίους έχει προσθέσει εκείνος εντός του Συστήματος.

Είσοδος στο Game Society

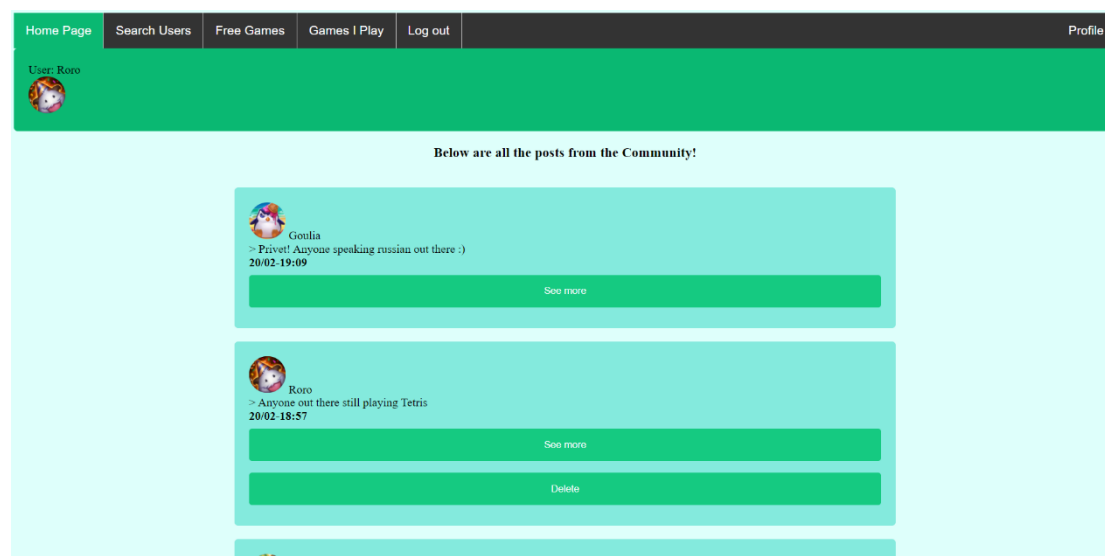
Από την σελίδα του Log in μπορούμε, αν δεν έχουμε κάνει εγγραφή, να πατήσουμε το κουμπί «Sign up» επάνω αριστερά και να μεταφερθούμε στην σελίδα για την εγγραφή μας η οποία φαίνεται στην Εικόνα 4. Αφού εισάγουμε επιτυχώς τα στοιχεία μας, θα μεταφερθούμε στην σελίδα Log in ξανά για να μπούμε στο Σύστημα.



Εικόνα 4: Σελίδα Εγγραφής

Homepage

Αφού εισάγουμε τα στοιχεία μας σωστά στην σελίδα Log in (Εικόνα 3), τότε θα μας εμφανίσει το Homepage το οποίο φαίνεται και στην Εικόνα 5. Σε αυτήν την σελίδα μπορούμε να δούμε όλες τις Δημοσιεύσεις που έχουν κάνει οι Χρήστες του Game Society. Οι δημοσιεύσεις είναι ταξινομημένες ώστε πάνω-πάνω να είναι η πιο πρόσφατη από όλες. Στο επάνω μέρος κάθε σελίδας, όσο είμαστε συνδεδεμένοι με κάποιον λογαριασμό, φαίνονται οι υπόλοιπες σελίδες στις οποίες έχουμε πρόσβαση ως Χρήστες (Home Page, Search Users, Free Games, Games I Play, Log out, Profile).



Εικόνα 5: Homepage

Επάνω αριστερά, βλέπουμε κάποια βασικά στοιχεία του Λογαριασμού από τον οποίο έχουμε συνδεθεί (Ψευδώνυμο και εικόνα profile).

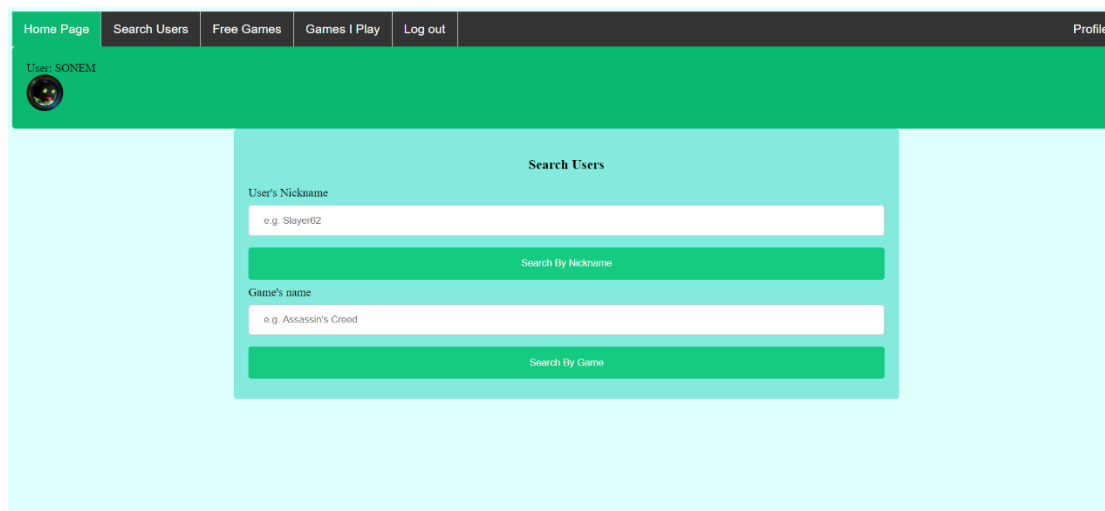
Search Users

Πατώντας στο tab Search Users μας εμφανίζεται μία φόρμα (Εικόνα 6) η οποία μας δίνει τις παρακάτω δύο επιλογές:

- Αναζήτηση όλων των Χρηστών που μοιάζουν με ένα ψευδώνυμο που θα εισάγουμε.
- Αναζήτηση όλων των Χρηστών που παίζουν ένα τίτλο παιχνιδιού που θα εισάγουμε.

Αφού δώσουμε την εισαγωγή μας και πατήσουμε το αντίστοιχο κουμπί, μας εμφανίζονται τα αποτελέσματα κάτω από την φόρμα.

Αναζητώντας για παράδειγμα το αγγλικό γράμμα «n» στο «Search by Nickname» μας εμφανίζονται τα αποτελέσματα της Εικόνας 7. Οι Χρήστες που έχουν εμφανιστεί, περιέχουν μέσα τους το γράμμα «n».



Εικόνα 6: Search Users

Σε κάθε Χρήστη που εμφανίζεται στις αναζητήσεις, πατώντας στο κουμπί «Go to Profile» μπορούμε να πάμε στο προφίλ του Χρήστη αυτού και να εκτελέσουμε λειτουργίες που εξηγούνται στο κεφάλαιο του «**Profile**».

Η αναζήτηση με τίτλο παιχνιδιού βασίζεται στους τίτλους που έχει προσθέσει ο κάθε Χρήστης στην λίστα του, η οποία λίστα εξηγείτε στο κεφάλαιο του «**Games I Play**».



Εικόνα 7: Αποτέλεσμα Αναζήτησης Nickname με το αγγλικό γράμμα «n»

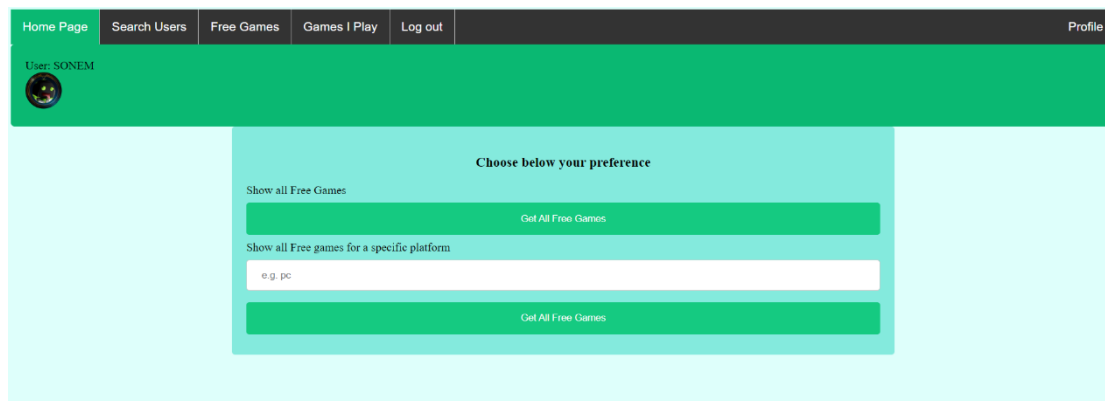
Free Games

Πατώντας στο tab «Free Games» κατευθυνόμαστε σε μία φόρμα (Εικόνα 8) στην οποία έχουμε τις παρακάτω δύο επιλογές:

- Να δούμε όλα τα διαδεδομένα δωρεάν παιχνίδια που υπάρχουν στο διαδίκτυο
- Να βρούμε για μία πλατφόρμα που θα εισάγουμε εμείς, ποια διαδεδομένα δωρεάν παιχνίδια υπάρχουν σε αυτήν.

Παρομοίως με το «Search Users» αφού πατήσουμε το αντίστοιχο κουμπί, θα μας εμφανιστούν κάτω από την φόρμα τα αποτελέσματα μας. Για κάθε παιχνίδι θα εμφανιστεί:

- Ένα thumbnail του.
- Ένα link για την σελίδα του παιχνιδιού.
- Ο τίτλος του παιχνιδιού.

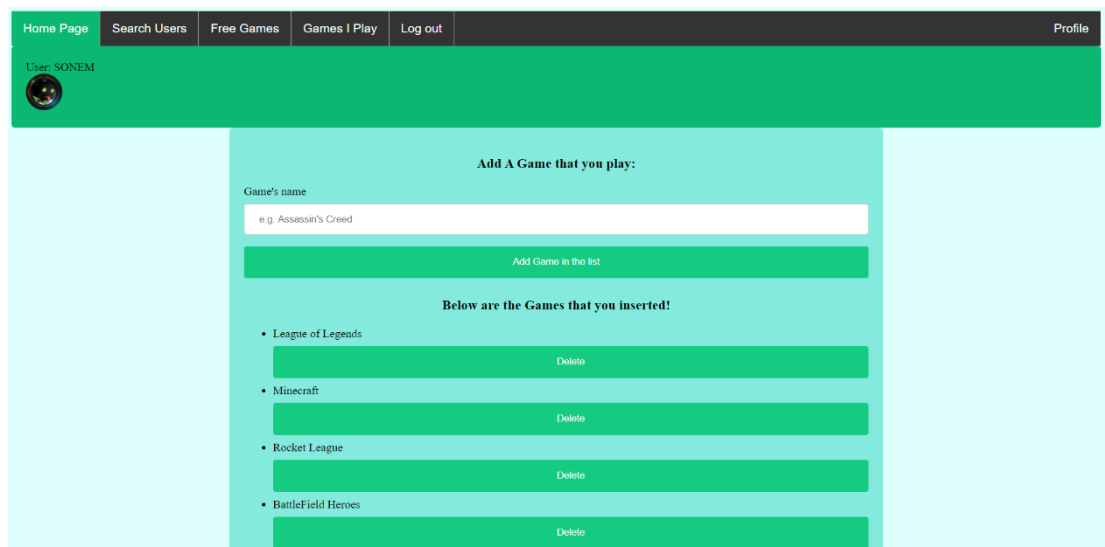


Εικόνα 8: Free Games

Games I Play

Στο tab «Games I Play» εμφανίζεται η Λίστα με τα παιχνίδια που έχουμε προσθέσει, τα οποία ο κάθε Χρήστης δηλώνει ότι παίζει (Εικόνα 9). Συγκεκριμένα ο Χρήστης SONEM στην εικόνα παίζει παιχνίδια όπως «League of Legends», «Minecraft» κλπ.

Ο Χρήστης έχει σε κάθε του εισαγωγή την επιλογή, πατώντας το κουμπί «DELETE», να την διαγράψει από την Λίστα του. Ακόμα στην φόρμα που του εμφανίζεται μπορεί να εισάγει παιχνίδια στην Λίστα του. Αυτά τα παιχνίδια είναι αυτά τα οποία παίζει και θέλει να μπορεί άλλοι Χρήστες να τον αναζητούν μέσω αυτών στο tab «**Search Users**».



Εικόνα 9: Game I Play

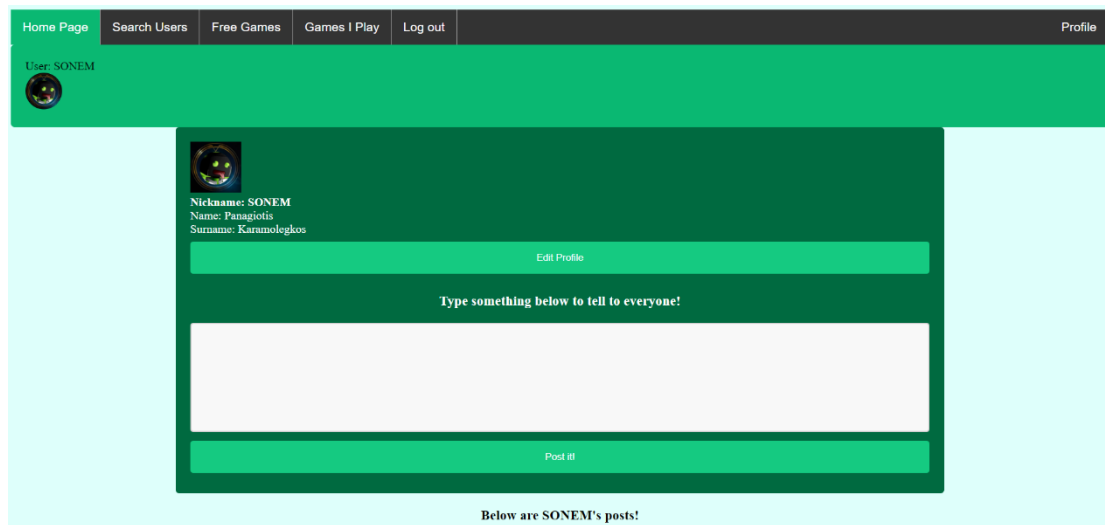
Log out

Πατώντας το tab «Log out» ο Χρήστης βγαίνει από την σύνδεση του στο σύστημα και του εμφανίζεται η φόρμα για να κάνει Log in.

Profile

Όταν ο Χρήστης πατήσει στο tab «Profile» (επάνω δεξιά), μπορεί να δει το προφίλ του, μαζί με τις Δημοσιεύσεις τις οποίες έχει κάνει (κάνοντας scroll down). Η συγκεκριμένη σελίδα εμφανίζεται και για τους άλλους Χρήστες όταν πατάμε επάνω στο «Go to Profile» στην σελίδα «**Search Users**».

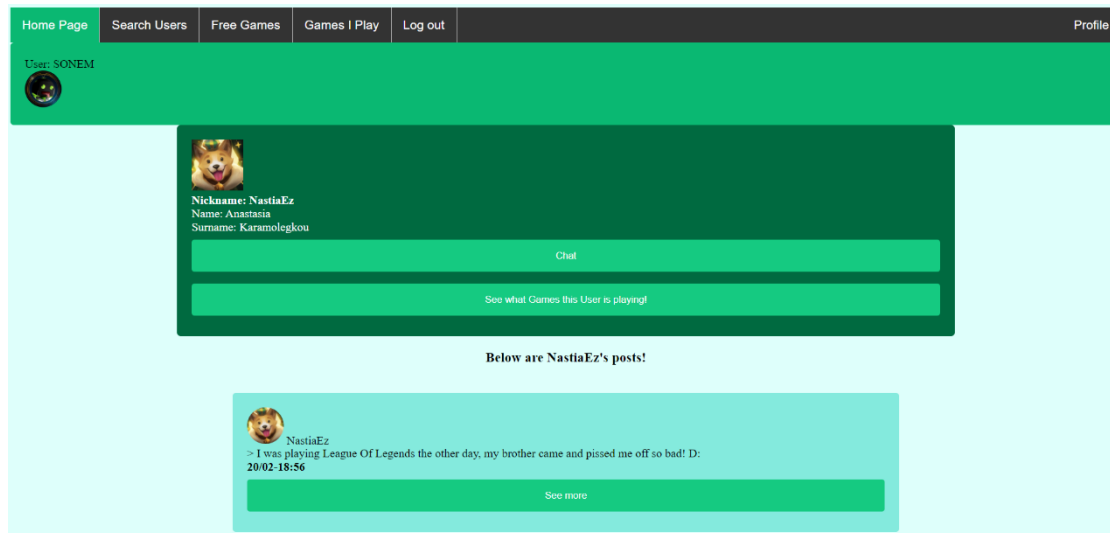
Στην περίπτωση που έχουμε μπει στο δικό μας Profile τότε βλέπουμε μία σελίδα, όπως στην Εικόνα 10. Σε αυτήν μπορούμε να δημιουργήσουμε μία δική μας Δημοσίευση και να την κάνουμε Post ώστε να την βλέπουν και οι υπόλοιποι Χρήστες του Game Society. Ακόμα μπορούμε να πατήσουμε επάνω στο «Edit Profile» για να αλλάξουμε πληροφορίες σχετικά με τον Λογαριασμό μας το οποίο θα εξηγηθεί περαιτέρω στο κεφάλαιο «**Edit Profile**».



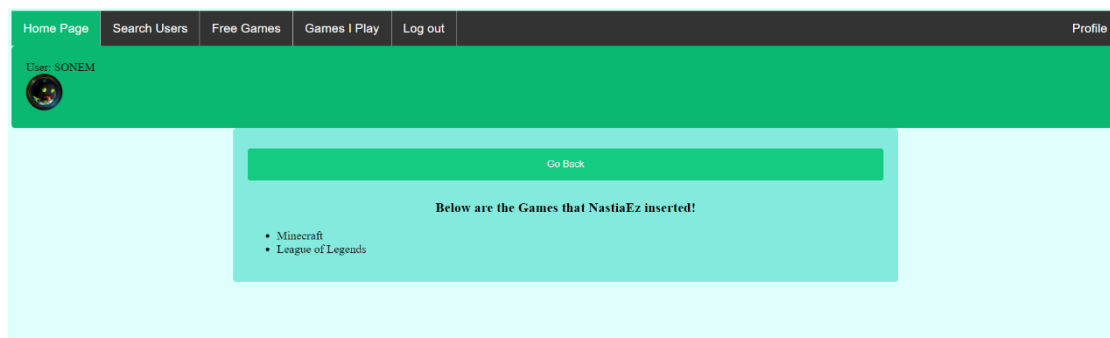
Εικόνα 10: Το δικό μας Profile

Εάν έχουμε μπει στο προφίλ κάποιου άλλου Χρήστη, τότε βλέπουμε μία σελίδα, όπως της Εικόνας 11. Σε αυτήν την σελίδα μπορούμε να επιλέξουμε αν θα κάνουμε chat με αυτόν τον Χρήστη ή αν θέλουμε να δούμε όλα τα παιχνίδια τα οποία έχει δηλώσει ότι παίζει. Το chat θα αναλυθεί στο κεφάλαιο «**Chat**».

Για να δούμε ποια παιχνίδια παίζει ο άλλος Χρήστης, αρκεί να πατήσουμε στο κουμπί «See what Games this User is playing!». Έτσι θα οδηγηθούμε σε μία σελίδα, όπως της Εικόνας 12. Μέσο αυτής της σελίδας, μπορούμε να δούμε τα παιχνίδια που βλέπει αυτός ο Χρήστης, αλλά και να πάμε πίσω στο Profile του, πατώντας το κουμπί «Go Back».



Εικόνα 11: Το Profile ενός άλλου Χρήστη



Εικόνα 12: Τα παιχνίδια που παίζει ένας άλλος Χρήστης

Edit Profile

Αφού πατήσουμε το κουμπί «Edit Profile» από το Profile μας, κατευθυνόμαστε σε μία σελίδα όπως της Εικόνας 13. Σε αυτή βλέπουμε στοιχεία τα οποία έχουμε εισάγει στο Σύστημα κατά την εγγραφή μας. Εδώ μπορούμε να αλλάξουμε το password, το όνομα και το επώνυμο που έχουμε δηλώσει. Ακόμα πατώντας σε μία από τις εικόνες που φαίνονται, μπορούμε να αλλάξουμε την εικόνα του προφίλ μας. Εάν κάνουμε scroll down μέχρι το κάτω μέρος της σελίδας, μπορούμε να δούμε το κουμπί «Delete My Account» (Εικόνα 14) ώστε να διαγράψουμε όλα τα σχόλια, τις συζητήσεις μας, τις δημοσιεύσεις μας, τα παιχνίδια που έχουμε προσθέσει, αλλά και το προφίλ μας από το Game Society.

Home Page	Search Users	Free Games	Games I Play	Log out	Profile
-----------	--------------	------------	--------------	---------	---------

Email: p.karamolegos@yahoo.gr
NickName: SONEM

To change your Name or Surname fill the field below

Name
Your Name...

Surname
Your Surname...

Update my Account

To change your Password fill the field below

Password
Your Password...

Change my password

Click an image below to update your profile picture

Εικόνα 13: Edit Profile

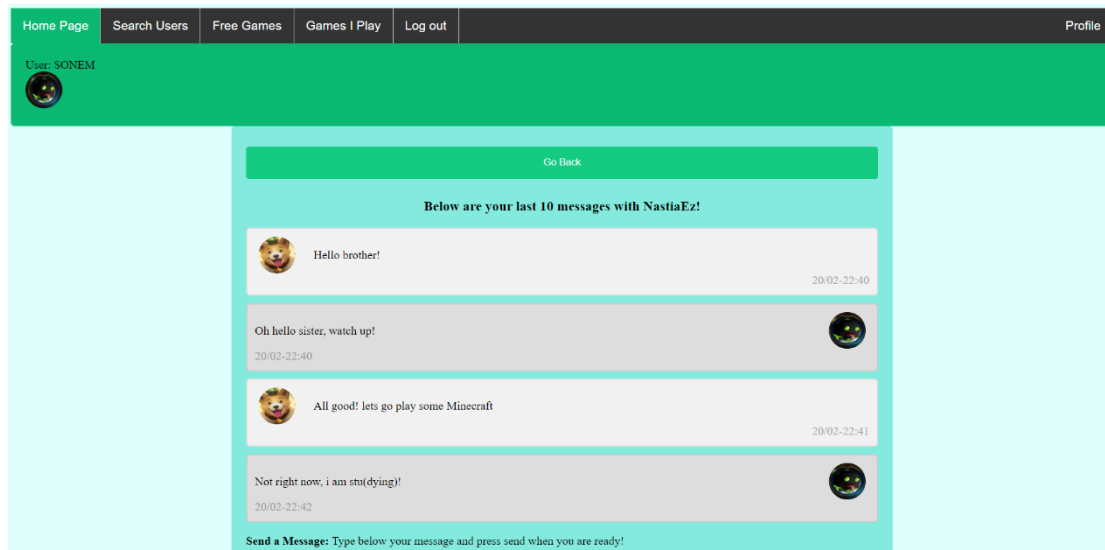
If you want to delete your account, press the button below

Delete My Account

Εικόνα 14: Το «Delete My Account» στο κάτω μέρος του «Edit Profile»

Chat

Αφού πατήσουμε στο κουμπί «Chat» από το Profile κάποιου άλλου Χρήστη, μεταφερόμαστε σε μία σελίδα, όπως της Εικόνας 15. Σε αυτήν την σελίδα μπορούμε να δούμε την συνομιλία μας με αυτόν τον Χρήστη και μπορούμε να πάμε πίσω στο Profile του πατώντας το κουμπί «Go Back». Αυτή η σελίδα θα δείξει τα τελευταία 10 μας μηνύματα με τον άλλον Χρήστη. Κάνοντας Scroll down (εκτός αν δεν έχουμε στείλει μηνύματα ακόμα με αυτόν τον Χρήστη) μπορούμε να δημιουργήσουμε ένα μήνυμα και να το στείλουμε στον άλλων Χρήστη (Εικόνα 16). Τα μηνύματα των Χρηστών είναι ταξινομημένα ώστε το πιο πρόσφατο μήνυμα να εμφανίζεται κάτω-κάτω στην συνομιλία, ώστε να μπορούν οι Χρήστες να ανατρέξουν αν θέλουν στα τελευταία μηνύματα και να τα διαβάσουν από πάνω προς τα κάτω.



Εικόνα 15: Chat

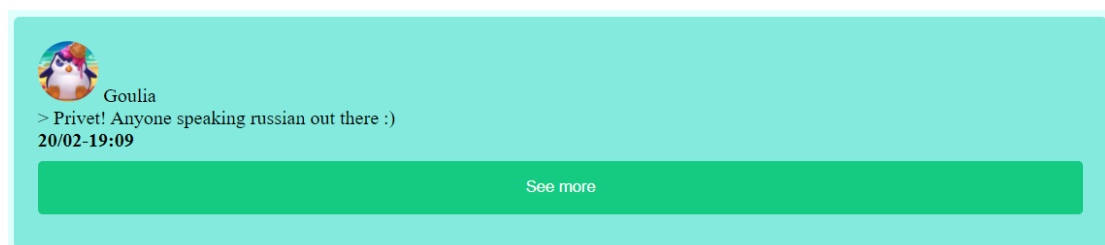
Send a Message: Type below your message and press send when you are ready!

Send it!

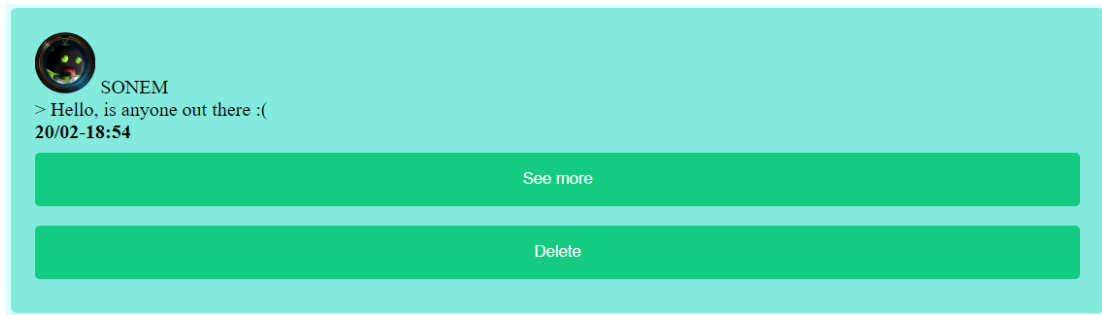
Εικόνα 16: Η φόρμα για την δημιουργία και αποστολή μηνύματος

Posts

Στο «**Homepage**» και στο κάθε «**Profile**» των Χρηστών φαίνονται οι Δημοσιεύσεις τους. Όπου εμφανίζονται Δημοσιεύσεις υπάρχει το κουμπί «See more» μέσα στην δημοσίευση (Εικόνα 17). Πατώντας αυτό το κουμπί, μεταφερόμαστε στην σελίδα αυτής της Δημοσίευσης ώστε να δούμε παραπάνω στοιχεία σχετικά με αυτήν. Όμως, όποτε συναντάμε δική μας δημοσίευση, εμφανίζεται μαζί και το κουμπί «Delete» (Εικόνα 18) ώστε πατώντας το να διαγράψουμε όλο το περιεχόμενο αυτής της Δημοσίευσης.

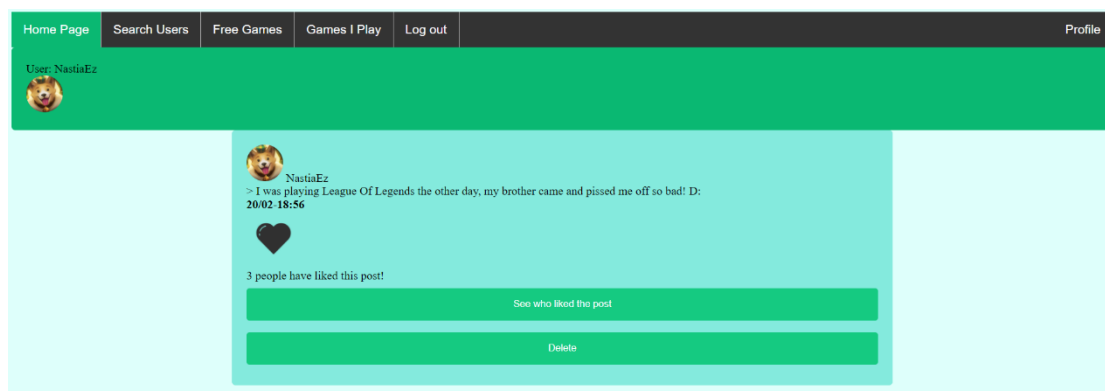


Εικόνα 17: Μία Δημοσίευση



Εικόνα 18: Μία Δημοσίευση του Χρήστη που είναι συνδεδεμένος στο Game Society

Αφού έχουμε πατήσει στο «See more» θα εμφανιστεί μία σελίδα, όπως αυτή που φαίνεται στην Εικόνα 19. Στην περίπτωση που η Δημοσίευση δεν είναι δικιά μας, τότε δεν θα υπάρχει το κουμπί «Delete», καθώς θα ήταν παράλογο να μπορούμε να διαγράψουμε την Δημοσίευση κάποιου άλλου Χρήστη.



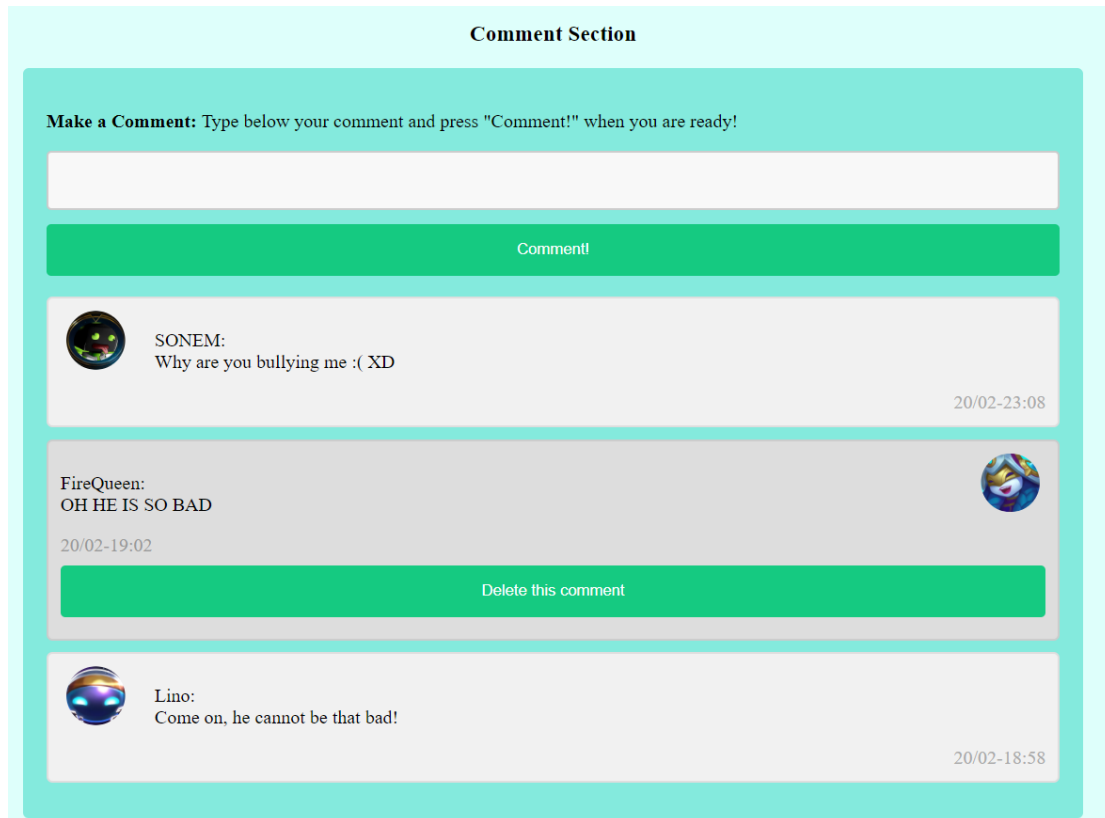
Εικόνα 19: Η σελίδα μίας Δημοσίευσης

Σε αυτήν την σελίδα μπορούμε να κάνουμε τα παρακάτω:

- Να κάνουμε Like ή να «πάρουμε πίσω» το Like μας από την Δημοσίευση – πατώντας επάνω στην εικόνα της καρδιάς.
 - Μαύρη καρδιά σημαίνει ότι δεν έχουμε κάνει Like.
 - Κόκκινη καρδιά σημαίνει ότι έχουμε κάνει Like.
- Να δούμε πόσοι άνθρωποι έχουν κάνει Like στην δημοσίευση (στην Εικόνα 19 είναι τρεις)
- Να δούμε ποιοι έχουν κάνει Like στην δημοσίευση – πατώντας στο κουμπί «See who liked the post» (εξηγείτε περεταίρω στο κεφάλαιο «**See who liked the post**»)
- Να δούμε όλα τα Σχόλια της Δημοσίευσης (εξηγείτε περεταίρω στο κεφάλαιο «**Comments**»)

Comments

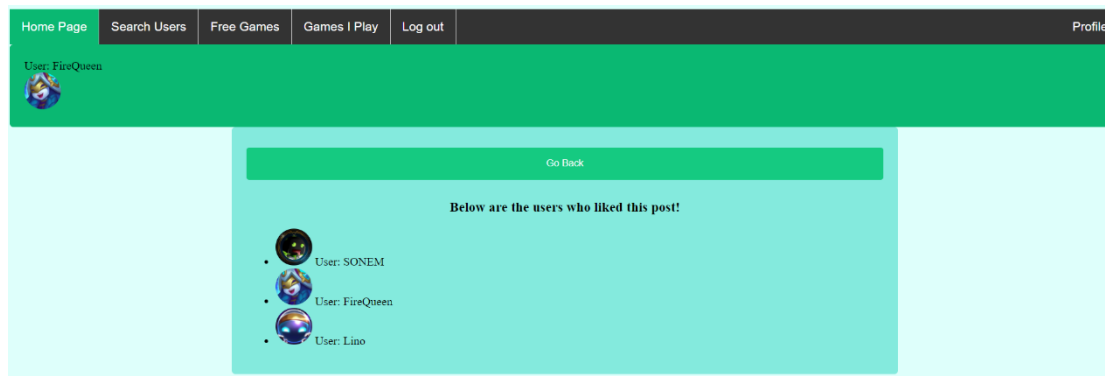
Κάνοντας scroll down στην σελίδα μίας Δημοσίευσης εμφανίζονται τα Σχόλια (Εικόνα 20). Εκεί μπορούμε να δούμε τα σχόλια της Δημοσίευσης, να διαγράψουμε τα δικά μας σχόλια, όπως και προσθέσουμε νέα σχόλια. Στην Εικόνα 20 το σχόλιο που έχει την εικόνα προφίλ από την δεξιά μεριά, ανήκει στον λογαριασμό από τον οποίο έχουμε συνδεθεί στο Σύστημα και συνεπώς μας επιτρέπεται να το διαγράψουμε πατώντας το κουμπί «Delete this comment».



Εικόνα 20: Comment Section

See who liked the post

Πατώντας το κουμπί «See who liked the post» από την σελίδα μίας Δημοσίευσης κατευθυνόμαστε σε μία σελίδα όπως της Εικόνας 21. Σε αυτήν, μπορούμε να δούμε όλα τα άτομα που έχουν κάνει Like στην Δημοσίευση και πατώντας «Go Back» μπορούμε να πάμε πίσω στην σελίδα της Δημοσίευσης (Εικόνα 19).



Εικόνα 21: See who liked the post

3.3.3. Η Μεριά του Διαχειριστή

Οι Διαχειριστές του Συστήματος, μπορούν να κάνουν οτιδήποτε κάνουν και οι απλοί Χρήστες, αλλά έχουν κάποιες επιπλέον ικανότητες όπως:

- Διαγραφή άλλων Χρηστών από το Game Society.
- Διαγραφή Δημοσιεύσεων άλλων Χρηστών από το Σύστημα.
- Διαγραφή Σχολίων άλλων Χρηστών από τις δημοσιεύσεις.
- Αλλαγή των δικαιωμάτων των απλών Χρηστών από απλούς Χρήστες σε Διαχειριστές και το ανάποδο.
- Μπορούν να βλέπουν μαζεμένους όλους τους Χρήστες του Συστήματος.

Στην περίπτωση που δεν προϋπάρχει η Βάση Δεδομένων, τότε χτίζεται αυτόματα Διαχειριστής με τα παρακάτω στοιχεία ως Default:

- Όνομα: admin
- Επώνυμο: admin
- Ψευδώνυμο: admin
- Κωδικός: admin
- Email: admin@admin.com

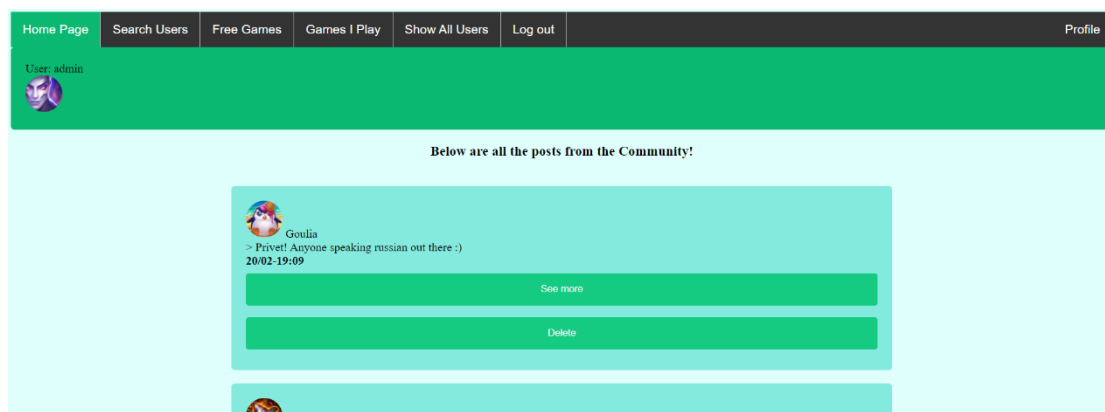
Ο παραπάνω Διαχειριστής, θα ξαναχτίζεται όποτε δεν υπάρχει άλλος Διαχειριστής εντός του Συστήματος αυτόματα.

Οι Διαχειριστές σε αντίθεση με τους Χρήστες, μπορούν να έχουν τα παρακάτω κουμπιά πάντα, ακόμα και σε αντικείμενα άλλων Χρηστών:

- Διαγραφή της Δημοσίευσης «Delete»
- Διαγραφή του Σχολίου «Delete this Comment»

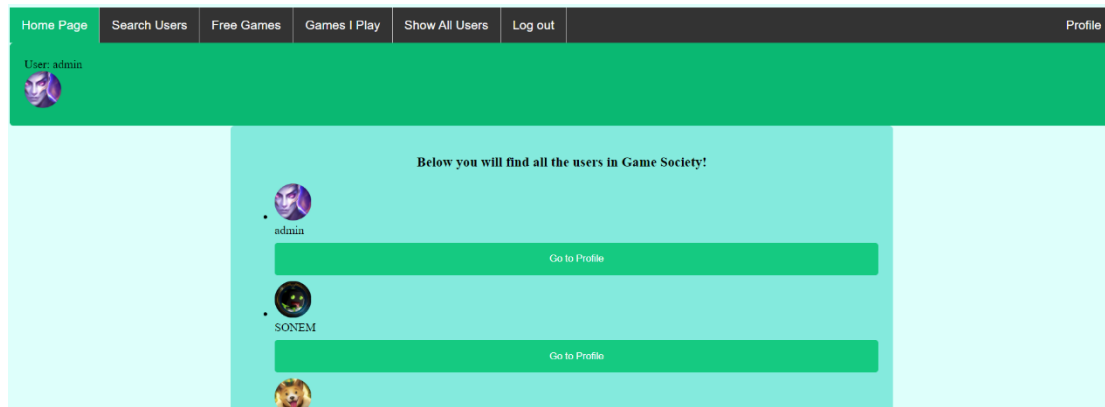
Show All Users

Αφού συνδεθούμε με λογαριασμό ενός διαχειριστή, προστίθεται ένα επιπλέον tab στο μενού του Game Society ώστε να μπορούμε να δούμε όλους τους Χρήστες μαζεμένους. Αυτό το tab είναι το «Show All Users» και είναι το προτελευταίο από τα αριστερά (φαίνεται στην Εικόνα 22).



Εικόνα 22: Show All Users tab

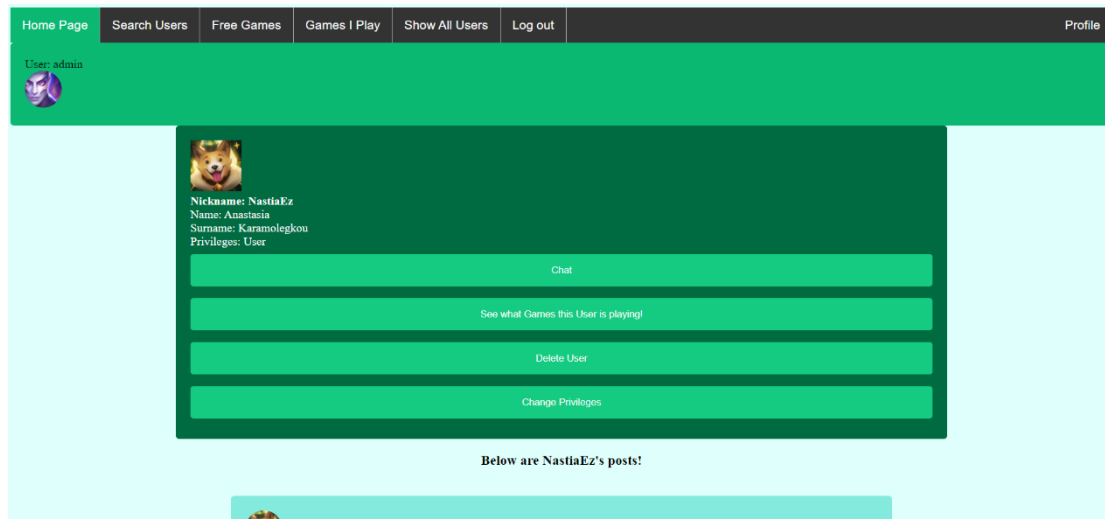
Αφού πατήσουμε σε αυτό το tab, μας εμφανίζεται μία σελίδα όπως της Εικόνας 23. Από εκεί μπορούμε να πάμε γρήγορα στο προφίλ όποιου Χρήστη θέλουμε.



Εικόνα 23: Show All Users

Profile Viewing – By Admins

Όταν ένας Διαχειριστής κοιτάει το προφίλ ενός άλλου Χρήστη, βλέπει μία σελίδα όπως της Εικόνας 24. Συνεπώς, έχουμε δύο επιπλέον κουμπιά.



Εικόνα 24: Profile Viewing – By Admins

Εάν πατήσουμε το κουμπί «Delete User» τότε ο Χρήστης αυτός θα διαγραφεί από το Game Society, διαγράφοντας όλες του τις συζητήσεις, τα σχόλια του, τα παιχνίδια που έχει στην Λίστα του, τις Δημοσιεύσεις του και το προφίλ του.

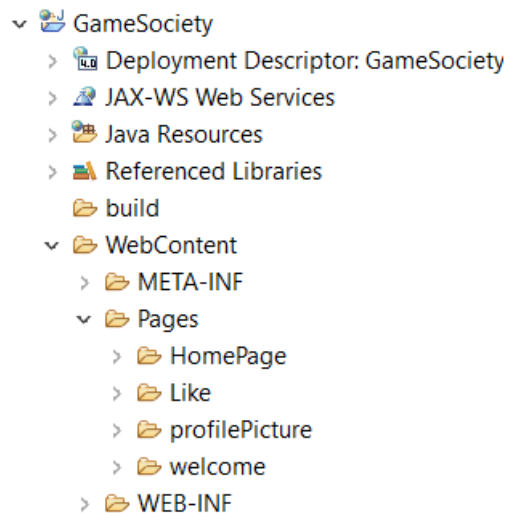
Πατώντας το κουμπί «Change Privileges» ο Χρήστης αυτός:

- Θα γίνει Διαχειριστής αν είναι απλός Χρήστης
- Θα γίνει απλός Χρήστης αν είναι Διαχειριστής

3.4. Τεχνικό Εγχειρίδιο

Στην συνέχεια θα εξηγηθεί το Σύστημα στην τεχνική μεριά του. Θα αναλυθούν όλες οι σελίδες εντός του φακέλου - Project «GameSociety» οι οποίες βρίσκονται στο παρακάτω μονοπάτι (το οποίο φαίνεται ανοιχτό και στην Εικόνα 25) ως προς τα Web Services που χρησιμοποιούν:

GameSociety/WebContent/Pages



Εικόνα 25: Το μονοπάτι για τις σελίδες του Συστήματος

Όπως αναφέρεται και στο **Εγχειρίδιο Χρήστη**, για ανοίξουμε το Σύστημα σωστά πρέπει να εκτελέσουμε το login.jsp από το παρακάτω μονοπάτι:

GameSociety/WebContent/Pages/welcome/login.jsp

Συνεπώς οι εξήγηση του Συστήματος θα ξεκινήσει από εκείνη την σελίδα.

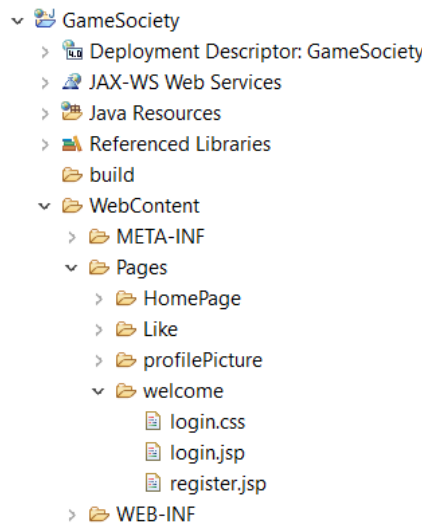
Web Services τα οποία χρησιμοποιούνται σε πολλές σελίδες για τον ίδιο λόγο (όπως η εμφάνιση των στοιχείων του συνδεδεμένου Χρήστη επάνω αριστερά) δεν θα εξηγούνται σε όλες τις σελίδες, παρά μόνο σε αυτές που πρωτοεμφανίζονται στο παρακάτω εγχειρίδιο.

Όλα τα Web Services τα οποία χρησιμοποιούνται παρακάτω έχουν εξηγηθεί στο **κεφάλαιο 2** αυτής την αναφοράς.

3.4.1. welcome

Ξεκινώντας από το παρακάτω μονοπάτι, βλέπουμε εντός του τρία αρχεία (Εικόνα 26):

GameSociety/WebContent/Pages/welcome



Εικόνα 26: welcome

Τα τρία αυτά αρχεία είναι τα login.css, login.jsp και register.jsp. Οποιαδήποτε αρχεία που καταλήγουν σε .css είναι απλά το στίλ που έχει επιλεγεί για το σύστημα, οπότε από εδώ και μπρος δεν θα γίνονται αναφορές ως προς αυτά.

login.jsp

Το login.jsp ξεκινάει «ανοίγοντας» την Βάση Δεδομένων, ώστε ο Χρήστης να μπορεί να πραγματοποιήσει αλληλεπίδραση με αυτήν (Εικόνα 27). Ακόμα, στην περίπτωση που η Βάση δεν υπάρχει, τότε αυτή δημιουργείται και μαζί της κατασκευάζεται και ο default admin. Οι περισσότερες σελίδες κάνουν αυτήν την ενέργεια ώστε να εξασφαλισθεί ότι η Βάση θα μπορεί να είναι διαθέσιμη για τον Χρήστη από της σελίδες του συστήματος, οπότε τις χρησιμοποιεί.

```
7 <%
8     Client client = Client.create();
9     WebResource webResource = client.resource("http://localhost:8080/GameSociety/rest/GameSociety/testForCellarDB");
10    ClientResponse myresponse = webResource.get(ClientResponse.class);
11 %>
```

Εικόνα 27: Άνοιγμα της Βάσης

Στην Εικόνα 28 φαίνονται οι Χρήστες, όταν η Βάση χτίζεται αυτόματα από το Web Service για πρώτη φορά. Ο Διαχειριστής που φαίνεται θα ξαναδημιουργείται, οπότε δεν θα υπάρχει άλλος Διαχειριστής εντός της Βάσης.

	userID	name	surname	profilePicturePath	nickName	isAdmin	password	email
▶	1	admin	admin	1	admin	1	admin	admin@admin.com
*	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

Εικόνα 28: Χτίσιμο της Βάσης

Συνεχίζοντας, οι σελίδες του Game Society είναι έτσι κατασκευασμένες, ώστε να μετακινούν τον τελικό χρήστη στην σωστή σελίδα. Για παράδειγμα ένας Χρήστης ο οποίος είναι ήδη συνδεδεμένος, δεν πρέπει να μπορεί να πάει στο login εκτός αν αποσυνδεθεί. Ένα τέτοιο είδος διαχείρισης φαίνεται στην Εικόνα 29.

```
12<=<%
13
14     if(session.getAttribute("userID")!= null){
15         response.sendRedirect("../HomePage/HomePage.jsp");
16     }
17
18 %>
```

Εικόνα 29: Έλεγχος ήδη συνδεδεμένου Χρήστη

Πριν από την εισαγωγή ενός Χρήστη στο σύστημα, γίνονται έλεγχοι σχετικά με την ορθότητα των δεδομένων που παραχωρεί στην φόρμα που έχει μπροστά του. Χρησιμοποιώντας το Web Service **testForUser**, η σελίδα παίρνει τα δεδομένα του Χρήστη από την βάση δεδομένων με βάση το ψευδώνυμο που δίνει ώστε να τακτοποιήσει τον κωδικό του (Εικόνα 30).

```
28     if(request.getParameter("login")!=null){
29         String nickName = request.getParameter("nickName");
30         String password = request.getParameter("password");
31
32         if(nickName.equals("") || password.equals("")){
33             error = "You must fill all the fields!";
34         }
35         else if(nickName.contains(" ") || password.contains(" ")){
36             error = "You must not use spaces!";
37         }
38         else{
39             client = Client.create();
40             String link = "http://localhost:8080/GameSociety/rest/GameSociety/testForUser/";
41             link+=nickName;
42             webResource = client.resource(link);
43             myresponse = webResource.accept("application/json").get(ClientResponse.class);
44             JSONObject user = new JSONObject(myresponse.getEntity(String.class));
45             if(user.length()>0){
46                 if(password.equals(user.getString("password"))){
47                     session.setAttribute("userID",user.getInt("userID"));
48                     if(user.getInt("isAdmin")==1){
49                         session.setAttribute("adminID",user.getInt("userID"));
50                     }
51                     response.sendRedirect("../HomePage/HomePage.jsp");
52                 }
53                 else{
54                     error = "The nickname or password is wrong!";
55                 }
56             }
57             else{
58                 error = "The nickname or password is wrong!";
59             }
60         }
61     }
```

Εικόνα 30: Έλεγχος ταυτοποίησης

register.jsp

Η εγγραφή του κάθε Χρήστη στο Game Society γίνεται από τον φάκελο welcome, στο αρχείο register.jsp:

Σε αυτήν την σελίδα, μετά από ελέγχους ως προς το αν έχει συμπληρώσει ο Χρήστης την φόρμα σε όλα τα πεδία της με σωστό τρόπο, γίνεται έλεγχος της ήδη ύπαρξης του Χρήστη στο Πληροφοριακό Σύστημα μέσω των **testForUser** (για την πιθανή ύπαρξη του ψευδωνύμου του) και **testForEmail** (για την πιθανή ύπαρξη αυτού του email). Αφού ανιχνευτεί ότι ο Χρήστης αυτός δεν υπάρχει και έχει συμπληρώσει σωστά την φόρμα, τότε γίνεται η εισαγωγή του στο σύστημα μέσω του Service **addAUser**. Αυτές οι κλήσεις φαίνονται στην Εικόνα 31.

Στην Εικόνα 32, φαίνεται η εισαγωγή ενός νέου χρήστη στο Πληροφοριακό Σύστημα. Όταν ένας Χρήστης κάνει εγγραφή, τότε του δίνεται εξ αρχής η ιδιότητα του απλού Χρήστη και η εικόνα με όνομα 1.png ως εικόνα προφίλ από τον παρακάτω φάκελο:

GameSociety/WebContent/Pages/profilePicture

```
46     else{
47         client = Client.create();
48         link = "http://localhost:8080/GameSociety/rest/GameSociety/testForUser/";
49         link+=nickName;
50         webResource = client.resource(link);
51         myresponse = webResource.accept("application/json").get(ClientResponse.class);
52         JSONObject user = new JSONObject(myresponse.getEntity(String.class));
53         if(user.length()==0){
54             client = Client.create();
55             link = "http://localhost:8080/GameSociety/rest/GameSociety/testForEmail/";
56             link+=email;
57             webResource = client.resource(link);
58             myresponse = webResource.accept("application/json").get(ClientResponse.class);
59             JSONObject userByEmail = new JSONObject(myresponse.getEntity(String.class));
60             if(userByEmail.length()==0){
61
62                 client = Client.create();
63                 link = "http://localhost:8080/GameSociety/rest/GameSociety/addAUser/";
64                 link+=name+"/";
65                 link+=surname+"/";
66                 link+=nickName+"/";
67                 link+=0+"/";
68                 link+=password+"/";
69                 link+=email;
70                 webResource = client.resource(link);
71                 myresponse = webResource.post(ClientResponse.class);
72
73
74                 session.setAttribute("registered",true);
75                 response.sendRedirect("login.jsp");
76             }
77         }
78         else{
79             error = "This email already exist!";
80         }
81     }
```

Εικόνα 31: Οι κλήσεις του register.jsp

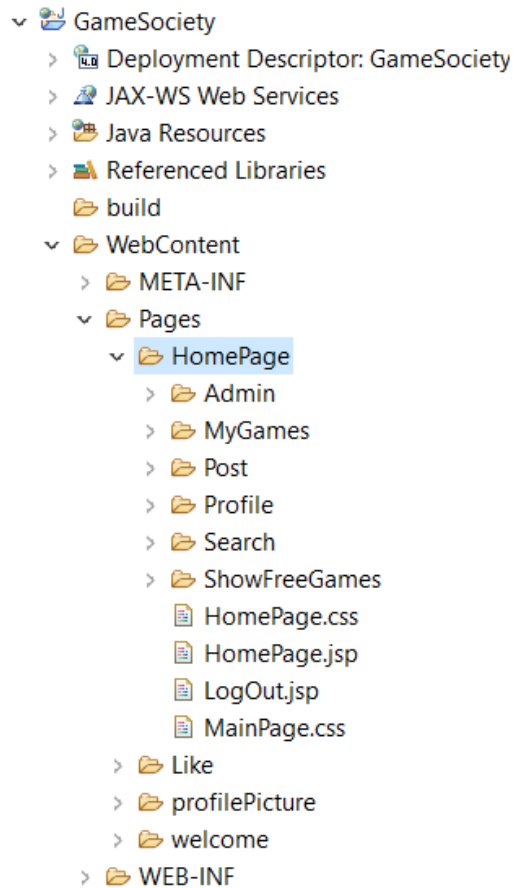
	userID	name	surname	profilePicturePath	nickName	isAdmin	password	email
▶	1	admin	admin	1	admin	1	admin	admin@admin.com
	2	Panagiotis	Karamolegkos	1	SONEM	0	12345	p.karamolegos@yahoo.gr
*	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

Εικόνα 32: Εισαγωγή νέου Χρήστη στην Βάση

3.4.2. HomePage

Αυτό το ύπο-κεφάλαιο αναφέρετε στον φάκελο του παρακάτω μονοπατιού, ο οποίος φαίνεται ανοιχτός και στην Εικόνα 33:

GameSociety/WebContent/Pages/HomePage



Εικόνα 33: HomePage

Στο 3.4.2. θα αναφερθούν τα HomePage.jsp και LogOut.jsp και τα υπόλοιπα περιεχόμενα του φακέλου αυτού θα εξηγηθούν στα κεφάλαια 3.4.3. έως 3.4.8.

LogOut.jsp

Η σελίδα αυτή λειτουργεί ως απλή διαγραφή του χρήστη από το session του browser και μεταφορά του στην σελίδα log in.

HomePage.jsp

Η σελίδα HomePage.jsp χρησιμοποιείται ως η «Αρχική» του Game Society. Για την εμφάνιση του ψευδώνυμου και της εικόνας προφίλ του Χρήστη χρησιμοποιείται η **getUser** από τις γραμμές κώδικα που εμφανίζονται στην Εικόνα 34.

```

22     else{
23         userID = (int)session.getAttribute("userID");
24         client = Client.create();
25         String link = "http://localhost:8080/GameSociety/rest/GameSociety/getUser/";
26         link+=userID;
27         webResource = client.resource(link);
28         myresponse = webResource.accept("application/json").get(ClientResponse.class);
29         JSONObject user = new JSONObject(myresponse.getEntity(String.class));
30         nickName = user.getString("nickName");
31         isAdmin = user.getInt("isAdmin");
32         profilePicturePath = user.getString("profilePicturePath");
33     }
34 %>

```

Εικόνα 34: Απόκτηση στοιχείων του συνδεδεμένου Χρήστη

Όλες οι λειτουργίες που μπορεί να πραγματοποιήσει ο Χρήστης στις σελίδες του Game Society δίνονται σε μορφή φόρμας, συνεπώς ανιχνεύονται με Δομές Επιλογής (if-else if- else) για την πραγματοποίηση της διαχείρισης τους.

Για την καλύτερη διαχείριση και ποιο εύκολη διόρθωση του συστήματος, κάθε φόρμα καλεί το ίδιο .jsp στο οποίο βρίσκεται ώστε το ίδιο να διαχειριστεί το που θα κατευθύνει τον χρήστη στο τέλος της εκτέλεσης της κάθε λειτουργίας του.

Στην Εικόνα 35 φαίνονται τα υπόλοιπα Web Services που καλεί το HomePage.jsp.

```

35<%
36     if(request.getParameter("deletePost")!=null){
37         int postID = Integer.parseInt(request.getParameter("postID"));
38         client = Client.create();
39         String link = "http://localhost:8080/GameSociety/rest/GameSociety/deletePost/";
40         link+=postID;
41         webResource = client.resource(link);
42         myresponse = webResource.delete(ClientResponse.class);
43     }
44
45     JSONArray allPosts = new JSONArray();
46     if(session.getAttribute("userID") != null){
47         client = Client.create();
48         String link = "http://localhost:8080/GameSociety/rest/GameSociety/getHomeScreenPosts/";
49         webResource = client.resource(link);
50         myresponse = webResource.accept("application/json").get(ClientResponse.class);
51         allPosts = new JSONArray(myresponse.getEntity(String.class));
52     }
53
54     if(request.getParameter("goToThePost")!=null){
55         int postID = Integer.parseInt(request.getParameter("postID"));
56         session.setAttribute("postViewingID",postID);
57         response.sendRedirect("Post/Post.jsp");
58     }
59 %>

```

Εικόνα 35: Web Services του HomePage.jsp

Όποτε ανιχνεύεται η παράμετρος deletePost τότε από την αντίστοιχη φόρμα της σελίδας δίνεται το ID της αντίστοιχης Δημοσίευσης και τελικά γίνεται η διαγραφή του, μαζί με όλα τα σχόλια και τα Likes που σχετίζονται με αυτό μέσω του Web Service **deletePost** (Εικόνα 36).

Όποτε ανιχνεύεται εντός του session του browser ότι υπάρχει Χρήστης μέσα τότε εμφανίζονται ζητούνται όλες οι Δημοσιεύσεις όλων των Χρηστών ώστε στην συνέχεια να του εμφανιστούν μέσω του **getHomeScreenPosts**.

post				
postID	content	userID	dateTime	
1	Hello World!!	2	2021-02-21 18:47:27	
2	Playing minecraft!	3	2021-02-21 18:47:51	
3	Anyone who wants to playing tetris	7	2021-02-21 18:49:01	
4	I just won a game in Empire Earth!	6	2021-02-21 18:51:20	
...				

upvote		
postID	userID	
1	3	
1	4	
2	4	
3	4	
1	5	
3	5	
4	5	
1	7	
...		

comment				
commentID	postID	userID	text	dateTime
1	1	7	Hello SONEM	2021-02-21 18:48:40
2	3	4	No	2021-02-21 18:49:19
3	2	4	I WANNA PLAY TOO SEND ME A DM	2021-02-21 18:49:31
4	1	4	Hello Brother	2021-02-21 18:50:11
5	1	3	Hey brooo	2021-02-21 18:50:34
6	4	5	Good job Dude!!	2021-02-21 18:51:47
7	3	5	LETS GOO	2021-02-21 18:52:30
...				

deletePost(1)

post				
postID	content	userID	dateTime	
2	Playing minecraft!	3	2021-02-21 18:47:51	
3	Anyone who wants to playing tetris	7	2021-02-21 18:49:01	
4	I just won a game in Empire Earth!	6	2021-02-21 18:51:20	
...				

upvote		
postID	userID	
2	4	
3	4	
3	5	
4	5	
...		

comment				
commentID	postID	userID	text	dateTime
2	3	4	No	2021-02-21 18:49:19
3	2	4	I WANNA PLAY TOO SEND ME A DM	2021-02-21 18:49:31
6	4	5	Good job Dude!!	2021-02-21 18:51:47
7	3	5	LETS GOO	2021-02-21 18:52:30
...				

Εικόνα 36: Βάση Δεδομένων πριν και μετά την διαγραφή Δημοσίευσης

3.4.3. Admin

ShowAllUsers.jsp

Εντός του παρακάτω μονοπατιού, εμφανίζεται η σελίδα ShowAllUsers.jsp:

GameSociety/WebContent/Pages/HomePage/Admin

Η σελίδα αυτή εμφανίζεται μόνο για τους Διαχειριστές του Πληροφοριακού Συστήματος και υπάρχει ώστε να μπορούν να δουν όλους τους Χρήστες μαζεμένους σε ένα σημείο, με σκοπό την γρηγορότερη πρόσβαση τους σε αυτούς. Αυτή η λειτουργία, πραγματοποιείται μέσω του Web Service **getAllUsers** το οποίο επιστρέφει στοιχεία για όλους τους Χρήστες του Συστήματος, η κλήση του φαίνεται στην Εικόνα 37.

```

35<%
36    if(session.getAttribute("adminID") != null){
37        client = Client.create();
38        link = "http://localhost:8080/GameSociety/rest/GameSociety/getAllUsers/";
39        webResource = client.resource(link);
40        myresponse = webResource.accept("application/json").get(ClientResponse.class);
41        allUsers = new JSONArray(myresponse.getEntity(String.class));
42    }

```

Εικόνα 37: Κλήση του getAllUsers

3.4.4. MyGames

MyGames.jsp

Εντός του παρακάτω μονοπατιού, εμφανίζεται η σελίδα MyGames.jsp

GameSociety/WebContent/Pages/HomePage/MyGames

Σε αυτή την σελίδα οι Χρήστες μπορούν να χρησιμοποιήσουν τα Web Services **deleteAUserGame**, **addAUserGame** και **getAllUserGames** για την διαχείριση της Λίστας με τους τίτλους των παιχνιδιών τους. Οι κλήσεις των Services αυτών φαίνεται στην Εικόνα 38.

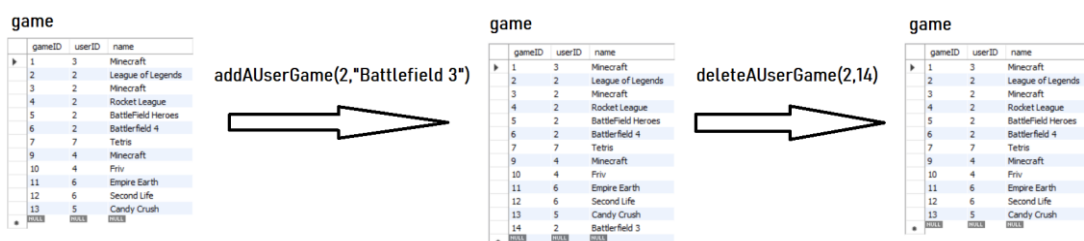
```

36<%
37    if(request.getParameter("deleteGame")!=null){
38        int gameId = Integer.parseInt(request.getParameter("gameID"));
39        client = Client.create();
40        link = "http://localhost:8080/GameSociety/rest/GameSociety/deleteAUserGame/";
41        link+=userID+"/";
42        link+=gameID;
43        webResource = client.resource(link);
44        myresponse = webResource.delete(ClientResponse.class);
45    }
46
47
48    if(request.getParameter("addGame")!=null){
49        String name = request.getParameter("name");
50        if(name.equals("")){
51            error = "You must fill the field";
52        }
53        else{
54            client = Client.create();
55            link = "http://localhost:8080/GameSociety/rest/GameSociety/addAUserGame/";
56            link+=userID+"/";
57            link+=name.replace(' ', '_');
58            webResource = client.resource(link);
59            myresponse = webResource.post(ClientResponse.class);
60        }
61    }
62
63    if(session.getAttribute("userID") != null){
64        client = Client.create();
65        link = "http://localhost:8080/GameSociety/rest/GameSociety/getAllUserGames/";
66        link+=userID;
67        webResource = client.resource(link);
68        myresponse = webResource.accept("application/json").get(ClientResponse.class);
69        UserGames = new JSONArray(myresponse.getEntity(String.class));
70    }
71 %>

```

Εικόνα 38: Τα Services της σελίδας Games I Play

Στην Εικόνα 39 φαίνεται η εισαγωγή ενός παιχνιδιού με όνομα «Battlefield 3» στον χρήστη με ID τον αριθμό 2 χρησιμοποιώντας το **addAUserGame** και στην συνέχεια η διαγραφή του με χρήση του **deleteAUserGame**.



Εικόνα 39: Εισαγωγή και Διαγραφή ενός παιχνιδιού

3.4.5. Post

Εντός του παρακάτω μονοπατιού,

GameSociety/WebContent/Pages/HomePage/Post

εμφανίζονται οι εξής σελίδες:

- Post.jsp
- Likes.jsp

Σε αυτές τις σελίδες εμφανίζονται όλες οι πληροφορίες για όλες τις Δημοσιεύσεις των Χρηστών του Συστήματος.

Post.jsp

Στην σελίδα Post.jsp οι Χρήστες μπορούν:

- Να δουν μία Δημοσίευση
- Να διαγράψουν την Δημοσίευση αν έχουν τα απαραίτητα δικαιώματα
- Να δουν αν έχουν κάνει οι ίδιοι Like στην εμφανιζόμενη Δημοσίευση
- Να δουν πόσοι έχουν κάνει Like στην εμφανιζόμενη Δημοσίευση
- Να δουν όλα τα σχόλια της Δημοσίευσης
- Να προσθέσουν σχόλια στην υπάρχουσα Δημοσίευση
- Να διαγράψουν σχόλια αν έχουν τα απαραίτητα δικαιώματα

Στην περίπτωση που προχωρήσει ο Χρήστης σε διαγραφή της Δημοσίευσης, σχολίου ή Like, γίνονται παρόμοιες τροποποιήσεις στην Βάση Δεδομένων όπως δείχνονται και στην Εικόνα 36 χρησιμοποιώντας τα Services **deletePost** (Εικόνα 40), **deleteComment** (Εικόνα 41), **likeOrDislike** (Εικόνα 42) αντίστοιχα.

```
66     if(request.getParameter("deletePost") != null){
67         int postID = Integer.parseInt(request.getParameter("postID"));
68         client = Client.create();
69         String link = "http://localhost:8080/GameSociety/rest/GameSociety/deletePost/";
70         link+=postID;
71         webResource = client.resource(link);
72         myresponse = webResource.delete(ClientResponse.class);
73     }
```

Εικόνα 40: Διαγραφή Δημοσίευσης από την σελίδα της

```
75     if(request.getParameter("deleteComment") != null){
76         int postID = Integer.parseInt(request.getParameter("postID"));
77         int commentID = Integer.parseInt(request.getParameter("commentID"));
78         client = Client.create();
79         String link = "http://localhost:8080/GameSociety/rest/GameSociety/deleteComment/";
80         link+=commentID;
81         webResource = client.resource(link);
82         myresponse = webResource.delete(ClientResponse.class);
83         session.setAttribute("postViewingID",postID);
84     }
```

Εικόνα 41: Διαγραφή ενός Σχολίου

```

55     if(request.getParameter("likeDislike") != null){
56         int postID = Integer.parseInt(request.getParameter("likeDislike"));
57         client = Client.create();
58         String link = "http://localhost:8080/GameSociety/rest/GameSociety/likeOrDislike/";
59         link+=userID+"/";
60         link+=postID;
61         webResource = client.resource(link);
62         myresponse = webResource.post(ClientResponse.class);
63         session.setAttribute("postViewingID",postID);
64     }

```

Εικόνα 42: Προσθήκη και Διαγραφή Like

Όταν προσθέτει Like ο χρήστης, τότε καλείτε το ίδιο Service με την Εικόνα 42. Για την προσθήκη ενός Σχολίου χρησιμοποιούνται οι γραμμές κώδικα που φαίνονται στην Εικόνα 43 με την χρήση του **addAComment**. Στην εικόνα 44 φαίνεται η Βάση Δεδομένων πριν και μετά την προσθήκη ενός Σχολίου.

```

94     if(request.getParameter("addAComment") != null){
95         int postID = Integer.parseInt(request.getParameter("postID"));
96         String content = request.getParameter("content");
97         if(content.equals("")){
98             error = "You must fill the field!";
99         }
100         String newline = System.getProperty("line.separator");
101         boolean hasNewline = content.contains(newline);
102         if(hasNewline){
103             error = "You may not use enter!";
104         }
105         else{
106             content = content.replace(" ", "_");
107             DateTimeFormatter dtf = DateTimeFormatter.ofPattern("yyyy-MM-dd HH:mm:ss");
108             LocalDateTime now = LocalDateTime.now();
109             String dateTime = dtf.format(now);
110             dateTime = dateTime.replace(" ", "_");
111             client = Client.create();
112             String link = "http://localhost:8080/GameSociety/rest/GameSociety/addAComment/";
113             link+=userID+"/";
114             link+=postID+"/";
115             link+=content+"/";
116             link+=dateTime;
117             webResource = client.resource(link);
118             myresponse = webResource.post(ClientResponse.class);
119         }
120         session.setAttribute("postViewingID",postID);
121     }

```

Εικόνα 43: Προσθήκη Σχολίου

comment

	commentID	postID	userID	text	dateTime
▶	2	3	4	No	2021-02-21 18:49:19
	3	2	4	I WANNA PLAY TOO SEND ME A DM	2021-02-21 18:49:31
	6	4	5	Good job Dude!!	2021-02-21 18:51:47
	7	3	5	LETS GOO	2021-02-21 18:52:30
*	NULL	NULL	NULL	NULL	NULL

comment

	commentID	postID	userID	text	dateTime
▶	2	3	4	No	2021-02-21 18:49:19
	3	2	4	I WANNA PLAY TOO SEND ME A DM	2021-02-21 18:49:31
	6	4	5	Good job Dude!!	2021-02-21 18:51:47
	7	3	5	LETS GOO	2021-02-21 18:52:30
	8	3	2	Lol no	2021-02-21 21:08:31
*	NULL	NULL	NULL	NULL	NULL

addAComment(2,3,Lol no, 2021-02-21 21:08:31)



Εικόνα 44: Προσθήκη Σχολίου στην Βάση Δεδομένων

Χρησιμοποιείται το **getHomeScreenPosts** για να βρεθεί εντός των αποτελεσμάτων του η Δημοσίευση που πρέπει να προβληθεί στον Χρήστη (Εικόνα 45).

Χρησιμοποιείται τα Web Services **showComments** και **getAllUsersLiked** για την προβολή όλων των σχολίων αλλά και για την εύρεση του αν ο Χρήστης που έχει κάνει σύνδεση έχει πατήσει Like στην συγκεκριμένη Δημοσίευση (Εικόνα 46).

Όπως επίσης χρησιμοποιείτε η **getAmountOfLikes** ώστε να προβληθεί στον Χρήστη το άθροισμα όλων των Like της Δημοσίευσης (Εικόνα 47).

```
128 myPostID = (int)session.getAttribute("postViewingID");
129 session.removeAttribute("postViewingID");
130 client = Client.create();
131 String link = "http://localhost:8080/GameSociety/rest/GameSociety/getHomeScreenPosts/";
132 webResource = client.resource(link);
133 myresponse = webResource.accept("application/json").get(ClientResponse.class);
134 allPosts = new JSONArray(myresponse.getEntity(String.class));
135 for(int i=0; i<allPosts.length(); i++){
136     JSONObject jsonObject = allPosts.getJSONObject(i);
137     if(jsonObject.getInt("postID")==myPostID){
138         myPost = allPosts.getJSONObject(i);
139         break;
140     }
141 }
142 myUserID = myPost.getInt("userID");
143 myProfilePicturePath = myPost.getString("profilePicturePath");
144 myContent = myPost.getString("content");
145 myDateTime = myPost.getString("dateTime");
146 myNickName = myPost.getString("nickName");
```

Εικόνα 45: Εύρεση της ως προς προβολή Δημοσίευσης

```
148 client = Client.create();
149 link = "http://localhost:8080/GameSociety/rest/GameSociety/showComments/";
150 link+=myPostID;
151 webResource = client.resource(link);
152 myresponse = webResource.accept("application/json").get(ClientResponse.class);
153 allComments = new JSONArray(myresponse.getEntity(String.class));
154
155 client = Client.create();
156 link = "http://localhost:8080/GameSociety/rest/GameSociety/getAllUsersLiked/";
157 link+=myPostID;
158 webResource = client.resource(link);
159 myresponse = webResource.accept("application/json").get(ClientResponse.class);
160 allLikes = new JSONArray(myresponse.getEntity(String.class));
161
162 for(int i=0; i<allLikes.length(); i++){
163     JSONObject jsonObject = allLikes.getJSONObject(i);
164     if(jsonObject.getInt("userID")==userID){
165         userLikesThisPost = true;
166         break;
167     }
168 }
```

Εικόνα 46: Απόκτηση Σχολίων και ανίχνευση Like του Χρήστη

```
170 client = Client.create();
171 link = "http://localhost:8080/GameSociety/rest/GameSociety/getAmountOfLikes/";
172 link+=myPostID;
173 webResource = client.resource(link);
174 myresponse = webResource.accept("application/json").get(ClientResponse.class);
175 JSONObject jsonObject = new JSONObject(myresponse.getEntity(String.class));
176 amountOfLikes = jsonObject.getInt("amount");
```

Εικόνα 47: Απόκτηση πλήθους Like μίας Δημοσίευσης

Likes.jsp

Η σελίδα αυτή, χρησιμοποιεί το Web Service **getAllUsersLiked** ώστε να εμφανίσει όλους τους Χρήστες που έχουν κάνει Like στην εμφανιζόμενη Δημοσίευση της σελίδας Post.jsp. Η κλήση της γίνεται στις γραμμές που φαίνονται στην Εικόνα 48.

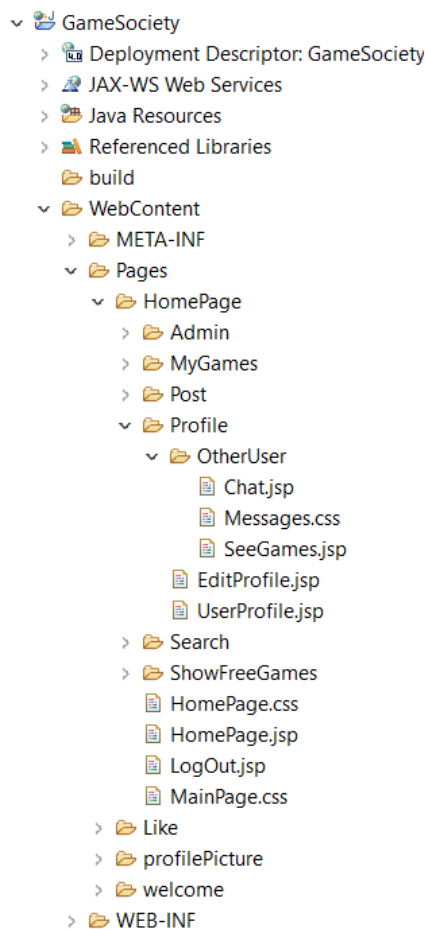
```
39     if(session.getAttribute("viewingLikesOfPostID") != null){
40         postID = (int)session.getAttribute("viewingLikesOfPostID");
41         session.removeAttribute("viewingLikesOfPostID");
42         client = Client.create();
43         link = "http://localhost:8080/GameSociety/rest/GameSociety/getAllUsersLiked/";
44         link+=postID;
45         webResource = client.resource(link);
46         myresponse = webResource.accept("application/json").get(ClientResponse.class);
47         allLikes = new JSONArray(myresponse.getEntity(String.class));
48     }
49     else if(request.getParameter("GoToPost")==null){
50         response.sendRedirect("../welcome/login.jsp");
51     }
52 %>
```

Εικόνα 48: Κλήση όλων των Χρηστών που έχουν κάνει Like σε μία Δημοσίευση

3.4.6. Profile

Στο παρακάτω μονοπάτι φαίνονται οι σελίδες οι οποίες σχετίζονται με τις σελίδες των προφίλ των Χρηστών, το οποίο φαίνεται και στην Εικόνα 49:

GameSociety/WebContent/Pages/HomePage/Profile



Εικόνα 49: Σελίδες των Προφίλ

UserProfile.jsp

Από το UserProfile.jsp μπορεί ένας χρήστης να διαγράψει τα Δημοσιεύσεις – αν έχει τα ανάλογα δικαιώματα - όπως έχει εξηγηθεί και στο κεφάλαιο 3.4.2. χρησιμοποιώντας το Web Service **deletePost** (Εικόνα 50)

```
134     if(request.getParameter("deletePost")!=null){
135         int postID = Integer.parseInt(request.getParameter("postID"));
136         client = Client.create();
137         link = "http://localhost:8080/GameSociety/rest/GameSociety/deletePost/";
138         link+=postID;
139         webResource = client.resource(link);
140         myresponse = webResource.delete(ClientResponse.class);
141
142         if(request.getParameter("profileUserID")!=null){
143             int profileUser = Integer.parseInt(request.getParameter("profileUserID"));
144             session.setAttribute("profileUserID",profileUser);
145             response.sendRedirect("UserProfile.jsp");
146         }
147     }
```

Εικόνα 50: Διαγραφή μίας Δημοσίευσης μέσω ενός προφίλ

Οι Διαχειριστές, μέσω του κώδικα που φαίνεται στην Εικόνα 51, μπορούν να διαγράψουν τον χρήστη του προφίλ, κάνοντας χρήση του **deleteAUser**. Στην Εικόνα 52, φαίνεται μία διαγραφή ενός χρήστη από την Βάση (με ID τον αριθμό 3). Κατά την διαγραφή ενός χρήστη, γίνεται διαγραφή όλων των Δημοσιεύσεων του, όλων των μηνυμάτων του, όλων των παιχνιδιών που έχει προσθέσει όπως και όλα τα Likes και τα σχόλιά του.

```
70     if(request.getParameter("deleteUser")!=null){
71         int userIDToDelete = Integer.parseInt(request.getParameter("profileUserID"));
72         client = Client.create();
73         link = "http://localhost:8080/GameSociety/rest/GameSociety/deleteAUser/";
74         link+=userIDToDelete;
75         webResource = client.resource(link);
76         myresponse = webResource.delete(ClientResponse.class);
77         response.sendRedirect("../HomePage.jsp");
78     }
```

Εικόνα 51: Διαγραφή ενός Χρήστη μέσω του Προφίλ του

userID	name	surname	profilePicturePath	nickName	isAdmin	password	email
1	admin	admin	11	admin	1	admin	admin@admin.com
2	Panagiotis	Karamolegkos	3	SONEM	0	12345	p.karamolegkos@yahoo.gr
3	Anastasia	Karamolegkos	9	NastkaEz	0	12345	nastka@gmail.com
4	Peni	Mpoupoul	7	FreeQueen	0	12345	peni@gmail.com
5	Paraskevas	Mpoupoulos	2	Rikos	0	12345	paris@gmail.com
6	Manolis	Karamolegkos	4	Lino	0	12345	manolis@gmail.com
7	Stavroula	Tsekoura	1	Roro	0	12345	roula@gmail.com
8	Gouli	Katznova	1	Goula	0	12345	goula@gmail.com

postID	content	userID	dateTime
2	Playing minecraft	3	2021-02-21 18:47:51
3	Anyone who wants to playing tetris	7	2021-02-21 18:49:01
4	I just won a game in Empire Earth!	6	2021-02-21 18:51:20

commentID	postID	userID	text	dateTime
2	3	4	No	2021-02-21 18:49:19
3	2	4	I WANNA PLAY TOO SEND ME A DM	2021-02-21 18:49:31
6	4	5	Good job Dude!!	2021-02-21 18:51:47
7	3	5	LET'S GOO	2021-02-21 18:52:30
8	3	2	Lol no	2021-02-21 21:08:31

gameID	userID	name
1	3	Minecraft
2	2	League of Legends
3	2	Minecraft
4	2	Rocket League
5	2	Battlefield Heroes
6	2	Battlefield 4
7	7	Tetris
9	4	Minecraft
10	4	Priv
11	6	Empire Earth
12	6	Second Life
13	5	Candy Crush

messageID	theUserID	friendUserID	content	dateTime
1	3	6	Hellooo	2021-02-21 21:50:47
2	6	3	Oh hey there!	2021-02-21 21:51:05

postID	userID
2	4
3	4
3	5
4	5

gameID	userID	name
2	2	League of Legends
3	2	Minecraft
4	2	Rocket League
5	2	Battlefield Heroes
6	2	Battlefield 4
7	7	Tetris
9	4	Minecraft
10	4	Priv
11	6	Empire Earth
12	6	Second Life
13	5	Candy Crush

Εικόνα 52: Διαγραφή ενός Χρήστη από την Βάση Δεδομένων

Ακόμα σε αυτήν την σελίδα χρησιμοποιείτε και το Web Service **changePrivileges** ώστε να αλλάξουν τα δικαιώματα ενός Χρήστη, από Διαχειριστή σε απλό Χρήστη και το αντίθετο. Η χρήση του φαίνεται στην Εικόνα 53 και η αλλαγή που προκαλεί στην Βάση Δεδομένων φαίνεται στην Εικόνα 54, χρησιμοποιώντας τον Χρήστη με ID τον αριθμό 2.

```

96     if(request.getParameter("changePrivileges")!=null){
97         int profileUser = Integer.parseInt(request.getParameter("profileUserID"));
98         client = Client.create();
99         link = "http://localhost:8080/GameSociety/rest/GameSociety/changePrivileges/";
100        link+=profileUser;
101        webResource = client.resource(link);
102        myresponse = webResource.put(ClientResponse.class);
103        session.setAttribute("profileUserID",profileUser);
104        response.sendRedirect("UserProfile.jsp");
105    }

```

Εικόνα 53: Κώδικας για αλλαγή δικαιωμάτων

user

	userID	name	surname	profilePicturePath	nickName	isAdmin	password	email
▶	1	admin	admin	11	admin	1	admin	admin@admin.com
	2	Panagiotis	Karamolegkos	3	SONEM	0	12345	p.karamolegos@yahoo.gr
	4	Peni	Mpoupouli	7	FireQueen	0	12345	peni@gmail.com
	5	Paraskevas	Mpoupoulis	2	Rikos	0	12345	paris@gmail.com
	6	Manolis	Karamolegkos	4	Lino	0	12345	manolis@gmail.com
	7	Stavroula	Tsekoura	1	Roro	0	12345	roula@gmail.com
	8	Giouli	Xatzinova	1	Goulla	0	12345	gioula@gmail.com
•	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL



changePrivileges(2)

user

	userID	name	surname	profilePicturePath	nickName	isAdmin	password	email
▶	1	admin	admin	11	admin	1	admin	admin@admin.com
	2	Panagiotis	Karamolegkos	3	SONEM	1	12345	p.karamolegos@yahoo.gr
	4	Peni	Mpoupouli	7	FireQueen	0	12345	peni@gmail.com
	5	Paraskevas	Mpoupoulis	2	Rikos	0	12345	paris@gmail.com
	6	Manolis	Karamolegkos	4	Lino	0	12345	manolis@gmail.com
	7	Stavroula	Tsekoura	1	Roro	0	12345	roula@gmail.com
	8	Giouli	Xatzinova	1	Goulla	0	12345	gioula@gmail.com
•	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL



changePrivileges(2)

user

	userID	name	surname	profilePicturePath	nickName	isAdmin	password	email
▶	1	admin	admin	11	admin	1	admin	admin@admin.com
	2	Panagiotis	Karamolegkos	3	SONEM	0	12345	p.karamolegos@yahoo.gr
	4	Peni	Mpoupouli	7	FireQueen	0	12345	peni@gmail.com
	5	Paraskevas	Mpoupoulis	2	Rikos	0	12345	paris@gmail.com
	6	Manolis	Karamolegkos	4	Lino	0	12345	manolis@gmail.com
	7	Stavroula	Tsekoura	1	Roro	0	12345	roula@gmail.com
	8	Giouli	Xatzinova	1	Goulla	0	12345	gioula@gmail.com
•	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

Εικόνα 54: Αλλαγή δικαιωμάτων Χρήστη στην Βάση

Για την εισαγωγή Δημοσιεύσεων από τους Χρήστες του Συστήματος, χρησιμοποιείτε ο κώδικας στην Εικόνα 55 κάνοντας την χρήση του Service **addAPost**. Ένα παράδειγμα αυτής της χρήσης στην Βάση Δεδομένων φαίνεται στην Εικόνα 56.


```

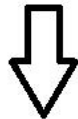
119         content = content.replace(" ", "_");
120         DateTimeFormatter dtf = DateTimeFormatter.ofPattern("yyyy-MM-dd HH:mm:ss");
121         LocalDateTime now = LocalDateTime.now();
122         String dateTime = dtf.format(now);
123         dateTime = dateTime.replace(" ", "_");
124         client = Client.create();
125         link = "http://localhost:8080/GameSociety/rest/GameSociety/addAPost/";
126         link+=userID+"/";
127         link+=content+"/";
128         link+=dateTime;
129         webResource = client.resource(link);
130         myresponse = webResource.post(ClientResponse.class);

```

Εικόνα 55: Κώδικας εισαγωγής Δημοσίευσης

post

	postID	content	userID	dateTime
▶	3	Anyone who wants to playing tetris	7	2021-02-21 18:49:01
	4	I just won a game in Empire Earth!	6	2021-02-21 18:51:20
*	NULL	NULL	NULL	NULL



post

addAPost(2,"Hello World","2021-02-21 22:18:15")

	postID	content	userID	dateTime
▶	3	Anyone who wants to playing tetris	7	2021-02-21 18:49:01
	4	I just won a game in Empire Earth!	6	2021-02-21 18:51:20
	5	Hello World	2	2021-02-21 22:18:15
*	NULL	NULL	NULL	NULL

Εικόνα 56: Εισαγωγή Δημοσίευσης στην Βάση Δεδομένων

EditProfile.jsp

Στην σελίδα EditProfile.jsp χρησιμοποιεί με παρόμοιο τρόπο το **deleteUser** όπως και στο UserProfile.jsp. Εκτός αυτού, η σελίδα αυτή χρησιμοποιεί διαδοχικά την **changeProfileInfo** ώστε να αλλάξει όποια στοιχεία θέλει ο εκάστοτε Χρήστης. Η λειτουργία αυτή απλά αλλάζει στοιχεία στην Βάση.

```

73     if(request.getParameter("changePassword")!=null){
74         String newPassword = request.getParameter("password");
75         if(newPassword.equals("")){
76             passwordError = "You must fill the field!";
77         }
78         else if(newPassword.contains(" ")){
79             passwordError = "You may not use spaces!";
80         }
81         else{
82             client = Client.create();
83             link = "http://localhost:8080/GameSociety/rest/GameSociety/changeProfileInfo/";
84             link+=userID+"/";
85             link+=name+"/";
86             link+=surname+"/";
87             link+=profilePicturePath+"/";
88             link+=nickName+"/";
89             link+=newPassword+"/";
90             link+=email;
91             webResource = client.resource(link);
92             myresponse = webResource.put(ClientResponse.class);
93             passwordError = "The update is done!";
94         }
95     }

```

Εικόνα 57: Κώδικας αλλαγής στοιχείων Χρήστη

Chat.jsp

Η σελίδα εντός του φακέλου OtherUser με όνομα Chat.jsp υπάρχει για την επικοινωνία μεταξύ των Χρηστών. Γίνεται Χρήση του **showMessages** ώστε να εμφανιστούν τα μηνύματα της συζήτησης ανάμεσα στους εκάστοτε δύο Χρήστες (Εικόνα 58).

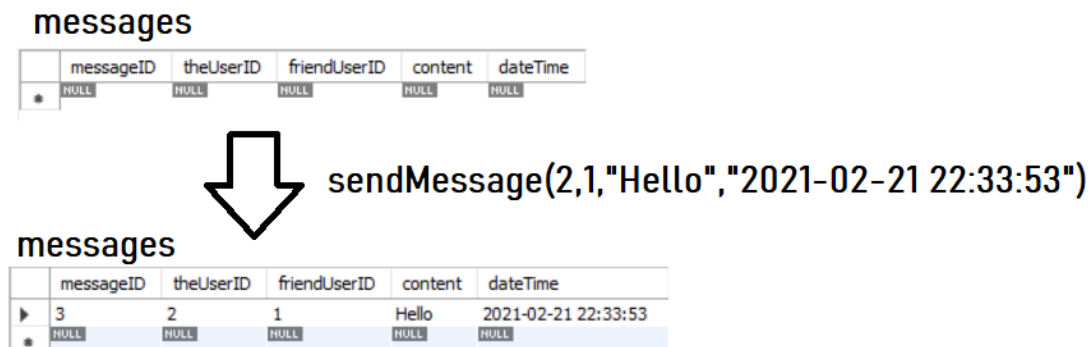
```
92      client = Client.create();
93      link = "http://localhost:8080/GameSociety/rest/GameSociety/showMessages/";
94      link+=userID+"/";
95      link+=profileUserID;
96      webResource = client.resource(link);
97      myresponse = webResource.accept("application/json").get(ClientResponse.class);
98      messages = new JSONArray(myresponse.getEntity(String.class));
```

Εικόνα 58: Κώδικας απόκτησης μηνυμάτων

Ακόμα, χρησιμοποιείται το Web Service **sendMessage** (Εικόνα 59) για την αποστολή των μηνυμάτων ανάμεσα στους δύο Χρήστες. Ένα παράδειγμα στην Βάση Δεδομένων είναι στην Εικόνα 60 όπου ο Χρήστης με ID το νούμερο 2, στέλνει το μήνυμα «Hello» στον Χρήστη με ID το νούμερο 3.

```
50      content = content.replace(" ", "_");
51      DateTimeFormatter dtf = DateTimeFormatter.ofPattern("yyyy-MM-dd HH:mm:ss");
52      LocalDateTime now = LocalDateTime.now();
53      String dateTime = dtf.format(now);
54      dateTime = dateTime.replace(" ", "_");
55      client = Client.create();
56      link = "http://localhost:8080/GameSociety/rest/GameSociety/sendAMessage/";
57      link+=userID+"/";
58      link+=profileUser+"/";
59      link+=content+"/";
60      link+=dateTime;
61      webResource = client.resource(link);
62      myresponse = webResource.post(ClientResponse.class);
```

Εικόνα 59: Web Service για την αποστολή μηνυμάτων



Εικόνα 60: Αποστολή μηνύματος ανάμεσα σε δύο Χρήστες

SeeGames.jsp

Όπως και στην σελίδα για το tab «Games I Play» γίνεται παρόμοια χρήση του **getAllUserGames** και για την εμφάνιση όλων των παιχνιδιών ενός άλλου Χρήστη (Εικόνα 61).

```
59      client = Client.create();
60      link = "http://localhost:8080/GameSociety/rest/GameSociety/getAllUserGames/";
61      link+=profileUserID;
62      webResource = client.resource(link);
63      myresponse = webResource.accept("application/json").get(ClientResponse.class);
64      userGames = new JSONArray(myresponse.getEntity(String.class));
```

Εικόνα 61: Web Service για την απόκτηση των παιχνιδιών ενός Χρήστη

3.4.7. Search

Εντός του παρακάτω μονοπατιού, εντοπίζουμε το αρχείο SearchSelection.jsp:

GameSociety/WebContent/Pages/HomePage/Search/SearchSelection.jsp

Σε αυτό το αρχείο γίνεται η εμφάνιση του tab «Search Users» όπου γίνονται αναζητήσεις για Χρήστες μέσω των παιχνιδιών που έχουν δηλώσει ότι παίζουν, όπως και του ψευδώνυμου τους.

Στην Εικόνα 62 φαίνεται η χρήση του **getUsersByNickName** ώστε να γίνει η εμφάνιση των χρηστών που έχουν ψευδώνυμο το οποίο μοιάζει με αυτό που έχει δώσει ως είσοδο ο Χρήστης.

Παρόμοια χρήση γίνεται και για το **getUsersPlayingTheGame** (Εικόνα 63) ώστε να εμφανιστούν οι Χρήστες που παίζουν το παιχνίδι που δίνει ως είσοδο ο Χρήστης.

```
48      client = Client.create();
49      link = "http://localhost:8080/GameSociety/rest/GameSociety/getUsersByNickName/";
50      link+=searchedNickname;
51      webResource = client.resource(link);
52      myresponse = webResource.accept("application/json").get(ClientResponse.class);
53      usersByNickName = new JSONArray(myresponse.getEntity(String.class));
```

Εικόνα 62: Απόκτηση Χρηστών Ψευδώνυμο παρόμοιο με την είσοδο

```
63      client = Client.create();
64      link = "http://localhost:8080/GameSociety/rest/GameSociety/getUsersPlayingTheGame/";
65      link+=searchedGame.replace(' ','_');
66      webResource = client.resource(link);
67      myresponse = webResource.accept("application/json").get(ClientResponse.class);
68      usersByGame = new JSONArray(myresponse.getEntity(String.class));
```

Εικόνα 63: Απόκτηση Χρηστών που παίζουν κάποιο δοθέν παιχνίδι

3.4.8. ShowFreeGames

Εντός του παρακάτω μονοπατιού, εντοπίζουμε το αρχείο FreeGamesSelection.jsp:

GameSociety/WebContent/Pages/HomePage/ShowFreeGames/
FreeGamesSelection.jsp

Σε αυτό το αρχείο γίνεται η εμφάνιση του tab «Free Games» όπου γίνονται αναζητήσεις για δωρεάν παιχνίδια τα οποία υπάρχουν στο διαδίκτυο. Τα Web Services τα οποία χρησιμοποιούνται είναι Third Party Web Services. Η χρήση τους φαίνεται στην Εικόνα 64.

```
37<%
38    if(request.getParameter("allFreeGames")!=null){
39        client = Client.create();
40        String link = "https://www.freetogame.com/api/games";
41        webResource = client.resource(link);
42        myresponse = webResource.accept("application/json").get(ClientResponse.class);
43        allFreeGames = new JSONArray(myresponse.getEntity(String.class));
44    }
45
46    if(request.getParameter("platformSearch")!=null){
47        String platform = request.getParameter("platform");
48        if(platform.equals("")){
49            error = "You must fill the field!";
50        }
51        else if(platform.contains(" ")){
52            error = "You must not use spaces!";
53        }
54        else{
55            client = Client.create();
56            String link = "https://www.freetogame.com/api/games?platform=";
57            link+=platform;
58            webResource = client.resource(link);
59            myresponse = webResource.accept("application/json").get(ClientResponse.class);
60            try{
61                platformFreeGames = new JSONArray(myresponse.getEntity(String.class));
62            }
63            catch(JSONException ex){
64
65            }
66        }
67    }
68%>
```

Εικόνα 64: Third Party Web Services

Στις Εικόνες 65, 66 και 67 φαίνονται τρεις επιστροφές αυτών των Services μέσω του εργαλείου Postman.

GET https://www.freetogame.com/api/games Send

Params Authorization Headers (6) Body Pre-request Script Tests Settings

Body Cookies Headers (25) Test Results Status: 200 OK Time: 222 ms Size: 161.59 KB Save

Pretty Raw Preview Visualize JSON

```
1 {
2   {
3     "id": 1,
4     "title": "Dauntless",
5     "thumbnail": "https://www.freetogame.com/g/1/thumbnail.jpg",
6     "short_description": "A free-to-play, co-op action RPG with gameplay similar to Monster Hunter.",
7     "game_url": "https://www.freetogame.com/open/dauntless",
8     "genre": "MMORPG",
9     "platform": "PC (Windows)",
10    "publisher": "Phoenix Labs",
11    "developer": "Phoenix Labs, Iron Galaxy",
12    "release_date": "2019-05-21",
13    "freetogame_profile_url": "https://www.freetogame.com/dauntless"
14  },
15  {
16    "id": 2,
17    "title": "World of Tanks",
18    "thumbnail": "https://www.freetogame.com/g/2/thumbnail.jpg",
19    "short_description": "If you like blowing up tanks, with a quick and intense game style you will love this game!",
20    "game_url": "https://www.freetogame.com/open/world-of-tanks",
21    "genre": "Shooter",
22    "platform": "PC (Windows)",
23    "publisher": "Wargaming",
24    "developer": "Wargaming",
25    "release_date": "2011-04-12",
```

Εικόνα 65: Σωστή Χρήση του Free To Game

GET https://www.freetogame.com/api/games?platform=pc Send

Params Authorization Headers (6) Body Pre-request Script Tests Settings

Body Cookies Headers (25) Test Results Status: 200 OK Time: 602 ms Size: 130.38 KB Save

Pretty Raw Preview Visualize JSON

```
1 {
2   {
3     "id": 1,
4     "title": "Dauntless",
5     "thumbnail": "https://www.freetogame.com/g/1/thumbnail.jpg",
6     "short_description": "A free-to-play, co-op action RPG with gameplay similar to Monster Hunter.",
7     "game_url": "https://www.freetogame.com/open/dauntless",
8     "genre": "MMORPG",
9     "platform": "PC (Windows)",
10    "publisher": "Phoenix Labs",
11    "developer": "Phoenix Labs, Iron Galaxy",
12    "release_date": "2019-05-21",
13    "freetogame_profile_url": "https://www.freetogame.com/dauntless"
14  },
15  {
16    "id": 2,
17    "title": "World of Tanks",
18    "thumbnail": "https://www.freetogame.com/g/2/thumbnail.jpg",
19    "short_description": "If you like blowing up tanks, with a quick and intense game style you will love this game!",
20    "game_url": "https://www.freetogame.com/open/world-of-tanks",
21    "genre": "Shooter",
22    "platform": "PC (Windows)",
23    "publisher": "Wargaming",
24    "developer": "Wargaming",
25    "release_date": "2011-04-12",
26    "freetogame_profile_url": "https://www.freetogame.com/world-of-tanks"
```

Εικόνα 66: Σωστή Χρήση του Free To Game For Platforms

GET https://www.freetogame.com/api/games?platform=wrongPlatform Send

Params Authorization Headers (6) Body Pre-request Script Tests Settings

Body Cookies Headers (25) Test Results Status: 404 Not Found Time: 127 ms Size: 1.19 KB Save

Pretty Raw Preview Visualize JSON

```
1 {
2   "status": 0,
3   "status_message": "No platform found, please check the correct parameters."
4 }
```

Εικόνα 67: Λάθος Χρήση του Free To Game For Platforms